Smart Radio Challenge Proposal
Spectrum Access for First Responders

Peiman Amini, David Palchak, Xuehong Mao, Scott Talbod, Salam Akoum, Shafagh Abbasi

Electrical and Computer Engineering Department
The University of Utah

Team leader: Peiman Amini
Advisor: Dr. Behrouz Farhang-Boroujeny

September 17, 2007

Chapter 1

Problem Summary

In disaster situations, it is frequently beneficial for law enforcement, rescue agencies, and other first-responders to have the ability to communicate and exchange information quickly and reliably. Since wired networks cannot be depended upon to survive in all types of disasters and maybe impractical, wireless networks are the ideal choice. However, spectrum allocation plays a major role in the development of any wireless network. Given the relative scarcity of available spectrum, coupled with the amount of time required to approve new uses, the issue of spectrum access is the single greatest obstacle in the development of a wireless network geared towards first responders.

Cognitive radio technology has been presented as one possible solution to the spectrum access obstacle. From a user's perspective, a cognitive radio network should operate identically to a standard wireless network. However, cognitive radio nodes are designed to be "aware" of the other users of the spectrum, and avoid interfering with these users. A cognitive radio network is built to coexist in a given portion of the spectrum with the legacy devices to which the spectrum is assigned.

Construction of such a network presents many difficult challenges. The most obvious problem to be solved is how to fulfill the awareness requirements. Each node must be able to identify the presence of non-cognitive communication. Also, some form of distributed logic must be evaluated to allow the nodes to communicate reliably while avoiding the frequencies used by the legacy devices. Since the legacy device communication is typically intermittent, a protocol must be developed that details the procedure followed by the cognitive nodes for switching frequencies during transmission.

As mentioned previously, from a user's perspective, a cognitive radio network should appear to be no different than a standard fixed-frequency network. Essentially, the process of building a cognitive radio network consists of automating the acquisition and evaluation of information necessary to establish a non-interfering wireless network.

Chapter 2

Technical Proposal

2.1 Notation

We denote the Family Radio Service (FRS) users as Primary Users (PU); these are the licensees and accordingly have the priority access to the channel. The cognitive radio (CR) users are called Secondary Users (SU). The unused parts of the spectrum are called spectral holes and can be used by SUs to transmit data.

2.2 Proposed Solution

Herein, we describe a cognitive radio system that addresses the requirements of the disaster scenario. The system design is presented first, broken down into Physical and Media Acesss Control (MAC) layer sections. This is followed by an analysis of implementation considerations and assessment of the hardware required for implementation. Finally, the proposed design is tentatively fitted to a suitable commercial software-radio platform offered by Texas Instruments and Lyrtech.

2.2.1 Physical Layer

In a CR network, SUs need to dynamically and reliably determine the spectrum holes. For this, SUs are equipped with a spectrum analyzer. This spectrum analyzer must be fast and accurate enough to sense the PUs and allow the SUs to avoid causing interference. Also, since there is always a possibility for a single node to be in deep fade on a certain frequency band, reliable channel characterization requires the cooperation of multiple nodes. For our solution, we propose a scheme where every node senses the channel and the channel sensing information is collected by a base station. The base station then compiles an allocation table which is distributed throughout the network.

Channel Sensing

Most practical sensing methods are based on energy detection, i.e. if the received energy for a given carrier is greater than a defined threshold, that carrier is assumed to be busy. In order to reliably detect available spectrum holes, the channel sensing mechanism needs to feature a high spectral dynamic range. While the FFT has been suggested as one channel sensing method [3], we note that it suffers from a number of shortcomings that originate from the large side lobes of the frequency response of the filters that characterize each subcarrier [1, 2]. These sidelobes produce spectrum leakage from neighboring subcarriers, resulting in significant inaccuracy and low dynamic range. Thus, with the FFT, SUs are less spectrum agile and cannot detect low power users. This might not matter much in systems that facilitate channel access in time decision duplex (TDD) and Time Decision Multiple Access (TDMA) for accessing. Unfortunately, for the frequency decision multiple access and frequency devisor duplex mode (FDMA/FDD), the limitations of FFT have to be considered.

In our solution [1], we propose using filterbanks as the sensing method. By using a filterbank sensing system, the side lobes of the filters associated with each carrier can be made arbitrary small by adjusting filter length and design. As a result, filters are no longer the limiting factor in achieving high dynamic range. The signal power of the output of the filterbank is then used to estimate the signal spectrum. Recently, Haykin showed that the multi-tapper method (MTM) is the optimal channel sensing method [4]. Unfortunately MTM comes at the expense of computational complexity. In [1] and [2], it was shown that a filterbank of prolate filters can be used for sensing with virtually ideal performance. In the filter bank sensing method, a template filter is designed and then modulated to match each subcarrier. Implementation can be performed efficiently through a polyphase structure as presented figure 2.1 [5]. In the figure, each ρ_i is one branch of the corresponding polyphase decomposition.

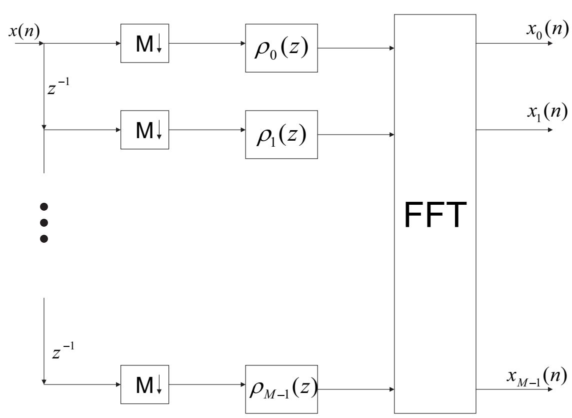

Figure 2.1: Polyphase structure for sensing.

To further simplify implementation, our system features a filter bank designed to support 256 subcarriers. The reason that this simplifies the implementation is that 256 is an integer power of 4. This allows the FFT shown in figure 2.1 to have an efficient four stage radix-4 structure.

Let us consider an example design suitable for the prescribed system. To provide good stopband attenuation, it is desirable that each branch of the polyphase decomposition, ρ_i in the figure, contains at least eight non-zero taps. Letting each branch have exactly eight taps yields a filter of length $L = 8 \times 256 = 2048$.

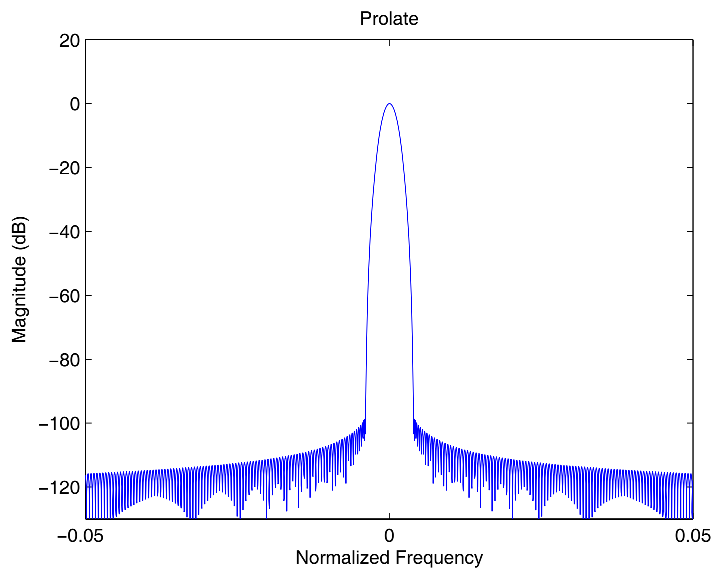

Figure 2.2: Example of magnitude response for a prolate filter.

As shown in figure 2.2, the frequency response of a prolate filter of this length achieves close to 100 dB stopband attenuation. Thus, when used for channel sensing, such a filter provides a theoretical dynamic range of close to 100 dB. We also wish to acknowledge that, in practice, the dynamic range is also limited by ADC nonlinearities, word-length limitations, and other system factors and may not actually reach such levels.

Given a filter length of 2048, we can compute the time required to perform channel sensing. The bandwidth of each subcarrier, as defined in the problem statement, is 25 kHz. This dictates that the combined filter bank bandwidth must be $25\,\text{kHz} \times 256 = 6.4\,\text{MHz}$. If the sampling is performed at the Nyquist rate, then a single channel sensing interval requires $2048 \times \frac{1}{6.4\,\text{MHz}} = 320\,\mu\text{s}$. For a more robust measurement, one with a higher confidence interval, an average of several successive filter outputs can be computed [2]. This is certainly feasible given the relatively small amount of time required for a single reading and can be performed depending on the channel environment.

Modulation and Coding

The digital modem shall provide two modes of operation for the two different types of services. One service is for Continuously Variable Slope Delta Modulation (CVSD) vocoded voice with a data rate of 16 kbps, RS channel coding, and QPSK modulation. The other service is a 19.2 kbps computer-to-computer data stream, with rate 1/2 convolutional coding and 8-PSK modulation. It has to be noted that although the two different modes use different channel coding and have different data rates, the channel coding, constellation mapping, and signal framing is such that the symbol rate at the input of the modulator is the same for both data streams.

The data stream shall be encoded using a rate 1/2, constraint length 7 convolutional encoder. The encoder generates the two outputs using 171 octal and 133 octal as the generator polynomials. The two output of the convolutional encoder are then converted to a serial data stream. Voice data shall be encoded through a Reed-Solomon (RS) (48,30, t = 9) shortened code derived from a RS (255,237, t = 9) code. This code has the ability to correct up to 9 random byte errors. The field generator polynomial is given as $p(x) = x^8 + x^4 + x^3 + x^2 + 1$. The convolutionally encoded data shall be interleaved using a row-column block interleaver. The data shall be written in the interleaver memory in rows and read out in columns. The Reed-Solomon encoded data shall not be interleaved.

The convolutionally encoded, interleaved computer data stream shall be divided into groups of three coded bits. Each group of three bits may be described as b0, b1, and b2, where b0, is the first bit output from the interleaver in each group, and b2 is the last bit output from the interleaver in each group. Each group of three bits shall be mapped to an 8-PSK constellation using a Gray code. The Reed-Solomon encoded voice data stream shall be divided into groups of two coded bits and mapped using a Gray code to a QPSK constellation.

A block diagram of the proposed transmitter is depicted in Figure 2.3. The pulse shaping filter shall be a root raised cosine filter with a roll-off facter to be determined during the implementation stage.

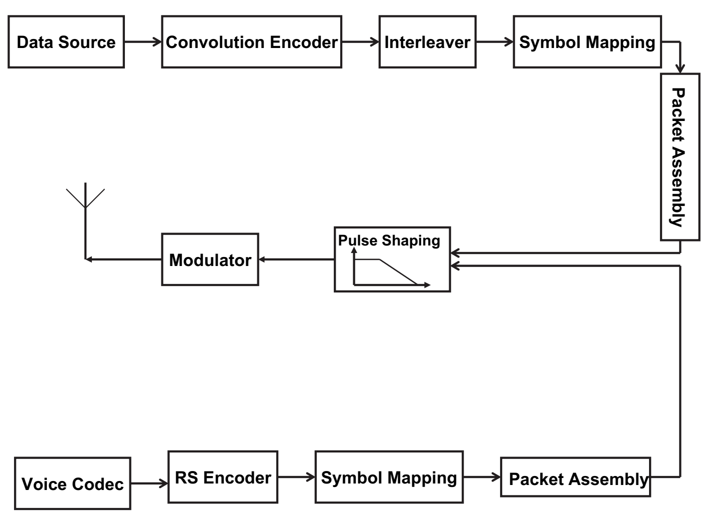

Figure 2.3: Transmitter block diagram.

Synchronization

The system shall feature a packet-based communications model where all syncronization information is included in a preamble to be appended to each packet. Figure 2.4 depicts the general structure of the common preamble. The carrier field of each packet shall consist of a sequence generated by repeating a single symbol. The actual symbol used shall be determined during the implementation stage. The timing field shall consist of an alternating sequence of BPSK symbols, e.g. $1, -1, 1, -1, \ldots$. Finally, the sync field consists of a unique sequence of QPSK symbols used to denote the end of the preamble.

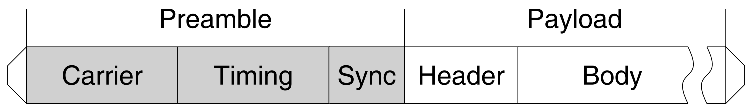

Figure 2.4: Common packet structure.

The carrier field of the preamble is used for automatic gain control (AGC) and carrier recovery. After pulse-shaping, the carrier field contains a DC signal, which when modulated and then demodulated becomes a complex sinusoid. The frequency and phase of the resulting sinusoid corresponds directly to the carrier mismatch between the transmitter and receiver. Estimation of the carrier frequency and phase shall be performed using a second order Extended Kalman Filter (EKF) [7].

The timing field of the preamble is used to compute an initial estimate for the ideal sample timing offset using the well-known power spectral line method [8]. Once such an estimate is obtained, retiming of the signal shall be accomplished by applying a fraction-delay FIR filter (FD-FIR) [6].

2.2.2 MAC Layer

In this part of our proposal, we propose the cognitive MAC layer for the disaster scenario. The proposed MAC layer has the ability to manage the medium access so as to co-exist with primary users. Coexistence is the *primary goal* in this design as well as in other CR systems, such as IEEE 802.22 [9]. The medium access method, as defined in the problem statement, shall be frequency-division multiple access (FDMA). SUs shall use frequency-division duplexing (FDD) to communication to each other.

Network Architecture

In our design, we envision a network of no more than 15 leaf nodes and a base station. The reason for this limitation is *not* imposed by our MAC layer. In fact as we show below, thanks to the ability to assign new signaling channels as the network grows, technically our solution can scale to an arbitrary network size. However, the maximum number of SUs is determined by the number of PUs as well as their traffic patterns – both of these factors are unknown to us at this point. Hence, only to make our proposed system as robust as possible, we restrict ourselves to this rather small number of leaf nodes.

It has been shown in IEEE 802.22 contributions that having a base station and a separate control channel is a practical topology for CR systems. As a consequence, our proposed system shall be based on this topology, also.

The control channels shall be used for coordinating sensing information, controlling leaf node communications and other management tasks. Our setup shall feature uplink and downlink control channels. The former shall be based on Aloha and the latter shall be a base station broadcast channel for broadcasting sensing information and other management packets. Aloha is required since channel sensing in carrier sense multiple access collision avoidance (CSMA/CA) is not always possible depending on the channel environment and propagation delay. To facilitate the discovery of the control channel for new leaf nodes, the base station shall send out periodic beacons, as in IEEE 802.11b networks. These beacons, as well as all other control channel traffic, shall feature a unique packet header denoting them as control messages. Once control channels have been established, access shall be granted on a first-come first-serve basis. Depending on CR size and traffic, the base station shall be able to increase the number of Aloha-uplink signaling channels and assign them to a subset of the leaf-node population. This may become necessary to guarantee reliable delivery of time sensitive sensing information. To summarize, in our system there is one downlink control channel and one or more uplink control channels. Similar to voice payloads, QPSK modulation and RS coding shall be used for the control channel.

Channel Sensing

The base station shall receive and compile the channel state data from all leaf nodes and itself. After a suitable delay, to be determined later, the base station shall broadcast the compiled channel allocation table to all leaf nodes. This allocation table shall be stored by the base station and leaf nodes until the conclusion of the next sensing interval. Each leaf node shall stop transmitting and sense the entire channel periodically. If the current sensing information differs from the last compiled channel state information, following a random delay, the channel state data shall be transmitted by each leaf node to the base station. To reduce congestion on the control channel, a single bit shall be used to indicate the state of each subcarrier. With 200 subcarriers in the channel, this results in a message body of 25 bytes in length. If a leaf node experiences three successive failures when attempting to transmit new channel state information, the node shall quit trying for that sensing period. If this occurs multiple successive periods, the leaf node shall ask for an additional signaling channel.

Scalability

We now elaborate for the requirement for dynamically allocating more Aloha control channels. In order to have an operational system that can coexist with PUs, the base station must compile an accurate allocation table. This can only be performed if all the SU sensing information is delivered to base station reliably and with low delay. At this stage, we want to give an approximate calculation of how many control channels we need as a function of network size and outline our methodology.

Compared to sensing information, the traffic load generated by channel requests, etc. is negligible. Our preliminary calculation shows that our physical layer is able to send around 20k symbols per second. With a one bit status identifier for each of the 200 subcarriers in the CR band, before coding, we obtain a total of 200 bits of sensing information. Assuming QPSK modulation, the packet length is approximately 150

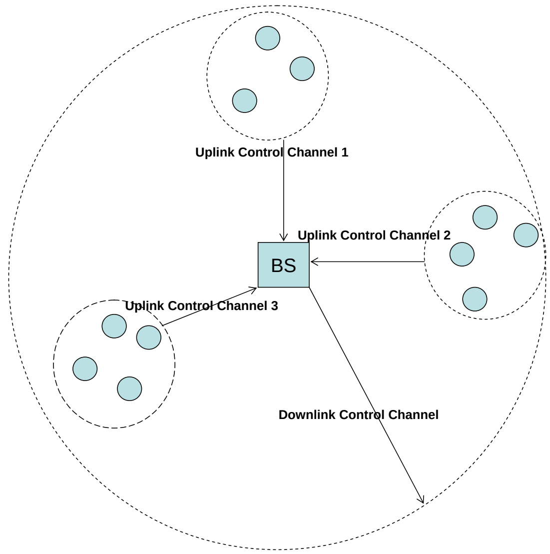

Figure 2.5: Example of logical control channel assignment.

symbols. In the worst case scenario, where the channel changes in every sensing period, leaf nodes transmit new sensing information every sensing period. With a sensing period of 100 ms, the arrival rate for each node is $\lambda = 10\,\frac{\text{packet}}{\text{second}} * 150\,\frac{\text{symbols}}{\text{packet}} * \frac{1}{20000\ \text{symbol}} = 0.075$. Hence, using the results from [10], one slotted Aloha channel can support up to 4 users with an average delay of 60 ms which is acceptably smaller than our sensing period duration. Therefore, as the number of leaf nodes increases beyond 4, more than one uplink control channel must be used. A graphical explanation is shown in figure 2.5. However, in section 2.2.3, our initial implementation features only two leaf nodes and therefore, one uplink control channel is sufficient for reliable testbed operation.

Channel Access, Unicast and Multicast Traffic

When a leaf node has data to transmit, it shall request a channel assignment from the base station. This request shall include the requested service type (voice or data), the transmission destination, and the expected transmission duration. Given such a request, the base station shall reserve two free channels, one for uplink and one for downlink, and assign them to the requesting and destination node. Two channels are required to allow the nodes to communicate in a FDD manner.

Once the channels have been allocated, communication between leaf nodes does not involve the base station. In the case of voice traffic, which requires real-time performance, a Quality of Service (QoS) assessment of the channels can be done by the leaf nodes before the actual communication starts. If the QoS test fails, then the leaf nodes shall request a different set of channels from the base station.

Especially in disaster scenarios, not only team leaders but also all the team members have to be able

Table 2.1: QoS Parameters.

	Real-time	Data
Delay	< 150ms	NA
Jitter	< 30ms	NA
Loss	$< 3\%$	$< 10^{-4}$

to reach all others in the CR network reliably and efficiently. Hence, in addition to leaf node-to-leaf node communication, our system shall include a special high-priority one-to-many multicast mode with push-to-talk (PPT)-like functionality. In the first two phases of the project, multicast operation shall only include voice traffic. During phase 3, PTT operation shall be extended to include data multicast, also. According to original PTT functionality, all leaf nodes in the CR network *share* one uplink and one downlink carrier. The assignment of the PTT subcarrier shall be performed by the base station out of the set of the free subcarriers and signaled to the leaf nodes over the downlink signaling channel. As long as a PTT downlink subcarrier is indicated, leaf nodes shall continuously sense that subcarrier. As soon as the leaf nodes detect payload activity on it, they shall cease transmitting and receiving on their assigned subcarriers and tune to the PTT carrier. In contrast to the classical PTT protocol, the leaf nodes shall only grant access to the PTT uplink carrier if for a number of consecutive packet intervals, to be defined, the downlink carrier was idle to avoid uplink blockage. To guarantee fair access to the service, a leaf node is only able to claim PTT access for a period of packets, to be defined. After the period, a new Application layer request has to be issued, i.e. "press the button", during which other nodes can claim the carrier. Also, the leaf nodes shall be able to detect periods of silence in the PTT traffic to release the PTT downlink subcarrier to avoid downlink blockage. A special packet, format to be defined, shall indicate the end of a PTT transmission after which the leaf nodes return to normal operation. In addition, our system shall support a "team leader" mode, where only one leaf node has PTT access rights, supervised by the base station.

Quality of Service

Three metrics shall be used to define quality of service for the system: delay, delay jitter and packet loss. Delay shall be defined as the time required for a voice packet to arrive at the target destination. In general, delay consists of a fixed codec delay and a variable transmission delay. Delay jitter is the variation in transmission delay. Packet loss refers to the ratio of unrecoverable packets to the total number transmitted packets. The first two metrics apply to voice communication while the last metric is used for both voice and data communication.

For our solution, two QoS classes are defined: real-time and data. The real-time class shall be used for voice communications, while the data class shall be used strictly for data communication. Table 2.1 shows the parameters for the two classes.

We note that for real time traffic, bit error rate (BER) is not the only reason for packet loss. In our application, when voice is played at the receiver, only in-order and on-time packets are presented to the application. All other packets are discarded. As such, delay causes packet loss in real time systems.

As mentioned previously, prior to commencing voice communications leaf nodes shall test the channel to ensure that the minimum QoS requirements are met. This shall be accomplished by transmitting a sequence of short packets from one node and measuring the QoS parameters at the other node [12]. The number and type of packets transmitted depend upon the desired precision of the measurements.

Since the wireless environment is time-variant, the channels used by the leaf nodes might not be reliable for the entire duration of the transmission. Therefore, both nodes shall continuously estimate the QoS parameters to decide whether the current channel suffices. This shall be performed through payload packet monitoring. If it is agreed between nodes that the channel may not fulfill the QoS requirements, a new set of channels shall be requested from the base station. Once the base station assigns new channels, both nodes shall cease communication on the old channels. When switching channels due to the appearance of a primary user, nodes shall maintain the established QoS parameters.

Base Station Summary

The base station shall be a centralized node capable of performing the following duties:

- Finding control channels: The base station shall sense the channel and attempt to locate empty subcarriers to use as the control channels. If this is successful the base station shall transmit a periodic beacon on each control channel. If there are no channels available, the base station periodically senses the channel to find available channels. The base station shall assign one or more uplink control channels depending on the number of nodes and the channel quality. Leaf nodes may ask for more control channels. The base station specifies which uplink control channel is available for new nodes in the periodic beacons. If there is more than one uplink control channel, the base station shall announce the control channel assignment on the downlink control channel periodically.
- Channel sensing: The base station shall perform channel sensing periodically.
- Announcing the available nodes: The base station shall announce the list of available nodes on the downlink channel periodically.
- Changing or adding control channels: If the base station receives message(s) from leaf nodes indicating low quality on the control channel(s), it shall change the control channel or assign more control channels depending on the channel status and the environment. Leaf nodes may also specifically request more control channels.
- The base station shall be able to denote a subcarrier as PTT carrier upon request and signal its decision to the leaf nodes over the downlink control channel.
- Compiling channel state information: The base station shall compile sensing information from itself and all leaf nodes into a subcarrier allocation table. Each leaf node senses the PU transmission as well

as other leaf node activities.

- Channel assignment: The base station shall mediate subcarrier access by the leaf nodes. When a leaf node requests access to the medium, the base station shall select one or more available subcarriers and assign them to that leaf node. After a certain duration, the base station shall reacquire control of the assigned subcarriers.
- Traffic forwarding: The base station shall provide a method for forwarding over-the-air (OTA) traffic from leaf nodes to wired nodes to which the base station is connected.
- Discovery: The base stain shall sense the channel to find an uplink and downlink control channels; then, it starts sensing periodic beacons on those channels.

Leaf Node Summary

Each leaf node shall be capable of performing the following duties:

- Channel sensing: Each leaf node shall perform channel sensing periodically. If the channel status is different from the last compiled channel state information received from base station, leaf node shall be able to transmit the new sensing data to the base station via one of the control channel(s).
- Locating Control Channels: Leaf nodes shall be able to locate control channels, using the unique header structure of periodic beacons transmitted by base station to be defined later. If there is more than one control channel, the leaf node must use the control channel that the base station assigns to this specific leaf node.
- Initiating a communication: If a leaf node has new information to send to other leaf nodes or a server outside the CR network, it shall be able to initiate a connection by requesting a channel from base station. Nodes shall be able to accept an incoming traffic.
- Measuring QoS metrics: Leaf nodes shall measure the delay, delay jitter, and packet loss for voice connections and packet loss for data connections. Leaf node shall request a new channel if the QoS measures are not satisfied.
- Leaf nodes shall support PTT voice multicast functionality.
- Discovery phase: A new leaf node shall be able to find the downlink control channel; then, it shall find the available downlink control channel that the base station announces for new nodes. The leaf node then must ask the base station for the control channel assignment.

Control Message Summary

The control messages recognized by both the base station and leaf nodes, at a minimum, shall include:

- Channel state update: This message shall be transmitted by the base station to all leaf nodes whenever the frequency allocation table changes.
- Channel sensing report: This message shall be transmitted by leaf nodes following a channel sensing interval.
- New channel request: This message shall be transmitted by a leaf node to request a channel from the base station. The message must specify whether a voice or data channel is being requested.
- Channel assignment: This messages shall be transmitted by the base station as the response to a channel request. It specifies the subcarrier(s) that a node is allowed to use to communicate with another node. It also specifies the time duration for which the access is granted.
- Channel change request: This message shall be transmitted by a leaf node when it realizes that the current channel can not satisfy the minimum QoS requirements. The message must specify the currently assigned channel.
- New control channel request: If delay or packet loss passes a threshold to be defined in the implementation phase, nodes shall ask the base station for a new control channel.
- Periodic beacons: The base station shall transmit periodic beacons over all control channels. In the periodic beacon, the base station shall specify the available uplink channel(s).

2.2.3 Implementation

Design Analysis

Our proposed physical layer design offers several notable advantages. First and foremost, the selection of a constant power envelope modulation scheme eases the requirements for the IF and RF amplifiers. Also, as the required symbol rate for each type of service are identical, both transmission modes have roughly equivalent effective signal bandwidths. This simplifies the selection of a suitable sampling rate for the ADC and DAC components and defines the minimum speed of the baseband processing components. Finally, the selection of popular, well-defined coding schemes allows for the use of previously developed decoders that benefit from ongoing refinement and optimization.

The proposed MAC protocol attempts to balance certain trade-offs that affect system performance. The centralized coordination of the SU channel state information (CSI) can reduce the probability of SU interfering with a Primary User (PU). For example, consider a PU that is close to one SU and far from another. The distant SU may not detect the presence of the PU. However, since the nearby SU detects the PU, both SUs avoid using the PU's channel. If the PU is moving, this ensures that it does not unknowingly move into an area of SU interference. The principle downside of coordinated CSI is it limits the scalability of the system. However, as shown in section 2.2.2, our method of allowing multiple control channels which are assigned to a subset of leaf nodes guarantees reliable and scalable network operation.

Another potential shortcoming of the proposed architecture is the potential for isolation of nodes. This situation can arise when not enough base stations are present to fully cover a given area. Alternatively, total network fragmentation can occur when the base station experiences extended deep fades as the result of unexpected environmental changes. Such large-scale environmental changes are not unlikely to occur in disaster scenarios.

Simulation

The first phase of implementation entails the development of a complete system simulation using Mathworks' MATLAB and Simulink products. This simulation is to include all aspects of the system behavior that are feasible to simulate. Feasible aspects include channel distortion, noise, frequency and timing offsets, oscillator drift, ADC/DAC overflow and underflow, and urban path loss. Note that these parameters shall be considered to be time-varying at reasonably expected rates.

All system components, i.e., filters, oscillators, state-machines, etc, shall be implemented in the simulation with the same set of execution constraints as imposed by the hardware. For example, the signal processing computations have to be performed in hardware using fixed-point arithmetic. Thus, in the simulation, the signal processing components shall use identical fixed-point arithmetic, including identical rounding and overflow behavior. Also, the buffer sizes in the simulation shall not exceed those available in the hardware.

Hardware Requirements

The hardware required to implement our proposed design can be broken down into two categories which roughly correspond to the physical and MAC layer processing. The former consists of stream-based computations such as filtering, synchronization and symbol decoding. These tasks are well suited for implementation in either an FPGA or a DSP. FPGAs generally offer much greater parallelism than DSPs, so this can help determine the correct hardware for a given component. Conversely, the MAC layer processing is mostly state-machine based control and resource management. This type of processing typically benefits from a greater awareness of the overall system state and is best performed by a microprocessor running some type of real-time operating system.

Based on the load types, we envision that the minimum hardware required to implement the proposed design includes at least one large FPGA for signal processing and interfacing and one 32-bit microprocessor for system control. The platform also needs to include high-speed ADC and DAC components for direct conversion of Intermediate Frequency (IF) signals. Additional platform features that are necessary for demonstrating system performance are RF front-ends, a second DSP or FPGA for additional signal processing power, and one or more external high speed interfaces for transferring data to and from the board.

System Layout

The Small Form Factor Software Defined Radio Development Platform, provided by Texas Instruments and Lyrtech, closely matches the ideal hardware platform for implementing the proposed design. To take full advantage of the resources offered by the hardware, we suggest an implementation that distributes the system components between the FPGA, DSP and ARM9 cores.

The flexibility of the FPGA as a programmable logic device makes it the primary candidate for providing the interface between the analog conversion hardware. Since the ADCs and DACs convert bandpass IF signals, the FPGA shall be used to perform complex digital up- and down-conversion as well as sample rate conversion. The potential parallelism that can be acheived with the FPGA makes it well suited for implementing the large filter bank employed for channel sensing. Finally, the relatively large distributed memory capacity of the FPGA enables it to be used a circular buffer or anther storage structure if necessary.

The DSP, as implied by its name, shall be used to perform an array of baseband processing steps that also require some algorithmic state. Examples include the routines for carrier recovery, symbol timing recovery, and packet frame synchronization. The DSP shall also execute convolutional and Reed-Solomon encoding and decoding as well as translation between symbols and bits. Lastly, depending on the exact interface design, the DSP may host the voice codec used by the system.

The overall system control and MAC layer processing shall be handled by the ARM9 microprocessor core. The nature of the system requires the use of a real-time operating system running on the ARM9. Tasks include MAC state machine, packet framing, run-time evaluation of QoS using hardware timers and user interfacing. As an option, the ARM9 could also host the voice codec instead of the DSP.

In terms of simulation, the system components tentatively assigned to the FPGA and DSP shall be implemented as Simulink blocks. This is advantageous as it allows us to use the Lyrtech software to assist in generating suitable HDL and DSP code from the simulation code. The ARM9 code shall be simulated as MATLAB functions, or possibly as MEX functions written in C. Writing the simulation code as MEX functions make it much easier to port to the ARM9 architecture, though it will also make the code more difficult to debug.

Interfacing

The Small Form Factor Software Defined Radio Development Platform includes both USB and Ethernet connections for interfacing. One or both of these connections shall be used to to exchange control messages and data between a host workstation and the hardware. As stated previously, the nature of the system requires the embedded microprocessor to run some kind of RTOS. Based on funding availability, we propose using FreeRTOS for this project. FreeRTOS offers a number of benefits for this application, including full access to the source code and a completely royalty-free license. Furthermore, μIP, an Open Source embedded TCP/IP stack, has previously been integrated with FreeRTOS on an ARM9 core and is also freely available. The combination of FreeRTOS with μIP allow us to implement interfaces based on UDP/IP or TCP/IP, thus simplifying protocol development.

Debugging and development on the TI/Lyrtech platform shall be performed through the integrated IEEE 1149.1 JTAG interface. While a number of commercial compilers and development environments are available for the ARM9 architecture, some of these packages require payment of upwards of $10,000 in licensing fees. Consequently, we propose using the GNU-ARM development tool chain, a freely available variant of the mature GNU Compiler Collection for ARM cores. The GNU-ARM compiler and integrated debugger can also be used with the Eclipse C/C++ IDE for a complete solution. Development and compilation of the source for the DSP and FPGA cores shall be accomplished using Texas Instruments Code Composer Studio and Xilinx's ISE Foundation software packages, respectively.

2.3 Deliverables

Completion of the project will include delivery of the following items prior to the established deadline.

2.3.1 System

1. An over-the-air demonstration of a functional system operating with at least two leaf nodes, one base station, and one Primary User.

2. A simulated demonstration of a functional system with our maximum system size of 15 leaf nodes, one base station, and 15 Primary Users. Both demonstrations will show the following minimum capabilities:

   - Channel sensing
   - MAC handshaking
   - Leaf node communication via both data and voice channels at the defined QoS levels
   - SU channel switching following appearance of Primary User
   - Maintenance of both QoS levels during channel switching

Software

1. A detailed functional simulation of the entire system, including RF transmission effects such as fading and Doppler shift
2. HDL source code for all FPGA components, unless prohibited by third-party licenses
3. DSP source code for all components, unless prohibited by third-party licenses
4. RTOS configuration files and build environment parameters
5. Microprocessor application source code, including MAC algorithms
6. All unit tests and test data sets
7. All integration test routines and data sets
8. Complete data sets used in system demonstrations

Documentation

1. Monthly status reports, at least one page in length, to include the following details:
   - Progress to date
   - Development issues that have arisen, including design changes
   - Proposed resolution to issues, including justification for design changes
2. One engineering notebook for each participating team member containing dated work entries with the following information:
   - Work performed that day
   - An explanation of design/implementation details
   - Identification of trade-offs and justification of final decisions
3. A written tutorial covering the architecture and internal flow of the system simulation. Document shall also specify the user configurable parameters as well as the limits of the simulation accuracy in all applicable areas.
4. A set of written instructions for configuring and building the RTOS application code on a standard workstation featuring the required toolchain.
5. A set of detailed block diagrams showing the interconnection between each component of the system. The detail level shall be at the level of individual filters, NCOs, memory buffers, decision devices, etc.
6. A paper, suitable for conference publication, detailing the MAC protocol developed for the project. This paper will be submitted for publication no later than July 2007.
7. A document explaining the parameters used for the over-the-air and simulated demonstrations. This document shall also specify the mechanism used for exchanging data with the hardware, if applicable.
8. A report breaking down the final design and implementation to the component level. For each component the report will include:
   - A description of the final design. This shall also include a comparison to the original design, if applicable, along with justification for the changes.
   - A description of the implementation, including hardware dependent optimizations.
   - An evaluation of the optimality of the component in perspective to its relative significance or complexity.

2.4 Materials

This section presents the materials that will be used to implement the physical layer of the cognitive radio system. Implantation and testing of the proposed system are facilitated through the use of two software radio platforms: the suggested Texas Instruments (TI) platform and the Flux Radio platform [11].

Thanks to co-location with the Flux networking group, we are in the unique position to use their Fixed Wireless testbed to model the interference from the primary users during implementation and testing stages. The Fixed Wireless testbed consists of custom RF front ends that provide a reconfigurable IF-to-RF solution and a DAC/ADC conversion. The front-ends are connected to PC workstations that run the GNU Radio signal processing package. GNU Radio is an Open Source software package that provides an efficient platform for performing stream-based signal processing in quasi-real time. Figure 2.6 shows a diagram of the scenario being modeled.

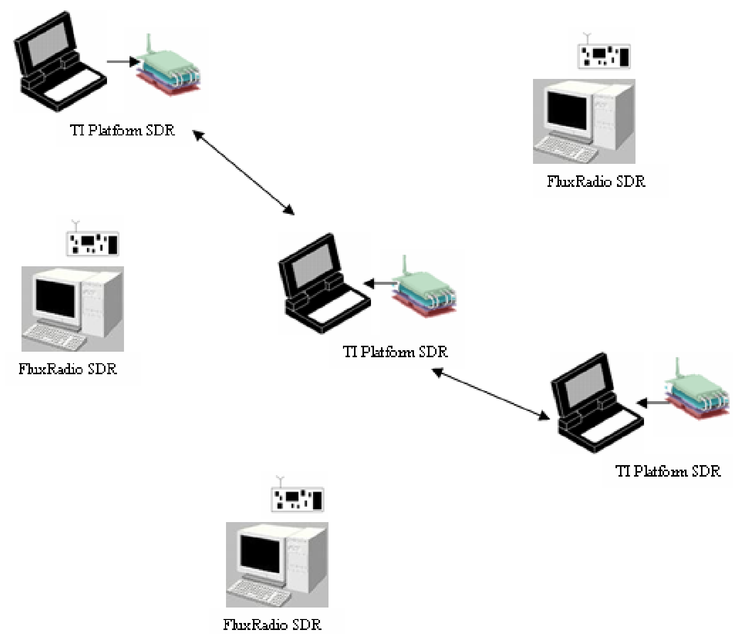

Figure 2.6: Graphical depiction of system test environment.

The components that need to be acquired for the testing and assembly processes include:

1. **Computer Workstations**: At least four computer workstations are needed for simulation, development and performance testing
2. **TI development platforms**: Three TI/Lyrtech SDR platforms are needed to form the demonstration system as specified in the Deliverables section
3. **Xilinx ISE Software**: Needed for programming the FPGA .

4. **Code Composer Studio**: Needed for DSP system configuration.

5. **MATLAB 7.5**: Used for modeling and testing the transmitted and received signals.

6. **GnuRadio software**: Used for performing the necessary signal processing on the FluxRadio platforms.

The total estimated cost of the project is summarized in table 2.2 below.

Table 2.2: The estimated cost of the project.

Category	**components**	**$ US cost**	**Description**
Labor		9/12*24,000*5 = 90,000	5 full time RA positions (20 hours/week)
Material	Desktop Computers	N/A	Available for Laboratory use
	Laptop Computers	N/A	Available for Laboratory use
	TI development Platforms	N/A	Provided by the SDR forum
	FluxRadio platforms	N/A	Provided by the Flux Radio group (Emulab)
	Matlab	N/A	Provided by the SDR forum
	Xilinx ISE tools	N/A	Provided by Xilinx
	Code Composer Studio	N/A	Provided by the SDR forum
Overhead	Miscellaneous	5000	documentation, unforseen costs
	Travel	5000	present project results at SDR forum 2007
Total		**100,000**	

2.5 Risk Assessment

We have identified several items as the primary risks in our proposed design and implementation plan. These items are listed below along with our plans to mitigate their impact on the success of the project:

- Personell changes: Changing team members are likely to be the main concern in our project. David Palchak, a team member with hardware experience, will graduate and leave the team prior to the completion of the project. However, he will be replaced in a timely manner by Ehsan Azarnasab who will join the team in January 2007.

  To compensate for the potential time lost due to personnel changes, incoming team members will be informed of all ongoing development prior to their arrival. Also, a clearly defined transition plan, to be drawn up closer to the arrival date for the new member, will be created and included as part of the status report deliverable for that month.

- Development tools: The current funding devoted to this project is not sufficient to being able to purchase professional development tools for the ARM processor. As a result, we decided to use freely available development tools available which are not as efficient and easy to use. Depending on the exact hardware and software configuration required, we anticipate requiring additional man hours to bring the ARM9 processor to a usable state.

We are currently in the process of assembling a set of solicition documents that we intend to submit to several prominent commercial ARM9 tool providers. The objective of this campaign is to acquire a donation of a commercial RTOS and supporting tools. This will potentially reduce much of the risk associated with the ARM9 development. If the solicitation process proves unsucessful, we will evaluate the current design status and reallocate development time accordingly.

- System complexity: Since the hardware platform to which we have matched the proposed design is new, complexity problems may arise in implementing the proposed design. Such problems typically arise when a certain combination of conditions is met, and can be difficult to debug without prior experience and knowledge of the platform quirks and particulars. It is possible that such problems could require undue time to rectify and stall development.

  As this type of risk is inherent in all such projects, no direct course of mitigating action can be planned. We have constructed the included work plan using modest time estimates for the various stages of development and implementation. It will be our goal to exceed these estimates during all phases of the project, so has to make available extra time if it becomes needed.

Chapter 3

Workplan

During the 10 months of the second phase of the Smart Radio Challenge, the system defined in our proposal will be developed through software simulation and hardware implementation. The expected goal is to setup a trial system with one base station, one primary user and two secondary users. Two traffic streams, voice and data, will be tested between the users. MAC layer processing, such as channel switching and QoS maintaining, will be implemented in the trial system as well. The following tasks will be fulfilled:

- Computers simulation of the designed system
- Implementation of physical layer on hardware
- Implementation of MAC layer processing
- Documentation

3.1 Time Schedule and Milestones

Besides the tasks and milestones listed above, monthly status reports will be submitted to keep trace of the development of our proposal.

Table 3.1: Workplan.

Stage 1 -	**Software simulation** **Hardware preparation**	
Dates	**Tasks**	**Participants**
12/02/2006 - 02/28/2007	Simulate Physical Layer - Channel Estimation - Modulation - Channel Coding/Decoding - Timing/Frequency Recovery Simulate MAC Layer - Channel Sensing - MAC Handshaking - Channel Switching - QoS Maintenance Documentation for software simulation	Peiman Amini Xuehong Mao Scott Talbot
	Hardware preparation - RTOS Installation - System familiarization	Ehsan Azarnasab Salam Akoum Shafagh Abbasi
Stage 2 -	**Hardware implementation** **Research paper**	
03/01/2007 - 05/31/2007	DSP Implementation - Timing/Frequency Recovery - Modulation/Demodulation - Channel Coding/Decoding - Channel Estimation FPGA Implementation - Channel sensing filter bank - ADC/DAC interfacing - Packet buffering	Salam Akoum Shafagh Abbasi Ehsan Azarnasab
	Publishable research paper	Peiman Amini Xuehong Mao
Stage 3 -	**Hardware implementation** **Final Report**	
06/01/2007 - 07/31/2007	ARM9 Implementation - MAC Layer protocol - QoS measurements	Salam Akoum Shafagh Abbasi Ehsan Azarnasab
	Final report	Peiman Amini Xuehong Mao
Stage 4 -	**System test and debug** **Final Report**	
08/01/2007 - 09/30/2007	Final report System test and debug	Salam Akoum Shafagh Abbasi Ehsan Azarnasab Peiman Amini Xuehong Mao

Chapter 4

Facilities

In this section, we provide a description of the environment where the project is to be implemented.

The laboratory facility is located on the second floor of the Merill Engineering Building (MEB) on the University of Utah Campus. The laboratory space hosts the Wireless Communications and Digital Signal Processing Groups and its content is limited to the equipment necessary for these two groups to conduct their research.

The MEB building is located in the upper part of the campus at a considerable distance from any toxic materials or hazardous waste producing facilities; hence, no environmental impact statement is required for the laboratory space proposed for hosting the implementation of the project. Furthermore, the University of Utah administration ensures compliance with federal, state and local government regulations pertaining to airborne emissions, waterborne effluents, external radiation levels, solid bulk waste disposal practices, and handling of toxic and hazardous materials. The testing equipment available in the laboratory include:

1. Two BK precision 4040A 20MHz sweep/function generators.
2. Two BK precision triple output DC power supplies Model 1760.
3. A Tektronix TDS 224 4-channel digital real-time oscilloscope.
4. An Agilent 2 G samples/sec oscilloscope.
5. An Agilent E4407B spectrum analyzer.
6. A number of RF mini-circuits used for amplification, attenuation, termination and coupling of the signals.
7. A number of antennas for various frequencies.
8. Basic electronics kits (soldering irons, connecting wires, ...) used for assembling the components.

The level of outdoor noise in the laboratory shall be measured with the spectrum analyzer; if judged unacceptable, the electromagnetics group at the University of Utah possesses a 10x10 m echoic chamber for total elimination of environmental noise.

Chapter 5

Intellectual Property

None of the information presented in this proposal constitutes protected intellectual property of the University of Utah or any 3rd party affiliates. Similarly, no part of the proposed design or implementation uses knowledge that is considered to be protected intellectual property.

Bibliography

[1] P. Amini, R. Kempter, R. R. Chen, L. Lin and B. Farhang-Boroujeny, "Filter Bank Multitone: A Physical Layer Candidate for Cognitive Radios", *Software Defined Radio Technical Conference, SDR 2005*, November 14-18, 2005, Orange County/California

[2] B. Farhang-Boroujeny and R. Kempter, "Multicarrier communication techniques for spectrum sensing and communication in cognitive radios", submitted to *IEEE Communications Magazine*, September 2006

[3] T.A. Weiss and F.K. Jondral, "Spectrum pooling: an innovative strategy for the enhancement of spectrum efficiency," *IEEE Communications Magazine*, Vol. 42, No. 3, March 2004

[4] S. Haykin, "Cognitive radio: brain-empowered wireless communications," *IEEE Journal Selected Areas in Communications*, vol. 23, Februrary 2005

[5] P.P. Vaidyanathan, *Multirate Systems and Filter Banks*, New Jersey, Prentice Hall, 1993

[6] Laakso, T.I. and Valimaki, V. and Karjalainen, M. and Laine, "Splitting the unit delay [FIR/all pass filters design]", *IEEE Signal Processing Magazine*, volume 13, January 1996

[7] Pool Yuen Kam, Qong Wu, and Wee Beng Lee,"Improved estimation of the frequency of a single sinusoid in noise using a new measurement model", *Communication Systems*, volume 2, 2002

[8] Oerder, M. and Meyr, H., "Digital filter and square timing recovery," IEEE Transactions on Communications, NO 5, volume 36, May 1988

[9] The IEEE LAN/MAN Standards Committee, "802.22 WG on Wireless Regional Area Networks (WRANs)", avilable at http://www.ieee802.org/22

[10] N. Abramson, "VSAT Data Networks", Proceesings of the IEEE, volume. 78, NO. 7, July 1990

[11] Emulab Fixed Wireless Test Bed, http://www.emulab.net/

[12] Manish Jain and Constantinos Dovrolis "Pathload: A Measurement Tool for End-to-End Available Bandwidth (2002)," Proceedings of the ACM SIGCOMM Conference, 2002