Implementation of Complex Wavelet Packet Modulation System Using FPGA Platform

Hikmat N. Abdullah$^1$, Fadhil S. Hassan$^1$ and Alejandro Valenzuela$^2$
$^1$Electrical engineering department Al-Mustansiryah University, Baghdad, Iraq
$^2$EMT department H-BRS University of applied sciences, Bonn, Germany

Abstract: Complex wavelet Packet modulation CWPM system has gained special interest as compared with traditional WPM due to its positive features like shift insensitivity, good directionality, and presence of phase information. The objective of this paper is to design a baseband $n$/4-DQPSK CWPM transceiver and implementing it on FPGA hardware. The design uses 8-point Discrete Wavelet Packet Transform DWPT/ Inverse Discrete Wavelet Packet Transform IDWPT as the core processing module. All modules are designed using VHDL programming language. Software tools used in this work are Altera Quartus II 9.1 and Altera ModelSim 6.5b. Cyclone III board EP3C120F780C7 is used as the target device.

Keywords: Field Programmable Gate Array, Complex Wavelet Packet Modulation, DQPSK,

1. Introduction

Orthogonal Frequency Division Multiplexing (OFDM) provides an efficient means to handle high speed data streams on a multipath Rayleigh fading channel that causes Intersymbol Interference (ISI). In recent years, wavelet packet modulation (WPM) has been proposed as one of the multicarrier modulation systems that use discrete wavelet transform (DWT)[1-4] to decrease the bandwidth waste brought by adding cyclic prefix and to reduce the transmission power. DWT uses only real arithmetic, as opposed to the complex valued FFT. This reduces the signal processing complexity and power consumption. However, DWT suffers from three major limitations: shift sensitivity, poor directionality, and absence of phase information. To solve these problems, Complex wavelet packet Modulation (CWPM) [5] is proposed as the new future of multicarrier modulation system. $\pi$/4- DQPSK is widely used modulation scheme in satellite radio applications with low error rates and high signal reliability [6]. The envelope of $\pi$/4- DQPSK is better than QPSK but is more susceptible to envelope variations than OQPSK. However, $\pi$/4- QPSK has the advantage over OQPSK in that it can be differentially detected.

The CORDIC (COordinate Rotation Digital Computer) is a fast technique to evaluate the elementary functions, such as trigonometric, exponential, and logarithmic functions because it uses only add-and-shift operators instead of multipliers to implement the functions. Furthermore, it does not require a complexity in implementation as compared with lookup tables. It was proposed by Volder [7] in 1959 and generalized by Walther [8]. The implementation of a CORDIC processor using Field Programmable Gate Array (FPGA) is introduced in [9]. Since CORDIC is originally designed to implement nonlinear functions, therefore; it uses fixed-point notation to represent the fractional numbers. Fixed-point notation is an interpretation of a 2’s compliment signed number but also there is not sign representation. Fixed-point representation of a number consists of signed, integer and fractional components.

Figure 1 shows the fixed-point representation. The notations $\kappa$ and $\ell$ are the total word length of the representation in bits and the word length of the integer part respectively. The first bit is the sign bit, the second word is the integer component of the number and the last word is the fractional component of the number with word length $\kappa - \ell - 1$ bits. In this representation the range of the number is $[-2^\ell, 2^\ell]$ with a step size (i.e. resolution) of $2^{-\left(\kappa - \ell - 1\right)}$ [10]. In this paper, the value of $\kappa$ is 13 bits and $\ell$ is 4 bits. For example if the real number is 0.7071 then the fixed-point format for that number is 0000010110101.

In this paper, $\pi$/4-DQPSK complex wavelet packet modulation system are proposed and implemented it using FPGA technology. CORDIC algorithm is used to implement the detection of $\pi$/4-DQPSK system and any trigonometric function in the design. All the numbers and mathematical operations in the design are performed as fixed point operations.

2. Complex Wavelet Packet Modulation (CWPM) System

![Figure 1 Fixed point representation.](image_url)
Figure 2 illustrates a generic block diagram of CWPM transceiver. CWPM system employs two filter banks i.e. Inverse Discrete Wavelet Packet Transform (IDWPT) (reconstruction) placed at the transmitter side, and Discrete Wavelet Packet Transform (DWPT) (decomposition) placed at the receiver side. The block "MAKE CPLM" accepts two N-dimensional real vectors as inputs. Its output is an N-dimensional complex vector whose ith complex element is formed from the ith real elements of the two input vectors. The output to the inverse wavelet transform is DQPSK complex symbols, therefore, there are two IDWPT blocks, one for real symbols and the other for imaginary symbols. The output of two IDWPT is combined together in complex form to introduce the transmitted signal x[n]. The transmitted signal is constructed as the sum of N waveforms \( \phi_j[k] \) individually modulated with the DQPSK symbols as follows:

\[
x[n] = \sum_{k} \left[ \sum_{j=0}^{N-1} (a_{k,j} + i b_{k,j}) \phi_j(n-kN) \right]
\]

where \( a_{k,j} \) and \( b_{k,j} \) are a real and imaginary constellation encoded kth data symbol modulating the jth wavelet packet basis function respectively. It is worth noted that CWPM symbols would have extended duration due the natural convolution of wavelet filters which results in symbol overlapping. This overlapping increases as the wavelet filter length increases.

3. FPGA Implementation of CWPM System

The parameters used for \( \pi/4 \)-DQPSK CWPM system are summarized as follows: the bit rate of the binary input is 50 M bits/sec. The number of subcarriers is 8. Wavelet filter type is db2. There are 14 overlapped samples as a result of wavelet filters; therefore, the output of IDWPT is 22 samples. The system employs \( \pi/4 \)-DQPSK modulation system with 2 bits/state, thus the useful rate of transmitted signal is 68.75 M samples/sec. The implementation of CWPM system consists of two parts: the transmitter and the receiver. The detailed descriptions of each part are presented below:

- **The Transmitter Section:**

  The transmitter is implemented with five main stages named: S2P1, Mapping, two blocks of S2P2, two blocks of IDWPT and two blocks of P2S, are shown in Figure 3. The transmitter module requires three clocks: clk1, clk2 and clk3. The relation between them is shown in Table 1. Note that clk1 is the master clock generated using in Quatrus II built in function called altpll while clk2 and clk3 are derived from clk1 using the block clock distributer as shown in Figure 3. The serial input data (BIT1) is converted to frame of two parallel bits using S2P1 entity. Mapping entity maps the two bits binary input into real and imaginary symbols according to Table 2. Each symbol is represented by fixed-point format. The two S2P2 entities are used to convert the serial samples to 8 parallel samples for both the in-phase part and the quadrature-phase part respectively. The parallel samples are then entered to DQPSK Modulation entity to obtain the in-phase and quadrature-phase components of \( \pi/4 \)-DQPSK baseband modulation. The in-phase and quadrature phase components of the kth symbol can be expressed in matrix form as:

\[
\begin{bmatrix}
I_k \\
Q_k
\end{bmatrix} =
\begin{bmatrix}
\cos(\Delta0_k) & -\sin(\Delta0_k) \\
\sin(\Delta0_k) & \cos(\Delta0_k)
\end{bmatrix}
\begin{bmatrix}
I_{k-1} \\
Q_{k-1}
\end{bmatrix}
\]

The entity DQPSK Modulation is implemented using VHDL behavioral module based on the calculation of Equation 2. The initial value is selected to be \( 1, e^{i0} \).

All the addition and multiplication operations are fixed-point mathematics. The in-phase and quadrature phase symbols are then applied to IDWPT entities, one for real and the other for imaginary part respectively to obtain the complex wavelet modulated signal. Finally, the outputs of IDWPT entities are converted back to serial form to be ready to send to DAC.

- **The Receiver Section:**

  Five main stages have been implemented in whole receiver, named: S2P (two blocks), DWPT (two blocks), DQPSK Demodulation, P2S1 (two blocks), Demapping and P2S2 as shown in Figure 4.

  The receiver module is similar to transmitter module in that it requires three clock signals with the same proportionality. After the serial input symbols are converted to 8 parallel symbols by S2P entities, CWPT entity is used to reconstruct the real and imaginary wavelet symbols. CWPT component has two DWPTs; one is applied for the real part and the other for the imaginary part. These symbols are then sent to differential detector which consists of an Angle Calculation entity to extract the signal phase and Difference Angle entity which determines the phase change over one symbol interval \( (\Delta0_k = 0_k - 0_{k-1}) \). These functions are introduced by using vectoring and rotating modes of CORDIC algorithm. The differences of phases angles are then applied to Sine and Cosine Calculation entity to calculate \( \cos(\Delta0_k) \) and \( \sin(\Delta0_k) \) components. P2S1 entities are used to convert the parallel 8-samples to serial real and imaginary samples. The serial output of the differential encoder is applied to Demapping entity which is a decision circuit. The decisions taken by Demapping entity are based on the following formulas:

<table>
<thead>
<tr>
<th>Clock</th>
<th>Ratio</th>
<th>Actual setting (MHz)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clk1</td>
<td>1/1</td>
<td>50</td>
</tr>
<tr>
<td>Clk2</td>
<td>1/2</td>
<td>25</td>
</tr>
<tr>
<td>Clk3</td>
<td>1/4</td>
<td>68.75</td>
</tr>
</tbody>
</table>

Table 1. The clock distribution
Table 2 Relation between input and output symbols for entity Mapping

<table>
<thead>
<tr>
<th>Information bits</th>
<th>Phase shift $\Delta \theta_k$</th>
<th>$\cos(\Delta \theta_k)$</th>
<th>$\sin(\Delta \theta_k)$</th>
<th>Fixed-point representation</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>$\pi /4$</td>
<td>0.7071</td>
<td>0.7071</td>
<td>000001011010101</td>
</tr>
<tr>
<td>01</td>
<td>$3\pi /4$</td>
<td>-0.7071</td>
<td>0.7071</td>
<td>100001011010101</td>
</tr>
<tr>
<td>11</td>
<td>$-3\pi /4$</td>
<td>-0.7071</td>
<td>-0.7071</td>
<td>000001011010101</td>
</tr>
<tr>
<td>10</td>
<td>$-\pi /4$</td>
<td>0.7071</td>
<td>-0.7071</td>
<td>100001011010101</td>
</tr>
</tbody>
</table>

Figure 2 Block diagram of the CWPM system model. Arrows with a strike indicate complex quantities.

Figure 3 The transmitter implementation.

Figure 4 The receiver implementation.
b_l = \begin{cases} 1 & \text{if } \cos(\Delta\theta) \geq 0 \\ 0 & \text{if } \cos(\Delta\theta) < 0 \end{cases}
\quad (3)

b_Q = \begin{cases} 1 & \text{if } \sin(\Delta\theta) \geq 0 \\ 0 & \text{if } \sin(\Delta\theta) < 0 \end{cases}

where \( b_l \) and \( b_Q \) are the detected bits in the in-phase and quadrature-phase component respectively. This decision depends on the first bit of each symbol since the first bit contains the sign bit. The resulting two parallel bits are then converted to serial binary stream by P2S2 entity. Finally, an extra VDHL entity has been written for the purpose of achieving synchronization between the transmitter and receiver waveforms.

4. Simulation Results

All block entities in this design were written as VHDL modules. The verification of implementation is done by MODELSIM program version 6.5b and Quartus II version 9.1. Figure 5 shows the transmitter mapping entities input/output simulation waveforms. Two control signals are used in the design:

- **data_valid** signal to indicate the end of 8 symbols loading in each frame.
- **start_P2S** signal to indicate the starting of parallel to serial operation in each frame.

Figure 6 shows the receiver mapping entities input/output simulation waveforms. Here, there are three control signals are used in the design:

- **Start_S2P** signal to indicate the starting of each frame and starting serial to parallel operation.
- **data_valid** signal to indicate that the 8 symbols loading is completed in each frame.
- **start_P2S** signal to indicate the starting parallel to serial operation in each frame.

It can be seen from figures 5 and 6 that the input data (bit1) at the transmitter is exactly the same as the recovered data at the output of receiver.

- **Synthesis Reports:**
  There is a large number of synthesis reports (hardware and software reports) obtained from synthesis operation. They describe all what concern the implementation process like storage resources required, I/O resources required, computation resources required, time delay at different points inside the chip ….etc [11]. Table 3 shows a summary of hardware synthesis reports for the designed CWPM system.

- **Hardware test:**
  The FPGA implemented CWPM system is downloaded to Altera Cyclone III board EP3C120F780C7. The board is interfaced with High Speed Mezzanine Card (HSMC) which contains two 14 bits ADC and DAC channels to test the proposed system with analogue inputs. Figure 7 shows hardware CWPM system. The analogue input is digitized using ADC with sampling rate 3.5 MHz. The A/D and D/A converter need two clocks, inverting and non-inverting clock input. These two clocks are used as maximum frequency to converter circuit. The system has been successfully tested with 32 kHz sinusoidal input signal as shown in the photo of Figure 8. The upper waveform in the oscilloscope screen shows that input signal while the lower one is the successfully recovered one.

5. Conclusions

A baseband \( \pi/4 \)-DQPSK CWPM system was designed and successfully implemented and tested using Altera Cyclone III EP3C120F780C7 FPGA development board. The design of the has used CORDIC algorithm to compute the Daubechies wavelet transforms and modulation/demodulation processes with fixed point notation. The implementation of \( \pi/4 \)-DQPSK CWPM system using mentioned tools makes it more faster, less complex more flexible to parameter changes and can be realized on-chip basis. The hardware simulation results show that CWPM module is working correctly and the components are synchronized together. A sinusoidal test signal has sent and recovered successfully to check the functionality of ADC and DAC converters of HSMC interfacing board. The synthesis results show that the implemented system could support the real time operation of CWPM system.

Table 3. Summary of hardware synthesis reports for CWPM system

<table>
<thead>
<tr>
<th>Resource</th>
<th>Used</th>
<th>Available</th>
<th>Utilization</th>
</tr>
</thead>
<tbody>
<tr>
<td>I/O pins</td>
<td>45</td>
<td>332</td>
<td>8%</td>
</tr>
<tr>
<td>Total combinational functions</td>
<td>77,463</td>
<td>119,088</td>
<td>64%</td>
</tr>
<tr>
<td>Dedicated logic registers</td>
<td>7,960</td>
<td>11,908</td>
<td>7%</td>
</tr>
<tr>
<td>Embedded multiplier 9 bits element</td>
<td>52</td>
<td>532</td>
<td>8%</td>
</tr>
<tr>
<td>Total FLSs</td>
<td>1</td>
<td>4</td>
<td>25%</td>
</tr>
</tbody>
</table>

References


Figure 5 The transmitter input/output simulation waveforms

<table>
<thead>
<tr>
<th>Channel</th>
<th>Signal</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>c1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>c2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>c3</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure 6 The Receiver input/output simulation waveforms

Inverting clock input for A/D and D/A converter

Analogue Input

Analogue output

Non-inverting clock input for A/D and D/A converter

Figure 7 Hardware testing of the FPGA implemented CWPM system.
Figure 8  The input and successfully recovered waveforms


