ISSN: 2350-0328

International Journal of AdvancedResearch in Science, Engineering and Technology

Vol. 5, Issue 8 , August 2018

Developed Lightweight Authentication and Key Management Protocol for Wireless Sensor Network

Shaymaa Mahmood Naser, Muayad Sadik Croock

MSc. Student, Department of Computer Science, Information Institute for Postgraduate Studies, Baghdad, Iraq
Assistant Professor, Department of Computer Engineering, University of Technology, Baghdad, Iraq

ABSTRACT: Nowadays, Wireless Sensor Network (WSN) is being an important part of our daily life in most of application, for example: health monitoring, environmental issues, agricultural and so on. In this paper, a modified lightweight authentication and key management protocol for wireless sensor network is proposed. The modification is done by generates new keys, used for authentication, utilizing a developed technique of Linear Feedback Shift Register (LFSR)-(Geffe generator) in combination with Chaotic map. The developed technique employs of SIGABA algorithm in generating the seed functions for LFSR-Geffe generator to enhance the key uncorrelation by increasing randomization. In addition, a designed simulator that simulates the three phases of protocol's form base station to sensor nodes is presented. The simulation treats another obstacle of lightweight protocol absence in the well-known simulators, such as NS2 and NS3. The results show the superior performance of the developed protocol in terms of key generation.

KEY WORDS: WSN, lightweight protocol, authentication, key management, SIGABA.

I.INTRODUCTION

The Wireless Sensor Network (WSN) consists of tiny chunks, called nodes, which are distributed in large size areas for sensing environment around each node. The sensor's readings can be processed either in base station or in the same node locally. Therefore, it must concern the security side to protecting the network from different attacks [1]. Security is a critical issue in the ad hoc sensor network. The main two issues in ad hoc network are the authentication and key management [2]. The node authentication and random key generation are considered in this research.

Normally, the traditional authentications are not sufficient for WSN. Therefore, for introducing new techniques that support the security in WSN numerous research works have been introduced. The authors of [3] compared different authentication and key management protocols that concentrated on security in WSN. This comparison showed the advantages and disadvantages of each protocols and related applications. In [4], the problem of malicious nodes in WSN network was solved using light-weight authentication and key management protocol by symmetric cryptographic primitives with (HMAC). For more security, the need in cryptographic techniques for generating key sequences with high randomness and statistical properties has been increased [5]. It is well known that the Linear Feedback Shift Registers (LFSRs) and chaotic map depend on the properties of random number generation schemes [6], [7], [8]. At the other hand, the unique design of SIGABA that provide the underlying systems with the ability of fighting all crypt-analytical attacks in the course of its usage [9].

In this paper, we introduce a development to the adopted lightweight authentication and key management protocol by providing an enhanced method of combination of LFSR based Geffe generator with Chaotic map approach under the SIGABA technique. Moreover, a specific simulator has been proposed to adapt with the requirements of lightweight protocol that are hard to address in NS2 and NS3 simulator. This simulator includes through three main phases for authentication of nodes, cluster heads and base-stations.

II. THE PROPOSED SYSTEM

As previously mentioned, the proposed simulator uses the lightweight protocol principles. This simulator performs the contributions of this paper, key management as well as the other phases of the protocols. This section is divided into four sub-sections for illustrating the presented system.

II.I SYSTEMBLOCK DIAGRAM

The block diagram, shown in Figure (1), introduces the proposed system. This diagram is divides into the following:

A. **Base Station Layer:** this layer includes the base stations that is connected with cluster heads. Base stations are treated as stable point for collecting and controlling the information that gathering form cluster head and generate keys for authenticate the clusters. Additionally, it performs the functions of monitoring, managing, and recording the actions for wireless sensor network.

B. **Cluster Head Layer:** depending on the number of clusters, this layer includes the management of the cluster headers. Each cluster head works as a gateway between the cluster sensors and related base station. It is selected form group of sensor nodes, included in the cluster, depending on some factors, such as distance and energy. It is also responsible for generating keys to authenticate the sensor nodes.

C. **Sensor NodeLayer:** it contains numbers of sensors, each sensor is structured with microcontroller, memory unit and battery. Each of one has the ability of generating different keys for other sensors in the same cluster when exchange the roles to work as cluster head.

D. **Node Authentications**: Each sensor node must authenticate itself to network through cluster head in the beginning of network initialization or even through other sensor nodes.

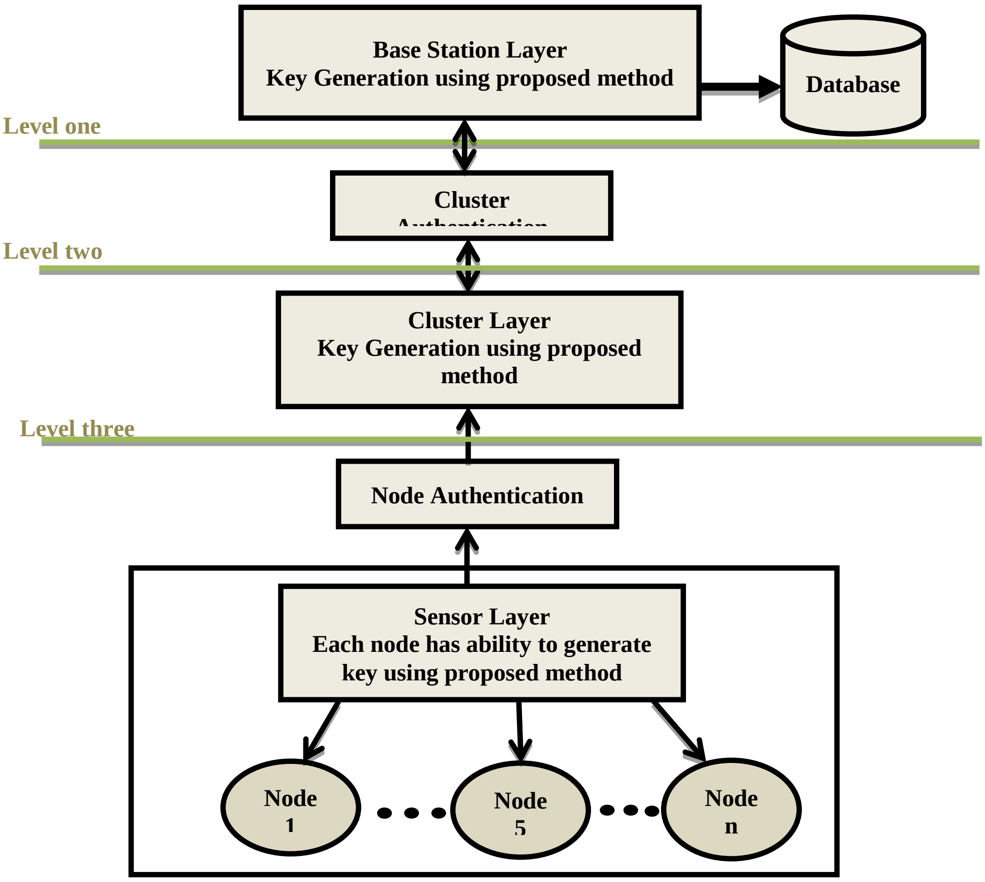


Fig. 1. Block Diagram of Presented System Using Proposed Key Generation

II.II THE PROPOSED ALGORITHM

The developed algorithm of the presented system has been constructed following the roles of lightweight authentication and key management protocol (AKMS). It is divided into different sub-sections to ease the reading flow.

A. System Algorithm

Figure (2) shows the proposed algorithm as flowchart. Throughout this flowchart, different actions are performed as follows:

Network Initialization stage:

In this stage, the neighbors of nodes are perceived others by broadcasting random number for each node and generate encryption key via Hashing master key and random number. Moreover, the selection of cluster head amongst the nodes within each cluster. Each cluster head is connected with a base station.

Key Generation:

In this stage the master key is generated by the proposed combination of LFSR (Geffe generator) with Chaotic map under the SIGABA technique with 128 bits length. This key is distributed to all cluster heads. Then, each cluster head generates the keys for sensor nodes in the same manner but without repeating keys and high randomness.

Authentication Protocol stage:

In this stage, the authentication action is performed between the base station and cluster heads, while each cluster head is authenticated with related cluster nodes. This stage is also activated when new node requires entering in the network. This protocol performs the authentication of the new node to make sure that it is legally added [4].

B. Proposed Key Generation Algorithms

LFSR is selected due to simplest performing way with acceptable randomness [7], [5].In addition, the Geffe generator method is adopted to increase the randomness of LFSR to be combined Chaotic Maps. As mentioned above, the SIGABA method is used in selecting the pronominal formulas of LFSR that are used in Geffe generator. The use of SIGABA increases the randomness of key generaton as proved later in the testing stage. Encryption-based chaos provides better protection than traditional ways. The most famous properties of chaos are "butterfly-effect" (the initial conditions sensitivity) that generated by deterministic equations from pseudo randomness. In fact, a chaotic system can make an infinite number of unsteady repeat paths. In our proposed algorithm, one-dimensional discrete chaotic maps is adopted. It is very sensitive to system variable and control parameter as explained in the following equation [7]:

$$x_{i+1} = \square x_i(1 - x_i)\text{------------- }\mathbf{(1)}$$

where $\boldsymbol{x_i}$ ($0<x_i<1$) is the system variable and α is the control parameter ($3.56\leq \boldsymbol{\alpha} \leq 4$) , respectively, and $\boldsymbol{i}$ is the number of iterations. Here, we refer to $\boldsymbol{x_0}$and α as the initial state of the logistic map[8]. The two techniques combined (LFSR and Chaotic Map) are commonly used in wireless sensor network [6]. Full details of the proposed key generation is illustrated in Figure (4) as a flowchart.

It is important to note that the algorithms runs in parallel the LFSR and Chaotic methods. The next step is done by testing the generated value of $X_{i.}$ If it is greater than or equal to the prim value, this value is buffered. Otherwise, the algorithm is stopped to generate numbers and selecting random values from the buffering stored. Now, the fraction of the selected value is removed and keeping the integer value. As a result, the obtained value is XOR with result of the combination of LFSR-Geffe generator under SIGABA technique.

In Geffe generator, we need three LFSR polynomial formulas selected from numerous ones using the SIGABA technique to increase the randomization. When the selected polynomials are received from SIGABA algorithm, they have been checked if anyone of them was used before. In this case, the proposed algorithm requests new polynomial from the SIGABA. The received formulas are considered by the Geffe generator mechanism as shown in Figure (5). Where, LFSR1, LFSR2, and LFRS3 are the received polynomials.

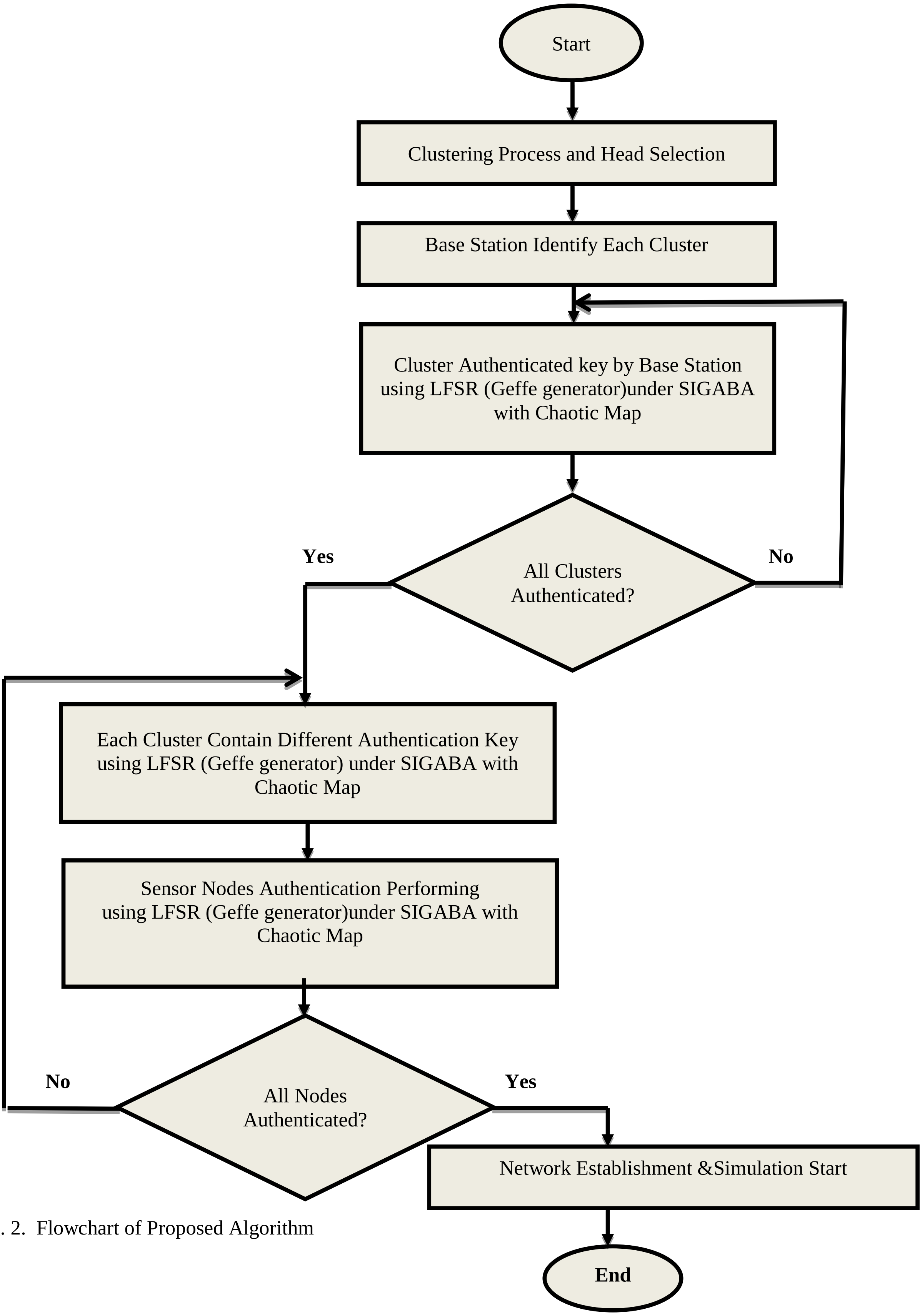

Start
Clustering Process and Head Selection
Base Station Identify Each Cluster
Cluster Authenticated key by Base Station using LFSR (Geffe generator)under SIGABA with Chaotic Map
All Clusters Authenticated?
Yes
No
Each Cluster Contain Different Authentication Key using LFSR (Geffe generator) under SIGABA with Chaotic Map
Sensor Nodes Authentication Performing using LFSR (Geffe generator)under SIGABA with Chaotic Map
All Nodes Authenticated?
No
Yes
Network Establishment &Simulation Start
End

Fig. 2. Flowchart of Proposed Algorithm

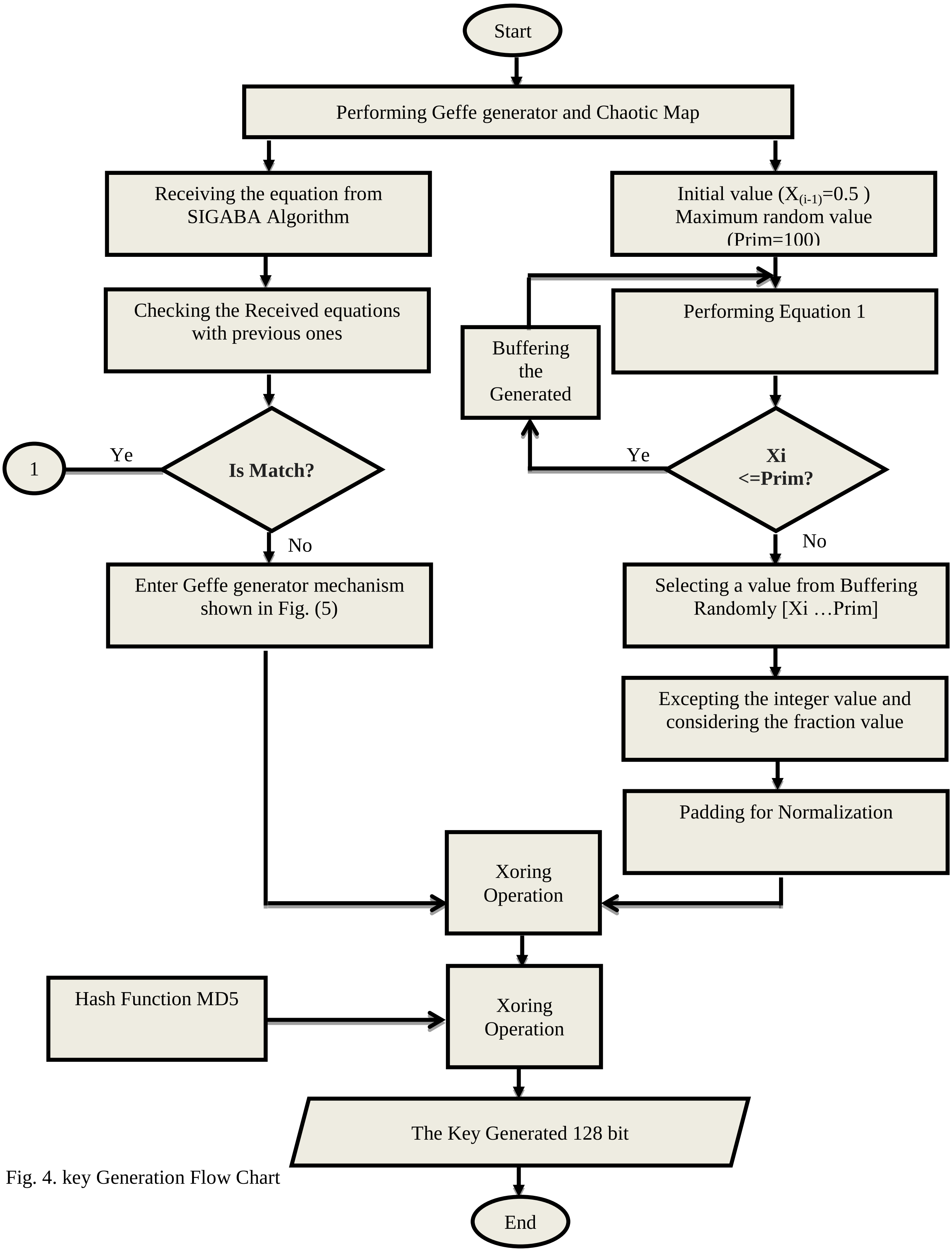

Start
Performing Geffe generator and Chaotic Map
Receiving the equation from SIGABA Algorithm
Initial value ($X_{(i-1)}$=0.5) Maximum random value (Prim=100)
Checking the Received equations with previous ones
Buffering the Generated
Performing Equation 1
1
Ye
Is Match?
Ye
Xi <=Prim?
No
No
Enter Geffe generator mechanism shown in Fig. (5)
Selecting a value from Buffering Randomly [Xi …Prim]
Excepting the integer value and considering the fraction value
Padding for Normalization
Xoring Operation
Hash Function MD5
Xoring Operation
The Key Generated 128 bit
End

Fig. 4. key Generation Flow Chart

ISSN: 2350-0328

International Journal of AdvancedResearch in Science, Engineering and Technology

Vol. 5, Issue 8 , August 2018

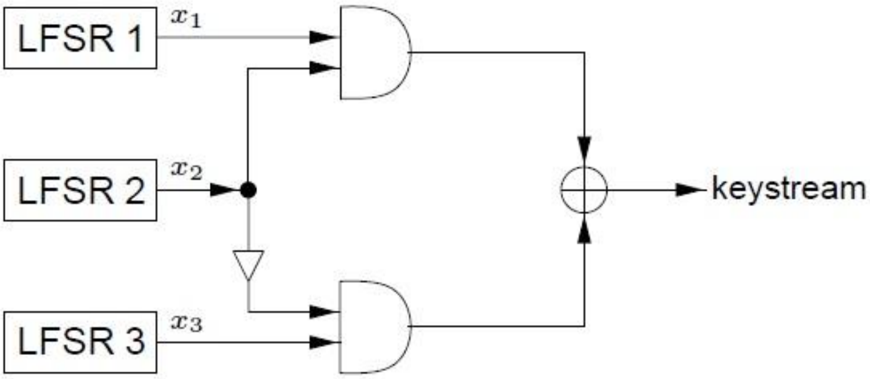


Fig. 5. Geffe Generator Mechanism [7]

C. SIGABA Algorithm

SIGABA method was adopted in the second world war to tackle the attacking problems faced the traditional approaches. It considers the alphabetic characters in code generation. In the proposed algorithm, the rules of SIGABA are adopted and binary system is employed instead of alphabetic characters. The idea of SIGABA can be defined as reliability of choosing as seed equation among services of them until obtaining source of bit stream [10]. The SIGABA flowchart is shown in Figure (3).

In this figure, the algorithm randomly selects five seed equations (polynomials) as initial values to be used later in LFSR. The selected equations have been chosen amongst array of (N=10) considered formulas. The fifth equation is responsible for choosing the selector equation between 3 and 4. The selector equation chooses either equation 1 or 2 to be the initial key. The initial key is Xoring with the left equation of 3 or 4 to produce the SIGABA key.

ISSN: 2350-0328

International Journal of AdvancedResearch in Science, Engineering and Technology

Vol. 5, Issue 8 , August 2018

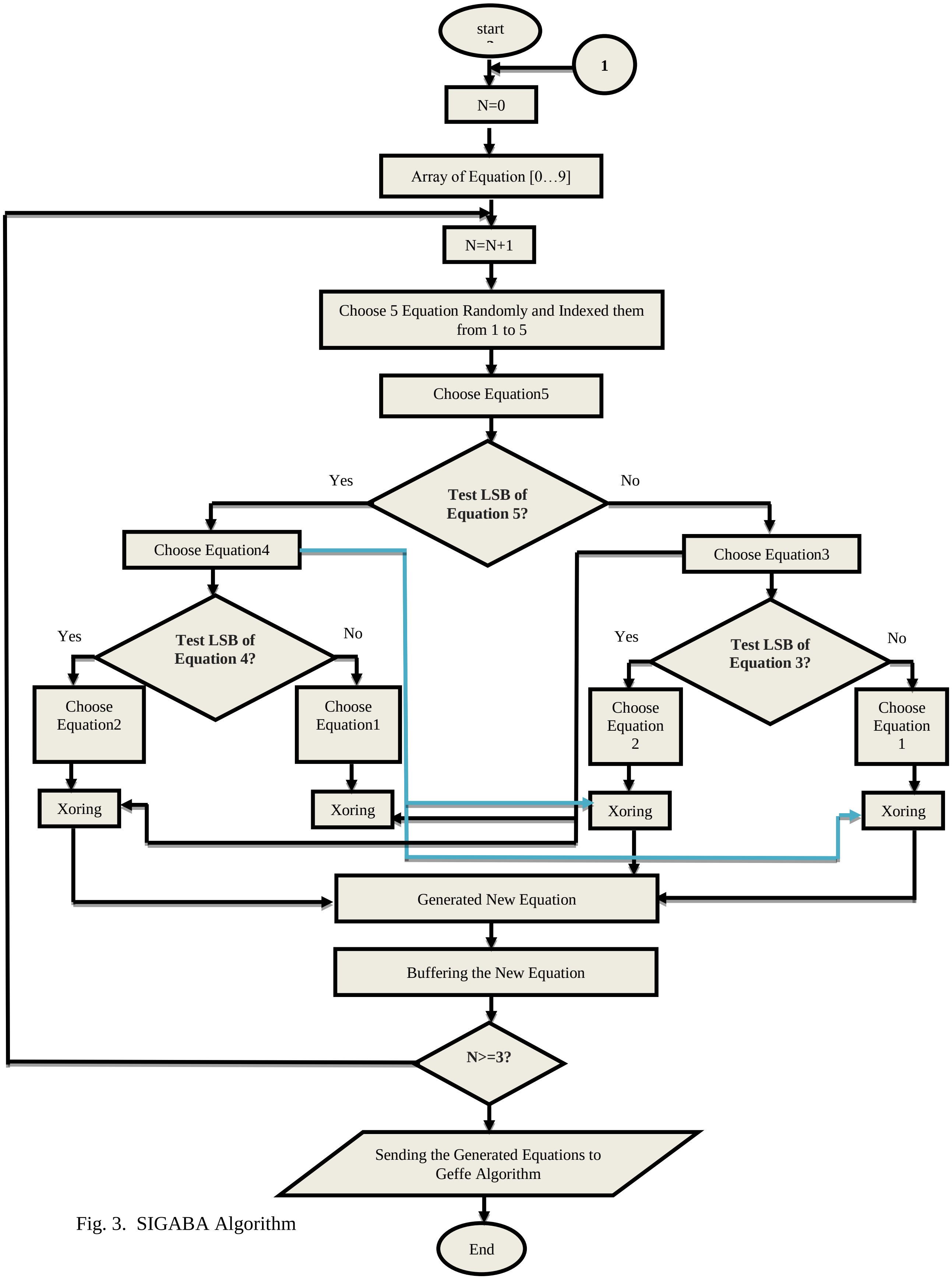


Fig. 3. SIGABA Algorithm

III. BULIT DATABASE

The sensor readings need to be stored in a built database. The database is built using SQL server to offer more flexibility and efficiency. In Figure (7) shows an ER-diagram of the adopted database in simulation.

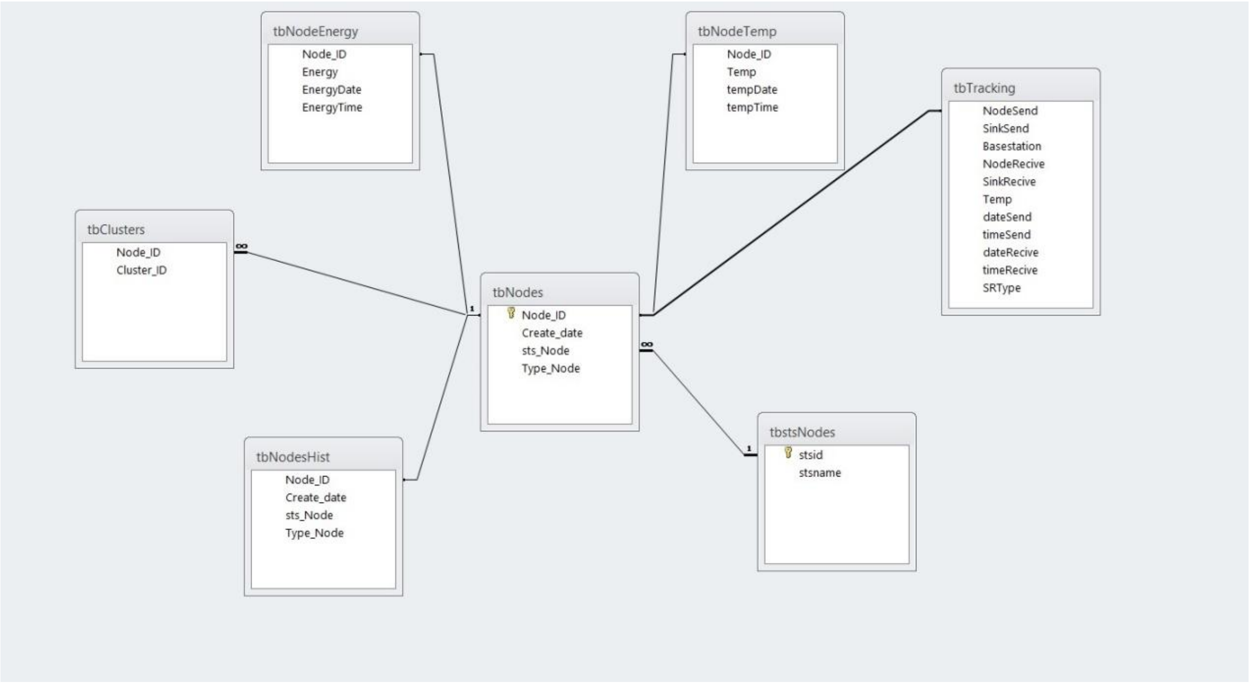


Fig. 7. ER-Diagram of The Built Database

The life-cycle of building this database explained in block diagram, shown in Figure (8). The flowed procedure is demonstrated as:

A. Requirement

In this stage, the gathering of data is done, such as energy, authentication information, data messages transmission inside the network, and all information around the external environment of nodes.

B. Design

Different tables are constructed as follows:

1. *TbNodes table*: this table contains the last statues of node and it is considered as the main table in the database.
2. *TbNodesHist table*: it records of history of all nodes in the network and it is related to tb_nodes table in fields of sts_node and node_id.
3. *TbNodeTemp table:* it contains the readings of nodes and their timing.
4. *TbNodeEnergy table*: it includes all information about energy of nodes and changes of these energies.
5. *TbTracking table:* it is the largest table size in structure of database that includs the traffic of the messages between nodes, cluster heads and base station.
6. *TbClusters table*: it describes each node and the connected cluster head.
7. *TbstsNodes table*: it is symbolic table for the statues (active, sleep, died) for nodes.

C. Implementation:

SQL server 2012 are used to implement the designed ER-diagram explained above.

D. **Testing:**

The database is tested by performing the insert, update, and delete actions.

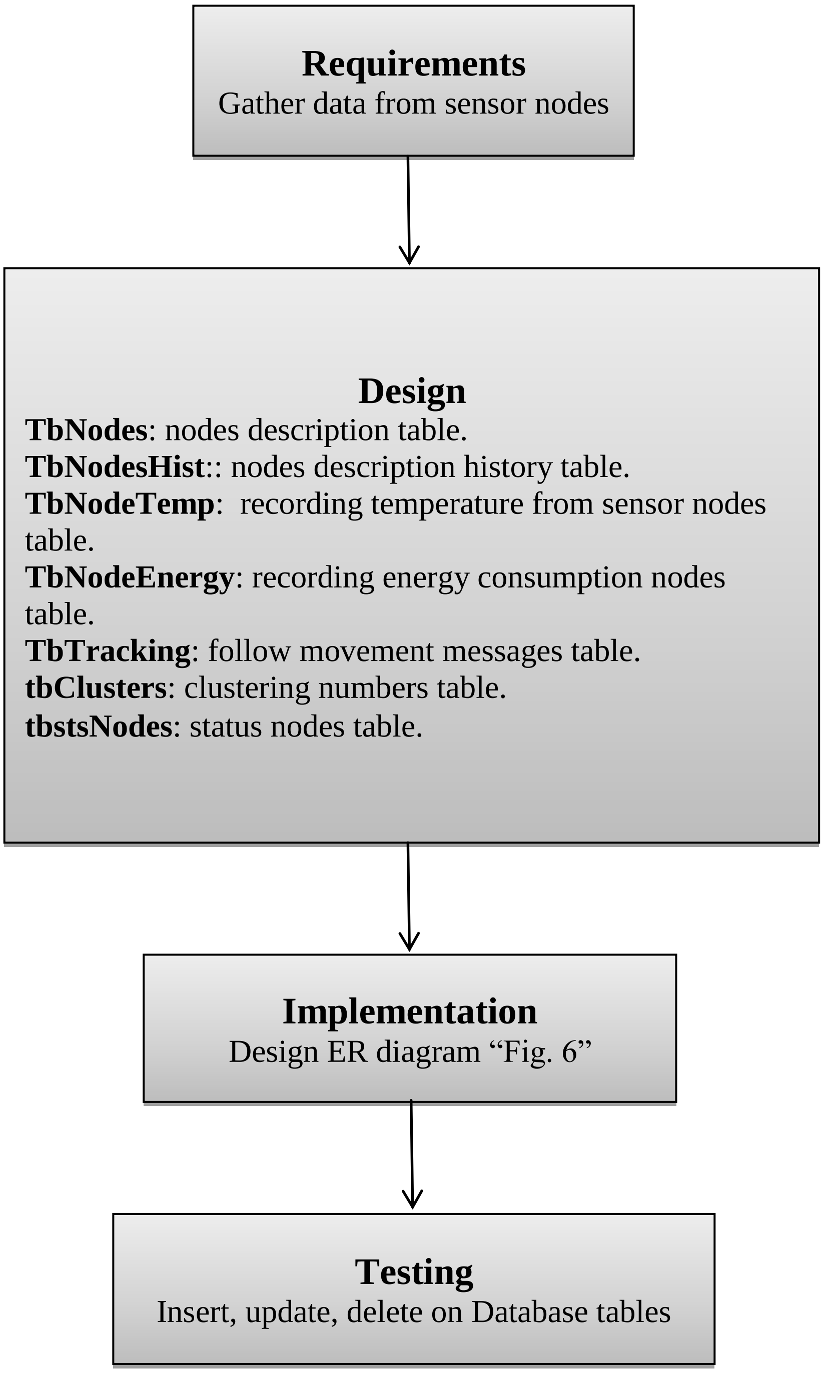

Requirements
Gather data from sensor nodes
Design
TbNodes: nodes description table.
TbNodesHist:: nodes description history table.
TbNodeTemp: recording temperature from sensor nodes table.
TbNodeEnergy: recording energy consumption nodes table.
TbTracking: follow movement messages table.
tbClusters: clustering numbers table.
tbstsNodes: status nodes table.
Implementation
Design ER diagram “Fig. 6”
Testing
Insert, update, delete on Database tables

Fig. 8. Built database block diagram

IV. GUI DESIGN

The simulator that implement the behaviors of authentication and key management of adopted lightweight protocol in addition to surviving plan of WSN. The simulator offers monitoring screens for administrators. The Graphical User Interfaces (GUI) of aforesaid simulator can be explained as follows.

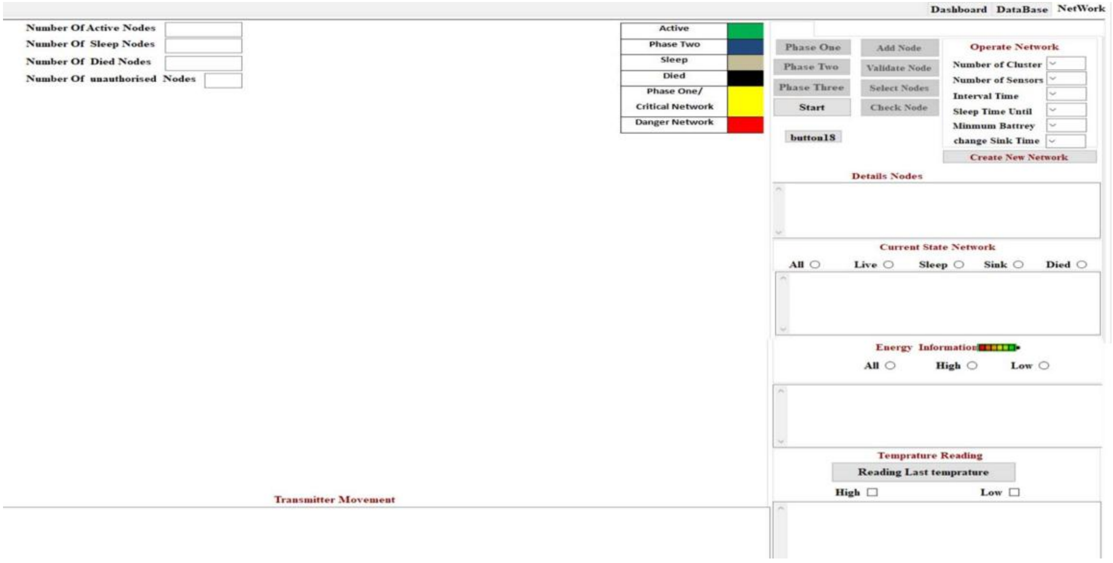
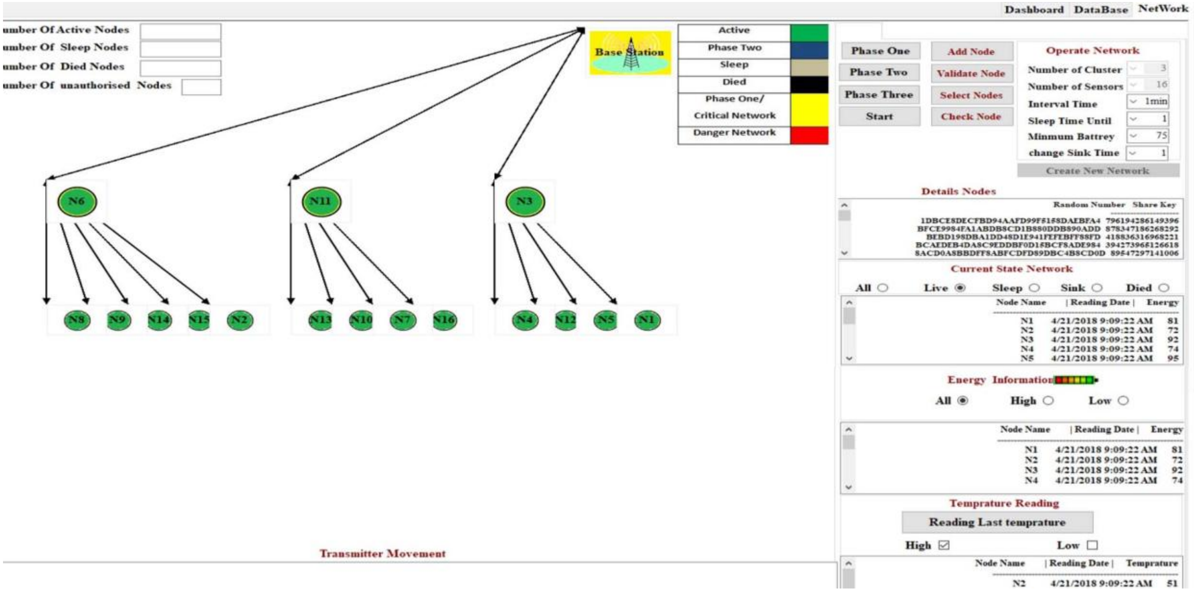


Fig. 9. Home page form

In Figure (9), the main GUI page is shown to explain the three phases of protocol and the main setting of simulation by entering the number of sensors, clusters ,time interval, sleeping time, battery and sink time. These parameters are used to create new network and the three phases is applied one by one to see the performance of it. Phase one represents the key redistribution phase by treatment the authenticated between main base station and each cluster head. In addition, the second phase performs the network initialization that activates the authentication between sensors. The last phase runs the authentication protocol for neighbored. After applying the three phases, the information of detailed node, current state network, energy information and temperature readings are appeared and stored in the database as shown in Figure (10).



umber Of Active Nodes
umber Of Sleep Nodes
umber Of Died Nodes
umber Of unauthorised Nodes

Operate Network

Number of Cluster 3
Number of Sensors 16
Interval Time 1min
Sleep Time Until 1
Minmum Battrey 75
change Sink Time 1

Details Nodes

Random Number	Share Key
1DBCE8DECFBD94AAFD99F5158DAEBFA4	796194286149396
BFCE9984FA1ABDB8CD1B880DDB890ADD	878347186268292
BEBD198DBA1DD48D1E941FEFEBFF88FD	418836316968221
BCAEDEB4DA8C9EDDBF0D15BCF8ADE984	394273968126618
8ACD0A8BBDFF8ABFCDFD89DBC4B8CD0D	89547297141006

Current State Network

All ○ Live ● Sleep ○ Sink ○ Died ○

Node Name	Reading Date	Energy
N1	4/21/2018 9:09:22 AM	81
N2	4/21/2018 9:09:22 AM	72
N3	4/21/2018 9:09:22 AM	92
N4	4/21/2018 9:09:22 AM	74
N5	4/21/2018 9:09:22 AM	95

Energy Information

All ● High ○ Low ○

Node Name	Reading Date	Energy
N1	4/21/2018 9:09:22 AM	81
N2	4/21/2018 9:09:22 AM	72
N3	4/21/2018 9:09:22 AM	92
N4	4/21/2018 9:09:22 AM	74

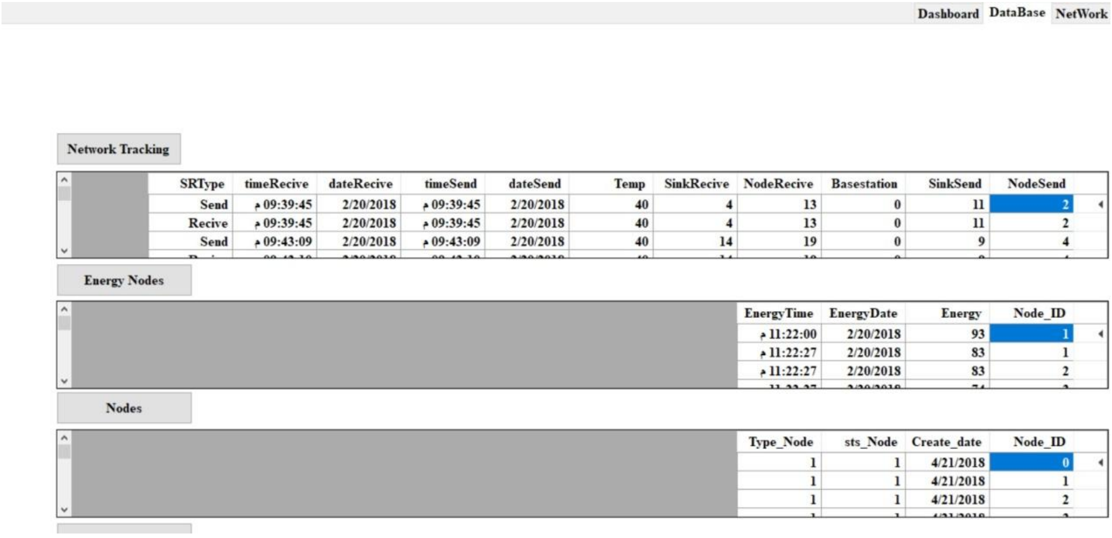
Temprature Reading

High ☑ Low □

Node Name	Reading Date	Temprature
N2	4/21/2018 9:09:22 AM	51

Fig. 10. Network Operation

The dashboard tab is also activated to show the recorded information in the database as shown in Figures (11). In addition, a statistical chart of the nodes status in terms of life and dead, active, sleeping, and unauthorized cases is shown in Figure (12).



Network Tracking

SRType	timeRecive	dateRecive	timeSend	dateSend	Temp	SinkRecive	NodeRecive	Basestation	SinkSend	NodeSend
Send	م 09:39:45	2/20/2018	م 09:39:45	2/20/2018	40	4	13	0	11	2
Recive	م 09:39:45	2/20/2018	م 09:39:45	2/20/2018	40	4	13	0	11	2
Send	م 09:43:09	2/20/2018	م 09:43:09	2/20/2018	40	14	19	0	9	4

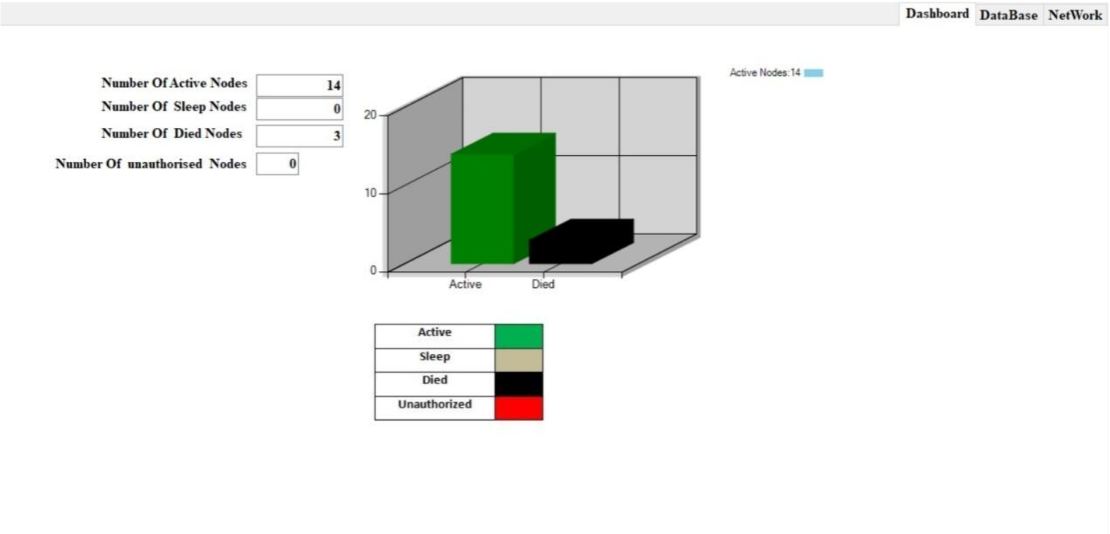
Energy Nodes

EnergyTime	EnergyDate	Energy	Node_ID
م 11:22:00	2/20/2018	93	1
م 11:22:27	2/20/2018	83	1
م 11:22:27	2/20/2018	83	2

Nodes

Type_Node	sts_Node	Create_date	Node_ID
1	1	4/21/2018	0
1	1	4/21/2018	1
1	1	4/21/2018	2

Fig. 11. Database Tab

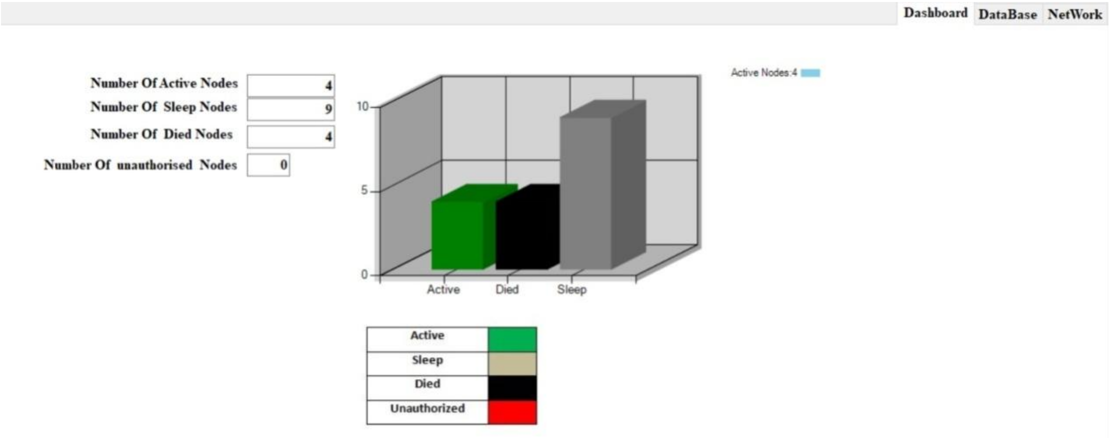


Number Of Active Nodes 14
Number Of Sleep Nodes 0
Number Of Died Nodes 3
Number Of unauthorised Nodes 0

Active	🟩
Sleep	▨
Died	■
Unauthorized	🟥

Fig. 12. Dashboard Tab

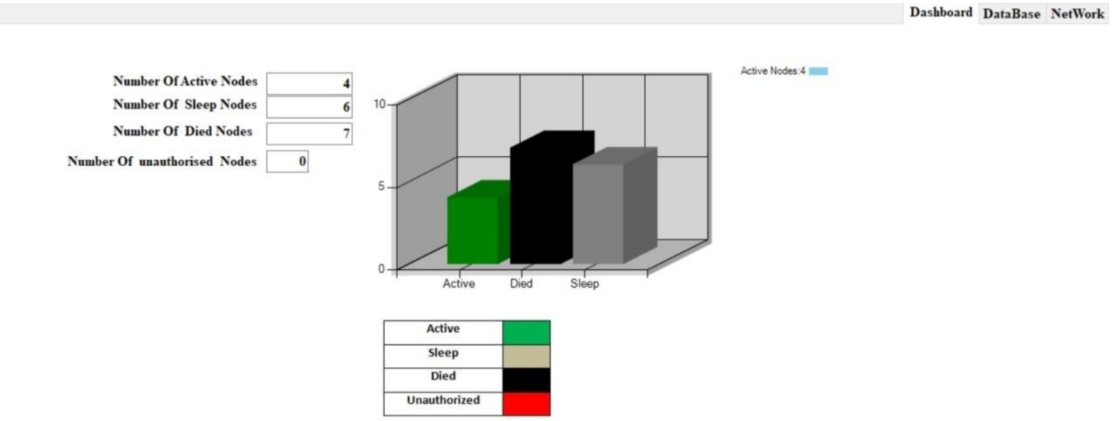
The statistical percentages of the nodes statuses are varieddepending on the number of active, sleep and died nodesas show in Figures (13) and (14).



Number Of Active Nodes 4
Number Of Sleep Nodes 9
Number Of Died Nodes 4
Number Of unauthorised Nodes 0

Active	🟩
Sleep	▨
Died	■
Unauthorized	🟥

Fig. 13. Dashboard Percentage



Number Of Active Nodes 4
Number Of Sleep Nodes 6
Number Of Died Nodes 7
Number Of unauthorised Nodes 0

Active	🟩
Sleep	▨
Died	⬛
Unauthorized	🟥

Fig. 14. Changing in Dashboard Percentage

V. SIMULATION RESULTS

In order to test the two parts of the proposed system key generation, different case studies are considered. the generated keys are tested according to NIST standards to evaluate the randomness. The generated key using the proposed method tested following the NIST standards. The 128-bit key is tested and compared with two traditional methods. These methods are ECC with Deffe-Hellman and LFSR-Geffe generator. Table (1) shows the obtained results.

Table (1): NIST based five tests

	ECC with Deffe-Hellman	LFSR with Chaotic map	Proposed Method	Stander Rate
Frequency test	0.740625	0.49375	0.165625	Must be≤ 3.8415
Serial test	1.134	1.431872	0.37176	Must be≤ 5.9915
Poker test	5.50471	7.561868	5.69521	Must be≤ 14.0671
Run test	3.39142	4.65126	2.342616	Must be≤ 9.4877
Auto correlation test	1	1	1	Must be≤ 3.84

From Table (1), it is appeared clearly that the proposed method is able to generate a more random key according the considered five tests of NIST. It is noted that the Poker test shows that the ECC with Deffe-Hellman is better than the proposed method by (0.186), which is a bit low deference.

At the other hand, the complexity in terms of execution time of the proposed method of key generation is tested amongst the adopted traditional methods as shown in Table (2).

Table (2): Execution time

	ECC with Deffe-Hellman	LFSR with Chaotic map	Proposed Method
Average execution time (millisecond)	533	397.1	433.7
Average execution time (second)	0.533	0.397	0.434
Average execution time (minute)	0.008	0.006	0.007

From this table, the execution time of the proposed system is lower than ECC with Deffe-Hellman. In addition, The LFSR with Chaotic map is lower than the proposed method by almost 100 milliseconds. This time difference can be ignored due to low value.

VI.CONCLUSION

A developed system for authentication and key management lightweight protocol for WSN was introduced. The introduced system included generated a more random key used for authentication amongst the WSN phases. This part adopted the combination of LFSR, Choatic map, and SIGABA technique for generating the required 128 bit length keys. Moreover, a database was built to store the sensor readings. At the other hand, a simulator was proposed to simulate the adopted protocols and the introduced system without the need for conventional one of NS2 and NS3. The conventional simulators were not supporting the lightweight protocols. The proposed system was tested over five tests based on NIST standards and execution time. The obtained results showed the superior randomness of the proposed key generation method and accepted execution time required.

REFERENCES

[1] Sagar D. Dhawale, Dr. B. G. Hogade, Dr. S. B .Patil, “Design and Implementation of a Dynamic Key Management Scheme for Node Authentication Security in Wireless Sensor Networks”, International Journal of Science, Engineering and Technology Research (IJSETR), Vol 4, No 4, 2015.

[2] Lidong Zhou and Zygmunt J. Haas, “Securing Ad Hoc Networks”, IEEE network, special issue on network security, P.P. 24-30, 1999.

[3] S. Raja Rajeswari1 and V. Seenivasagam, “Comparative Study on Various Authentication Protocols in Wireless Sensor Networks”, the Scientific World Journal, 2016.

[4] Danyang Qin, Shuang Jia, Songxiang Yang, ErfuWang, and Qun Ding, “A Lightweight Authentication and Key Management Scheme for Wireless Sensor Networks”, Journal of Sensors, 2016.

[5] Musheer Ahmad1 and Omar Farooq2 S. Raja Rajeswari1 and V. Seenivasagam, “Comparative Study on Various Authentication Protocols in Wireless Sensor Networks”, The Scientific World Journal, 2016, "Chaos Based PN Sequence Generator for Cryptographic Applications", International Conference on Multimedia, Signal Processing and Communication Technologies, 2011.

[6] Kamanashis Biswas,"Energy Efficient Secure Routing in Wireless Sensor Networks",School of Information and Communication Technology,Griffith Sciences,Griffith University.Submitted in fulfilment of the requirements of the degree of Doctor of Philosophy, March, 2016.

[7] A. Menezes, P. van Oorschot, and S. Vanstone," Handbook of Applied Cryptography", CRC Press, 1996.

[8] Haider Mohammed Abdul-Nabi Al-Mashhadi,"Simulation of Proposed Secure Method for wireless Sensor Network", PhD Thesis, Department of Computer Science in University of Technology, 2015.

[9] Heather Ellie Kwong," Cryptanalysis of the SIGABA Cipher", A M.Sc. Thesis Presented to The Faculty of the Department of Computer Science San José State University, December 2008.

[10] Mark Stamp and Wing On Chan," SIGABA: Cryptanalysis of the Full Keyspace", Department of Computer Science, San Jose State University, San Jose, California, 2007