SDR IN UNDERGRADUATE ENGINEERING EDUCATION

Vuk Marojevic, Ismael Gomez, Joan Olmos, Gabriel Montoro, and Antoni Gelonch

Dept. Signal Theory and Communications
Universitat Politècnica de Catalunya
{marojevic|ismael.gomez|olmos|montoro|antoni}@tsc.upc.edu

ABSTRACT

This paper presents the subject *Digital Techniques for Communications* (DTC). DTC is taught at the Escola Politècnica Superior de Castelldefels (EPSC), which belongs to the Polytechnic University of Catalonia (UPC). It is a compulsory subject for undergraduate electrical engineering students specialized in telecommunication systems. It covers sampling theory, efficient IQ modulation and demodulation, phase-locked loops, and direct digital synthesis. The practical part consists of four laboratory exercises, where students in groups of two gradually implement a simple communications system on a fixed-point digital signal processor. This paper discusses the SDR practices that our students learn during the laboratory session. We conclude with our positive experience of teaching SDR principles to undergraduate students.

1. INTRODUCTION

One of the major goals in university education is providing a clear view of the complexity of current and emerging systems. While technology evolves quickly, the students need to gain enough background to be able to understand and eventually participate in this development. Wireless system engineering, in particular, requires knowledge of emerging technologies and skills in many methodologies, including digital signal processing, software design (programming in C and VHDL), real-time execution, parallel and distributed computing, and data conversion. The Escola Politècnica Superior de Castelldefels (EPSC, http://epsc.upc.edu/en/) of the Polytechnic University of Catalonia (Universitat Politècnica de Catalunya – UPC), therefore, follows a project-based learning concept.

Software-defined radio (SDR) is a modern approach to radio engineering [1]. It adds additional difficulties to wireless communications system design, requiring computer engineering capabilities. Many universities incorporate courses on SDR in their Master and Ph.D. programs. The EPSC considers SDR as part of the undergraduate electrical engineering curriculum; the compulsory subject *Digital Techniques for Communications* (DTC) introduces many SDR principles and practices.

DTC consists of a theoretical and a practical part. The theoretical part includes an introduction to SDR, sampling theory, analog-to-digital and digital-to-analog conversion, multirate digital signal processing, and phase-locked loops. The practical part features four laboratory exercises, where students in groups of two gradually implement a simple transceiver using the code composer studio and the DSP Starter Kit (DSK) TMS320C6416 from Texas Instruments [2]. The DSK features a fixed-point digital signal processor (DSP) and a 16-bit stereo codec with a 48 kHz sampling frequency. The students use a pair of loudspeakers, one acting as the transmitter and the other as the receiver, to complete the setup.

During the laboratory sessions the students learn how to processes samples in real time, how to design cycle-efficient implementations, and manage fixed-point arithmetic, among others. The exercises apply several techniques and concepts learned in preceding courses: C programming, digital filter design, coding and decoding, digital modulation, processing eye and constellations diagrams, visualizing delay and synchronization effects, analyzing BER implications, and so on. The hands-on exercises enable getting a better understanding of the propagation phenomenon and the channel degradation effects. The students explore many digital signal processing techniques for compensating these effects.

This paper presents DTC and discusses our experience of teaching SDR principles to undergraduate students. The course reveals several important implementation issues in modern wireless communications and provides hands-on experience on the basic SDR practices. DTC regularly motivates students to join our research team and contribute to our SDR Framework [5]. Although the graduate education in electrical engineering (M.S. and Ph.D. programs) generally focuses on other topics than SDR, the course provides the means for understanding and considering at least the fundamental implementation issues of wireless communications.

2. COURSE OVERVIEW

DTC is taken by electrical engineering students specialized in telecommunication systems in their second year of undergraduate studies (Spring Term). It was introduced in 2001 and focuses on physical layer implementation aspects of wireless communications systems. The objective is experiencing the difficulty of developing a system that works (not just in theory). The course therefore combines theoretical radio engineering concepts with DSP programming skills. It promotes the SDR principles modularity and flexibility. The students learn how to deal with signal propagation issues, analog distortion, and real-time digital signal processing, among others.

The evaluation is as follows: the midterm and final exams make up 40 % and the laboratory exercises 60 % of the final grade. The evaluation of the laboratory exercises is continuous: There are four laboratory exercises, which are spread over 13 sessions (two hours, weekly). Each exercise consists of 10 sub-exercises. Exercises are scheduled for two, three, or five weeks, depending on the difficulty. Points can be acquired during these sessions, whereas penalties of 30 and 60 % are applied for delays of one and two sessions. Students work in groups of two. There are up to 10 groups per teacher and session.

2.1. Theoretical Part

The theoretical part of DTC is organized under four main topics:

1. Sampling theory and data converters,
2. Efficient IQ modulation,
3. Phase-locked loops (PLLs),
4. Direct digital synthesis (DDS).

The first topic focuses on bandpass sampling and analog-to-digital conversion requirements for SDR systems. The second topic covers QPSK modulation and, specifically, addresses the design of efficient QPSK modulators and demodulators using polyphase filters [3]. The third part covers the basic aspects of PLLs and their efficient implementation [4], whereas the final part explains the DDS concept and some optimization techniques [1].

The theoretical part of TDC provides the necessary theoretical background and explains useful techniques for dealing with real implementation issues, supporting the laboratory exercises. The theoretical course material is adjusted each year and does not cover topics and skills that are taught in other complementary courses taken beforehand, such as digital filter design and C programming.

2.2. Practical Part

The practical part of TDC consists of four hands-on laboratory exercises. These exercises guide the student in gradually building the different transceiver modules that assemble a simple communications system. The objective is applying digital signal processing techniques learned in theory for solving practical problems. Although the teacher introduces each one of the exercises and provides some hints for solving the different problems, the students themselves need to think of possible solutions to arising problems, implement them, and discuss the results with the teacher. The first two exercises are more guided then the third or fourth, where the student should be able to apply what she learned during the course and be able to observe and appreciate her own progress.

The labs are modified each year and regularly exchanged following wireless communications trends. If interested, please contact the authors or check [5] for the course material.

Prior to the four laboratory exercises, Lab-1 to Lab-4, a hands-on session is dedicated to learn how to use the code composer studio (CCS). We briefly introduce all exercises, including the introductary session Lab-0.

2.2.1. Lab-0: Introduction to CCS

Lab-0 is guided by the teacher except for a simple programming task at the end of the session. It explains the setup of a new project, revises the different configuration options (simulator/DSK mode, among others), explains the compile, link and load procedures, introduces breakpoints and some debugging facilities, graphical display and memory view options, and so forth. As a final exercise the students are asked to implement two simple routines for creating two arrays of different data types and visualize the results.

2.2.2. Lab-1: Channel Analysis

Before developing a wireless communications system we need to analyze the transmission channel. The channel here includes the codec, speakers, analog amplifiers, and the radio link. Since each group uses a different pair of speakers, one with and the other without amplification, the channel differs from group to group.

The main objective of this exercise, scheduled for two sessions, is to determine the suitability of the speakers and identify the usable and unusable frequency bands for over-the-air transmission. Distortion usually makes the band below 1 kHz unusable, whereas most (low-cost) speakers start degrading in performance above 7 kHz, approximately.

Two methods are suggested for performing the channel analysis: impulse response and spectrum analysis. The first consists in sending a delta over the channel and capturing the impulse response in the time and frequency domains. The second method generates a sequence of tones which are

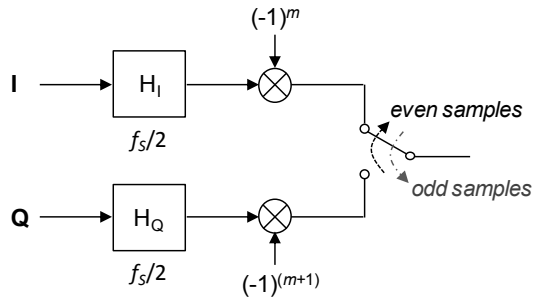

Fig. 2. Efficient QPSK modulator.

sent over the channel computing the magnitude and phase of the received signals (Fig. 1 at the end of the paper). The functions implementing these two methods are provided. The students need to set the proper breakpoints, employ each method individually, capture the results and analyze them. After comparing the results of the two methods, each group follows a methodology for defining their channel model, which they can use for preparing the following laboratory sessions (using the simulation mode since the DSK is only accessible during the laboratory hours).

2.2.3. Lab-2: QPSK Modulator and Demodulator

A simple QPSK modulator/demodulator is developed in Lab-2. The information to be send and received is, initially, a phrase of 32 characters, chosen by each group, and later a bit sequence loaded from an external file. The relation between the symbol and the sampling rate is 1:8, 1:16, or 1:32. The pulse-shaping filter is a route-raised cosine that needs to fulfill the specified inter-symbol-interference characteristics of 20 dB attenuation. The carrier frequency is 5 kHz.

Transmission over the real channel (codec + speakers + radio link) shows the influence of the analog parts of the system and the signal propagation issues. Bit error rate (BER) measurements and analysis is carried out in two ways: bypassing the real channel and including it.

During three sessions the students learn how to implement the basic blocks of the QPSK modulator and demodulator, become a first idea about real-time processing and synchronization issues, and experience the problem of radio transmission and reception.

2.2.4. Lab-3: Efficient QPSK Modulator and Demodulator

Lab-3 is scheduled for three sessions. The students implement the efficient QPSK modulator of Fig. 2 and the corresponding demodulator based on the multi-rate digital signal processing concept. They learn how to design polyphase filters, design a filter bank of four polyphase filters and measure the group delay for choosing the pair of filters for the I and Q paths of the modulator and demodulator.

After successfully implementing the different modules of the transceiver, the students compare the computing cost of the new system with that of Lab-2. They realize a substantial reduction of DSP cycles, using less than 30 % of the DSP resources for real-time digital signal processing (grey blocks in Fig. 3). Further techniques for improving the computing efficiency are discussed in Section 3.

2.2.5. Lab-4: Completing the Communications System

During the remaining five sessions, the students integrate their modules and develop the remaining blocks to complete the system of Fig. 3. This exercise teaches some new concepts, but also fosters the knowledge acquired so far.

First of all and in order to minimize the effects of the channel impulse response, the students design a power ramp while taking the results of Lab-1 into account. At the receiver, a bandpass filter, automatic gain control (AGC), carrier synchronization, phase rotation, and symbol synchronization completes the modular system design.

The students follow a frame-based transmission pattern, where the user information is sent in packets. BER and computing cost analyzes and optimizations conclude this exercise.

3. SDR PRACTICES

The complexity of wireless communications suggests modular system designs following a component-based development approach. Most communications services have hard real-time processing requirements that need to be met.

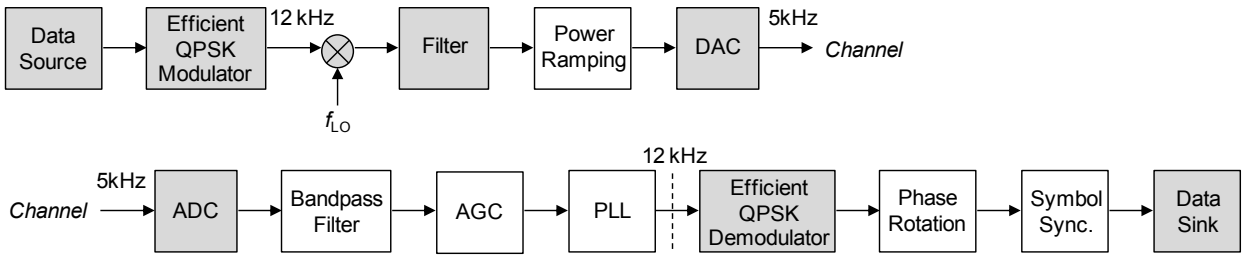

Fig. 3. Transceiver modules of Lab-4.

This claims for an efficient implementation of the physical layer while providing the desired radio performance.

DTC provides hands-on experience with digital signal processing techniques, with emphasis on the basic SDR principles *modularity* and *flexibility*. This section discusses several relevant aspects of SDR that DTC students learn.

3.1. Fixed-Point DSP Programming

Fixed-point DSPs are popular in communications system engineering because of their energy efficiency and high processing capacity. The fixed-point arithmetic, however, introduces new programming challenges to students, which are used to Matlab or other simulation tools. Fixed-point DSPs are inefficient for processing floating point numbers: A single floating-point multiply-accumulate operation (MAC) requires around 50 DSP cycles on the C6416 device, as opposed to one cycle for performing an integer MAC. Floating-point operations should therefore be avoided whenever possible, leading to the next issue: dynamic range.

Integer approximations of filter coefficients, sine and cosine values, and so forth, amplify the signal. Hence, gain control should be considered at each module in the processing chain, adjusting the signal amplitude to reasonable values. The laboratory exercise announcements suggest a dynamic range of 12-16 bits. (Although up to 32 bits are available for processing samples, the codec is limited to 16 bits per sample.) The students are encountered with the dynamic range issue throughout the entire course. The teacher tries to point out the problem, asking for a single-time representation of the corrupted signal and encouraging the students to interpret the results themselves.

The selection of the C6416 fixed-point DSP for the laboratory exercises was intentional to provide practical experience on fixed-point arithmetic and prepare our students for their engineering career.

3.2. Real-Time Processing

Real-time processing is one of the most challenging concepts for undergraduate students. DTC therefore focuses on this issue from the very beginning of the course, not only in the laboratory sessions, but, also, in the theoretical part and the examinations (partial and final exams).

In the lab sessions we insist on a modular transmitter and receiver implementation with special emphasis on processing one sample at a time. The student creates an infinite loop that processes data samples one by one. This way the continuous data flow—characterizing wireless communications—can be properly processed in real-time. The main part of the program thus essentially contains this infinite loop and function calls. Each function (module) processes a single data sample per invocation. The teacher does not debug the students' codes, in general, but takes care that they follow these concepts.

We explain to our students why the amount of resources for processing data samples is limited in a real-time system. In this case, the digital-to-analog converter (DAC) converts one transmitter sample and the analog-to-digital converter (ADC) generated one receiver sample each T_s = 1/48000 = 20.83 µs. With a DSP clock frequency of 1 GHz, there are roughly 20 800 cycles for the transmitter and receiver processing blocks. This obliges the student to be aware of the cycles consumed by each module, identify critical modules, and apply the corresponding optimizations if necessary.

Each time the codec is accessed by calling the *codec_io()* function, one sample is written to the DAC and one is read form the ADC at regular time stamps—each 20.83 µs if possible. The students need to organize their code according to this approach and should take care by themselves to not compromise real-time processing. They use the CCS tools for counting DSP cycles and graphically analyze the transmitted and received signals.

The polling approach, where the DSP waits for the codec to consume and produce a sample, is very useful for experiencing real-time processing issues: Samples are lost if the transmitter and receiver digital signal processing functions do not finish within 20.83 µs. An initial version of the laboratory exercises was based on interrupts for converting samples from the digital to the analog domain and vice versa, but this approach did not show the real-time violation effects as clearly to the students as does the polling approach.

3.3. Processing Gain versus Processing Efficiency

Processing efficiency is important in many system development aspects. DTC focuses on applied digital signal processing solutions and, particularly, on the importance of avoiding unnecessary operations, such as multiplication by 0 or excessive attenuation. The students learn how to analyze their modules not only in the sense of functionality and performance, but also in the sense of radio performance (BER) versus computing complexity (DSP cycles). Modules, such as the convolution block, used for pulse shaping and filtering, are designed in Lab-1 and optimized in Lab-3. The channel effects are analyzed in all exercises in order to deploy the necessary digital signal processing for satisfying the radio performance requirements.

Apart from dealing with implementation issues, as discussed in Sections 3.1 and 3.2, the students need to find a good compromise between radio performance and computing complexity, but, also, between computing complexity and programming effort: The importance of the optimization effort rises with increasing system complexity, since more digital signal processing tasks need to be

Table I. Computing cost of the transceiver blocks of Fig. 3.

Module	DSP cycles
Data source	426
Efficient QPSK modulator Pulse shaping and frequency translation to 12 kHz	1687+64
Frequency translation to 5 kHz	82
Image rejection filter	675
Power ramping	67
Channel selection filter	1388
Frame detection Power ramp detection	76
AGC	1158
PLL Carrier recovery and frequency translation to 12 kHz	9930
Efficient QPSK demodulator Matched filtering and frequency translation from 12 kHz to baseband	2006+64
Phase synchronization Constellation rotation	5800
Symbol synchronization Correlation with a 13-bit Barker code	404
Bit decision	100

performed in the real-time loop. One particular course objective is maintaining the number of DSP cycles for implementing the transmitter and receiver blocks of Fig. 3 below 90 % of the DSP capacity. Table I indicates the DSP cycles needed for processing a transmitter and receiver sample. These data have been obtained by one group during the spring term of 2010. The total cycle count is 23 927. Nevertheless, since the AGC, PLL, and phase synchronization do not run concurrently, the overall cycle count is below the 90 % threshold.

4. LESSONS LEARNED

TDC is being given for over seven years already. The many initial errors, especially related with the practical part, have been systematically eliminated. Each year we discuss what the students should learn throughout the course and in each laboratory session and revise the information and material we need to provide. The current methodology and organization of the course provides satisfactory results with room for improvement: Although several students are satisfied about what they take from the course, we are not happy about the students' average score. We have noticed that if a student is not motivated from the beginning, he will struggle at the end of the course. (Some students give up towards the end of the semester.) Therefore, we emphasize that each sub-exercise is important for the evolution of the whole project, which is to design a simple communications system that works in real time. Although the evaluation process takes this into account, penalizing delays, we still observe the lack of consistency from the students' side. Less than 20 % of the students fully complete the final lab and transmit and receive in real time with a decent BER. We thus need work on methods for improving the learning process, especially during this initial phase of the course, where student should obtain the fundamental skills needed for accomplishing the course objective. A fundamental change for the future may be to revise the first laboratory sessions and make them even more guided than they are today.

5. SDR UNDERGRADUATE PROJECTS

DTC is a challenging course. Nevertheless, we have caught the attention of several students to join our research team for doing undergraduate and graduate projects on SDR. These students regularly contribute to our SDR framework ALOE (abstraction layer and operating environment) [5]. They have contributed in porting some ALOE services to DSPs and FPGAs as well as in developing waveform modules for UTRAN and WiMAX. Some 20 students at the EPSC have obtained their B.S. degree in electrical engineering working on an SDR project.

6. CONCLUSIONS

This paper has introduced *Digital Techniques for Communications*, a compulsory undergraduate electrical engineering subject taught at the EPSC. The course provides hands-on experience on fixed-point DSP programming, while gradually implementing a simple communications system following the SDR principles modularity and flexibility. DTC points out some of the real-world problems in wireless communications system design at an early stage of the engineering education. Albeit the extra effort the students need to spend for understanding and taking benefit of DTC, we consider it suitable for preparing student for entering their profession or the grad school.

The grading system obliges students to develop modules that work properly. Students therefore interact, exchanging ideas and experiences while preparing the laboratory sessions. The possibility of extra scores, on the other hand, stimulates positive competition. Although we think that it requires more effort to prepare the laboratory sessions than to study for the exams, the average grade of the practical part is notably higher than that of the theoretical part of the course.

ACKNOWLEDGMENT

We thank the EPSC for their yearlong support. This paper is dedicated to our students.

REFERENCES

[1] J. H. Reed, *Software Radio – A Modern Approach To Radio Engineering*, 2002 Prentice Hall, Upper Saddle River, NJ.

[2] Texas Instruments (TI) Web Site, www.ti.com

[3] L. E. Pellon, "A double Nyquist digital product detector for quadrature sampling," *IEEE Trans. Signal Processing*, vol. 40, no. 7, July 1992.

[4] G.-C. Hsich, J. C. Hung, "Phase-locked loop techniques – A survey," *IEEE Trans. Industrial Electronics*, vol. 43, no. 6, Dec. 1996.

[5] Flexible Wireless Communications Systems and Networks (FlexNets) Web Site, http://flexnets.upc.edu/trac

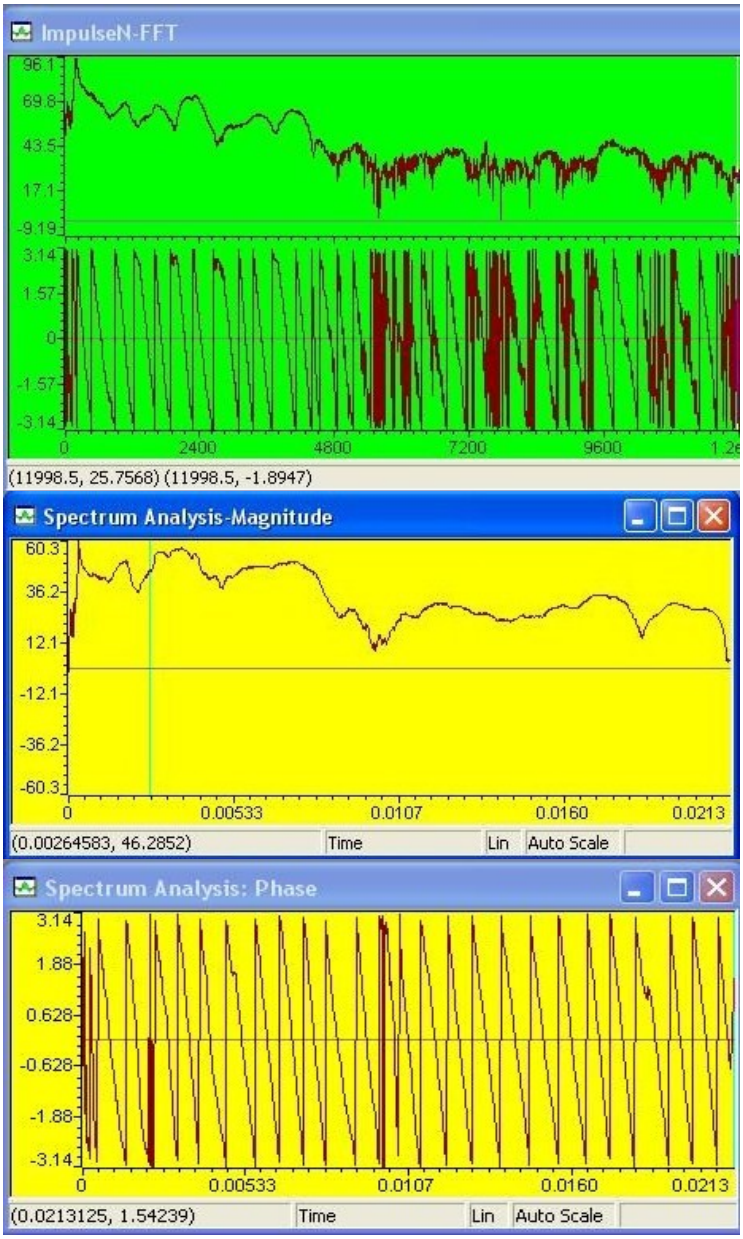

Fig. 1. Channel analysis: The topmost figure shows the magnitude and phase of the channel impulse response after sending a delta over the channel. The bottom figures show the magnitude and phase of the channel obtained with the spectrum-analyzer method. The frequency range is from 0 to 12 kHz in all three figures.