FPGA Implementation of Novel Hybrid RSA and DSA based Diffie Hellman Key Exchange Algorithm

*P. Vijayakumar[1], Sujan Krishna[2], R Yuvaraj[3], Harivamsi Reddy[4], Polineniramakrishna[5], Rajashree R[6], Ananiah Durai[7]

[*1,7]Associate Professor, Vellore Institute of Technology, Chennai, Tamilnadu, India

[2,4,5]Undergraduate Students, Vellore Institute of Technology, Chennai, Tamilnadu, India

[3]Assistant Professor (SG), Institute of ECE, Saveetha School of Engineering, Chennai,Tamilnadu,India

[6]Research Scholar, Vellore Institute of Technology, Chennai, Tamilnadu, India

Abstract: The internet is reaching a new level every day, in the last couple of decade's data security has become a main concern for anyone who has been using the internet. The data sharing is vigorously growing every day in which data security plays a crucial role in daily life. In this modern-day of digitization, it is necessary to protect data from misuse, falling into wrong hands, and exploitation, where this data may range from important user credentials to bank account information, to logs of a company, etc. As the data is shared or transmitted enormously there is a significant increase in the importance of data privacy as well as security, thus one has to focus more on ways to secure and strengthen our way of communication. One way to achieve this is cryptography. In order to connect securely and quickly for electronic data transfer through the web or for secure bank transactions, etc the data should be encrypted. Encryption is the process of transforming plain text into ciphered-text, it a type of text which is not understandable or altered easily by undesirable people. Traditional Diffie Hellman key exchange is susceptible to man in the middle attack. This is the main disadvantage of the traditional algorithm, to overcome this problem a new RSA based Diffie Hellman key exchange with digital signature algorithm have been proposed and implemented through FPGAs. In the proposed algorithm the problem of man in the middle attack is solved. The proposed algorithm is more useful in case of client and server-based interaction like in case of wireless mobile internet(4G/LTE) usage by an individual and in case bank transactions that are made by the individuals. Security, resistance to side attacks and collisions, larger key size, non-factorization of the prime numbers are the advantages of the RSA which is used in the proposed algorithm. Synthesis and implementation of the encrypted block have been compared and analyzed on Spartan 3, Spartan-6e, Virtex-4, and Spartan-6 FPGA boards. Complete security is achieved using digital signature and previous problems of the traditional algorithm.

Keywords: Digital Signature, Authentication, Cryptography, RSA, Diffie Hellman, FPGA.

I. INTRODUCTION

In the present day of digital life almost everything is being digitalized, there is a tremendous growth of data transfer through online. Data can of any type and form for example it can be messages, Photos, Videos to Bank transactions. In this modern-day of digitization, it is necessary to protect data from misuse, falling into wrong hands, and exploitation. In order to connect securely and quickly for electronic data transfer through the web or for secure bank transactions etc, the data should be encrypted. Encryption is the process of transforming plain text into ciphered-text, it a type of text which is not understandable or altered easily by undesirable people. Hence, security solutions that strengthen data privacy and confidentiality, boost scalability, reduce computational power are desired. Single layered security can be compromised whereas end to end encryption of data is of utmost importance and this can't be compromised. The RSA encryption algorithm, is resistant to collision attacks, easily computable, larger key size, hard to break, and has been used to provide the first layer of encryption for the model. Basic digital signature algorithm has been used to develop the second layer of encryption. Digital signatures can provide three important security services Integrity, Non-repudiation, and Authentication. The proposed algorithm provides a secure way for key exchange compare to the traditional Diffie Hellman key exchange algorithm. This allows us to upload sensitive data that remains protected, cannot be modified or altered by any other party as all the transactions are public. Hardware implementation, however, concentrates on optimizing power, area, energy, computational speed.

Symmetric key cryptography or symmetric encryption is a type of encryption standard in which the same key is used for encryption as well as for decryption of messages. There are five major components in the symmetric encryption system. The components are secret key, Plaintext, decryption algorithm, encryption algorithm,

ciphertext,. This cryptosystem is more efficient than asymmetric key cryptosystem (Public-key cryptography) and therefore preferred when large amounts of data need to be exchanged. Establishing the shared key is difficult using only symmetric encryption algorithms, so in many cases, an asymmetric encryption is used to establish the shared key between two parties. Examples for symmetric key cryptography include Analog Encryption Standard (AES), Digital Encryption Standard (DES), and 3DES. Key exchange protocols that are used to establish a shared key between the sender and receiver include Diffie-Hellman (DH), elliptic curve (EC), and RSA.

Public key cryptography or Asymmetric key cryptography uses two keys a public key that is known to everyone and a private key or a secret key which is known only to the recipient of the message or a signal. This cryptography requires the owner to protect their own private key or secret key while public keys are not secret at all and can be made available to the public. There are three types of families of public-key cryptography. First of all the system that is based on integer factorization. RSA cryptographic system, this is named after the inventor's Rivest, Shamir, and Adleman is one of the most popular crypto-system. Secondly, the system which is based on the discrete logarithm. These algorithms mentioned do provide support for digital signature, (DSA) and key agreement (Diffie-Hellman) as well. Third, the system that is based on arithmetic using elliptic curves is called Elliptic curve cryptography. It is a new family of public-key algorithms. This provides shorter key lengths and depending on the environment and application in which it is used it can provide improved performance over other systems based on integer factorization and discrete logarithms.

The main objective of the work focuses on implementing an efficient RSA based Diffie Hellman Key exchange using Digital Signature algorithm for secure key exchange between two systems. As per the security requirements, the following objectives were formulated:

- Implementation of RSA based Diffie Hellman algorithm to solve the problem of factorization of Diffie Hellman Key. RSA key pair is generated to solve this problem.
- To use SHA-1 code and generate hash values and analyze the data
- Implementation of a digital signature algorithm as this is a solution to the problem of the traditional Diffie Hellman algorithm which is a man in the middle attack.
- To compare the statistics by implementing the code on different models of FPGA like Vertex 4, Vertex 5, and Spartan 6.

The paper has been organized in such a way that each chapter deals with the major concepts and contributions involved in the research work. In section I, there's a brief introduction about the project that been implemented and types of cryptography i.e. symmetric and asymmetric key cryptography.it also show sthe organization of the paper.Section II shows the literature survey of the base and reference papers.Section III describes the basic types of encryption and hashing algorithms, Advantages and disadvantages of encryption and hashing algorithms and also it gives the reason for being chosen for implementation. Section IV proposes the design and implementation method of RSA based Diffie-Hellman key exchange using the digital signature.Section V gives the summary of the entire work is presented with the comparison of different FPGAs. It also shows the results and its discussion. Section 6 gives the conclusion and future opportunities.

II. Related Works

Turan et., al., [10] proposed "Compact and Flexible FPGA Implementation of Ed25519 and X25519". In this article they have briefly described about FPGA implementation of the Elliptic curve based ED25519 digital signature. They have also described field-programmable gate array cryptographic architecture with the combination of the above-mentioned algorithm. They compared the implementation of Ed25519 and X2551. The key establishment schema of X25519 is briefly explained in this publication. The main goal of this publication is to synthesize a lightweight FPGA module where a proposed algorithm is optimized both in terms of memory consumption and reduce the time taken for execution of the process. This is especially useful for Internet Of Things based applications. To reduce the time of execution i.e. for faster execution they utilized the maximum use of DSP-Digital Signal Processing slice on a Field Programmable gate array. An arithmetic unit has also been proposed along

with the algorithm where this unit supports all the operations with no modulus arithmetic and with two modulus arithmetic, with this proposed arithmetic unit they have eliminated the memory access bottleneck. The proposed algorithm is implemented in a single module. To implement the above algorithm, it required only 2.6k registers, 16 DSP slices, and 11.1K Lookup Tables. The implemented algorithm takes only 1.6ms for the generation of digital signature and signature verification it takes only 3.6ms time. The time mentioned above is the time taken for a 1024 bit message which is operated at 82MHz clock. Their design targeted 7-series Xilinx field-programmable gate array boards.

Zhang, J. et., al.,[2] proposed "Recent Attacks and Defenses on FPGA-based Systems". This publication is a survey of the attacks on Field-Programmable gate array-based systems and defenses on these systems too. A detailed survey on the trust issued on Field programmable gate array from the developer's point of view, from the vendor's point of view as well as from consumers point of view is presented. In this survey it is briefly explained about the trust problems that one faces who used field-programmable gate array-based systems and the available solutions to such problems that one must be aware of. The opportunities of Field programmable gate array-based systems that are used in large scale like in date centers and some challenges in such cased is also presented. They presented the demand and supply flow chain in which it starts with the FPGA vendors who introduces new FPGA boards into the market to the customized FPGA chip fabrication. They explained the market model of Field Programmable Gate Array. This model consists of three major vendors namely FPGA vendors, IP core vendors, EDA tool vendors. Foundries, Field Programmable gate array-based system developers, and end-users are also a part of this market model. The security and trust issued that are face like side-channel attack, cloning attack, reverse engineering, Trojans, etc. are also briefly presented in this publication. The opportunities and challenges in this field in the recent times is also discussed.

Mitra et., al., [31] proposed "An FPGA-Based Phase Measurement System". In this publication they proposed a new sensitive phase measurement system with more accuracy and precision compared to the conventional method. This proposed method has the resolution and precision in the range of few picoseconds. This proposed model is modularized in such a way the one can replace the module of this proposed method according to the use of the designer for more robustness of the proposed design. The high-speed transceivers that are present in the field-programmable gate array do not ensure the chip latency after each reset cycle, after every power cycle. The chip latency may vary even when the firmware of the FPGA is updated, to overcome this a new phase measurement system is proposed. The working principle of this phase measurement system is based on subsample accumulation. This is done by taking the samples systematically over the phase detector. To illustrate the working principle of this phase measurement system a mathematical model has been developed and demonstrated with that. This measurement system can be operated over a wide of FPGA with a wide range of clock frequencies ranging from few kilohertz to the maximum frequency that is supported by the Field Programmable Gate Array board that is selected. They also made the performance comparison on different FPGA boards like Intel Cyclone IV FPGA, Stratix V and Arria 10. The resolution changed from 8-10 picoseconds based on the FPGA board.

M.Khalil et., al., [4] published a paper in the name of "Field Programmable Gate Array Implementation of SHA-2 Hash Function that is used for the generation of Digital Signature". This SHA – 2 hash function is implemented in the Altera nios II Stratix FPGA board. The importance of the hash function in the present day of digital life is explained and it is said that it is the building block for most of the secret key message authentication codes. In this publication the Advanced Encryption Standard is briefly explained and its importance too. The comparison is made between many SHA algorithms like SHA 224, SHA 256, SHA 358, etc. and it is also mentioned that the above-mentioned hash functions match the security standard of AES by National Institute of Standards and Technology. The process of generating digital signature with the use of SHA – 2 hash function and the architecture of SHA – 2 is also explained briefly in this publication. The time taken by the above-mentioned hash functions is also compared. The hash value that is generated by passing the same message to different hash functions that are mentioned above is also displayed.

Jeppesen, et., al., [32] proposed to enhance the functional safety in Field Programmable gate array-based motor drivers. The FPGA has a wide range of applications ranging from electronics to mechanical systems like motor drivers. In this article the application of FPGA in hybrid electric vehicles has been explained. The methods that are involved in the presented algorithm are also described this involves the usage of the FMEDA tool which is developed by Intel. This tool has features like calculation of false metrics like diagnostic coverage (DC), Safe fail fraction (SFF), etc. The FDMEA flow to analyze a design involves Architecture, Safety separation, Resource usage

extraction, Add safety mechanism and calculate metrics. The architecture dual-core lockstep nios and full redundancy is also explained briefly in this article. With the help of FMDEA toll changed fault metrics as new diagnostic were added to the proposed designChen, D et., al., [29] proposed "Introduction to the Special Section on Deep Learning in Field-Programmable Gate Arrays's". This article described the advancement of Deep Learning, more via Deep Neural Network. This is because this DL has the capability of exceeding human capabilities in many tasks like large scale information retrieval, in playing complex games, and also in image recognition. Field programmable gate array is highly efficient in interfacing with Deep Learning. FPGA has the flexibility of reusability. The comparison of FPGA with ASIC is also presented in this paper. The design methodologies and architecture of Deep Learning is also briefly explained in this article.

Hoque, T., et., al., [27] proposed a method that acts as a countermeasure to the FPGA bitstream tampering attack. In this paper[27] brief description of bitstream tampering is presented. There are many types of modifications in this proposed method one of such modification is Rule-Based modification. An attacker can do random modification in the bitstream. The existing solution to may bitstream attacks is also explained in this publication. They have achieved a solution to secure the field operation of FPGA. This operation is done in the presence of a bitstream tampering attack. It has been proved that the proposed method is more efficient than most of the encryption-based solutions to such attacks. They also proved that by keeping the overhead low they can increase the complexity of the obfuscation. Lam, S.-K et., al., [28] proposed "Rapid Evaluation of Custom Instruction Selection Approaches with FPGA Estimation". The main aim of the proposed method is to show that the efficient and fast FPGA estimation engine is indispensable. They have proposed a novel method based on clusters generation strategy. This is a new strategy in which the custom instructions are divided into a set of clusters so that they can be mapped effectively. The proposed model is coded using VHDL. The proposed model is simulated and synthesized with Xilinx ISE version 11.2. The field-programmable gate array board that is used for this implementation is Xilinx Vertex 4. There is a detailed explanation of speedup evaluation with hardware estimation as well as speedup valuation with area constraint. The delay-area is illustrated with an example and experimental results are also displayed in this publication.

Farooq et., al., [30] proposed "An Inter-FPGA Routing Exploration Environment for Multi-FPGA Systems". There has been a tremendous increase in the FPGA based applications in the electronic industry. In the paper they have presented a novel and efficient way of inter-routing exploration systems for multi-FPGA prototyping systems. They have proposed three different routing approaches on FPGA. These routing approaches are implemented on two FPGA boards. The comparison of different proposed routing approaches by the use of custom field-programmable gate array is presented in this article. It is been concluded that for the proposed three different approaches custom boards give 9.7% better frequency results when compare to the generic field-programmable gate arrays that are available in the market. Its also concluded that 8.3% better frequency results compare to the multiterminal routing approach for the field-programmable gate array that are present in the market. Lonla Moffo et., al., [33] proposed "A Novel Digital Duty-Cycle Modulation Scheme for FPGA-Based Digital-to-Analog Conversion". In this paper they presented a novel and efficient Digital to Analog Converter for FPGA based system. This paper presents the fast DDCM schema outline. The fundamental basis of digital duty cycle modulation as well as the proposed dual digital duty cycle modulation for digital to analog converter is explained in this article. The proposed digital part is embedded into a field-programmable gate array chip which consists of lookup digital duty cycle modulation. It also consists of dual DDCM counting service with holder and interrupt service.

Saleh Saraireh [34] proposed "A Secure Data Communication System Using Cryptography and Steganography". This article mainly focused on securing the way of communication in daily life. They presented a unique and novel method by combining both stenography and cryptography to encrypt the secret messages. To encrypt the secret message they used Filter bank cipher as it provides high security. This filter bank cipher is highly scalable and operates at a high speed. This encrypted message is embedded into a cover image, this is done by changing the coefficients of wavelet. The performance of the proposed system is checked with a peak signal to noise ratio. A comparison has been made between histogram analysis of stego image and cover image. It is concluded that the proposed system is more secure and is resistant to the attacks. Fard, et., al., [35] proposed "FPGA-Based Bit Error Rate Performance Measurement of Wireless Systems". This article focuses mainly on developing the bit error rate tester for performance validation of wireless communication systems on field-programmable gate based system. The bit error rate tester structure is explained briefly in this article. The simulation results and the hardware implementation of the proposed model is presented in this publication. This bert i.e bit error rate tester is integrated with a typical wireless communication system for more accurate results. The bert that is developed on a field-

programmable gate array for repeated and accelerated tests and this reduces the need for many simulation software tools, this results in productivity. It is developed on a field-programmable gate array based system instead of the ASIC system because the FPGA's are cost-effective and flexible to develop. It has been concluded that the proposed model is more robust and efficient.

Fang, D et., al., [40] proposed "Security for 5G Mobile Wireless Networks". The has been rapid growth in mobile internet usage as well as in the generations of mobile wireless networks. This publication is a survey on 5G networks its security, potential attack etc. This article described the importance of data integrity, data confidentiality and privacy in multiple communication like device to device communication systems, multiple-input or multiple-output software networks etc. A detailed study on data integrity, privacy, and confidentiality of data has been presented in this paper. The challenges that one may face in the usage of G wireless networks and the solution to the common attacks that one must be aware of is presented. The new security attack models and advancements in the 5G network are also presented. Han, C.-K et., at., [39] proposed "Security Analysis of Handover Key Management in 4G LTE/SAE Network"**.** Security plays a vital role in wireless mobile communication. The goal for this proposed system is to not jeopardize the entire network if a one-person device is compromised in the network. To achieve they proposed handover key model in which changing the root key or updating it periodically are an integral part of this key management system. The goal of LTE/SAE is to move wireless mobile networks into the fourth generation. A detailed description of EPS security, as well as security analysis for the proposed system, is also presented in this article. The effect of root key updates in the vulnerable period is also explained. It is concluded that selecting the optimal key in period intervals can make the device more secure.

Kim, J et., al., [6] proposed "Comprehensive Study on mmWave-based Mobile Hotspot Network System for High-Speed Train Communications". The main goal of this article is to present a study on mmWave based MHN for HST communication. A detailed explanation of MHS for HST has been presented along with the mobile hotspot network enhanced physical layer specification details. To simulate the proposed system a computer simulation is done at a mobility rate up to 500kmph. The study on the MHN system prototype design is also presented in this publication. The performance comparison includes the Wi-Fi data rate, as well as throughput along the test route and signal to noise ration at downlink, received It has been concluded that the proposed system i.e. MHN-E is capable of transmitting 5Gbps at a speed of 500Km/h. Rupprecht et., al., [37] proposed "On Security Research Towards Future Mobile Network Generations". This article describes the possible security attacks that may occur in the upcoming 5G network which is under development. It has a detailed explanation of various generations of networks namely GSM (2G) – Global System for Mobile Communication, UMTS (3G) – Universal Mobile Telecommunications Systems, and LTE (4G) – Long Term Evolution. The security requirements such as data confidentiality, availability, and system integrity for the 5G wireless network is also briefed in this publication. The wireless network is open and it is accessible by everyone, thus can be easily exploited by jamming attacks countermeasure for such attacks are also described in this article.

Rashwan, A. M et., al., [36] proposed "Characterizing the Performance of Security Functions in Mobile Computing Systems"**.** After the development of next-generation mobile networks the transmission of information will be in a sophisticated way to improve the security and privacy of the data. In this article they presented a new benchmark system for cryptography-based function. It is important that one has to understand the computational characteristic from the communication point of as these is a tremendous increase in the computational complexity of most of the cryptography-based security systems. This paper presents a detailed explanation on the performance evaluation of most of the security functions that are used in the communication systems. These functions consist of hashing, keying, encryption, and decryption. The benchmarking analysis is compared based on the different bit sizes on multiple processors using different software is presented.

III. PROPOSED HYBRID RSA AND DSA BASED DIFFIE HELLMAN KEY EXCHANGE ALGORITHM

The proposed algorithm consists of both RSA and SHA1 to provide key pair for key generation and authentication of data. The key pair is generated by the RSA algorithm which is required for the key exchange process. Authentication is achieved by the SHA-1 hashing function which generated a 160-bit unique hashed output. The message that is passed to the hash function is key that is generated by the RSA. This acts as the digital signature which solves the problem of man in the middle attack which is in the traditional Diffie-Hellman algorithm, which helps us store and safeguard the data without any third-party alterations.

The following steps are involved in the proposed algorithm for the generation of the key pair namely public key and private key for the exchange of the key in Diffie-Hellman and thereby finding the hash value of the key that is generated by the RSA. In order to generate key pairs choose two prime numbers.

- Find the phi value that is the product of p-1 and q-1 where p and q are the prime numbers that are chosen.
- Choose the private key such that it has to be in been 1 and n where n is the product of prime numbers.
- Find the public key with the formula E*D mod n =1. The hash value is generated by passing the public key value to the hash function that is SHA-1. The hash value that is generated as the output of the SHA – 1 is used for digital signature. This hash value is concatenated with the message key that is needed to be exchanged and sent to the receiver.

At the receiver side the message that is received which is concatenated with the hash value is separated. The message that is received is again sent to the hash function SHA-1 which generated the hash value. The hash value that is generated and the hash value that is received is compared and check if both are the same. If both the hash values are the same then the process key exchange is successful and is confirmed that it is sent by the sender (Alice). If the hash values are not matched then the process of key exchange is again initiated with the generation of new key pairs and the process is continued as mentioned above. We have demonstrated the project by simulation and analyzed the RSA algorithm, DH algorithm, and SHA1 function on a software tool which enables the user to select different target devices and configure those devices according to the requirement. The software tool used is Xilinx ISE Design suite version 14.7. Thereafter we combined all the algorithm to obtain the desired output made a module named rsa_top. The target device vertex 5 is well supported by the Xilinx ISE Design suit 14.7. The technical specifications are vertex 5 family, the package we used is FT256, speed grade selected is -4, The type of Top-level source HDL, Synthesis tool - XST (VHDL/Verilog). A further comparison was made between the statistics of Virtex-4, Vertex-5, and Spartan-6 Field Programmable Gate Array's to compare the slice registers, LUT-FF, and check for the best possible architecture, with least resources and memory utilization.

Step 1: Select two prime numbers A and B for the generation of key pair namely public key and private key.
Step 2: Select the public number P = 23 which is know to all in Diffie-Hellman algorithm
Step 3: Calculate the pmitive root and let it be G = 5 for Diffie Hellman algorithm.
Step 4: Calculate N with the formulate N = A*B.
Step 3: Calculate phi value with the formulae $\varphi(N) = (A - 1) * (B -1)$.
Step 4: Select the value of E such the E is coprime to $\varphi(N)$ and $0< E< \varphi(N)$ i.e $GCD(E, \varphi(N))=1$.
Step 5: After the selection of E calculate the value of D with the formulae
$\Rightarrow D = ((k* \varphi(N))+1)/E$ for any random value of k.
Step 6: Pass the values of public key and private key namely E and D to Diffie Hellamn algorithm as n1 and n2 as these are the random numbers that are to be chosen for key generation.
Step 7: Calculate the value of X with the formulae
$\Rightarrow X = G^{n1} \bmod P$
Step 8: Calculate the value of Y with the formulae
$\Rightarrow Y = G^{n2} \bmod P$
Step 9: Pass the value of X to the SHA 1 hash function and store the hash value that is generated.
Step 10: Concatenate the hash value along with the the value of X that is to be exchanged with the receiver and let it be X1.
Step 11: Exchange the value X1 of with Y i.e send X1 to receiver and receive Y from receiver.
Step 12: Slice X1 in to two parts and store the hash value that is received in another variable let it be h1 and value of X in X2.
Step 13: Calculate the hash value of X2 i.e pass the value of X2 to SHA-1 hash function and store the hash value in another variable let it be h2.
Step 14: Compare the hash value h1 and h2, if both are equal then continue to follow bellow steps if the hash value is not equal abort the process and start from begning.
Step 15: Calulate the value of K_a with the formule
$\Rightarrow K_a = Y^{n1} \bmod P$.

Step 16: Calulate the value of K_b with the formule

$\Rightarrow K_b = X1^{n2} \bmod P$.

Therefore the share value is $K = K_a = K_b$

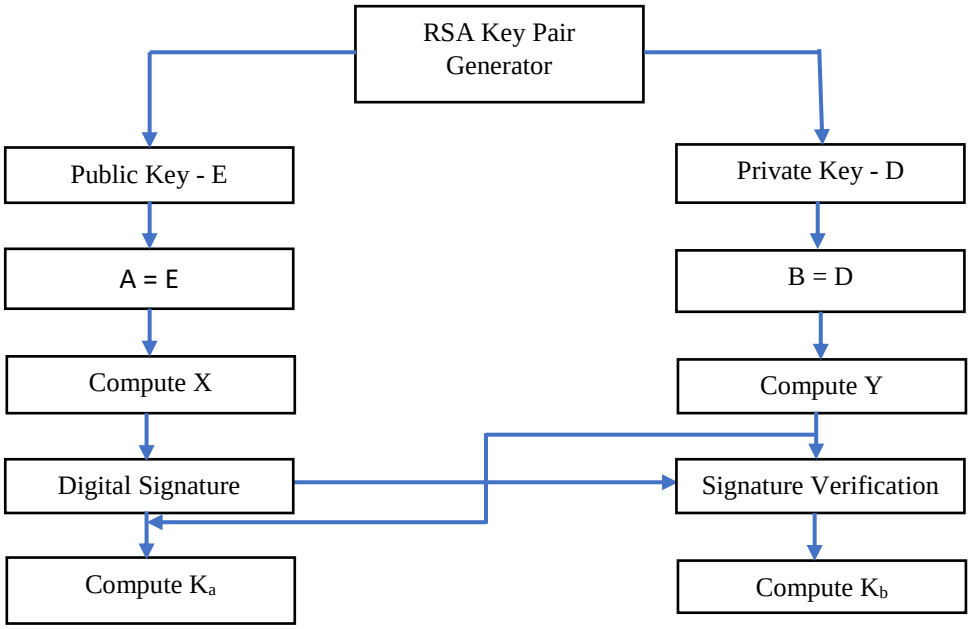

Figure 1 Proposed Hybrid RSA based Diffie Helman Key Exchange Algorithm

The figure above is the block diagram it represents the flow of data in the proposed algorithm. The key pair is generated from the RSA key pair generator and the public and private keys are passed as a random values to the DH algorithm. In this algorithm before exchange process digital signature algorithm is used, so that it acts as a counter to man in the middle attack.

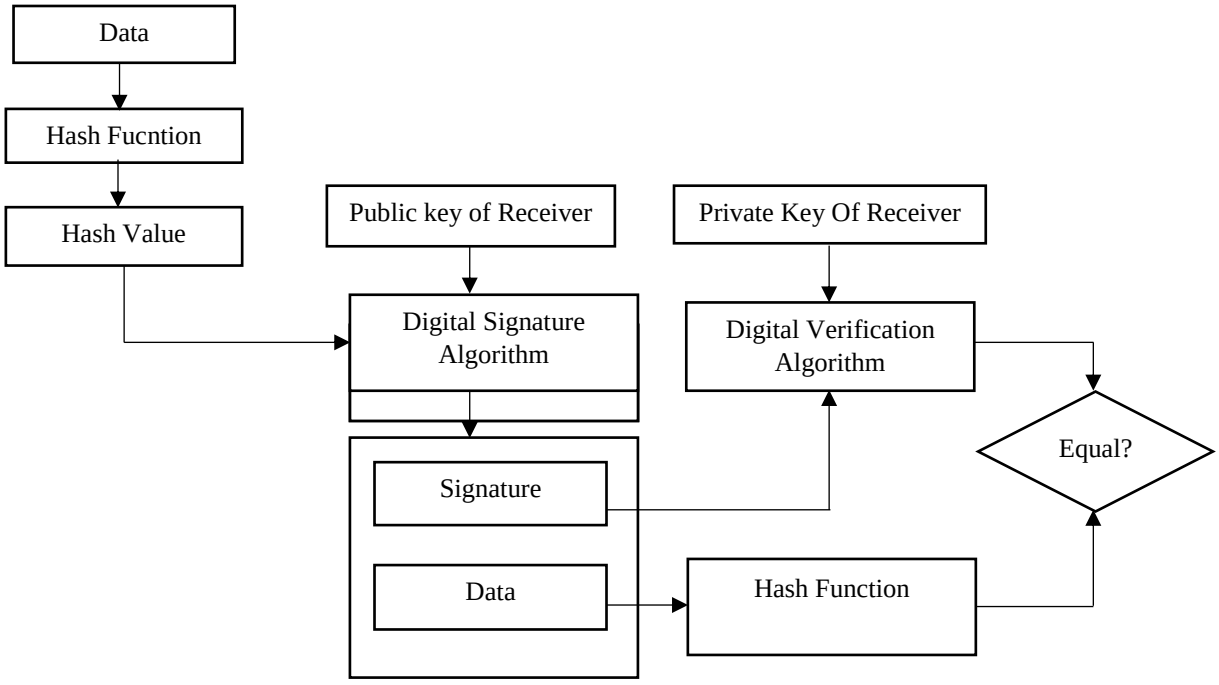

Figure 2 Block diagram of Digital Signature Algorithm

Typical example:

Step 1: Select two prime numbers A and B for the generation of key pair namely public key and private key let it be 17, 19

Step 2: Select the public number P = 23 which is known to all in Diffie-Hellman algorithm

Step 3: Calculate the primitive root and let it be G = 5 for Diffie Hellman algorithm.

Step 4: Calculate N with the formulate N = A*B.

$\Rightarrow$ N = A * B = 323

Step 3: Calculate phi value with the formulae $\varphi(N) = (A - 1) * (B -1)$.

$\Rightarrow \varphi(N) = (A - 1) * (B -1) = (17-1) * (19-1)$

$\Rightarrow \varphi(N) = 288$

Step 4: Select the value of E such the E is coprime to $\varphi(N)$ and $0< E< \varphi(N)$ i.e GCD(E, $\varphi(N)$)=1.

$\Rightarrow$ E = 5

Step 5: After the selection of E calculate the value of D with the formulae $D = ((k* \varphi(N))+1)/E$ for any random value of k. Let it be 8

$\Rightarrow D = ((k* \varphi(N))+1)/E = ((8*288)+1)/5$

$\Rightarrow$ D = 461

Step 6: Pass the values of public key and private key namely E and D to Diffie Hellamn algorithm as n1 and n2 as these are the random numbers that are to be chosen for key generation.

$\Rightarrow$ E = n1 = 5

$\Rightarrow$ D = 461 = n2

Step 7: Calculate the value of X with the formulae $X = G^{n1} \bmod P$

$\Rightarrow X = G^{n1} \bmod P = 5^5 \bmod 23$

$\Rightarrow$ X = 20

Step 8: Calculate the value of Y with the formulae $Y = G^{n2} \bmod P$

$\Rightarrow Y = G^{n2} \bmod P = 5^{461} \bmod 23$

$\Rightarrow$ Y = 14

Step 9: Pass the value of X to the SHA 1 hash function and store the hash value that is generated. Let the hash value is stored in h.

The words generated for message X is as follows

Word[0]=32308000	Word[1]=0
Word[2]=0	Word[3]=0
Word[4]=0	Word[5]=0
Word[6]=0	Word[7]=0
Word[8]=0	Word[9]=0
Word[10]=0	Word[11]=0
Word[12]=0	Word[13]=0
Word[14]=0	Word[15]=10
Word[16]=64610000	Word[17]=0
Word[18]=20	Word[19]=c8c20000
Word[20]=0	Word[21]=40
Word[22]=91840001	Word[23]=20
Word[24]=c8c20080	Word[25]=23080003
Word[26]=0	Word[27]=100
Word[28]=46100006	Word[29]=a0
Word[30]=ebca0203	Word[31]=8c20006c
Word[32]=59460001	Word[33]=400
Word[34]=18400099	Word[35]=23080283
Word[36]=af28080f	Word[37]=308000b2
Word[38]=23080003	Word[39]=1080
Word[40]=42080067	Word[41]=a00
Word[42]=bca0203e	Word[43]=c2000688
Word[44]=5e40014	Word[45]=4080
Word[46]=a7080912	Word[47]=138829f1
Word[48]=61c80fe	Word[49]=8000b23
Word[50]=308002b2	Word[51]=af29088f

Word[52]=3880677 Word[53]=a400
Word[54]=d24203f2 Word[55]=20006a0c
Word[56]=f168094f Word[57]=308408b2
Word[58]=53889129 Word[59]=38828f91
Word[60]=23c00f87 Word[61]=8000b8b0
Word[62]=97a80b1e Word[63]=30908eb2
Word[64]=8fe86766 Word[65]=a0000
Word[66]=a02036bc Word[67]=308688f0
Word[68]=e4001405 Word[69]=408000
Word[70]=80912a7 Word[71]=8829f013
Word[72]=5a90fe00 Word[73]=b2b08
Word[74]=b082b002 Word[75]=a52883a3
Word[76]=a0c67728 Word[77]=a40c00
Word[78]=6ac3faf9 Word[79]=30ea1312

In the above word 0-15 are generated by padding the input message and breaking the message into sixteen 32-bit blocks. The words 16-79 are generated with the formulae:
Word[x] = (Word[x-3] xor Word[x-8] xor Word[x-14] xor Word[x-16])
Word[x] = Word[x] rotate left 1 time

Do 80 iteration to get the value of A,B,C,D,E. The value of A,B,C,D,E are as follows:

Iteration	**A**	**B**	**C**	**D**	**E**
0	f0b5e89d	b2d37ac7	abcd3f1d	3e13ae5	10d1e12f
1	25f2a8ab	f0b5e89d	ecb4deb1	abcd3f1d	3e13ae5
2	8b5a973	25f2a8ab	7c2d7a27	ecb4deb1	abcd3f1d
3	929654a	8b5a973	c97caa2a	7c2d7a27	ecb4deb1
4	e8a0fbb1	929654a	c22d6a5c	c97caa2a	7c2d7a27
5	ab4d5465	e8a0fbb1	824a5952	c22d6a5c	c97caa2a
6	fb709d4	ab4d5465	7a283eec	824a5952	c22d6a5c
7	3d9b3bec	fb709d4	6ad35519	7a283eec	824a5952
8	acf87aa	3d9b3bec	3edc275	6ad35519	7a283eec
9	7264f43b	acf87aa	f66cefb	3edc275	6ad35519
10	1d5b1d1f	7264f43b	82b3e1ea	f66cefb	3edc275
11	18f6cadb	1d5b1d1f	dc993d0e	82b3e1ea	f66cefb
12	277ca1e5	18f6cadb	c756c747	dc993d0e	82b3e1ea
13	912a8f6e	277ca1e5	c63db2b6	c756c747	dc993d0e
14	22ac8b1f	912a8f6e	49df2879	c63db2b6	c756c747
15	be89ddcc	22ac8b1f	a44aa3db	49df2879	c63db2b6
16	bfb88961	be89ddcc	c8ab22c7	a44aa3db	49df2879
17	243df120	bfb88961	2fa27773	c8ab22c7	a44aa3db
18	f62e657f	243df120	6fee2258	2fa27773	c8ab22c7
19	e16a72b1	f62e657f	90f7c48	6fee2258	2fa27773
20	5c99f4bf	e16a72b1	fd8b995f	90f7c48	6fee2258
21	87f53dca	5c99f4bf	785a9cac	fd8b995f	90f7c48
22	e15e1286	87f53dca	d7267d2f	785a9cac	fd8b995f
23	c0b1b245	e15e1286	a1fd4f72	d7267d2f	785a9cac
24	5db1f260	c0b1b245	b85784a1	a1fd4f72	d7267d2f
25	f8622e74	5db1f260	702c6c91	b85784a1	a1fd4f72
26	b2e72402	f8622e74	176c7c98	702c6c91	b85784a1
27	23383015	b2e72402	3e188b9d	176c7c98	702c6c91
28	27b02de3	23383015	acb9c900	3e188b9d	176c7c98
29	2de597c5	27b02de3	48ce0c05	acb9c900	3e188b9d
30	19375acc	2de597c5	c9ec0b78	48ce0c05	acb9c900
31	7b669f48	19375acc	4b7965f1	c9ec0b78	48ce0c05
32	196414fb	7b669f48	64dd6b3	4b7965f1	c9ec0b78
33	9b9ac686	196414fb	1ed9a7d2	64dd6b3	4b7965f1

34	47dc8898	9b9ac686	c659053e	1ed9a7d2	64dd6b3
35	d6db3c49	47dc8898	a6e6b1a1	c659053e	1ed9a7d2
36	3fa660c3	d6db3c49	11f72226	a6e6b1a1	c659053e
37	bc49b9c6	3fa660c3	75b6cf12	11f72226	a6e6b1a1
38	1de76413	bc49b9c6	cfe99830	75b6cf12	11f72226
39	43d48f8e	1de76413	af126e71	cfe99830	75b6cf12
40	514fea4e	43d48f8e	c779d904	af126e71	cfe99830
41	505377da	514fea4e	90f523e3	c779d904	af126e71
42	d6bb321b	505377da	9453fa93	90f523e3	c779d904
43	804f53b5	d6bb321b	9414ddf6	9453fa93	90f523e3
44	c3f35216	804f53b5	f5aecc86	9414ddf6	9453fa93
45	35e9187d	c3f35216	6013d4ed	f5aecc86	9414ddf6
46	690f8810	35e9187d	b0fcd485	6013d4ed	f5aecc86
47	eb3d8a4d	690f8810	4d7a461f	b0fcd485	6013d4ed
48	c67c2099	eb3d8a4d	1a43e204	4d7a461f	b0fcd485
49	631871c9	c67c2099	7acf6293	1a43e204	4d7a461f
50	ca73a16a	631871c9	719f0826	7acf6293	1a43e204
51	1a9c354b	ca73a16a	58c61c72	719f0826	7acf6293
52	b9d0d7ab	1a9c354b	b29ce85a	58c61c72	719f0826
53	55729ad3	b9d0d7ab	c6a70d52	b29ce85a	58c61c72
54	1b0c0504	55729ad3	ee7435ea	c6a70d52	b29ce85a
55	89afcd97	1b0c0504	d55ca6b4	ee7435ea	c6a70d52
56	5c80ac12	89afcd97	6c30141	d55ca6b4	ee7435ea
57	c3f90358	5c80ac12	e26bf365	6c30141	d55ca6b4
58	7de50112	c3f90358	97202b04	e26bf365	6c30141
59	4e6a9341	7de50112	30fe40d6	97202b04	e26bf365
60	781c97ab	4e6a9341	9f794044	30fe40d6	97202b04
61	c7042ecc	781c97ab	539aa4d0	9f794044	30fe40d6
62	288e5aa1	c7042ecc	de0725ea	539aa4d0	9f794044
63	f6d194e7	288e5aa1	31c10bb3	de0725ea	539aa4d0
64	4f60e002	f6d194e7	4a2396a8	31c10bb3	de0725ea
65	21c2f205	4f60e002	fdb46539	4a2396a8	31c10bb3
66	cd99587c	21c2f205	93d83800	fdb46539	4a2396a8
67	47e6a043	cd99587c	4870bc81	93d83800	fdb46539
68	bf1d2079	47e6a043	3366561f	4870bc81	93d83800
69	7f0fd3ea	bf1d2079	d1f9a810	3366561f	4870bc81
70	5a59ecc3	7f0fd3ea	6fc7481e	d1f9a810	3366561f
71	9261d457	5a59ecc3	9fc3f4fa	6fc7481e	d1f9a810
72	ed8542ff	9261d457	d6967b30	9fc3f4fa	6fc7481e
73	c611f096	ed8542ff	e4987515	d6967b30	9fc3f4fa
74	bc72c684	c611f096	fb6150bf	e4987515	d6967b30
75	ae63667c	bc72c684	b1847c25	fb6150bf	e4987515
76	12c667c6	ae63667c	2f1cb1a1	b1847c25	fb6150bf
77	5030c34f	12c667c6	2b98d99f	2f1cb1a1	b1847c25
78	305b2d6	5030c34f	84b199f1	2b98d99f	2f1cb1a1
79	8a39646a	305b2d6	d40c30d3	84b199f1	2b98d99f

Calculate the total hash with the formulae

$\Rightarrow$H0 = H0 + A

$\Rightarrow$ H1 = H1 + B

$\Rightarrow$H2 = H2 + C

$\Rightarrow$H3 = H3 + D

$\Rightarrow$H4 = H4 + E

Now in the present example

H_0 = b2d37ac7 + 8a39646a = 3d0cdf3

H_1 = af34fc76 + 305b2d6 = b23aaf4c
H_2 = 03e13ae5 + d40c30d3 = d7ed6bb8
H_3 = 10d1e12f + 84b199f1 = 95837b20
H_4 = f5b25ca1 + 2b98d99f = 214b3640

Finally
⇒ Message Digest is 3d0cdf31 b23aaf4c d7ed6bb8 95837b20 214b3640
⇒ The above mentioned value is the hash value of the messag "20"

Step 10: Concatenate the hash value along with the the value of X that is to be exchanged with the receiver and let it be X1.

This concatenation process can be done in many no of ways for example break the has code into 3 parts and place the X value at the starting of the third part. The concatenated value for this example is X1

$$\Rightarrow X1 = 203d0cdf31b23aaf4cd7ed6bb895837b20214b3640$$

Step 11: Exchange the value X1 of with Y i.e send X1 to receiver and receive Y from receiver.
Step 12: Slice X1 in to two parts and store the hash value that is received in another variable let it be h1 and value of X in X2.

⇒ X2 = 20
⇒3d0cdf31 b23aaf4c d7ed6bb8 95837b20 214b3640

Step 13: Calculate the hash value of X2 i.e pass the value of X2 to SHA-1 hash function and store the hash value in another variable let it be h2.

⇒X2=20
⇒h2 = 3d0cdf31 b23aaf4c d7ed6bb8 95837b20 214b3640

Step 14: Compare the hash value h1 and h2, if both are equal then continue to follow bellow steps if the hash value is not equal abort the process and start from begning.

⇒ h1 = 3d0cdf31 b23aaf4c d7ed6bb8 95837b20 214b3640 = h2

Step 15: Calulate the value of K_a with the formule $K_a = Y^{n1} \bmod P$.

$\Rightarrow K_a = Y^{n1} \bmod P = 14^5 \bmod 23$
$\Rightarrow K_a = 15$

Step 16: Calulate the value of K_b with the formule $K_b = X1^{n2} \bmod P$.

$\Rightarrow K_b = X1^{n2} \bmod P = 20^{461}$ mode 23
$\Rightarrow K_b = 15$

Therefore the share value is $K = K_a = K_b = 15$.

IV. IMPLEMENTATION RESULTS AND ITS DISCUSSION

The synthesis and simulation of the proposed RSA based Diffie-Hellman using Key using Digital Signature algorithm is done using Verilog, Xilinx ISE Design Suite Version 14.7. Xilinx is a Integrated Synthesis Environment, it is used for the analysis like performance, execution and implementation of Hardware descriptive language (HDL) structures. The user interface of Xilinx is user friendly and can be easily understood for better understanding of ISE. It allows the user to set up modules, synthesize and combine designs, implement the design on hardware, perform timing evaluation, study RTL schematics, simulate the code with test benches and map the pins on the required target device. It gives a structure to the design only for FPGA products from Xilinx, not any other brand platforms. For circuit synthesis and design Xilinx ISE is used while for testing ISIM or Model Sim test system is used. There are two stages in the implementation of the proposed algorithm. The first stage is the software implementation of the algorithm and the second stage is a hardware implementation of the algorithm in Field-Programmable Gate Array. The proposed algorithm has been implemented in different FPGAs. The FPGA boards have been selected for the optimum performance for the proposed algorithm. The FPGA boards that are selected are

Spartan 3, Spartan 6, and Vertex 4. The no of resources that are used by multiple FPGA boards is also compared and analyzed. The results obtained have been verified and are in accordance with the desired output.

Xilinx ISE Design Suit 14.7 running on Windows 10	
Developer(s)	Xilinx
Final release	14.7 / October 23, 2013; 6 years ago
Operating system	RHEL, SLED, FreeBSD, Microsoft Windows
Size	6.1 Gigabytes
Available in	English
Type	EDA
License	Shareware

4.1 RTL Schematic of Top Module

The below figure shows the hardware outline of the overall proposed algorithm, p and q represents the prime numbers that are to be chosen to generate key pairs. K1 and k2 represents the keys that are to be exchanged in between the exchange process k1 is considered as a message and digital signature is generated.

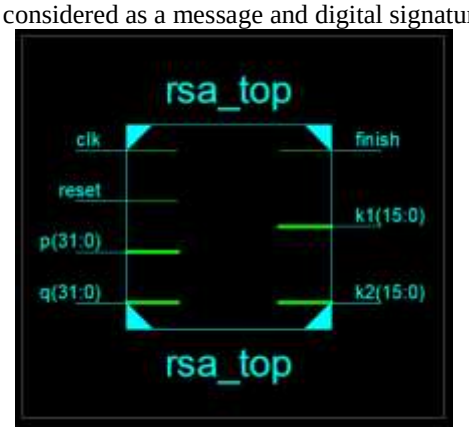

Figure 3 Schematic model

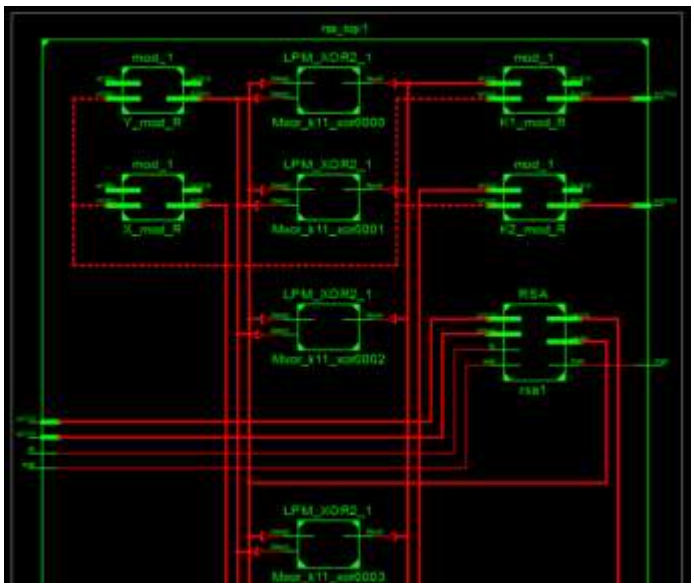

Figure 4 Schematic model zoomed in

This above figure 3 and 4 is the zoomed-in version of the schematic model of the proposed algorithm. It clearly shows the interconnection between the module that are synthesized for the hardware implementation of the proposed algorithm. Mod represents the function to find the reminder for the keypair generation process. The modules that are present in the above figure are the submodule of the rsa_top module i.e. these are present in the top module in Figure 5

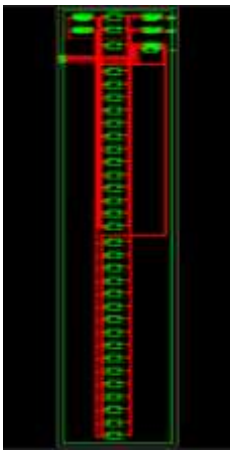

Figure 5 Schematic model zoomed out

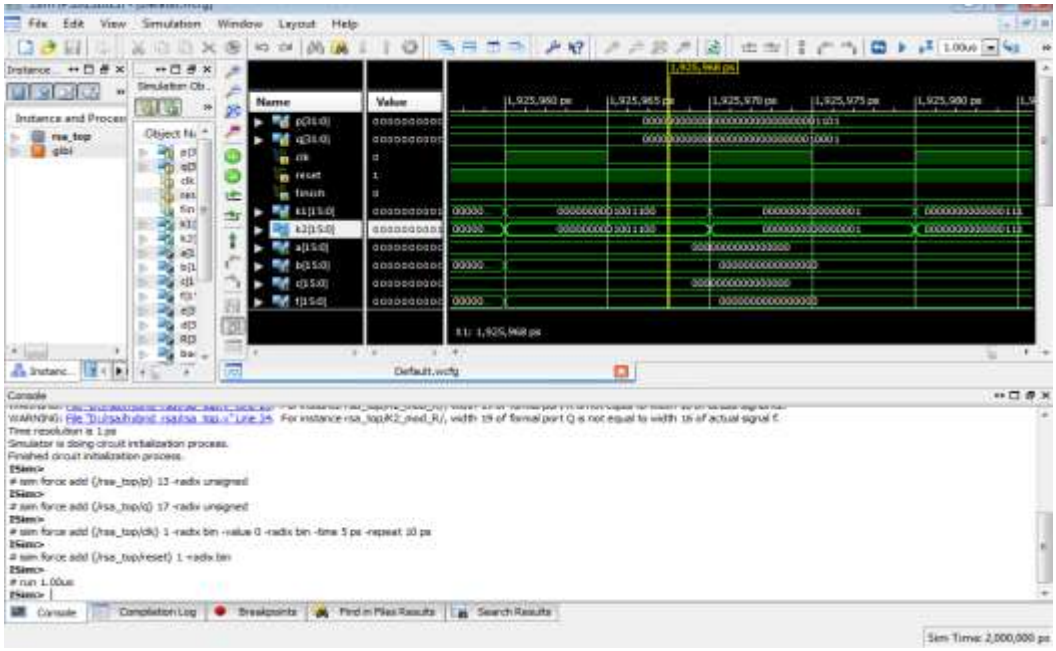

Figure 6 Output of the proposed algorithm

In the above figure 6 p and q represents the prime numbers chosen to generate key pair. The variable a, b, c, f are intermediates in the process of key generation. K1 and K2 are the keys that are to be exchanged as in the Diffie Hellman algorithm.

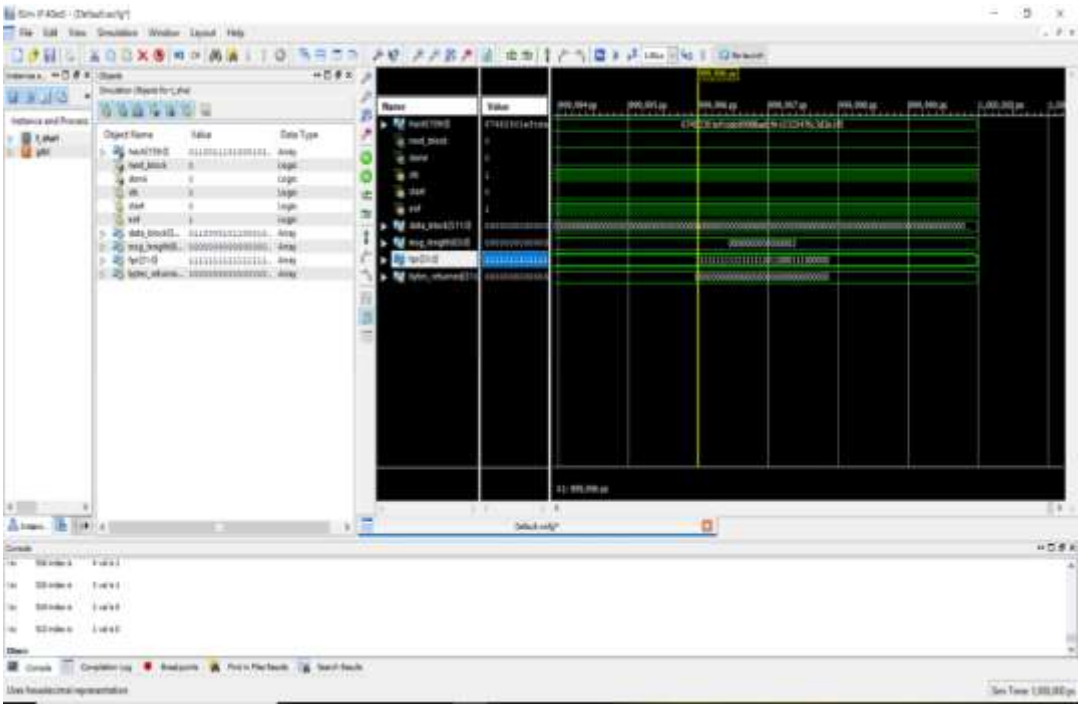

Figure 6.5 Output of SHA 1 hash function

The above figure shows that the hash value that is generated after passing the message that is key k1 to SHA 1 hash function.

4.2 Comparision of Different FPGA's

Table 6.9 Comparison of FPGA's

FPGA Type	Number of slice registers	Number of slice LUTs	Number of fully used LUT-FF pairs	Number of Bonded IOBs	Number of BUFG/BUFGCTRLs	Number of DSP48E1s
Virtex-4	5750	2528	10542	323	3	11
Virtex-5	341	13565	253	99	1	24
Spartan-6	2589	10177	2252	323	3	11

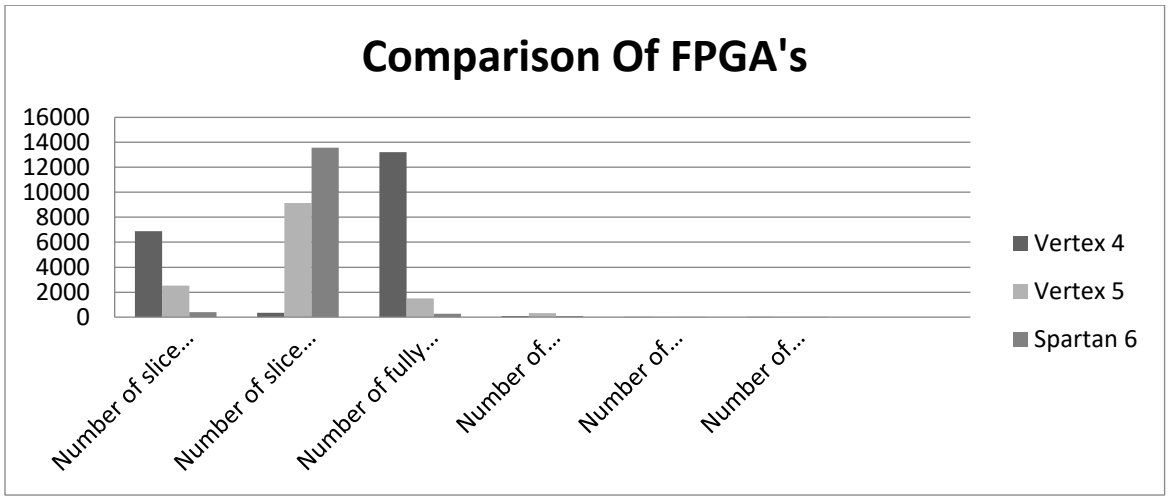

Figure 7 Comparison of resource consumed by different FPGA's

V. CONCLUSION AND FUTURE WORK

This project is the combination of three algorithms for key generation, digital signature, and authentication, and generated hash value has been presented. The proposed algorithm is simulated in Xilinx 14.7 ISE Design Suite. The technical specifications are vertex 5 family, the package we used is FT256, speed grade selected is -4, The type of Top-level source HDL, Synthesis tool XST (VHDL/Verilog). Various parameters like Slice logic utilization, the number of pins used, etc. have been analyzed, for 4 FPGA boards namely Virtex-4, Spartan 3, and Spartan-6 boards. The key pair that is to be shared is generated by the RSA. This key pair is generated by choosing large prime numbers. One of the generated keys is passed to a hash function that is SHA -1. The generated hash value is concatenated with the key and sent to the receiver. At the receiver end the hash value that is concatenated with the message is compared with a hash value that is generated by passing the message that is received to the hash function. If both the hash values are the same then a secure session is established, if the hash value is not matched then the keys are generated again and shared. The time complexity of the algorithm can be further revised and optimized for the better performance and efficiency of the proposed algorithm. The key size for digital signature can be revised for low memory consumption. Further it can be made more secure with better digital signature algorithms. A grater no of encryption layers can be used for more security. Instead of using RSA for generating key pair, Elliptic curve cryptography can be used for better key length and strength of it. Instead of SHA – 1 more better hashing algorithms can be used for higher security. The usage of memory in FPGA can be optimized further.

REFERENCES

[1] Lee, S., Cho, S. M., Kim, H.,&Hong,S. "A Practical Collision-Based Power Analysis on RSA Prime Generation and Its Countermeasure". IEEE Access,vol-7,2019

[2] Zhang, J., & Qu, G. (2019). "Recent Attacks and Defenses on FPGA-based Systems". ACM Transactions on Reconfigurable Technology and Systems, Volume: 12, Issue: 3, 2019

[3] Paul, R., & Shukla, S. (2018). "Partitioned security processor architecture on FPGA platform". IET Computers & Digital Techniques, Volume: 12 , Issue: 5 , 9 2018

[4] M.Khalil , M.Nazrin , Y.W. Hau, Implementation of SHA-2 Hash Function for a Digital Signature System-on-Chip in FPGA, International Conference on Electronic Design, 2018

[5] Amit Thobbi, ShriniwasDhage, Pritesh Jadhav and Akshay Chandrachood, "Implementation of RSA Encryption Algorithm on FPGA" American Journal of Engineering Research (AJER), Volume-4, Issue-6, 2015

[6] Kim, J., Schmieder, M., Peter, M., Chung, H., Choi, S.-W., Kim, I., & Han, Y. (2018). "A Comprehensive Study on mmWave-based Mobile Hotspot Network System for High-Speed Train Communications". IEEE Transactions on Vehicular Technology, Volume: 68 , Issue: 3 , 2019

[7] Na Qi Jing Pan Qun Ding, The implementation of FPGA-based RSA public-key algorithm and its application in mobile-phone SMS encryption systemInternational Conference on Instrumentation, Measurement, Computer, Communication and Control , 2011.

[8] Wankai Tang, Xiang Li, Jun Yan Dai, Shi Jin, Yong Zeng, Qiang Cheng, Tie Jun Cui, "Wireless communications with programmable metasurface : Transceiver design and experimental results", IEEE China Communications, Volume: 16 , Issue: 5, 2019

[9] Hridoy Jyoti Mahanta, Ajoy Kumar Khan, "Securing RSA against power analysis attacks through non-uniform exponent partitioning with randomisation". IET Information Security, Volume: 12 , Issue: 1 , 2018

[10] Turan, F., & Verbauwhede, I. (2019). "Compact and Flexible FPGA Implementation of Ed25519 and X25519". ACM Transactions on Embedded Computing Systems, Volume 18, Issue 3, 2019

[11] K.,Rebeiro, C., & Hazra, A. (2018). "An Algorithmic Approach to Formally Verify an ECC Library". ACM Transactions on Design Automation of Electronic Systems, Volume 23 Issue 5, October 2018

[12] Khan, Z. U. A., & Benaissa, M. (2017). "High-Speed and Low-Latency ECC Processor Implementation Over GF($2^{m})$ on FPGA". IEEE Transactions on Very Large Scale Integration (VLSI) Systems, Volume: 25 , Issue: 1 , 2017

[13] Liao, K., Cui, X., Liao, N., Wang, T., Yu, D., & Cui, X. (2017). "High-Performance Noninvasive Side-Channel Attack Resistant ECC Coprocessor for GF(2m)". IEEE Transactions on Industrial Electronics , Volume: 64 , Issue: 1 , 2017

[14] Chien-Chung Ho ; Yu-Ping Liu ; Yuan-Hao Chang ; Tei-Wei Kuo, "Antiwear Leveling Design for SSDs With Hybrid ECC Capability", IEEE Transactions on Very Large Scale Integration (VLSI) Systems, Volume: 25 , Issue: 2 , 2017

[15] Armando Faz-Hernández , Julio López, Ricardo Dahab, "High-performance Implementation of Elliptic Curve Cryptography Using Vector Instructions". ACM Transactions on Mathematical Software (TOMS) TOMS Homepage table of contents archive, Volume 45 Issue 3, August 2019

[16] Keller, M., Byrne, A., and Marnane, W. P. "Elliptic curve cryptography on FPGA for Low-Power applications", ACM Transactions on Reconfigurable Technology and Systems, Volume 2 Issue 1, March 2009

[17] Zhe Liu, Jian Weng, Zhi Hu, and Hwajeong Seo. "Efficient elliptic curve cryptography for embedded devices". ACM Transactions on Embedded Computing Systems, Volume 16 Issue 2, April 2017

[18] Pascal Sasdrich and Tim Guneysu. "Implementing Curve25519 for side-channel–protected Elliptic Curve Cryptography", ACM Transactions on Reconfigurable Technology and Systems, Volume 9 Issue 1, November 2015
[19] Pan, W., Zheng, F., Zhao, Y., Zhu, W.-T., & Jing, J. "An Efficient Elliptic Curve Cryptography Signature Server With GPU Acceleration". IEEE Transactions on Information Forensics and Security, Volume: 12 , Issue: 1 , Jan. 2017
[20] Chang, S.-Y., Lin, Y.-H., Sun, H.-M., and Wu, M.-E. 2012. "Practical RSA signature scheme based on periodical rekeying for wireless sensor networks". ACM Transactions on Sensor Networks, Volume 8 Issue 2, March 2012
[21] HeeSeok Kim, Dong-Guk Han, Seokhie Hong, and JaeCheol Ha, "Message blinding method requiring no multiplicative inversion for RSA". ACM Transactions on Embedded Computing Systems, Volume 13 Issue 4, November 2014
[22] Wang, D. M., Ding, Y. Y., Zhang, J., Hu, J. G., & Tan, H. Z. "Area-efficient and ultra-low-power architecture of RSA processor for RFID", IEEE Electronics Letters Volume: 48 , Issue: 19 , September 2012
[23] Fueyo, M., & Herranz, J. "On the Efficiency of Revocation in RSA-Based Anonymous Systems". IEEE Transactions on Information Forensics and Security, Volume: 11 , Issue: 8 , Aug. 2016
[24] Ma, K., Liang, H., & Wu, K. "Homomorphic Property-Based Concurrent Error Detection of RSA: A Countermeasure to Fault Attack". IEEE Transactions on Computers, Volume: 61 , Issue: 7 , July 2012
[25] Huang, X., & Wang, W. "A Novel and Efficient Design for RSA Cryptosystem With a Very Large Key Size". IEEE Transactions on Circuits and Systems II: Express Briefs, Volume: 62 , Issue: 10 , Oct. 2015
[26] Sutter, G. D., Deschamps, J.-P., & Imana, J. L. "Modular Multiplication and Exponentiation Architectures for Fast RSA Cryptosystem Based on Digit Serial Computation". IEEE Transactions on Industrial Electronics, Volume: 58 , Issue: 7 , July 2011
[27] Hoque, T., Yang, K., Karam, R., Tajik, S., Forte, D., Tehranipoor, M., & Bhunia, S. "Hidden in Plaintext: An Obfuscation-based Countermeasure against FPGA Bitstream Tampering Attacks". ACM Transactions on Design Automation of Electronic Systems, Volume 25 Issue 1, December 2019
[28] Lam, S.-K., Srikanthan, T., & Clarke, C. T. "Rapid evaluation of custom instruction selection approaches with FPGA estimation". ACM Transactions on Embedded Computing Systems, Volume 13 Issue 4, November 2014
[29] Chen, D., Putnam, A., & Wilton, S. (Eds.). "Introduction to the Special Section on Deep Learning in FPGAs". ACM Transactions on Reconfigurable Technology and Systems, Volume 11 Issue 3, December 2018
[30] Farooq, U., Baig, I., & Alzahrani, B. "An Efficient Inter-FPGA Routing Exploration Environment for Multi-FPGA Systems". IEEE Access, Volume: 6 2018
[31] Mitra, J., & Nayak, T. K. "An FPGA-Based Phase Measurement System". IEEE Transactions on Very Large Scale Integration (VLSI) Systems, Volume: 26 , Issue: 1 , Jan. 2018
[32] Jeppesen, B. P., Rajamani, M., & Smith, K. M. "Enhancing functional safety in FPGA-based motor drives". The Journal of Engineering, Volume: 2019 , Issue: 17 , 2019
[33] Lonla Moffo, B., & Mbihi, J. (2015). "A Novel Digital Duty-Cycle Modulation Scheme for FPGA-Based Digital-to-Analog Conversion". IEEE Transactions on Circuits and Systems II: Express Briefs, Volume: 62 , Issue: 6 , June 2015
[34] Saleh Saraireh, "A Secure Data Communication System Using Cryptography and Stenography", International Journal of Computer Networks & Communications (IJCNC) Vol.5, No.3, May 2013
[35] Alimohammad, A., & Fard, S. F. (2014). "FPGA-Based Bit Error Rate Performance Measurement of Wireless Systems". IEEE Transactions on Very Large Scale Integration (VLSI) Systems, Volume: 22 , Issue: 7 , July 2014
[36] Rashwan, A. M., Taha, A.-E. M., & Hassanein, H. S. (2014). "Characterizing the Performance of Security Functions in Mobile Computing Systems". IEEE Internet of Things Journal, Volume: 1 , Issue: 5 , Oct. 2014