

Design Of An Architecture For Error Monitoring In FEC Communication Channel Using Turbo Codes

Ms. R. Karthiga.¹, Mr. K. Saravanan ², Mr. M.R. Bibin ³, Mr. M. Raja⁴

Karthiga635@gmail.com¹, saravanan.k2006@gmail.com²,
bibinmrme@gmail.com³, raja.muthiah1@gmail.com⁴

PG Scholar¹, Assistant Professor ^{2,3,4},

Dept of ECE, Vel Tech Multi Tech Dr.Rangarajan
Dr.Sakunthala Engineering College, Avadi Chennai.

Abstract

Error correcting codes is the detection of error and reconstruction of the original, error free data. An error correcting code is a system of adding redundant data or parity data to message, such that it can be recovered by a receiver. When a number of error we introduced, either during the process of transmission or a storage data. In this project, the proposed system used Turbo codes. It is one type of forward error correcting code. A turbo codes separated by an interleaver. Turbo codes are found to be used in 3G mobile communications and satellite communication as well as other application. This turbo codes architecture which detects and corrects the error.

Key words: Error correcting codes, Turbo codes, Interleaver, Simulation, Encoder, Decoder, Forward error correction.

I. INTRODUCTION

Error detection is the detection of errors caused by noise or other impairments during transmission from the transmitter to the receiver. Error correction is the detection of errors and reconstruction of the original, error-free data. An error-correcting code (ECC) or forward error correction (FEC) code is a system of adding redundant data, or parity data, to a message, such that it can be recovered by a receiver even when a number of errors (up to the capability of the code being used) were introduced, either during the process of transmission, or on storage. Since the receiver does not have to ask the sender for retransmission of the data, a back-channel is not required in forward error correction, and it is therefore suitable for simplex communication such as broadcasting.

Error-correcting codes are frequently used in lower-layer communication, as well as for reliable storage in media such as CDs, DVDs, hard disks, and RAM. Random-error-correcting codes based on minimum distance coding can provide a suitable alternative to hash functions when a strict guarantee on the minimum number of errors to be detected is desired. Repetition codes, described below, are special cases of error-correcting codes: although rather inefficient, they find applications for both error correction and detection due to their simplicity.

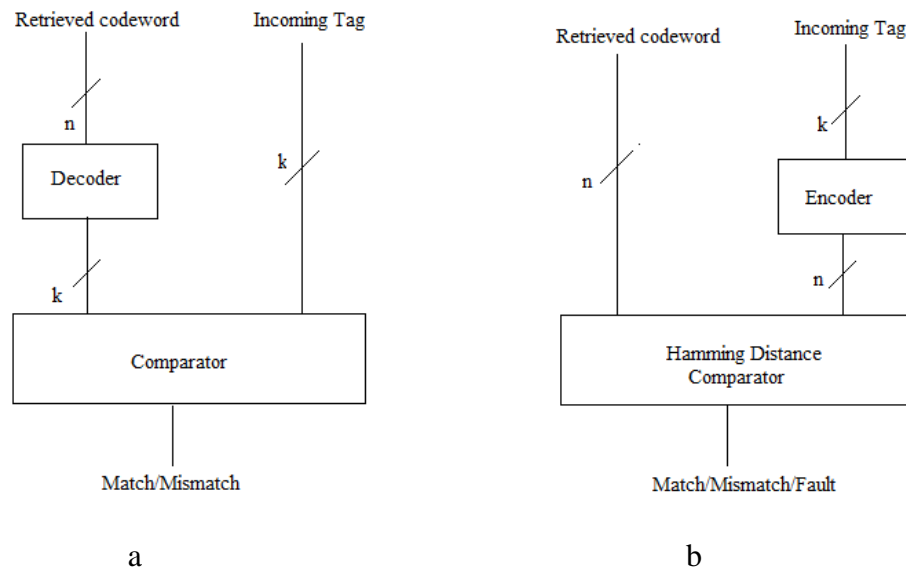


Fig 1. (a) Decode and compare architecture (b) encode and compare architecture.

FEC processing in a receiver may be applied to a digital bit stream or in the demodulation of a digitally modulated carrier. For the latter, FEC is an integral part of the initial analog-to-digital conversion in the receiver. The Viterbi decoder implements a soft-decision algorithm to demodulate digital data from an analog signal corrupted by noise. Many FEC coders can also generate a bit-error rate (BER) signal which can be used as feedback to fine-tune the analog receiving electronics. The maximum fractions of errors or of missing bits that can be corrected is determined by the design of the FEC code, so different forward error correcting codes are suitable for different conditions.

II. TURBO CODES

A turbo code is formed from the parallel concatenation of two codes separated by an interleaver. Turbo codes are finding use in 3G mobile communications and satellite communications as well as other applications where designers seek to achieve reliable information transfer over bandwidth or latency constrained communication links in

the presence of data corrupting noise. Turbo codes are nowadays competing with LDPC codes, which provide similar performance.

The name "turbo code" arises from the feedback loop used during normal turbo code decoding, which was analogized to the exhaust feedback used for engine turbo charging. Hagenauer has argued the term turbo code is a misnomer since there is no feedback involved in the encoding process.

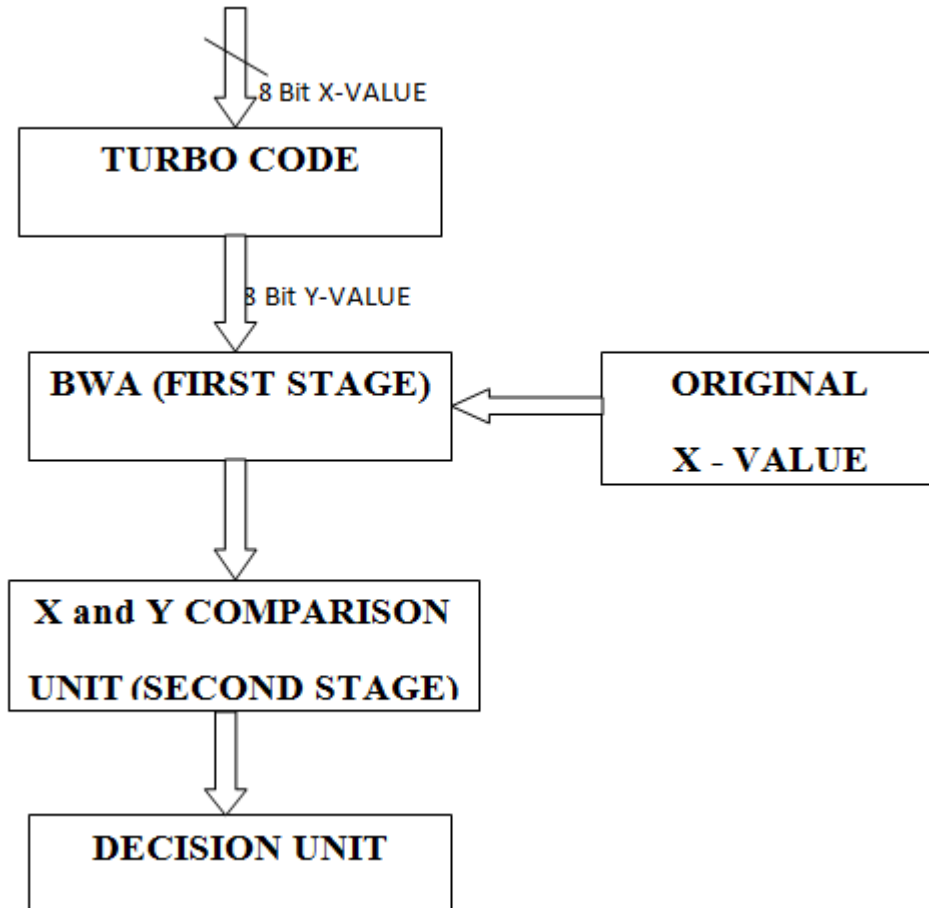


Fig 2. Implementation of turbo code with BWA architecture for error correction code

III. BLOCK DIAGRAM FOR TURBO CODES

Recently, double binary circular turbo code has received tremendous attention. Due to its better error-correcting capability than classical turbo code, it has commenced practical applications in current communication standards, such as DVB-RSC and IEEE 802.16 (Wimax). However current decoding schemes will incur a huge computation complexity. In this paper, authors present a novel decoding scheme for double binary circular turbo codes, which will not only reduce the computation

complexity, but also give at least 0.5 dB performance gain compared with current decoding schemes

Forward error correction (FEC) requires that the decoder should also be capable of correcting a certain number of errors, i.e. it should be capable of locating the positions where the errors occurred. Since FEC codes require only simplex communication, they are especially attractive in wireless communication systems, helping to improve the energy efficiency of the system. In the rest of this paper we deal with binary FEC codes only. Next, we briefly recall the concept of conventional convolutional codes. Convolutional codes differ from block codes in the sense that they do not break the message stream into fixed-size blocks. Instead, redundancy is added continuously to the whole stream. The encoder keeps M previous input bits in memory. Each output bit of the encoder then depends on the current input bit as well as the M stored bits.

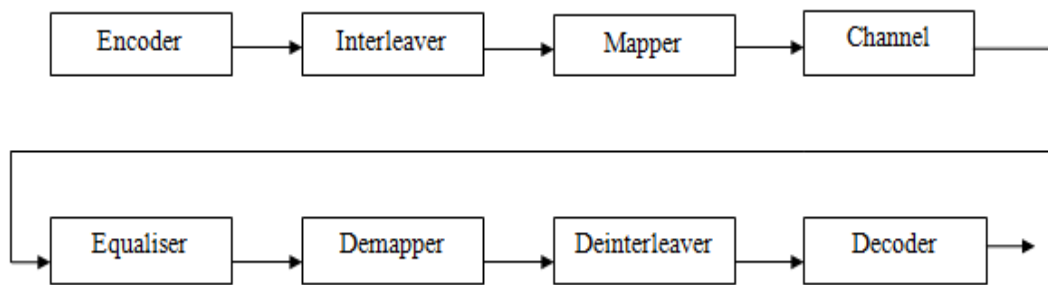


Fig 3. Turbo code flow

IV. (16,8)CODES BUTTERFLY WEIGHTED ACCUMULATOR

An architecture grounded on the data path. It contains multiple butterfly-formed weight accumulators (BWAs) proposed to improve the latency and complexity of the Hamming distance computation. The basic function of the BWA is to count the number of 1's among its input bits. It consists of multiple stages of HAs where each output bit of a HA is associated with a weight. The HAs in a stage are connected in a butterfly form so as to accumulate the carry bits and the sum bits of the upper stage separately. In other words, both inputs of a HA in a stage, except the first stage, are either carry bits or sum bits computed in the upper stage. This connection method leads to a property that if an output bit of a HA is set, the number of 1's among the bits in the paths reaching the HA is equal to the weight of the output bit.

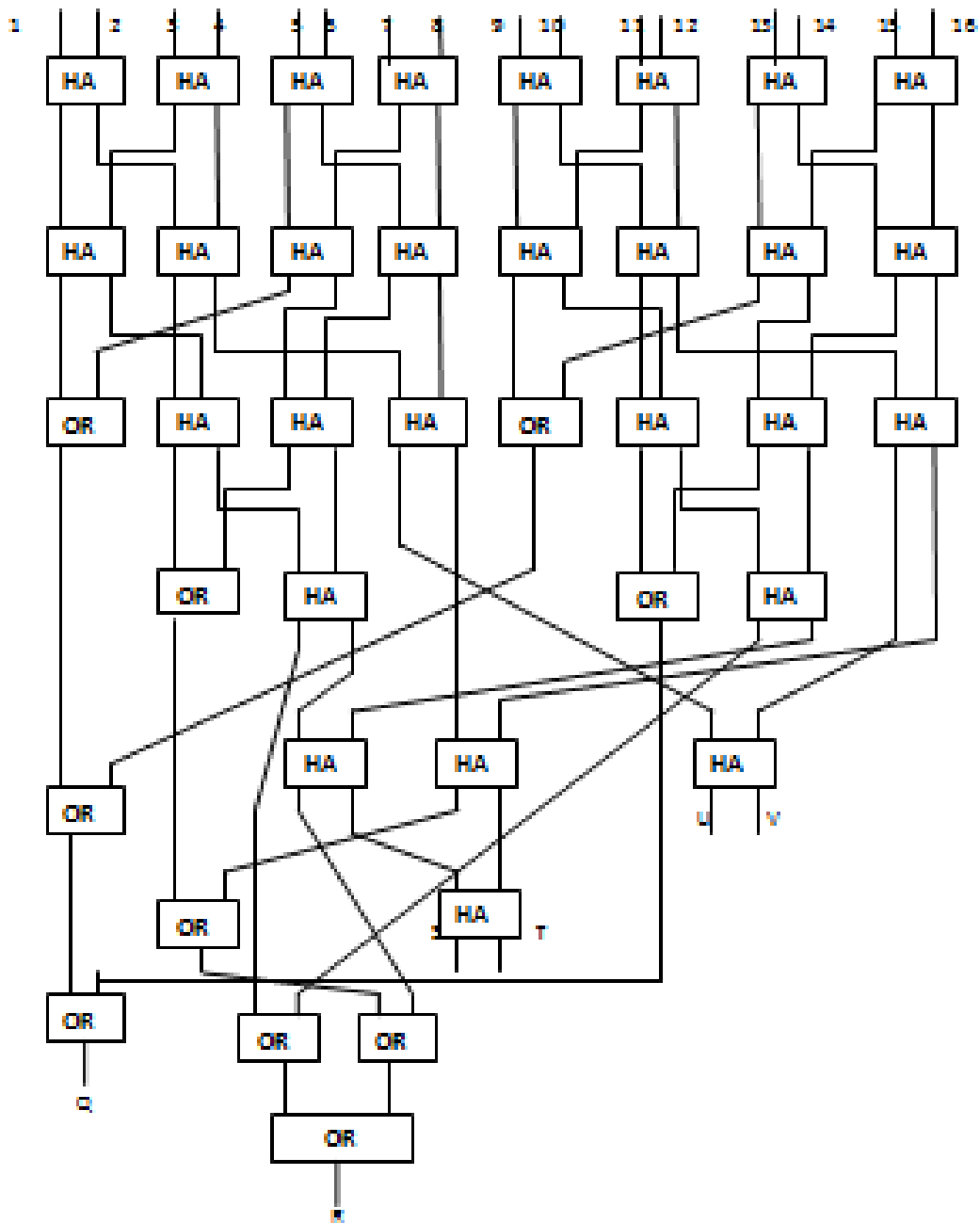


Fig 4. First and second level circuits for a (16, 11) code

V. ENCODING WITH INTERLEAVING

The general concept allows for free choice of the encoders and the interleaver; most designs follow the ideas presented. The two encoders used are normally identical; the code is in a systematic form, i.e. the input bits also occur in the output.

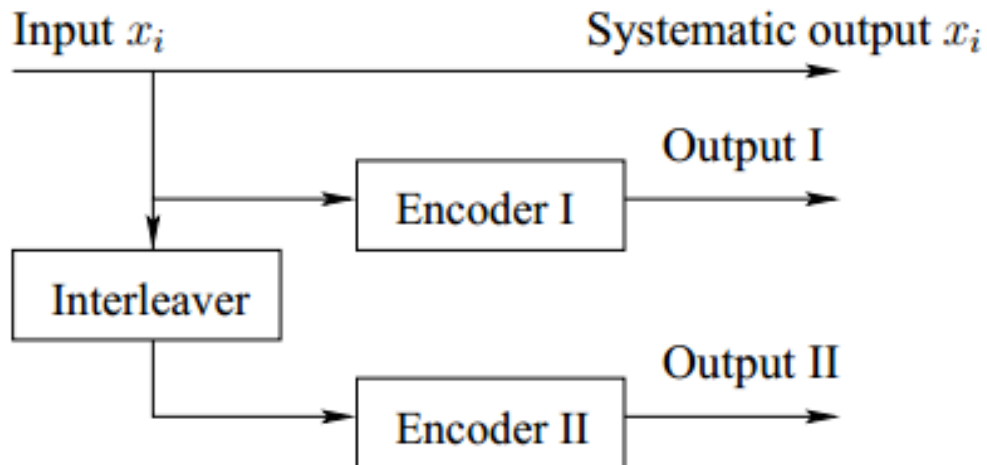


Fig 5. The generic turbo encoder.

The interleaver reads the bits in a pseudo-random order. The choice of the interleaver is a crucial part in the turbo code design. The task of the interleaver is to “scramble” bits in a (pseudo-)random, albeit predetermined fashion. This serves two purposes. Firstly, if the input to the second encoder is interleaved, its output is usually quite different from the output of the first encoder. This means that even if one of the output code words has low weight, the other usually does not, and there is a smaller chance of producing an output with very low weight. Higher weight, as we saw above, is beneficial for the performance of the decoder.

VI. DECODING APPROACH

The demodulator makes a “hard” decision of the received symbol, and passes to the error control decoder a discrete value, either a 0 or a 1. The disadvantage of this approach is that while the value of some bits is determined with greater certainty than that of others, the decoder cannot make use of this information. A soft-in-soft-out (SISO) decoder receives as input a “soft” (i.e. real) value of the signal. The decoder then outputs for each data bit an estimate expressing the probability that the transmitted data bit was equal to one. In the case of turbo codes, there are two decoders for outputs from both encoders. Both decoders provide estimates of the same set of data bits, albeit in a different order. If all intermediate values in the decoding process are soft values, the decoders can gain greatly from exchanging information, after appropriate reordering of values. Information exchange can be iterated a number of times to enhance performance.

VII. SIMULATION

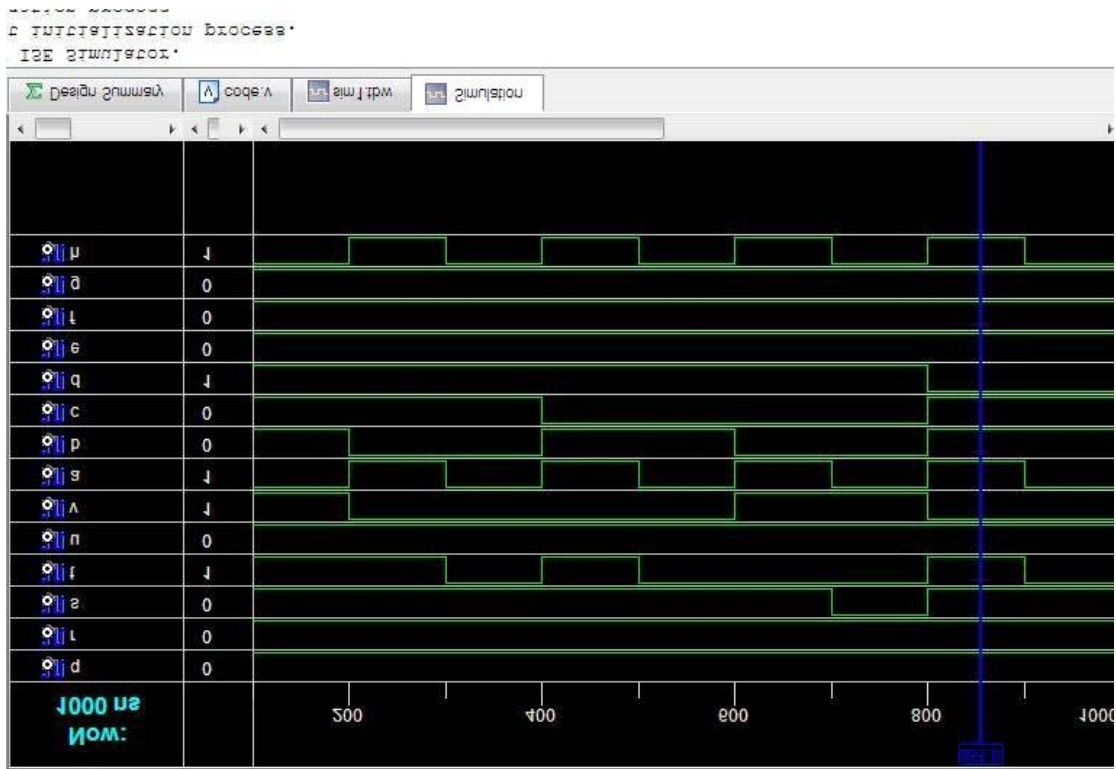


Fig 6. Output wave form of (16, 8) codes

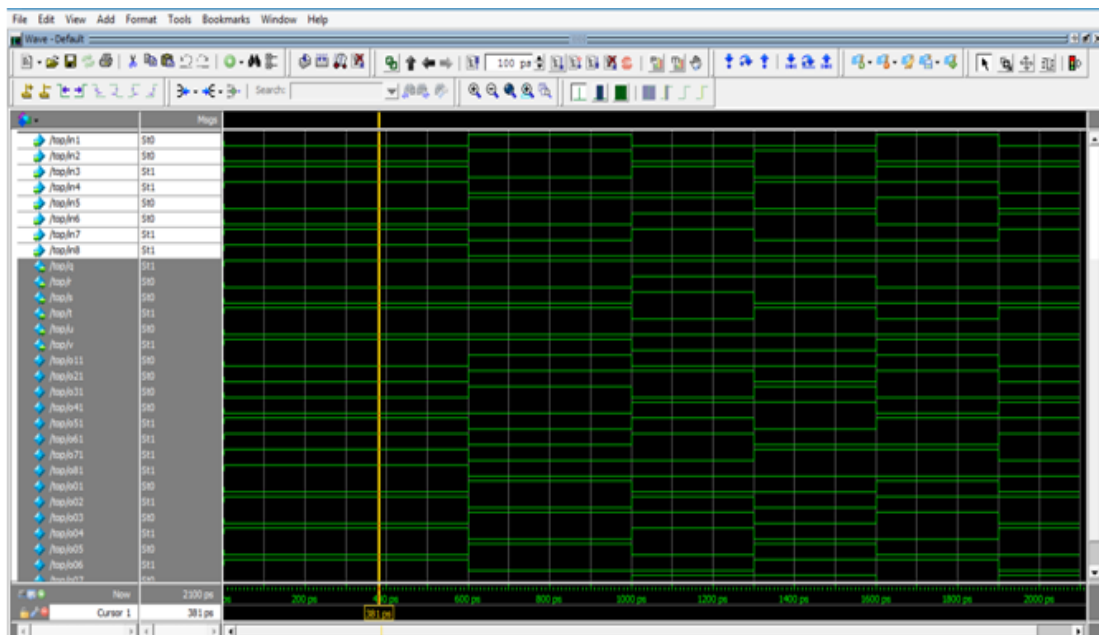


Fig 7. Output wave form for with error

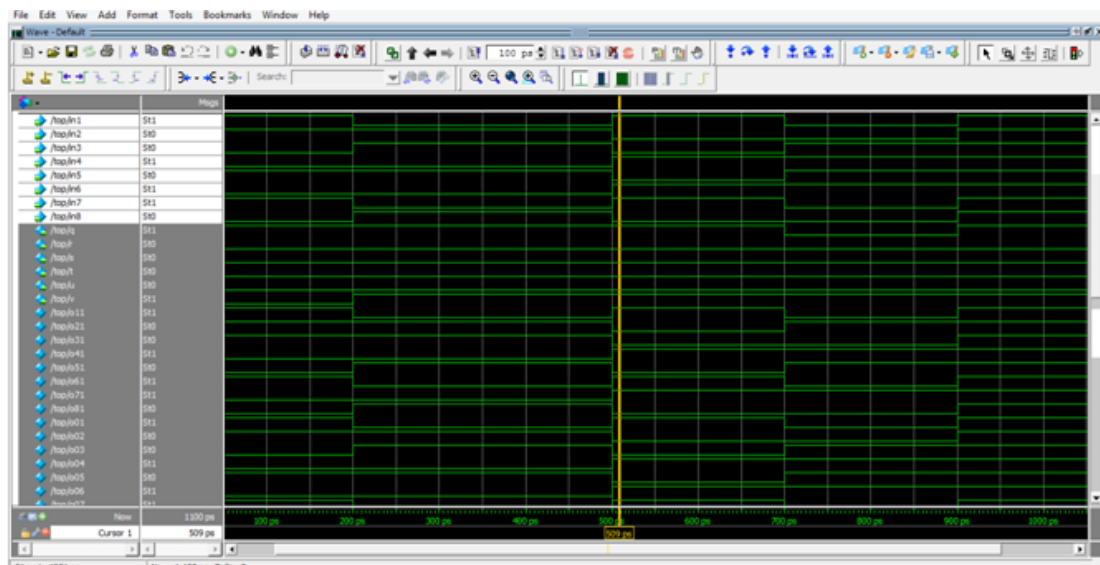


Fig 8. Output waveform for without error

VIII. CONCLUSION

In this paper, turbo code are using in 3G mobile communications and satellite communication as well as other application. This turbo codes architecture which detects and corrects the error. Turbo codes are a recent development in the field of forward error correction channel coding. The codes make use of three simple ideas parallel concatenation of codes to allow simpler decoding; interleaving to provide better weight distribution; and soft decoding to enhance decoder decisions and maximize the gain from decoder.

REFERENCE

1. Byeong Yong Kong, Jihyuck Jo, Hyewon Jeong, Mina Hwang, Soyoung Cha, Bongjin Kim, and In-Cheol Park "Low-Complexity Low-Latency Architecture for Matching of Data Encoded With Hard Systematic Error-Correcting Codes" IEEE Trans. Very Large Scale Integr. (VLSI) Sys. vol. 22, no. 7, July 2014.
2. W. Wu, D. Somasekhar, and S.-L. Lu, —Direct compare of information coded with error-correcting codes,|| IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 20, no. 11, pp. 2147–2151, Nov. 2012.
3. J. D. Warnock, Y.-H. Chan, S. M. Carey, H. Wen, P. J. Meaney, G. Gerwig, H. H. Smith, H. Chan, J. Davis, P. Bunce, A. Pelella, D. Rodko, P. Patel, T. Strach, D. Malone, F. Malgioglio, J. Neves, D. L. Rude, and W. V. Huott —Circuit and physical design Implementation of the microprocessor chip for the zEnterprise system,|| IEEE J. Solid-State Circuits, vol. 47, no. 1, pp. 151–163, Jan. 2012.

4. Y. Lee, H. Yoo, and I.-C. Park, —6.4Gb/s multi-threaded BCH encoder and decoder for multi-channel SSD controllers,|| in ISSCC Dig. Tech. Papers, 2012, pp. 426–427.
5. M. Tremblay and S. Chaudhry, —A third-generation 65nm 16-core 32-thread plus 32-scout-thread CMT SPARC processor,|| in ISSCC. Dig. Tech. Papers, Feb. 2008, pp. 82– 83.
6. J. Chang, M. Huang, J. Shoemaker, J. Benoit, S.-L. Chen, W. Chen, S. Chiu, R. Ganesan,
- G. Leong, V. Lukka, S. Rusu, and D. Srivastava, —The 65-nm 16-MB shared on-die L3 cache for the dual-core Intel xeon processor 7100 series,|| IEEE J. Solid-State Circuits, vol. 42, no. 4, pp. 846–852, Apr. 2007.
7. S. Lin and D. J. Costello, *Error Control Coding: Fundamentals and Applications*, 2nd ed. Englewood Cliffs, NJ, USA: Prentice-Hall, 2004.
8. H. Ando, Y. Yoshida, A. Inoue, I. Sugiyama, T. Asakawa, K. Morita, T. Muta, and T. Motokurumada, S. Okada, H. Yamashita, and Y. Satsukawa, —A 1.3 GHz fifth generation SPARC64 microprocessor,|| in IEEE ISSCC. Dig. Tech. Papers, Feb. 2003, pp. 246–247.
9. AMD Inc. (2010). Family 10h AMD Opteron Processor Product Data Sheet, Sunnyvale, CA, USA [Online].

