

# Design and FPGA Implementation of Non-Data Aided Timing and Carrier Recovery Techniques for EDR Bluetooth Standard

Khaled Salah Mohammed, Member IEEE  
khaled\_mohamed@mentor.com

**Abstract:** The main design issues for Bluetooth transceivers are not only low cost and low power consumption, but also quality performance. Classical designs of the Bluetooth receiver utilize data-aided techniques to correct carrier frequency offsets and symbol timing errors. Such techniques offer low cost and reasonable performance. Non-data aided techniques offer an alternate higher-performance approach to correct the same problems, at the penalty of an increased hardware complexity and cost. The purpose of this paper is to investigate the trade off between cost and performance when a Bluetooth 2.0 (Enhanced Data Rate) transceiver is designed using non-data aided techniques for clock and timing recovery.

**Keywords:** Bluetooth, EDR, GFSK, DPSK.

## 1 Introduction

### 1.1 Background

Bluetooth is a short-range wireless communication standard designed with an intention of replacing cables connecting portable and desktop devices and builds Wireless networks for such devices and enables communication to be established between these devices up to maximum distance of 100 meters. Bluetooth is featuring robustness, low cost, low complexity and low power, it uses the free industrial, scientific and medical (ISM) frequency band (2.4 -2.48 GHz). The Bluetooth specification is an open specification that is governed by the Bluetooth Special Interest Group (SIG)[1].

One of the disadvantages of the original version of Bluetooth in some applications was that the data rate was not sufficiently high, especially when compared to other wireless technologies such as 802.11. In November 2004, a new version of Bluetooth, known as Bluetooth 2.0 was ratified. This not only gives an enhanced data rate but also offers other improvements as well.

There are two cores of Bluetooth; Bluetooth 1.1 which is the basic data rate (1Mbps) and Bluetooth 2.0 which is the enhanced data rate (2 or 3 Mbps). Table 1 provides an overview of the Bluetooth cores. The hardware level specification for the Bluetooth standard consists of four layers; the Radio layer, Baseband, LMP and HCI layer.

Table 1 Bluetooth Cores and Transmission Rates

Bluetooth core	Mode	Bit Rate
1.1	basic rate	1 Mbps
2.0	Enhanced rate	2 or 3 Mbps

The important difference between the basic rate Bluetooth packet and the enhanced rate Bluetooth packet is that modulation scheme is changed within a high rate as basic rate specifications supports only binary modulation where the enhanced rate Bluetooth packet is M-ary modulated [2]

### 1.2 Contribution

This research differs from the researches published up to date in that it presents Bluetooth 2.0 Transceiver architecture suitable for implementation in an FPGA. It will examine several digital algorithms for modulation and demodulation of Bluetooth signals, locking on the carrier phase, and synchronizing the symbol. Many of these previously analog designs have been translated to the digital domain. The existing literature does not study modem architectures to determine an effective implementation to handle the variation in bit rate due to different constellations while minimizing redundant hardware required to operate on multiple constellations in an FPGA. Literature is also deficient on synchronization techniques for multiple constellations and the implementation of these techniques in an FPGA.

Unlike ASICs, FPGAs are reconfigurable, that is, their internal structure is only partially fixed at fabrication, leaving to the application designer the wiring of the internal logic for the intended task. This can significantly shorten design and production, and thus time to market, for FPGA-based systems. Although FPGAs tend to be slower and to consume more power than ASICs, FPGA reconfigurability can benefit platform longevity (which is extremely important in an era of fastchanging wireless communications standards) by allowing design changes/upgrades even in systems already in operation. This flexibility can be effectively exploited for rapid prototyping of advanced communications signal processing.

## 2 System Architecture

### 2.1 GFSK Transceiver

The Bluetooth standard specified GFSK modulation due to its robustness against signal fading and interference, maintaining a good spectral efficiency. GFSK belong to a class of digital modulation known as Continuous Phase Modulation (CPM). Signals Modulated by these kinds of modulation have a constant envelope. This allows the use of a simple amplifier during transmission which doesn't require a high cost linear power amplifier. In addition Detection of GFSK can be based on relative frequency changes between symbol states and thus does not require absolute frequency accuracy in the channel. GFSK is thus relatively tolerant to local oscillator (LO) drift and Doppler Shift.

Everything in a GFSK modulator is the same as in a FSK modulator, except that before the baseband pulses go into the FSK modulator, it is passed through a Gaussian filter to make the pulse smoother and limits its spectral width. The spectral width for FSK is unlimited, in contrast; there is a limitation on GFSK[3]

#### 2.1.1 Proposed GFSK transmitter

A block diagram of proposed GFSK is shown in Figure 1. This design was preferred over other schemes because of its simplicity and continues phase as discussed before. The Numerical controlled oscillator for look up tables (save area). For the first step of modulation, No encoding schemes are used in GFSK modulation, The binary stream is used directly in the next step of modulation which is pulse shaping to reduce bandwidth. Pulse shaping is performed with a Gaussian filter. The GFSK signal is mixed with two different modulation frequencies  $F_1$ ,  $F_2$ .

#### 2.1.2 Proposed GFSK Receiver

The phase discriminator structure was chosen over other schemes because it is suitable for both GFSK and PSK modem (Bluetooth2.0) to save hardware area. The demodulation process is the mixing with the same modulation carrier frequency as used in the transmitter part, after the mixing the signal is low pass filtered to obtain sine and cosine components of the phase and then we get the ARCTAN of the upper and lower branches, it is differentiated and then sampled and enter the decision device (SLICER) to determine which symbol was transmitted, The GFSK Receiver is Shown in Figure2.

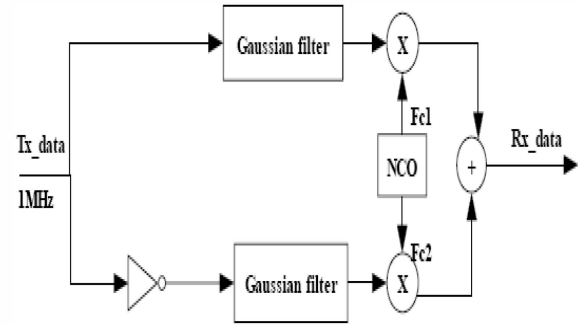


Figure 1 Proposed GFSK Transmitter

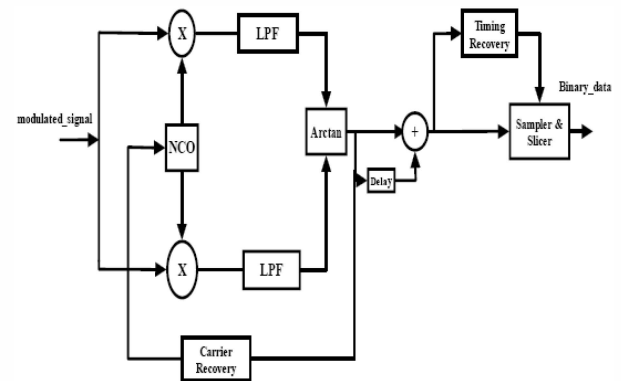


Figure 2 Proposed GFSK Receivers

#### 2.1.3 Proposed Timing Recovery Algorithm

The simplicity of the early-late gate algorithm made it a very good choice compared to other algorithms. While the other algorithms claim to have faster response time or estimation accuracy, these algorithms use more resources. The tradeoff between resources and performance led to the determination that the early-late gate algorithm is the preferred choice.

it consists of timing error detector (based on early-late gate method), second order loop filter, and numerical controlled clock (NCC). The block diagram of the proposed timing recovery algorithm is shown in Figure 3

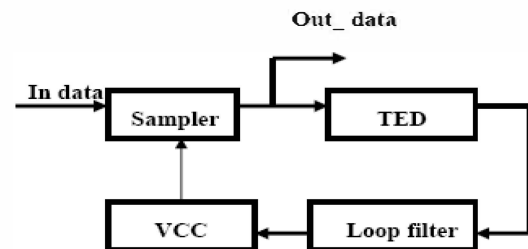


Figure 3 Proposed Timing Recovery Algorithm

The input to the timing error detector splits into three branches: one which has no delay, one which has one cycle of delay, and the final has two cycles of delay. The slope of the input is determined by subtracting the top branch from the lower branch. If the slope is zero, the clock is locked. The center branch is multiplied by the slope to determine if the slope is positive or negative. Thereby, the algorithm will sample either earlier or later until the ideal sampling time is determined. If the derivative of the input is flat, the ideal sampling time has been reached[4]

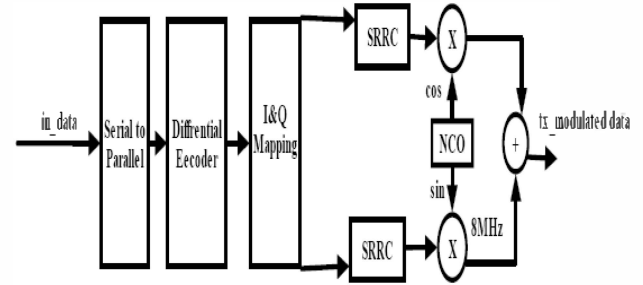


Figure 4 DPSK Transmitter

## 2.2 DPSK Transceiver

### 2.2.1 Proposed DPSK transmitter

The structure of a typical digital DPSK transmitter is shown in Figure 4. The transmitter consists of two branches: one for the In-phase (I) channel and one for the quadrature (Q) channel. The operation of the transmitter can be understood by tracing the flow of data through the functional blocks inside the transmitter. This modulator is the same for both  $\pi/4$ -DQPSK and 8-DPSK except for the serial to parallel clock (2MHz for  $\pi/4$ -DQPSK, 3MHz for 8DPSK) and in in-phase quadrature mapping rules. The Serial to Parallel block converts incoming serial data into two N/2 bit words per symbol for  $\pi/4$ -DQPSK and N/3 bit words per symbol for 8DPSK and Hence, the symbol rate is 1/N times the bit rate. The data is then differentially encoded at the symbol rate and then fed to in-phase and quadrature phase mapper. The mapped values are subsequently filtered by SRRC filters to limit the bandwidth of the transmitted signal without introducing Inter-Symbol Interference (ISI).

### 2.2.2 Proposed DPSK Receiver

The structure of a typical digital DPSK receiver implemented in digital logic is shown in Figure 5. There is no non-coherent detection equivalent for DPSK as non-coherency implies no phase information. The operation of the receiver can be understood by tracing the flow of data through the functional blocks inside the receiver. The demodulation process is the same for both  $\pi/4$ -DQPSK and 8-DPSK except for the sampler clock, slicer (de-mapping rules), and parallel to serial clock. The demodulation process involves the mixing with the same modulation carrier frequency as used in the transmitter part, after the mixing the signal is low pass filtered to obtain sine and cosine components of the phase and then we get the ARCTAN of the upper and lower branches, it then sampled and enter the decision device (slicer) to determine which symbol was transmitted [5].

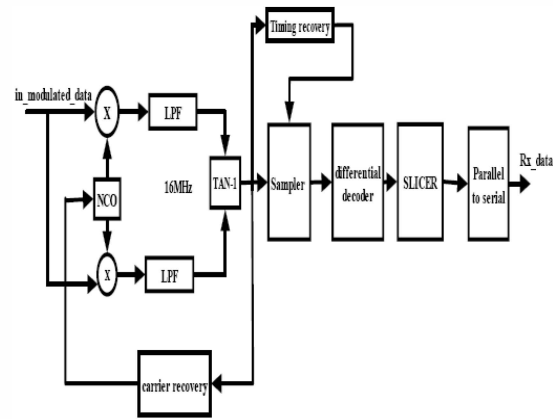


Figure 5 Proposed DPSK Receiver

### 2.2.3 Proposed Carrier Recovery Algorithm

If the receiver's and transmitter's clocks are misaligned, the resulting can impair reception on either arm of the receiver. Thus, a carrier recovery algorithm is required. The proposed Carrier recovery algorithm is shown in Figure 6 where the phase error is generated from the in-phase and quadrature phase branches. This scheme was preferred because it gives more accurate results; the carrier recovery algorithm consists of a numerically controlled oscillator, a phase detector and a second order loop filter[6].

The loop filter is used to filter the phase error signal in order to provide the required correction to the NCO. From control theory, it is known that a proportional path can be used to track out a phase error; however, it cannot track out a frequency error. For a carrier recovery loop to track out a sampling frequency error, a loop filter containing an integral path is needed. The integral path multiplies the error signal by an integral gain  $K_i$  and then integrates the scaled error using an adder and a delay block. A second-order filter can track out both a sampling phase and a sampling frequency error as shown in figure 7.

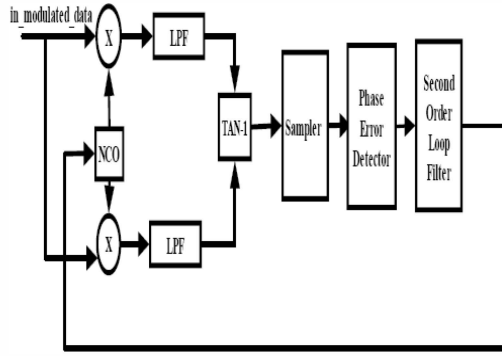


Figure 6 Proposed Carrier Recovery

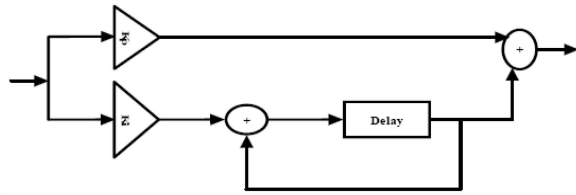


Figure 7 Second order Loop Filter

## 2.3 Bluetooth2.0 Transceiver

Here we integrate both transceiver on a common hardware platform .The Bluetooth 2.0 transceiver Block Diagram is shown in Figure 8,9. At the transmitter side It can be noticed that different modulation schemes for Bluetooth 2.0 Transceiver can be chosen using multiplexers ("00" GFSK,"01"  $\pi/4$ DQPSK,"10" 8DPSK),. At the receiver side the same low path filters and the same ARCTAN function block are used for both GFSK and DPSK transceiver which saves hardware area that is why we chose this architecture as the receiver part for GFSK Transceiver as our target is low cost Bluetooth2.0 transceiver chip, but in case of GFSK the data is differentiated after the ARCTAN block. The GFSK or DPSK are then sampled and enter the decision maker block, in case of DPSK they finally pass through serial to parallel converter. The carrier recovery and timing recovery blocks are included in the receiver side too.

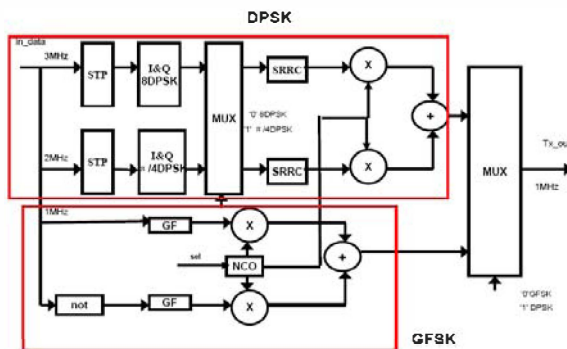


Figure8 Integration of Bluetooth 2.0 Transmitter

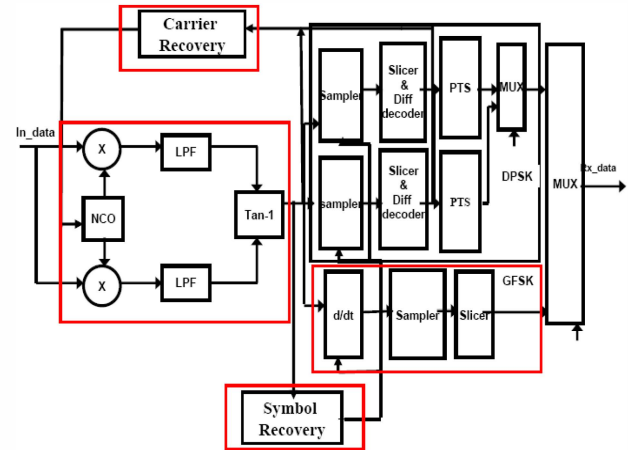


Figure 9 Integration of Bluetooth 2.0 Receiver

## 3 Hardware Implementation

### 3.1 VHDL Implementation of Bluetooth2.0

#### 3.1.1 VHDL Model for NCO

The VHDL model for the numerically controlled oscillator can be shown in Figure 10 Its design is based onCORDIC theory (rotation mode) as it more area effective than using look-up tables for sine and cosine. It consists of a phase accumulator, first quadrant adjustment; a pipelined unrolled CORDIC which consists of subsystem of thirteen iterations of a CORDIC base block, sine-cosine rebuilds, and delay sub-modules. And the detailed block diagram is shown in figure 11[7].

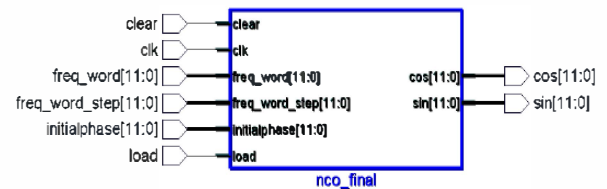


Figure 10 VHDL Model for the Numerically Controlled Oscillator

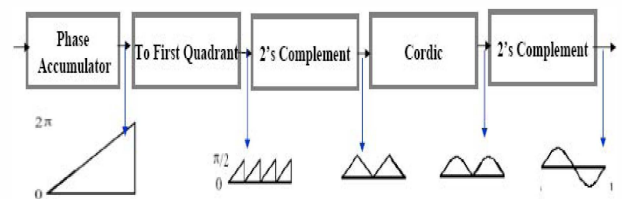


Figure 11 detailed diagram of NCO

### 3.1.2 VHDL Model for ARCTAN Function

The ARCTAN block is implemented using the CORDIC Algorithm. CORDIC algorithm has two modes of operation: Rotation mode (to generate sine and cosine functions) and vectoring mode (to calculate ARCTAN function). as shown in table2.

### 3.1.3 VHDL Model for Digital Filter

The coefficients and the VHDL codeS were generated using FDATAOL in matlab7 (65Taps filter). The design was generated using the windowing method (sampling frequency is 16MHz and the cutoff frequency is 0.5 MHz). The FDATAOL method was preferred over hand – crafted filter method because the outcome consumes less area. It was preferred over using Xilinx core generator because it is technology–dependent (maps only into Xilinx chips). Table3 shows a comparison between designing the FIR filter using hand-crafted method, FDATAOL method and Xilinx core generator. Synthesis results refer to mapping into a spartan3 (200g256) chip.

### 3.1.4 VHDL Model for Timing recovery

The timing recovery block consists of timing error detector (based on early-late gate method), second order loop filter (to filter the timing error in order to provide the required correction to the numerical controlled clock, and a numerically controlled clock (NCC). This is implemented in the same way as the NCO except that it generates square wave rather than a sinusoidal. The top level of the VHDL entity model of timing recovery is shown in Figure 2 -.The RTL level of the timing recovery block is shown in Figure 12.



Figure 12 Top Level Entity for Timing Recovery

Table2 CORDIC Modes of operation

Rotation mode	Vectoring mode
$X_{n+1} = X_n - S_n 2^{-2n} Y_n$ $Y_{n+1} = Y_n + S_n 2^{-2n} X_n$ $Z_{n+1} = Z_n - S_n \arctan(2^{-2n})$ $S_i = \begin{cases} -1 & \text{if } z_i < 0 \\ +1 & \text{otherwise} \end{cases}$ <p>➤ After K iterations (K = F(Accuracy))</p>	$s_i = \begin{cases} -1 & \text{if } y_i < 0 \\ +1 & \text{otherwise} \end{cases}$ <p>➤ The difference from rotation mode is that direction of rotation is determined by the sign of y instead of z</p>
<div> <div>0.6072</div> <div>0</div> <div>z</div> </div> <div> <div>Xi</div> <div>Yi</div> <div>Zi</div> </div> <div> <div>xi+1</div> <div>yi+1</div> <div>zi+1</div> </div> <div> <div>cos z</div> <div>sin z</div> </div>	<div> <div>x</div> <div>y</div> <div>0</div> </div> <div> <div>Xi</div> <div>Yi</div> <div>Zi</div> </div> <div> <div>xi+1</div> <div>yi+1</div> <div>zi+1</div> </div> <div> <div>Tan<sup>-1</sup>(y/x)</div> </div>

Table3 Comparison of Different Methods of Implementation FIR Filter

	Hand-crafted	FDATOOL	Core generator
Number of 4 input LUTS	6 %	2.5 %	2%

### 3.1.5 VHDL Model for Carrier recovery

The phase-locked loop (PLL) is a critical component in coherent communications receivers that is responsible for locking on to the carrier of a received modulated signal. A PLL adjusts the phase of a numerically-controlled oscillator to match that of the received signal. The top level entity of carrier recovery consists of exact phase and received phase as input to calculate phase error then the error is fed to the loop path filter to generate the filtered error signal that control the numerically controlled oscillator .

## 3.2 Verification

The testing can be divided into two phases, a simulation phase where all testing is done on the PC and a hardware phase where testing is done on the hardware.

### 3.2.1 Simulation Verification

Modelsim tests were run when changes were made to the system; test benches were developed to read test input data produced by a MATLAB model and to generate MATLAB-readable output data. Data representation could then be changed to allow comparison with reference data produced by the MATLAB model and error patterns were studied to determine the cause of errors.

### 3.2.2 Hardware Verification

All the designs were thoroughly tested on the FPGA using Xilinx Chip-Scope logic analyzer We compare sent symbol with received symbols for both GFSK and PSK Transceiver and calculate the bit error rate .

## 3.3 Performance and Related Work

The tests have been passed successfully. The SIMULINK BER versus SNR curve and VHDL Measured/simulated Eb/N0 versus SNR for Bluetooth2.0 Transceiver are shown in Figure 4-13, these curves are drawn at frequency offset=150 KHz , phase offset =0.1 RAD and timing offset=0.1 us.

A comparison of my results with previous results obtained from other theses is shown in table4 and table 5.

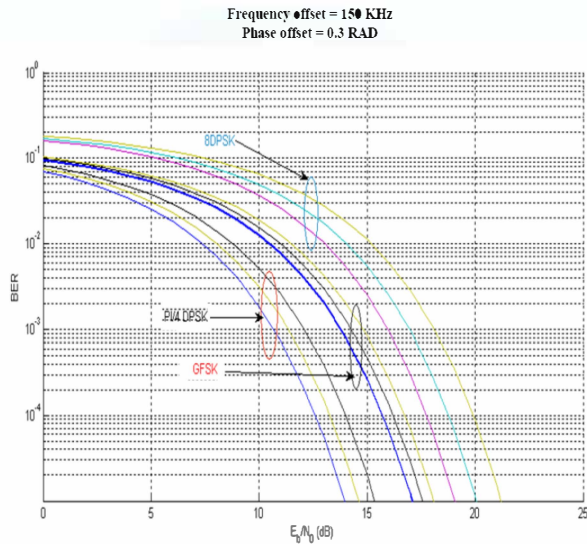


Figure 4-13 Eb/N0 Vs BER Curves (GFSK & DPSK)

Table 4 Comparison of Simulation Results of GFSK Transceiver with other technologies at frequency offset=0

Design	Eb/N0	Technology
My Design	12.3	FPGA
[5] 2004	20	0.18um CMOS
[6] 2003	14.8	Matlab simulation
[7] 2004	14	C simulation
[8] 2005	15.5	0.18 um CMOS
[9] 2007	16	0.25 um CMOS

Table 5 Comparison of Simulation Results of DPSK Transceiver

Design	$\pi/4$ DQPSK	8DQPSK	Technology
My Design	10.8	14.8	FPGA
[7] 2004	13	19	C simulation

### 3.4 Implementation Results

Mapping result for Bluetooth2.0 Transceiver (Xilinx spartan3 (400pq 208) is shown in Table 6 and Utilization of Each Component in Bluetooth2.0 Transceiver (3,113) is shown in Table 7

Table 6 Mapping result for Bluetooth2.0 Transceiver

Resource type	used	available	utilization
Logic utilization			
Number of slice flip flops	2,663	7,168	37%
Total number of 4 input LUTs	4,922	7,168	68%
Number used as logic	3,781		
Number used as route-thru	1,211		
Number used as shift registers	8		
Number of mult 18 x18	8	16	50%
Number of GCLKS	1	8	12%

Table 7 Utilization of Each Component in Bluetooth2.0

Component	Utilization	Total number	Total utilization
Numerically Controlled Oscillator	8 %	2	16%
Numerically Controlled clock	8 %	1	8 %
Arctan function block	6 %	1	12 %
Gaussian Filter	1 %	2	2%
SRRC filter	6 %	2	12%
Low pass filter	6 %	2	12%
Timing recovery	2.5 %	1	2.5 %
Carrier recovery	1.5 %	1	1.5 %
Others	2 %	1	2 %
route-thru	32%		32%
			100 %

## 4 Conclusion

Our Bluetooth transceiver design is encoded in the VHDL hardware description language, and implemented successfully on Spartan 3 Xilinx FPGA. The performance of the transceiver is experimentally tested using the Xilinx FPGA-embedded ChipScope logic analyzer. The measurement results meet Bluetooth specification v2.0+EDR and show the suitability of the presented design approach. Non data aided timing recovery improves the BER performance by as high as 3dB at an area penalty of ~ 15 % of the total design size.

Verifications at worst case frequency offset show an error rate performance of  $10^{-3}$  at Eb/N0 of 15 dB for GFSK,  $10^{-4}$  at Eb/N0 of 13.4 dB for  $\pi/4$ DQPSK and  $10^{-4}$  at Eb/N0 of 20 dB for 8DPSK.

## 5 References

- [1] Specification of the Bluetooth System, Core Specifications, version 2.0+EDR, v1.2, v1.1; <https://www.bluetooth.org/spec/>
- [2] Bluetooth SIG "Specification of the Bluetooth System [http://www.bluetooth.Com/developer/specification/Bluetooth\\_11\\_Specifications\\_Book\(2001\)](http://www.bluetooth.Com/developer/specification/Bluetooth_11_Specifications_Book(2001))
- [3] John A.C Bingham, "The Theory and practice of MODEM design." John Wiley& Sons , 1988.
- [4] [W.HAN "The Application of Multirate Techniques for Synchronization in Fully Digital Demodulation" 1995
- [5] H.Savla "Design and Simulation of a Low Power Bluetooth Transceiver" Master Thesis, Sardar Patel University, India, jan 2004.
- [6] R.Schiphorst, F.Hoeksema and K.Slump "Bluetooth demodulation algorithms and their Performance" University of Twente, 2003
- [7] Alberto Gozzi" Complexity and performance analysis of the digital modem of the Enhanced Data Rate Bluetooth standard " Pisa University,2004
- [8] Henk Jan "A Low power Highly Digitized Receiver for 2.4 Band GFSK Applications" IEEE Transactions on Microwave Theory and Techniques, Feb 2005.
- [9] Andre Neubauer "A Digital Receiver Architecture for Bluetooth in 0.25 um CMOS Technology and Beyond" IEEE Transactions on Circuits and Systems, Sept 2007