**Design of Low Density Parity Check Decoder for WiMAX and FPGA Realization**

M.K Bharadwaj*1, Ch.Phani Teja*2, K.S ROY*3

*1Electronics and Communications Engineering, K.L University
*2Electronics and Communications Engineering, K.L University

**Abstract**

Error detection and correction is one of the most important blocks of a communication system. In order to achieve reliable communication over noisy channels with low power consumption, error correction codes are used for error correction codes. Parity check codes low density are the most recent error correcting codes and are increasingly popular because of their excellent performance. As the performance of the codes is only fraction dB away from the boundaries of the channel capacity of these codes have become very popular and have become strong competitors to turbo codes.

The emphasis in this work is the iterative decoding algorithms for LDPC codes for WiMAX standard and their hardware implementations. A software reference model is designed to perform LDPC codes with Min-Sum, Sum Regular algorithms and other codes rate decoding the decoding in IEEE802.16e standard. A modified version of Min-Sum algorithm chosen for hardware modeling of the LDPC decoder based on the performance results of the simulation software reference model. A GUI is designed to provide all algorithms, frame lengths and code rates together with a comparison of the performance and make records of the results easier. Hardware Models of check node and variable node and the switching network is designed to build a semi-parallel decoder in Verilog for frame length of 576 and 2304. An efficient semi-parallel decoder for LDPC codes is designed to optimize hardware implemented in Verilog FPGA.

Comparison of the outputs of software reference model for different algorithms clearly the performance loss in Min-Sum algorithm with reduced complexity. The hardware model as implemented using Min-Sum algorithm is synthesized for implementation on Xilinx FPGA family Virtex2 XC2V40 device with input clock rate of 50MHz. The hardware design can handle a maximum frequency of 171.939MHz with a maximum delay of 4.840ns. Total equivalent gate count for the design is 8089 ports. The estimated Static power consumption of all logic and memory units in the design is about 333mW.

**Key Words:** FPGA, Linear Block Codes, Tanner Graph, Sparse Matrix, Min-Sum Algorithm, Frame Length, Code Rate

**Abbreviations**

BER : Bit Error Rate
BP : Belief Propagation
BS : Bank Selector
BWA : Broadband Wireless Access
CN : Check Node
ECC : Error Correcting Codes
LDPC : Low Density Parity Check
LLR : Log Likelihood Ratio
LP : Location Pointer
MAN : Metropolitan Area Network
SNR : Signal to Noise Ratio
VN : Variable Node
WiMAX : Worldwide Interoperability for Microwave Access

**I. INTRODUCTION**

To achieve over noisy channels, with low power consumption, error correction codes are used. Reliable communication Error correction codes add redundancy in the transmitted data stream so that the receiver can detect and correct errors. LDPC codes are the most recent error correction codes are becoming increasingly popular because of their excellent performance. LDPC codes linear block codes originally by Gallager in the early 1960s [1]. The parity check matrix is sparse with a low density of ones. The uniform column and row weight in a parity check matrix other codes are referred to as irregular. If the performance of these codes is only a fraction of dB away from the capacity limits of the channel, these codes have become very popular and have become strong competitors for the turbo codes. This popularity has led to proposals for the use of this code for different applications and standards both wireless and wired, examples include the WiMAX standard. WiMAX has become synonymous with the IEEE 802.16 standard family, an emerging standard for fixed and mobile MAN (Metropolitan Area Network). The latest 802.16e amendment supports mobility vehicles speeds of around 120 km / h in the WiMAX system. The 802.16 standard supports high data rates up to about 70 Mbps, with a variety of channels encoding options. The mandatory scheme is convolutional and Reed-Solomon code. Convolutional turbo codes, turbo product codes and LDPC codes in 802.16e are optional. These optional additional codes can be used to ensure robustness in extreme fading.

Belief Propagation (BP) algorithm also sum-product algorithm is the most popular LDPC decoding algorithm. The message passing algorithm works on the Tanner...
The Tanner graph is a bipartite graph that represents the parity check matrix, and is composed of two nodes referred to as variable nodes and check nodes. The message passing algorithm is an iterative algorithm that starts with the initialization and during each iteration, message passing takes place of each check node to all the adjacent variable nodes marking the completion of the first half of iteration and then in the second half, of each variable node adjacent check nodes. The decoding performance can be improved by repeated complexity of check node processing, flexibility to accommodate different code rates and frame lengths, etc. [6]. Apart from these architectural limitations, there are several challenges in the hardware implementation, such as processing speed, clock frequency, memory, gate count and efficient re-configurability.

There are many algorithms developed in recent years based on the regular sum-product algorithm and change it to the decoding procedure to reduce simplify. Hardware complexity Reduction of hardware complexity is directly linked to the check node processor and the edge memory design requirements. A popular algorithm in this direction is the Min-Sum algorithm, including the compensated version [2], [3], [6], [7]. In this work, software and hardware implementation of LDPC decoder for WiMAX standard Min-Sum algorithm with a semi-parallel architecture is executed. The 19-frame lengths defined in the IEEE802.16e standard varies from 576 to 2304. Software model is designed for frame lengths of 576 1440 and 2304 in which the standardized lowest, intermediate and maximum lengths for all code rates. Hardware model is designed and implemented for frame lengths of 576 and 2304 for code rate of 1/2 on FPGA.

![Figure 1 Block diagram of LDPC decoder for WiMAX](image)

**II. LDPC DECODER SOFTWARE MODEL**

In the first phase of this work, is a software developed to reference the performance of different iterations of the message, passing along the edges of the graph with some stopping criterion after a few iterations, [4], [5]. Powerful research and efforts are put in VLSI realization of LDPC decoders and encoders because of the importance of error control coding in modern communication systems. Concentration on the decoders, one can see the proposals of the various structures on various platforms [5],[6],[7]. The different aspects in the execution, serial, parallel, semi-parallel architectures, edge memory requirement, processing decoding algorithms analyze different code rates and frame lengths of LDPC codes defined in the WiMAX standard. The reference model is designed to replicate the Tanner graph functionality with check node processor, variable node processor and permuter. Figure 1 shows a modified version of the standard model of LDPC decoder architecture. The decoder operates iteratively the input data to a collection of nodes involved by selectively communicating data and decode again. The processing in the check node processor, implemented during the iterations is given by equation (1) which is the sum of the two received from the variable nodes along the product of their characters minimum values.

\[ E_{n,m}^{(i)} = \prod_{n \in N(m)/n} \text{sgn}(T_{n,m}^{(i)}) \times \Phi[T_{n,m}^{(i)}] \]  

(1)

And where, m is the extrinsic information check out the variable node to node processors. Equation (2) defines the functionality of the variable node processor that the sum of the intrinsic value and extrinsic values obtained new calculates the check node processor [2].

\[ T_{n,m}^{(i)} = I_n + \sum_{m \in M(n)/m} E_{n,m}^{(i)} \]  

(2)

Where Tn, m is the intrinsic information from the check node to variable node processors. Permuter network plays a very important role between the nodes in architecture. The functionality of the permuter switching network is similar to that of a telephone that connects two nodes based on the address. A control word containing the address of a node is used to move to connect to the end node based on the control. The model is implemented for code rates of 1/2, 2/3A, 2/3B, 3/4A, 3/4B and frame lengths of 576, 1440 and 2304. The model is designed with regular belief propagation, min-sum and min-star algorithms built. To the analysis of the results a known sequence of zeros is supplied as an input to the decoder. This helps to find out the difference in decoding and error correction of the same for a constant number of iterations. The snapshot of GUI software developed for the reference model with all of the above features is shown in Figure 2. The developed GUI allows the user to feed the required code rates, selection algorithm, frame length, number of iterations and the number of frames to be processed simultaneously.
maximum of 50 iterations is considered for implementation. The outputs obtained the frame error rate and bit error rate.

Table 1 shows the tabulated values obtained from the reference software. It can be observed from the results obtained with more number of iterations, the bit error rate decreases. The bit error rate is obtained for 5-50 repetitions frame lengths of 576 and 2304 that the least and the maximum length defined in WiMAX standard.

Table 2.2 summarizes the simulation results of the three algorithms considered. The performance of Min-Sum algorithm known theoretical loss in performance has been demonstrated with the results obtained. This loss is due to the changes that simplify the computational complexity. On the basis of the simulation results obtained for a variety of algorithms, frame lengths and code rates of the software reference model, was Min-Sum algorithm chosen for the implementation in order to optimize FPGA. Hardware requirements The advantage of using the min sum algorithm is reduced edge calculations and memory requirements for hardware implementation.

III. LDPC DECODER HARDWARE MODEL
The hardware model of Low Density Parity Check Decoder for WiMAX unit was designed for the Xilinx FPGA implementation. The hardware architecture is designed to make the calculations between the nodes and reduce memory requirement.

3.1 Design of hardware model architecture
The hardware architecture model of Low Density Parity Check Decoder for WiMAX unit was designed to optimize the hardware and achieve higher operational speed while providing flexibility for further hardware modifications. A compromise between speed and reduced hardware to achieve, complexity is a semi-parallel architecture designed and implemented. The following are required for the designed LDPC decoder unit the IO ports.

![Figure 4 Block diagram of hardware architecture of LDPC decoder for WiMAX](image)

The architecture shown in Figure 4 has memory banks in parallel for variable nodes and check nodes, check node and variable node units, bank selector, location pointer, and switching network unit. The cyclical behavior of the codes is used to obtain. The reconfigurable architecture. The decoder architecture shown in Figure 4 is for code rate 1/2, and for frame lengths of 576 and 2304. The configurable for different lengths and different price can be obtained with a slight modification of the number of memory banks again and the number of addresses stored in each bank. As shown in Figure 4, the architecture consists of a selector bank and the location pointer is automatically initialized by the Tanner graph connectivity. The input of the selector bank location pointer is obtained by the mathematical expressions (3) and (4) resp

\[
LP_c = \frac{[r \mod (I, z)]}{z}
\]  

\[
LP_v = \frac{[C \mod (C, z)]}{z}
\]

Where \(I\) is the address of the variable node by a specific parity check node. The data can be carried out by selecting a parity-check node in the group to be worked. With this, the 0 bank for each processor is exactly the opposite bank in Figure 4 and the remaining is counted down in a circular fashion. Further, in Equation (3) \(mod (i, z)\) indicates the selector bank value. Need all the various registry entries is provided. Rates for different Equation (4) gives the value of the pointer location address where \(C\) is the check-node address and \(mod (C, z)\) is the bank selector value.

For the WiMAX standard codes, the addresses of the neighbors of a control node as well as that of a variable in relation to the frequency-dependent node parallelization factor \(z\) for both the variable nodes and the check nodes, in a similar pattern was observed. For \(N = 2304\) and rate 1/2, this factor is 96. For a set of \(z\) nodes from the 0th node to 1 - \(z\) node, there would be an increase of the corresponding node connected in general, for example, if we have a control node variable nodes connectivity for each set of \(z\) control nodes, the address values of the variable nodes is generally increased. So, what the variable nodes part is concerned, for each variable \(z\) node address of the control node is incremented. The hardware modeling requires equal number of nodes on both sides although only one half of the control nodes are connected to the code rate 1/2. This is achieved by the addressing scheme with pre-calculated addresses are stored in the lookup table.

### 3.2 Check node processing

The basic functionality of the check node processor is defined by the equation (1). The internal blocks are shown in Figure 5. Depending on the addresses that the check node processor retrieves the values of the variable...
node memory banks. Location by the pointer. All processors of the same address location indication that the location of the variable node in the memory bank accounts.

Control for 1152 nodes with 96 processors, the check node processor 12 runs once the updation of all check nodes in that particular iteration rounds. This means that Bank selectors 12 and pointer location vectors are necessary to specify the variable node addresses. The number of items in each vector is equal to the degree of the check node. When it reaches the counter value of the degree of the node, the selector bank pointer location to be loaded with a new series of items. The check node processor sequentially retrieves a value in a time and performs the calculation of these values, and stores the results that are required in the variable node updation of the current iteration and the check node updation of the following iteration. All the P number of check node processors gets their input values and performs calculations on them parallel.

The calculation of each check node processor includes determination of the first minimum (min1) and the second minimum (min2) values of the absolute input value, the storage of the address of the first minimum, the product of the signals of all the input values, and individual signs values of its input. These results are obtained after the check node processing is stored in RAM check node. In this manner, after the completion of the processing of the 96 processors 12 times, the check node processors is deactivated which the completion of the control processing node indicating that iteration. After completion of the check node processing, the variable node processors activated

3.3 Variable node processing

The basic functionality of the variable node processor is defined by equation (2). The internal blocks of variable node processor shown Figure 6, the process of generating the address and retrieving the corresponding values of check nodes the variable node processor is similar to the process performed in the check node processor. The bank selector and location pointer used to control the addresses of the nodes are connected to variable node indicate. To make the process of updating the variable nodes 2304 in an iteration round, the variable node processors 96 to 24 times an operation corresponding input values. The corresponding input values are given by the bank selector and pointer location. 96 vectors of the bank and selector 24 instead pointers to complete the updating of the variable node processors. The number of items in each vector is equal to the degree of the variable node.

The variable node processor retrieves values of control nodes one by one and adds these values to the intrinsic information and stores the result. This value is required for updating the check nodes in the next iteration. The value achieved ROM control node will be + or - min1 min1 min2 or + or - min2. The first variable node processor has the min1 address that is stored in one of the six memory locations of the control node. Min1 address is compared with the address of the variable node that is updated and if it matches, it retrieves the value min2. Otherwise, it takes the value min1. The sign of min1 or min2 is obtained by dividing the total plate with its own sign multiply. The inputs collected by the variable node processor be added intrinsic value that is retrieved from the memory banks Vin and the result is stored in the variable node memory bank.

3.4 Switching network

The basic functionality of the switching network is also called as the permuter provide the interconnection
between the nodes during the iterative decoding process. The permuter is the most complex hardware of the decoder unit in which a large number of connections between the internal blocks of the permuter [8], [9].

![Figure 7 Functional diagram of permuter network](image)

Required connectivity between two nodes is controlled by means of addresses which are stored in the look up tables in advance. The switching network in Figure 7, allows for the coupling mechanism. The permuter functionality is realized by means of a barrel shifter network with 96 inputs and 96 outputs. On the basis of the values 96 is retrieved from the memory banks and 96 are in place the pointer values are applied to inputs of the switching network. The switching network is shifting these values, the appropriate processing elements according to the contents of the bank selectors. The successive input values for the soil working members can be obtained by increasing the address of the bank selector and the location pointer lookup tables.

3.5 Simulation of hardware model

The simulation of the designed LDPC decoder unit is carried out with the aid of ModelSim simulator. Each of the sub-blocks were tested to verify the functionality and has been incorporated in order to check. The general operation of the decoder. The basic block s considered is:

1. Check Node processor
2. Variable node processor
3. Permuter network

Each of these sub-units is the core of the decoder that the Tanner graph. Test machines were developed to verify the functionality of the individual units. The simulation and synthesis results are discussed in this section. Figure 8 shows the simulation results of permuter network. A simpler approach is used to test designed by high all of the address lines and supplying the data to the network on a channel at a time.

![Figure 8 Simulation result of permuter network](image)

Figure 8 Simulation result of permuter network

Figure 9 and 10 shows the synthesis of the results permuter network. One can see that the permuter only take the most resources for the selected FPGA. This is because the data width and number of input and output ports of the permuter. The permuter has 96 inputs and outputs each 60 bits wide.

![Device utilization summary: Spartan 3E](image)

<table>
<thead>
<tr>
<th>Device utilization summary: Spartan 3E</th>
</tr>
</thead>
<tbody>
<tr>
<td>Selected Device: 3s008256-4</td>
</tr>
<tr>
<td>Number of Slices: 256/79 out of 1920 1337% (*)</td>
</tr>
<tr>
<td>Number of Slice Flip Flops: 446/57 out of 3940 112/5% (*)</td>
</tr>
<tr>
<td>Number of input LUTs: 40325 out of 3940 1050% (*)</td>
</tr>
<tr>
<td>Number of bonded IOBs: 11531 out of 173 66634% (*)</td>
</tr>
<tr>
<td>Number of OCLs: 1 out of 0 12%</td>
</tr>
<tr>
<td>Minimum period: 7.152ns (Maximum Frequency: 139.880MHz)</td>
</tr>
<tr>
<td>Maximum input arrival time before clock: 1056.7ns</td>
</tr>
<tr>
<td>Maximum output required time after clock: 7.165ns</td>
</tr>
</tbody>
</table>

Figure 9 Synthesis report of permuter network for spartan3E

![Device utilization summary: Vertex 5](image)

<table>
<thead>
<tr>
<th>Device utilization summary: Vertex 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Selected Device: 5v5220f1760-3</td>
</tr>
<tr>
<td>Number of Slices: 405028 out of 138240 32%</td>
</tr>
<tr>
<td>Number of Slice LUTs: 40325 out of 138240 29%</td>
</tr>
<tr>
<td>Number used as LUTs: 40325 out of 138240 29%</td>
</tr>
<tr>
<td>Number of LUT/Flop pairs: 405028 out of 138240 32%</td>
</tr>
<tr>
<td>Number of bonded IOBs: 11531 out of 800 1441%</td>
</tr>
<tr>
<td>Minimum period: 1.299ns (Maximum Frequency: 775.735MHz)</td>
</tr>
<tr>
<td>Maximum input arrival time before clock: 1.526ns</td>
</tr>
<tr>
<td>Maximum output required time after clock: 2.517ns</td>
</tr>
</tbody>
</table>

Figure 10 Synthesis report of permuter network for vertex5

Figures 9 and 10 show a summary of the resources used by the network for the least two permuter and of the maximum frame length is defined in the IEEE802.16e standard. Figure 11 shows the simulation result of check node processor. The inputs to the check node processor clock, reset, state, treats from pointer, bank selector, and 60 bit input of memory banks.
Simulation result of variable node processor, as shown in Figure 12. The inputs to the variable node processor clock, reset state and 60 bit input memory banks. The output generated by the sum or the difference of the intrinsic and the new input values of the check node processor according to the sign of the inputs.

LDPC Decoder unit is synthesized and implemented using Xilinx ISE 8.2i. The following configuration is used for the implementation of the design:

- **Product Category:** General Purpose
- **Family:** Virtex2
- **Device:** XC2V40
- **Speed Grade:** -5

The upper level diagram obtained after synthesis is shown in Figure 13.

The inputs and outputs of the design referring to Figure 13 are:

**Inputs:**
- \( P_{\text{clk}}_{50\text{mhz}} \) – Global clock
- \( P_{\text{reset}}_{\text{n}} \) – Global reset
- \( P_{\text{wr}}_{\text{n}} \) – Global write / read select
- \( P_{\text{datain}}_{\text{channel}(15:0)} \) – Global input
- \( \text{Rate}_{\text{sel}}(1:0) \) – Global rate select

**Outputs:**
- \( P_{\text{dataout}}_{\text{decoder}(15:0)} \) – Output of Decoder unit

Figure 14 shows the use of resources summary of the LDPC decoder design for a frame length of 576, and code rate of 1/2. The total gate count of the design proves to be 8069, and an additional 1,776 ports for IOBs is used. The screenshot of the post place and route power report generated using the Xilinx XPower tool for the LDPC decoder block. The Power report obtained the estimate of static power consumption of all logic and memory units in the design, which is about 333mW. The maximum frequency of the design is 171.939MHz. From Timing the report suggests that the LDPC decoder unit design is carried out works on 50MHz onboard global clock. However, the design has a data input delay and output delay of 5.557ns 4.840ns to be taken care of while handing Input / Output data on the device. The draft has no input / output end-to-end combinational path. It can also be seen that the clock signal drives a load of 222 units of apparatus.

![Figure 11 Simulation result of check node processor](image1)

![Figure 12 Simulation result of variable node processor](image2)

![Figure 13 Top level block diagram of LDPC decoder](image3)

![Figure 14 Synthesis report of LDPC decoder](image4)


IV. LIMITATIONS

The software and hardware models of Low Density Parity Check have few restrictions. Decoder unit designed They are as follows:

- The software reference model of Low Density Parity Check Decoder is designed for only three of the 19 frame lengths defined in WiMAX standard.
- The architecture can be extended further designed to meet. To other standards
- The hardware model of Low Density Parity Check decoder is designed more input from the source in a sequential manner that leads to the system latency.
- Hardware model is designed to be one of six standardized code speeds forward. IEEE802.16e
- Hardware model can be made to other frame lengths suitable if a memory management system is introduced to design flexible.

V. CONCLUSIONS

In this work, an efficient VLSI architecture for decoder of Low Density Parity Check codes is presented. The architecture is useful for VLSI implementation in a variety of applications such as WiMAX and DVB-S2. The architecture is implemented in Matlab rates for all the code in the WiMAX standard. The frame lengths considered to be 576, 1440 and 2304. The algorithms used are regular, star min and min sum. It is proved that min sum algorithm is best suited for modeling hardware between the three algorithms due to reduced calculations and edge. The performance loss of the Min-Sum algorithm shows very less by comparing the results of the software model EbNo = 2.5. The hardware model designed decodes the LDPC code block length of 576 and 2304, and code rate of 1/2. The design sub blocks are modeled in Verilog language and they are synthesizable. The designed architecture operates at a speed of 171.939 MHz using 333mW power for frame length of 576. It should should be made to efficient integration of the sub-modules and extension of the system to use different block lengths and for different applications. However, some work done.

REFERENCES