İSTANBUL TECHNICAL UNIVERSITY
ELECTRICAL – ELECTRONICS ENGINEERING FACULTY

An Indoor Localization Algorithm On Fpga

BSc Thesis

Alp ORAN

040100520

Department: Electronics and Communication Engineering

Programme: Electronics and Communication Engineering

Supervisor: Assis. Prof. Dr. Sıddıka Berna ÖRS YALÇIN

MAY 2015

ACKNOWLEDGMENT

May 2015 Alp ORAN

INDEX

ABBREVIATIONS

RFID **:** Radio Frequency Identification
RTLS **:** Real Time Localization System
FPGA **:** Field Programmable Gate Array
ISE **:** Integrated Synthesis Environment
XPS **:** Xilinx Platform Studio
EDK **:** Embedded Development Kit
SDK **:** Software Development Kit
LUT **:** Look-up Table
XOR **:** Exclusive Or
SDES **:** Simplified Data Encryption Standard
BaaS **:** Building as a Service
LANDMARC **:** LocAlizatioN iDentification based on dynaMic Active Rfid Calibration
RSSI **:** Received Signal Strength Indicator

LIST OF FIGURES

IMPLEMENTATION OF A SECURE RFID BASED INDOOR LOCALIZATION SYSYEM ON FPGA

SUMMARY

In recent years, numerous developments related with the wireless communication and hence the mobility of electronic devices have been made. With these developments, location-sensing systems have been researched extensively due to increasing demand for accurate positioning of people and objects. In this sense, Radio Frequency Identification (RFID) is one of the most widely used wireless technology for automatic identification and data capture applications. A RFID module employs electromagnetic waves, and communicate other modules with same working frequency within the range.

In this study, a RFID based indoor localization algorithm and a RFID mutual authentication protocol is implemented by hardware software co-design techniques. Firstly, background knowledge is given related with used tools and software, as well as the important concepts regarding indoor localization. After that part, implemented algorithms and protocols is presented for a better comprehension about the topic.

In the implementation Chapters, hardware implementation of LANDMARC indoor localization algorithm and the software implementation of Molnar-Wagner Mutual Authentication Protocol is presented in details, respectively. Verilog hardware description language and Xilinx's ISE Design Suite are used for hardware implementation, whereas C++ programming language, MicroBlaze soft-processor and Xilinx's XPS tools are used for software implementation.

AN INDOOR LOCALIZATION ALGORITHM ON FPGA

ÖZET

1. INTRODUCTION

In the next century, consumer computing will heavily rely on mobility. In this respect, numerous developments related wireless communication have been made in recent years. With developing mobile communication systems, accurate location-sensing systems have been researched extensively due to increasing demand for accurate positioning of people and objects. Outdoor localization is realized with high accuracy due to the developments in space-based satellite navigation systems such as Global Positioning System (GPS), Global Navigation Satellite System (GLONASS) and Galileo [1]. These satellite systems provide a reliable and powerful tool for outdoor positioning with their prevalence in mobile devices like phones, tablets and recently wide-spreading drones (remotely controlled aircrafts). Global navigation satellite systems provide sufficient accuracy for outdoor applications whereas for indoor applications they simply being useless because of the need of line-of-sight communication with the satellite. The main factors that make the indoor localization more challenging than outdoor localization are high interference and shadowing of the building. Recent years have seen various solutions using indoor sensor technologies to deal with this situation, which are explained in Chapter 2 in detail.

Indoor localization has been actively researched recently due to security and safety as well as service matters. The majority of indoor localization systems utilize the radio waves. Among the various types of technologies Radio Frequency Identification (RFID) [2] offers a practical technology due to their light weight, low power consumption and low cost. Especially non-line-of-sight communication ability and identification features are the main factors that leads to RFID usage in this work.

Another current topic regarding wireless communication and thus mobility is the intelligent building applications. In this respect, The European Union carries on a project called as Building as a Service [3]. The goal of the project is “establishment of an open service platform enabling smart commercial buildings” [3], with their own words. In other words BaaS aims to create a basis for building automation applications, and make it cost-efficient as well as easy to be implemented. Another expected impact of creating technologically equipped buildings is regarding the nature. According to BaaS, buildings managed by automation systems reduces of energy consumption and emissions in

significant percentages [4]. The BaaS project also includes real-time indoor localization [1] application regarding emergency time practices like alarming, access control and the evacuation of the building, and these aspects are related the study presented in this paper..

Evacuation of the building in case of an emergency is highly based on the fast localization of people aimed to be evacuated safely. In this sense fast detection of location, and making it by secure communication are forms the basis of our work.

Our proposed setup is consist of reference RFID tags fixed onto wall, RFID readers aimed to be located and a processing unit on which the localization algorithm run. Considering the amount of data generated by people in a huge building, a hardware working along with the main computer system is aim to be design for this processing unit, due to some performance concerns. The main blocks of the indoor location system can be seen in the figure.

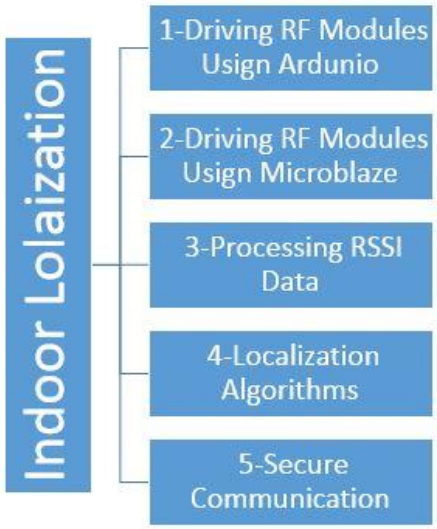


Figure 1.1 : Parts of the project

In this paper, a RFID-based indoor localization algorithm [4] and a RFID mutual authentication algorithm [5] are presented. In the chapter 2 some preliminary knowledge is given related with used tools and software, as well as the important concepts regarding indoor localization. Similarly, some preliminary knowledge is given about the indoor

localization and RFID mutual authentication algorithms in the chapter 3. Chapter 4 and 5 includes hardware and software implementations respectively. And finally in chapter 6 results and conclusions regarding these implementations are discussed.

2. BACKGRAUND INFORMATIONS

2.1. Radio Frequency Identification

Nowadays, mobility and thus wireless communication is primary concern in consumer computing. Radio frequency identification is one of the most widely used wireless technology for automatic identification and data capture applications. Due to their light weight, low power consumption, cost-effectiveness and non-line-of-sight readability RFID offers practical technology for context-aware computing like indoor localization systems [2].

An RFID module uses electromagnetic waves and employs a chip and an antenna for two-way transfer of the data. RFID systems can be classified as tag and reader, where reader reads data generated by tag as the names suggest. Nowadays, various types of applications such as asset tracking, supply chain management and payment systems (electronic tickets) includes RFID systems [7].

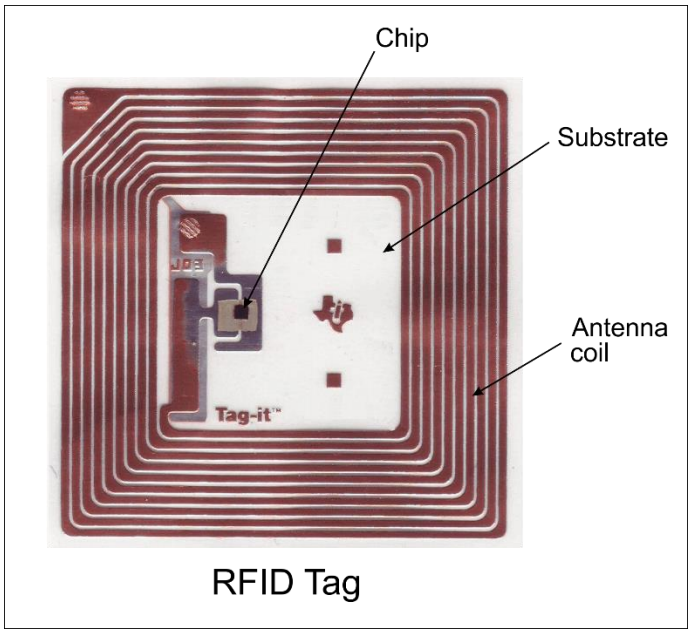


Figure 2.1 : RFID tag architecture [8]

The most distinctive property of RFID from earlier barcode technology is that they does not require a line of sight. RFID provides identification from a distance varying according to type of the module employed. Based on their power consumption, RFID tags can be categorized as active tags, passive tags and semi-passive (battery-assisted) tags. Active

tags employs internal power sources and can continuously communicate with the reader for long ranges, whereas passive tags collect power from interrogating waves and backscatter the radio signal to the nearby reader [7]. The main advantages of passive RFID systems over active RFID systems are the lower costs, smaller sizes, and longer life-spans, however they are not always practical since the power constraint result in lack of functionality. Another categorization can be made by frequency range since RFID systems operate at a variety of radio frequencies from 120Hz to 10Ghz. Depending on working frequency RFID systems are named as low frequency, high frequency, ultra-high frequency. Generally, a system operating at a higher frequency faster data transfer rates and longer read ranges than lower frequency systems [9].

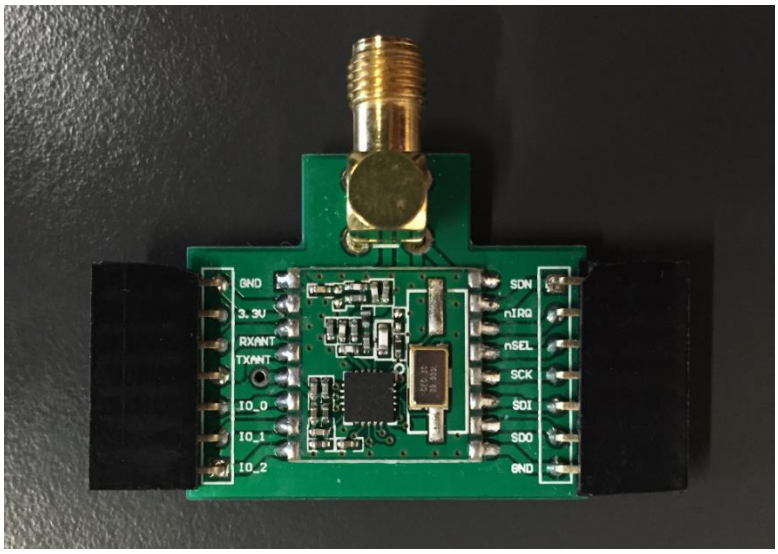


Figure 2.2 : RFM23 - active RFID module

2.2. Arduino UNO

Ardiuno is a microcontroller board based on ATMEL series microcontrollers [10]. Arduino UNO is the simplest and the cheapest board among Arduino board family. Other than microcontroller, the board consists of 14 digital input/output pins, 6 analog inputs, a 16 MHz crystal oscillator, a USB connection, a power jack, an (In-Circuit Serial Programming (ICSP) header, and a reset button. Arduino has a specific software called Arduino Suite to program any Arduino compatible board. With growing number of product-specific, libraries Arduino offers useful tools for designs related with embedded systems.

Technical specifications of Arduino UNO is listed as follows [10];

- Microcontroller: ATmega328
- Operating Voltage: 5V
- Input Voltage (recommended): 7-12V
- Input Voltage (limits): 6-20V
- Digital I/O Pins: 14 (of which 6 provide PWM output)
- Analog Input Pins: 6
- DC Current per I/O Pin: 40 mA
- DC Current for 3.3V Pin: 50 mA
- Flash Memory: 32 KB of which 0.5 KB used by boot loader
- SRAM: 2 KB
- EEPROM: 1 KB
- Clock Speed; 16 MHz

  Front view and the main components of Arduino UNO board can be seen in the Figure 2.3.

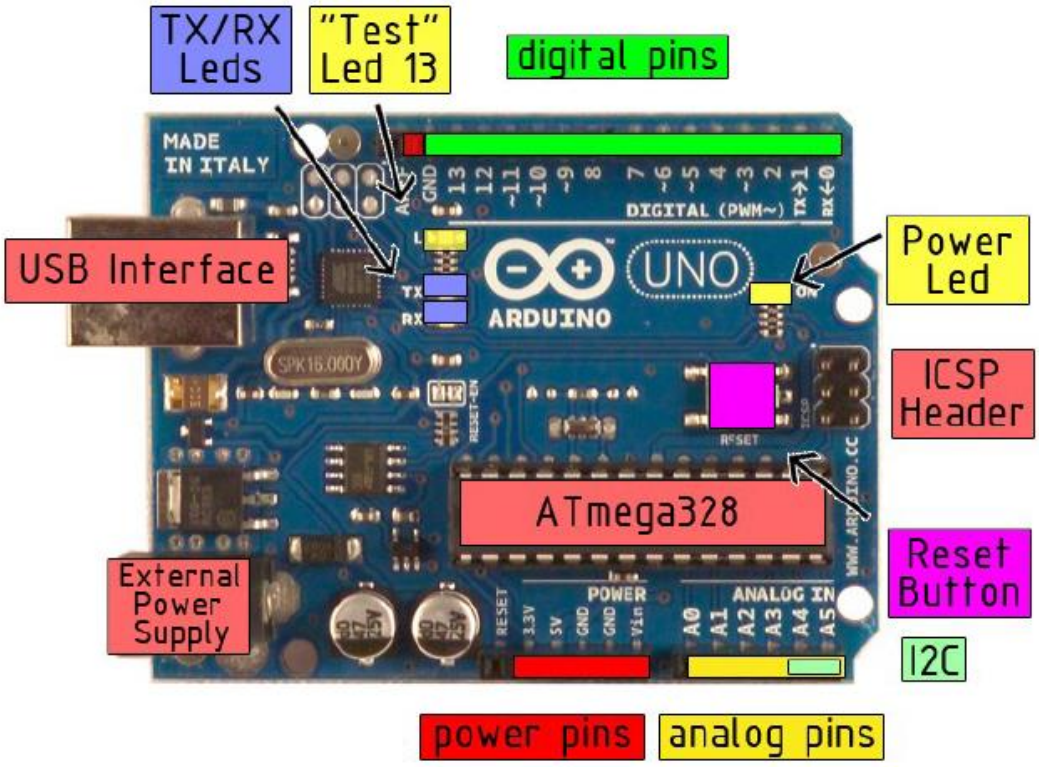


Figure 2.3 : Arduino UNO board [10]

2.3. Localization Systems

The demand of an accurate location-sensing systems for positioning of people and objects have been increased extensively with developing mobile communication systems in recent years. Localization System can be categorized in two main groups as outdoor (global) positioning systems and indoor (local) positioning systems. Location-sensing systems are aimed to be accurate enough to provide the position (longitude, latitude, and altitude) of the target with sufficient approximate. This goal is relatively easy to be achieved in global positioning where meters of inaccuracy can be tolerated due to large coverage area. Outdoor localization is realized with high accuracy thanks to the developments in space-based global navigation satellite systems like Global Positioning System (GPS), Global Navigation Satellite System (GLONASS) and Galileo [1]. Especially with its prevalence in mobile devices GPS provides a convenient and powerful tool for outdoor positioning. One main disadvantage of global navigation satellite system that prevents it to be used in indoor localization systems is the need of line-of-sight communications. High interference caused by presence of obstacles and shadowing of the buildings make global navigation satellite systems unreliable to be used in local positioning systems by means of accuracy. In this respect, solutions have been developed to deal with indoor localization challenge.

There are four widely used fundamental methods for both local and global positioning systems and these methods are time of arrival method (TOA), direction of arrival method (DOA), time difference of arrival method (TDOA) and received signal strength method (RSS) [1].

	Accuracy (m)[a]	LOS/NLOS	No. of Base Stations
TOA	M	LOS	≥3
TDOA	M	LOS	≥3
DOA	L	LOS	≥2
RSSI	H to M	Both	≥3

Figure 2.4 : Comparison of basic RTLS methods [1]

TOA: This method allows the measurement of distance from the signal propagation times to multiple nodes with pre-defined locations. There must be at least three time-of-arrival

value and therefore three nodes in order to find target's location via triangulation technique.

DOA: This method uses a fraction of the signal's wavelength, to estimate distance. Phase measurements is needed for this localization method

TDOA: The principle of this method lies on the idea that is measurement of distance from the multiple signals propagation time differences. There must be at least three base nodes to calculate location like TOA method.

RSS: This measurement method is ubiquitous in wireless systems like positioning. RSS method generally accurate than other methods since it is more tolerant to the drawbacks of non-line-of-sight communication as it is illustrated in the figure. Generally, RFID modules has embedded RSSI measurement, and that is why they are commonly used localization-sensing applications. In the most of RFID modules RSSI is an 8 bit value hence outputs a value between 0 and 255 inversely proportional to distance between tag and reader.

2.4. Field Programmable Gate Arrays

Field Programmable Gate Arrays is a type of programmable logic device (PLD) that is used to realize a digital function and it is developed in mid 1980s [11]. The term field-programmable means that the device can be reprogramed by customer after fabrication. However, overhead of programmability is one major drawback of FPGAs, which makes them slower and more expensive than application specific integrated circuits (ASIC). Because of this drawback FPGAs are mostly used as prototyping platforms for ASIC designs.

A FPGA device consists of two dimensional array of programmable switches and logic cells (logic blocks). Logic cells is used for realization of logic functions, whereas the programmable switches is used for interconnection of logic cells. By constructing basic functions in logic cells and connect them together by programming switches, any desired digital circuit meeting constraints can be realized on a FPGA device.

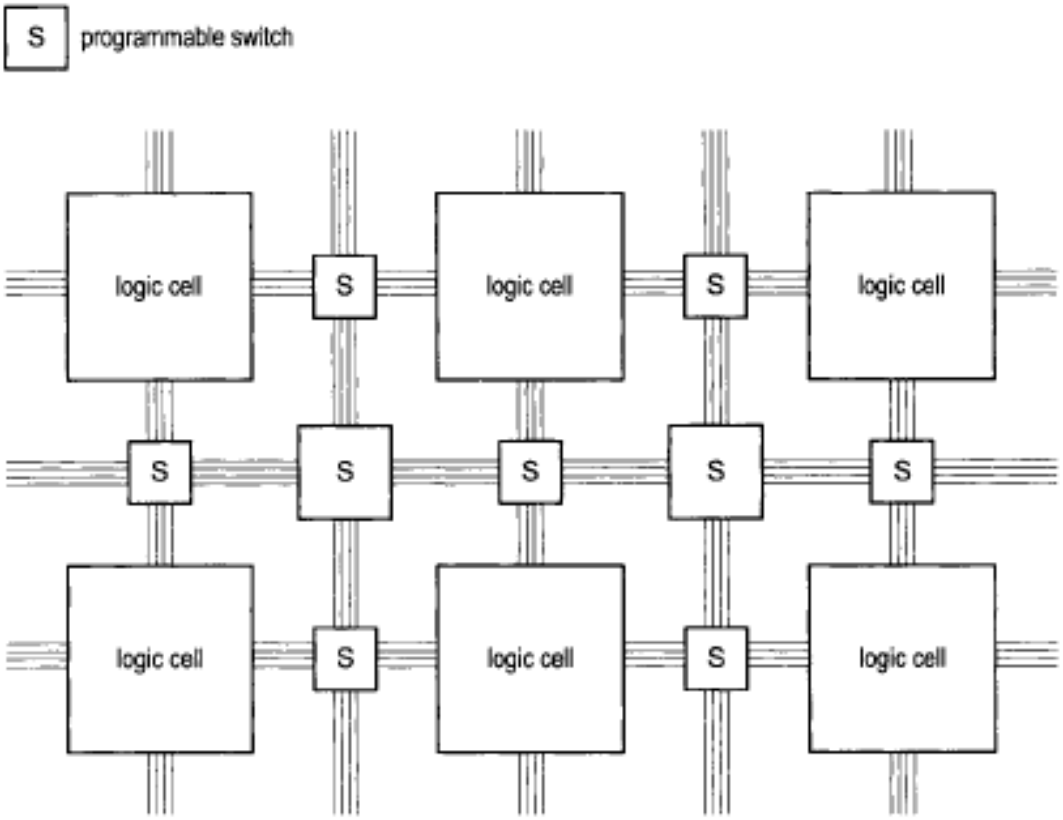
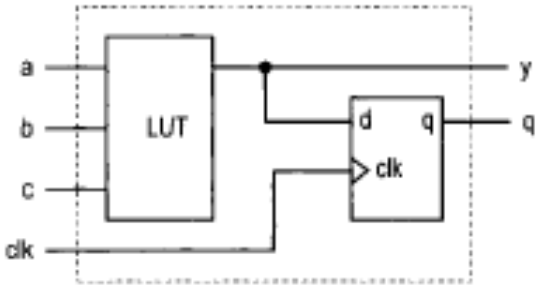


Figure 2.5 : Conceptual structure of an FPGA device [11]

The conceptual structure of a logic cell contains slices, LUTs, registers and multiplexers.[11]

- Slice: Logic resources (LUTs, flip-flops and multiplexers) are grouped in slices to create configurable logic blocks.
- LUT: LUTs are used to implement small logic functions with outputs of the truth table of that logic functions.
- Registers: A register is a group of flip-flops that stores a bit pattern. A register on the FPGA has a clock, input data, output data, and enable signal port.
- Multiplexers: Multiplexer is a circuit that selects between two or more inputs and outputs the selected input.



a b c	y
0 0 0	0
0 0 1	1
0 1 0	1
0 1 1	0
1 0 0	1
1 0 1	0
1 1 0	0
1 1 1	1

Figure 2.6 : Conceptual structure of a logic cell [11]

FPGAs has an ability of parallel processing, which provide an opportunity to design circuits with high speeds. In this respect, it is possible to realize a microprocessor on a FPGA. Being able to realize a microprocessor on a FPGA makes it a suitable device for hardware-software coo-design.

2.5. MicroBlaze Soft Processor

MicroBlaze is a soft processor with 32bit reduced instruction set and it is optimized for Xilinx's FPGA [12]. MicroBlaze can be created by Xilinx's EDK tool, and programmed by using C or C++ programing languages with Xilinx's SDK tool. The archicetural feature of this soft cored processor can be listed as [12]:

- Thirty-two 32-bit general purpose registers.
- 32-bit instruction word with three operands and two addressing modes.
- 32-bit address bus.
- Single issue pipeline.
- Allows selective enabling of additional functionality.
- Floating Point Unit (FPU)
- Memory Management Unit (MMU)

In this study, implementation of Molnar Wagner Protocol is planned to be done on MicroBlaze. The architecture of Microblaze processor is illustrated in the figure;

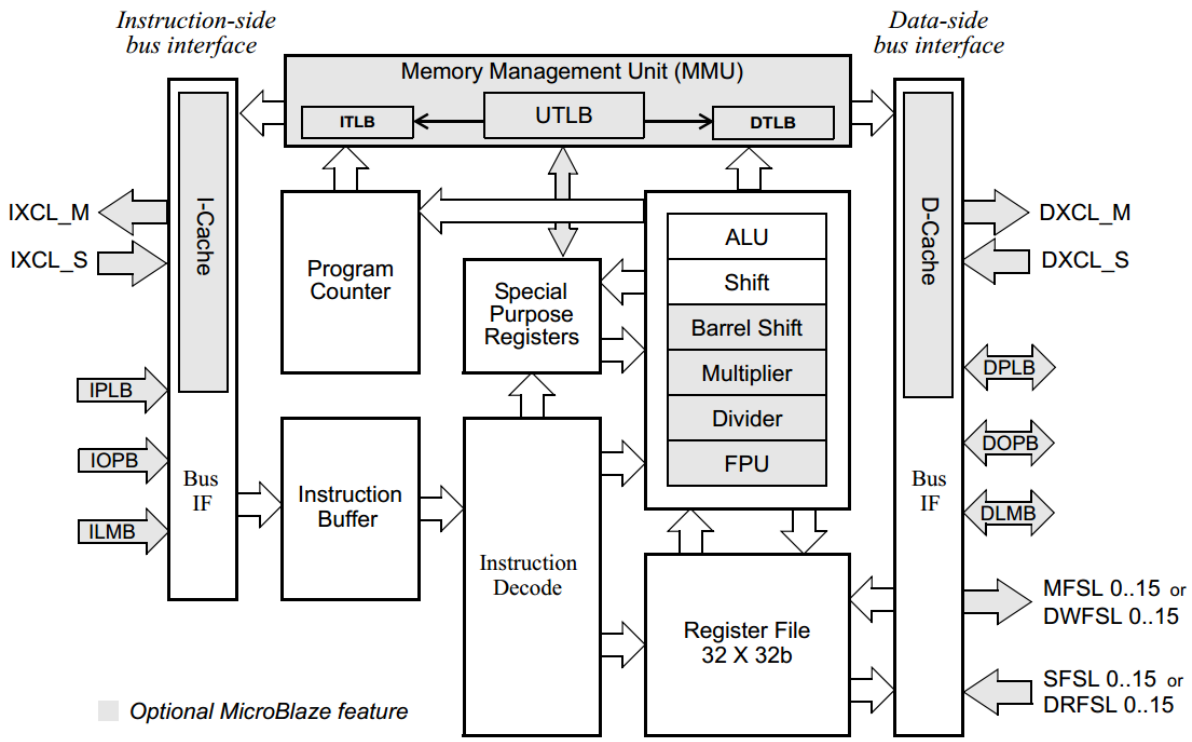


Figure 2.7 : MicroBlaze Core Block Diagram [12]

2.6. Verilog Hardware Description Language

Verilog is a hardware description language and it developed in the 1980s by Philip Moorby and Prabhu Goel. With the VHDL (Very High Speed Integrated Circuit Hardware Description Language), Verilog is one of two most commonly used hardware description language in recent years due to its simplistic and effectiveness. Verilog is structurally familiar with C programing language, and this is one important reason why it is used in digital circuit design widespread. In this study it is used for implementation of indoor localization algorithm along with Xilinx ISE Design Suite tool [11].

2.7. Xilinx ISE Design Suite Environment

Xilinx's ISE Design Suite Environment tool is used for implementation of digital circuits on Xilinx compatible FPGA boards. It allows to design a digital circuit with either VHDL or Verilog hardware description language. This suite offers several tools for schematic drawing, synthesis and implementation steps, as well as provides a test bench environment

for testing the design under both no-delayed (behavioral test) and delayed conditions. After each step error and warning reports are also generated to inform the user about mistakes that has been done. Apart from that, it also offers timing area and speed report related with the design. Implementation steps of a generic digital circuit can be seen in the figure 2.8 [11].

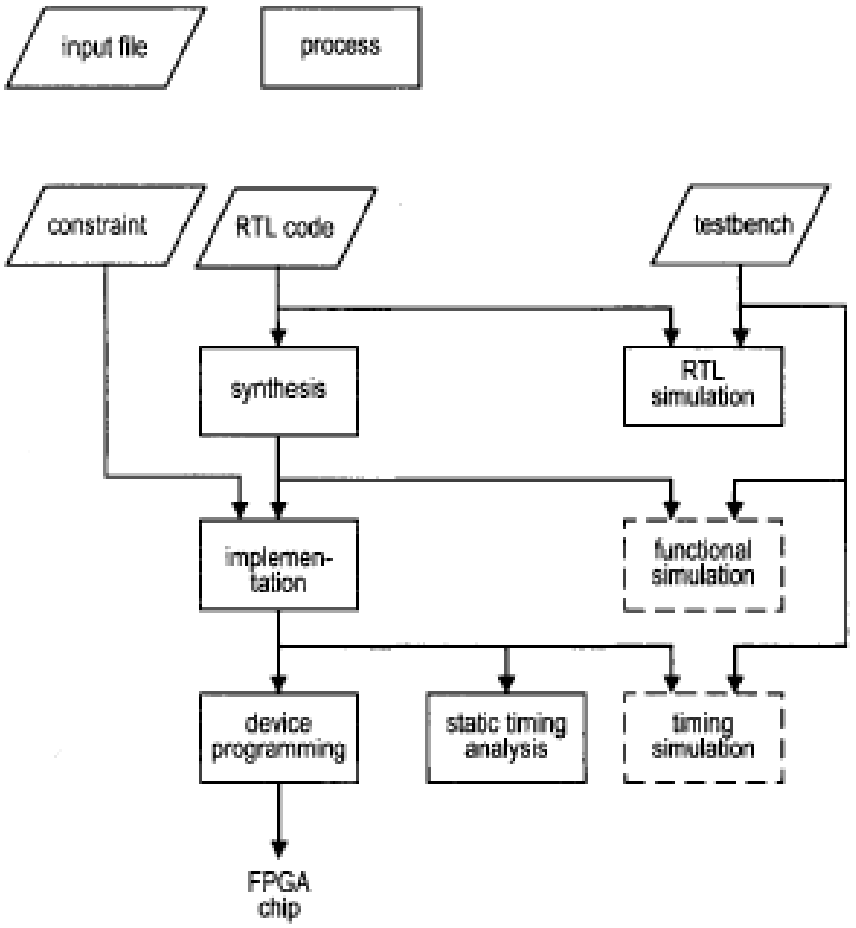


Figure 2.8 : Flow chart of design development [11]

Synthesis: This step generate a netlist file.

Implementation: This step has three inner steps

- Translate: Merge multiple design files into a single netlist.
- Map: Group the gates from the netlist into physical components called slices and bonded inputs-outputs (IOB).

- Place & Route: Place components onto the chip, connect the components, and extract timing data into reports.
- Generate Programming file: Generates a file wit .bit extension which programs the FPGA.

Figure 2.9 : A typical ISE window [11]

2.8. Xilinx Embedded Development Kit

Xilinx Embedded Development Kit used for creating microprocessor based designs is for Xilinx's FPGAs [13]. User can construct an embedded microprocessor system with its all specifications such as core and peripheral parameters. EDK lets the user select from various optional features, which is useful by means of creating microprocessor based designs without unnecessary parts. EDK tool makes it relatively easy to realize a

microprocessor on FPGAs, without dealing with complicated flow diagrams for microprocessor design. MicroBlaze soft-cored processor mentioned in Section 2.5 can be realized on FPGA by using EDK tool.

The generic window of Xilinx Embedded Development Kit can be seen in the picture below.

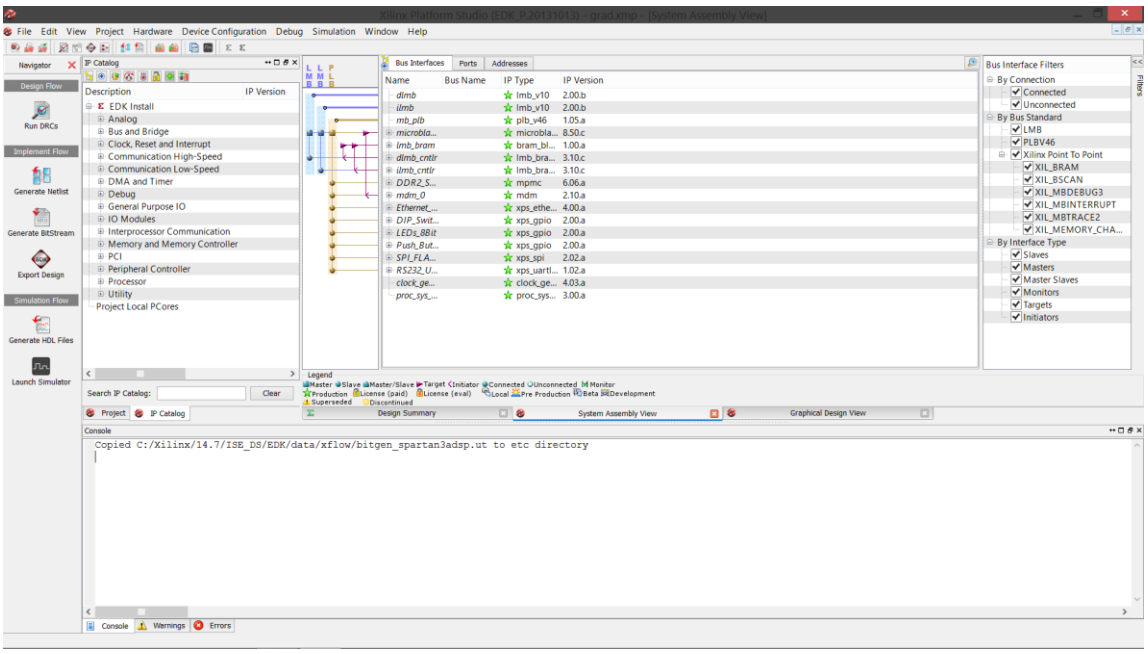


Figure 2.10 : A typical EDK window

2.9. Xilinx Software Design Studio

Xilinx Software Design Studio is an integrated design environment that helps the user about the development of software application projects [14]. SDK tool is used for MicroBlaze soft-cored processor mentioned in Section 2.5 can be programed by using SDK tool. General features of SDK tool are listed below [14];

- Feature-rich C/C++ code editor and compilation environment
- Project management
- Application build configuration and automatic Makefile generation
- Error navigation
- Well-integrated environment for seamless debugging and profiling of embedded targets

- Source code version control

The generic programming window of Xilinx Software Design Studio can be seen in the picture below.

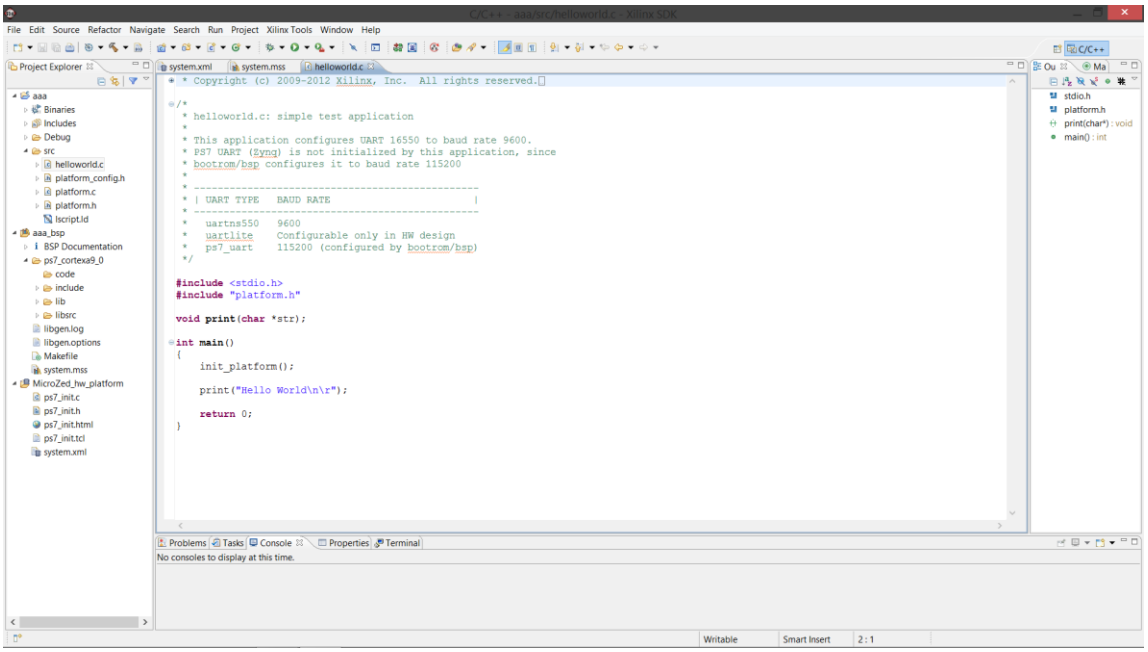


Figure 2.11 : A typical SDK window [14]

3. LOCALIZATION IDENTIFICATION BASED ON DYNAMIC ACTIVE RFID CALIBRATION

LocAlizatioN iDentification based on dynaMic Active Rfid Calibration (LANDMARC) is an indoor localization system based on RFID technology and signal strength analysis [5]. It is developed by Michigan State University and Hong Kong University of Science and Technology together [15]. Especially with its reasonable system price and dependability, RFID technology provides attractive tools for LANDMARC indoor localization applications. In this respect, LANDMARC is a localization method which employs reference RFID tags (base stations) along with RFID readers to increase accuracy [15]. LANDMARC system is also a prior method by means of using reference tags and readers together.

Localization methods employing reference tags along with readers are known tag-based methods, whereas localization methods employing only readers are known tag-based methods. The major problem with the reader-based RFID localization is that obstacles in environment causes the signal fluctuation that results in inaccurate distance computation. By using base stations with pre-known locations, drawbacks resulted from obstacle interferences are minimized, and hence the calibration between reference tags and reader is ensured [16]. Minimizing these drawbacks caused by non-line-of-sight communication is the most powerful side of LANDMARC system that makes it attractive for indoor localization applications. Another advantage of using base stations is that the system can work with less number of relatively expensive reader modules, which provides cost-effectiveness to the system.

The setup showed in the figure below is proposed to implement a LANDMARC based indoor localization algorithm. The setup is consist of readers, reference tags and a target tag whose location is aim to be computed. The numbers at the bottom-right corner of reference tags denotes ID of related reference tag. The length d is distance between two consecutive reference tags.

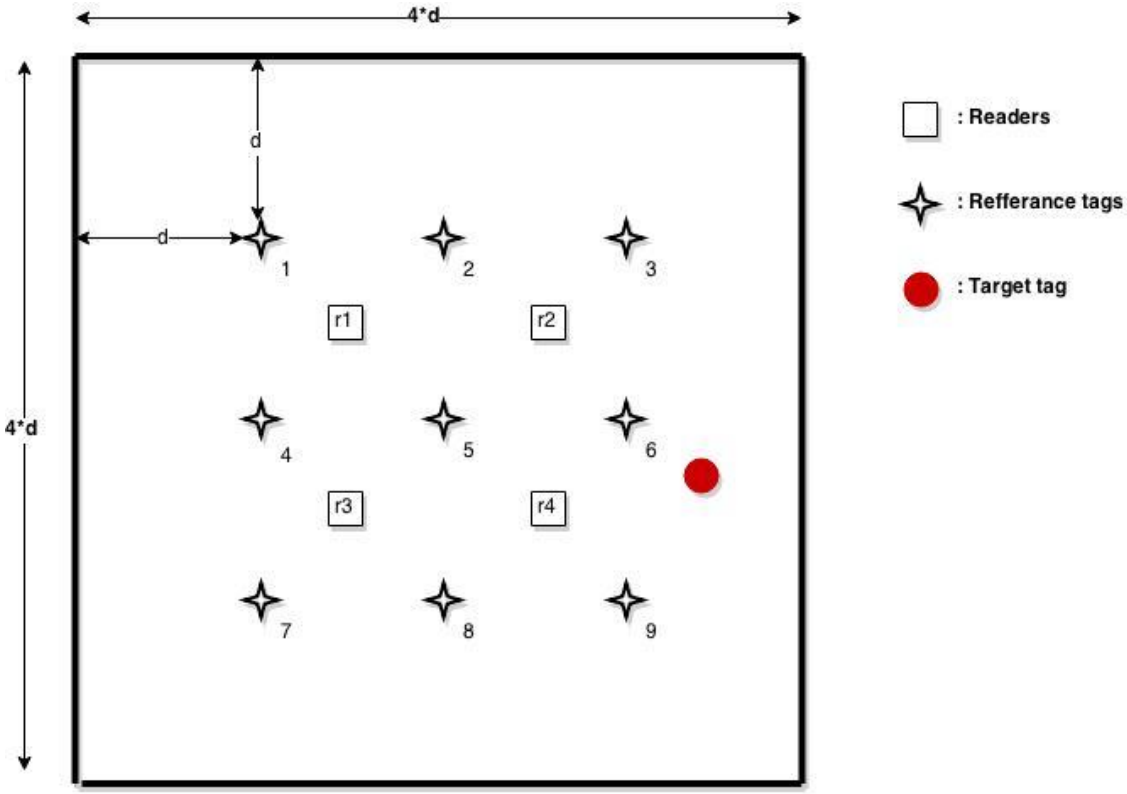


Figure 3.1 : Proposed setup for implementation of LANDMARC algorithm

Locations of reference tags are fixed, and they serve as reference points in the system as the name suggest. As it stated earlier, presence of reference tags is helps the location calibration in order to provide more accurate and reliable results. Both the placement and the selection of reference tags can significantly affect the accuracy of location estimation. Locations of reader are also fix and they are used to obtain the signal strength values of target tag and reference tags.

The main operation of LANDMARC is comparison of the signal strength values of target tag and reference tags detected by reader and then calculation of weighted values for k

nearest reference tags. k is denotes the number of nearest reference tags to be used in weighted value calculating. The calculated weighted values are used to find location of the target by computing them with nearest reference tags' pre-defined location values [15].

Used parameters are defined as follows;

- Si : The signal strength value of target tag detected by ith reader.
- Qi,j: The signal strength value of jth reference tag detected by ith reader.
- Ej : Euclidian RSSI value of jth reference tag calculated by using S and Q values.
- Wj : Weighted value of jth nearest reference tag calculated by using E values of kth nearest neighbors.

S and Q Parameters' notations are illustrated in the figure below.

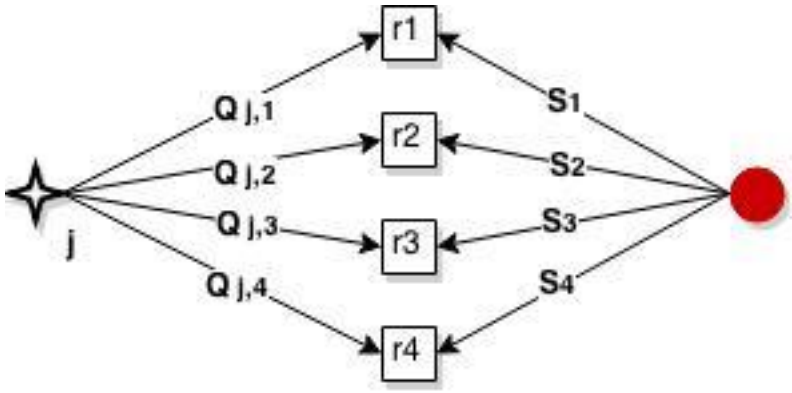


Figure 3.2 : Signal notations in LANDMARC

The implemented LANDMARC based algorithm is consists of 3 blocks which are Euclidian (E) block, kth nearest neighbor (kNN) block and weighted coefficients (W) Block. In following sections these sub blocks are presented. Equations related with this block can be seen below;

$$(x, y) = \sum_{i=1}^{k} Wi * (xi\ , yi) \tag{3.1}$$

Used parameters in the equations are defined as follows;

- (x, y) : two dimensional position of target tag in horizontal axis
- (xi, yi) : two dimensional position of ith reference tag in horizontal axis

- Wj (Wi) : Weighted value of jth (ith) nearest reference tag calculated by using E values of kth nearest neighbors.

The block diagram with top sub-blocks of LANDMARC algorithm can be seen in the figure;

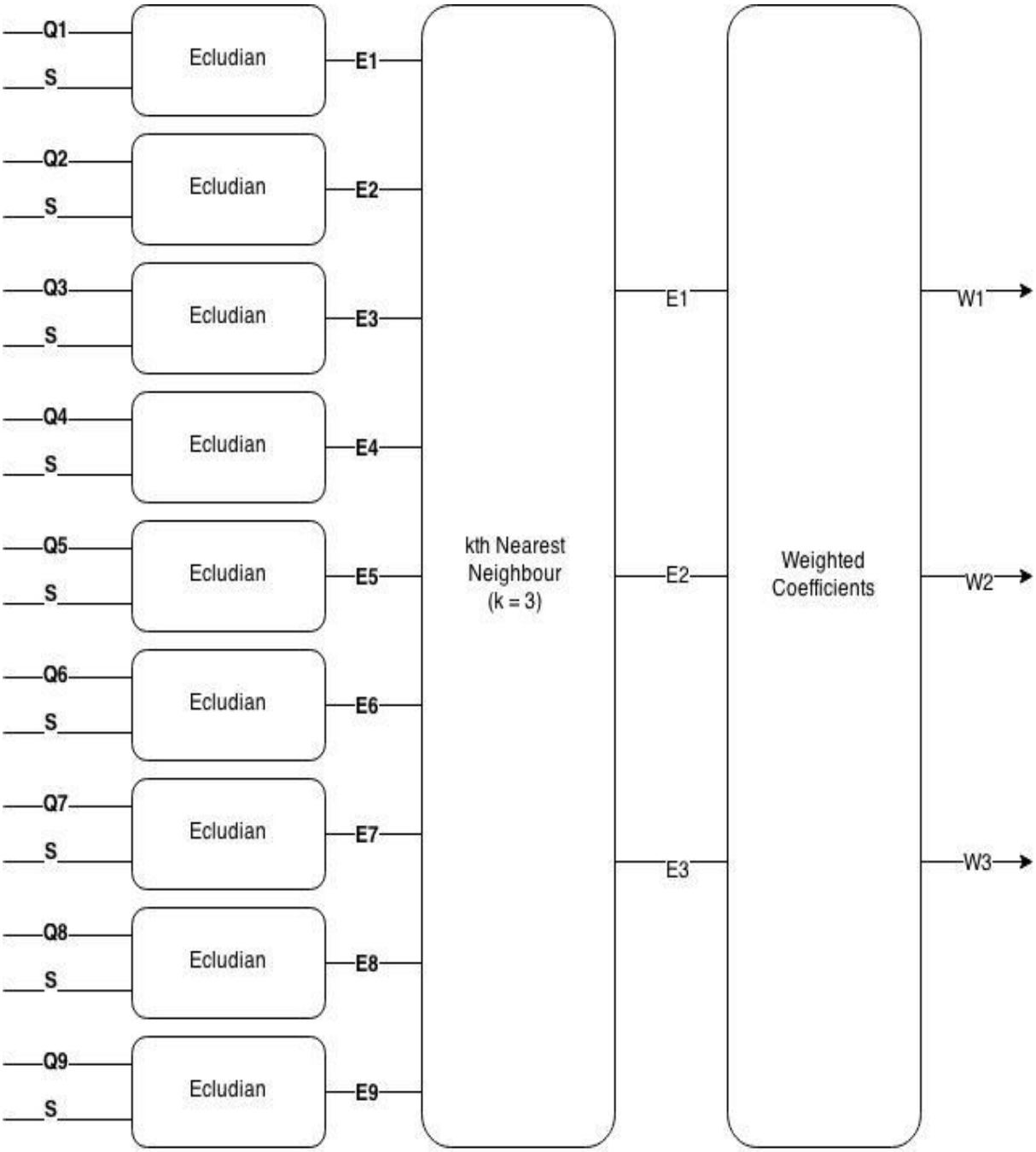


Figure 3.3 : Block diagram of LANDMARC

3.1. Euclidian Distance Block

Euclidian Distance terms is named after famous Greek mathematician Euclid and defined as; the length of the line segment connecting two points [17]. In our case, this block is

used in the comparison of the signal strength values of target tag and reference tags detected by reader.

$$Ej = \sqrt{\sum_{i=1}^{n}(Qj, i - Si)^2} \quad (3.2)$$

The output of the block represent a distance inversely proportional to signal strength value. In other words, the higher the RSSI value, the smaller the distance between tag and target. Among the calculated Euclidian Distance values, the smallest ones are used in the computation of weighted values in further steps.

3.2 kth Nearest Neighbor Block

This block is used to sort the calculated Euclidian Distance in previous block is ascending order, and then choose a number (k) of smallest values in order to calculate weighed coefficients. k is the amount of chosen numbers. By increasing k more accurate weighted coefficient and hence more reliable localization results can be accomplished.

For the implementation of the sort operation bubble sort algorithm (also known as sinking sort) [18] is chosen due to its simplicity. This algorithm considered as inefficient for extensive work of comparing every element with each other

Proposed algorithm for sort operation presented in detail in Chapter 4th chapter.

3.3. Weighted Coefficients Block

Weighted coefficients are calculated by using E values of kth nearest neighbors. These values indicates a weighting factor which is greater for the nearest tag. Amongst the varieties of weighted coefficients calculation methods equation is provides satisfactory results by means of accuracy. The logic behind this formula is that RSSI value is roughly inversely proportional to the square of the distance [5].

$$Wj = \frac{\frac{1}{Ej^2}}{\sum_{i=1}^{k}\frac{1}{Ei^2}} \quad (3.3)$$

Implementation of the equation as a combinational circuit is presented in detail in Chapter 4.

4. MOLNAR WAGNER MUTUAL AUTHENTICATION PROTOCOL

The Molnar Wagner Mutual Authentication Protocol is implemented as software to be operate along with LANMARC localization algorithm. Details about software implementation of the protocol is presented in Chapter 3.

4.1. Security Aspects of Radio Frequency Identification

RFID offers practical technology for context-aware computing like indoor localization systems due to their light weight, low power consumption, cost-effectiveness and non-line-of-sight readability. In this sense RFID is one of the most widely used wireless technology for automatic identification and data capture applications. However, because of this widespread usage, some cryptographic needs have begun to emerge.

The main security requirements in RFID applications are confidentiality and authentication. Confidentiality is making information secret for keeping it from unauthorized entities. Confidentiality can be ensured either by encrypting the message's meaning or hiding the existence of a message. As another main security requirement, authentication means corroboration of identity or source of information [19]. In this study the main concern regarding cryptographic services is source and destination authentication (mutual authentication).

As it can be seen in the figure, setup for localization system consist of three types of RFID modules. Among them readers are the most probable targets for attacks, since the final localization result computed and transferred over them. In this sense the system is defenseless in case an attacker pretend as a tag, because reader does not require an authorization for communicating. Likely, an attacker with reader can collect information since tags do not authorization for communicating. In this respect, Molnar Wagner Protocol is proposed to meet mutual authentication needs of the system.

4.2. Background of Molnar Wagner Mutual Authentication Protocol

Molnar and Wagner have given specific proposals for improving privacy in RFID tags, in their work published in 2004 [6]. With their mutual authentication protocol they proposed a lightweight algorithm in order to corroborate the source and the destination in RFID

systems. Their work is arisen from the need that elimination privacy risks of RFID technology in the applications like supply chain application, proximity cards, and tagging of books, tapes, CDs in libraries [6].

Molnar and Wagner's mutual authentication protocol is initially intended to be applied on library RFID system. However due to its simplicity and hence the effectiveness by means of speed, it can be applied various types of application and localization systems are one of them. Especially algorithm's simplicity is make it able to be used in systems having time constraints like real time localization systems.

Molnar and Wagner's mutual authentication scheme based on Pseudo Random Number Generator (PRNG) functions, and involves two parties, namely a reader and a tag. Proposed mutual authentication protocol by Molnar and Wagner can be seen in figure below.

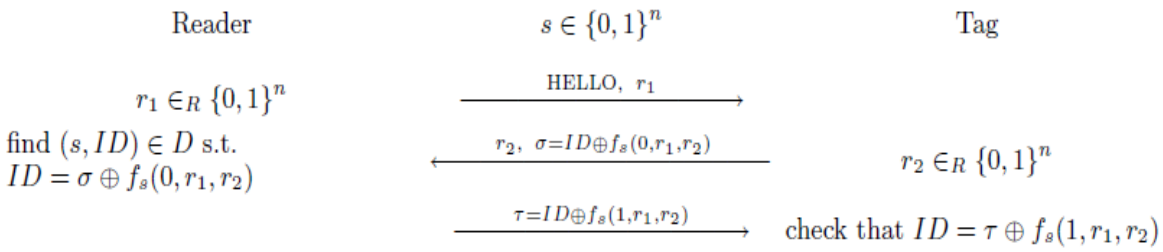


Figure 4.1 - The Molnar-Wagner Protocol [6]

The notations that are used are explained in following.

- ID : Denotes the identification of the tag
- $HELLO$: Denotes the message sent by the reader to query the tag.
- f : Denotes an encryption function used by the tag and the reader in the system.
- $r1$: Denotes the random number generated by the reader.
- $r2$: Denotes the random number generated by the tag.
- σ: Denotes the result of computation between ID and a pseudo random function f generated by the tag.
- τ : Denotes the result of computation between ID and a pseudo random function f generated by the reader.

At setup time, a unique identification is given to each tag and a secret initial key that is used in the generation of subkeys is shared between tag and reader. In other words, the protocol does not provide a secure sharing for key and the identification, and sharing is assumed to be done before the protocol starts. The server generates a random number r1 queries a tag by sending a HELLO. After tag being queried, it generates a random number r2 and computes σ by using its ID, r1 and r2 values, and sends them to the reader. Then reader use this σ value in computation to conclude tag authentication. If the result of computation is matched with pre-known ID, tag is said to be authenticated. After the tag authentication is done, the reader computes τ value by using authenticated ID, r1 and r2 values, and sends it to the tag. Then the tag use this T value in computation to conclude reader authentication. If the result of computation is matched with the tag's ID, server is said to be authenticated.

4.3. Simplified Data Encryption Standard Algorithm

SDES is a reduced version of the DES (Data Encryption Standard) algorithm, and is also a block cipher algorithm [20]. Since SDES and DES algorithms has many common features and operations, understanding DES algorithm is the key to implement a SDES algorithm. As a once common symmetric-key encryption algorithm, DES stands as a cryptographically strong algorithm. DES is developed at IBM in mid 70's and had been the most widely used cryptosystem until the Advanced Data Encryption Standard (AES) has developed. DES algorithm defined for 32 bit plain text block and operates with 48 bit subkeys [21]. DES is an iterated cryptosystem in which every iteration called round, and every round a round function operates. At each round, output of previous round gets into computation with a round specific subkey, and outputs input for the next round. The computations includes an expansion box, a XOR operation, substitution box operations and permutation operation. Apart from round algorithm, DES also includes a key scheme taking a randomly chosen key as its input in order to generate sub keys. The secrecy of the data depends on the randomly chosen key, and this situation is also valid for the SDES algorithm

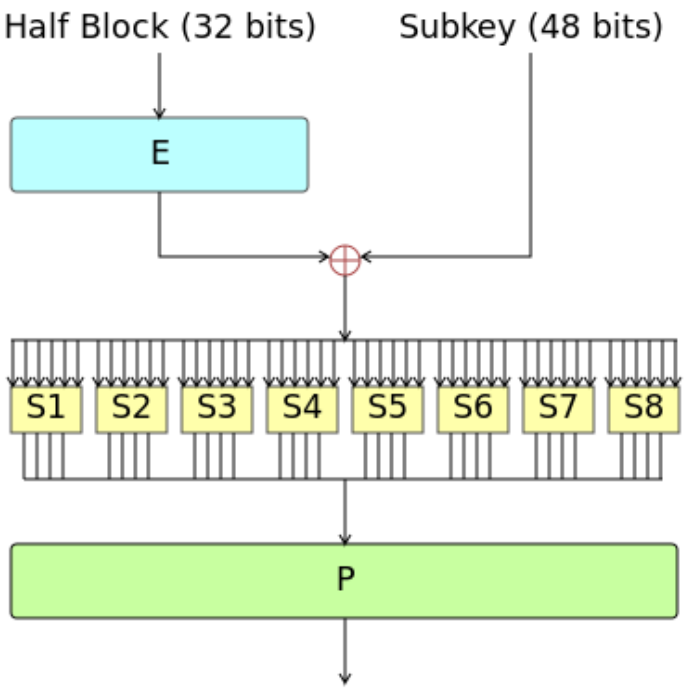


Figure 4.2 : Generic round for DES [22]

The notations that are used are explained in following.

- E: Denotes the expansion box.
- Sn: Denotes the substitution boxes for 6 bit blocks.
- P: Denotes the permutation operation.

SDES and DES algorithms has many common operations. However as a lightweight version of DES, SDES operates on 8bit message blocks and 10bit sub keys instead of 32 bit message blocks and a 48bit subkeys. Considering a RFID module stores data like ID and RSSI as 8 bit values, SDES is a perfect match for cryptographic applications regarding RFID. Like it is mentioned for DES algorithm, SDES is also an iterated cryptosystem in which every iteration called round, and every round a round function operates. Likely, the computations includes an expansion box, a XOR operation, substitution box operations and permutation operation. As it can be anticipated, round and substitution box numbers are much lesser than DES in order to operate SDES in lightweight systems like RFID. SDES also includes a key scheme taking a randomly chosen key as its input in order to generate sub-keys.

For the pseudo-random number function f in The Molnar Wagner Mutual Authentication Protocol, Simplified Data Encryption Standard algorithm is implemented. All permutation functions, extension functions and substitution boxes are used as defined in [20]. Details about software implementation of SDES is presented in Chapter 6.

5. HARDWARE DESIGN

5.1. LANDMARC Localization Algorithm

Localization algorithms is tend to be complex designs due to processing huge amount of data and the large coverage area they aim to offer. Especially, in the applications using real time processing such as localization-sensing, this complexity in algorithm results in delays which are the primary drawback for the entire system. Apart from the not-meeting the timing constraints, running complex algorithms on the main computer system causes some performance problems related the other programs running on the same computer. Instead of running all the program blocks in the main computer system, separating them as hardware and software blocks according to their degree of complexity helps to deal with performance issues. These are the main reasons why LANDMARC Localization Algorithm is implemented as a separate hardware to operate along with a computer system.

With their parallel processing capacity and hardware-software co-design availability, FPGAs offers suitable platforms for prototyping these type of system. LANDMARC Localization Algorithm is planned to be implemented on a Digilent FPGA Board, and for this reason Xilinx's ISE Design Suite tool is chosen for implementation. Before the hardware implementation, the algorithm is tested by using C++ programing language. After getting satisfying results same algorithm is parted to 3 main sub blocks and implemented as a hardware this time. Results is discussed in the chapter 6. General mechanism of LANDMARC is presented in chapter 3.1 in details. In this chapter software and hardware implementation of the algorithm is presented.

The implemented LANDMARC based algorithm is consists of 3 blocks which are Euclidian (E) block, kth nearest neighbor (kNN) block and weighted coefficients (W) block.

5.1.1. Euclidian Block

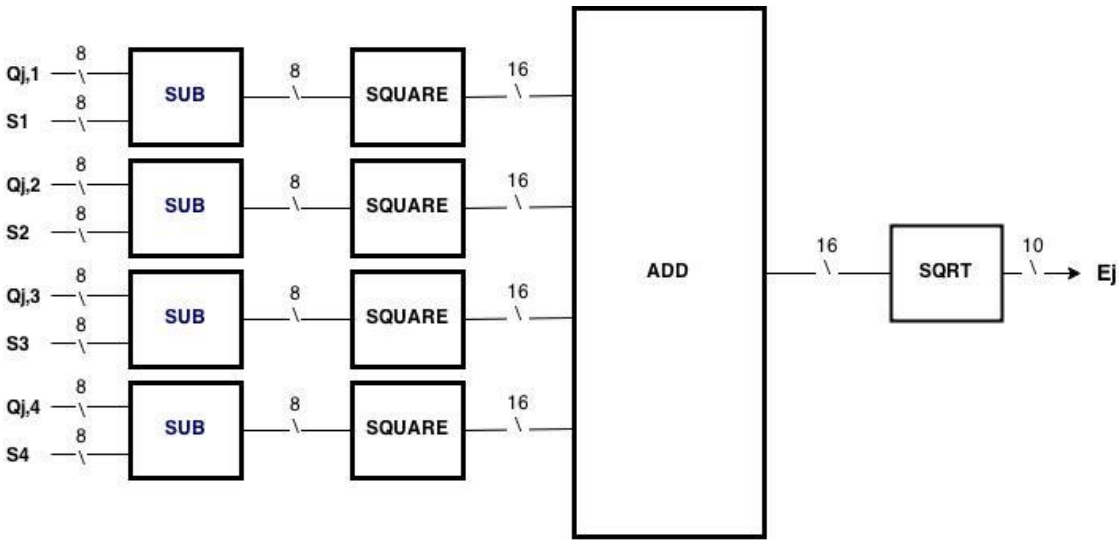


Figure 5.1 : Diagram of Ecludian block

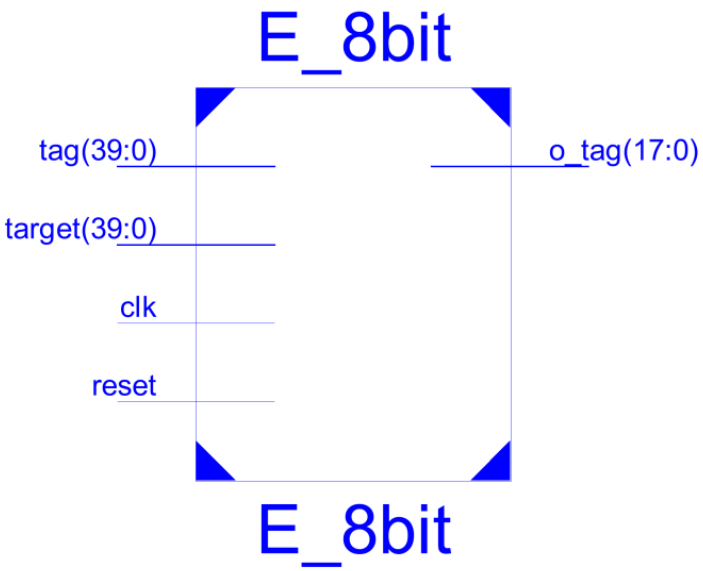


Figure 5.2 : Top RTL schematic of Ecludian block

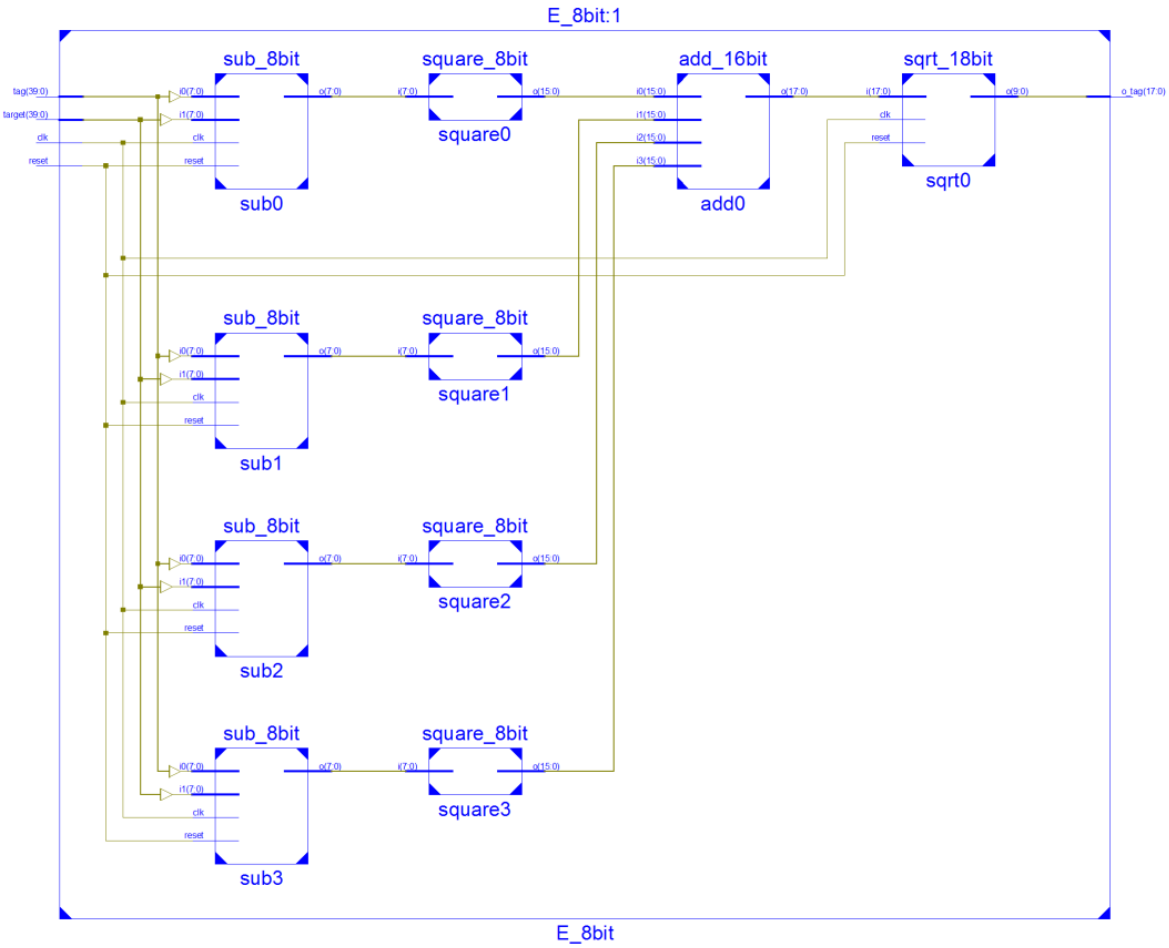


Figure 5.3 : Inner RTL schematic of Ecludian block

5.1.2. kth nearest neighbor

This block sorts its input to ascending order and outputs first k smallest values. For keeping the design simple enough for hardware implementation, k value is chosen to be 3, which preserve enough accuracy as the results indicates. By increasing k more accurate weighted coefficient and hence more reliable localization results can be accomplished. The most challenging part of this sub block design is the sort algorithm. Implemented bubble sort algorithm [18] offers simplicity that is the main criteria for designed hardware.

Pseudo-code:

for i from 1 to N

for j from 0 to N - 1

if a[j] > a[j + 1]

hold = a[j]

a[j] = a[j + 1]

a[j + 1] = hold

In order to sort the inputs bubble sort algorithm is implemented as a sequential circuit.

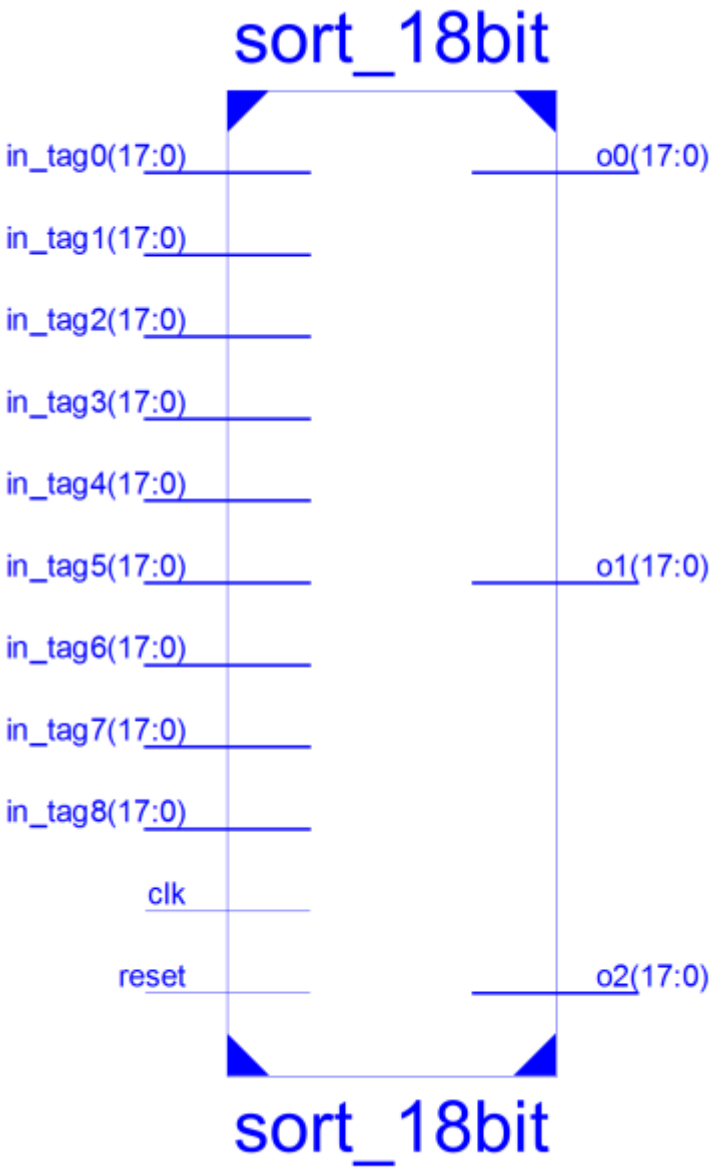


Figure 5.4 : Top RTL schematic of sort block

5.1.3. Weighted Coefficients Block

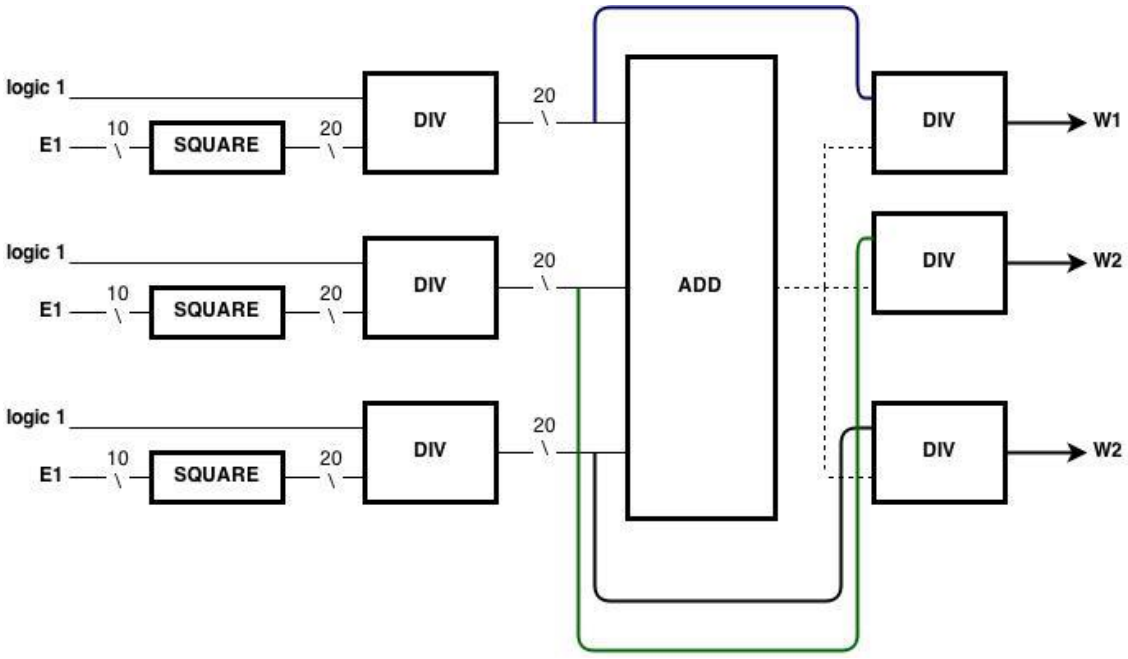


Figure 5.5 : Diagram of weighted coefficients block

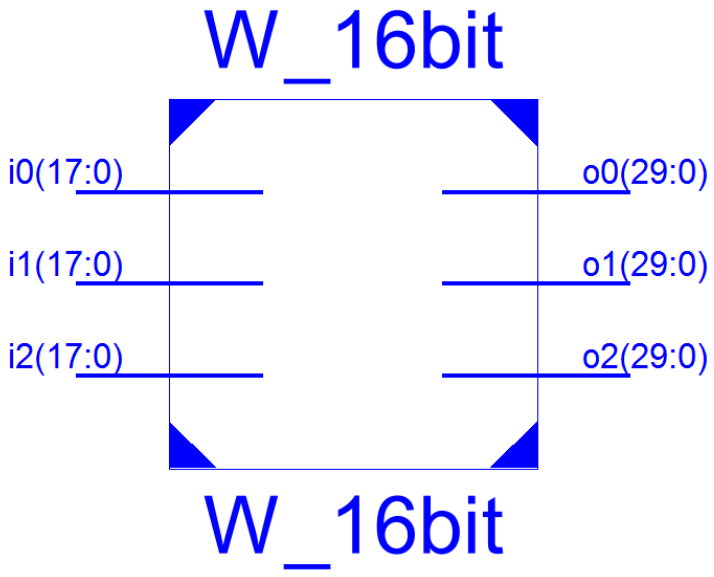


Figure 5.6 : Top RTL schematic of weighted coefficients block

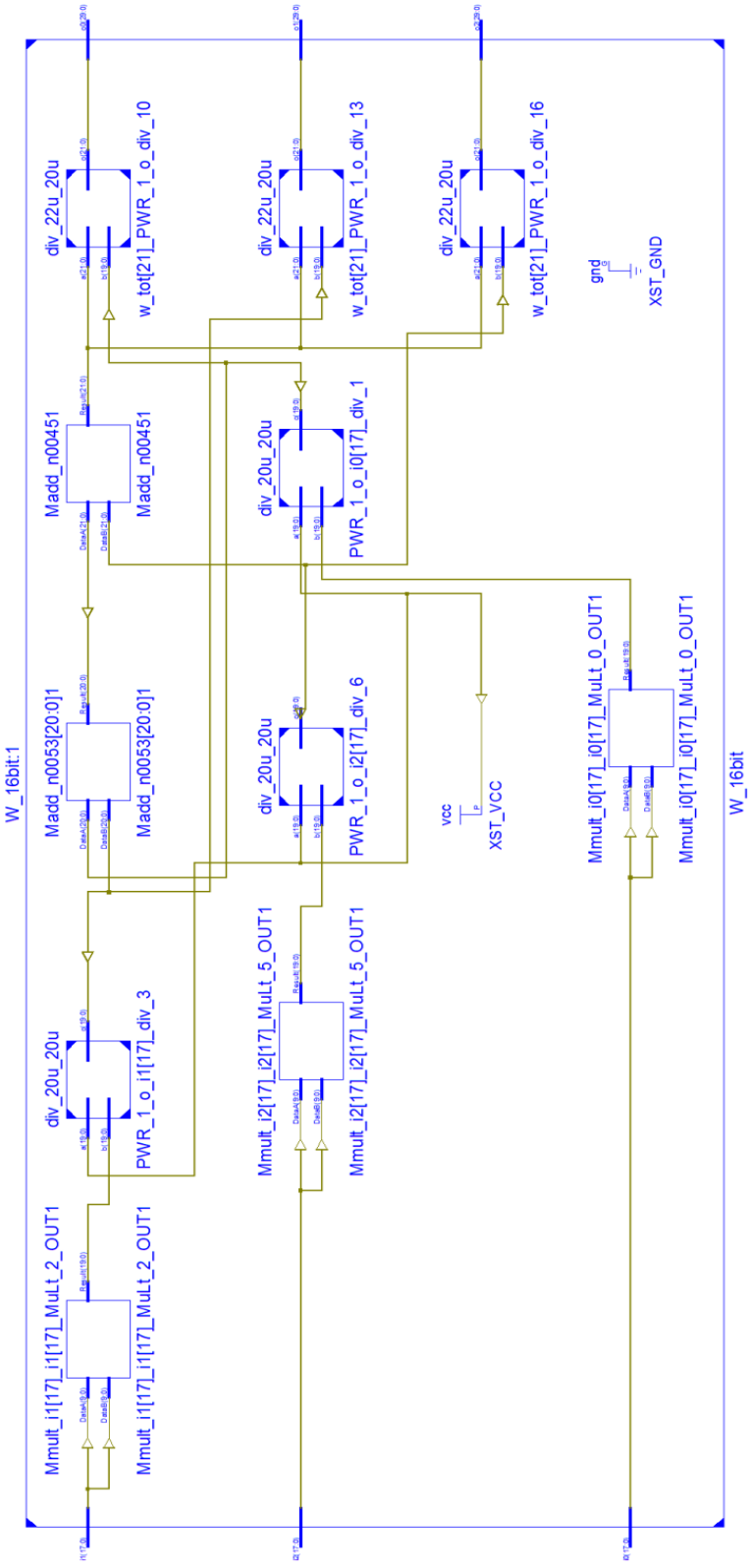


Figure 5.7 : Inner RTL schematic weighted coefficients

6. SOFTWARE IMPLEMENTATION

6.1. Molnar Wagner RFID Authentication Protocol

Molnar and Wagner's mutual authentication protocol is initially intended to be applied on library RFID system. However due to its simplicity and hence the effectiveness by means of speed, it can be applied various types of application and localization systems are one of them. Especially algorithm's simplicity is make it able to be used in systems having time constraints like real time localization systems. Protocol steps is explained in detail in chapter 4. In this chapter software implementation steps of The Molnar Wagner Mutual Authentication Protocol is presented. During the implementation steps shown in figure below is followed.

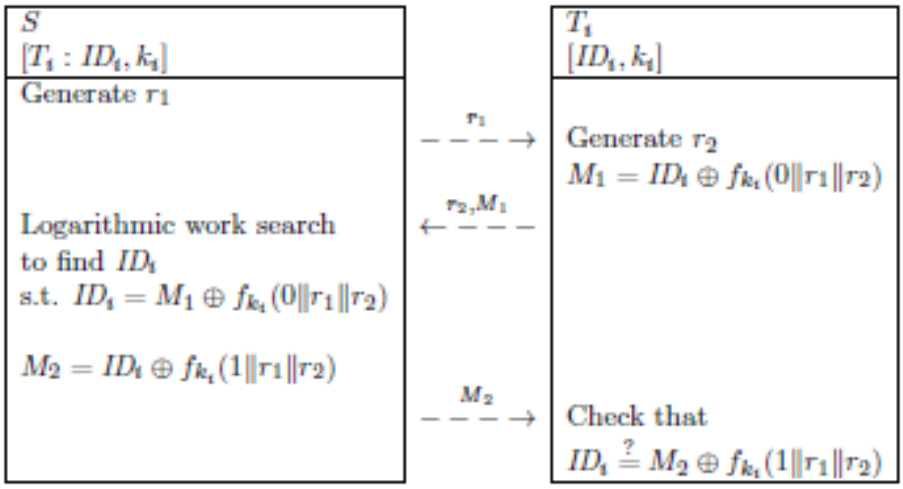


Figure 6.1 - Steps of Molnar Wagner Algorithm [23]

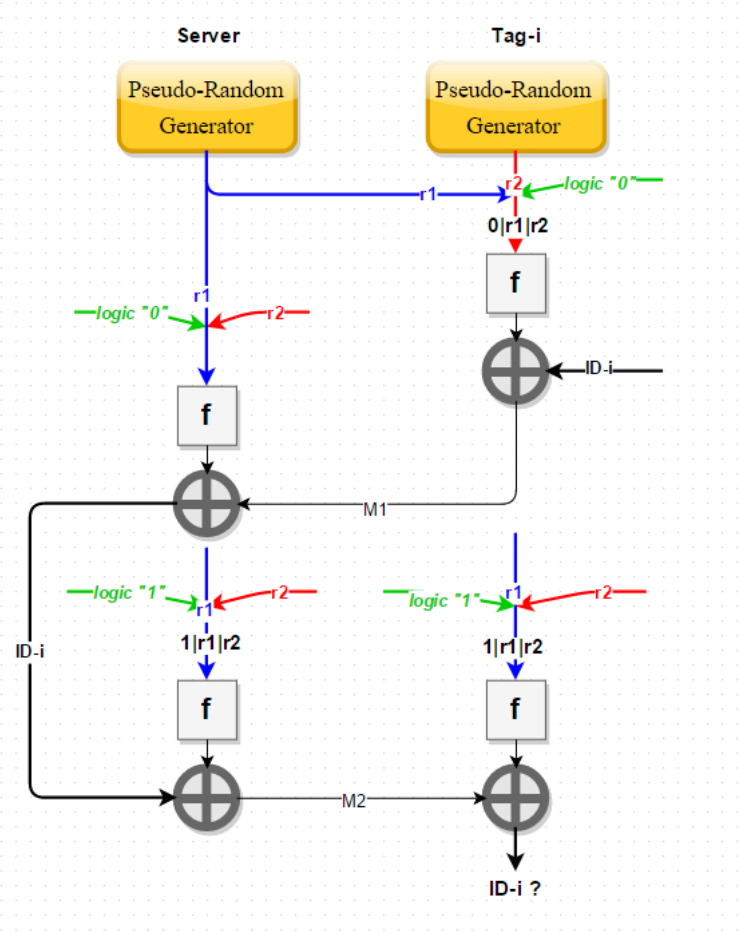


Figure 6.2 : Block diagram of Molnar Wagner Algorithm

6.2. Simplified Data Encryption Standard Algorithm

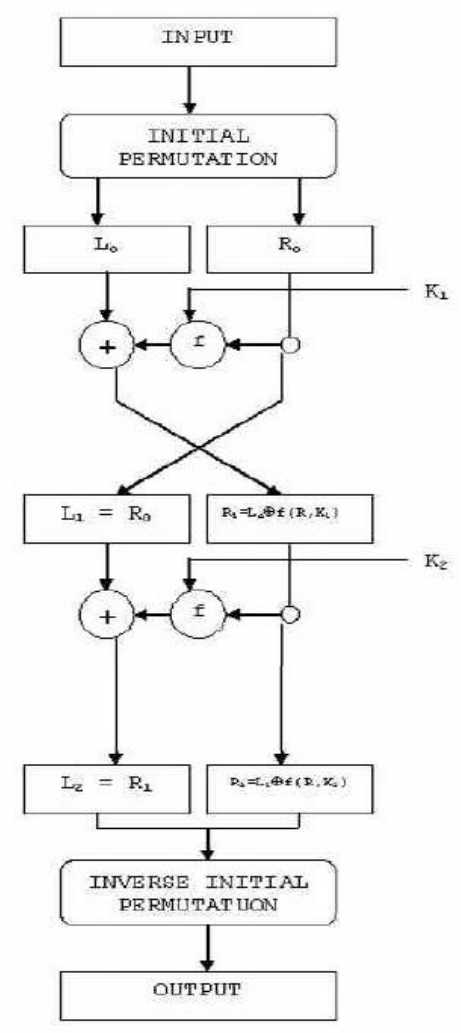


Figure 6.3 : Block diagram of SDES [20]

5.2.1 Key Scheme

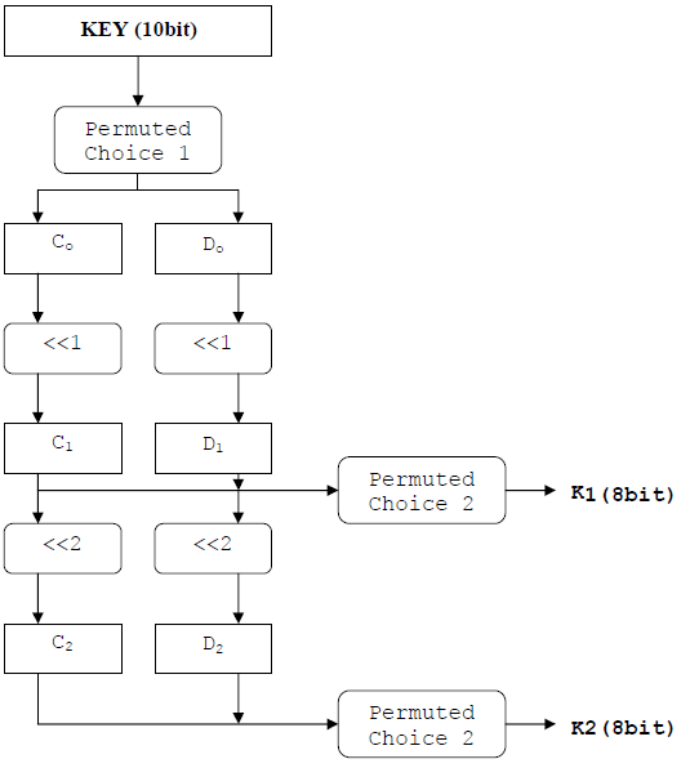


Figure 6.4 : Block diagram of SDES's key scheme [20]

5.2.2 The Round Encryption Block

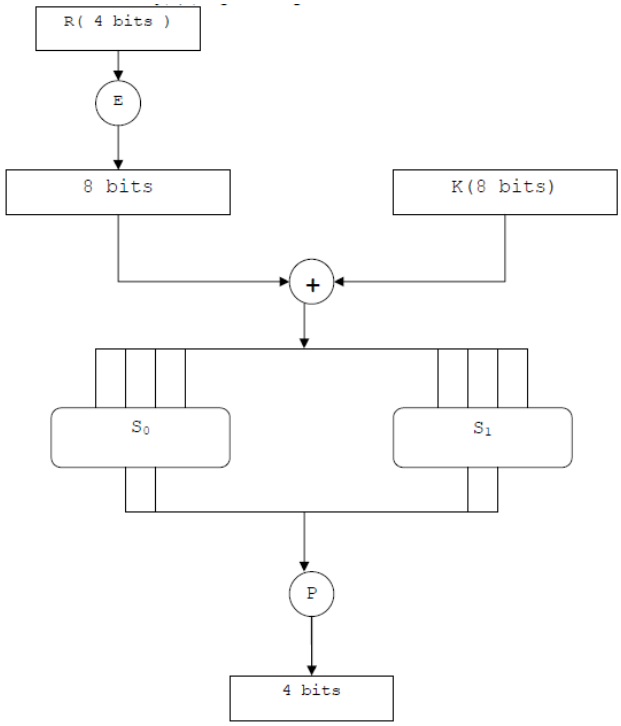


Figure 6.5 : Block diagram of SDES's encryption function [20]

	S_0 Column Number			
Row No.	0	1	2	3
0	1	0	2	3
1	3	1	0	2
2	2	0	3	1
3	1	3	2	0

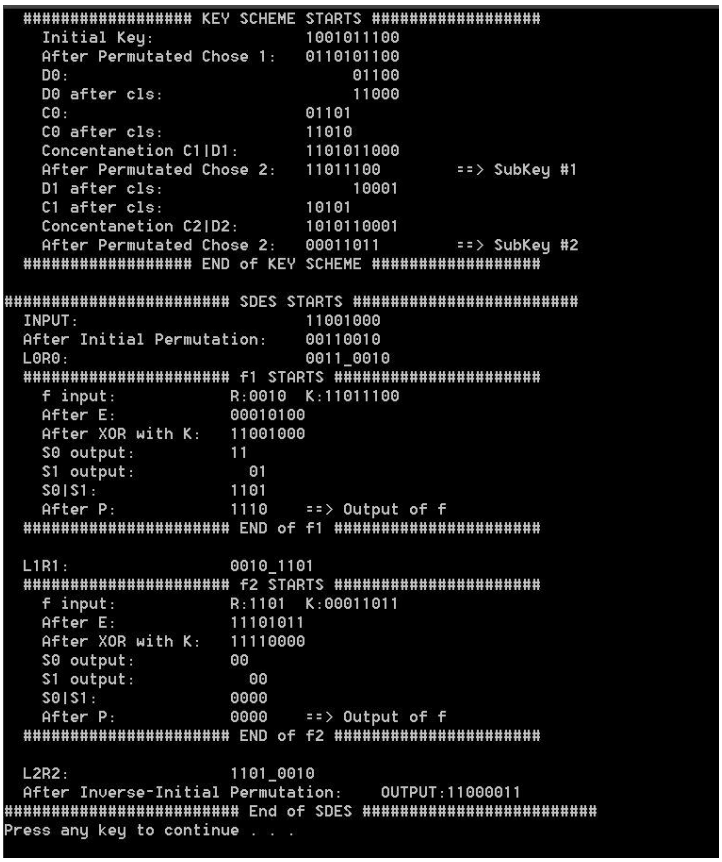
	S_1 Column Number			
Row No.	0	1	2	3
0	0	3	1	2
1	3	2	0	1
2	1	0	3	2
3	2	1	3	0

Figure 6.6 : S-Boxes for SDES implementation [20]

7. CONCULISIONS AND RESULTS

7.1. LANDMARC Localization Algorithm

7.2. Molnar Wagner RFID Authentication Protocol



REFERANCES

[1] **Zekavat, S. A., Kansal S., Levesque A. H.,** *Fundamentals of Position Localization.* John Wiley & Sons, 2012.

[2] **Violino, B**, "RFID Business Applications," *RFID Journal* Http://www.Rfidjournal.com/article/view/1334, **2012**(10/9) .

[3] "Project 12011 BaaS - Building As A Service" https://itea3.org/project/baas.html

[4] "BaaS Project" https://www.baas-project.eu/

[5] **Ni, L.M.; Yunhao Liu; Yiu Cho Lau; Patil, A.P.,** "LANDMARC: indoor location sensing using active RFID", Pervasive Computing and Communications, 2003. (PerCom2003). Proceedings of the First IEEE International Conference on 23-26 March 2003 Page(s):407 – 415

[6] **Molnar D., Wagner D.,** "Privacy and security in library RFID: issues, practices, and architectures, "Conference on Computer and Communications Security CCS'04, 2004, pp. 210-219.

[7] **Weis, S. A.,**

[8]

[9]

[10]

[11] **Chu, Pong P.,** 2008. FPGA Prototyping by Verilog Examples. Wiley-Interscience, New Jersay.

[12] **Xilinx**, 2007. MicroBlaze Processor Reference Guide.

http://www.xilinx.com/tools/microblaze.htm

[13] **Xilinx,** 2007. Embedded System Tools Reference Manual.

[14] **Xilinx,** Software Development Kit Help Contents, [Reference Date: 3 May 2015], http://www.xilinx.com/support/documentation/sw_manuals/xilinx12_2/SDK_Doc/index.html.

[15]

[16] **Deza, E., Deza, M. M.,** *Encyclopedia of Distances*. Springer. p. 94.

[17]

[18]

[19]

[20] **K. S. Ooi, B. C. Vito**, "Cryptanalysis of S-DES", University of Sheffield Centre, Taylor's College, April 1, 2002.

[21]

[22]

[23] **B. Song,** "RFID authentication protocols using symmetric cryptography," University of London, Department of Mathematics, Technical Report RHUL-MA-2009-24, December 16, 2009.