Software-Defined Radio using Xilinx (SoRaX)

Functional Requirements List and Performance Specifications

By:

Anton Rodriguez
&
Mike Mensinger

Project Advisors:

Dr. In Soo Ahn
&
Dr. Yufeng Lu

December 31, 2009

Introduction

Software Defined Radios (SDR) are highly configurable hardware platforms that provide the technology needed to realize the rapidly expanding third (and future) generation digital wireless communication infrastructure. The concept for Software Defined Radio (SDR) is not new. However, its capabilities and applications are ever expanding. Digital communication systems adopt more sophisticated coding and modulation technologies to meet ever-increasing demand of audio, video, and data services. The SDR is a solution for rapid integration of various emerging wireless standards and technologies.

The objective of this project is to design a communication radio system on the FPGA board. The main focus will then lie on the carrier synchronization, and symbol phase ambiguity correction based on the recovered signal.

Project Goals

(1) Gain an in-depth understanding about the FPGA implementation of carrier synchronization.
(2) Create a test signal of known hard-coded values (preamble) to estimate the channel state.
(3) Regenerate the carrier and symbol timing to decode the transmitted digital data.
(4) Achieve fast acquisition of carrier synchronization and symbol timing through efficient Xilinx programming.
(5) Construct a working Simulink model.
(6) Implement the Simulink model on the FPGA board.
(7) If time permits, implement different modulation schemes (i.e. QAM).

Block Diagram

The high level flowchart is shown in Figure 1. The input to the communication system includes a stream of data, for example, sound, image, or computer-generated data. The quadrature phase-shift keying (QPSK) will be used to modulate the data stream. This modulated data stream is then transmitted wirelessly, using two antennas linking the transmitter and receiver on the SignalWave Virtex II FPGA board. The channel will introduce noise and distortions due to the multi-paths.

The QPSK constellation is shown in Figure 2. Due to channel imperfections, phase error is introduced in the received signal causing an incorrect representation of the transmitted data. In order to recover the data, a coherent detection is required and is achieved through a digital phase lock loop. The coherent detection requires estimation of the instantaneous phase offset of every data point in the constellation grid with respect to where it should be. That information is then fed to a direct digital synthesizer (DDS), which generates coherent sine and cosine carriers that correct the phase error. Upon completing the carrier synchronization and symbol timing, the data can be decoded. QPSK demodulation will be subject to phase ambiguity. This can be resolved by transmitting a preamble sequence or differentially encoding the data. The overall system will be implemented and verified using Simulink. Then it will be downloaded to an FPGA board using Xilinx tools.

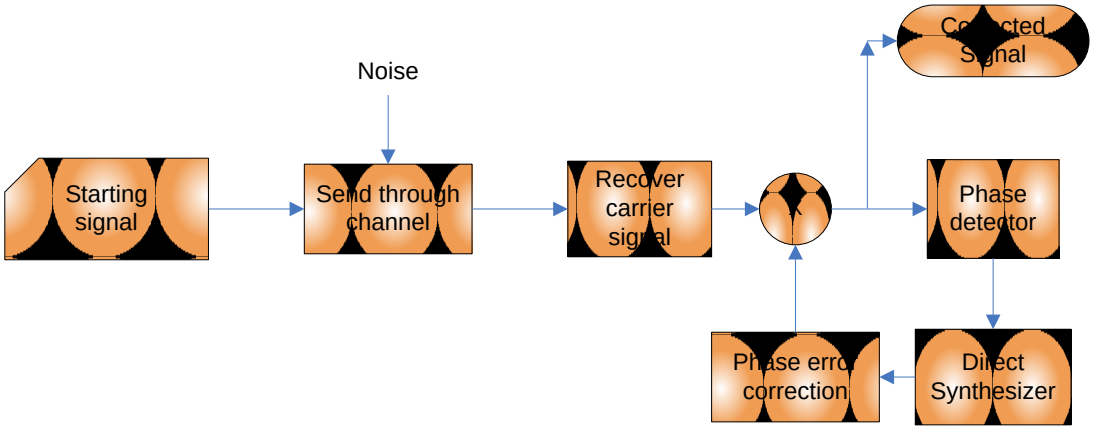

Figure 1 High Level Flowchart

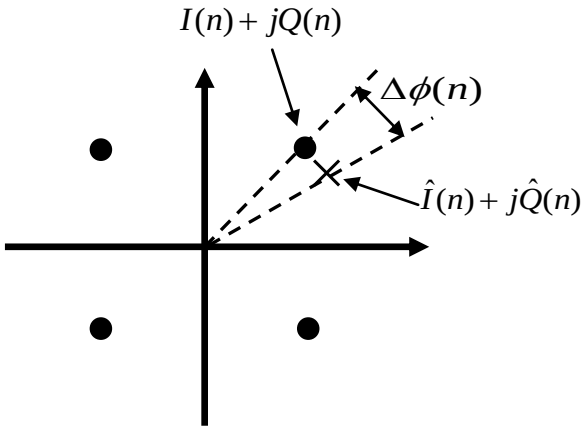

Figure 2 QPSK Constellation [1]

Functional Requirements & Specifications

The success of this project lies heavily on the construction of an accurate and implementable Simulink model. In order to do so, there are several different functional requirements and specifications to follow:

- The model shall operate with a system clock of 50 MHz.
- There shall be an explicit sampling period of 1/8 throughout the Simulink model.
- The sampling frequency for the model should be 12.5 MHz which is ¼ of the system clock. (This provides a convenient clock signal).
- We shall use a DDS compiler 4.0 to simulate our Voltage Controlled Oscillator (VCO) for the Phase-Locked Loop.
- The frequency offset provided should be no larger than 1 kHz.
- The system will have a minimum data rate of 160 kbps.

We will attempt to achieve data encoding for both QPSK and 16-QAM types. Also, we desire that phase-locking shall occur in fewer than 100 samples. This can be accomplished by implementing proper gain values for the proportional controller in the loop filter design.

Equipment List:

The heart of this project revolves around the development of the simulink model. Thus very little equipment will be used, except for the Xilinx ISE 9.2 software, and the Xilinx Virtex-II FPGA board. We are also implementing Windows' remote desktop feature to access Dr. Lu's computer to run Xilinx ISE 9.2 from off campus.

Proposed Spring Semester Schedule:

Weeks	Tasks
(Winter Break)	**Compile a functional Simulink model** **Design Loop Filter**
1 - 2	**Load model onto the Virtex 4**
2 - 5	**Develop training sequence**
5 - 9	**Develop an algorithm to correct phase ambiguity of QPSK symbols**
10 - 13	**Implement 16 – QAM modulation scheme**
13 – 15	**Prepare Final Report and Oral Presentation**

The workload will be distributed as evenly as possible. Anton will focus on the website design and layout as we achieve our project goals. Mike and Anton will collaborate over the winter break to successfully compile a working phase-locked loop and communication system. From there, the Mike will work on developing the symbol phase ambiguity solution. Anton will focus on hardware implementation.

Bibliography

[1] Chris Dick, Fred Harris, and Michael Rice, *FPGA Implementation of Carrier Synchronization for QAM Receivers*, Journal of VLSI Signal Processing, Copyright © 2004 Kluwer Academic Publishers, Netherlands.

[2] Stephens, Donald R. *Phase-locked loops for wireless communications digital and analog implementation*. Boston: Kluwer Academic, 1998. Print.