ISSN 2278-3091
Volume 8, No.1.6, 2019
International Journal of Advanced Trends in Computer Science and Engineering
Available Online at http://www.warse.org/IJATCSE/static/pdf/file/ijatcse4781.62019.pdf
https://doi.org/10.30534/ijatcse/2019/4781.62019

Low power Multicarrier- Code Division Multiple Access Receiver on Field Programmable Gate Array

Siti Lailatul Mohd Hassan[1], Nasri Sulaiman[2], Azilah Saparon[3],Yusnani Mohd Yussoff[4], Haslina Jaafar[5,]
[1,3,4] Faculty of Electrical Engineering, Universiti Teknologi MARA,40450 Shah Alam, Malaysia,
[1]sitilailatul@uitm.edu.my, [3]azilah574@uitm.edu.my, [4]yusna233@uitm.edu.my
[2,5]Faculty of Engineering, Universiti Putra Malaysia, 43400 Serdang, Malaysia,
[2]nasrisulaiman@upm.ed.my, [5]jhaslina@upm.edu.my

ABSTRACT

This paper presents a low power multi-carrier code division multiple access (MC-CDMA) receiver on field-programable gate array (FPGA). Most of the wireless application nowadays such as wireless sensor networks, portable computation and many more require a low power design. Time-division multiple access (TDMA) is used in most wireless receivers are not very efficient since they adopt scheduling technique. The first objective of this paper is to design and verify a low power MC-CDMA receiver and the second objective is to implement the MC-CDMA receiver on FPGA. MC-CDMA act as a processor with the ability to process transmit or receive data simultaneously over a single communication channel. The MC-CDMA design in this paper consists of pipelined FFT and combiner. The primary purpose of pipelined FFT plus combiner module in this research is to execute the instruction on communication (data send and receive) and self-organization. Besides these two modules, there is a memory for temporarily storing the data and an internal clock, among other things. To accomplish these, the designs have been carried out using Verilog coding in Modelsim software, and the design verifications are done through Matlab. The design implementation is via Quartus and on DE2-115 Altera FPGA board. The functionality analyses have been carried out on simulation, and the hardware implementation of the MC-CDMA receiver is tested. Both simulation and hardware execution are successful where the receiver received and displayed the output accordingly. MC-CDMA achieves 39.13mW total power consumption.

Key words: combiner, FPGA, MC-CDMA, pipelined FFT, receiver

1. INTRODUCTION

MC-CDMA can be described as a form of CDMA or spread spectrum, but instead of time domain as in Direct Sequence CDMA (DS-CDMA) [1], it applies the spreading in the frequency domain. But somehow, MC-CDMA is a form of DS-CDMA, but after spreading, fast Fourier transform (FFT) is performed. MC-CDMA also is known as OFDM-CDMA, a combination of OFDM and CDMA [2]. Specifically, MC-CDMA is a modulation method that uses multi-carrier transmission. In this paper, the MC-CDMA receiver contains two main blocks, FFT block and Combiner block. These block functions are to demodulate the OFDM signals [3] and to equalises the signal and separates the coded users, respectively.

The current wireless receiver[4] uses TDMA[5] for its receiver. With TDMA, it needs a scheduling technique for data communication. Scheduling [6] is a process where all sensor nodes are given their channels and times slots to communicated. It is a challenging process as each node have different path length to the destination and disparate channel environments. Efficient scheduling becomes crucial when multiple channels and time slots are utilized.[7] Three main problem arises with scheduling are as follows: (1) the communication resources are shared, (2) during transmission, signals were interference with each other and (3) some of the data may be lost due to impairments, such as fading and attenuation. Other problems related to scheduling are hidden nodes in the network, run time error and real-time data communication [8].

This paper aims to have efficient low power consumption MC-CDMA receiver with FPGA[9] implementation to eliminate problems caused by the TDMA receiver and low power requirement for wireless applications[10].

2. MC-CDMA TRANSMITTER AND RECEIVER

Before MC-CDMA is introduced for multiple users, there is a method called DS-CDMA used widely. DS-CDMA method share spectrum among numerous simultaneous users. Using a RAKE receiver, DS-CDMA can exploit frequency diversity [11]. However, in a dispersive multipath channel, only with highly sophisticated interference cancellation techniques are used, DS-CDMA with a spread factor N can accommodate N simultaneous users, which becomes the disadvantage of DS-CDMA as it is challenging to implement. MC-CDMA is more comfortable to apply, and it can handle N simultaneous users with good bit-error-rate (BER)[12], using standard receiver techniques [3]. As mention earlier,

MC-CDMA is a hybrid of OFDM[13] and CDMA. OFDM is a special form of multi-carrier modulation, suited for transmission over a dispersive channel. OFDM typically applies coding to avoid excessive bit errors on subcarrier needed is larger than the number of bits or symbols transmitted simultaneously. MC-CDMA applies the OFDM but replaces the encoder in OFDM with *NxN* matrix operation, improving the BER.

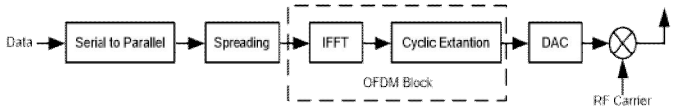

Figure 1: Block diagram of low power MC-CDMA transmitter architecture

Figure 1 shows the block diagram of low power MC-CDMA transmitter architecture. MC-CDMA transmitter consists of a serial to parallel converter, spreading unit, inverse FFT(IFFT) unit [14], cyclic extension [2] and digital-to-analog converter (DAC) [15]. First, serial to parallel converter splits the data stream to be transmitted into N parallel streams to decrease the data rate. Then, the data streams are spreads by spreading unit using a given spreading code or chip in the frequency domain. Next, each of the data streams is modulated by the IFFT unit with different subcarrier [16]. After that, if the kth chip of the spreading code for user u is defined as $c(k,u) \epsilon -1,+1$, then the transmitted baseband signal for the m^{th} data symbol $b(m)$ is as shown in equation 1 below:

$$x(n) = \sum_{n=0}^{N-1} e^{\frac{j2\pi kn}{N}} c(k,u)b(m) \qquad (1)$$

To remove the intersymbol interference (ISI) [3], the baseband signal is the cyclically extended by more than the channel delay spread. Finally, the resulting symbol is passed through DAC before upconversion to the high-frequency RF carrier.

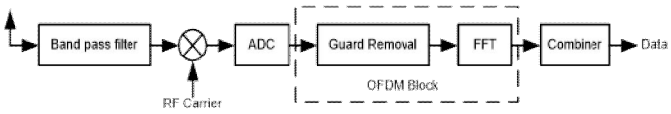

Figure 0: MC-CDMA receiver block diagram

The MC-CDMA receiver block diagram [17] is shown in Figure 2. In MC-CDMA receiver, two main systems blocks are the FFT and combiner. A guard removal places before the FFT block to select the portion of the signal that is free from ISI. FFT block then demodulates the sub-carrier. At the FFT output, the channel effect of a multipath channel $h(n)$ is narrowband for each sub-carrier, $H(k)$, and therefore equalization and de-spreading can be combined into a single combining operation to estimate the transmitted data bit. The combining operation can be represented by equation 2 if the output of FFT at frequency bin k is defined as $Y(k)$.

$$x_{rec}(n) = sign(\sum_{k=0}^{N-1} \Re(c(k,u)A(k)Y(k))) \qquad (2)$$

Where $c(k,u)$ is the corresponding CDMA chip, and $A(k)$ is the equalizer coefficient. The combiner block can be realised by setting $A(k)$ by equation 3 for the minimum mean square error (MMSE) [18]. λ in the equation represent the parameter-dependent upon the signal to noise level and the number of users.

$$A(k) = H*(k)/(|H(k)|^2 + \lambda) \qquad (3)$$

The design of FFT focuses on how to proficiently map the FFT algorithms [19], as shown in equation 4, to the hardware to accommodate the serial input for computation. Some of the architecture commonly used are pipelined, memory, parallel and many more. Pipelined has widely adopted architecture for FFT [20]. The reason why pipelined is a very popular choice is that it has the advantage of parallelism and pipelining making the architecture very fast.[21]

$$\mathrm{X[k]} = \sum_{n=0}^{N-1} x[n]W_N^{nk} \qquad (4)$$

where k= 0,1, 2,…,$N-1$ and W_N is given by

$$W_N = e^{-j2\pi/N} \qquad (5)$$

Combiner module[17] in MC-CDMA act as a signal equalizer and code user separator. Combiner in this thesis uses MMSE detection. As signal equalizer, the combiner is placed in the receiver to reverse the distortion occurred during transmission through a channel. When a channel has been equalized, the frequency domain attributes of the signal at output is expected to be the same at the input.

3. METHODOLOGY

This research can be divided into four-part, Matlab modelling, Verilog design, hardware (FPGA) implementation and power analysis. Further discussion on these research methodology steps are discussed below.

3.1 MC-CDMA receiver Matlab modelling

To model the MC-CDMA transmitter, each of the transmission data is assumed to be in the frame. Each of the frames contains several symbols. The first symbol in each frame is called a pilot or training symbol. Shown in Figure 3 is coding flow in Matlab for MC-CDMA transmit and receive. The first step is to create a pilot or training symbol creation in the transmitter. This pilot symbol function is as an indication for the receiver to calculate the impulse response of the channel. It is assumed that the channel transfer function remained fixed between pilot symbols, and it is a slow fading channel. To estimate the data symbol at receiver, the channel

transfer function is used. After creating the pilot symbol, the transmitter module generated sets of random number for transmission. Next step is to transmit the pilot symbol and data symbol to the receiver part. Then, the receiver prepared to receive a data symbol from the transmitter. And lastly, the data are received by the receiver part. [22]

Results from this Matlab modeling are used as benchmark and verifying purpose for Verilog design and hardware implementation.

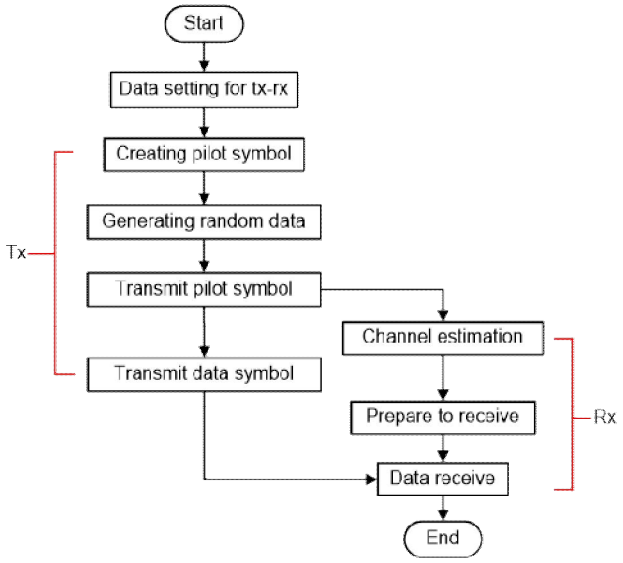

Figure 0: MC-CDMA transmitter and receiver modelling in Matlab

2.2 MC-CDMA receiver Verilog design

Shown in Figure 4 is low power MC-CDMA receiver consists of a pipelined FFT processor and a combiner. Pipelined FFT processor function is to extract the frequency elements of the received signal, while the combiner function is for de-spreading and equalisation.

Pipelined FFT and combiner modules were designed using Verilog coding[23] in Modelsim software simulation. Figure 5. shows the general flow in designing pipelined FFT, starting with designing the sub-module in pipelined FFT such as data buffer, rotator and shifter. Each of this sub-module is compiles for errors. When no error detected, these sub-modules are integrated. Lastly, the pipelined FFT module is compiled for error detection if any. The same design flow is used for the combiner module.

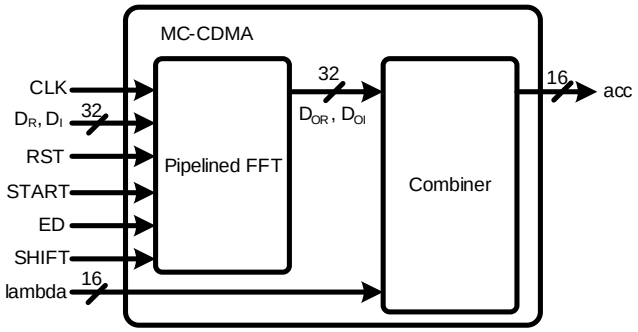

Figure 4: Low power MC-CDMA receiver module

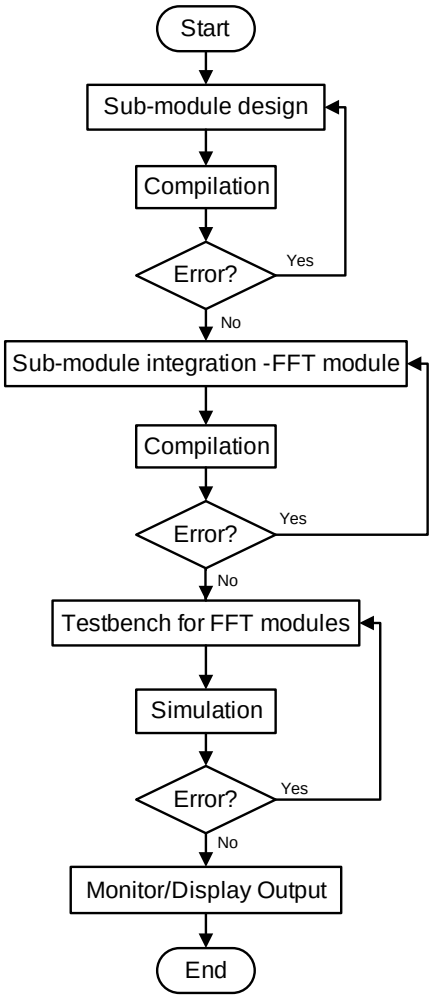

Figure 5: Pipelined FFT design flow

2.3 MC-CDMA receiver FPGA implementation

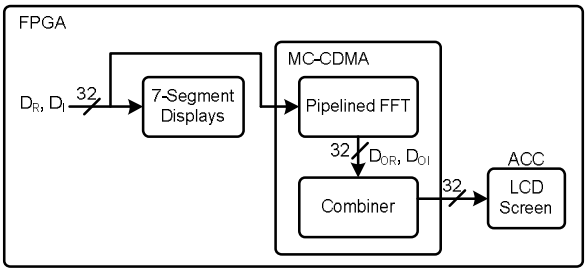

Figure 6: Block diagram of MC-CDMA on the FPGA

Shown in Figure 6 is the block diagram of MC-CDMA on the FPGA. The MC-CDMA design was verified on both Modelsim and Matlab; the design then goes through several stages on Quartus 13.1 before can be fully implemented on the FPGA [24][25]. In this research, the Altera DE2-115 FPGA board with Cyclone IV(EP4CE115F29C7N) chip is used. First, the MC-CDMA design was compiled in Quartus 13.1 compiler to check for errors. Next, the LCD and seven-segment modules were designed and compile for errors. In this research, the LCD is connected to the output of the MC-CDMA, while the input is connected to seven-segment. The input has two parts, real and imaginary with 16bits width each. Every four bits is represented by one seven-segment unit. In total, it requires eight seven-segment for the input display. The output of MC-CDMA can be displayed on one-line LCD. When both the MC-CDMA module and display modules were compiled with no error, these modules were integrated.

Before these modules can be program into the hardware, inputs and outputs need to be set in the pin planner, because each of the pins had been assign to the dedicated connection; for example, PIN_Y2 is allocated for 50MHz clock. The guidelines for this pin connection can be found in the DE2-115 manual. After the pin setting, the MC-CDMA module was programmed into the FPGA board. The codings are downloaded into the Cyclone IV (EP4CE115F29C7N) chip on the FPGA board. If there is no error in pin set, it shows 100% (successful), which mean the coding has been successfully downloaded into the FPGA.

Both seven-segment display and LCD were already embedded in the FPGA board. These two types of display need their modules for drive and control purposes. The seven-segment used seven different and individual LED's to display the hexadecimal symbol. In this research, the seven-segment display is set automatically to display the input of MC-CDMA. While for LCD, it also needs to be driven by a module that drives an eight-bits data bus and three control signals, E (enable), RW (read/write) and RS (register select). The LCD has 16 characters on two lines. The characters set are based on universal ASCII hexadecimal table. For example, if 'A', hex value needs to be set is 8'h41, indicating row 4, column 1.

2.4 Power analysis

For power analysis, Synopsys Design Compiler (DC) with Saed 90nm CMOS technology is used. Shown below are the libraries uses for this paper.

synthetic_library	: dw_foundation.sldb
link_library	: saed90nm_max_pg.db
target_library	: saed90nm_max_pg.db
symbol_library	: generic.sdb

After the design libraries and environment set, the design is synthesis and run for errors. Finally, the power report is generated and analyzed.

3. RESULTS AND ANALYSIS

This part is divided into three-segment, (1) MC-CDMA design verification, (2) MC-CDMA design implementation and (3) MC-CDMA power analysis.

3.1 MC-CDMA design verification

In this sub-section, the functionality of the MC-CDMA design is verified. Shown in Figure 7 is the sample input for MC-CDMA receiver and Figure 8 shows the expected output from the sample input. Input and output shown here are in hexadecimal. These results are generated from Modelsim. As can be seen in the waveform simulation, output can be generated simultaneously without any time gap.

Figure 7: MC-CDMA receiver sample input

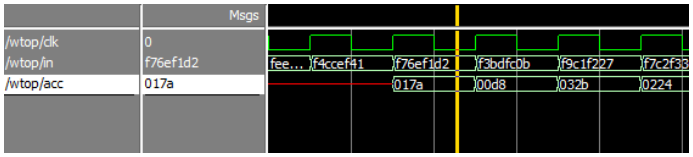

Figure 8: MC-CDMA receiver sample output

To verify the design, Verilog design results, as shown in the figure above, are tabulated and compared to Matlab Modelling results. Table 1 shows the Matlab input (DOR, DOI) and output (acc) as well as the simulation results via Modelsim. Shown in this table is part of the simulation (1-16) since actual simulation results can be generated up to thousands of data.

Table 1: Simulation results and output percentage difference of MC-CDMA

	Matlab (HEX)	Matlab (HEX)	Simulation (HEX)	% output difference
	DOR, DOI	Acc	acc	
1	F677F42F	0179	017A	0.265
2	032F022E	00D8	00D8	0
3	02940008	032B	032B	0
4	0012FFCD	0224	0224	0
5	00220091	014B	014B	0
6	FFF10038	01CC	01CD	0.217
7	00300019	01CE	01CE	0
8	FFE6FFCD	0353	0354	0.117
9	002AFFF1	00F8	00F9	0.403
10	FFC2FFB4	01CF	01D0	0.215
11	FFC7FF96	033D	033C	0.120
12	FFDF0019	0244	0245	0.172
13	FFF7FFAB	00D8	00D9	0.462
14	004CFFB2	01F1	01EF	0.402
15	0012FF5A	0170	0173	0.815
16	000D0027	038C	038D	0.110

For data accuracy, the percentage differences between the output of Matlab and Verilog simulation are compared. The Matlab output is considered as expected results, and Verilog output is the actual signal received. The accuracy calculations are based on this equation (6) below.

$$\%\ \boldsymbol{Output\ difference} = \left(\frac{\boldsymbol{X_1(n) - X_2(n)}}{\boldsymbol{X_1(n)}}\right) * \mathbf{100\%} \quad (6)$$

Where $X_1(n)$ is expected results from Matlab and $X_2(n)$ is the actual results from Verilog simulation. Lower output difference percentage indicates that the designs are fully functioning as expected and have high data accuracy.

3.2 MC-CDMA design implementation

MC-CDMA in Verilog design is then implemented on DE2-115 FPGA board. Shown in Figure 9 is a sample of input and output of the MC-CDMA receiver. Its input and output displayed on seven-segment and LCD respectively. It is proven that MC-CDMA is successfully implemented on the FPGA board.

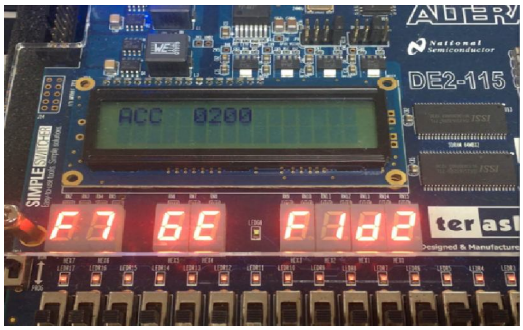

Figure 9: Sample input and output of MC-CDMA receiver on FPGA

Table 2 shows a sample of Matlab result, simulation result and hardware implementation result of MC-CDMA receiver. Data received are processed through pipelined FFT and combiner and produce output data acc. To verify the results, Matlab generated data are compared to the hardware implementation results. Hardware implementation is successful as the FPGA results displayed the same value as in the simulation.

Table 2: MC-CDMA hardware implementation

Matlab (HEX)	**Simulation – Modelsim (HEX)**	**Hardware – DE2-115 FPGA (HEX)**
ACC	**ACC**	**ACC**
0179	017A	

3.2 MC-CDMA power consumption

Shown in Table 3 is the power and area consumption of this thesis MC-CDMA design compared to design in [7]. MC-CDMA design in this research has significantly lower power consumption with 90.12mW less power consumption compared to the design in [7].

Table 3: Power consumption of MC-CDMA

	Power (mW)
[17]	129.25
Our MC-CDMA	39.13

5. CONCLUSION

This paper has successfully presented a novel low-power MC-CDMA receiver along with its modules, pipelined FFT and combiner. From the verification results, it is shown that the MC-CDMA is functioning as it is intended. The problems related to scheduling in the previous studies have been eliminated with MC-CDMA capabilities to simultaneously process all received data without data loss. This low-power MC-CDMA receiver design is successfully being implemented on FPGA and shows the expected results. MC-CDMA receiver design in this paper also indicates a significantly low power results compare to other research. In conclusion, low power MC-CDMA receiver is successfully designed and implemented on FPGA.

ACKNOWLEDGEMENT

The authors would like to thank the Faculty of Electrical Engineering, UiTM for providing necessary financial and facilities support.

REFERENCES

[1] B. H. Liu, N. Bulusu, H. Pham, and S. Jha, "CSMAC : A Novel DS-CDMA Based MAC Protocol for Wireless Sensor Networks," pp. 33–38, 2004.

[2] C. Prefix, "OFDMA , OFDM vs CDMA Comparison," pp. 1–4, 2017.

[3] P. Manhas and M. K. Soni, "Optimized OFDM model using CMA channel equalization for ber evaluation," *Bull. Electr. Eng. Informatics*, vol. 6, no. 2, pp. 133–139, 2017.

[4] N. S. Alotaibi, "An efficient technique for reducing transmission energy of sensor node to enhance the lifetime of the WSN's," *Int. J. Adv. Trends Comput. Sci. Eng.*, vol. 8, no. 2, pp. 108–111, 2019. https://doi.org/10.30534/ijatcse/2019/02822019

[5] A. Razaque and K. Elleithy, "Energy-Efficient Boarder Node Medium Access Control Protocol for Wireless Sensor Networks," *Sensors*, vol. 14, no. 3, pp. 5074–5117, 2014.

[6] N. Lavanya and T. Shankar, "A Review on Energy-Efficient Scheduling Mechanisms in Wireless Sensor Networks," vol. 9, no. August, 2016.

[7] N. Nasser, L. Karim, and T. Taleb, "Dynamic Multilevel Priority Packet Scheduling Scheme for Wireless Sensor Network," vol. 12, no. 4, pp. 1448–1459, 2013.

[8] K. E. Tepe, P. R. Casey, and N. Kar, "Design and implementation of a testbed for IEEE 802.15.4 (Zigbee) performance measurements," *Eurasip J. Wirel. Commun. Netw.*, vol. 2010, 2010. https://doi.org/10.1155/2010/103406

[9] R. Yap, G. Giron, L. M. Lanto, L. Garcia, D. S. Maria, and L. Materum, "Fpga library based design of a hardware model for convolutional neural network with automated weight compression using K-means clustering," *Int. J. Adv. Trends Comput. Sci. Eng.*, vol. 8, no. 4, pp. 1004–1011, 2019. https://doi.org/10.30534/ijatcse/2019/04842019

[10] B. Vinoth Kumar, M. Ramaswami, P. Swathika, and A. Raymon, "Wireless body area network with enhanced object identification, optimal storage and security for integrated healthcare system," *Int. J. Adv. Trends Comput. Sci. Eng.*, vol. 8, no. 3, pp. 847–853, 2019. https://doi.org/10.30534/ijatcse/2019/79832019

[11] D. Nogulic, M. Gadze, and a. Jankovic, "Multicarrier Modulation in Advanced Wireless Communication Systems," *Proc. ELMAR 2006*, no. June, pp. 7–9, 2006.

[12] R. Kumaravel and K. Narayanaswamy, "Performance enhancement of MC-CDMA system through novel sensitive bit algorithm aided turbo multi user detection," *PLoS One*, vol. 10, no. 2, pp. 1–23, 2015. https://doi.org/10.1371/journal.pone.0115710

[13] G. P. C. Rao, B. Ashok, and B. Saritha, "Design and Implantation of High Speed FFT Processor for OFDMA System Using FPGA," vol. 2, no. 7, 2013.

[14] M. R. Khalil and A. T. Ibrahem, "Design and Implementation of FFT / IFFT System Using Embedded Design Techniques," vol. 3, no. 6, pp. 258–264, 2013.

[15] M. H. Hasan, T. A. Arslan, and J. T. Thompson, "Low-Power-Adaptive MC-CDMA Receiver Architecture," *ETRI J.*, vol. 29, no. 1, pp. 79–88, Feb. 2007.

[16] C. Yu, M. H. Yen, P. A. Hsiung, and S. J. Chen, "A low-power 64-point pipeline FFT/IFFT processor for OFDM applications," *IEEE Trans. Consum. Electron.*, vol. 57, no. 1, pp. 40–45, 2011.

[17] M. Hasan, "Low Power Techniques and Architectures for Multicarrier Wireless Receivers," no. August, 2003.

[18] H. Zhou, "Design and FPGA Implementation of OFDM System with Channel Estimation and Synchronization," no. June, 2013.

[19] S. L. M. Hassan, N. Sulaiman, S. S. Shariffudin, and T. N. T. Yaakub, "Signal-to-noise ratio study on pipelined fast fourier transform processor," *Bull. Electr. Eng. Informatics*, vol. 7, no. 2, pp. 230–235, 2018.

[20] K. K. Parhi, "An efficient pipelined fft architecture," *IEEE Trans. Circuits Syst. II Analog Digit. Signal Process.*, vol. 50, no. 6, pp. 322–325, Jun. 2003.

[21] S. M. Senthil, J. S. Arockia, T. Arputharaj, and M. Banupriya, "Design of adaptive MC-CDMA receiver using low power parallel-pipelined FFT architecture," *2013 Pan African Int. Conf. Inf. Sci. Comput. Telecommun. PACT 2013*, pp. 29–33, 2013. https://doi.org/10.1109/SCAT.2013.7055085

[22] S. Navya, "Design and Implementation of Ofdm Transmitter and," pp. 1–4, 2015.

[23] P. S. Isasare and M. T. Kolte, "Fast Fourier Transform Implementation on FPGA Using Soft-Core Processor NIOS II," *Int. J. Eng. Trends Technol.*, vol. 10, no. 8, pp. 411–414, Apr. 2014.

[24] M. D. Firouzabadi and H. Heidari, "A Hybrid Hardware Verification Technique in FPGA Design," *Bull. Electr. Eng. Informatics*, vol. 3, no. 1, pp. 51–54, 2014. https://doi.org/10.12928/eei.v3i1.185

[25] K. S. Yadav and V. Charishma, "Design and simulation of 64 point FFT using Radix 4 algorithm for FPGA Implementation," vol. 4, no. 3, pp. 109–113, 2013.