

A GSM Simulation Platform using MATLAB

Mr. Suryakanth.B* , Mr. Shivarudraiah.B* , Mr. Sree Harsha H.N**

*Asst Prof, Dept of ECE, BMSIT Bangalore, India

**Asst Prof, Dept of EEE, CMR Institute of Technology, Bangalore, India
suryakanthbm@gmail.com, Shiv2byn@gmail.com, harsha.hns@gmail.com

Abstract— the objective is to originate a software platform capable of producing a series of appropriate GSM data blocks and subsequently perform correct reception of these. GSMsim provides a mean for assessing the performance of both transmitter and receiver front-ends in a GSM system. The performance analysis is based on a BER measure calculated by comparing a random input bit sequence and the resulting sequence estimated by the receiver.

Keywords- BER, ISI, S/N, GMSK

I. INTRODUCTION

To analyze specific front end architectures and designs for GSM operation one just have to insert a software description of the receiver or transmitter prior to running the demodulator. The front end description must, of course, comply with some predefined interface restrictions. The tool box described consists of base band parts only.

II. TRANSMITTER SECTION

A. Data Generation

The origination of data and the tasks of interleaving the data, performing the channel encoding, and multiplexing the resulting data segments are implemented in three separate blocks. These blocks then make sure that a correct GSM normal burst bit format structure is generated. This is done by first generating a series of random bits which, in turn, are inserted in a prescribed frame structure a single GSM.

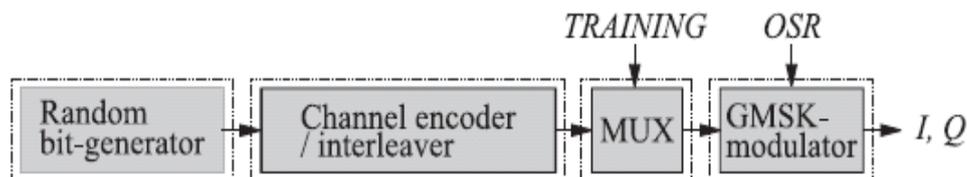


Fig: 1 Transmitter Block diagram

B. Channel Encoding

The objective of the channel encoder is to give the GSM receiver with the ability to find transmission errors and eventually correct some of these. This will improve the transmission quality from a bit error point of view. The encoding implemented here goes for the burst type TCH/FS (*Traffic CHannel Full rate Speech*) which is a normal speech burst. Divides the incoming 260 information bits into three different classes, i.e. class Ia, class Ib, class II, depending on the importance of the bits. For instance, any transmission errors in the class Ia bits effect the overall speech quality more severely than errors in class II bits. Due to this variation in bit importance the different classes of bits are encoded accordingly. The channel encoding scheme utilized in GSM is illustrated in Figure 2

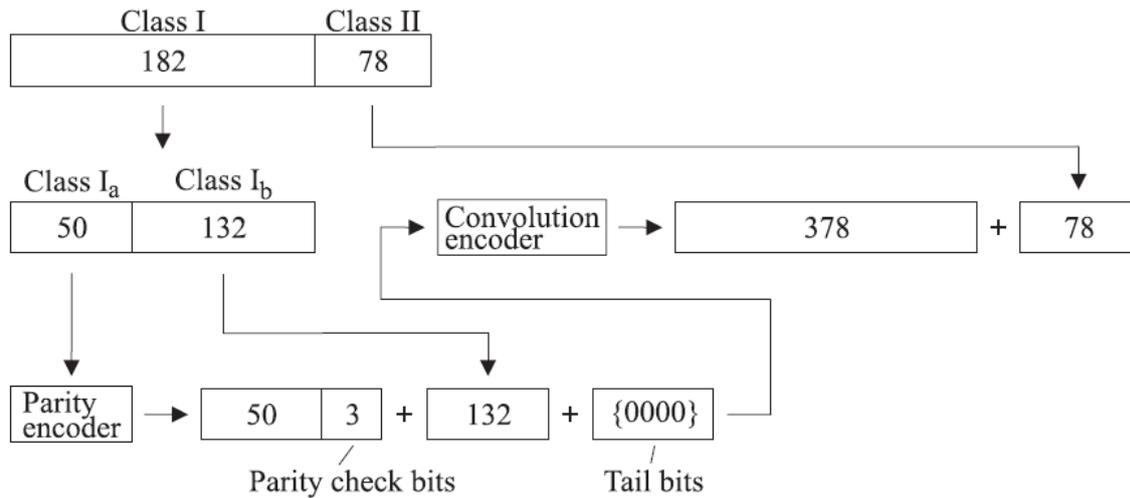


Fig 2: Channel encoding in GSM. A total of 196 redundant bits are added

The channel encoding scheme is thus as follows. The 50 most significant class I bits, the class Ia bits, are extracted from the sequence and parity encoded. The parity encoder used in GSM is a systematic cyclic encoder based on three check bits. Systematic means that the parity bits are added to the original class Ia bit sequence. This way the class Ia bits are left unchanged and the resulting code word has the structure.

$$\bar{V} = \{[k \text{ data bits}][r \text{ check bits}]\}$$

The generator polynomial used in the encoder has a length of 4 bits.

$$G(x) = x^3 + x + 1 \rightarrow \bar{G} = \{1011\}$$

The check bits are found as the remainder, $r(x)$, of the division.

$$\frac{x^r \cdot D(x)}{G(x)} = Q(x) + \frac{r(x)}{G(x)},$$

where the number of check bits is given by r , $D(x)$ represents the data bits intended for encoding and $Q(x)$ the division quotient. The remainder, $r(x)$, is then directly used to form the check bit sequence required in generating \bar{V} . After parity encoding of the class Ia bits these are recombined with the class Ib bits and a tail sequence of four zeros is finally added. The resulting class I sequence, now consisting of 189 bits, is then feed to the convolution encoder.

The convolution encoder takes a block of k bits as input and returns a block of n bits as output. The rate of the encoder, defined as the ratio k/n , is in the GSM system specified to be $1/2$. In the convolution encoding scheme each output bit, c_n , is depending not only on the input bit presently being encoded, b_k , but also on some of the previous input bits. The number of input bits required in the processing of encoded output bit is called the constraint length of the encoder. GSM specifies a constraint length of 5 in its encoding scheme defined as:

$$c_{2k} = b_k \oplus b_{k-3} \oplus b_{k-4}$$

$$c_{2k+1} = b_k \oplus b_{k-1} \oplus b_{k-3} \oplus b_{k-4},$$

Where \oplus implies modulo 2 addition, and

$$k \in \{0, 1, 2, \dots, 189\} \text{ and } b_k = 0 \text{ for } -\infty \leq k < 0$$

As the convolution encoder is defined as a rate $\frac{1}{2}$ encoder two output bits are generated for every input bit, hence the two expressions. When operated as a shift register the convolution encoder takes on the form illustrated in Figure 3.

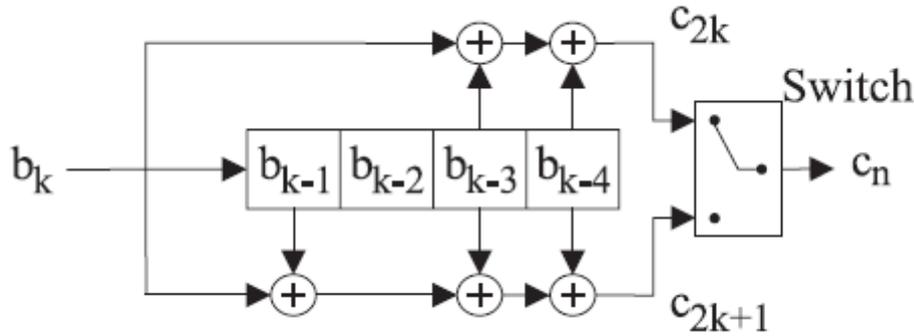


Fig 3: The convolution encoder scheme used in GSM for encoding of TCH/FS bursts. All additions are modulo 2 additions.

C. Interleaving

The interleaver shuffles the bits contained in the data blocks output from the channel encoder, and distributes them over a number of bursts. The purpose of this procedure is to ensure that the errors that appear in a received data block are uncorrelated. The motivation for reducing the correlation between bit errors is that the convolution code used to protect the class I bits has better performance when errors are not correlated. The interleaver operates according to the following two formulae:

$$b = ((57 \cdot (T \bmod 4) + t - 32 + 196 \cdot (t \bmod 2)) \bmod 456)$$

$$B = ((T - (b \bmod 8)) \text{div } 4),$$

Fig 4: Illustration of the interleaving process.

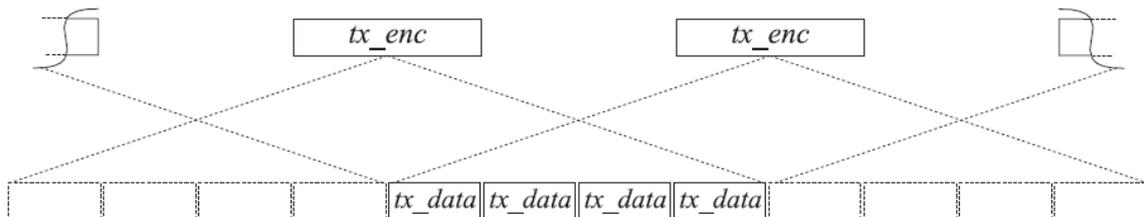
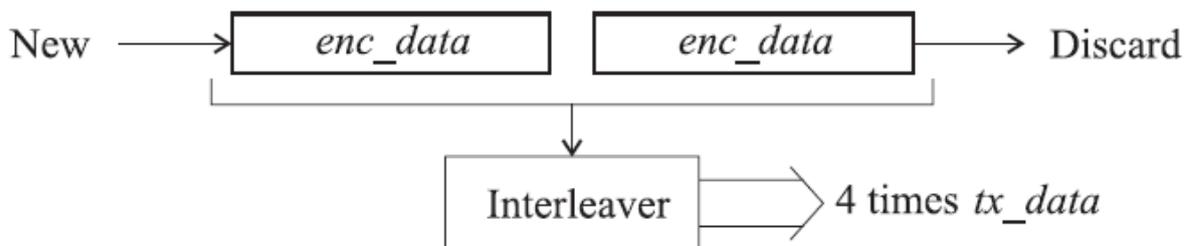


Fig 5 Operation of the interleave, aided by a queue



The interleaver reads out the entire content of the queue. The queue has two slots, and in each interleaving pass a new block is pushed into the queue, and the oldest block is discarded.

D. Multiplexing

The GSM recommendations dictate specific burst structures for different transmission purposes. The implemented burst, referred to as a GSM normal burst, has the structure displayed in Figure 6:

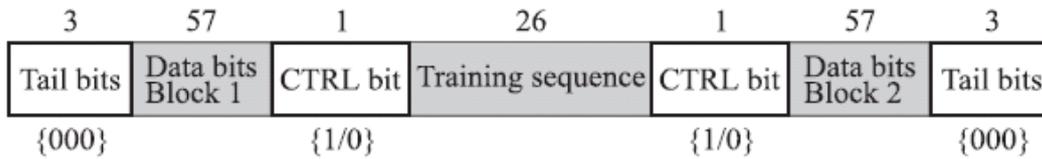


Fig 6 the prescribed GSM normal burst structure.

The number of bits in each part of the burst is indicated by the integers above the individual parts. The GSM normal burst is made up of $2 \cdot (3+57+1)+26+148$ bits in total. Of these 148 bits $2 \cdot 57 + 114$ are pure data bits. Of the 114 data bits input $114 \cdot (260/456) =$ bits are true information bits, as can be seen. The training bit sequence can be any of eight prescribed ones.

E. GMSK-Modulation

These three functions implement two separate tasks, as both a differential encoding of the burst sequence as well as the actual GMSK modulation is performed. These two operations are described further in the following sections.

F. Differential Encoding

GSM makes use of the following combined differential encoding and level shifting scheme, where $d \in \{0, 1\}$ and $a \in \{-1, 1\}$ represent input and output sequences, respectively

$$\begin{aligned} \hat{d}[n] &= d[n] \oplus d[n-1] \\ a[n] &= 1 - 2 \cdot \hat{d}[n], \end{aligned}$$

When calculating a [0] and thereby also d [0] it may be assumed that d [-1] is one.

Modulation

After the differential encoding of the information burst the signal is GMSK modulated. GMSK is a modulation form derived from the very similar MSK (*Minimum Shift Keying*). Both are variants of the more general CPFSK (*Continuous Phase Frequency Shift Keying*) modulation forms. Mathematically the generation of a MSK signal may be described as

$$s(t, \bar{a}) = \sqrt{\frac{2E_c}{T_b}} \cos \{2\pi f_c t + \Theta(t, \bar{a})\},$$

where E_c is the bit energy, f_c the carrier frequency, and $\Theta(t, \bar{a})$ the information carrying phase of the MSK signal.

$$\begin{aligned} \bar{s}(t, \bar{a}) &= s_c(t, \bar{a}) + j \cdot s_s(t, \bar{a}) = e^{j\Theta(t, \bar{a})} \\ &= \cos \{\Theta(t, \bar{a})\} + j \cdot \sin \{\Theta(t, \bar{a})\}, \end{aligned}$$

represents the complex envelope of the modulated signal. Making use of the following pulse shaping function $p(t)$

$$p(t) = \begin{cases} \cos\left(\frac{\pi t}{2T_b}\right) & -T_b \leq t \leq T_b \\ 0 & \text{otherwise,} \end{cases}$$

the in-phase and quadrature phase components $s_c(t, \bar{a})$ and $s_s(t, \bar{a})$ may be rewritten using the following linear form

$$\begin{aligned} s_c(t, \bar{a}) &= p(t) * a_c(t) = \sum_{n \text{ even}} a_c[n] \cdot p(t - nT_b) \\ s_s(t, \bar{a}) &= p(t) * a_s(t) = \sum_{n \text{ odd}} a_s[n] \cdot p(t - nT_b), \end{aligned}$$

where the weighted impulse responses, a_c and a_s

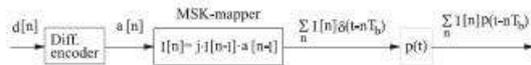
t, are given as

$$a_c(t) = \sum_{n \text{ even}} a_c[n] \delta(t - nT_b); a_c[n] = \cos(\Theta[n])$$

$$a_s(t) = \sum_{n \text{ odd}} a_s[n] \delta(t - nT_b); a_s[n] = \sin(\Theta[n]),$$

$\Theta[n]$ is given as

$$\Theta[n] = \frac{\pi}{2} \sum_{k=0}^{n-1} a[k]$$



Final QAM-model for MSK including the differential encoding prescribed in GSM.

III. RECEIVER SECTION

Synchronization

Synchronization, channel estimation, and matched filtering are done in a two step process. For the matched filter to operate correctly the synchronization and channel estimation must be done first.

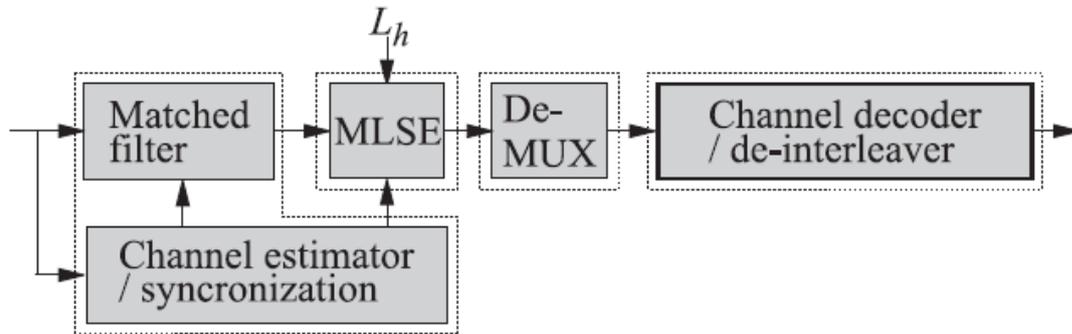


Fig 7: Synchronization, channel estimation, and matched filtering are divided into two parts.

The channel estimator and the matched filter have the sampled received signal, r, as input. r is a sampled sequence which is expected to contain the received GSM burst. Also, the over sampling factor, OSR, described as f_s/r_b , with f_s Being the sample frequency, and r_b the symbol rate, is input to both of these two blocks. Finally, these two blocks have L_h as input, where L_h is the desired length of the channel impulse response measured in bit time durations. The channel estimator passes an estimate of the channel impulse response, h, to the matched filter. Also, the channel estimator passes the sample number corresponding to the estimated beginning of the burst in r.

To interface correctly with the MLSE implementation return a down sampled one sample per symbol – version of the now matched filtered burst. Also, the MLSE requires information concerning the matched filter. This information is supplied by also returning the impulse response autocorrelation, i.e. R_{hh} . The method used for obtaining synchronization is based on the mathematical properties of this training sequence. The training sequence, TRAINING, used in GSM sim is as follows

$$TRAINING = [0, 0, 1, 0, 0, 1, 0, 1, 1, 1, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 1, 0, 1, 1, 1]$$

De-Multiplexing

The tasks of de-multiplexing, de-interleaving and decoding the data, are implemented in three separate blocks. The overall task of these three blocks is to regenerate the transmitted coded data blocks.

De-Interleaving

The de-interleaver reconstructs the received encoded data, The operation is the inverse of the interleaver, and may thus be considered as a reordering of the shuffled bits. The deinterleaver operates according to the following two formulae

$$R = 4 \cdot B + (b \bmod 8)$$

$$r = 2 \cdot ((49 \cdot b) \bmod 57) + ((b \bmod 8) \text{ div } 4)$$

Channel Decoding

The coding scheme utilized in the GSM system may be viewed as a two level coding where an inner and an outer coding is made use of. The inner coding scheme is here made up of the GMSK modulation and demodulation while the outer code is a regular convolution encoding scheme as in figure 8.

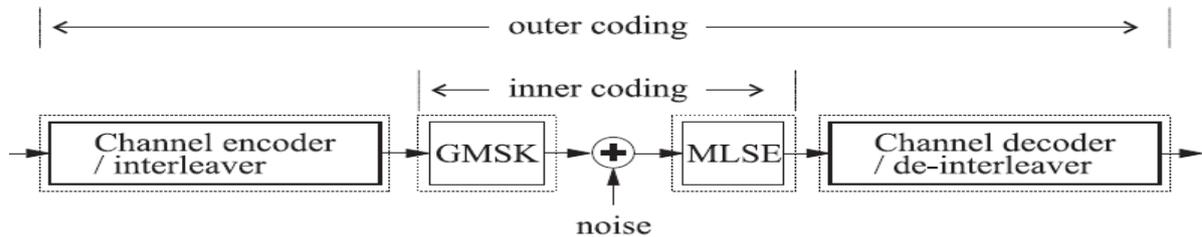


Fig 8: the two level coding scheme utilized in the GSM system.

The outer code used in GSM can be explained with a state transition diagram that may be used in the decoding of the received sequence as in figure 9. State transition diagram deduced from the transmitter encoder with states ordered in a binary manner.

Transmitted bit information

- '1'
- '0'

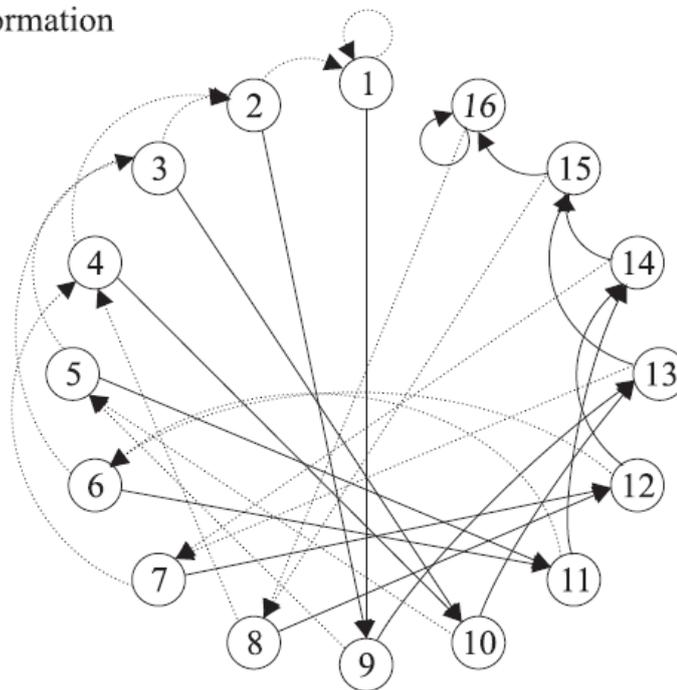


Fig 9: State transition diagram for the GSM system.

That is, state 1 represents the situation where the encoder has all zeros in its registers, i.e $s_1 = \{0,0,0,0\}$ while state 2 is given as $s_2 = \{0,0,0,1\}$. This way it is possible to characterize the encoder completely.

The optimum decoder for a convolution encoded signal is the Viterbi decoder it represents a recursive optimal solution to the problem of estimating a state transition sequence of a finite state Markov process. As the convolution decoder operates on binary information a much simpler metric definition is used. At any particular discrete time instance, k , a set of metrics is given by

$$\lambda(s_0^k) = \sum_{i=0}^{k-1} \lambda(\xi_i)$$

$s_0^k = \{s_0, s_1, s_2, \dots, s_k\}$ represents a given

sequence of states and represents the i th state transition. The metric increase definition does, in principle, not differ much from the Hence, the following definition stands

$$\begin{aligned} \lambda(\xi_i) &= (y_k - x_k)^2 \\ &= \text{XOR}(y_k^{bin}, x_k^{bin}) \end{aligned}$$

where y_k^{bin} is the received binary symbol and x_k^{bin} the binary symbols expected to cause a given state transition.

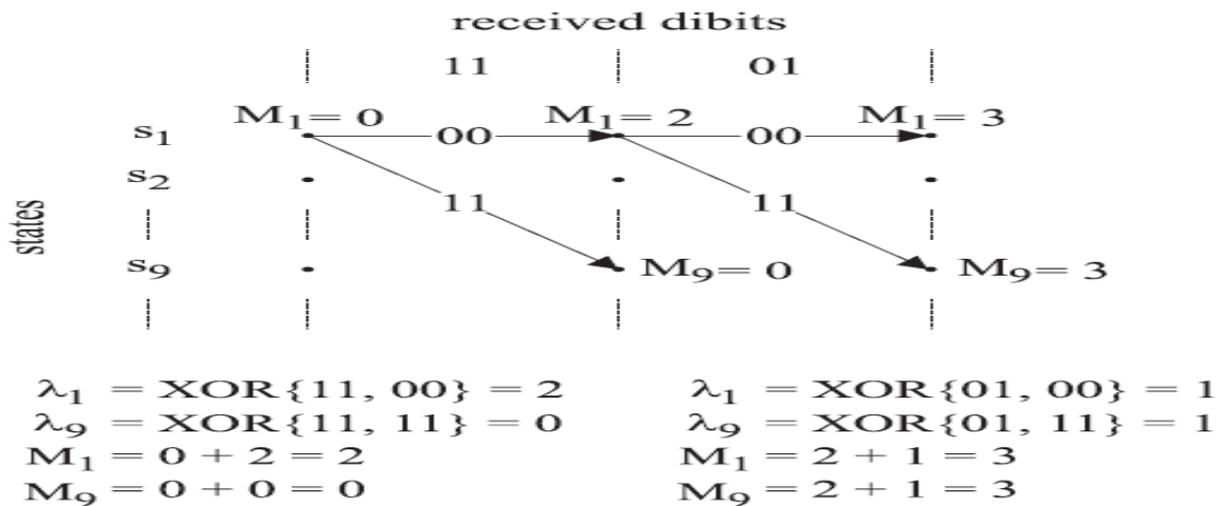


Fig 10: the metric calculations that comprises part of the channel decoder implementation.

IV. Transmitter and Receiver Test

Transmitter Test

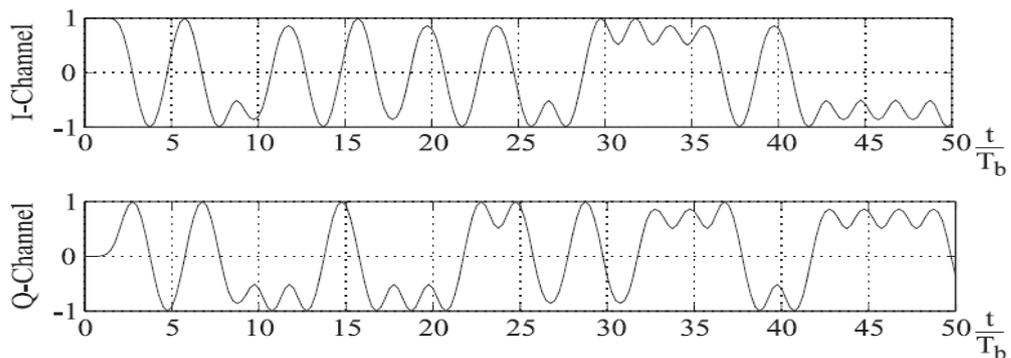


Fig 11: I and Q baseband outputs from the implemented modulator when given a random input sequence.

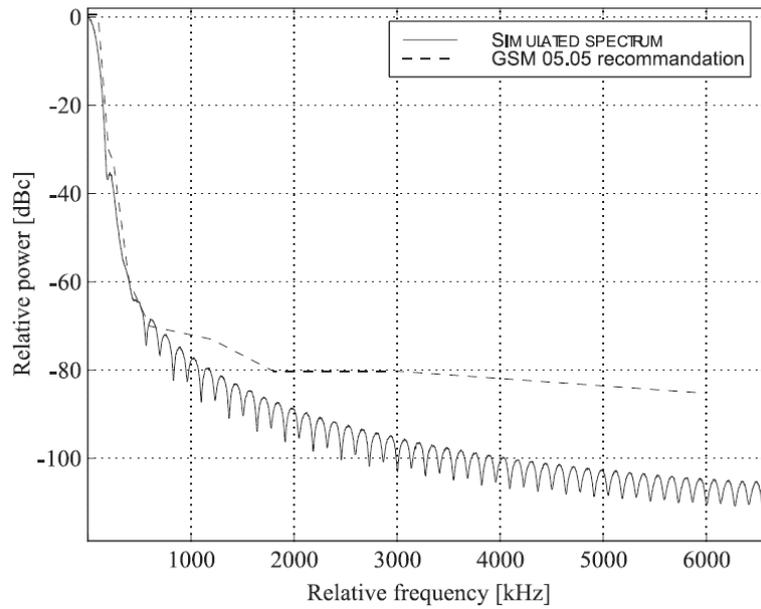


Fig 12: The power spectrum generated by the modulator. The spectrum is generated by averaging over 10000 spectras produced by GMSK modulated sequences each 1024 bits long. In the simulation a sample rate of $f_s 64^* r_b$ is used. The dashed line represents the GSM 05.05 requirement.

Receiver Test

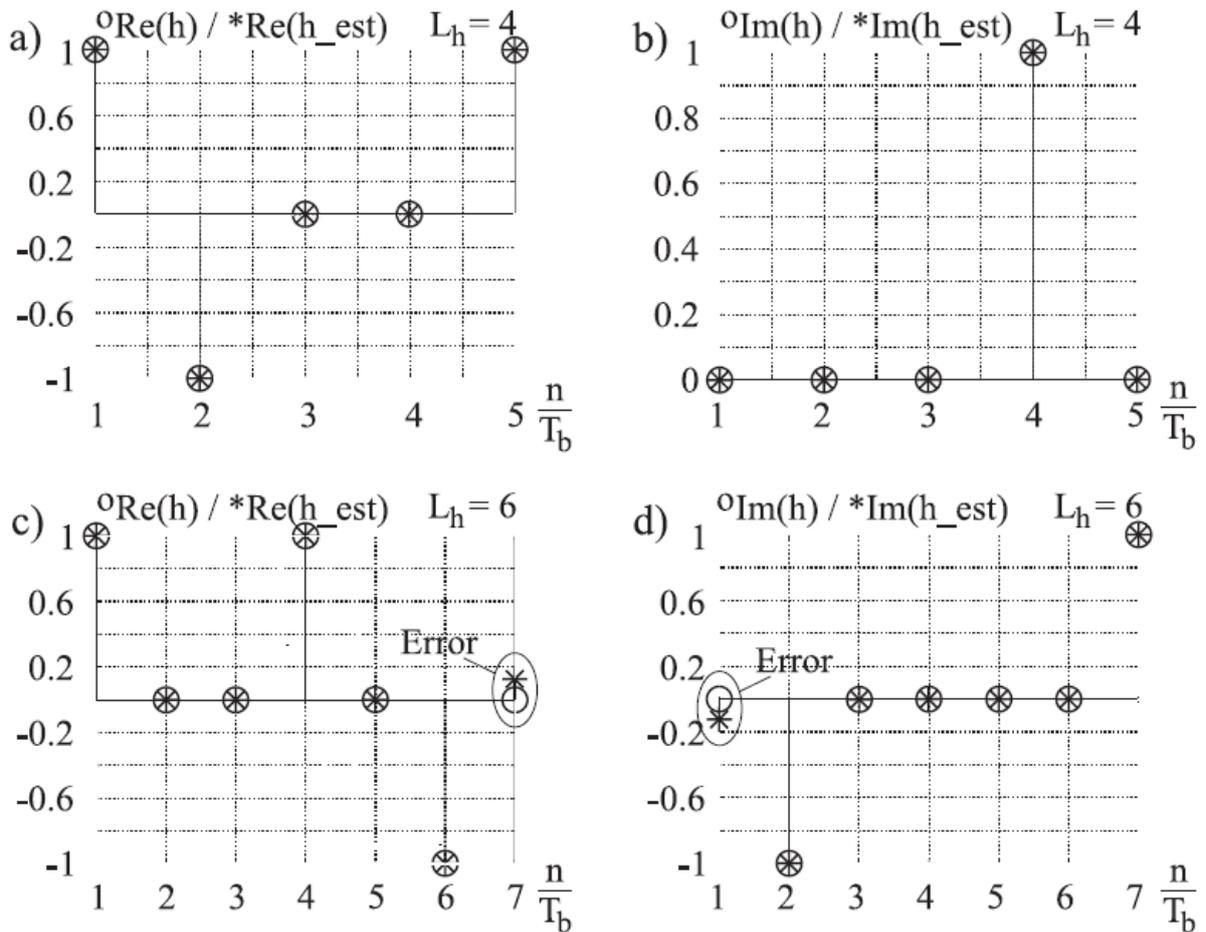


Fig 13: The results from two m.f.m tests. Real parts and imaginary parts of the actual response, h , and the estimated response, h_{est} , are compared. Actual values are indicated using 'o' while the estimated values are indicated using '*'.

- a) Real parts using a Lh of 4. b) Imaginary parts using a Lh of 4.
- c) Real parts using a Lh of 6. d) Imaginary parts using a Lh of 6

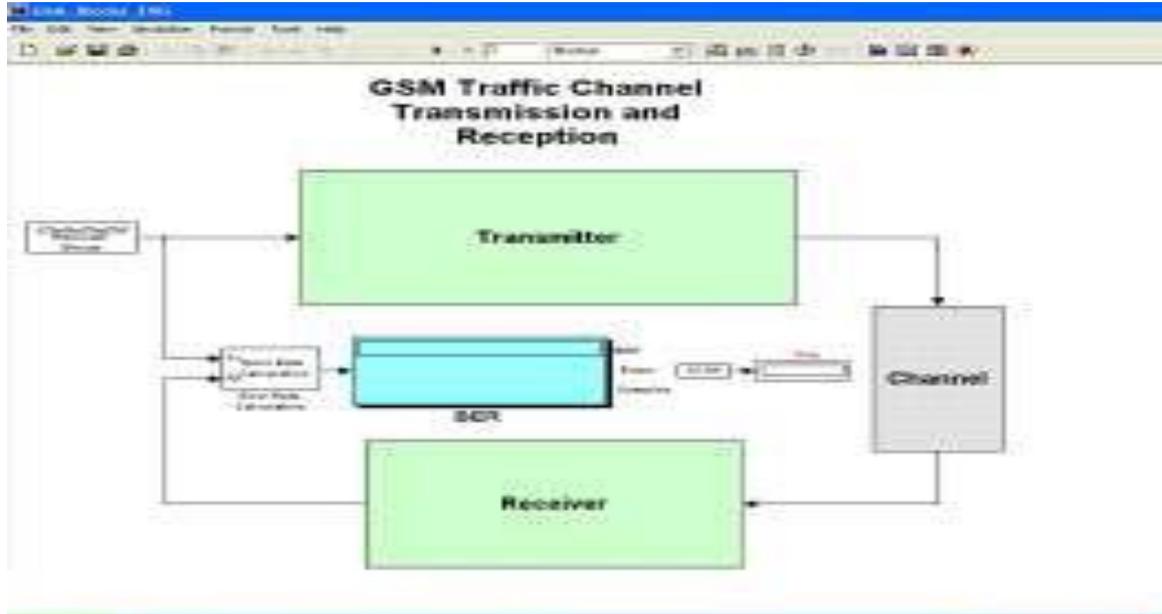


Fig 14: GSM Traffic channel Transmission and Reception

Conclusion

A set of SIMULINK blocks has been proposed for modeling, simulation and synthesis of GSM Radio link. This paper presents a comprehensive simulation tool that simulates the entire physical layer of GSM. We investigated the bit error rate. The BER can be pushed back to an acceptable limit with channel coding and antenna diversity techniques.

V. Acknowledgments

We wish to take this opportunity to express our sincere thanks to those who helped us directly or indirectly for the completion of this work.

References

- [1] Johan Brøndum. GSM data receiver structures. Technical report, Institute for Electronic Systems Department of Communication Technology, 1993.
- [2] Simon Haykin. *Communication Systems*. John Wiley & Sons, 2 edition, 1983.
- [3] John G. Proakis. *Digital Communications*. McGraw-Hill Book Company, 2 edition, 1989