

DETECTION OF THE DSSS SIGNAL WITH VARIABLE PN CODES USING THE ANN

Aya Y. Khudhair ¹, Rajaa Aldeen A. Khalid ²

^{1,2} College of Information Engineering, Al-Nahrain University, Baghdad, Iraq
ayya9137@gmail.com ¹, rajaaldeen@coie-nahrain.edu.iq ²

Received:20/4/2021, Accepted:21/5/2021

Abstract- The Direct sequence spread spectrum(DSSS) systems are one of the solutions for reliable and secured communications. Also, it is one of the approaches used by signals for transmitting bandwidth larger compared to the satisfied frequency related to the original information. The communication systems of Spread Spectrum were vital to suppress interference, complicating the detection and processing of secure communications, the technology of spread spectrum (DSSS) has been initially created for military applications. pseudo-noise PN code is considered the primary key that should be the same length and type in both the transmitter and the receiver. in the previous works, the PN code must be known by the receiver for retrieving the original data signal. In this work, it is considered that the transmitter sends data bits and wants to protect the sent data by making each bit send with variable PN code consisting of 127 bits randomly without knowing the receiver of that. Here the artificial neural network (ANN) is used as a tool in the receiver to find the correct PN code for each initial value of 7 flip-flops. so, the receiver could detect the transmitted data with BER = 0.

keywords: DSSS, PN code generator, ANN, 7 flip-flops.

I. INTRODUCTION

Communication intelligence (COMINT) receivers are becoming a necessity due to the widespread use of the spread spectrum in both civilian and military applications [1]. Wireless communications have been increasingly important since early attempts in the late 1800s and were widely embraced on civil and military markets worldwide. New problems, risks and opportunities for community and government agencies are being developed via the development of wireless communication networks. While there is a risk of breaking privacy laws, electronic surveillance in law enforcement, counter-terrorism and military operations has become an essential technology [2]. A benefit may be achieved by obtaining information from, or interfering with, an adversary's contact signals by the interception of wireless communications signals. Only after the existence of the contact signal is detected will surveillance be carried out. Data signals, however, are usually not meant for third-party receipt and protection protocols are also used for protecting the information transmission from violation. Therefore, advanced approaches were needed for adequately detecting the existence of the contact signal of interest and to extract information from it [3]. The spectrum spreading system is ensuring a very high level of communication security. This system will spread information across a wide bandwidth and make it difficult for a narrow band receiver for deciphering significant information. Similarly, the same system might prevent a high degree of deliberate or accidental rejection [4]. The spread spectrum (SS) has been defined as a means of transmission in which the signal occupies bandwidth much over the minimum necessary to send the information, the band spread is accomplished by utilizing a code that is independent of the data and a synchronized reception with the code at the receiver is used for de-spreading and subsequent data recovery.

II. DIRECT SEQUENCE SPREAD SPECTRUM

Data bits of DSSS technology are multiplied to spread bandwidth by pseudo-noise (PN) sequence. In addition, spread bits were transmitted at high data rates. In addition, the PN sequences were deterministic, regular as well as binary noise

sequences. Also, it is referred to as random pseudo-noise, as the user who doesn't know the code seems random. DSSS is a common kind of SS technology that easily accesses multi-point communication through strong anti-interference, resistance to multipath interference [5]. In a traditional DSSS system in the transmitter the input data bits are multiplied by module-2 (EXOR-ing operation) with the spreading signal (PN code), the transmitted data signal at the receiver must multiply by the same PN code to enable the receiver for recovering the data signal. Fig. 1 is showing the steps of signal processing on the receiver and transmitter of the DSSS system. Also, DSSS work via the modulation of a data signal with the signal of the carrier, after that spreading it with the spreading signal (PN code) with the frequency that is more than the frequency that is related to the data signal (i. e. the bit-rate). Concerning the receiver, the de-spreading signal was created from multiplication via the same PN code in the transmitter, which reverses the transmitter's process of encoding. On the receiver, the last process is filtering for eliminating the other signals, whereas maintaining the data signal [6].

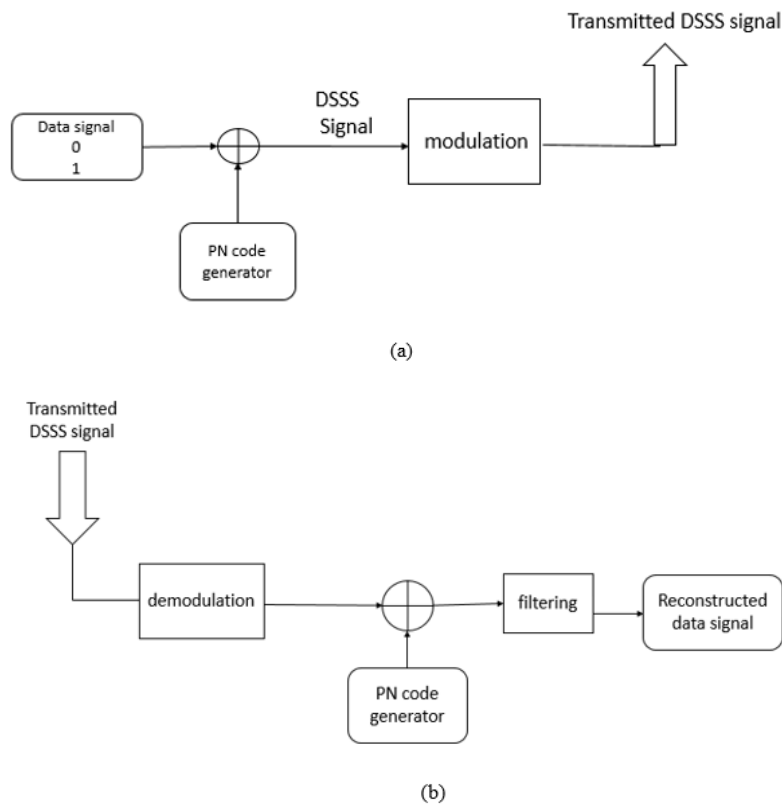


Figure 1: The traditional DSSS system (a) Transmitter (b) Receiver [6]

- PN code generator Fig. 2 displays the block diagram for the implementation of a PN generator. A sequence with polynomial $f(x) = 1 + X + X^7$ has been created as maximal linear code, with a PN generated with two feedback from 7th stage D Flip-Flop Taped (X^7, X) XOR at the input of 1st stage D flip flop for achieving maximum length of 127 bit [7].

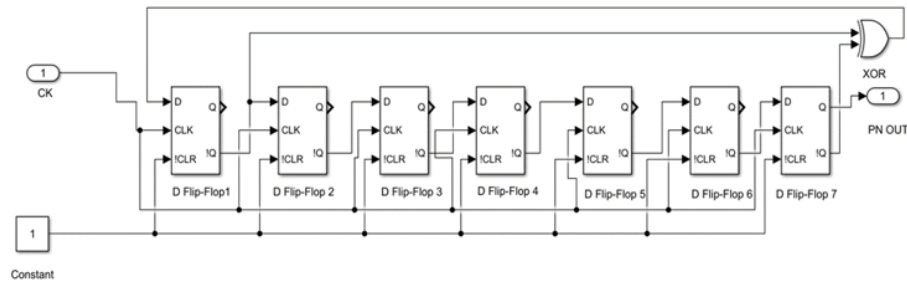


Figure 2: Block diagram of PN code generator [7]

III. THE CONTRIBUTION OF THIS WORK

Communication intelligence (COMINT) receivers are becoming a necessity due to the widespread use of the spread spectrum in both civilian and military applications [1]. In previous DSSS systems in the transmitter the input data bits are multiplied by the spreading signal (PN code), the transmitted data signal at the receiver must multiply by the same PN code to enable the receiver for recovering the data signal. The problem of this work If the receiver doesn't know the specified PN code that used in the transmitter. So, it can't detect the data signal. Therefore, the utilizing of the Artificial Neural Network (ANN) is used to solve this problem by finding the correct PN code.

A. Artificial Neural Network (ANN)

Artificial neural network (ANN) is a flexible and powerful machine learning technique. However, Artificial neural networks (ANN) mimic the human brain in processing input signals and transform them into output signals. The ANN model was developed using the Back Propagation (BP) algorithm. BPANN (Artificial Neural Network) standard topology includes three Layers: the input layer, where data is loaded into the network, hidden layer The data is processed, and the output layer on which the input data are obtained [8]. The ANN structure is shown in Fig. 3 where y represents the output signal, x represents input variables, $\Phi()$ represents activation function, and w represents the weight value that has been assigned to every one of the input variables. There are n input variables. Every one of the input variables is in analogue to regression model predictors. Weight is every predictor's coefficient [9]. Every one of the neurons receives a multiplied version of the inputs and arbitrary weight values that have been added than with the value of the static bias (unique to every one of the neuron layers), which is passed afterwards to a suitable activation function, deciding the final value that will be given out of the neuron. There have been many different activation functions that are available as per the input values' nature. As soon as output has been produced from the final layer of the neural network, loss function (i.e. output vs. input) is computed, and the back-propagation has been carried out where weights are adjusted to make the minimum loss. Finding the optimum weight values is what the entire operation is focused on [10]. An activation function is used to produce the output and it is also referred to as a squashing function, in that it squashes (limits) the amplitude range of the output signal to a finite value. The bias b_k has the effect of increasing or decreasing the net input of the activation

function, depending on whether it is positive or negative, respectively. Mathematically, the output on the neuron k can be described as

$$y_k = \varphi\left(\sum_{i=1}^m x_i \cdot w_{ki} + b_k\right) \quad (1)$$

where $x_1, x_2, x_3, \dots, x_m$ are the input's signals. $w_{k1}, w_{k2}, w_{k3}, \dots, w_{km}$ are the respective weights of neuron, b_k is the bias, φ is the activation function [11].

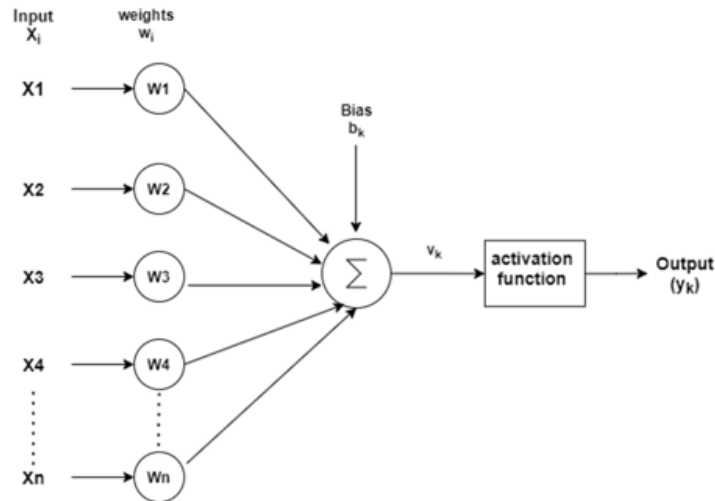


Figure 3: ANN model [9]

B. The Proposed DSSS System

In the proposed system is considered the receiver doesn't know the correct PN code that is used in the transmitter. The PN code generator consists of 7 flip-flops and that means the length of the PN code is $2^7 - 1 = 127$ bits, the receiver doesn't know the PN code that used in the transmitter and the PN code can be changed by changing the initial value of 7 Flip-Flops, so, we use the ANN to find the initial value of the PN code that is used in the transmitter to make the receiver recovers the original data bits as shown in Fig. 4. At the beginning is a generation of 127 vectors of length 127 bit PN code as shown in Table I.

- Preparation of the data to the ANN
 - Training the input Train the ANN with matrix size 127×254 , Each column contains 127 bits, which are produced from each initial value of 7 flip-flops for both bits (0,1) as shown in Table II.
 - Training the output Training the output with matrix size (7×254) with all possible cases of the initial value of 7 flip-flops as shown below in Table III to generate 127 vectors of the different PN code consisting of 127 bits for both data bits (1,0).

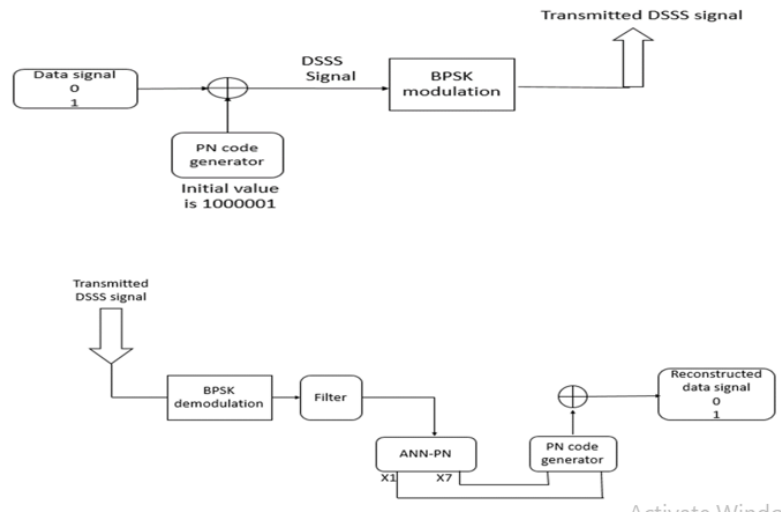


Figure 4: The proposed DSSS system

TABLE I
 127 Case of Initial Values to Generate 127 Different PN Codes

Initial value	X7	X6	X5	X4	X3	X2	X1	Initial value Polynomial form	PN code(127 bit)
1	0	0	0	0	0	0	1	1+X1	PN 1
2	0	0	0	0	0	1	0	1+X2	PN 2
3	0	0	0	0	1	0	0	1+X3	PN 3
4	0	0	0	1	0	0	0	1+X4	PN 4
5	0	0	1	0	0	0	0	1+X5	PN 5
6	0	1	0	0	0	0	0	1+X6	PN 6
7	1	0	0	0	0	0	0	1+X7	PN 7
.
.
.
.
65	1	0	0	0	0	0	1	1+X1+X7	PN 65
.
.
.
.
127	1	1	1	1	1	1	1	1+X1+X2+X3+X4+X5+X6+X7	PN 127

After the training the network with input and the output values make the following:

- convert those values by the bipolar converter to (1, -1)
- Randomize the training values because when Train the output with 1 and -1 showed the problem in the training operation So, we use an equation:

$$output1 = \exp(-0.5 * output) \tag{2}$$

to change these values , where 1 become 0.6065 and -1 to 1.6487.

TABLE II
 The process of training the input for the ANN

Initial value	Bit	PN Code
1	1	PN1-1
	0	PN1-0
2	1	PN2-1
	0	PN2-0
3	1	PN3-1
	0	PN3-0
.	.	.
.	.	.
.	.	.
127	1	PN127-1
	0	PN 127-0

TABLE III
 The process of training the output for the ANN

Received PN code	Initial value	Binary form
PN 1-1	1	0 0 0 0 0 1
PN 1-0		
PN 2-1	2	0 0 0 0 1 0
PN 2-0		
.	.	.
.	.	.
.	.	.
PN 127-1	127	1 1 1 1 1 1 1
PN 127-0		

- The results of training the ANN It is noticed the results of training is very good where there is no error and the best value of Mean Square Error (MSE) as shown in Fig. 5. In the training process, the MSE equals 6.35×10^{-31} at (35 epoch) iteration, this is very good training that means there is no error in attaining the target.
- The ANN Layers The ANN consists of two layers :
 - layer 1: contains 20 nodes for training the input 127×254 (127 vector of 127 PN code).
 - layer 2: contains 20 nodes for training the output 7×245 (that is the 127 case of initial value of 7 flip-flops).

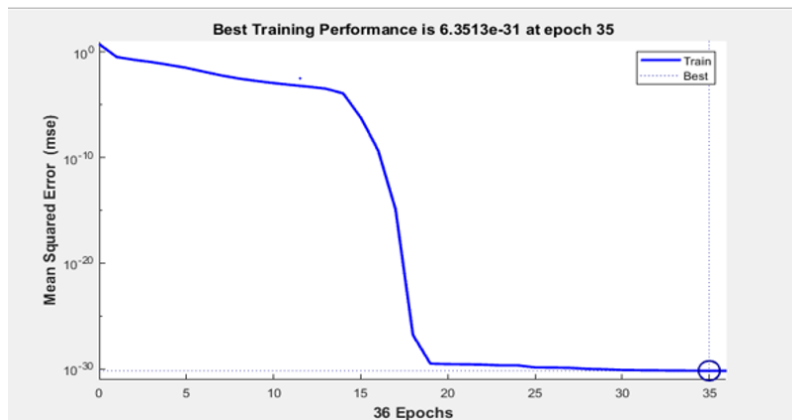


Figure 5: The best value of MSE in the training process

IV. RESULTS OF TESTING THE PROPOSED SYSTEM

After the training process, the performance of the trained network will be evaluated by applying data to it and checking whether its outputs are still relevant to the targets. Matlab R2020a is used to measure the network performance. The initial value of the received PN code is detected correctly by the ANN. In the testing process, The receiver is perfectly succeeding 100% in finding the initial value of the correct PN code for the transmitted data either 1 or 0 without any knowledge about the initial value in the transmitter. When testing the ANN, it succeeded in finding the initial value to generate the correct PN code that is used in the transmitter. As shown in Fig. 6 below if the transmitted PN code is PN1 this means the ANN produce the first case of the initial value 0 0 0 0 0 1, and also we check if the transmitter used the last case of the PN code, the ANN output was 1 1 1 1 1 1 as shown in Fig. 7.If the transmitter sends data bit with 100 bits and it changes the PN code randomly to each transmitted bit for security purposes, the results show the receiver succeeds in detecting the correct initial value to each PN code of 127 cases and retrieving the transmitted data bits 100% with BER = 0 as shown in Fig 8.

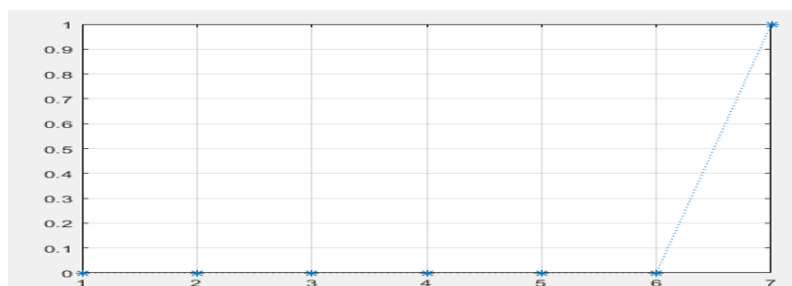


Figure 6: The result of testing for the first initial value 000001

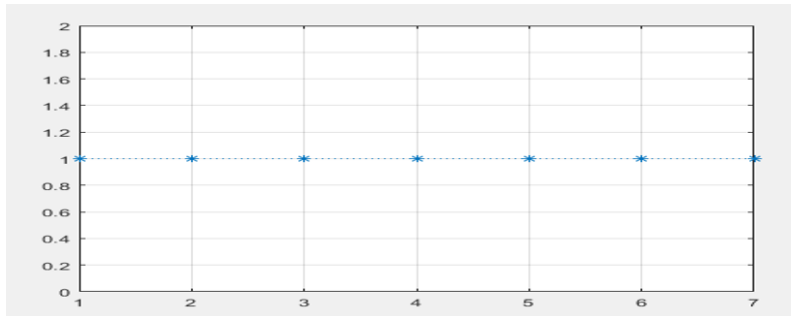


Figure 7: The result of testing for the last initial value 1111111

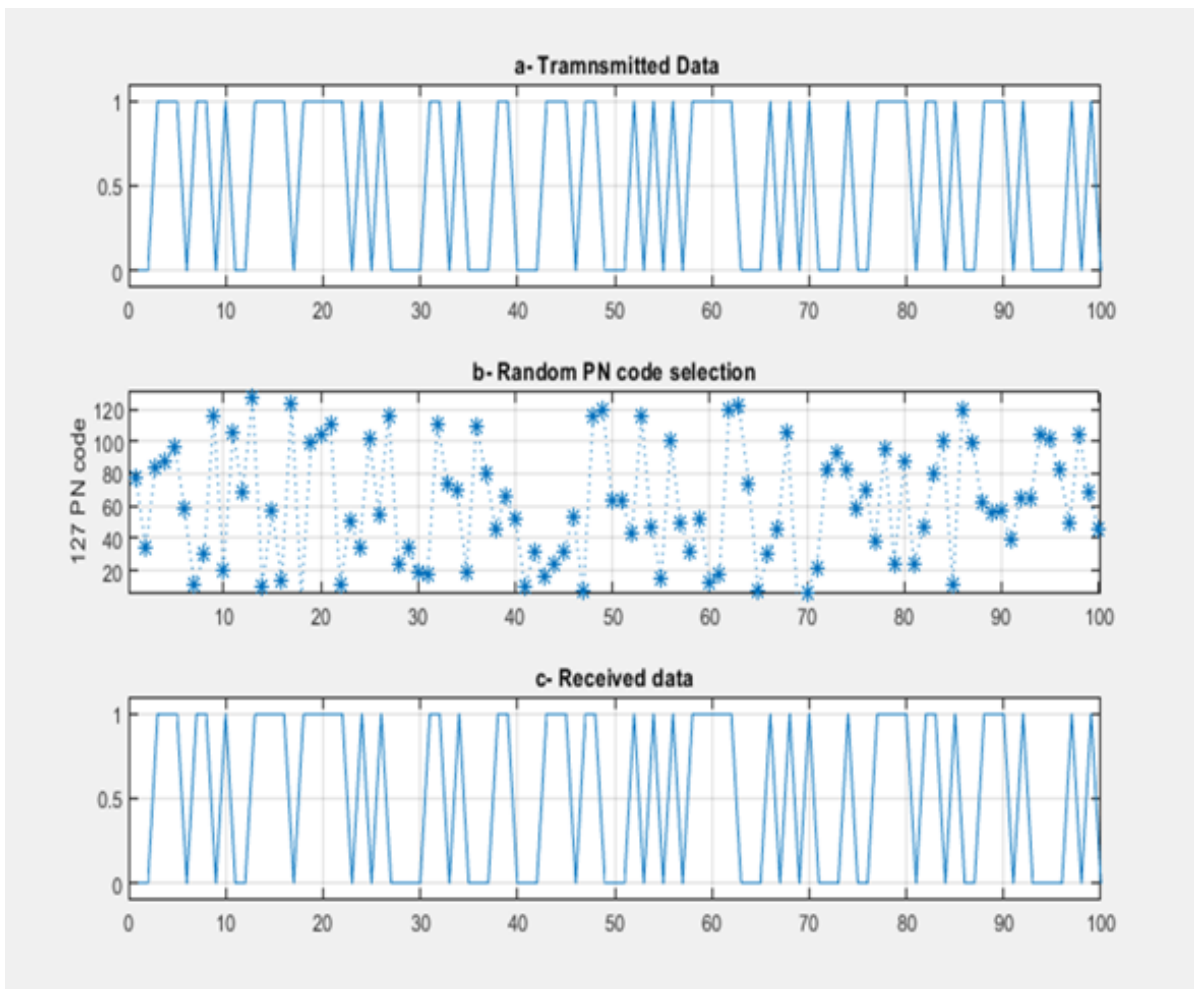


Figure 8: Detection of the transmitted signal with random PN codes by the receiver

V. CONCLUSION

The degree of security or privacy is a function of the codes utilized. Only the authorized receiver can detect and demodulate the data because it has the key (PN code) for spreading the spread spectrum signal, DSSS signals can be made to have any degree of message privacy that is desired when the transmitter changes the PN code randomly to each bit for security purpose, the receiver doesn't have any knowledge about the specified PN code that is used in the transmitter so it can detect the initial value for every 127 cases of PN code with 7 flip-flops using the ANN for any different case of PN code and it recovers the transmitted bits 100% with BER = 0. Also, the receiver can detect the signal of interest among many of the competing other signals transmitted in the same channel by finding the PN code for each signal using the ANN.

REFERENCES

- [1] D. Torrieri, "Principles of Spread-Spectrum Communication Systems" , 2018.
- [2] H. N. Abdullah, "Simulation of a DS/SS System Using MATLAB-Simulink" , Engineering and Development Journal, Iraq, ISSN 1813-7822, No. June, 2014.
- [3] J. D. Vlok, "Detection of Direct Sequence Spread Spectrum Signals" , M.Eng (Electronic), 2003.
- [4] R. aldeen Abad Khalid, "A Cheap Zilog Mp Applied To Spread Spectrum Systems Design" , J. Eng. Sustain. Dev. , doi: 10.31272/jeasd, 2018.
- [5] M. J. Abbas, M. Awais, and A. U. Haq, "Comparative Analysis of Wideband Communication Techniques: Chirp Spread Spectrum and Direct Sequence Spread Spectrum" , Int. Conf. Comput. Math. Eng. Technol. Inven. Innov. Integr. Socioecon. Dev. iCoMET 2018-Proc. , doi: 10.1109/ICOMET, 2018.
- [6] F. Ghalib, A. Karim, E. Kurniawan, and E. Yuniarti, "Performance Evaluation and Mathematical Analysis of Direct Sequence and Frequency Hopping Spread Spectrum Systems Under Wideband Interference" , No. November, 2018, doi: 10.26555/ijain.v4i3.261.
- [7] R. aldeen Abad Khalid, "Design and Implementation of Direct Sequence Spread Spectrum System" , University of Technology, thesis, 2002.
- [8] S. P. Siregar and A. Wanto, "Analysis of Artificial Neural Network Accuracy Using Backpropagation Algorithm In Predicting Process (Forecasting)" , IJISTECH (International J. Inf. Syst. Technol. , Vol. 1, No. 1, pp. 1-34, 2017, doi: 10.30645/ijistech.v1i1.4.
- [9] Z. Zhang, "A Gentle Introduction to Artificial Neural Networks " , No. October, 2016, doi: 10.21037/atm.2016.06.20.
- [10] "https://www.mygreatlearning.com/blog/types-of-neural-networks/" , Accessed Online in 23/12/2020 .
- [11] A. Zayegh and N. Al Bassam, "Neural Network Principles and Applications" , Digit. Syst. , 2018, doi: 10.5772/intechopen.80416.