# Abstract

The development of wireless technology changing rapidly because along with the need of data services is very large and fast (high data rate) and has a good Quality of Service. one way to improve the Quality of Service is to reduce the error rate information received at the receiver, One of them is by using channel coding. Currently, channel coding technique is close to perfect as it nears the Shannon theorema on the large amount of data is LDPC. An important aspect from LDPC encoding technique is how the implementation on hardware.

Steps that must be done in implementing a system is to design a system, then perform synthesis, and then implemented in hardware. Hardware used in this thesis were Spartran 3E FPGA. While the step for designing LDPC are two steps, encoding design step and LDPC decoding design. On ecoding design used lower triangular algorithms , while decoding design use bit flipping algorithms. Parity check matrix used is Regular LDPC with size 6x12 and its code rate is ½.

From the simulation results, the correction capability from this Flipping bit can only be corrected for the 1-bit error correction with capability correction by 100%. If the frequency of the use of 100 MHz FPGA then worth clock 1 worth 10 ns, if in the one process requires 28 clock delay, then the whole period is 280 ns. That impact bit rate 21,4 Mbps and the frequency 3,57

MHz. The parallel working on system make the bit rate are big and the periode was small. From the synthesis result, the design was copared with the other design, it must be good after because the result show it consume a lot of memory and use a long of periode than other design that can process a more larger data. After the design implemented, resource utilization like number of slices 0%, number of flip flop slices 0%, number of LUT 4 input is 0% , number of IOB is 3%, and number of GCLK is 3%. It impact that the architecture design can be implemented on Virtex-4 FPGA because of the resource consumed under 100%

Keywords *:* LDPC, *lower triangular*, *bit flipping* , FPGA Virtex-4