Asian Journal of Applied Science and Technology (AJAST)
Volume 1, Issue 3, Pages 153-156, April 2017

Design of Digital ZigBee Transmitter for IoT Application

A.Helen Joyce[1] and M.Saravanan[2]

[1]*UG Scholar, Department of Electronics and Communication Engineering, IFET College of Engineering, Villupuram. India.*
[2]*Associate Professor, Department of Electronics and Communication Engineering, IFET College of Engineering, Villupuram. India.*

Article Received: 24 March 2017 *Article Accepted: 04 April 2017* *Article Published: 07 April 2017*

ABSTRACT

The very short development in the wireless technology within the last few years, it's far concentrated in high speed utility. For IoT packages, the wireless device having minimal bandwidth, much less power consumptions are required for the transmission system. So that the virtual ZigBee transmitter is designed to fulfill the IoT software needs. Alongside that ZigBee offers low price, lengthy battery life. So it is mostly used in energy and control related programs. ZigBee works under following 3 frequency bands 868MHz, 915MHz, and a couple of 2.4GHz. The ZigBee is going to be designed with baud rate of 250kbps and frequency band of 2.4GHz.It includes three blocks, such as error detection block, encoder block, modulation block. The parallel CRC is used for error detection technique. The encoder carries bit-to-symbol, symbol-to-chip blocks. DSSS is used to design symbol-to-chip block. QPSK modulation technique is used within the modulator. To reduce the dimensions of ZigBee and reduce the power requirement for the IoT application ZigBee goes to be designed and simulated by using VHDL programming in xilinx9.1.

Keywords: Parallel CRC, DSSS, QPSK and Bit-to-symbol.

1. INTRODUCTION

The internet of things (IoT) is a computing concept that describes the concept of normal physical items being linked to the net and being able to identify themselves to different devices. The time period is intently identified with RFID because the technique of conversation, even though it can encompass other sensor technology, wireless technologies. It requires low electricity consumption Wi-Fi devices for communication with everyday data rate.

ZigBee uses very low power requirement and better battery life, which makes it extensively useful in monitoring and manipulate programs. Therefore markets as building automation, commercial manage, lighting fixtures in smart houses, non-public fitness care, and business control are best suits.

The entire world is going smart, smart energy grids, clever sensor networks, clever homes, and clever water distribution structures. The most suitable structure for this is IoT. The IoT make use of intelligently related gadgets, and systems to switch facts over a network without requiring human-to-human or human-to-computer interplay.

This venture especially focuses on designing of Zigbee transmitter at the architectural or hardware point of view for the IoT programs. ZigBee transmitter may be designed with the usage of analog additives. The analog devices use bigger additives, and additionally data transmission is extraordinarily no longer correct. So that the ZigBee is going to be designed using digital components. The digital model will permit correct records transmission. Furthermore, electricity deliver necessities of digital devices are much less. Because of these benefits, digital ZigBee transmitter is suitable for IoT programs. The IoT does now not require excessive information quotes, because IoT gadgets in particular use manage signals. Additionally, less power is consumed in Zigbee compared to different tool. On this paper each block of Zigbee transmitter is going to be designed and simulated.

ZigBee transmitter

In this project, we focusing on 2.4 Hz band application which has 16 channels with the spacing of 5MHz. The data rate is saved at 250 Kbps. The ZigBee general employs Direct sequence spread Spectrum (DSSS) technique to keep away from interference. The modulation is accomplished using Quadrature Phase Shift Key (QPSK) modulator, which sum the in-phase signal with a half of cycle delayed quadrature phase signal. The ZigBee transmitter has been designed for the usage of acknowledgment frame format, which is the best MAC sub layer frame layout without Medium access manipulate (MAC) payload. It offers the lively comments from the receiver to sender that the packet has been acquired without errors. Figure 1 provides the data frame format of both PHY and MAC layer.

MAC layer

The MAC layer contains of MAC Header (MHR) and MAC Footer (MFR). The MHR includes MAC frame control and data collection variety, even as MAC Footer (MFR) constitutes frame control Sequence (FCS). Frame Control Sequence is a 16-bit long subject, which gives records about the frame kinds, source and destination addressing modes. Frame sorts imply whether or not the body is beacon type, ACK body, MAC command body and so on. The Data Sequence Number (DSN) is an eight-bit length that is used while sending the data. It acts as a counter those increments itself after every frame. This of MAC layer is given to the payload field of PHY layer.

PHY layer

The PHY Payload which certainly comes from acknowledgment frame (PSDU) is prefixed with Synchronous

Header (SHR) and PHY Header (PHR). The SHR has preamble sequence and start of frame delimiter. The preamble collection is a 32-bit field used for synchronization between transmitter and receiver. It consists of string of 0s and 1s. After detecting subject, the receiver starts synchronization with the incoming data. Begin of body delimiter is an eight-bit long area which indicates starting of the frame. Frame duration is an 8-bit long area, wherein 7-bits suggest length of PSDU and reserved 1-bit suggests if packet is received or not. Table 1 summarizes the bit period and price of every field in PHY layer.

Table 1.Bit length of each field

FIELD	BIT LENGTH
Preamble sequence	32 (set at logic 0)
SFD	8 (11100101)
PHR	8 (10100000)
Frame control	16 (0100010000000000)
Sequence number	8 (10000000)
FCS	16 (from CRC)
Total	88

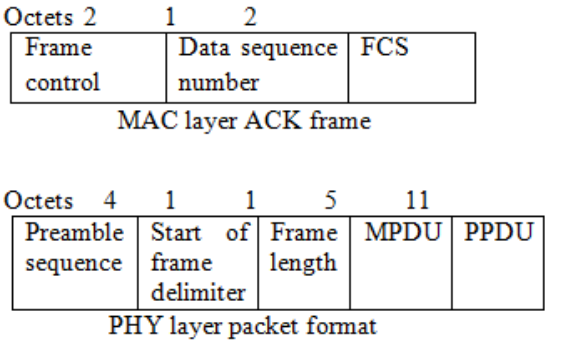

Fig 1. Acknowledgement frame format

Block diagram

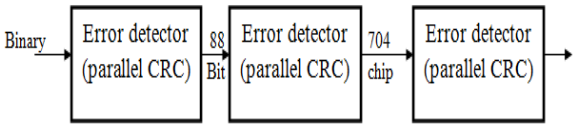

Fig 2.Block diagram

Parallel CRC

There are different techniques for parallel CRC is given as follows.

1. A table-based algorithm for Pipelined CRC calculation.
2. Fast CRC replace
3. F matrix based parallel CRC generation.
4. Unfolding, Retiming and pipelining algorithm

Parallel processing used to increasing the throughput via generating the number of output same time. Retiming used to increasing clock price of circuit with the aid of reducing the computation time of vital direction. The F matrix based totally structure are extra easy and low complex. The F matrix procedure is represented in diagrammatic form in figure 3.

In this implementation, the statistics input can be four bits at one clock pulse if the statistics enter length can be growth so clock pulse could be decreased. It's going to complete in bringing about lesser clock pulse so it will be used for high overall performance. The facts sixty four bit disbursed into 4 blocks. Each consists of sixteen bit. Right here, 4 execution units are used. They may be parallel pipelined with each different and four 32-bit remainders are received. They're xor with each different and got final CRC that is of 32-bit.

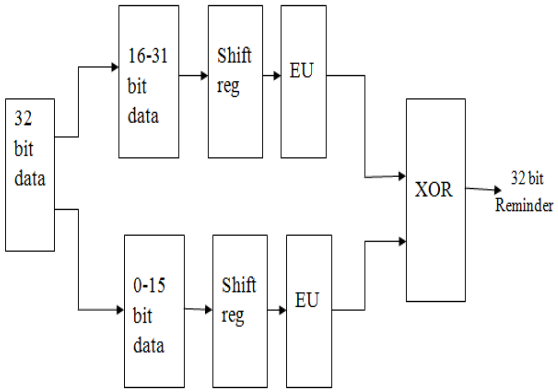

Fig 3.Parallel CRC diagram

Bit-to-symbol

All of the 88 bits from the CRC block is inserted into the bit-to-symbol block. This binary information is mapped into the statistics image. The 4 LSBs (b0,b1,b2,b3) of every octet is mapped into one facts symbol and the 4 MSBs (b4,b5,b6,b7) of each octet is mapped into the subsequent information image. Every octet of PPDU is processed through the bit-to-image block sequentially, beginning with the preamble field and ending with the ultimate octet of the PPDU. For the final result, 22 symbols might be the output of the bit-to-symbol block. The block has five input ports such as the "clk", "data_in", "load", "shift", and "reset_bit". The output ports are classified as "symbol1" till "symbol22" with four bus widths. The frequency utilization is 250 KHz and the input records could be loaded into "data-in". If "reset_bit" is at logic 1, then all the registers on this block may be reset. In any other case, if "load" is at 1, the data from "count_in" might be loaded into "count_out "check in. But, if the "load" is at logic zero, then the procedure will depend upon "shift" situation. If the "shift" is at logic 1, then the data from the "count-out" register may be stored in the "output" register. In any other case, the data within the "output" unit will constantly be at common sense zero. Ultimately, the data from this check in might be divided and shifted to "symbol1" till "symbol22".

Symbol-to-chip

The DSSS is used for the symbol to chip conversion operation. The symbol-to-chip has 27 enter ports: "symbol1_chip" till "symbol22_chip", "reload", "reset_symbol", "shift_symbol", "clk1" and "clk2" and "data_out" as the output port. The frequency for "clk1" and "clk2" are 250 KHz and 2 MHz respectively. The entered data from "symbol_chip" might be shifted to a register categorized as "symbol_2" within "clk1". Then at 'clk2", if "reset_bit" is at common sense 1, then all other registers on

this block could be cleared. Otherwise, if "reload" is at logic 1, every information symbol can be processed right into a 32 chip PN collection. However if the "reload" is at logic 0 then the procedure will depend upon "shift_symbol" condition. If the "shift_symbol" is at 1 the LSB of 32 chip might be shifted to "data_out" and this technique is repeated until 704 chip. In any other case the "data_out' will usually be at common sense zero. The table 2 gives the output sequence for corresponding input symbols.

Table 2. Symbol to chip block table

Data symbol (decimal)	Data symbol (binary)	Chip sequence (c0,c1,….c30,c31)
0	0000	11011001110000110101 001000101110
1	1000	11101101100111000011 010100100010
2	0100	00101110110110011100 001101010010
3	1100	00100010111011011001 110000110101
4	0010	01010010001011101101 100111000011
5	1010	00110101001000101110 110110011100
6	0110	11000011010100100010 111011011001
7	1110	10011100001101010010 000101110110
8	0001	10001100100101100000 111011110111
9	1001	10111000110010010110 000011101111
10	0101	01111011100011001001 011000001111
11	1101	01110111110111000100 010011011000
12	0011	00000111011101111000 110010010110
13	1011	01100000011101111011 100001100100
14	0111	10010110000001111011 111011100011
15	1111	11000100101011000001 111011111011

A Direct Sequence Spread Spectrum (DS-SS) unit spreads the baseband information by manner of directly multiplying the baseband data pulses with a pseudo-noise pulses that is produced through a pseudo-noise generator. A single pulse or symbol of the PN waveform is referred to as a chip.

QPSK modulator

The modulation operation is represented in figure 4.The binary data coming from the symbol-to-chip block obtained in the form of serial data. This serial binary data is given as input to QPSK modulator. It will change that serial data into parallel data and then filtered using low pass filter. Then the in phase and quadrature phase components are obtained by multiplying the data with the signal coming from local oscillator with 90 degree phase shift. Then finally adding both the components we get the modulated data.

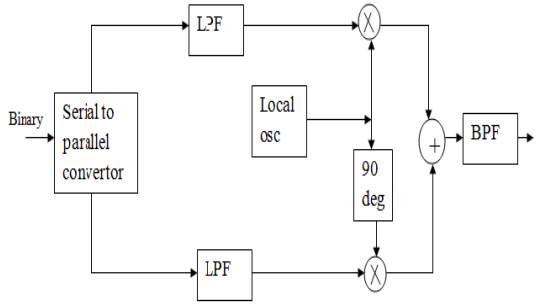

Fig 4. QPSK modulator

2. RESULT AND DISCUSSION

(a) CRC simulation

Input signal is represented by crcin. The signal d [15:0] represents 16 bits CRC values. These 16 bits along with the input 72 bits are the input of the bit-to symbol block.

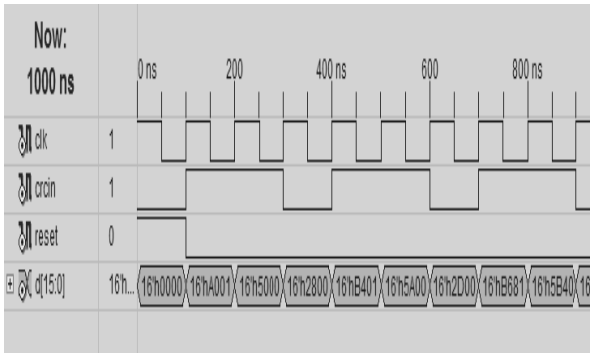

Fig 5.CRC output

(b) Bit-to-symbol simulation

The 88 bits inputs are represented by the signal data [87:0]. In the same figure, different symbol outputs are given. Not all the symbols are shown in the figure. In actual case, there are 22 symbols.

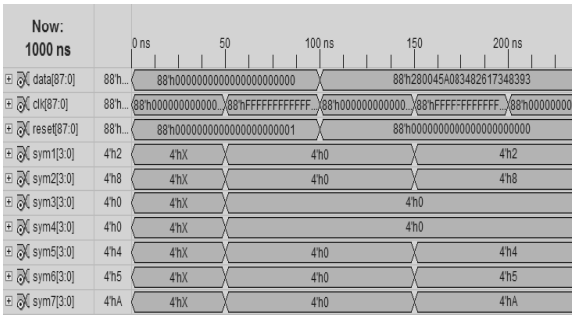

Fig 6. Bit-to-symbol output

(c) Symbol-to-chip simulation

Different chip values are shown according to Table 1. Each symbol from the bit-to-symbol block is mapped into unique chips of 32 bits. The output chips from this block have given to the input of QPSK modulator.

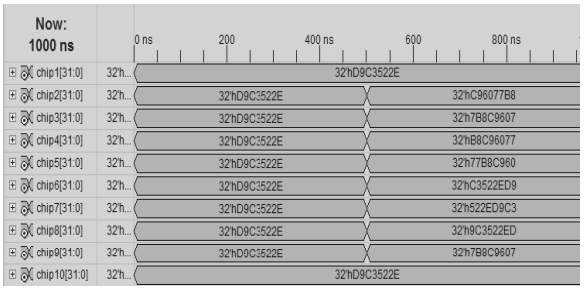

Fig 7. Symbol-to-chip output

(d) QPSK simulation

The signal out gives the simulated output waveform of the QPSK modulator.

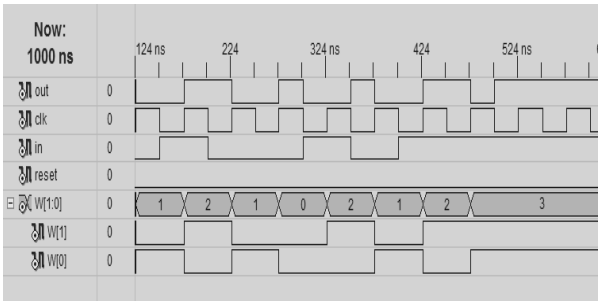

Fig 8. QPSK output

3. FUTURE WORK

This ZigBee is appropriate for low distance applications. In future we must implement the ZigBee for long distance IoT applications. Use O-QPSK modulation to boom the accuracy.

4. CONCLUSION

This paper describes the design of four three blocks of ZigBee transmitter for IoT application. The parallel CRC is used to detect the error, the bit-to-symbol and symbol-to-chip blocks were used to design the encoder block and QPSK modulation is used for the modulation purpose. These blocks were designed and simulated using VHDL programming.

REFERENCES

[1] Boca Raton, "*ZigBee Network Protocols and Applications"*, FL, CRC Press 2014.

[2] Chaitali Tohgaonkar, Prof. Sanjay B. Tembhurne, Prof. Vipin S. Bhure, "*Design of Parallel CRC Generation for High Speed Application"* International Journal of Advanced Research in Computer and Communication Engineering Vol. 4, Issue 6, June 2015.

[3] Digi International (2015) [online]. *Available: http:// www.digi.com*

[4] Lamia Chaari and Lotfi Kamoun, *"Performance Analysis of IEEE 802.15.4/ZigBee Standard under Real Time Constraints",* International Journal of Computer Networks & Communications (IJCNC) Vol.3, No.5, Sep 2011.

[5] Nidhi Patel, Hiren Kathiriya, Arjav Bavarva, *"Wireless Sensor Network Using ZigBee"*, IJRET: International Journal of Research in Engineering and Technology ISSN: 2319-1163 Volume: 02 Issue: 06 | Jun-2013.

[6] R.Divyabharti, Design and Simulation of ZigBee using Verilog, *Information Communication and Embedded Systems*, pp 882-888, Chennai, India, Feb 2013.

[7] S.Benjamin Arul, International Journal of Scientific & Engineering Research, Volume 5, Issue 12, December-2014 133, http://www.ijser.org "*Wireless Home Automation System Using Zigbee"*.

[8] Salleh, M. K. Ismail, N. R.Mohamad, M. Z. A Abd Aziz, M. A. Othman, M. H. Misran *"Development of Greenhouse Monitoring using Wireless Sensor Network through ZigBee Technology"*, International Journal of Engineering Science Invention, Volume 2 Issue 7 ǁ July. 2013 ǁ PP.06-12.

[9] Sangeeta Singh, S. Sujana, I. Babu, K. Latha, *"VLSI Implementation of Parallel CRC Using Pipelining, Unfolding and Retiming"*, IOSR Journal of VLSI and Signal Processing (IOSR-JVSP) Volume 2, Issue 5 (May. – Jun. 2013), PP 66-72 e-ISSN: 2319 – 4200, p-ISSN No. : 2319 – 4197.

[10] Somya Goel, Dr. Ranjit Singh *"VHDL Based Design and Implementation of Zigbee Trans-receiver on FPGA",* ISSN (Print) : 2320 – 9798 ISSN (Online): 2320 – 9801 International Journal of Innovative Research in Computer and Communication Engineering *Vol. 1, Issue 2, April 2013*.

[11] Yash Vardhan Varshney, Anil Kumar Sharma, "*Design & Simulation of Zigbee Transceiver System Using Matlab",* International Journal of Engineering Trends and Technology (IJETT)-Volume4, Issue4- April 2013 ISSN: 2231-5381 http://www.ijettjournal.org Page 1316.