Efficient Multi-Standard Cognitive Radios on FPGAs

Thinh H. Pham*, Suhaib A. Fahmy*, Ian Vince McLoughlin†
*School of Computer Engineering
Nanyang Technological University, Singapore
Email: hung3@e.ntu.edu.sg
†School of Information Science and Technology
University of Science and Technology of China, Hefei, China

***Abstract*—Cognitive radios that support multiple standards and modify operation depending on environmental conditions are becoming more important as the demand for higher bandwidth and efficient spectrum use increases. Traditional implementations in custom ASICs cannot support such flexibility, with standards changing at a faster pace, while software baseband implementations fail to achieve the performance required. Hence, FPGAs offer an ideal platform bringing together flexibility, performance, and efficiency. This work explores the possible techniques for designing multi-standard radios on FPGAs, and explores how partial reconfiguration can be leveraged in a way that is amenable for domain experts with minimal FPGA knowledge.**

I. Introduction

The spectral resource demands of wireless telecommunication systems continue to increase. Cognitive Radios (CRs), that are able to adapt to channel conditions ensuring effective spectrum usage, are an important technology for overcoming this. They are designed to transmit in currently unused spectrum without causing harmful interference to primary users (PUs) or incumbent users (IUs). Apart from the critical issues of spectrum sensing and band allocation, the lower priority of secondary users (SU) raises a challenge in term of transmission capability and quality of service in cognitive radios. When the spectrum allowed for a CR system is fully occupied by PUs and IUs, the transmission of CRs can be blocked. Multiple Standard Cognitive Radios (MSCRs) are able to operate in multiple frequency bands with different specified standards. MSCRs are hence a more flexible generalisation of CRs as they can operate across different bands and standards.

Most practical CRs are built using powerful general purpose processors to achieve flexibility through software, but they can still fail to offer the computational throughput required for advanced modulation and coding techniques and they often have high power consumption. GNU Radio [1] has been a widely used platform in academia. It is a software application that runs on a computer or an embedded ARM processor platform, e.g. on the Ettus USRP E100. Computational limitations mean that while it has been successful for investigating CR ideas, it is not feasible for implementing advanced embedded radios using complex algorithms. Other software based frameworks like Iris [2], have some limited support for FPGAs but suffer from poor bandwidth between software and hardware.

In an application area supporting multiple standards with fast moving standards, custom ASIC implementation is prohibitive. Delorme et al. [3] presented a heterogeneous reconfigurable hardware platform for Cognitive Radio. It can adapt its hardware structure to support standards like GSM, UMTS, wireless LAN. Most processing components run as embedded software on the nodes in a network-on-chip, while the channel coder and the mapping of the RX chain are implemented inside an FPGA. Partial reconfiguration is used to switch the channel coder from one context to an another depending on the SNR. A processor manages data movement between the different processors, the ASIC, and the FPGA. The need for a large data buffer and inefficient data transfer mechanisms lead to increased power consumption and reduced throughput.

Projects at Virginia Tech [4] have shown dynamically assembled radio structures on FPGAs, where the target radio system is defined at a high-level with datapaths connecting relatively large functional modules. The modules are wrapped, and each of them consists of a PR module with compiled partial bitstreams stored in a dynamic library. Using PR eliminates the need for run time compilation, but affords flexibility. A flexible radio controller can insert and remove compiled modules to adapt to current conditions.

Our proposed research focuses on designing the baseband for a MSCR system. This work explores the advantages of coupling PR and parameterised functional units to offer flexibility while minimising reconfiguration time. We aim to design a dynamically configurable radio that supports multiple real world standards and has an abstracted API for upper layers of the radio stack to be accessible to non FPGA designers.

II. Challenges and Proposed Approach

A feasible MSCR requires the ability to switch baseband processing from one standard to another. One way this can be done is for separate implementations of each standard to be implemented as partial bitstreams which are swapped at runtime. However, a large monolithic bitstream entails a long reconfiguration time. Our proposed approach uses a finer granularity. Each module in the processing chain is investigated and commonalities across standards are analysed. For modules requiring only minor modifications, parameterised versions are created. For those that require significant changes, PR can be

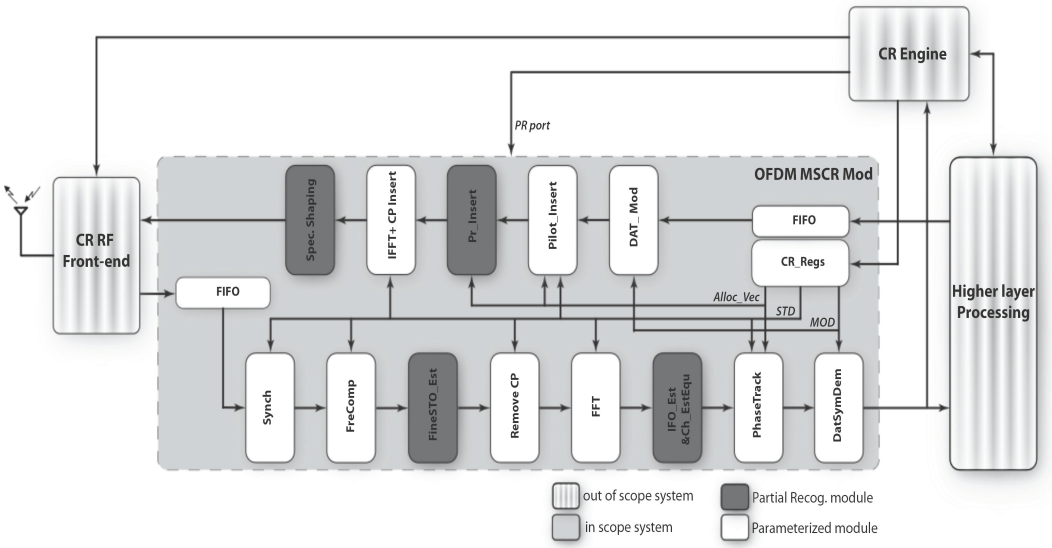

Fig. 1. The structure of an MSCR system.

used on a per-module level. As a result, when switching from one standard to another, only part of the FPGA needs to be reconfigured.

Multi-carrier modulation techniques offer an ideal opportunity for such systems due to their regularity and parameterisation. Orthogonal Frequency Division Multiplexing (OFDM) and Filter Bank Multi Carrier (FBMC) are two types of multi-carrier modulation. OFDM has been the dominant technique adopted for many standards and has been investigated in terms of spectral sensing and carrier allocation for CRs. The wide configuration space in OFDM systems means such a radio can support existing standards like 802.11, 802.16, and 802.22, as well as supporting future OFDM-based standards.

Fig. 1 illustrates the proposed structure of our OFDM-based MSCR. A mix of partially reconfigurable and parameterised modules make up the baseband implementation. FIFOs are included to help overcome the reconfiguration latency when PR modules are reconfigured. Since these modules are now just a small part of the system, buffering is significantly reduced over a more general implementation.

Two key components of the baseband are the synchroniser and spectral shaper. OFDM systems typically tolerate a small carrier frequency offset (CFO) leading to strict constraints on the design of the RF front-end. In an MSCR system, the RF front-end should access a wide range of frequencies depending on the standard in operation. Such a precise and yet wide ranging RF front-end would be very expensive. We have investigated new synchronisation methods that are robust to higher CFO for this purpose [5]. CRs also demand small spectral leakage for both in-band and out-of-band transmitted signals requiring some attention with OFDM. Flexibility in the baseband can allow a frequency guard extending technique to achieve spectral leakage requirements [6].

The interface to higher layer processing is another important factor. Many hardware radio platforms are extremely difficult to design with. Hence, only hardware experts can use them. While optimisation of low level blocks is important, providing a general interface for implementing higher layer processing is important. This allows radio experts to use the system to investigate CR techniques without the need for low-level FPGA expertise. We aim to build a standardised software interface for this purpose that simplifies the process of retrieving systems status and initiating reconfiguration.

III. Conclusion

Our research explores the feasibility of designing low power, low cost multi-standard radios to improve bandwidth efficiency and avoid spectrum congestion. Our proposed system is based on OFDM and implemented on an FPGA, coupling parameterised modules and PR modules to achieve flexibility while minimising reconfiguration time.

References

[1] Free Software Foundation, Inc. (2009) GNU Radio - the GNU software radio. [Online]. Available: http://www.gnu.org/software/gnuradio/

[2] P. Sutton, J. Lotze, H. Lahlou, S. Fahmy, K. Nolan, B. Özgül, T. Rondeau, J. Noguera, and L. Doyle, "Iris – an architecture for cognitive radio networking testbeds," *IEEE Communications Magazine*, vol. 48, no. 9, pp. 114–122, Sep 2010.

[3] J. Delorme, J. Martin, A. Nafkha, C. Moy, F. Clermidy, P. Leray, and J. Palicot, "A FPGA partial reconfiguration design approach for cognitive radio based on NoC architecture," in *Joint Int. IEEE Northeast Workshop on Circuits and Systems and TAISA Conf.*, June 2008, pp. 355–358.

[4] P. Athanas, J. Bowen, T. Dunham, C. Patterson, J. Rice, M. Shelburne, J. Surıs, M. Bucciero, and J. Graf, "Wires on Demand: run-time communication synthesis for reconfigurable computing," in *Proceedings of Int. Conf. on Field Programmable Logic and Applications*, 2007.

[5] T. H. Pham, I. V. McLoughlin, and S. A. Fahmy, "Robust and efficient OFDM synchronization for FPGA-based radios," *Circuits, Systems, and Signal Processing*, vol. 33, no. 8, pp. 2475–2493, 2014.

[6] T. H. Pham, I. V. McLoughlin, and S. A. Fahmy, "Shaping spectral leakage for IEEE 802.11p vehicular communications," in *Proceedings of the IEEE Vehicular Technology Conference (VTC Spring)*, 2014.