International Journal of Applied Engineering Research ISSN 0973-4562 Volume 12, Number 21 (2017) pp. 10921-10925

Design and FPGA Implementation of Power Efficient Turbo Decoder for 4G LTE Standards

Manjunatha K N[1],* and Vaibhav A Meshram[2]
[1]*Research Scholar, Electronics Engineering, Jain University, Bengaluru, Karnataka, India.*
**Orcid Id: 0000-0003-1891-3900*
[2]*Member IEEE & Professor of Electronics and Communication Engineering Department, Dayanand Sagar University, Bengaluru, Karnataka, India.*

Abstract

The wireless communication has two significant blocks across transmitter and receivers are encoders and decoders. This work focuses on the design and implementation of turbo decoder in hardware description language (HDL) in verilog version. The turbo codes are very efficient in channel coding and are reaching the Shannon limit. The proposed design for turbo decoder uses the max-log algorithms instead of using max-log-MAP algorithm which computes on approximation. The design reduces the fixed number of iteration and performs the early termination which greatly reduces the power consumption utilized even after the decoding is completed. For early termination, the Sign Difference Ratio (SDR) is considered and across the hardware coding clock gating is introduced to avoid the unnecessary clock supply to achieve the power efficiency. The entire design has been implemented on vertex 4 and vertex 5 of Xilinx FPGAs. The power analysis is made and compared with recent existing technologies.

Keywords: 4GLTE, FPGA, Turbo Decoder, VLSI, Interleaver, Deinterleaver

INTRODUCTION

The high speed wireless communication system requires the high through-put decoding units. The decoding complexity increases as the speed increases in the system like WiMAX, HSDPA, HDTV and Ultra-HDTV applications. The 4G LTE provides the solution for the high speed wireless applications.

The forward error correcting codes and Turbo codes offer good performance and are robust in channel coding. The methods to reduce the number of errors normally occur at the time of decoding process. The log-BCJR algorithm [3] is time complex and has high latency. The turbo codes performance error reduces as the SNR grows, in other words, the curve becomes saturated as the SNR linearly increases. Due to this limitation the turbo codes are avoided being used for applications with very low error rate requirements.

To increase the rate of transmission the parallel turbo decoders are studied where configurable interleaver are introduced on a very- large-scale Integration platform. This fully operated parallel turbo decoders will support LTE and WiMax wireless standards [4]-[6]. The moderate FPGA versions are chosen to meet the power efficiency with all device utilization parameters fulfilled. The ASIC and any other VLSI integration are not considered due to the cost limitation and the high time to design. The FPGA single chip board with Xilinx® vertex-4 and above series is proposed to design and implementation, below vertex-4 series are over mapped for the device utilization.

SYSTEM DESIGN COMPONENTS

A. Turbo Decoder

This section discusses on turbo encoder uses the Recursive Systematic Convolutional (RSC) encoders arranged in parallel. The first encoder is simply taken out with three delay units. The second encoder is placed after the interleaver block (Π). These turbo encoders together produce the encoded frames which includes parity frame and a systematic frame. Each RSC code rate is R=1/3, with parity $P_{1,i}$ and $P_{2,i}$ with a total length of 3N bits.

The RSC encoder performs on the basis of M = 8 - state transition diagram [7] as shown in Figure2. The transition of encodes starts from the initial state $S_0 = 0$ and the subsequent state becomes $S_i \epsilon \{0,1,2,\ldots..M\text{-}1\}$ according to the respective message bit $S_i \epsilon \{0,1\}$, so it has two potential state to change from previous S_{i-1} to the current state S_i.

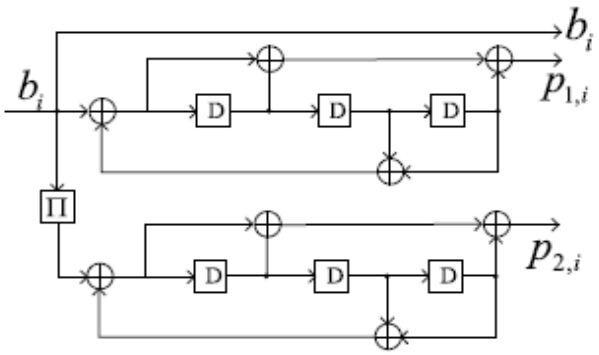

Figure 1: Turbo Encoder [1]

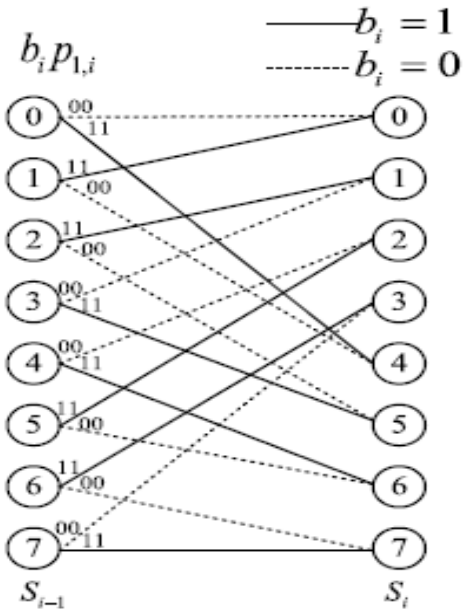

Figure 2: Trellie's Structure [1]

The extracted information from the turbo encoder is divided into systematic (Sys 1 and Sys 2) frame and parity frame. At the receiver side turbo decoder is placed to receive and decode the information given. This decoder has two sub decoders, decoder 1 and decoder 2 as shown in figure3. The sub decoder 1 process the two signals sys 1 and par 1 along with the extrinsic information from the decoder 2 block which is deinterleaved (Π^{-1}). The signal par 2 and interleaved signal of sys 1 called sys 2 are given to sub decoder 2 blocks along with interleaved signal from the sub decoder 1. The decision is made by the LLR 1 and LLR 2 as per the diagram shown.

According to the BCJR, the encoded sequence X = x_1, x_2… x_N, X_i=[x_{i1}, x_{i2}, x_{i3}] is the coded word for each i/P bit b_i and x_{i1}, x_{i2}, x_{i3} are sys1, par1 and par2 respectively. The log-likelihood ratio (LLR) as

$$L(b_i) = ln\frac{P(b_i=1)}{P(b_i=0)} \quad (1)$$

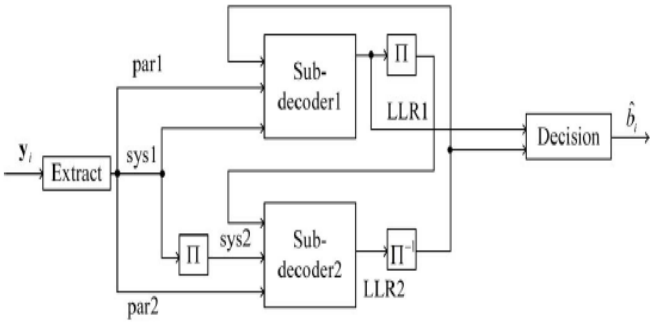

Figure 3: Turbo Decoder [1]

The sequence y=y_1, y_2, ……., y_N are received and given to the receiver for computation of the actual bit b_i. This decoding algorithm of posteriori LR is expressed as

$$L(b_i|y) = ln\frac{P(b_i=1|y)}{P(b_i=0|y)} \quad (2)$$

Here $L(b_i|y)$ is converted to a bit format later hard decision of the same is calculated. If $L(b_i|y)$<0, then the estimation of the message bit is b_i`=0 else $b_i$`=1. The extrinsic information is obtained after calculating the LLR.

B. Interleaver and Deinterleaver

The interleaver is an important block in a channel coding of turbo codes. The interleaver or deinterleaver is also designed in parallel to meet the decoding specification of parallel turbo decoder. Contention of memory is a very common problem in interleaving and it resolved by adopting the contention free interleaver [10]. In the proposed design, the block size is N, then the interleaver is defined as

$$\begin{cases} \Pi(i) = A(i) mod N \\ A(i) = f_1 i + f_2 i^2 \end{cases} \quad (3)$$

where, i=0,1,2,…..,N-1

Where f_1 is an odd number and f_2 is an even number, 'i' is the index number of input data. y_i and $\prod$(i) is the index number after interleaving.

The generation of interleaving target address becomes difficult because the A (i) uses the real time multiplication operation, so the computation complexity increases as the index number I increases till N-1. The low complexity on recursion and the derivation is as follows.

From (3)

$A(0)=0, A(1)=f_1+f_2$

$A(i+1)-A(i)=f_1+f_2+2i.f_2$

$A(i+2)-A(i+1)=f_1+3f_2+2i.f_2$ (4)

Then,

$A(i+2)-2A(i+1)+A(i)=2f_2$ (5)

Since A(0) and A(1) are initial values as defined, the interleaving index is calculated by recursion from (5). From this method, no multiplication is required which reduces the computation complexity very effectively. The same address generator cannot be used in parallel design because the computation takes place simultaneously. Therefore, to solve this, parallel address generator is used.

C. Reverse Address Generator

The frame of backward variables $\beta_{i,j}(s)$ must be reversed to compute the extrinsic information. This task takes N clock cycles for segmental decoding. Once the sequence of interleaver i/P is reversed the address generator is changed accordingly. To reduce the computation complexity, a reverse address generator [15] is proposed for parallel turbo decoder [10] which reduces the latency as well.

The address of target many can be modified as

$$\begin{cases} \Lambda_{p,k} = \Pi(pk) mod K \\ = A(pk) mod K, \ k = K-1, K-2, \ldots .0 \\ p = 1,2, \ldots \ldots, P \\ A(pk) = f_1 pk + f_2 (pk)^2 \end{cases} \quad (6)$$

The first two address of each sub block need to be generated and are $\Lambda_{p,K-1}$ and $\Lambda_{p,K-2}$ from (6)

$$\Lambda_{p,K-1} = f_1 p(K-1) + f_2 (p(K-1))^2 mod\ K$$

$$\Lambda_{p,K-2} = f_1 p(K-2) + f_2 (p(K-2))^2 mod\ K \quad (7)$$

Here, f1, f2 and p are integers, so reduce to

$$\Lambda_{p,K-1} = -f_1 + 2f_2 mod\ K$$

$$\Lambda_{p,K-2} = -2f_1 + 4f_2 mod\ K \quad (8)$$

From (4), by recursion, the following address of each filter bank is formulated as

$$\Lambda_{p,k+2} - 2\Lambda_{p,k+1} + \Lambda_{p,k} = 2f_2 mod\ K \quad (9)$$

From (9), it is observed that recursion and initial values are nothing to do with p, so address of all these sub-blocks are same and only channel address generation is required.

$$\Lambda_{k+2} - 2\Lambda_{k+1} + \Lambda_k = 2f_2 mod\ K \quad (10)$$

The target bank LLR of a sub-block is mapped into is identified by the value ∏(i)/K. the value has division operation and is expensive, so recursive computation is used and is restructured as

$\Gamma_{P,k} = \pi(p,k)/K$

$= [A(pk) \bmod N]/K$

Here the initial value of k=K-1 to K=0 of Γp,K-1 and Γp,K-2 are known. The parallel interleaved Deinterleaver can minimize the processing time compared to the other methods [15] but it cost a little in computation resources.

TURBO DECODER IMPLEMENTATION

The implementation of Turbo Decoder is done on the Field Programmable Gate Array (FPGA), due to its low cost and very short development cycle. The design is coded in the verilog hardware programming language and simulated using Xlinx® simulator of version 14.2 and the selected device for implementation is FGPA 4VLX25H676-12. The device uses the speed grade of -12 and 4 Digital Clock Managers (DCM) with one Global Clock (GCK).

Lower versions than the vertex 4 are over mapped with the required device utilization. Due to this device over mapping vertex 4 and vertex 5 are used for implementation and other designers are compared. The RTL schematic of the proposed design is as shown in the figure.

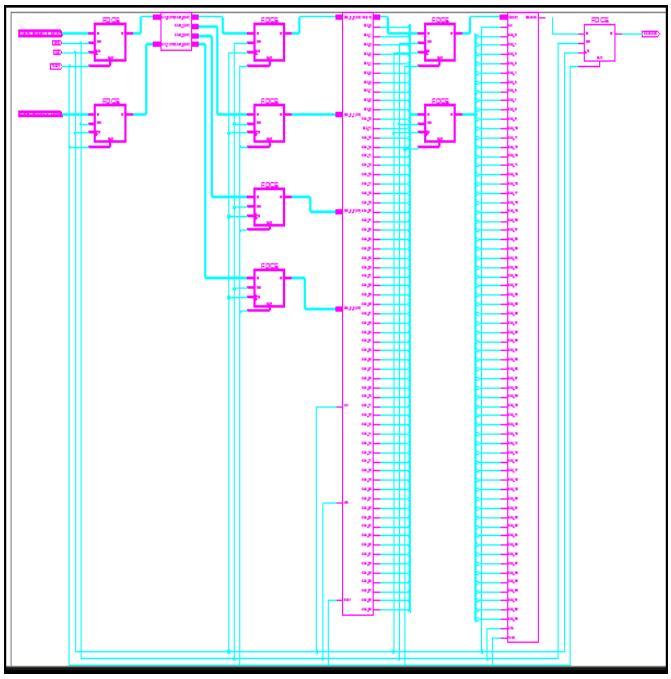

Figure 4: RTL schematic of Turbo Decoder.

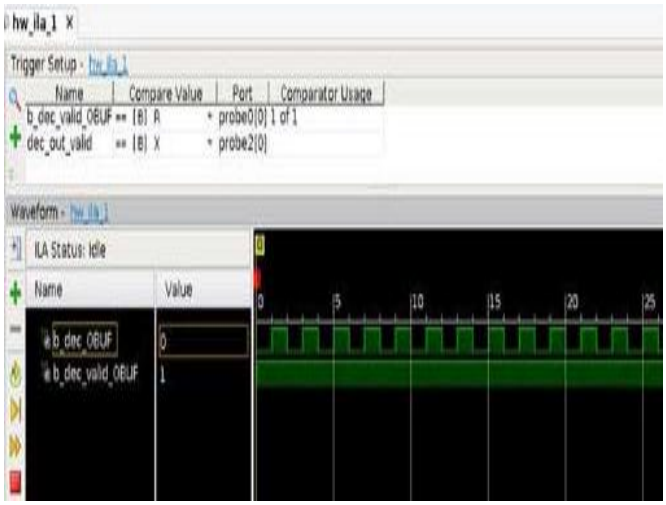

Figure 5: Simulation results of turbo decoder

Table 1: Summary of Design Implementation

Sl. No.		**Vertex 4**	**Vertex 5**
1	Device name	4vlx15sf363	5vlx30ff324
2	Speed	-12	-3
3	Clock frequency	160 MHz	550 MHz
4	Power consumption	131 mW	267 mW
5	Total Time	20.397ns	18.48ns

Table 2: Device Utilization Summary of Vertex 5

Slice Logic Utilization	**Used**	**Available**	**Utilization**
Number of Slice Registers	4.852	19.200	25%
Number of Slice LUTs	4.189	19.200	21%
Number used as logic	4.143	19.200	21%
Number of route-thrus	272	38.400	1%
Slice Logic Distribution			
Number of Occupied Slices	2.097	4.800	43%
Number with an unused Flip Flop	1.811	6.663	27%
Number with an unused LUT	2.474	6.663	37%
Number of fully used LUT-FF pairs	2.378	6.663	35%
IO Utilization			
Number of bonded IOBs	11	220	5%
Specific Feature Utilization			
Number of BUFG/BUFGCTRLs	2	32	6%

Table 3: Device Utilization Summary of Vertex 4

Logic Utilization	**Used**	**Available**	**Utilization**
Number of Slice Flip Flops	4.865	12.288	39%
Number of 4 input LUTs	4.843	12.288	39%
Number of occupied slices	4.193	6.144	68%
Number of slices containing only related logic	4.193	4.193	100%
Number of slices containing unrelated logic	0	4.193	0%
Total number of 4 input LUTs	4.867	12.288	39%
Number of bonded IOBs	11	240	4%
Number of BUFG/BUFGCTRLs	1	32	3%

EXPERIMENTAL RESULTS

This design is simulated and_implemented on the Xilinx FPGA platform with vertex 4 and vertex 5 series. The block length is varied between 0 to 2048 frame lengths. The clock gating technique is applied across the fundamental blocks of ACS, Trace back and branch metric units. This results in very much reduced power consumption.

The various iterations are also simulated to meet the Bit Error Rate (BER).

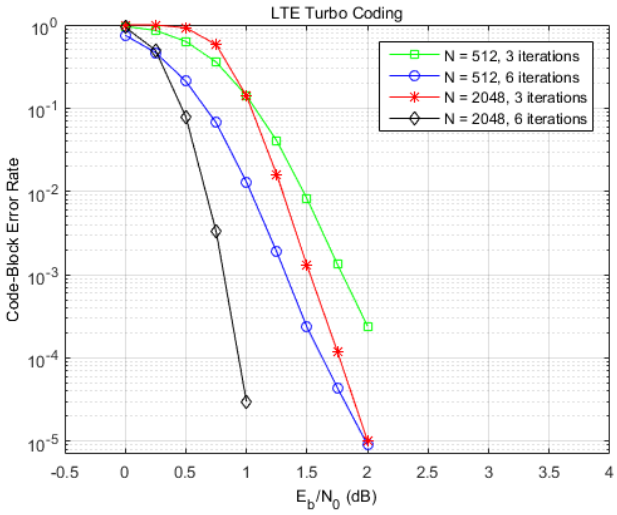

Figure 6: BER measured at various frame length and iterations

CONCLUSION

Turbo decoder implementation on FPGA board is one of the low cost and less time to design to prove the real time design and implementation. In the proposed design, a reverse address generator is introduced to reduce the latency and decoding complexity. Contention free memory interleaver/deinterleaver is proposed to avoid the memory related issue when number of iteration and bits/frame are increased. The clock-gating technique is adopted at the fundamental blocks level like trace back unit, branch metric and ACS units, which reduces the power dissipation in an acceptable scale. The power efficient turbo decoder is compared with respect to its device utilization. Clock frequency and the technology with the selected Xilinx device, the BER is mapped in Matlab simulink with various iterations on the very low SNR value of Additive White Gaussian Noise (AWGN). There are few observation are drawn from the experiment module function in address generation reduces the decoding complexity rather than the usual division operation and reduces the power dissipation as well. But still, there is a scope to enhance the increased through put with power efficient turbo decoder architecture design.

REFERENCES

[1] Hua Luo, Yue Zhang, "Wei Li, Low Latency Parallel Turbo Decoding Implementation for Future Terrestrial Broadcasting Systems", *IEEE Transactions on Broadcasting*, pp. 1-8, Jan. 2017.

[2] T. Biatek, W. Hamidouche, J.-F. Travers, and O. Deforges, "Optimal bitrate allocation in the scalable HEVC extension for the deployment of UHD services," *IEEE Trans. Broadcast.*, vol. 62, no. 4, pp. 826–841, Dec. 2016.

[3] S. Saito *et al.*, "8K terrestrial transmission field tests using dual polarized MIMO and higher-order modulation OFDM," *IEEE Trans. Broadcast.*, vol. 62, no. 1, pp. 306–315, Mar. 2016.

[4] D. Vargas, Y. J. D. Kim, J. Bajcsy, D. Gómez-Barquero, and N. Cardona, "A MIMO-channel-precoding scheme for next generation terrestrial broadcast TV systems," *IEEE Trans. Broadcast.*, vol. 61, no. 3, pp. 445–456, Sep. 2015.

[5] J. Calabuig, J. F. Monserrat, and D. Gómez-Barquero, "5th generation mobile networks: A new opportunity for the convergence of mobile broadband and broadcast services," *IEEE Commun. Mag.*, vol. 53, no. 2, pp. 198–205, Feb. 2015.

[6] L. Dai, Z. Wang, and Z. Yang, "Next-generation digital television terrestrial broadcasting systems: Key technologies and research trends," *IEEE Commun. Mag.*, vol. 50, no. 6, pp. 150–158, Jun. 2012.

[7] J. P. Woodard and L. Hanzo, "Comparative study of turbo decoding techniques: An overview," *IEEE Trans. Veh. Technol.*, vol. 49, no. 6, pp. 2208–2233, Nov. 2000.

[8] Y. G. Debessu, H.-C. Wu, H. Jiang, and S. Mukhopadhyay, "New modified turbo decoder for embedded local content in single-frequency networks," *IEEE Trans. Broadcast.*, vol. 59, no. 1, pp. 129–135, Mar. 2013.

[9] I. A. Chatzigeorgiou, M. R. D. Rodrigues, I. J. Wassell, and R. A. Carrasco, "Comparison of convolutional and turbo coding for broadband FWA systems," *IEEE Trans. Broadcast.*, vol. 53, no. 2, pp. 494–503, Jun. 2007.

[10] M. F. Brejza *et al.*, "20 years of turbo coding and energy-aware design guidelines for energy-constrained wireless applications," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 1, pp. 8–28, 1st Quart., 2015.

[11] Z. He, P. Fortier, and S. Roy, "Highly-parallel decoding architectures for convolutional turbo codes," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 14, no. 10, pp. 1147–1151, Oct. 2006.

[12] O. Muller, A. Baghdadi, and M. Jezequel, "From parallelism levels to a multi-ASIP architecture for turbo decoding," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 17, no. 1, pp. 92–102, Jan. 2009.

[13] *IEEE Standard for Air Interface for Broadband Wireless Access Systems*, IEEE Standard 802.16-2012, pp. 1–2542, Aug. 17, 2012.

[14] LTE: *Evolved Universal Terrestrial Radio Access (E-UTRA); Multiplexing and Channel Coding*, ETSI, Sophia Antipolis, France, Feb. 2013.

[15] L. Christodoulou, O. Abdul-Hameed, A. M. Kondoz, and J. Calic, "Adaptive subframe allocation for next generation multimedia delivery over hybrid LTE unicast broadcast," *IEEE Trans. Broadcast.*, vol. 62, no. 3, pp. 540–551, Sep. 2016.

[16] "5G radio access," Ericsson White Paper, Stockholm, Sweden, Tech. Rep. 284 23-3204 Uen, Jun. 2013.

[17] P. Robertson, E. Villebrun, and P. Hoeher, "A comparison of optimal and sub-optimal MAP decoding algorithms operating in the log domain," in *Proc. IEEE Int. Conf. Commun.*, vol. 2. Seattle, WA, USA, Jun. 1995, pp. 1009–1013.

[18] D. Vogrig *et al.*, "A 0.35-μm CMOS analog turbo decoder for the 40-bit rate 1/3 UMTS channel code," *IEEE J. Solid-State Circuits*, vol. 40, no. 3, pp. 753–762, Mar. 2005.

[19] Q. T. Dong, M. Arzel, C. J. Jego, and W. J. Gross, "Stochastic decoding of turbo codes," *IEEE Trans. Signal Process.*, vol. 58, no. 12, pp. 6421–6425, Dec. 2010.

[20] Manjunatha K N ,Lohith Kumar H G , "Design and Performance analysis of a 3GPP LTE/LTE-Advance turbo decoder using software reference models", *International Journal of Scientific & Engineering Research*, Volume 2, Issue 7, July-2011, ISSN 2229-5518.

[21] Manjunatha K N, Kiran B, "Design and ASIC Implementation of a 3GPP LTE Advance Turbo Encoder and Turbo Decoder", *International Journal of Engineering Research and Applications* (IJERA), Vol. 2, July-August 2012, Issue 4, pp 006-010.