Defence Science Journal, Vol. 56, No. 2, April 2006 pp. 259-268

Software-based Solution for Analysis and Decoding of FSK-2 Modulated, Baudot-coded Signals

Mukesh Malhotra,
Scientific Analysis Group, Delhi–110 054

and

Neena Gupta and R.S. Prasad
Punjab Engineering College, Chandigarh–160 012

ABSTRACT

In the present-day scenario, digital communication has become predominant choice over analog communication worldwide. Digital modulation schemes form the main block of the digital communication. Among these, frequency shift keying (FSK-2) is a widely used technique employed for Baudot-coded english text transmission. A Baudot-coded-FSK-2 modulated signal was simulated corresponding to an english text file. A technique has been developed for its analysis and decoding in the MATLAB environment. This technique does the signal analysis, its parameter extraction, and then digital demodulation, to retrieve its corresponding composite bit-stream. An efficient method for edge detection using the number of zero-crossings has been devised and implemented successfully. From the composite bit-stream, overhead bits were removed and decoding was performed to get back the text output.

Keywords: Frequency shift keying, fast Fourier transform , autocorrelation, sliding window, hopping window, FSK, FFT

1. INTRODUCTION

The digital communication is the transmission of information from one point to the other in digital form[1-4]. The original information can be either analog or digital. The analog information signal (eg, speech) is first converted into digital form, then processed suitably to enable it to travel on a communication link (using a digital modulation scheme, eg ASK, FSK, PSK, QAM, etc). Among the various digital modulation schemes, FSK-2 is found to have very good error-performance, while considering channel noise and synchronisation aspects[1-4]. Baudot code is an asynchronous code, suited for low-speed data transmission[5]. It has now been standardised and is officially known as ITA-2 (CCITT-2) code. With 5 bits, only 2^5 combinations are possible, which are inadequate for transmitting 26 alphabets, 10 numerals, and certain special characters (eg, space, brackets, +, ?, etc) used in transmission of english text. To achieve this, two of these 32 combinations, called letter shift (11111) and figure shift (11011), are used to select alternate character sets. Each character code is preceded by a START bit and followed by a STOP bit. START bit is of 1-bit duration, whereas

Revised 01 December 2005

STOP bit can be of 1/1.5/2-bit duration. A Look-Up Table for the Baudot code (ITA-2) is given in *Appendix* 1.

The fundamental parts of a digital transmission system are the transmitter, the medium over which information is transmitted and the receiver, which estimates the original information from the received signal[1]. The transmitter contains encoder and modulator while the receiver has demodulator and decoder as its subparts.

2. WORK OBJECTIVE

Frequency shift keying (FSK-2) is the popular digital modulation technique for sending Baudot coded, english text messages. These signals are of prime interest in the area of communication signal analysis, as many agencies, eg post and telegraph department, use machines based on such a technique for their communication. This study aims at simulation, frame length analysis, and decoding of FSK-2 modulated, Baudot-coded signals to develop a software-based solution. Performance analysis of the developed tool was also done. For this work, input was an english text file, made in notepad. Baudot code was taken as coding scheme for the text and FSK-2 as digital modulation scheme. The communication was considered to be of asynchronous type. The application programs to accomplish the work undertaken, were developed on MATLAB (version 5.2) platform[6].

3. TECHNIQUE DEVELOPMENT

The technique development work for achieving the work objective was completed in the following four phases:

3.1 Phase I: Simulation

During this phase, a Baudot-coded, FSK-2 modulated signal was generated by a software program whose input was an english text file made in a notepad; and the user-defined parameters, viz. required baud rate, mark-space frequencies, polarity, etc. The program takes one input character at a time, performs its Baudot encoding, appends START bit in the beginning and STOP bit at the end, stores it in a file, and then takes next character, and so on. These bits were used to generate an FSK-2 modulated signal. For the present work, the frequencies taken for generating an FSK-2 signal were 1200 Hz and 1800 Hz.

3.2 Phase II: Signal Analysis/Digital Demodulation

The simulated signal was analysed and its parameters, which were required for digital demodulation, and further decoding, were extracted. This was accomplished in the following steps:

Step 1. *Spectral & spectrographic analysis*

The frequency spectrum of the signal shows two distinct peaks confirming the signal to be of FSK-2 type (Fig. 1). Also, its spectrogram (Fig. 2) reveals the presence of two discrete frequencies which were switching among themselves with time.

Step 2. *Identification of mark/space frequencies*

In this step, the two frequencies present in the signal (called Mark and Space) were found out, by performing 2048 points FFT of the signal[7,8] (Fig. 2). As a result, two MAXIMAS (peaks) in the signal were detected and their corresponding frequencies were extracted.

Step 3. *Edge detection & estimation of baud rate*

After detecting the mark and space frequencies, the next step was to find out at what instant, the transition from one frequency to another was taking place in the signal. These transitions are called edges. For achieving this , the developed technique makes use of a sliding window and a parameter 'number of zero-crossings (# zero-crossings)'. One complete sinusoidal cycle has two zero-crossings, ie # zero-crossings per cycle = 2. For a fixed-window size, the number of zero-crossings for a particular frequency will be a fixed value, eg consider a window size of 50 samples, at sampling frequency of 9000 Hz. Hence, a 1200 Hz sinusoidal wave will have # zero-crossings = (50/9000) x 1200 x 2 = 120/9 = 13 (approx).

Similarly, a 1800 Hz sinusoidal wave will have # zero-crossings = (50/9000) x 1800 x 2 = 20.

A sliding window is the window that is slided over the data and # zero-crossings are computed

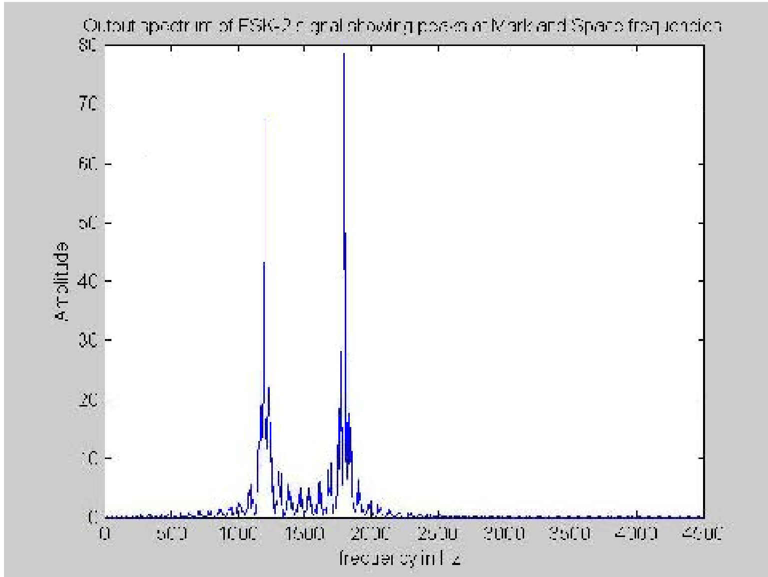

Figure 1. Spectrum of FSK-2 signal showing two discrete frequencies

for that window. In the present work, the sliding window length was taken to be 50 samples, with a slide length of 10. At the edge, there would be a change in # zero-crossings. This change was picked up and taken as a position of the edge. The change in # zero-crossings was visible in the transition phase (where the sliding window contains the edge), before settling to a final value, viz., when the edge is over, current window contains data corresponding to a single frequency. With information about all the edges detected in the data, the difference between the two consecutive edges (in terms of number of samples) was determined (edge_difference). Taking the edge_difference and its frequency of occurrence, a histogram (Fig. 3) was plotted, from which the baud rate was computed. The most frequently found edge_difference was taken and its reciprocal was multiplied by sampling frequency to get the estimated baud rate. [If the most frequently found edge_difference is

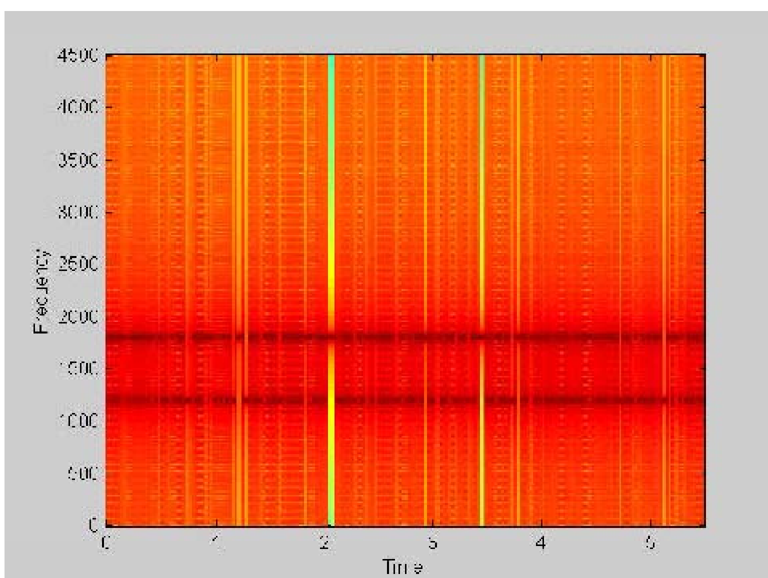

Figure 2. Spectrogram of the simulated signal

divided by the sampling frequency used, it will give bit duration in seconds]

Step 4. *Bit-extraction*

Once the number of samples/bit was known, the developed algorithm makes use of a hopping window, for extracting bit-stream corresponding to the signal. The window, of size equal to the number of samples/bit, was hopped throughout the data and bits were extracted by taking decision based on the FFT of the windowed data[7,8]. This exercise was done with both the polarities. If FFT results indicate the presence of a frequency of 1200 Hz, a '1' was written in the output file , else a '0'. Similarly, for the other polarity, if FFT results indicate a frequency of 1800 Hz, a '1' was written in the output file, else a '0'. In addition, the developed technique does the STOP-bit classification (detecting whether STOP bit is of 1/1.5/2-bit duration) and accordingly puts back bits in an output file. For a 1/1.5 bit duration STOP bit, it puts back a single bit in the output file, whereas for a 2-bit duration STOP bit, it puts back two identical bits.

3.3 Phase III: Frame-length Analysis

In this phase, frame-length analysis was performed on the extracted bit-stream, which involves performing autocorrelation for detecting the presence of periodicity in the signal, if any[4,7]. It provides a measure of similarity between a signal and its time-delayed version.

Mathematically, autocorrelation function of an energy signal $g(t)$ is given by:

$$R_g(\tau) = \int_{-\infty}^{\infty} g(t)g(t-\tau)\,dt \qquad (1)$$

The parameter τ is called lag. If the signal $g_p(t)$ is periodic, with period T_0, its autocorrelation function $Rg_p(\tau)$, is given by

$$R_{g_p}(\tau) = \int_{-T_0/2}^{T_0/2} g_p(t)g_p(t-\tau)\,dt \qquad (2)$$

An important property of $Rg_p(\tau)$ is that it exhibits periodicity with the same period as the periodic signal $g_p(t)$ itself. If the signal is discrete, ie in the form of samples, stored in a vector $u(n)$ with M samples, then its autocorrelation function $R(n)$ is given by

$$R(n) = \sum_{k=1}^{M} u(k)u(k+n) \qquad (3)$$

where n is lag.

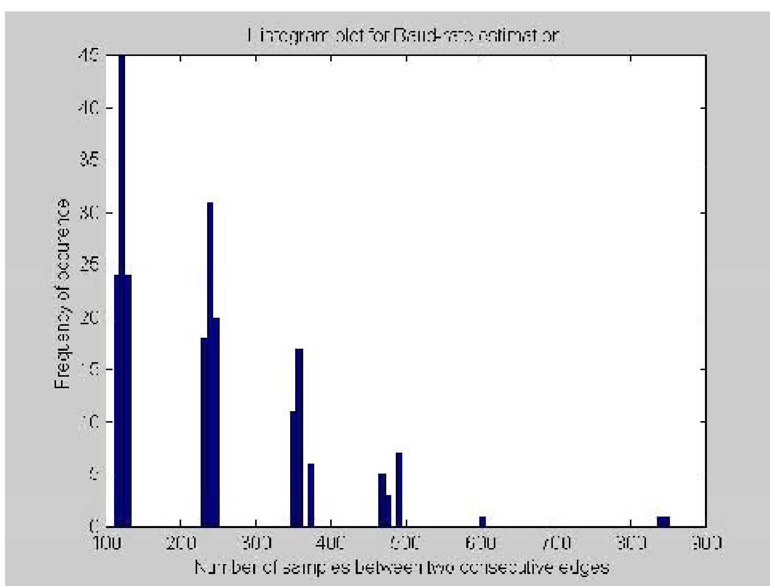

Figure 3. Histogram plot for baud-rate estimation

Biased estimate of autocorrelation function is given by

$$R_{biased}(n) = \mathrm{R}(n)/M \tag{4}$$

Unbiased estimate of autocorrelation function is given by

$$R_{unbiased}(n) = R(n)/(M - |n|) \tag{5}$$

Also $R(0) = 1$, normalises the estimate of autocorrelation function, so that element at zero lag is identically 1.

The autocorrelation (unbiased) was performed on the extracted bit-stream to look for any periodicity in it. The software program written for this purpose computes and plots autocorrelation function of the output bit-sequence, at various lags. This plot can be zoomed for a more clear view of the periodicity. The periodicity in the bit stream is due to the fact, that by convention, START bit is always encoded as a '0' and STOP bit is always encoded as a '1'. Since, it is an asynchronous type of communication, STOP-START bit combination will be seen in the data {ie presence of [1 0] pattern in case the STOP bit is of 1/1.5-bit duration, and [1 1 0] pattern in case the STOP bit is of 2-bit duration}. Hence, the algorithm developed for this purpose, gives a frame size of 7 bits in case the STOP bit is of 1/1.5-bit duration, and frame size of 8 bits, in case of STOP bit of 2-bit duration.

3.4 Phase IV: Synchronisation & Decoding (to Retrieve Text)

After the data is found to have periodicity, first removal of overhead bits was done (START/STOP bits in the composite bit-stream). This was accomplished after synchronising with the first STOP-START bit pair and then removing all STOP-START bits in the data, leaving only information bits. These information bits were later subjected to Baudot decoding, to retrieve text.

For synchronisation with the first STOP-START pair, in case of STOP bit of 1/1.5 bit-duration, the developed algorithm first considers a row vector $t1$ having 10 entries:

$$t1 = [1\ 0\ 1\ 0\ 1\ 0\ 1\ 0\ 1\ 0]$$

A row vector c1 (size = 10) now is formed from a vector *output* (contains extracted bit-stream, obtained as a result of bit extraction) taking its first two entries as it is, then leaving next 5 entries, again taking next two entries, leaving next 5 entries, and so on.

In the next step, XOR of $t1$ and $c1$ is done and the result is stored in vector *xor_result* (of the same size, viz, 10). If the *xor_result* is [0 0 0 0 0 0 0 0 0 0], ie, an all-zero vector, then the $t1$ matches with $c1$. Hence, the algorithm finds that data is synchronised with the first STOP-START pair at SHIFT=0. However, if this is not the case, then vector *output* is slided by a SHIFT=1. Again $c1$ is formed in the same way and its XOR result with $t1$ is seen. This exercise is repeated with SHIFT = [1, 2,….(frame_size–1)] till one gets the *xor_result* as an all-zero vector. The corresponding SHIFT is displayed and the first SHIFT entries in the vector output are deleted. Hence, a new vector *output_new* is formed, having the values same as vector output, starting from (SHIFT + 1) till end of the vector. This vector has all $(7N + 1)$ entries as 1 and $(7N + 2)$ entries as 0, where $N = 0,1,2,\ldots\ldots$ The synchronisation was thus achieved covering five characters which was accurate. In the next step, these repeating [1 0] entries, viz Overhead bits, were removed from the composite bit stream and resulting bit stream, comprising only information bits, was stored in a file.

In case of STOP bit of 2-bit duration, row vector $t1$ (having 15 entries) is taken as

$$t1 = [1\ 1\ 0\ 1\ 1\ 0\ 1\ 1\ 0\ 1\ 1\ 0\ 1\ 1\ 0]$$

A row vector $c1$ (size = 15) is now formed from the vector *output* taking its first three entries as it is, then leaving next 5 entries, again taking next three entries, leaving next 5 entries, and so on. Algorithm will now look for periodic [1 1 0] pattern, rest of the procedure remains the same.

Once, the information bits are available, in the next step, Baudot decoding was performed.

Software program written for Baudot decoding makes use of the Look-Up Table for Baudot. The program takes 5 bits at a time and performs decoding. Special care was taken, while decoding bit combinations, indicating letter-shift and figure-shift. Default mode taken, while decoding, is the letters mode. Once a bit combination for FIGSHIFT (11011) comes, it remains in FIGSHIFT till it gets a LTRSHIFT combination (11111). Also, it displays proper error messages, as and when required. For instance, bit combination for carriage return (CR) is always followed by bit combination of line feed (LF). If while decoding, it is not the case, an error message is displayed on the screen.

The decoded text, ie, the text retrieved after Baudot decoding, was found to be matching perfectly with the input text (which was originally subjected to encoding).

4. OBSERVATIONS & RESULTS

Corresponding to many input text files (referred to in sequel as test cases), Baudot-coded FSK-2 modulated signals were simulated. These test cases were selected so as to cover the range of baud rate worked upon (10-75), covering both types of polarity, letters (alphabets), numerals (figures) as well as special characters appearing in english text transmission. The idea was to have sufficient toggling between figures and letters so that LTRSHIFT and FIGSHIFT codes should also appear, and hence, tested thereafter. The developed technique was tried on about 150 text messages and decoded text output were compared with corresponding input text file (which was subjected to Baudot encoding followed by FSK-2 digital modulation). In more than 95 per cent of the test cases, the retrieved text was found to be matching with the input text file. The analysed results for one of the test cases have been presented here.

Test Case

For this test case, the input text message taken was:

r0123456789
!#$() abcd
++++ dollar rupee pound

Corresponding to this message, a Baudot- coded, FSK-2 modulated signal was simulated with the following parameters (using sampling frequency 9000 Hz):

Polarity	: Reverse
Baud rate	: 75
STOP-bit duration	: 2

The simulated signal can be listened by *sound* command of MATLAB. Figure 2 shows spectrogram of the simulated signal.

This signal was analysed and its parameters were extracted, followed by digital demodulation to retrieve bit stream, and decoding to retrieve text.

The 2048 points fast Fourier transform (FFT) of this signal was performed to get the two discrete frequencies in the signal as 1200 Hz and 1800 Hz (plot shown in Fig. 1).

Using the extracted frequencies (ie, 1200 Hz and 1800 Hz), and sliding a window of size 50 samples with a slide length of 10 over the signal, the positions of all the edges in the signal were found and a histogram (Fig. 3) was plotted with the edge_difference on X-axis and their frequency of occurrence on Y-axis, for the estimation of the signal's baud rate.

From the histogram plot, the most frequently occurring edge_difference was found to be 121, and hence, estimated baud rate = (1/121)*9000 = 74.38.

In the next step, using a hopping window of size 121, bits were extracted by taking a decision based on the FFT of the windowed data. The bit stream obtained is shown in Fig. 4.

On this bit stream, autocorrelation was performed which gave a frame size of 8. The autocorrelation plot and its zoomed version are shown in Figs 5 and 6, respectively.

Next, the algorithm synchronised itself with first STOP-START bit combination and found out the first [1 1 0] pattern (which is periodic) at SHIFT = 4. In the next step, first SHIFT number

1	0	1	0	1	1	0	1	1	0	1	1	1	1	0
0	1	1	0	1	1	1	0	1	1	1	0	1	1	1
0	1	1	0	0	1	1	1	0	1	0	0	0	0	1
1	0	0	1	0	1	0	1	1	0	0	0	0	0	1
1	1	0	1	0	1	0	1	1	1	0	1	1	1	0
0	1	1	0	0	1	1	0	0	1	1	0	0	0	0
1	1	1	1	0	0	0	0	1	0	1	1	0	0	1
0	0	0	1	1	0	1	0	1	1	0	1	1	0	0
0	1	0	1	1	1	0	1	0	0	1	0	1	1	0
1	1	1	1	0	1	1	0	0	1	0	0	1	1	1
0	1	1	1	1	1	1	1	0	0	0	1	0	0	1
1	0	1	1	0	0	0	1	1	0	1	0	0	1	1
1	1	0	0	1	1	1	0	1	1	0	1	0	0	1
0	1	1	0	0	0	0	1	0	1	1	0	0	1	0
0	0	1	1	0	1	1	0	1	1	1	1	0	1	0
0	0	1	1	1	0	1	0	0	0	1	1	1	0	1
0	0	0	1	1	1	0	1	0	0	0	1	1	1	0
1	1	1	1	1	1	1	0	0	0	1	0	0	1	1
0	1	0	0	1	0	1	1	0	0	0	0	1	1	1
1	0	0	1	0	0	1	1	1	0	0	1	0	0	1
1	1	0	1	1	0	0	0	1	1	0	0	1	0	1
0	1	1	0	0	0	1	0	0	1	1	0	0	1	0
1	0	1	1	0	1	1	1	0	0	1	1	0	0	1
1	0	1	1	1	0	1	0	0	0	0	1	1	0	1
0	0	0	0	1	1	0	0	0	1	0	0	1	1	0
0	1	1	0	1	1	1	0	0	0	0	1	1	1	1
0	1	1	1	0	0	1	1	0	0	0	1	1	0	1
1	0	1	0	0	1	0								

Figure 4. Bit-stream obtained using a hopping window of size 121

of bits was deleted from the composite bit-stream and next, all overhead bits (all STOP-START bits) were deleted, followed by retrieval of text by applying Baudot decoding on leftover information bits. Following is the decoded text obtained:

0123456789

!#$() ABCD

++++ DOLLAR RUPEE POUN

[The algorithm starts detecting edges from the first edge found and stops at the last edge detected (not beginning and end of signal), and hence, first and last character in the input text may not appear in the decoded text].

5. CONCLUSION

This study is an effort towards analysis and decoding of Baudot-coded, FSK-2-modulated signals. The study involved simulation of a Baudot-coded, FSK-2 digitally modulated signal corresponding to an english text file, depending on the user-defined parameters, eg, baud rate, polarity, STOP-bit duration, etc. A technique has been developed for analysis and decoding of such a signal which does the signal analysis, its parameter extraction, and then digital demodulation to retrieve its corresponding composite bit-stream. Once the nature of signal is confirmed to be FSK-2 type, the two frequencies present in the signal are found by taking 2048 point FFT of the signal. This information is used in the edge-detection algorithm, based upon the parameter 'number of zero-crossings'. With the edge_difference and its frequency of occurrence, a histogram was plotted and the baud rate of the signal was estimated straightaway. The estimated baud rate makes way for extracting composite bit-stream which was achieved using a hopping window FFT. The autocorrelation was performed on the extracted bit-stream to look for any periodicity in it. A novel approach for synchronisation for the identification of START/STOP bits has been evolved and implemented. From the composite bit-stream, overhead bits were removed and decoding was performed to get back the text output. The retrieved decoded text was found to be matching with input text file in more than 95 per cent test cases.

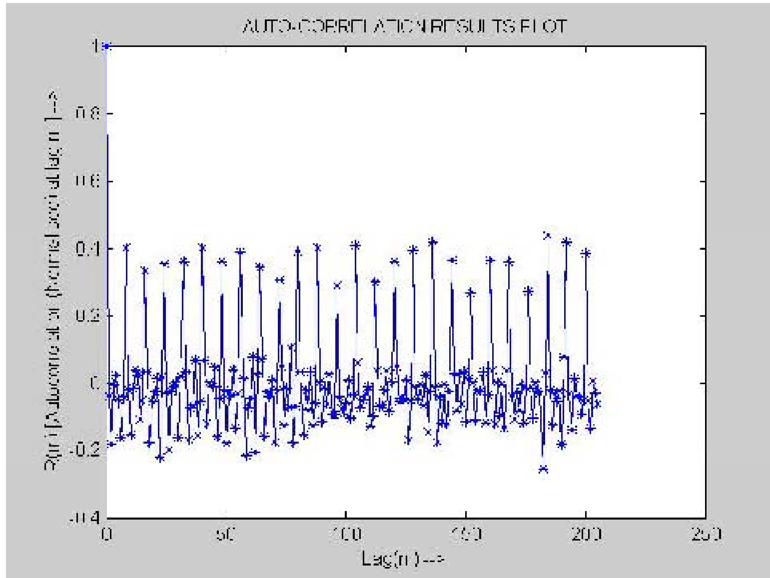

Figure 5. Autocorrelation plot at various lags

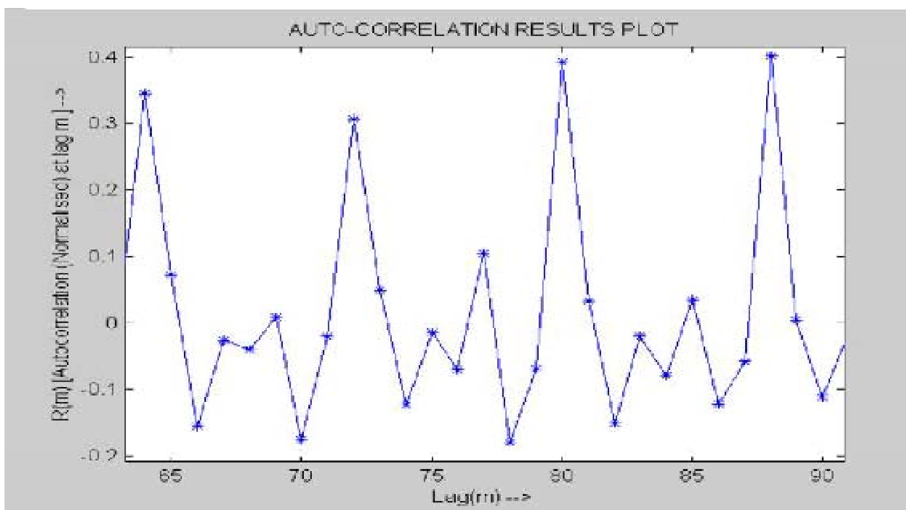

Figure 6. Autocorrelation plot (zoomed)

The developed technique can be regarded as an efficient, easy to use analysis tool for the FSK-2 modulated signals. It gives accurate and satisfactory results for baud rates up to 75. For the baud rates ranging between 75-90, the performance starts degrading in terms of accuracy of the retrieved decoded text. Beyond baud rate 90, there is a need for the change of parameters (sliding window length, slide length, etc) and also their optimisation. The need arises due to an increased baud rate which causes edges to appear very close. Hence, selecting the size of the sliding window will be crucial, so that two or more bits shouldn't appear in the selected window size. Rest of the procedure of decoding will remain the same, while the analysis and decoding of such higher baud-rate signals can be taken up as an extension of the present study.

ACKNOWLEDGEMENTS

The authors express their sincere gratitude to Dr P.K. Saxena, Director, Scientific Analysis Group (SAG), Delhi, Prof C.E. Veni Madhavan, former Director, SAG, and Shri Baljeet S. Kapoor, Principal, Punjab Engineering College, Chandigarh for their guidance and support. The authors also acknowledge support given by Shri H.P. Agarwal, Scientist E, Defence Electronics Application Laboratory (DEAL), Dehradun, in the area of digital communications.

REFERENCES

1. Smith, David R. Digital transmission systems, Ed. 2. Van Nostrand Reinhold, New York, USA.
2. Alisouskas, Vincent F. & Tomasi, Wayne. Digital and data communications. Prentice Hall Inc, Englewood Cliffs, New Jersey 07632, USA.
3. Haykin, Simon. An introduction to analog and digital communications. John Wiley & Sons (Asia) Pvt Ltd.
4. Lathi, B.P. Modern digital and analog communication systems. Oxford University Press.
5. Kennedy, George. Electronic communication systems, Ed. 3. McGraw Hill Publishing Co.
6. MATLAB reference manual. The Math Works Inc, MA 01760, USA.
7. Oppenheim, Alan V. & Schafer, Ronald W. Digital signal processing. Prentice Hall Inc, Englewood Cliffs, New Jersey 07632, USA.
8. Ramirez, Robert W. The FFT : Fundamentals and concepts. Prentice Hall Inc, Englewood Cliffs, New Jersey 07632, USA.

Contributors

Mr Mukesh Malhotra obtained his BTech (Electronics) from the Guru Nanak Dev University, Amritsar, in 1988 and ME (Electronics) from the Punjab Engineering College, Panjab University, Chandigarh in 2003. He joined DRDO as Scientist in 1991. His area of research is digital communications.

Dr R.S. Prasad did his BSc Engineering (Telecommunication) from the Birla Institute of Technology, Sindri, MSc Engineering (Electronics) and PhD from the Panjab University while serving in the Punjab Engineering College, Chandigarh. He was on the faculty of BIT, Sindri, from 1966 to 1968 and from 1968 to 1974, he served in the Central Intelligence Bureau, MHA, Govt of India, New Delhi. He joined as Lecturer in Punjab Engineering College in 1974 and subsequently served as Assistant Professor and Professor. He has published many research papers in national/international journals and presented research papers in seminars.

Dr Neena Gupta obtained her BE (Electronics and Communication Engg) in 1988 and ME (Electronics) from the Panjab University in1992. She is Assistant Professor, Department of Electronics and Electrical Communication Engineering, Punjab Engineering College, Chandigarh. She did her PhD from Panjab University in 2002 in the field of Optical Communications. She has published many research papers in national/ international journals and presented research papers in conferences. Her area of research is digital and optical communications.

APPENDIX 1

Baudot Code Look-Up Table

No.	I TA-2 Alphabet	Letter	Figure
1	1 1 0 0 0	A	-
2	1 0 0 1 1	B	?
3	0 1 1 1 0	C	:
4	1 0 0 1 0	D	
5	1 0 0 0 0	E	3
6	1 0 1 1 0	F	
7	0 1 0 1 1	G	
8	0 0 1 0 1	H	
9	0 1 1 0 0	I	8
10	1 1 0 1 0	J	BELL
11	1 1 1 1 0	K	(
12	0 1 0 0 1	L	)
13	0 0 1 1 1	M	.
14	0 0 1 1 0	N	,
15	0 0 0 1 1	O	9
16	0 1 1 0 1	P	0
17	1 1 1 0 1	Q	1
18	0 1 0 1 0	R	4
19	1 0 1 0 0	S	’
20	0 0 0 0 1	T	5
21	1 1 1 0 0	U	7
22	0 1 1 1 1	V	=
23	1 1 0 0 1	W	2
24	1 0 1 1 1	X	/
25	1 0 1 0 1	Y	6
26	1 0 0 0 1	Z	+
27	0 0 0 1 0	Carriage Return	
28	0 1 0 0 0	Line Feed	
29	1 1 1 1 1	Letter Shift	
30	1 1 0 1 1	Figure Shift	
31	0 0 1 0 0	Space	
32	0 0 0 0 0	Unperforated Tape	