

A Reconfigurable Radio Architecture for Cognitive Radio in Emergency Networks

Qiwei Zhang, Andre B.J. Kokkeler and Gerard J.M. Smit
 University of Twente, Enschede, The Netherlands
 Email: Q.Zhang@utwente.nl

Abstract—Cognitive Radio has been proposed as a promising technology to solve today's spectrum scarcity problem. Cognitive Radio is able to sense the spectrum to find the free spectrum, which can be optimally used by Cognitive Radio without causing interference to the licensed user. In the scope of the Adaptive Ad-hoc Freeband (AAF) project, an emergency network built on top of Cognitive Radio is proposed. New functional requirements and system specifications for Cognitive Radio have to be supported by a reconfigurable architecture. In this paper, we propose a heterogenous reconfigurable System-on-Chip (SoC) architecture to enable the evolution from the traditional software defined radio to Cognitive Radio.

Index Terms—Cognitive Radio, Emergency Network, Spectrum Sensing, OFDM, Heterogenous Reconfigurable System-on-Chip, Montium, Design Methodology.

I. INTRODUCTION

Recent studies show that most of the assigned spectrum is underutilized while the increasing number of wireless applications leads to a spectrum scarcity. Cognitive Radio [1] is proposed as a technology to solve the imbalance between spectrum scarcity and spectrum underutilization. In Cognitive Radio, spectrum sensing is done to locate unused spectrum segments and optimally use these segments without harmful interference to the licensed user. This technology is also mentioned in [2] as *Spectrum Pooling*.

Our research on Cognitive Radio is undertaken in the Adaptive Ad-hoc Freeband (AAF) project [3]. The goal of the project is to demonstrate an ad-hoc wireless communication network based on Cognitive Radio used for emergency situations. Current emergency services rely much on the public networks which is not reliable in emergency situations where the public networks can get overloaded. Moreover, public networks like GSM/GPRS are vulnerable to the destruction of the infrastructure. The major limitation of emergency networks is spectrum scarcity [4]. The idea of applying Cognitive Radio to the emergency network is to alleviate this spectrum shortage problem by dynamically accessing free spectrum resources. This implies that the radio has to work in multi-band, cope with various wireless channels and support various services such as voice, data and video. The basic requirement for Cognitive Radio is that it has a reconfigurable architecture to support multi-band and adaptive operations. In this paper, we will introduce a heterogeneous reconfigurable architecture to enable the evolution from the traditional software defined radio to Cognitive Radio.

The paper is organized as follows. In section II some system requirements for a Cognitive Radio based emergency network

are discussed. We will focus on the new features in the physical layer of Cognitive Radio in section III and section IV. A heterogeneous reconfigurable architecture to support Cognitive Radio is proposed in section V. The last section concludes the paper.

II. COGNITIVE RADIO SYSTEM REQUIREMENTS

Cognitive Radio based emergency networks have different requirements compared to ordinary networks. Some requirements for the AAF Cognitive Radio have been presented in [5]. Here we mention some physical layer related requirements: 1) the multiple services should be supported including real-time voice, data message, still pictures and video. To support multiple services, different constraints on QoS have to be met. 2) the radio needs to be robust to combat bad physical channel conditions. 3) energy-efficiency is a concern because the battery life of radio devices can be a limitation for successful operations. 4) the radio should be operational in presence of intentional jamming. All these requirements have to be supported by a flexible and reconfigurable radio architecture.

III. SPECTRUM SENSING

In order to identify the licensed user and locate unused spectrum, the system has to sense the spectrum. For spectrum sensing, three signal processing techniques are commonly used: matched filtering, energy detection and cyclostationary feature detection [6]. Matched filtering is an optimal way for signal detection in communication systems. However, it requires prior knowledge on the licensed user signal which may not be available. Energy detection is often used to determine the presence of signals without prior knowledge. However, there are limitations for the energy detection: 1) the decision threshold is subject to changing signal to noise ratios. 2) it can not distinguish interference from a user signal. 3) it is not effective for signals whose signal power has been spread over wideband. Therefore, the power detection is not adequate for spectrum sensing. Cyclostationary feature detection is used to extract signal features in the background of noise. This technique is a promising option for the spectrum sensing of Cognitive Radio, especially in the situation where energy detection is not so effective. Cyclostationary feature detection is a combination of a FFT and spectral correlation in Figure 1. Because the computational complexity of cyclostationary feature detection ($O(N^2 + \frac{N}{2} \log_2 N)$) is much

larger than energy detection ($O(\frac{N}{2} \log_2 N)$), it is better to use cyclostationary feature detection as a complimentary option when energy detection fails than to build a dedicated feature detector. A spectrum sensing system with different frequency resolutions, which is achieved by using variable sized FFT, can also be beneficial in terms of performance and computational power. The proposed spectrum sensing architecture needs to be supported by a reconfigurable platform where the size of FFT can be reconfigurable and the processing elements for spectral correlation can be switched on/off for different techniques.

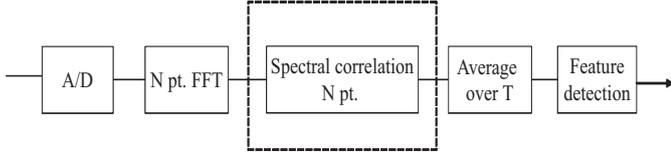


Fig. 1. System level architecture of energy detection with the cyclostationary feature detection option

IV. OFDM BASED COGNITIVE RADIO BASEBAND SYSTEM

Theoretically, an OFDM based Cognitive Radio system is optimal in the sense that it approaches the Shannon capacity in the segmented spectrum by adaptive resource allocation including adaptive bit loading and adaptive power loading. Information bits are loaded as different modulation types onto each subcarrier depending on the subcarrier's signal to noise ratio and its availability to Cognitive Radio. For example, we could minimize system power under the constraint of a constant data rate. We formulate it as follows:

$$\begin{aligned}
 & \text{Min} \sum_{k=1}^K p_k = P_{total} \\
 & \text{Subject to: } R = \sum_{k=1}^K \frac{F_k}{K} \log_2 \left(1 + \frac{h_k^2 p_k}{N_0 \frac{B}{K}} \right) \\
 & F_k = \{0, 1\} \text{ for all } k \\
 & p_k = 0 \text{ for all } k \text{ which satisfies } F_k = 0
 \end{aligned} \quad (1)$$

where R is the data rate; K is the number of the subcarriers. N_0 is the noise power density, B is the band of the interest for Cognitive Radio, h_k is the subcarrier gain and p_k is the power allocated to the corresponding subcarrier. F_k is the factor indicating the availability of subcarrier k to Cognitive Radio, where $F_k = 1$ means the k th carrier can be used by Cognitive Radio. A simulation result of adaptive bit loading is shown in Figure 2. In the simulation, we assume a constant data rate of 400 bits for 512 subcarriers in a frequency selective channel. A binary spectrum mask is given by the higher layer to indicate the spectrum availability, a 1 means the subcarrier is available to Cognitive Radio.

An OFDM radio is also easy to integrate with spectrum sensing because both of them use FFT cores. Cognitive Radio can operate in different frequency bands, provide multimedia services with various QoS and cope with different channel conditions, therefore Cognitive Radio uses different OFDM systems. Although there are a lot of differences between

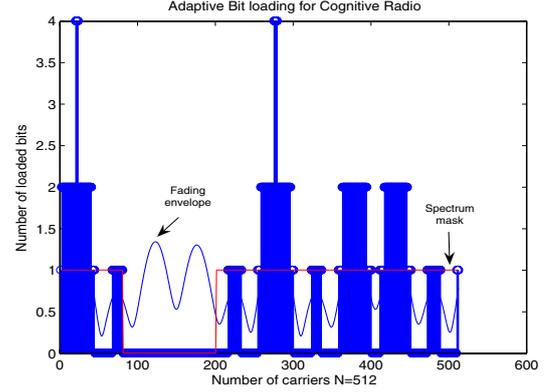


Fig. 2. Adaptive bit loading for Cognitive Radio

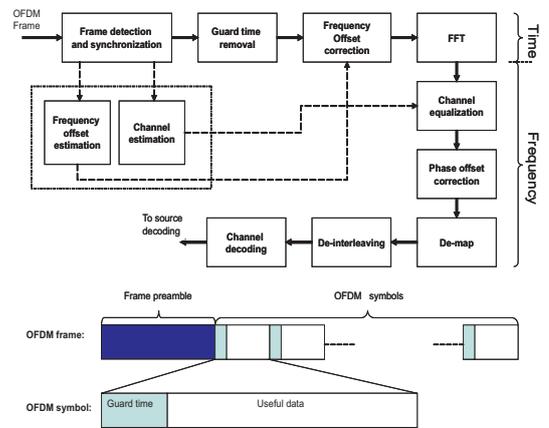


Fig. 3. Basic OFDM processing tasks

various OFDM systems, the basic baseband processing is rather similar. This makes OFDM based Cognitive Radio feasible to be implemented. An OFDM baseband receiver generally consists of the basic tasks shown in Figure 3. For simplicity we only show the tasks performed by the receiver. The transmitter basically does the inverse operations which are less complex. These functional blocks can be found in all OFDM systems, but different standards may use different algorithms or parameters for each functional block. This means that the system can select an algorithm to perform each function depending on its requirements. For example, different channel encoders/decoders (codecs) can be applied to achieve different QoS requirements. For a specific algorithm, there are also opportunities for adaptivity by changing parameters of the algorithm. For example the size of FFT can be tuned by different standards or modes. A fully reconfigurable hardware platform will be an ideal solution to achieve the adaptivity for different OFDM systems.

V. A HETEROGENEOUS RECONFIGURABLE ARCHITECTURE FOR COGNITIVE RADIO

As already foreseen by Mitola [1], a Cognitive Radio is the final point of software-defined radio platform evolution: a fully reconfigurable radio that changes its communication functions depending on network and/or user demands. His definition

on reconfigurability is very broad and we only focus on the reconfigurable hardware platform for Cognitive Radio. In this section, we present a heterogeneous reconfigurable hardware platform for Cognitive Radio.

A. A Heterogeneous Reconfigurable System-on-Chip Architecture

With the evolution of semiconductor technology, more and more transistors can be integrated on a single chip which makes it possible to build large systems on a chip level. This approach is called System-on-Chip (SoC). The reconfigurable platform we propose for Cognitive Radio is a heterogeneous reconfigurable SoC architecture shown in Figure 4. This SoC is a heterogeneous tiled architecture, where

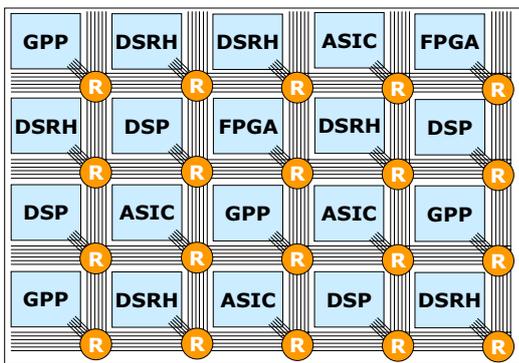


Fig. 4. An example of a heterogeneous System on Chip (SoC). DSRH = Domain Specific Reconfigurable Hardware

tiles can be various processing elements including General Purpose Processors (GPPs), Field Programmable Gate Arrays (FPGAs), Application Specific Integrated Circuits (ASICs) and Domain Specific Reconfigurable Hardware (DSRH) modules. The tiles in the SoC are interconnected by a Network-on-Chip (NoC). Both the SoC and NoC are dynamically reconfigurable, which means that the programs (running on the reconfigurable processing elements) as well as the communication links between the processing elements are configured at run-time. Different processing elements are used for different purposes. The general purpose processors are fully programmable to perform different computational tasks, but they are not energy-efficient. The dedicated ASICs are optimized for power and cost. However, they can not be reconfigured to adapt to new applications. FPGAs which are reconfigurable by nature, are good at performing bit-level operations but not that efficient for word level DSP operations. The Domain Specific Reconfigurable Hardware (DSRH) is a relatively new type of processing element, where the configurable hardware is tailored towards a specific application domain. The Montium [7] tile processor (see Figure 5) developed at the University of Twente is an example of DSRH. It targets the digital signal processing (DSP) algorithm domain, which is the heart of the wireless baseband processing. In our previous work [8] [9], several DSP algorithms used in wireless communication have been mapped onto the Montium architecture. The implementation results show that the Montium architecture is flexible enough to adapt to different algorithms with good energy-efficiency. For

Cognitive Radio devices working in the emergency network, energy-efficiency is a crucial issue because the battery life is a limitation for successful operations. Therefore the reconfigurable platform we propose not only targets flexibility but also energy-efficiency. In the next section, we will introduce the key element of this platform, the Montium tile processor.

B. The Montium Tile Processor

The Montium is an example of DSRH which targets the 16-bit digital signal processing (DSP) algorithm domain. At first glance the Montium architecture bears a resemblance to a Very Long Instruction Word (VLIW) processor. However, the control structure of the Montium is very different. For (energy-) efficiency it is imperative to minimize the control overhead. This can be accomplished by statically scheduling instructions and using instruction decoders. The lower part of Figure 5 shows the Communication and Configuration Unit (CCU) and the upper part shows the reconfigurable Tile Processor (TP). The CCU implements the interface for off-tile communication. The TP is the computing part that can be configured to implement a particular algorithm. Besides it's

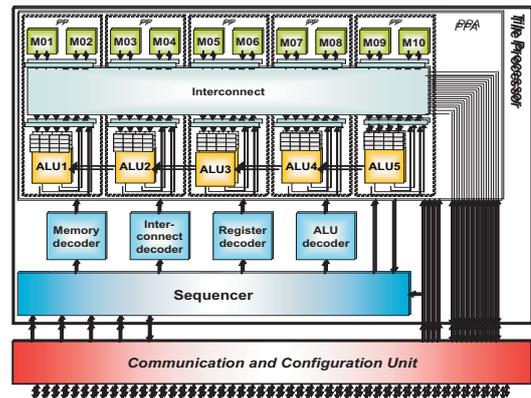


Fig. 5. An example of DSRH: a Montium processor

flexibility, the Montium is very energy-efficient. By statically scheduling instructions at compile time, the overhead of both communication and control is reduced. A good example, presented in [7], is that a FIR filtering algorithm implemented on the Montium does not change the instructions in 99% of the time. Therefore, the instruction decoding does not result in excessive switching of control signals. To give a better idea how energy-efficient the Montium is, we take the implementation results of the FFT as an example. This algorithm is one of the most computationally intensive parts in OFDM baseband processing and spectrum sensing. According to [7], the execution part of an FFT butterfly takes 21 clock cycles on an ARM920T running at 250MHz while it takes only 1 clock cycle on the Montium running only at 100MHz. Put in a simple way, a Montium is about ten times faster than an ARM. However, the Montium is 15 times more energy efficient than the ARM [7].

C. The Design Methodology

To implement Cognitive Radio on a heterogenous reconfigurable platform, we propose a design methodology which

has two new features: 1) Transaction level modelling of an application into a parallel task graph, 2) Run-time spatial mapping of tasks onto heterogenous processing tiles.

First, a high level application such as the baseband processing part of a Cognitive Radio, is partitioned into tasks. The applications are typically streaming applications. The tasks within the applications communicate with each other via channels which are accessed by tasks through interfaces. Communications between tasks are data transactions including synchronization and data transfer. In a transaction level model, the details of communication among computation components are separated from the details of the computation components. For example an OFDM receiver can be modelled by the computation components (or tasks) in Figure 3 while communications are modelled at transaction level. In our implementation model, we use the Task Transaction Level (TTL) approach [10]. Here we give a pseudo code example of the FFT task implementation.

```

process Task_FFT {
  initialization;
  while (true)
  { \ declaration of local variables
    complex Y[64], Z[64];
    for(i=0; i<64; i++)
      \ read input samples
      InRead(&(Task_FFT->inport), &Y[i]);
      \ perform the FFT
      call function Z=FFT(Y);
      for(i=0; i<64; i++)
        \ write output samples
        OutWrite(&(Task_FFT->outport), &Z[i]);
  }
}

```

While the condition is true, samples are read into the local buffer from the channel which is connected to the input port. When samples are ready, data processing is executed by calling the FFT function. Results are written to the channel connected to the output port. Both synchronization and data transfer are done by simple read and write function calls. For each task of an application, one or more task implementations have to be provided. A task implementation is the implementation of a task on a particular tile, e.g. object code for an ARM or a DSP or configuration data for an FPGA or a Montium. Computation components can be plugged-in and replaced as functions, which allows application adaptivity within the same TTL framework. So, it is advantageous to use the transaction level model to speed up simulation and allow exploring and validating design alternatives at a high level of abstraction. A task implementation has several characteristics, e.g. the amount of energy it takes to execute the task on a particular tile of the architecture, the delay or tile utilization. This implementation library with characteristics is composed at design-time.

At run-time, a spatial mapping function module selects the tile implementation for each task of an application. The objective of mapping is to find a mapping solution which optimizes a certain design metric such as the energy consumption given other system constraints (e.g. timing, tile availability). Such a mapping problem is known to be NP-hard and in practice, the corresponding optimization algorithms may run for hours.

Therefore, heuristics have to be used to find a solution with a reasonable quality within an acceptable time. In [11], a minimal weight algorithm is proposed to map a digital audio broadcast receiver onto a tiled SoC in a reasonable time, where the optimization metric is to minimize the energy consumption. Run-time mapping offers a number of advantages over design-time mapping, especially for a Cognitive Radio which dynamically changes its behavior.

- First, run-time mapping can adapt to the available hardware resources. The available resources may vary over time due to dynamic applications.
- Second, run-time mapping can adapt to dynamically changing applications. A task within the application may change its computational behavior at run-time, for example, the size of the FFT or a new task can be added to the application.

VI. CONCLUSIONS

In this paper, we introduce a heterogenous reconfigurable SoC platform to support Cognitive Radio in the context of emergency networks. The heterogeneous reconfigurable architecture includes heterogenous processing elements such as general purpose processors (GPP) and ASICs. A key element in this heterogenous reconfigurable architecture is the Domain Specific Reconfigurable Hardware (DSRH), which can achieve the reconfigurability in combination with the energy efficiency. A Domain Specific Reconfigurable Hardware (DSRH) core, called Montium [7], developed in our group is introduced. A design methodology is needed to map applications onto a heterogeneous platform which has two new features: transaction level modelling of applications and run-time spatial mapping.

REFERENCES

- [1] J. Mitola III. Cognitive Radio: An Integrated Agent Architecture for Software Defined Radio. PhD Thesis, Royal Institute of Technology, Sweden, May, 2000.
- [2] T.A. Weiss and F.K. Jondral. Spectrum pooling: An innovative strategy for the enhancement of spectrum efficiency. *IEEE Commun. Mag.*, Mar. 2004.
- [3] <http://www.freeband.nl>.
- [4] M. A. McHenry. NSF spectrum occupancy measurements project summary. <http://www.sharespectrum.com>.
- [5] Przemyslaw Pawelczak et al. Cognitive radio emergency networks - requirements and design. *Proc. of the IEEE Symposium on New Frontiers in Dynamic Spectrum Access*, Nov. 2005.
- [6] Danijela Cabric et al. Implementation issues in spectrum sensing for cognitive radios. *Asilomar Conference on Signals, Systems, and Computers*, 2004.
- [7] Paul M. Heysters. Coarse-Grained Reconfigurable Processors; Flexibility meets Efficiency. PhD. Thesis, September 2004.
- [8] Paul M. Heysters and Gerard J.M. Smit. Mapping of dsp algorithms on the montium architecture. *Proceedings of Reconfigurable Achitecture Workshop*, 2003.
- [9] Arnaud Rivaton, Jrme Quevremont, Qiwei Zhang, Pascal T. Wolkotte, and Gerard J.M. Smit. Implementing non power-of-two ffts on coarse-grain reconfigurable architectures. *Proceedings of the International Symposium on System-on-Chip*, 2005.
- [10] Pieter van der Wolf et al. Design and programming of embedded multiprocessors: An interface-centric approach. *Proceedings of the CODES+ISSS*, 2004.
- [11] Lodewijk Smit et al. Run-time mapping of applications to a heterogenous reconfigurable tiled system on chip architecture. *Proceedings of the International Conference on Field Programmable Technology*, Dec. 2004.