International Journal of Engineering and Advanced Technology (IJEAT)
ISSN: 2249-8958 (Online), Volume-4 Issue-6, August 2015

Implementation of Max Log BCJR Algorithm in Turbo Decoder Architecture for Wireless Sensor Networks

Shamila. N, Manju. M. S

Abstract— The transmission of signal in a compressed form can cause a high sensitivity of error in wireless sensor networks. Error control coding (ECC) is used to determine the error occurring in the communication networks during the transmission of information from one point to another. It provides gain and energy reduction during transmission at the cost of decoder power consumption. The BCJR algorithm named after its inventors: Bahl,Cocke, Jelnik and Raviv is critical to iteratively decoded error correcting codes including turbo codes and parity check codes. The turbocodes used in the algorithm helps in the modification of the original BCJR algorithm and helps in the simplification of calculations. In this paper Max-Log-BCJR (Bahl, Cocke, Jelnik and Raviv) Algorithm is used. This algorithm appears to lend itself to both low complexity energy-constrained scenarios, as well as to the high-throughput scenarios. The Algorithm is very sensitive to SNR mismatch and requires accurate estimation of noise variation. This algorithm simply finds the minimum of the LLRs (Logarithmic Likelihood Ratios),so it uses only one ACS Operation.

Index Terms— Error Control Coding, Logrithamic Likelihood Ratios, Max-Log BCJR Algorithm, Turbocodes

I. INTRODUCTION

A sensor network is composed of a large number of sensor nodes, which are densely deployed either inside the phenomenon or very close to it. The position of sensor nodes does not have been predetermined, allowing random deployment in inaccessible terrains or dynamic situations; however, this also means that sensor network protocols and algorithms must possess self-organizing capabilities. During the transmission of information from one node to another in the compressed form the possibility of occurrence of error is high. To reduce this error some of the error control techniques are used. Turbo codes are a class of high-performance forward error correction (FEC)[2] codes developed in 1993, which were the first practical codes to closely approach the channel capacity, a theoretical maximum for the code rate at which reliable communication is still possible given a specific noise level. Turbo codes have been used also for its high coding gain. FEC is accomplished by adding redundancy to the transmitted information using a predetermined algorithm. Each redundant bit is invariably a complex function of many original information bits.

Manuscript published on 30 August 2015.
***** Correspondence Author (s)
Shamila.N*, P.G Scholar, Department of Electronics and Communication, Lourdes Matha College of Science and Technology, University of Kerala, Kerala. India.
Ms Manju.M.S, Asst. Professor, Department of Electronics and Communication, Lourdes Matha College of Science and Technology, University of Kerala, Kerala. India.

An important problem in WSN's is the efficient representation of data generated by a discrete source. This process of representation is accomplished by a method called source coding. The device that performs source encoding is called source encoder. Source Coding [2] technique takes the advantage of statistical knowledge of the source signal to enable efficient encoding. For the development of an efficient source encoder it must satisfies two functional requirements

- Code words produced by the encoder are in the binary form.
- Source code is uniquely decodable, so that the original source sequence can be reconstructed perfectly.

The design of a channel code is always a trade-off between energy efficiency and bandwidth efficiency. Codes having more redundant bits can usually correct more errors. That means the wsn's operate at lower transmit power can tolerate more interference and noise and transmit at higher data rate, thus becomes more energy efficient. However, these codes also have a large overhead and have more bandwidth consumption. Also, the decoding complexity of the code also grows exponentially with code length. Thus, low rate codes set high computational requirements to the conventional decoders. The central problem of channel coding id encoding is easy but decoding is hard. For every combination of bandwidth (W), channel type, signal power (S) and received noise power (N), there is a theoretical upper limit on the data transmission rate for a given bandwidth, channel type, signal power and received noise power such that the data transmission is error-free. The limit is called the channel capacity or the Shannon capacity[1]. The formula for additive white Gaussian noise (AWGN) channel is

$$R < W \log_2\left(1 + \frac{S}{N}\right) \ [bits/seconds]. \qquad (1)$$

In practical transmission, no such thing as an ideal error free channel exists. Instead, the bit error rate is brought to an arbitrarily small constant often chosen at 10^{-5} or 10^{-6}. If the transmission rate, the bandwidth and the noise power are fixed, we get a lower bound on the amount of energy that must be expended to convey one bit of information. Hence, Shannon capacity[1] sets a limit to the energy efficiency of a code.

II. TURBO ENCODER AND TURBO DECODER

The general structure used in turbo encoders is shown in Fig. 1. Two component codes are used for encoding the same input bits, but an interleaver[7] is placed between the encoders. The Turbo Encoder produces three outputs (one systematic output bit and two parity bits) for every input bit.

The general structure of the turbo decoder shown in Fig. 2. Two component decoders are linked by interleavers[7] in a structure similar to that of the encoder.

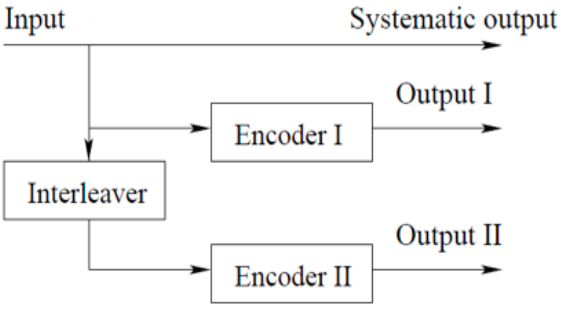

Fig 1.Turbo Encoder

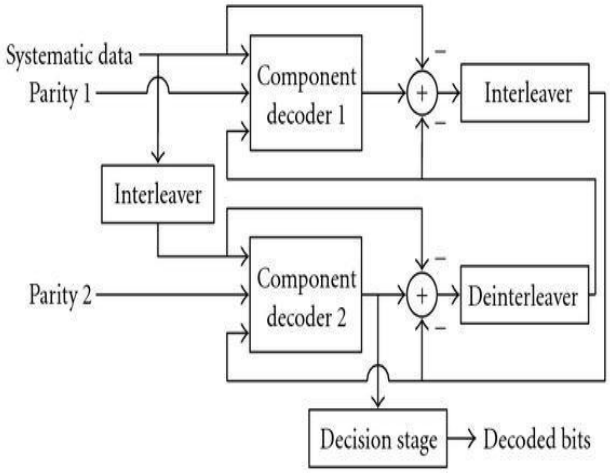

Fig 2.Turbo Decoder

Each decoder takes three inputs: the systematically encoded channel output bits, the parity bits transmitted from the associated component encoder, and the information from the other component decoder about the likely values of the bits concerned. This information from the other decoder is referred to as a priori information. The component decoders have to exploit both the inputs from the channel and this a priori information. They must also provide what are known as soft outputs for the decoded bits. This means that as well as providing the decoded output bit sequence, the component decoders must also give the associated probabilities for each bit that it has been correctly decoded.

The soft outputs from the component decoders are typically represented in terms of the so-called log likelihood ratios (LLRs)[4], the polarity of which gives the sign of the bit, and the amplitude the probability of a correct decision. The LLRs [4]are simply, as their name implies, the logarithm of the ratio of two probabilities. For example, the LLR $L(u_k)$ for the value of a decoded bit (u_k) is given by

$$L[u_k] = \ln\left(\frac{P(u_k=+1)}{P(u_k=-1)}\right) \qquad (2)$$

III. MAX -LOG BCJR DECODER

Fig.3 shows the basic building block of the MAX-Log-BCJR[3] architecture using a single recursion unit, the data gathering for the output calculation.

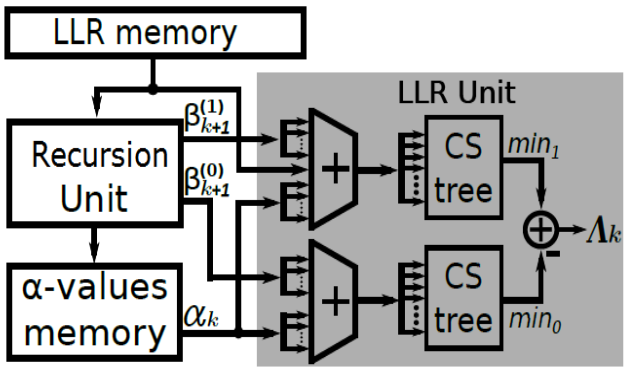

Fig 3: The MAX-Log-BCJR decoder architecture.

Both α and β metrics [3]are needed to compute the LLRs. The set of α -metrics calculated during the forward recursion must be stored in the α-values memory for the whole data block. During the backward recursion, all the information necessary for the computation of the LLRs is obtained. The computation of the LLRs starts from the last bit of the block. The remainders of the probabilities are calculated while the backward recursion advances. The minimum of the sums of α, β and γ-metrics for all zero-transitions and for all one-transitions have to be found and afterwards subtracted from each other. The input LLRs participate only in the one-transitions summation, since the branch metrics of all zero transitions are zero.

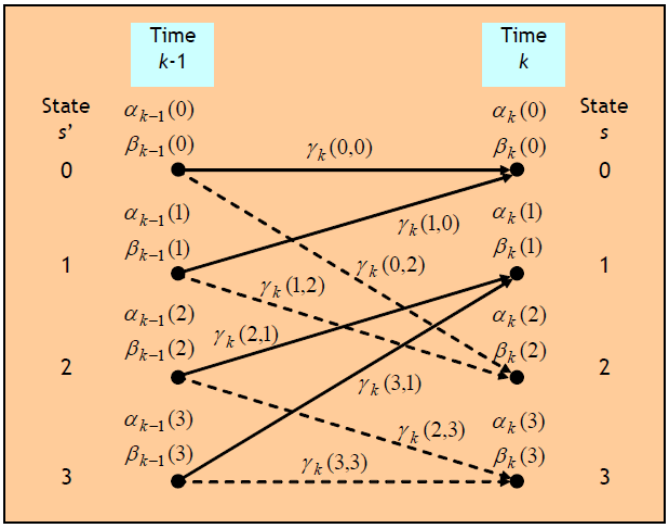

Fig 4. Ccalculation of α,β and γ.

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

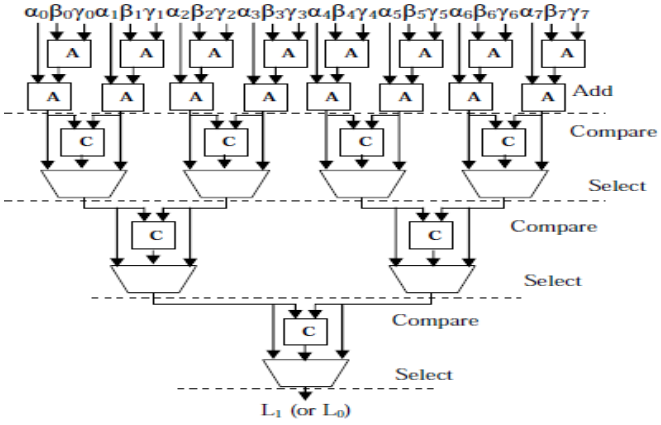

Fig 5. Architecture of LLR Unit.

The α-metrics[3], starting with the metrics from the states S_{n-1}, are retrieved from the memory. Two blocks of adders add the -metrics in sequence with the $\beta^{(1)}$ and $\beta^{(0)}$-metrics. As the branch metrics also participate in the one-transition summation, the upper adders block in Fig. 5 has 2^{n-k+1} adders, while the lower one has 2^{n-k} adders. After the sums are done, a binary tree of compare select (CS) units selects the lowest values among all the one-transitions sums, min_1. Since we also have to calculate the minimum of the sums for all the zero-transitions, min_0, a second binary tree is used. Fig.5 shows the architecture of a CS tree with 8 inputs. Each tree is comprised of 2^{n-k} -1 CS units, which are implemented like the compare select functions of the ACS units (a subtracter and a multiplexer). Pipeline is considered, given the high delay of such structure.After the lower values have been obtained from both trees, the soft-output Λ_k is calculated by subtracting min_0 from min_1.

IV. IMPLEMENTATION RESULTS

The Turbo Decoder Architecture is synthesized and simulated using XILINX ISE 14.7. The RTL schematic and simulation wave window is given here.

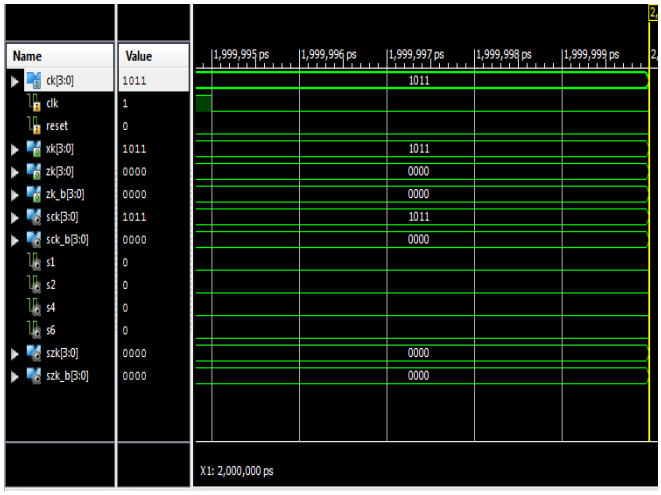

Fig 6. Simulation Result of Turbo Encoder

Fig 7. Simulation Result of recursion unit

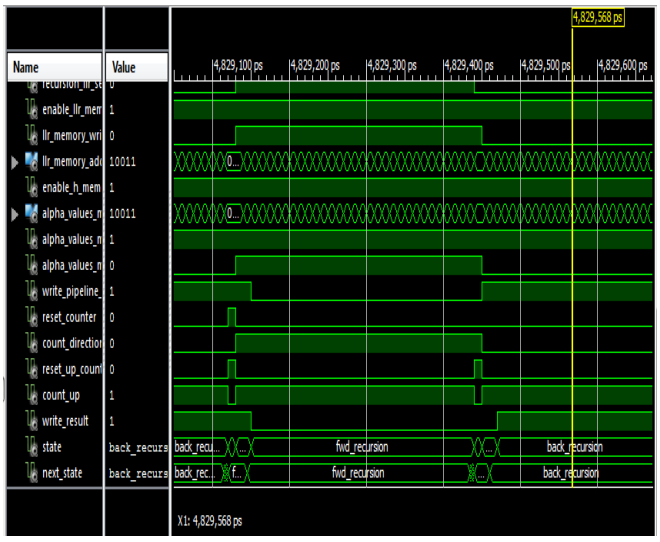

Fig.8 Simulation Result of Max log BCJR decoder

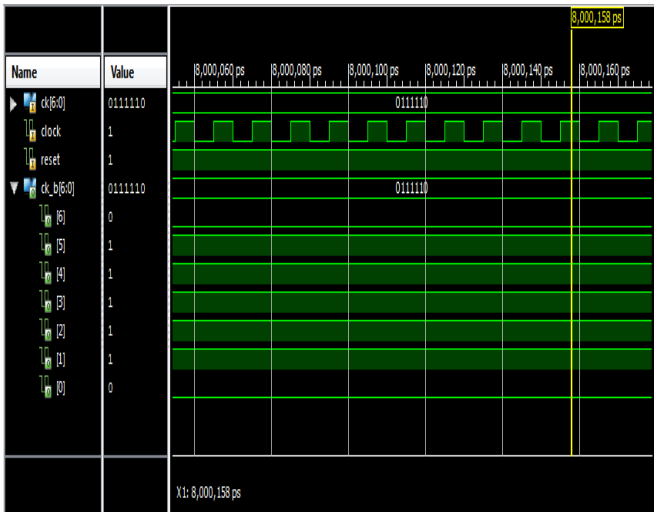

Fig 9. Simulation Result of Interleaver

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

Device Utilization Summary			
Slice Logic Utilization	Used	Available	Utilization
Number of Slice Registers	0	93,120	0%
Number of Slice LUTs	0	46,560	0%
Number of occupied Slices	0	11,640	0%
Number of LUT Flip Flop pairs used	0		
Number of bonded IOBs	7	240	2%
Number of RAMB36E1/FIFO36E1s	0	156	0%
Number of RAMB18E1/FIFO18E1s	0	312	0%
Number of BUFG/BUFGCTRLs	0	32	0%
Number of ILOGICE1/ISERDESE1s	0	360	0%
Number of OLOGICE1/OSERDESE1s	0	360	0%

Fig 10. Device Utilization Summary

V. CONCLUSION

In this paper we introduce a Max- Log BCJR Algorithm for a low throughput communication system. The work has been implemented in Spartan-6 FPGA and simulated in Xilinx ISE 14.7 and the result has been computed. Results show the reduced delay in processing and low hardware complexity.

REFERENCES

1. C. Berrou, A. Glavieux, and P. Thitimajshima, “Near Shannon Limit Error Correcting Coding and Decoding: Turbo Codes,” in Proceedings of the IEEE International Conference on Communications, vol. 2,Geneva, Switzerland, 1993, pp. 1064–1070
2. L. Li, R. G. Maunder, B. M. Al-Hashimi, and L. Hanzo, “An Energy efficient error correction scheme for IEEE 802.15.4 wireless sensor networks,” Transactions on Circuits and Systems II, vol. 57, no.3, pp. 233–237, 2010.
3. P. Robertson, P. Hoeher, and E. Villebrun, “Optimal and Sub-Optimal Maximum A Posteriori Algorithms Suitable for Turbo Decoding,”European Transactions on Telecommunications, vol. 8, no. 2, pp. 119–125, 1997.
4. A LowComplexity Turbo Decoder Architecture for Energy-Efficient Wireless Sensor Networks ,IEEE Transactions on Very LargeScale Integration (VLSI) Systems Volume: 21 , Issue: 1 ,2013
5. F. Jelinek L.R. Bahl, J. Cocke and J. Raviv. Optimal decoding of linear codes for minimizing symbol error rate. IEEE Trans. on Inform., Vol. IT-20., pages 284–287, March 1974.
6. Claude Berrou and Alain Glavieux. Near optimum error-correcting coding and de-coding: Turbo Codes. IEEE Transactions on Communications, 44(10):1261–1271, October 1996.
7. Y. Sun and J. R. Cavallaro, “Efficient hardware implementation of a highly-parallel 3GPP LTE, LTE-advance turbo decoder,” Integr., VLSI J., vol. 44, no. 1, pp. 1–11, 2010.

Shamila.N received her B.Tech degree from University of Kerala, Kerala in 2009. Currently pursuing her M Tech Degree also from Lourdes Matha College of Science and Technology under University of Kerala.

Ms.Manju.M.S received her B.Tech from University of Kerala and M.Tech Degree from Cochin University of Science and Technology (CUSAT), Kerala in 2010. Currently working as Asst.Professor in Lourdes Matha College of Science and Technology. Her areas of interest are VLSI, Embedded Systems, and Microprocessors.

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)