Secret Message Hiding in WAVE PCM Sound File

Raja Salih
Assist lecturer
Institute of medical technology

Zaid Sadiq naama
Assist lecturer
College of science

Sherna Aziz Tome
Assist lecturer
College of medicine

Abstract

Communications today has been done among millions of users using many application devices like internet or satellite communication channels, the services they can transform many varieties of files like text, images, videos, and audio among different places. Therefore the security of data has been of extreme importance in today's information-based society, including the fields of military, diplomacy, corporation, medicine, and etc….

A form of data hiding is steganography, which is contemporary way for protecting the information by embedding data into digital media for the purpose of copyright, and sending secret messages.

In this paper a scheme of steganography system for hiding secret text message in audio file WAV, (Windows Audio Visual) format is proposed, the hiding mechanism was based on using Low-Bit Encoding (LSB) Least Significant Bit substitution techniques.

To support the immunity of the hiding system, encryption methods with some other support methods (i.e., hiding and hopping) were added to the proposed hiding system, a pseudo random number generator has been designed and implemented to generate non-uniform integer jumps between successive hiding events. The jumps mechanism of the suggested generator is based on the linear feedback shift register of length 23 with feedback function $F(x)=1+X+X^{23}$ produce the maximum period $2^{23}-1= 8388607$.

Keywords: ciphertext**,** cryptography, decryption, encryption , PCM, plaintext, steganography.

إخفاء الرسائل السرية بواسطة الملف الصوتي Wave PCM

رجاء صالح محمد حسن
مدرس مساعد
المعهد الطبي التقني المنصور

زيد صادق نعمه
مدرس مساعد
كلية العلوم للبنات / جامعة بغداد

شيرنه عزيز توما
مدرس مساعد
كلية الطب / جامعة بغداد

الخلاصة

تتم الاتصالات في الوقت الحاضر من قبل ملايين المستخدمين بواسطة أنواع متعددة من التطبيقات مثل الأنترنيت أو قنوات الأتصال عبر الأقمار الصناعية ،وهذه الخدمات من خلالها يمكن تحويل العديد من أنواع الملفات مثل النصوص ، الصور ، وأشرطة الفديو والرسائل الصوتية بين مواقع مختلفة من الكرة الرضية . وبالتالي فأن أمن البيانات وسريتها لها أهمية بالغة وتتطلب عناية قصوى أثناء تداولها في مجتمعات اليوم التي تعتمد على المعلومات والبيانات أساسا لها في المجالات العسكرية ، الدبلوماسية ، الشركات التجارية والمؤسسات الطبية وحتى على مستوى الأفراد.

اقتراح نظام لأخفاء النصوص السرية في صيغة الملفات الصوتية WAV (Windows Audio Visual) وقد اعتمدت آلية الأخفاء بطريقة (LBE) Low-Bit encoding اوتقنيات الإستبدال Least Significant Bit (LSB) .

ولدعم حصانة نظام الإخفاء هذا تم اضافة طرق تشفير أخرى encryptionمع أساليب الدعم الأخرى مثل (الأختباء والتنقل) ، اضيفت هذه الطرق لدعم نظام الإخفاء المقترح وتم تصميم مولد رقم عشوائي زائف لتوليد عدد صحيح غير موحد بين أحداث الأخفاء المتعاقبة .

ان ميكانيكية القفز لمولد عشوائي مقترح يعتمد على مسجل التزحيف ذو التغذية الخلفية الخطية (LFSR) بطول 23 وبدالة تغذية خلفية $F(x)=1+X+X^{23}$ بحيث تولد اعلى قيمة $2^{23}-1= 8388607$

1- Introduction

Steganograph (literally, covered writing) is the hiding of secret messages within another seemingly innocuous message, or carrier (cover). Digital carriers include email, audio, and video messages, disk space, disk partitions, and images [1].

The subjective quality of the audio data depending on our hearing sense could not recognize all voices and noises that are accompanied with original wave media. Data hiding in audio signals are especial challenge, because the (HAS) Human Auditory System operates over a wide dynamic range. The HAS perceives over a range of power greater than billion to one and range of frequencies greater than one thousand to one.[2]

Sensitivity to additive random noise is also acute. When performing data hiding on audio, one must exploit the weakness of the HAS, while at the same time being aware of extreme sensitivity of the human auditory system. [3]

In this research has been the builder a hiding system of secret messages in wave files without producing any significant distortion

2 –Least Significant Bit Encoding (LSB)

Low-bit encoding is the simplest way to embed data into other data structure. By replacing the least significant bit of each sampling point by a code binary string, we can encode a large amount of data in an audio signal. The bit rate will be 8 Kbps in an 8KHz sampled sequence and 44Kbps in a 44KHz sampled sequence.[4]

The simple algorithm of LSB for low-bit encoding is as follows :

- Represent the object as vector of integers.
- Change the least significant bit in either all or some integers to represent a 1 or 0 in the mark

Depending on the amount of embedded information and the amount when the wave media carries, it is quite unperceivable. For example the first thing to do is to hide an ASCII code of A, which represent 01000001, and then part of wave media cover data will be

(11101011 10001011 10101010 11001010 10001010 10111101 11111111 01001111)

After information is hidden the wave file will be

(1110101**0** 1000101**1** 1010101**0** 1100101**0** 1000101**0** 1011110**0** 1111111**0** 0100111**1**)

3 - Wave PCM Sound file Format

The most popular waveform coding technique used to present the human speech using Pulse Code Modulation (PCM).[3]

The WAVE file format is a subset of Microsoft's RIFF specification for the storage of multimedia files. A (RIFF) resource enter change file starts out with a file header followed by a sequence of data chunks as shown in figure (1). A WAVE file is often just a RIFF file with a single "WAVE" chunk which consists of two sub-chunks—a "fmt" chunk specifying the data format and a "data" chunk containing the actual sample data which is the "canonical form". As shown on table. (1)

File offset (byte)	Field name	Field size (byte)	The RIFF chunk descriptor
0	Chunk ID	4	The "RIFF" chunk descriptor The format of concern here is "WAVE ", which requires two sub-chunks:"fmt" and "data"
4	Chunk size	4	
8	Format	4	
12	SubChunk1ID	4	The "fmt" sub-chunk Describes the format of the sound information in the data sub-chunk
16	SubChunk1size	4	
20	Audio Format	4	
22	Num Channels	2	
24	Sample Rate	2	
28	Byte Rate	4	
32	Block Align	2	
34	Bit spersample	2	
36	SubChunk2ID	4	The "data' sub-chunk Indicates the size of the sound information and contains the raw sound data
40	SubChunk2size	4	
44	Data	Subchunk2size	

Figure (1) The Canonical WAVE file format [4]

4 - Linear Feedback Shift Register

A commonly used method for generating binary sequences, especially pseudo-random sequences, is to feed a binary function of the state of a shift register back to its input. The stages of shift register serve as the input of logical circuit whose output is connected to the input of the shift register. Shown in figure (2)

An n-stage shift register $s_0, s_1, \ldots, s_{n-1}$. The contents of the stages change in time with a clock pulse according to the rule:

Let $S_i(t)$ denote the content of si after the t^{th} time pulse $(t=0,1,2,\ldots)$.

$S_{n-1}(t+1)=f(c_0s_0(t),\ c_1s_1(t),\ldots\ c_{n-1}s_{n-1}(t))$, where the c_i are all specified as 0 or 1.

The function build is called the feedback function of the register and if $f(s_0,s_1, \ldots,s_{n-1})=c_0s_0+c_1s_1+\ldots+c_{n-1}s_{n-1}$ then the register called linear register. This is represented by the $c_i=1$ denotes a closed connection and $c_i=0$ an open one.[6]

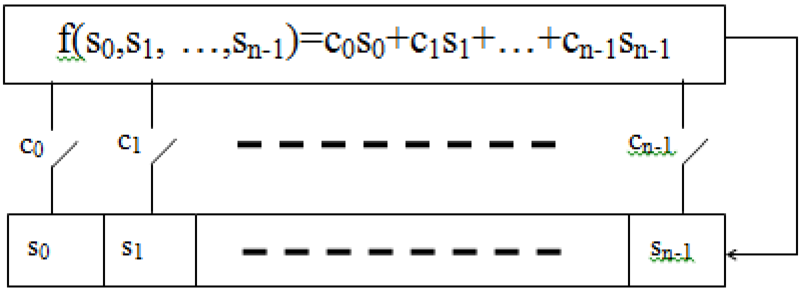

Figure (2) : Linear feed back shift register

A assume $c_0=1$ so that $s_{n-1}(t+1)$ is dependent on $s_0(t)$. Let $s_t=s_0(t)$, an infinite binary sequence denoted s_t satisfy the linear recurrence relation as $s_{t+n}=\sum_{i=0}^{n-1} c_i s_{t-i}$, for t=0,1,….n.

The shift register has been identified as characteristic polynomial as shown [8].

$F(x)=1+c_1x+c_2x^2+\ldots+c_{n-1}x^{n-1}+x^n$ (remembering $c_0=1$).

5 - Cryptography

Steganography is a way that deals with finding the best place in cover media to hide data.If the data encrypted before hiding it, this will give more security immunity to the cover data.

In this paper research, before hiding the secret data were encrypted by using stream cipher generator. [7]

5.1 - Stream Cipher

One of the cryptographic primitives used to ensure secure communication over public and unsecured channels (such internet, mobile) is the stream cipher. In a stream cipher the plaintext is encrypted on bit by bit basis. In the encrypting of data flow transmitted, the key is fed into an algorithm called running key generator (RKG) to generate a long pseudorandom binary sequence. This “Key Stream” is then mixed with the plaintext sequence, usually by using exclusive-or (XOR bitwise module 2 additions) logic gate, to produce the cipher text. A typical stream cipher is shown in figure (3)

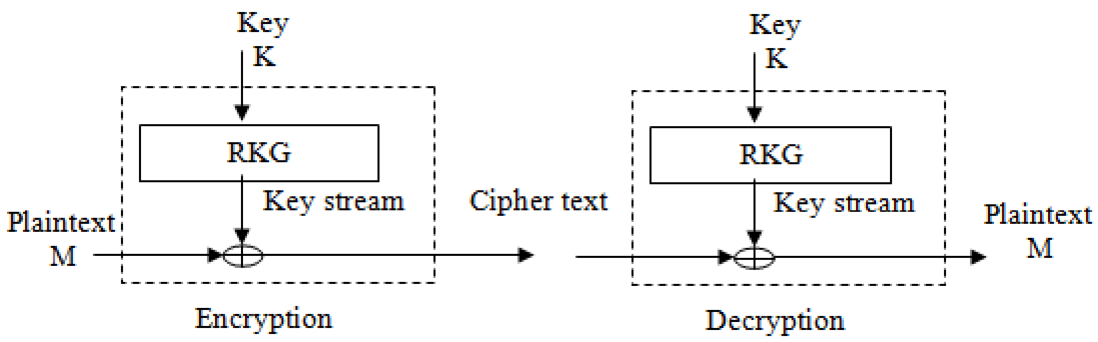

Figure (3) Stream Cipher

A common type of RKG employed in stream cipher system consists of n (mostly maximum length) LFSRs whose output sequences are combined in a nonlinear function F to produce the key stream.[7]

5.2- Encryption and Decryption

Cryptographic algorithm have been carefully designed for maximum security, it divide into two subsystem:

Driving subsystem, which are consists of 8-LFSRs with maximum period are corresponding feedback polynomials

$F_1(X)=1+X^{13}+X^{33}$ $\quad$ $F_2(X)= 1+X^3+X^{31}$

$F_3(X)= 1+X^2+X^{29}$ $\quad$ $F_4(X)= 1+X^3+X^{28}$

$F_5(x)= 1+X^3+X^{25}$ $\quad$ $F_6(X)= 1+X^5+X^{23}$

$F_7(x)= 1+X^{14}+X^{17}$ $\quad$ $F_8(X)= 1+X^{14}+X^{17}+X^{18}+X^{19}$

The second subsystem is non-linear compost combining subsystem F. which represent a 2D matrix 16x8 bits each bit addressed by driving subsystem, the output determined through the intersection of row by first 4 LFSRs (0..15 row) and column by next 3 LFSRs (0..7 row) the result mixed with the output of the register 8 using x or to produce the key sequence, as shown in figure (4)

6 -

original code 6bit			b_5	b_4	b_3	b_2	b_1	b_0
	b_5	b_4	b_3	b_2	b_1	A_0		
	b_5	b_4	b_3+b_5	b_2+b_4	b_1+b_3	$b_0 + b_2$	b_1	b_0
New extended code 8bit	a_7	a_6	a_5	a_4	a_3	a_2	a_1	a_0

Cryptographic Algorithm

Build a coding table of 6-bit has been build of printable keyboard characters as in the coding table(1). The basic key is 16-character of 6-bit extended into 8-bit by using the original character shift left 2 bits and x.

Example:

After extended the basic key ino 128 bits we use this bits to initial the driving and the nonlinear combining parts, each shift of driving subsystem will produce one bit output from combining and then to produce key bit.

original code 6bit			b_5	b_4	b_3	b_2	b_1	b_0
	b_5	b_4	b_3	b_2	b_1	A_0		
	b_5	b_4	b_3+b_5	b_2+b_4	b_1+b_3	b_0+b_2	b_1	b_0
New extended code 8bit	a_7	a_6	a_5	a_4	a_3	a_2	a_1	a_0

dec	code	binary		dec	code	Binary		Dec	code	binary		dec	code	binary
0	S	000000		16	T	010000		32	f	100000		48	3	110000
1	q	000001		17	9	010001		33	C	100001		49	d	110001
2	t	000010		18	/	010010		34	8	100010		50	J	110010
3	p	000011		19	m	010011		35	7	100011		51	H	110011
4	x	000100		20	Q	010100		36	O	100100		52	k	110100
5	g	000101		21	c	010101		37	n	100101		53	L	110101
6	0	000110		22	y	010110		38	G	100110		54	K	110110
7	w	000111		23	u	010111		39	X	100111		55	E	110111
8	R	001000		24	i	011000		40	Z	101000		56	6	111000
9	o	001001		25	b	011001		41	5	101001		57	V	111001
10	B	001010		26	M	011010		42	D	101010		58	e	111010
11	l	001011		27	v	011011		43	P	101011		59	r	111011
12	z	001100		28	W	011100		44	I	101100		60	space	111100
13	A	001101		29	F	011101		45	a	101101		61	h	111101
14	4	001110		30	1	011110		46	s	101110		62	j	111110
15	Y	001111		31	N	011111		47	2	101111		63	U	111111

Table (1) Cryptographic Algorithm

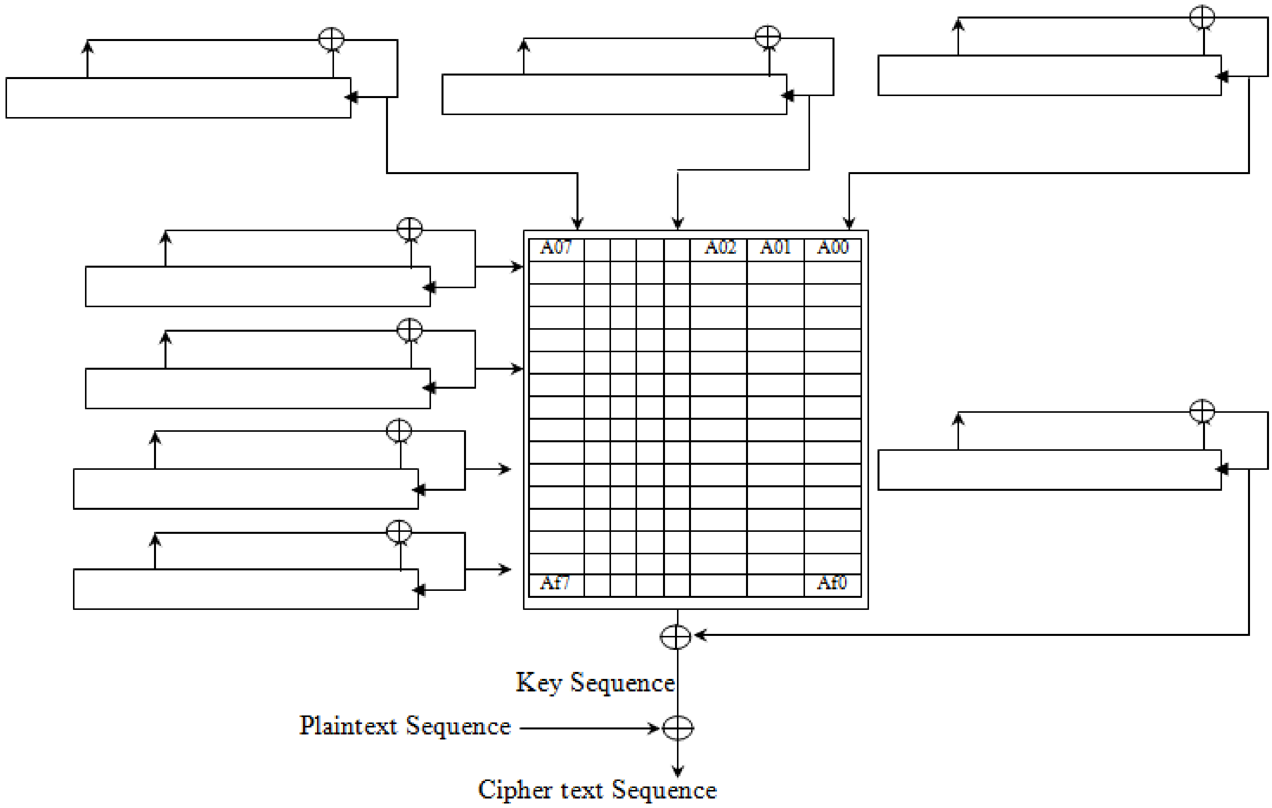

Figure (4) Cryptographic mechanism

7 - Hopping and Hiding

If the data embedding is done sequentially over the entire wave file (i.e. hiding in byte after byte), will make the stego system very vulnerable against simple tools of steganalysis. Therefore jump with variable (pseudo-random) length between successive hiding events will greatly increase the security level of stego system. In this research a pseudo random number generator was designed and implemented to generate non-uniform integer jumps between successive hiding events. The jumps mechanism of the suggested generator is based on the linear feedback shift register of length 23 with feedback function $F(x)=1+X+X^{23}$ produce the maximum period $2^{23}-1=8388607$.[9]

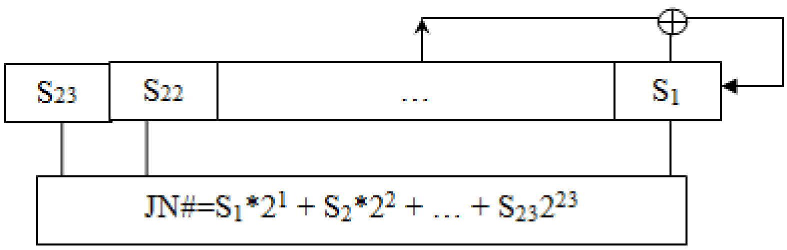

Figure (5) : The jumps mechanism

The Algorithms of the proposed system :-

Hiding poses

Step 1 : read plan text

Step 2: split play text into an ASCII

Step 3 : covered each ASCI info eight bits

Step 4 : read he decimal six, bits key

Step 5 : shift key char 2 bit

Step 6 : hopping the eight bits with R key G inside audio cover

Step 7 : hide the cipher text bit in the LSB of determined Audio byte

Step 8: if plan file finished

go to step 9 etc

go to step 6

extracting process

step 9 : hopping inside Audio file (cover)

step 10 : mask with LSB of determined audio byte and extract the cipher text bit

step 11 : extract hix decimal key of six bits code

step 12 : shift key char 2 bit left then mixed it with the original by using XOR

step 13: collect bit into ASCII code

step 14 : if cipher text finished

go to step 15 etc

go to step 9

step 15 : end

8 - The proposed system

The block diagram explains how to hide the message and how to extract it shown in figure (6)

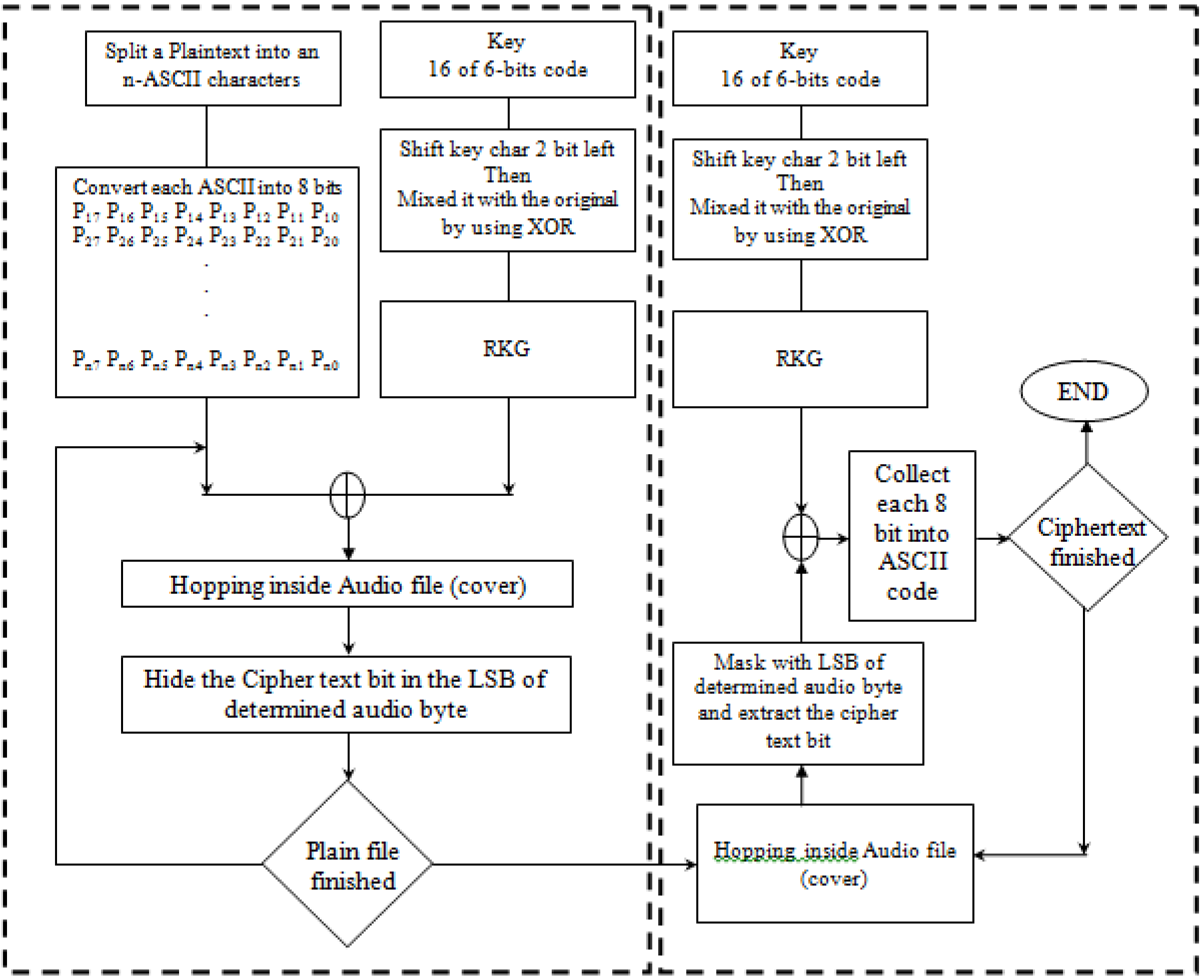

Figure (6) The block diagram of proposed system

9 - Fidelity Criteria

Signal-to-noise (SNR) measures are estimates of the quality of a quality reconstructed image compared with an original image. Reconstructed images with higher metrices are judged better quality. Traditional SNR measures do not equate.

First compute the mean squared error (MSE) of the reconstructed audio as follows:

$$MSE = \text{MSE} = \frac{\Sigma[f(I,j)-f^*(I,j)]^2}{N^2} \quad \ldots\ldots\ldots\ldots(1)$$

The summation is over all signals. The root mean square error (RMSE) is the square root of MSE.

PSNR in decibels (dB) is computed by using:

$$PSNR = 20\text{PSNR} = 20\log 10 \; \frac{(L-1)^2}{RMSE} \quad \ldots\ldots\ldots\ldots(2)$$

L: Max Value

$$20\text{SNR} = 20\log 10\left(\frac{\text{Max value}}{RMSE}\right) \quad \ldots\ldots\ldots(3)$$

Mean absolute error (MAE) can the measure the quality to the different of a reconstructed audio compared with one original audio, the value of this measure be between 1 and 0, the actual value be good if the value near from zero.

$$MAE = \text{Bit Per Sample} \quad BPS = \frac{\text{Hidden data Bit}}{\text{Total Cover size (sample)}} \ldots\ldots\ldots\ldots(4)$$

Typical PSNR values range will be greater than or equal to 50 .They are usually reported to two decimal points (e.g 25.47). The actual value is not meaningful, but the comparison between two values for different constructed audio signals gives one measure of quality. An informal threshold of 0.5 dB PSNR is used to decide whether to incorporate a coding optimization because they believed that an improvement of that magnitude would be visible.

The data hiding in wave data, and samples, the following table (2) illustrates the PSNR, MSE, and BPS results for eight types of data with different sizes are hidden in a “boop.wav” file whose size is (79561).

“boop” file whose size is (64162 byte).					
File	**Length(byte)**	**MSE**	**PSNR**	**BPS**	**SNR**
1	**1925**	**0.01**	**68**	**0.19**	**66**
2	**3208**	**0.02**	**65**	**0.32**	**64**
3	**6416**	**0.04**	**62**	**0.64**	**61**
4	**8341**	**0.05**	**61**	**0.83**	**59**
5	**10266**	**0.07**	**60**	**1.03**	**58**
6	**12832**	**0.08**	**59**	**1.29**	**57**
7	**17324**	**0.1**	**58**	**1.74**	**55**
8	**19249**	**0.11**	**58**	**1.93**	**55**

Table (2) : Output result of MSE,BPS, SNR and PSNR

The general structure of the proposed system is illustrated in figure (6) it consists of two basic modules: hiding and extraction modules . The input to this system are the cover file (wave file) and secret file (binary file) . These input are processed in the hiding part with various operations to produced stego wave file . The stego audio entered to extraction stage is processed through a set of operations to retrieve the secret data.

10- Conclusions:-

From the test results listed in propose system the following remarks wave derived

1- Hiding in voiced block sample is more suitable to avoid noise occurrence which is more probably happen when unvoiced blocks are used as host area.

2- Large threshold value provide more power in cover audio signal by avoiding unvoiced blocks and increased correct retrieved bits but decreased in hiding.

3- The results show acceptable hiding performance and the quality of reconstructed wave file is not subjectively different from the original wave.

11-References:

1- **Roue**, B. and Chassery, J. (2004) Improving LSB steg analysis using marginal and joint probabilistic distribution, *Proceeding of the 2004 workshop on Multimedia and security*. ACM New York, NY, USA, pp.75-80, 2004.

2- **Dumitrescu**, S., Wu, X., Wang, Z. (2003) Detection of LSB steganography via sample pair analysis,

IEEE Transactions on Signal Processing, Vol.51, No.7, pp.1995-2007, 2003.

3- **Gary** C. Kessler (2004) An Overview of Steganography for the Computer Forensics Examiner, *Forensic Science Communications*, Vol.6, No.3.

4- **Wang**, H. Wang**, S. (2004)** Cyber warfare: steganography vs. steganalysis. *Communications of the ACH* Vol 47 No. 10.

5- **Menezes**, A.J Oorschot, P.C., Vanstone,S.A. (2007) *Handbook of Applied Cryptography*, CPC Press, NY.

6- **Katzenbeisser**, S., Fabian A.P. (2000) Petitcolas, *Information Hiding Techniques for Steganography and Digital Watermarking*. Artech House Publishers. London.

7- **Lu** P., Luo, X.,Tang Q., Shen Li. (2004) An improved sample pairs method for detection of LSB embedding", *Proceedings of the 6th Information on Information Hiding Workshop*, Berlin ,pp 116-127.

8- **Johnson**, N. F., Duric, Z. and Jajodia, S. (2001) *Information Hiding: Steganography and Watermarking: Attacks and Countermeasures*. Kluwer Academic, Norwell, Massachusetts.

9- **Wayner**, P. (2009) *Disappearing Cryptography: Information Hiding: Steganography & Watermarking*. 3rd ed., Morgan Kaufmann, San Francisco, California.