# Implementation of Turbo Encoder and Decoder

**Mrs.K.M.Bogawar, Assistant Professor, Ms.Sharda Mungale Dr.Manish Chavan,**
*Dept. of Electronics,Priyadarshini College of Engineering Nagpur,*
*R.T.M.Nagpur University,India*
*Dept. of Electronics &Tele.*
*Ramdeobaba College of Engg.&*
*Technology,Nagpur,India*

**ABSTRACT :Turbo code has become the coding technique of choice in many communication and storage systems due to its near Shannon limit error correction capability. With requirement on increasing data rates for deep space mission, it is required to have efficient encoder and decoder. Turbo codes provide up to 0.8 dB improvement in Eb/No over the current best codes used by deep space missions. The total number of decoder iterations depends on the physical channel characteristics. In this paper, we proposed a turbo encoder with the 1/3,1/4,1/6 rate and turbo decoder.**

**Keywords:-Turbo coding, forward error correction, interleaving, puncturing, iterative decoding, MAP decoding**

## I. INTRODUCTION

In 1948, Shannon proved that every noisy channel has a maximum rate at which information may be transferred through it and that it is possible to design error-correcting codes that approach this capacity, or Shannon limit, provided that the codes may be unbounded in length. For the last six decades, coding theorists have been looking for practical codes capable of closely approaching the Shannon limit.

Turbo codes, a new technique of error correction coding developed in the 1990s.In 1993, a concatenated forward error correction (FEC) scheme, turbo coding was introduced by Berrou.Turbo coding is a very powerful error correction technique that has made a tremendous impact on channel coding in the last few years. It outperforms all previously known coding schemes by achieving near Shannon limit error correction using simple component codes and large interleavers. Turbo codes have become a 3G standard The iterative decoding mechanism, recursive systematic encoders and use of interleavers are the characteristic features of turbo codes. The use of turbo codes enhances the data transmission efficiency in digital communications systems. Turbo code development proceeded from theoretical analyses of polynomial selection, weight distributions imposed by interleaver designs, decoder error floors, and iterative decoding thresholds. A family of turbo codes was standardized and implemented and is currently in use by several spacecraft. JPL’s LDPC codes are built from protographs and circulants Turbo codes enable reliable communication over power-constrained communication channels at close to Shannon’s limit. However, a significant number of iterations are required to produce this result leading to higher latency. The task of channel coding is to encode the information sent over a communication channel in such a way that in the presence of channel noise, errors can be detected and/or corrected. Thus efficient implementation of turbo codes in order to meet real-time constraints is an active area of research.

Designing a channel code is always a tradeoff between energy efficiency and bandwidth efficiency. Codes with lower rate (i.e. bigger redundancy) can usually correct more errors. If more errors can be corrected, the communication system can operate with a lower transmit power, transmit over longer distances, tolerate more interference, use smaller antennas and transmit at a higher data rate. These properties make the code energy efficient. On the other hand, low-rate codes have a large overhead and are hence heavier on bandwidth consumption. Also, decoding complexity grows exponentially with code length, and long (low-rate) codes set high computational requirements to conventional decoders. According to Viterbi, this is the central problem of channel coding: encoding is easy but decoding is hard.The paper is organized as follows. In the I section this paper presents the complete analysis of turbo encoding for the different data rates that is for 1/2,1/3,1/4,1/6 followed by the design of interleaver and discusses the aspect of puncturing used in the turbo encoding.In the section II we have explained the turbo encoder .In the section III there is explanation of different methods used for the implementation of stopping rules for the turbo decoder .In the section IV simulation result have been given.

## II. GENERIC TURBO ENCODER

A turbo code is formed from the parallel concatenation of two codes separated by an interleaver. The two encoders used are normally identical. The code is in a systematic form, i.e. the input bits also occur in the output. Encoders are recursive systematic convolutional codes(RSC).The interleaver reads the bits in a pseudo-random order. The fundamental turbo code encoder is built using two identical recursive systematic convolutional (RSC) codes with parallel concatenation. An RSC encoder is typically $r$ = 1/2 and is termed a component encoder. The two component encoders are separated by an interleaver. The main purpose of the interleaver is to randomize bursty error patterns so that it can be correctly decoded. It also helps to increase the minimum distance of the turbo code. Only one of the systematic outputs from the two component encoders is used, because the systematic output from the other component encoder is just a permuted version of the chosen systematic output.

The 3GPP UMTS Turbo encoding specified in the 3GPP UMTS specification uses parallel concatenated convolutional code (PCCC). An information sequence is encoded by a convolutional encoder, and an interleaved version of the information sequence is encoded by another convolutional encoder. It supports 3GPP UMTS compliant with support for block sizes from 40 to 5,114. All 3GPP UMTS interleaver block sizes are selectable at run time. Code rate is 1/3 only. Other code rates can be achieved by external rate matching Double-buffering allows the encoder to receive data while processing the previous data block.

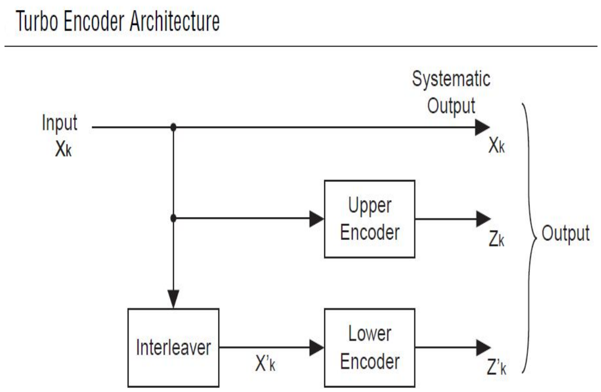

Figure.1 Generic turbo encoder

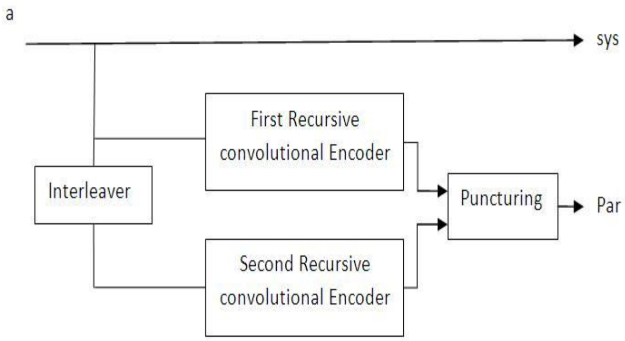
a
sys
First Recursive convolutional Encoder
Interleaver
Puncturing
Par
Second Recursive convolutional Encoder

Figure.2.Turbo encoder with puncturing

## III. INTERLEAVER DESIGN

The interleaver is a very important constituent of the turbo encoder.It spreads the bursty error pattern and also increases the free distance.Thus,it allows the decoders to make uncorrelated estimates of the soft output values The convergence of the iterative decoding algorithm improves as correlation of the estimates decreases. Turbo encoder consists of interleaver unit which can be used to increase the BER performance by varying the interleaving size. In turbo code, interleaver unit is a random block that is used to rearrange the input data bits with no repetition. Interleaver unit is used in both encoder and decoder part. At the encoder side it generates a long block of data, whereas in decoder part it correlates the two SISO (soft in soft out) decoder and helps to correct the error. At the decoder side after passing the encoded data from first decoder some of the errors may get corrected, then we again interleaver, this first decoded data and pass through the second decoder. Here, remaining error may get correct. Like this, we are repeating the process for more number of times. The purpose of the interleaver is to offer each encoder an uncorrelated or a "random" version of the information, resulting in parity bits from each RSC that are independent. How "independent" these parity bits are, is essentially a function of the type and length/depth of the interleaver. The design of interleaver in itself is a science. In a typical Viterbi code, the messages are decoded in blocks of only about 200 bits or so, where as in Turbo coding the blocks are on the order of 16K bits long. The reason for this length is to effectively randomize the sequence going to the second encoder. The longer the block length, the better is its correlation with the message from the first encoder, i.e. the correlation is low. On the receiving side, there are same number of decoders as on the encoder side, each working on the same information and an independent set of parity bit. This type of structure is called Parallel Concatenated Convolutional Code or PCCC.

There are certain types of interleavers "row-column" interleaver in which data is written row-wise and read columnwise. While very simple, it also provides little randomness. In helical" interleaver data is written row-wise and read diagonally. In an odd-even interleaver first, the bits are left uninterleaved and encoded, but only the odd-positioned coded bits are stored. Then, the bits are scrambled and encoded, but now only the even-positioned coded bits are stored. Odd-even encoders can be used, when the second encoder produces bone output bit per one input bit. In pseudo-random interleaver defined by a pseudo-random number generator or a look-up table.

## IV. PUNCTURING

Puncturing is a technique used to increase the code rate. A rate 1=3 encoder is converted to a rate 1=2 encoder by multiplexing the two coded streams. The multiplexer can choose the odd indexed outputs from the output of the upper RSC encoder and its even indexed outputs from the lower one.

## V. TURBO ENCODER

Turbo codes are constructed by applying two or more simple-to-decode encoding rules to different permutations of the same information sequence. Figure shows the functional diagram of turbo encoder which consists of encoder A and encoder B in which parallel concatenation scheme is used. An interleaver is used between the encoders to improve burst error correction capacity or to increase the randomness of the code. First component encoder operates on the information bit sequence directly whereas the input to second component encoder is given using the interleaver .The interleaver used is random interleaver or pseudo random interleaver uses a fixed random permutation and maps the input sequence according to the permutation order. The output from the two component encoder is periodically mixed with uncoded bits to form the turbo encoder output.

The code rate is 1 /2,1/3,1/4,1/6 bit per symbol and the code type is systematic parallel concatenated turbo code with two component code. The component codes are recursive convolution code. Backward and forward connection vectors are used. To drive the encoder to all zero state at the end of the block, an additional t tail bits are appended. This is called trellies termination to bring encoder to all zero states. Convolutional code polynomial used is primitive feedback polynomial for the recursive convolutional component code, instead of a nonprimitive polynomial. The encoded weights of self-terminating weight-2 input sequences increase linearly with the distance between its two 1s, with a slope that depends on the characteristics of the constituent codes. When they are far apart, the encoded sequence looks like the code's infinite impulse response up to the point where the second bit error terminates the further accumulation of weight. With a primitive divisor polynomial, all weight-2 inputs are guaranteed to cause a traverse through all of the trellis states before any possible return to the all-zero state, piling up maximum encoded weight.

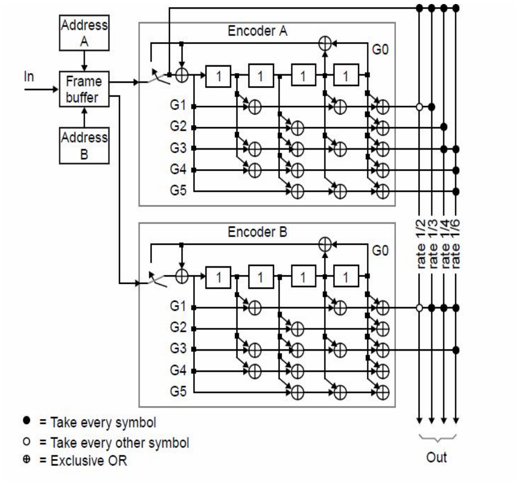

Figure.3 Turbo Encoder Functional Diagram

Weight distribution for turbo codes can be done by random, non-random and semi-random permutations. Random permutations do a very good job of teaming low weights with high weights for the vast majority of possible information sequences. It is possible to design nonrandom permutations that ensure that the minimum due to weight-2 input sequences. But there are semi-random permutations that perform nearly as well as the designed nonrandom permutations with respect to Weight-2 input sequences and are not as susceptible to being foiled by higher weight inputs.

## VI.TURBO DECODER

A Turbo decoder consists of two single soft-in soft-out (SISO) decoders that work iteratively. The output of the first (upper decoder) feeds into the second to form a Turbo decoding iteration. Turbo codes are decoded using a method called the Maximum Likelihood Detection or MLD. Filtered signal is fed to the decoders, and the decoders work on the signal amplitude to output a soft "decision" The a priori probabilities of the input symbols is used, and a *soft* output indicating the reliability of the decision is calculated which is then iterated between the two decoders. The form of MLD decoding used by turbo codes is called the Maximum a-posteriori Probability or MAP. However, ML decoder is often too complex to be implemented for turbo decoding because of the very complex trellis structure caused by the interleaver between the two constituent codes(CCs).In iterative decoding algorithm the two constituent decoders are used to perform SISO decoding over the coded sequences generated by the two CCs respectively, where the reliability information is exchanged between them during the decoding iterations.

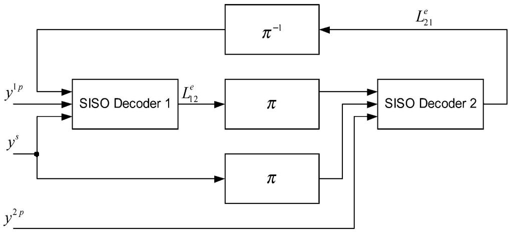

Figure.4 Block diagram of iterative decoder

## REFERENCES:

[1] By Kenneth S. Andrews, Dariush Divsalar, Sam Dolinar "The Development of Turbo and LDPC Codes for Deep-Space Applications" IEEE | Vol. 95, No. 11, November 2007

[2] "Development Plan for Turbo Encoder Core and Devices Implementing the Updated CCSDS Telemetry Channel Coding Standard" ,Sandi Habinc,Gian Paolo Calzolari,Enrico Vassallo

[3] " Comparison and Analysis of ECSS based Turbo Decoder Stopping Rules" Adnan Aslam Chaudhry, Mussadiq Ali Shah, 2012 IEEE

[4] "Parallel Interleaver Design and VLSI Architecture for Low-Latency MAP Turbo Decoders" Rostislav (Reuven) Dobkin, Michael Peleg, IEEE TRANSACTIONS ON VERY LARGE SCALE INTEGRATION (VLSI) SYSTEMS, VOL. 13, NO.pp.427-438, APRIL 2005

[5] "VHDL Implementation of a Turbo Decoder With Log-MAP-Based Iterative Decoding" Yanhui Tong, Tet-Hin Yeap, *Member, IEEE,* and Jean-Yves Chouinard, *Senior Member,* IEEE TRANSACTIONS ON INSTRUMENTATION AND MEASUREMENT, VOL. 53, NO. 4,pp.1268-1278, AUGUST 2004

[6] " VLSI Architectures for Turbo Decoding Message Passing Using Min-Sum for Rate-Compatible Array LDPC Codes" Kiran Gunnam, Weihuang Wang, Gwan Choi, Mark Year, 2007 IEEE

[7] " PARALLEL VLSI ARCHITECTURE FOR MAP TURBO DECODER" Rnstislav (Reuven) Dnbkin, Michael Peleg, Ran Ginnsar,pp.384-388,2002 IEEE