Design and Implementation of Simulation Engine for Very High-Rate Communication over Power Grid

Sungsoo Choi and Hui-Myoung Oh
Power Telecommunication Research Center
Korea Electrotechnology Research Institute (KERI)
Ansan, S. Korea
{sschoi, hmoh}@keri.re.kr

***Abstract*—This paper discusses the design of an engine to simulate a Very high-rate Power Line Communication (VPLC) infrastructure that handles up to a rate of 400 Mbps. The simulation engine complies with the ISO/IEC 12139-1 standard for power line communication protocols. The engine can be used as one of the access network elements deployed in the advanced metering infrastructure in a smart grid. We first propose a feasible system model of VPLC in line with the design goals with pre-simulations, i.e., an event-driven simulation and a timing-driven simulation. Next, we design a semiconductor Intellectual Property (IP) core, focusing on the implementation of two main functional IP cores of the entire system: a fast Fourier transform for modulation/demodulation and a low-density parity-check encoder/decoder for forward-error correction. Finally, the designed VPLC based on the main functional IP cores is implemented on an available FPGA chipset, targeting the Virtex-6 xc6vlx240T.**

Keywords—very high-rate power line communications; advanced metering infrastructure; smart grid; FPGA; ISO/IEC 12139-1.

I. INTRODUCTION

The Smart Grid (SG) plays a major role in achieving a low carbon footprint and is therefore a key component of the sustainable energy infrastructure. Since 2010, to achieve the vision of "Low carbon, green growth," an SG test-bed has been built in Jeju island, Korea. Several power IT projects have been initiated integrating electric power technology as well as Information and Communication Technology (ICT) in five implementation areas, namely the smart power grid, smart consumer, smart transportation, smart renewables, and smart electricity service [1].

These implementation areas have completely different electrical environments and use different types PLC channels. Therefore, to meet various technical requirements for different network communication scenarios, several protocols and advanced modulation techniques are employed by Power Line Communication (PLC) systems in the SG. In addition, both the broadband spectrum from 1.8 MHz to 30 MHz (or 205 MHz) as well as the narrowband spectrum from 3 kHz to 500 kHz [2,3] are used for PLC. Advanced Metering Infrastructure (AMI), a typical SG application, has been successfully implemented using broadband high-rate communication at 24 Mbps over the power grid in compliance with the ISO/IEC 12139-1 standard, which is based on the Korea Standard (KS) 4600-1 [3,4].

In Section II, we briefly introduce the specifications of the engine designed for Very high-rate Power Line Communication (VPLC) that is compliant with ISO/IEC 12139-1, by introducing a static simulation with an event-driven operation and a dynamic simulation with a timing operation. In Section III, we explain the design and implementation of two Intellectual Property (IP) cores, namely MOD (for the functions of modulation/ demodulation) and FEC (for the function of error correction), at the system level with a cycle-based design approach, which is based on the design methodology at a clock-based register transfer level. Appropriate hardware-efficient algorithms are selected to design the hardware architectures, taking into account resource constraints of timing and area. All the design steps are carried out using both floating-point and fixed-point operations to ensure design consistency from the system level to the architecture level. Actually, both floating-point and fixed-point designs are useful to obtain a test bench for verifying the designed digital logic block at the 32-bit level, compared with the results of the dynamic simulation. Section IV explains how for a hardware gate level, specific functional blocks are edited by a Hardware Description Language (HDL), which enables the synthesis of logic gates in a 32-bit process. The IP cores are designed to support high-speed digital signal processing using a pipelining technique, parallelizing technique, and retiming technique to meet the system requirements. Finally, the main IP core circuit of VPLC is realized using a Field-Programmable Gate Array (FPGA) and is presented on the prototyping board. Conclusion is presented in Section V.

II. SYSTEM DESIGN ISSUES FOR VPLC

A. IP Design Issues for VPLC

VPLC is proposed to support utility applications in the implementation areas of the smart consumer and the smart transportation. To achieve this, we design a system with the following basic design goals:

- Supporting maximum rates of 200 Mbps and 400 Mbps in a dual transmission rate mode, in the frequency bandwidth less than 30 MHz and less than 80 MHz

 ISBN: 978-1-61208-360-5

- Adopting a multicarrier modulation technique for VPLC, such as Discrete Multi-Tone (DMT) modulation or Orthogonal Frequency-Division Multiplexing (OFDM)
- Adopting a Forward Error Correction technique (FEC) of a Low Density Parity Check (LDPC) to improve the link margin by 8.8 to 9.4 dB at BER = 10^{-5}
- Supporting coexistence of other systems compliant with ISO/IEC 12139-1 (or KS 4600-1) [3, 4].

B. *Specifications of VPLC engine*

To set the specifications of the VPLC engine, we first extract the main parameters from the allowable spectral efficiency from the typical multicarrier transmission system. After estimating the main parameters, we consider ways of improve the system performance to meet the system design goals. The formula to calculate the multicarrier transmission rate can be written as follows:

$$Bps = B_w \cdot M \cdot \frac{B_u}{B_w} \cdot \frac{T_{FFT}}{T_{FFT} + T_{CP}} \cdot C_r \quad (1)$$

where B_w is the entire signal bandwidth, B_u is the usable signal bandwidth, M is the number of bits in each subcarrier, T_{FFT} is the signal time to process a Fast Fourier Transform (FFT), T_{CP} is a cyclic prefix time, and C_r is the code rate for error correction. For designing a feasible system, the required overhead has to be adjusted, which can be decided by the factor of a guard band (B_u/B_w), a guard interval (GI) time ($T_{FFT}/(T_{FFT} + T_{CP})$), and a code rate ($C_r$). When considering a low frequency band of 0–28 MHz, the available bandwidth ratio becomes 0.8667. For convenience, we set the number of subcarriers to 2048 and the frequency band to 60 MHz in a high frequency band to reach a goal of greater than 400 Mbps. As shown in Table I, we need to set the main parameters as at least 1024-QAM modulation in the subcarrier for OFDM/DMT, the code rate greater than 7/8, and the guard interval time less than 2.13 μs.

By setting these parameters, we can achieve the required received *SNR* greater than 34 dB to meet the 1% packet error rate requirement for a typical multicarrier transceiver with a simple receiver structure in an ideal additive white Gaussian noise channel without any reflection phenomenon, as shown in Fig. 1.

TABLE I. MAIN PARAMETERS FOR MULTICARRIER TRANSMISSION.

CR	Modulation in a Sub-Carrier for OFDM /DMT	Number of Coded Bits per Symbol	Number of Bits per Symbol	Data Rate (Mbps)		Spectral Efficiency (bps/Hz)	
				GI = 4.27 μs	GI = 2.13 μs	GI = 4.27 μs	GI = 2.13 μs
1/2	256 QAM	13824	6912	180	190.5	3	3.17
3/4	256 QAM	13824	10368	270	285.8	4.5	4.76
5/6	256 QAM	13824	11520	300	317.6	5	5.29
7/8	256 QAM	13824	12096	315	333.5	5.25	5.55
1/2	1024 QAM	17280	8640	225	238.2	3.75	3.97
3/4	1024 QAM	17280	12960	337.5	357.3	5.62	5.95
5/6	1024 QAM	17280	14400	375	397.0	6.25	6.61
7/8	1024 QAM	17280	15120	393.75	416.9	6.56	6.94

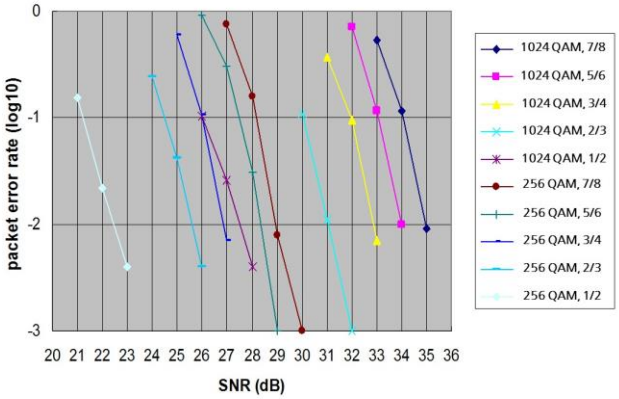

Figure 1. Performance of a Typical Multicarrier Transceiver.

Even under the real channel environment conditions of a power grid, it is possible to ensure more margins to meet the basic performance requirements. We consider additive techniques such as a data frame header check sequence [4] for minimizing the SNR, and a channel encoder [7,8] to protect the data frame header such as the LDPC. For designing an LDPC with a variable code rate of 1/2, 2/3, 3/4, 5/6, and 7/8, the system complexity can be effectively reduced by using the circular shift technique, considering a size-limited unit matrix of 24 by 24, to operate the large matrix of the LDPC.

TABLE II. SPECIFICATION OF VPLC.

Features	DMT Symbol variables		
	Preamble Frame	Control Frame	Long Symbol Frame, Data Frame Header, Data Frame
Bandwidth	25 MHz	25 MHz	75 MHz
Sampling Frequency	50 MHz	50 MHz	200 MHz
Tone Space	97.65625 kHz	97.65625 kHz	48.828125 kHz
IFFT Space	512 Samples	512 Samples	4096 Samples
Prefix Space	0 Samples	128 Samples	448 Samples
Roll-off Space	16 Samples	16 Samples	-
Symbol Duration	512 Samples	624 Samples	4544 Samples
FFT Period	10.24 μs	10.24 μs	20.48 μs
Symbol Length	10.24 μs	12.48 μs	22.72
Tone(or sub channel) Modulation	16 PSK	DBPSK	BPSK, QPSK, 16 QAM, 64 QAM, 256 QAM, 1024 QAM

In addition, to improve the system reliability, we consider a bit-loading technique to adjust variable modulation selection and variable channel code-rate selection in the subcarriers even at the cost of the transmission rate. Further, to overcome the performance degradation due to an impulsive noise on the power grid, we adopt a diversity technique, i.e. repeatedly transmitting a signal block in the time domain and using subcarrier utilization in the frequency domain. In other words, to ensure the compliance of the given deployed transceiver to

 ISBN: 978-1-61208-360-5

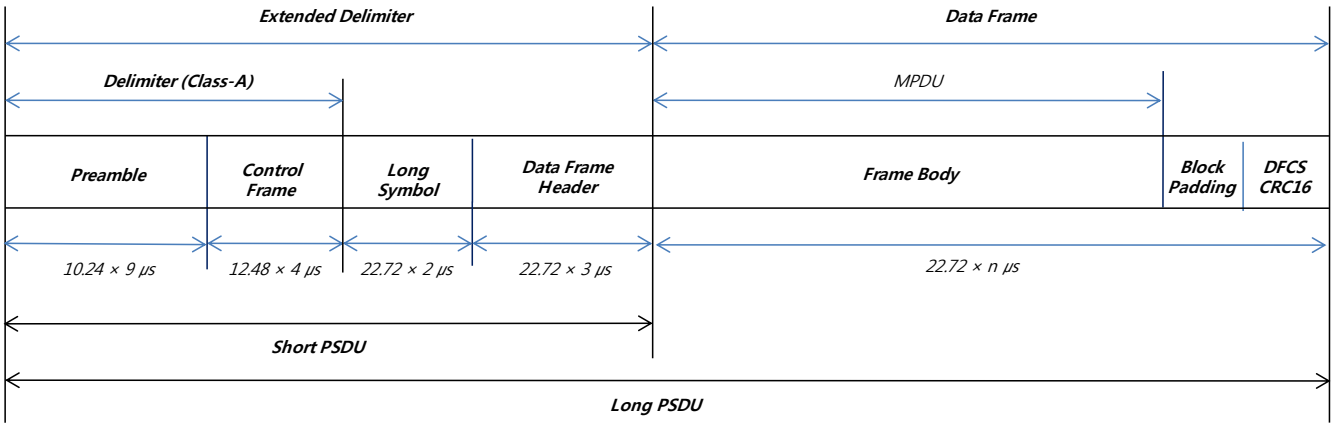

Figure 2. Transmitted VPLC Packet Service Data Unit Signal.

the ISO/IEC 12139-1 standard, we use an ISO/IEC 12139-1 based check sequence in the control frame, diversity mapping, Differential Phase Shift Keying (DPSK) modulation in subcarriers, forward error correction of RS(5,3), and 512-point FFT. Table II lists the specifications of VPLC. As shown in Fig. 2, the transmitted VPLC Packet Service Data Unit (PSDU) signal consists of a preamble frame, a control frame, and the rest of frames. The Delimiter signal has the preamble frame and the control frame. The Extended Delimiter has additional frames of the Long Symbol frame and the Data Frame Header. The Long Symbol consists of 2 DMT symbols and the Data Frame Header consists of 3 DMT symbols. The Data Frame consists of *n* DMT symbols. Each DMT symbol has 1,536 subcarriers in the frequency band of 75 MHz. The subcarrier bandwidth is 48.828125 kHz, and each DMT symbol length is 22.72 μs, because of the addition of a cyclic prefix with 2.24 μs for minimizing multipath channel effects. The DMT symbol sets to the identical field of the Delimiter signal according to the Class-A version of ISO/IEC 12139-1 for supporting the function of coexistence. The structure of the PSDU can be divided into the long PSDU and the short PSDU whether the Data Frame is included or not.

C. System Simulation for VPLC over Power Grid

1) Event-Driven (ED) Simulation

According to the proposed specification of VPLC, we first develop the ED simulator by using C++ to check the feasibility. The ED simulator can be independently operated by appropriate selection of the inter-frame types in PSDU. The transmit data, generated from a Medium Access Control (MAC) layer, pass through the first functional block viz. the scrambler that removes the unique pattern of the transmit data. In the following FEC block of LDPC, the transmit data is encoded to prevent data redundancy. At the Mapper block, the encoded data stream is converted to the bit-to-symbol data for the purpose of allotting it into each subcarrier. Next, the sequence of data is added to process the modulation step by using Inverse Fast Fourier Transform (IFFT) [6], giving an orthogonal property between subcarriers. The rest of the transmit signals except a Preamble signal are added by a Cyclic Prefix sequence to minimize the effect of an Inter-Symbol Interference (ISI) in the communication link channel. At the receiver side, the receive signal can be recognized by adding the Carrier Sensing block for detecting the signal frame, and by synchronizing the symbol to the starting point of the DMT symbol. Through the Windowing block, the size of the received signal sequence is reformatted to the length of FFT, and then it is demodulated through the FFT block. First of all, using the Long symbol, the receiver estimates the status of the given channel and compensates a timing error caused by the difference in the clock frequencies of the digital-to-analog converter and the analog-to-digital converter. The Demapper computes the log-likelihood ratio in each binary data. The LDPC decoder corrects the channel errors by using the parity information. Finally, in the descrambler block, the transmitted signal can be recovered. In this simulation, we use a conventional power line channel model such as a wireless Rayleigh channel model [5] as follows:

$$H(f) = \sum_{i=1}^{N} g_i \, e^{-\left(a_0 + a_1 f^k\right) d_i} \, e^{-j2\pi f\left(d_i / v_p\right)} \qquad (2)$$

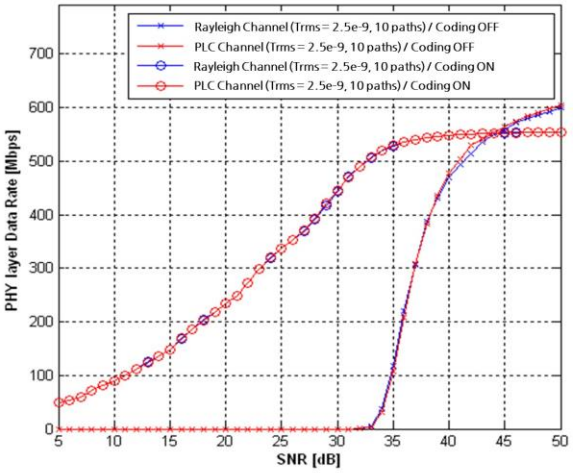

Figure 3. Simulation Result based on ED simulator.

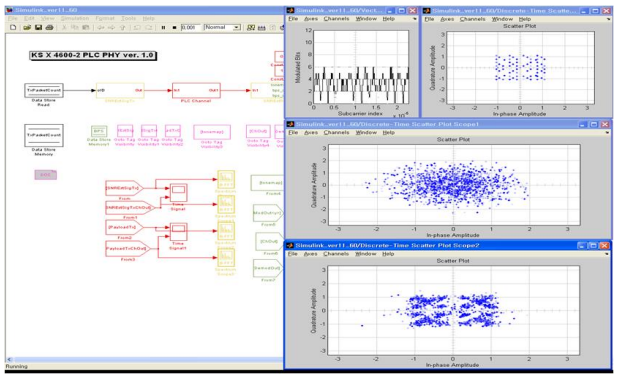

Figure 4. TD Simulator and Simulation Result.

where i is the number of paths, g_i is a weighting factor, k is an attenuation factor of exponent, a_0 and a_1 are attenuation parameters, and d_1 is the length of the path [5]. Fig. 3 shows communication performance of the proposed VPLC in the ED simulation for the Rayleigh channel model in (2) with the parameter of time = 2.5 e^{-9} s and 10 paths. It meets the minimum communication requirement of the signal-to-noise ratio (SNR) of about 28 dB to guarantee the rate of 400 Mbps. Without any FEC, to guarantee the rate of 400 Mbps, an SNR greater than approximately 37 dB is required.

2) Timing-Driven (TD) Simulation

The ED simulation is for a static design approach, whereas the TD simulation is for a more specific timing design approach, which operates in a symbol time basis. The GUI environment of the TD simulator is developed by using the S/W tool of the Simulink and the Matlab of MathWorks. The sample values can be analyzed and verified by both floating-point and fixed-point numerals, which support functional features required to design the architecture for controlling each parameter of the structural elements in the VPLC system. Besides, the status of the current signal stream can be checked by plotting the transmission symbols in both time and frequency domains. Fig. 4 shows the symbol-by-symbol transmission from the transmitter to receiver and a symbol-based scattering plot. The TD simulator developed was found to operate satisfactorily in accordance with the specifications of VPLC.

III. IP DESIGN ISSUES FOR VPLC

From the simulator introduced in previous section, reference vectors called a test bench can be extracted for basically designing and testing the specific logic architecture of IP in the chipset. In this section, we briefly study the IP design issues by introducing a couple of important IP engines, a MOD and a FEC for the proposed VPLC system.

A. Designing Signal Process of MOD

The MOD is one of core engines for signal processing and integrating the VPLC system. The engine supports the modulation/demodulation function in the proposed VPLC system, adopting an Orthogonal Frequency Division Multiplex (OFDM) technique. Functionally, the MOD maps binary data into OFDM signals, based on the modulation coefficients, by computing the 4096-point FFT. However, the complex 4096-point FFT may cause a bottleneck related its processing in a given clock time period. Therefore, a special architecture needs to be considered to effectively resolve this design problem related to timing and sizing. For an OFDM transmitter, the MOD can be processed by the Inverse FFT (IFFT) and it can be structurally designed just by hardwiring and switching the input and output ports of the FFT. The typical formula of the FFT is as follows [6]:

$$X[k]=\sum_{n=0}^{N-1}x[n]W_N^{nk} \qquad for \;\; k=0,1,...,N-1$$
$$W_N^{nk}=e^{-j\frac{2\pi}{N}nk} \qquad twiddle \;\; factor \qquad (3).$$

We adopt a Decimation-In-Frequency (DIF) FFT algorithm with Radix 4 (R4), which allows an easy design to have a semi-systolic parallel structure. The output sequence of the Radix-4 FFT is decimated as

$$X[4r]=\sum_{n=0}^{(N/4)-1}g_1[n]\cdot W_{N/4}^{nr}$$
$$X[4r+1]=W_N^n\sum_{n=0}^{(N/4)-1}g_2[n]\cdot W_{N/4}^{nr}$$
$$X[4r+2]=W_N^{2n}\sum_{n=0}^{(N/4)-1}g_3[n]\cdot W_{N/4}^{nr}$$
$$X[4r+3]=W_N^{3n}\sum_{n=0}^{(N/4)-1}g_4[n]\cdot W_{N/4}^{nr} \qquad (4)$$

where

$$g_1[n]=x[n]+x[n+(N/4)]+x[n+(N/2)]+x[n+(3N/4)]$$
$$g_2[n]=x[n]-jx[n+(N/4)]-x[n+(N/2)]+jx[n+(3N/4)]$$
$$g_3[n]=x[n]-x[n+(N/4)]+x[n+(N/2)]-x[n+(3N/4)]$$
$$g_4[n]=x[n]+jx[n+(N/4)]-x[n+(N/2)]-jx[n+(3N/4)]$$

Figs. 5 and 6 show the proposed architecture of the 4096-point FFT with parallelizing and pipelining. When designing the architecture of the R4 FFT algorithm with an internal clock speed that takes into account the critical path delay by adopting the design approach of a Single-path Delay Commutator (SDC) with a pipelined architecture. This architecture has the advantage of improving the utilization of the Butterfly module used to compute the complex multiplication performed repeatedly in each stage of the 4096-FFT. Each element of the Butterfly module consists of a 4-point DFT process.

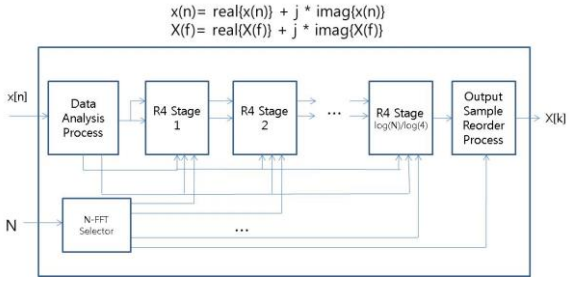

Figure 5. Architecture of the Proposed N-FFT (N = 4096).

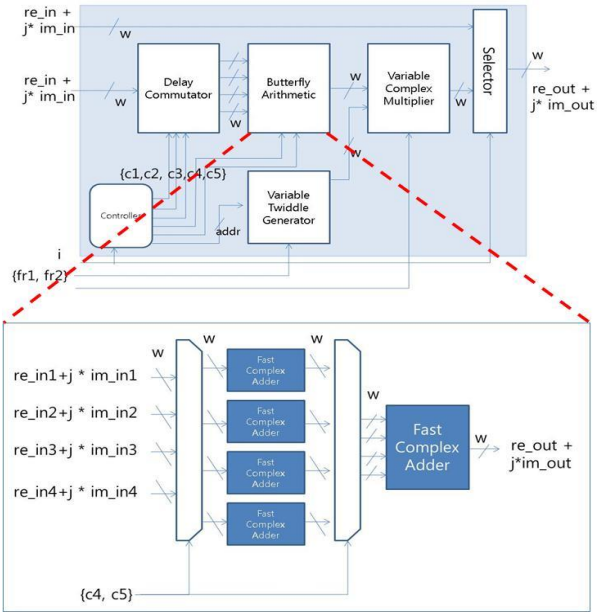

Figure 6. Architecture of the Basic R4 Butterfly Module in *i* th Stage in FFT.

Furthermore, to reduce the buffer size of the SDC, we logically design a RAM-type SDC to minimize quantization errors without any changes to the hardware architecture. By varying the magnitude of the input signal in the 4096-point FFT, the number of fixed point numerals can be automatically controlled. These are done in the front element of the data analysis process, which can be simply designed as a look-up table. In each stage, we consider controlling a variable twiddle-factor generator and complex multiplication with a fast complex adder. To reduce the complexity of the 4096-point FFT chip-level implementation, the basic functional blocks are shared. By sharing, the utilization factor can be increased up to 75% for the computation of complex multiplications. We obtain the result of the physical timing simulation and a Netlist of synthesized logic gate for targeting the Xilinx Vertex-6 chip. For verification of the design, we perform a harness test between the input/output signal results of the upper design layer with floating point values and that of the lower design layer with fixed point values. Furthermore, the input/output signals with binary data from the result of the HDL timing simulation are also verified with the harness test using the test bench of the outputs from the TD simulator.

B. Designing Signal Process of FEC

We design a FEC core engine by using a LDPC, which has features of the parity-check matrix with code rates of 1/2, 2/3, 3/4, (4/5), 5/6, (6/7), 7/8 and with code words of 576, 864, 1152, (1440), 1728, (2016), 2304. When designing the FEC, the LDPC uses a very large parity-check matrix for improving the decoding performance. In addition, it has different code rates in the proposed specification for VPLC. It means that the structure of the LDPC uses an individual parity-check matrix. Thus, it severely causes the problem of complexity in chip implementation. To reduce the complexity, the concept of sharing the function of a base matrix effectively is used. It can be operated adaptively according to the constant variation informative matrix in the parity-check matrix. We adopt the Richardson's algorithm for designing the architecture of the parity-check matrix and a structure of Quasi-Cycle (QC) for the submatrix [7,8]. Each submatrix uses a unit matrix with the size of G × G in accordance with the values of a circular cyclic shift in the parity-check matrix. It can be expressed as

$$\mathbf{I}^{s} = \left(a_{ij}\right), \quad a_{ij} = \begin{cases} 1, & if \ \bmod_{G}(i+s) = j \\ 0, & otherwise \end{cases} . \tag{5}$$

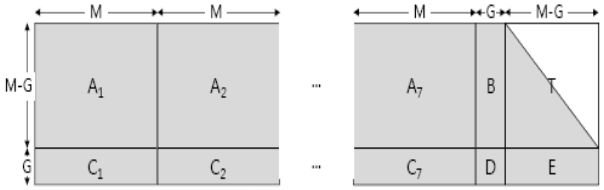

Figure 7. Structure of Parity-Check Matrix of LDPC for the FEC IP.

For the structure of the parity-check matrix, a redundancy part has a fixed length of M, and it can be changed from M to 7M, as shown in Fig. 7. In the part of redundancy, the parity vector p_1^T and p_2^T can be derived as follows:

$$\begin{aligned} \mathbf{p}_1^{\mathbf{T}} &= -\boldsymbol{\varphi}^{-1}\left(-\mathbf{E}\mathbf{T}^{-1}\mathbf{A} + \mathbf{C}\right)\mathbf{u}^{\mathbf{T}} \\ \mathbf{p}_2^{\mathbf{T}} &= -\mathbf{T}^{-1}\left(\mathbf{A}\mathbf{u}^{\mathbf{T}} + \mathbf{B}\mathbf{p}_1^{\mathbf{T}}\right) \\ \boldsymbol{\varphi} &= -\mathbf{E}\mathbf{T}^{-1}\mathbf{B} + \mathbf{D} \end{aligned} . \tag{6}$$

Fig. 8 shows the architecture of the LDPC encoder, composed of the 24 × 24 submatrix (i.e., G = 24), with the length of information bits of 24 × 12 × n (where n = 1–7), and the final code words of 24 × 12 × (n + 1). For encoder design, we adopt the pipelining process technique with four stages, supporting the 24 clocks per each stage and the latency of 96 clocks. As shown in Fig. 8, we use circular shift registers for storing and dividing the input information bits according to code rates at the first step in the encoder. We devise on an arithmetic process for computing the sequential multiplication of the parity-check matrix and the information bits, and the parity vector arithmetic process (6) at the second step and the third step, respectively. In the last step, we obtain the output combining the parity vector. In other words, the decoder is designed for a variable iteration number (n) of the pipelining architecture with 2(n + 1) + 2 stages, 288 clocks, and 288(2n + 4) latency clocks. Besides, we set a stop option of iteration as necessary. The decoder consists of a PreDecoder with two stages, followed by a IterationDecoder with two stages and a ParityChecker with two stages, and, lastly, a Terminator with two stages, as shown in Fig. 9. Thus, according to the channel

environment, a proper code rate from the designed LDPC can be obtained by controlling the information bits.

the IP of MCU, a 32-bit SE3208 microprocessor, is mounted on the Xilinx FPGA, targeting Virtex-6

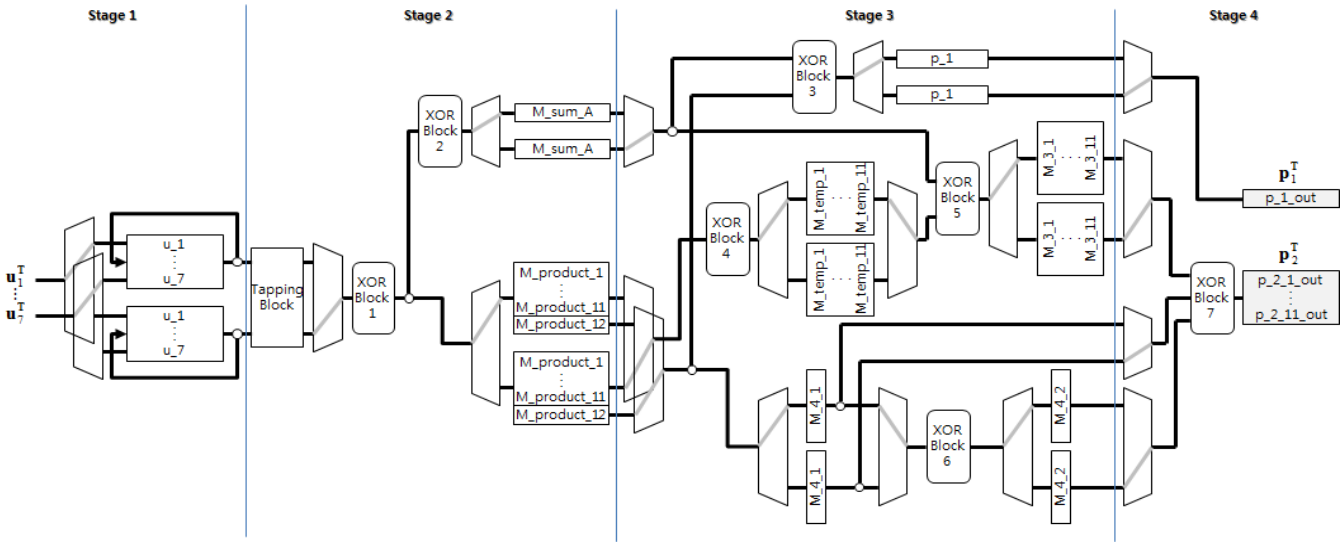

Figure 8. Architecture of the LDPC Encoder for the FEC IP.

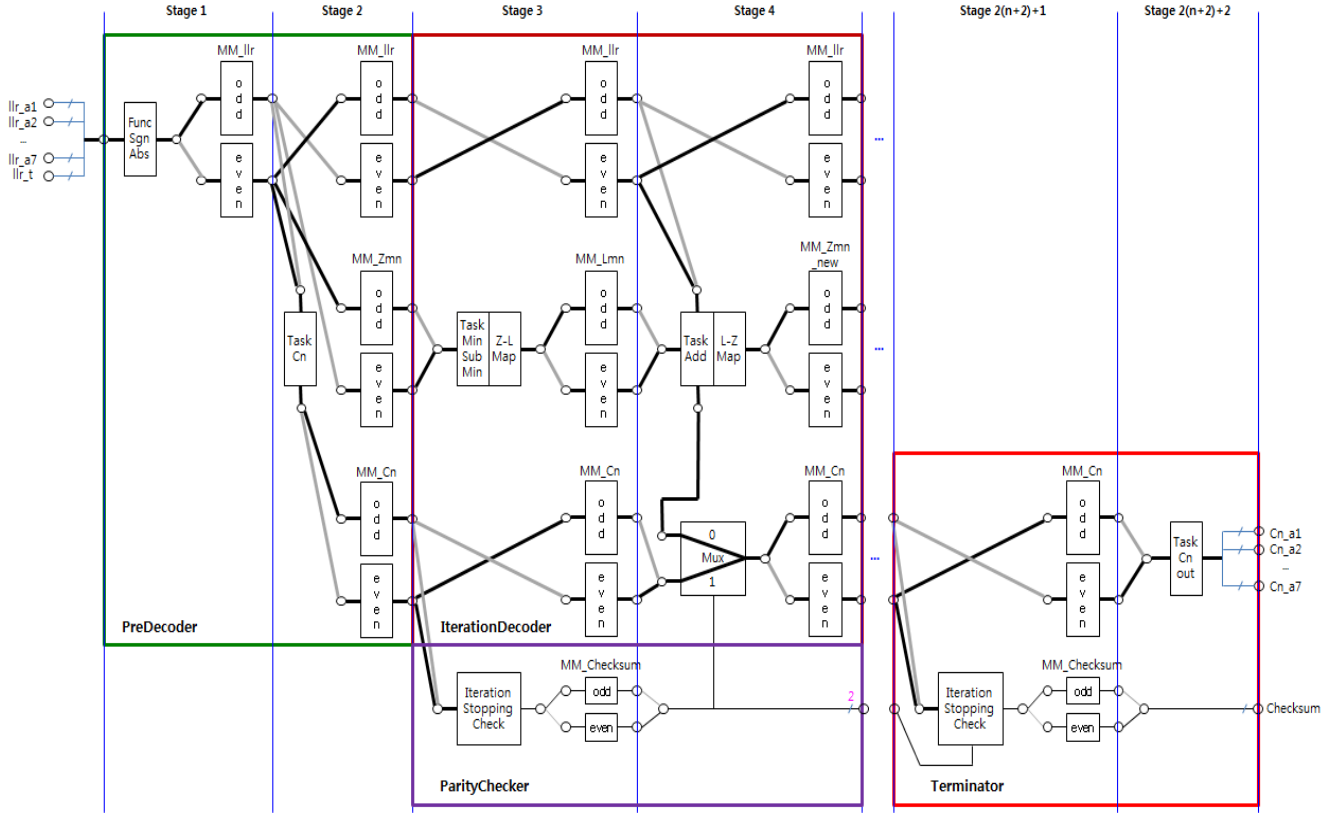

Figure 9. Architecture of the LDPC Decoder for the FEC IP.

IV. IMPLEMENTATION OF TEST PLATFORM

After the back-end design of the VHLC has been completed, the physically synthesized logic, combined with xc6vlx240T. The test platform is developed and demonstrated to transfer the data with a moving picture as a real operation, as shown in Fig. 10. It has an operational

function to monitor streaming data and directly control system configurations on the test board.

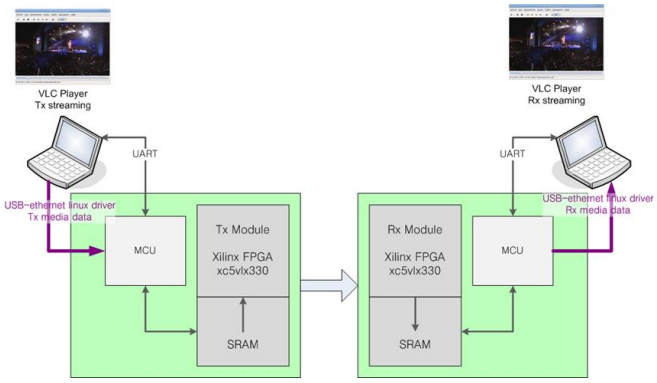

Figure 10. Test Platform for the VPLC Physical Layer.

V. CONCLUSION

In this study, we proposed a VPLC and demonstrated its feasibility of operation at a rate of more than 400 Mbps in the physical layer of the power grid environment. Further, we designed and implemented two core engines, namely the MOD and the FEC, necessary for simulating the VPLC system. We proposed and adopted the appropriate design algorithms and the design architecture to meet system specifications and requirements so that sharing main functional modules will reduce the system complexity and parallelize it effectively. All the designs were achieved by step-by-step verification of the front-end design. These designs were successfully tested on the prototyping board of the implemented test platform. This fundamental work is expected to contribute significantly to achieve the single-run VPLC chipset implementation.

REFERENCES

[1] http://www.smartgrid.or.kr/ [accessed: May 2014]

[2] S. Choi, H. M. Oh, Y. Kim, and Y. H. Kim, “Study on Very High-Rate Power Line Communications for Smart Grid,” KIEE Trans. Inst. Electrical Engineers, vol. 60, no. 6, Jun. 2011, pp. 1255-1260.

[3] S. Eom et al., “Physical layer certification system of power line communication Korea standard,” IEEE International Symposium on Power Line Communications and Its Applications, Apr. 2009, pp. 325-330.

[4] ISO/IEC12139-1, Information technology - Telecommunications and information exchange between systems - Power line communication-High speed PLC medium access control and physical layer-Part1: General requirements, Jul. 2009.

[5] M. Zimmermann and K. Dostert, “A Multipath Model for the Power Line Channel,” IEEE Trans. on Comm., vol. 50, no. 4, Apr. 2002, pp. 553-559.

[6] W. Li, Y. Ma, and L. Wa., “Word Length Estimation for Memory Efficient Pipeline FFT/IFFT Processors,” IEEE Signal Processing Systems, 1999.

[7] T. J. Richardson and R. L. Urbanke, “Efficient Encoding of Low-Density Parity-Check Codes,” IEEE Trans. Inform. Theory, vol. 47, no. 2, Feb. 2001, pp. 638-656.

[8] M. P. C. Fossorier, “Quasi-Cyclic Low-Density Parity-Check Codes From Circulant Permutation Matrices,” IEEE Trans. Inform. Theory, vol. 50, no. 8, Aug. 2004, pp. 1788-1793.

 ISBN: 978-1-61208-360-5