# IMPLEMENTING A BASE STATION USING THE SDR PLATFORM FOR COEXISTENCE OF HETEROGENEOUS WIRELESS SYSTEMS[1]

Shenghui Liao and Lichun Bao
University of California, Irvine, CA, USA
Emails: {shenghul,lbao}@uci.edu

## ABSTRACT

We propose a new approach to solving the spectrum scarcity problem, called spectrum access scheduling (SAS), which allows heterogeneous wireless systems to share the same spectrum band using protocol engineering methods. SAS treats the collection of wireless systems as an ecosystem, and optimizes the system performance by collaborative designs. The key architectural component in realizing SAS is the base stations of the wireless systems. We implement a base station using state-of-the-art software and hardware components, namely GNU Radio, OpenBTS and USRP software and hardware platforms, and demonstrate the feasibility of the coexistence of two heterogeneous wireless systems, GSM and CSMA, in the GSM-900 band.

## 1. INTRODUCTION

According to a recent spectrum usage investigation conducted by the FCC, the RF wireless spectrum is far from fully utilized [1]. According to the report, the typical channel occupancy was less than 15%, and the peak usage was only close to 85%. On the other hand, traffic demands in the wireless networks are growing exponentially over the years, and quickly overwhelm the network capacity of wireless service providers in some parts of the regions, such as hotspots or disaster-stricken areas where limited number of base stations remain. Adaptive and efficient spectrum reuse mechanisms are highly desirable in order to fully utilize the wireless bands.

We propose a spectrum access scheduling (SAS) approach to sharing spectrum among multiple heterogeneous wireless systems. For that purpose, we integrate and implement the base station functions of the wireless systems on the same hardware platform, and require that the wireless systems access the shared medium in a coordinated TDMA fashion so as to avoid channel access collisions. The concurrent execution of multiple wireless systems on the same base station hardware platform allows time-stringent protocol messages to be exchanged in real time.

We implement SAS among heterogeneous wireless systems using the SDR (software defined radio) hardware, because of the programmability and reconfigurability of the software and hardware components of the SDR platform. Specifically, we explore the feasibility of enabling the coexistence of two wireless systems, GSM and CSMA/CA systems, in the GSM-900 spectrum band using the USRP (Universal Software Radio Peripheral) hardware [6] and OpenBTS [4] and GNU Radio [5] software platforms.

In contrast to cognitive radios (CRs) [3] which differentiate wireless spectrum users as primary and cognitive ones, and share the spectrum in an opportunistic manner, our approach using SAS treats all spectrum users as equal share holders of the spectrum, and avoids system operation disruptions using a holistic protocol engineering approach. Overall, the base station hosting the heterogeneous wireless systems appears as if it is multilingual and coordinates the wireless systems to fully utilize the channel.

## 2. DEMO SYSTEM SPECIFICATION

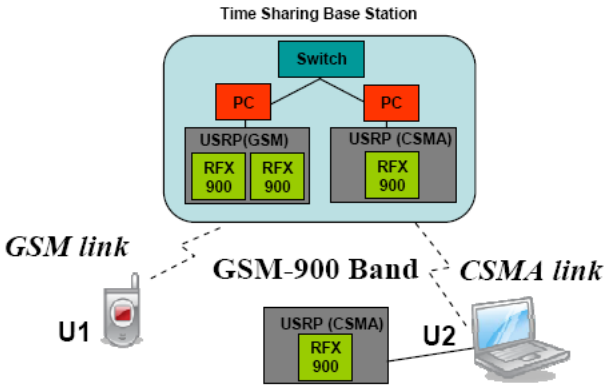


Figure 1 Configuration of base station

Figure 1 illustrates the system architecture for the coexistence of heterogeneous wireless systems in the GSM-900 band using SDR based base station. There are three components in the architecture, namely the base station

[1] This work was supported in part by the National Science Foundation (NSF) under grant No. 0725914.

comprised of two sets of one PC and a USRP interconnected by an Ethernet switch, the regular GSM mobile handset, and the mobile computer attached with a USRP. All components operate over the GSM-900 radio frequency band. We specify the details in the following sections.

### 2.1 Spectrum Configuration

| | GSM | CSMA |
|---|---|---|
| **Modulation** | GMSK | GMSK |
| **Center Frequency** | UL:895.8MHz<br>DL:940.8MHz | 895.8MHz |
| **Channel Bandwidth** | 200KHz | 1MHz |

Table 1 RF characteristics

We examine SAS mechanisms using smaller settings. As an example in this demonstration, we explore the coexistence mechanisms for GSM and CSMA systems over the GSM-900 RF bands. The RF characteristics of GSM and CSMA are shown in Table 1.

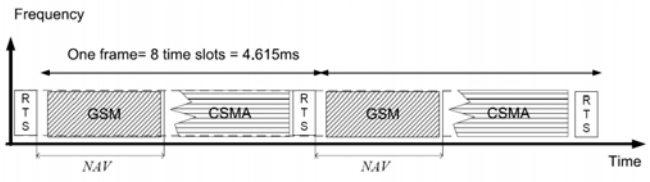


Figure 2 GSM and CSMA time share the GSM-900 band.

The goal of this system is to enable the two wireless systems, GSM and CSMA/CA, to share the 25 MHz wide uplink GSM band at the 895.8 MHz carrier spectrum in a way as shown in Figure 2. In the time domain, GSM and CSMA systems share the GSM-900 band in a round robin fashion. The granularity of the channel division is the GSM frame which lasts for 4.615 milliseconds. Controlled by the base station, three time slots of each GSM time frame are assigned to the GSM system, and the rest of the time slots are assigned to the CSMA system. However, the portions allocated to GSM and CSMA can be flexible, depending on the traffic demands of the wireless systems. In this demonstration, we only discuss the fixed time slot allocations to each of the wireless systems for simplicity.

In order to prevent mobile CSMA/CA stations from accessing the wireless channel inadvertently while the GSM stations are accessing the channel, the base station sends RTS (Request To Send) control frames with NAV (Network Allocation Vector) information to itself so as to reserve the channel time periods for GSM operations [7]. In Figure 2, the NAV period covers the duration in which the GSM system operates. The RTS control frame has been addressed to the base station itself so that no other stations will reply to the RTS request.

### 2.2 Computing Hardware Configuration

An SAS base station consists of two sets of a general purpose computer (PC), one USRP (Universal Software Radio Peripheral) board for signal sampling, and an RFX 900 boards for RF signal transmission and reception.

For simplicity, we use two sets of computing hardware systems to virtually simulate a single base station that executes the GSM and CSMA systems, respectively. The two computers are synchronized within a few micro-seconds accuracy using the IEEE 1588 Clock Synchronization standard, so that the individual wireless systems can fine tune their channel access at very high precision. Individually, each of the hardware sets executes the wireless system operations using the existing software components during its allocated time slots. If we were to implement the base station using a single computer, the real-time scheduling function alternating the two wireless system executions for channel access would have been a complicated issue, involving OS kernel programming to satisfy the real-time scheduling requirements. Using two separate computers to implement a virtual base station eliminates the complexity of the OS kernel programming, and allows us to focus on protocol engineering issues for coexistence.

The PCs are configured with Pentium-D Dual Core 2.8GHz processors and 1GB RAM, with USB 2.0 ports for connections with the USRP boards.

The USRP provides a low cost development platform for supporting the SDR RF frontend. A USRP board consists of one motherboard that holds ADC (Analog to Digital Converter) and DAC (Digital to Analog Converters) circuits, and a Field Programmable Gate Array (FPGA). The FPGA implements the control logics to reduce the sampling rate and send the signal samples to the computer over a USB 2.0 connection. The USB 2.0 connection has a theoretic data rate limited to 480 MBit/s. In practice, the data rate of the USB connection is much lower [8].

In order to operate in the GSM-900 bands, both USRPs for the GSM and CSMA systems use daughterboard RFX 900 as the RF frontend for RF signal transmission and reception purposes. RFX 900 modulates the output signal over the carrier frequency and extracts input signals from the carrier. The daughterboard and the USRP motherboard compose a complete RF transceiver system. The daughterboard RFX 900 operates at 750 to 1050 MHz and its transmit power of RFX 900 is 200mW [10][11].

### 2.3 Software Configuration for the GSM Component

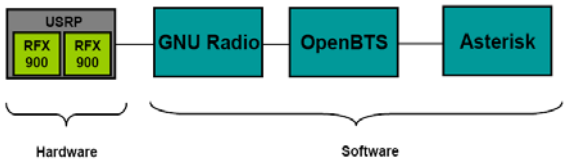


Figure 3 GSM system setting

In this demonstration, the GSM system is implemented over an open-source project, OpenBTS [4]. Figure 3 illustrates the control and data flows of applying the OpenBTS system to support communication through GSM.

The OpenBTS system uses the USRP and RFX 900 daughterboards to receive and transmit GSM signals. The Asterisk is used to control and manage calls for the cellular phones under the OpenBTS network. In OpenBTS, OpenBTS interfaces with `libusrp` libraries to use USRP.

GNU Radio is a free software development toolkit [5]. In GNU Radio, the signal processing runtime and processing blocks are provided to implement software radios using readily-available, low-cost external RF hardware, such as USRP and commodity processors. While performance-critical signal processing blocks are implemented in `C++`, GNU Radio applications are primarily written by `Python`, a dynamic object-oriented scripting language. `SWIG` is used as an interface compiler to allow easy integration of `C++` and `Python`. This way, `Python` codes simply connect the signal processing blocks and allow them to run at native speed without any interpretation.

Overall, the OpenBTS uses the USRP to present a GSM air interface ("Um") to standard GSM handsets and uses the Asterisk PBX (Private Branch Exchange) software to control and manage calls between GSM handsets and VoIP endpoints. This way, OpenBTS supports GSM functions without supporting functions provided by traditional GSM components – the Base Station Controllers (BSCs), Mobile Switching Centers (MSCs) or Visitors Location Registers (VLRs). Instead, each GSM handset is treated as a Session Initiation Protocol (SIP) endpoint. OpenBTS can be installed and operated at about 1/10 the cost of current GSM technologies, and still be compatible with most of the handsets already on the market [12].

Specifically, when a GSM circuit is established over the air interface, a GSM physical channel is defined by a certain time slot and an Absolute Radio Frequency Channel Number (ARFCN), which includes a pair of physical radio carriers. Furthermore, each physical channel is time-multiplexed into multiple logical channels. The GSM timing is driven by the serving Base Transceiver Station (BTS) through the Synchronization Channel (SCH) and Frequency Correction Channel (FCCH). The SCH transmits the current value of the TDMA clock. The FCCH generates a tone on the radio channel that is used by the mobile station to discipline its local oscillator.

### 2. 4 Software Configuration of the CSMA Component

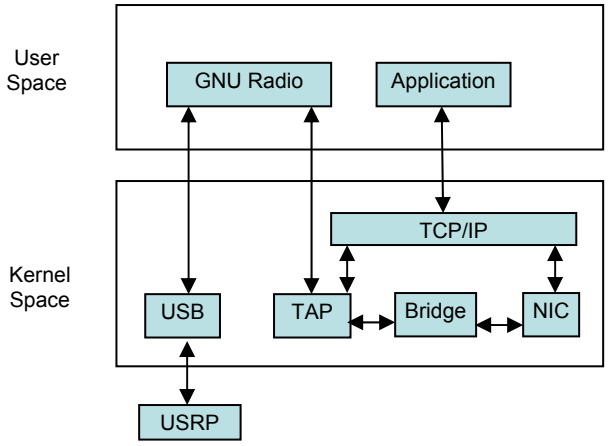


Figure 4 Integration of SDR in the CSMA access point

Figure 4 shows the implementation of the CSMA based wireless system for communication purposes using the USRP and GNU Radio components. One of the GNU Radio examples, `tunnel.py`, has been modified to implement the CSMA system in this demonstration. GNU Radio process runs in the user space, subject to the operating system scheduler.

A TAP [13], a virtual Ethernet network device, is generated in the kernel, so Ethernet frames can be sent and received through the TAP. A virtual bridge is generated to connect the TAP and the physical network interface card (NIC) that connects to the Internet. Application handles the traffic from both TAP and NIC interfaces over the TCP/IP protocol suite.

## 3. DEMONSTRATION RESULTS

As shown in Figure 1, we use existing software and hardware platforms to demonstrate our spectrum access scheduling scheme for the coexistence of heterogeneous wireless systems over the GSM-900 band.

Specifically, we will show that we can carry out the following two communication sessions simultaneously:

- To make a phone call using regular GSM cell phones through the GSM components of the base station configured in this demo;

- To ping a host or browse the Internet using a regular computer through the CSMA components of the base station configured in this demo.

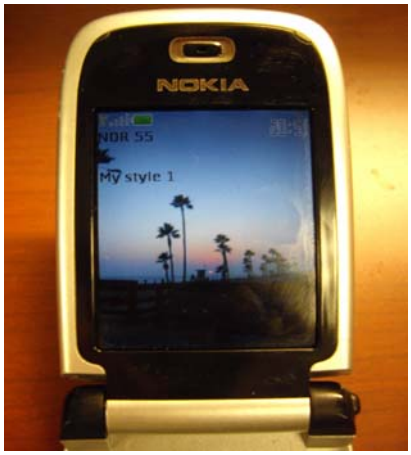


Figure 5 A cell phone under our network (NOR 55).

To demonstrate the communication through GSM system, we show that we can make a phone call through our base station, configured as network NOR55 shown in Figure 5, using our commodity cell phone. The modified GSM-900 system only transmits on the time slot 0, 1, and 2.

To demonstrate the communication through the CSMA system, we carry out a `ping` operation through the CSMA link using a regular notebook. The IP address of the regular computer is 128.195.55.229. In addition, we can run different applications (such as web browsing, emails etc.) through the CSMA wireless system. Figure 6 shows a screenshot while running the `ping` function.

In either application, the key point that we demonstrate is the SAS capability that allows heterogeneous wireless systems to share the same RF bands to improve the spectrum utilizations.

## 4. CONCLUSION

We have demonstrated the spectrum access scheduling (SAS) approach and system implementation details for spectrum reuse purposes. In the demonstrations, we showed that two wireless systems, the GSM and CSMA systems, can share the GSM-900 uplink band in a TDMA fashion. We proved that SAS is not only feasible, but also a promising approach to solving the spectrum scarcity problems.

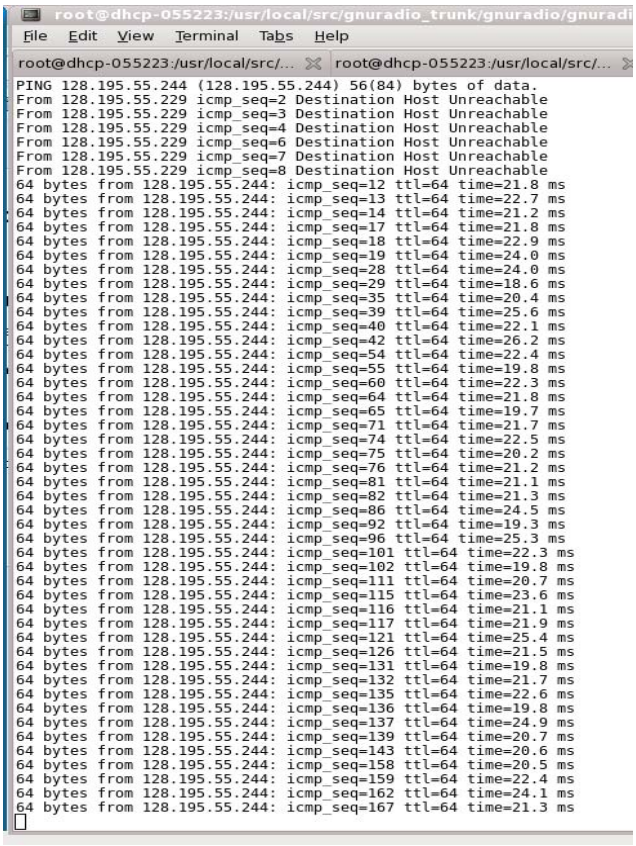


Figure 6 A screenshot while using the ping function

## REFERENCES

[1] FCC. *Spectrum Policy Task Force Report*, ET Docket No. 02-155, Nov. 2002.

[2] M. Golio and J. Golio. *RF and Microwave Applications and Systems*, John Wiley, 2008.

[3] I.F. Akyildiz, W.-Y. Lee, M.C. Vuran, and S. Mohanty. *Next generation dynamic spectrum access cognitive radio wireless networks: a survey*. Compute Networks, pg 2127-2159, 2006.

[4] OpenBTS. http://openbts.sourceforge.net.

[5] Gnu Radio. http://www.gnu.org/software/gnuradio.

[6] USRP. http://www.ettus.com.

[7] C. Chiasserini and R. Rao. *Coexistence Mechanisms for Interference Mitigation between IEEE 802.11 WLANs and Bluetooth. In* Proceedings of the IEEE INFOCOM, 2002.

[8] T. Schmid, O. Sekkat, M.B. Srivastava, *An Experimental Study of Network Performance Impact of Increased Latency in Software Defined Radios*, in Proc. of the Second ACM International Workshop on Wireless Network Testbeds, Experimental Evaluation and Characterization, pg 59-66, 2007.

[9] S. Valentin, H. Malm, H. Karl, *Evaluating the GNU Software Radio Platform for Wireless Testbeds*, Technical Report TR-RI-06-273, Feb. 2006.

[10] *Transceiver Daughterboards for the USRP Software Radio System*.http://www.ettus.com/downloads/ transceiver_dbrds_v3b.pdf.

[11]*Universal Software Radio Peripheral,* Wikipedia, http://en.wikipedia.org/wiki/Universal_Software_Radio_Peripheral.

[12]*Kestrel Signal Processing* http://www.kestrelsp.com/ OpenBTs.html.

[13] *Universal TUN/TAP driver* http://vtun.sourceforge.net/tun.