WARP WIRELESS OPEN-ACCESS
RESEARCH PLATFORM

WARP MAC Lab: cogMac

Chris Hunter & Patrick Murphy

Rice University
WARP Project

April 27, 2011

1 Introduction

In this lab we will extend the halfMac protocol to support dynamic changes to the active channel. Upcoming channel changes will be advertised by the server node, and clients will use hardware timers to schedule the channel change.

By default, this project implements a full halfMac client with one notable difference; here, ACKs are automatically generated by our autoResponse core. Details about this subsystem can be found at http://warp.rice.edu/w/OFDM/MIMO/Docs/AutoResponse.

The skeleton code for this exercise is in the `COGMAC_CLIENT` software application in XPS. Make the necessary changes to the skeleton code to complete cogMac.

1.1 Channel Change Notifications

The server node will periodically schedule a channel change. It will advertise this scheduled change by transmitting a broadcast packet of type `PKTTYPE_COGHOPPENDING`. Each broadcast packet will specify the new channel (in field `header.cogParam`) and the time left before the change occurs (in field `header.timeLeft`, in units of milliseconds).

Your node must listen for these broadcast packets, and interpret the header fields to schedule your own channel hop. You will use hardware timers and a new user-level callback function for timer events (see the skeleton code) to implement the schedule. Figure 1 illustrates the new states you will implement.

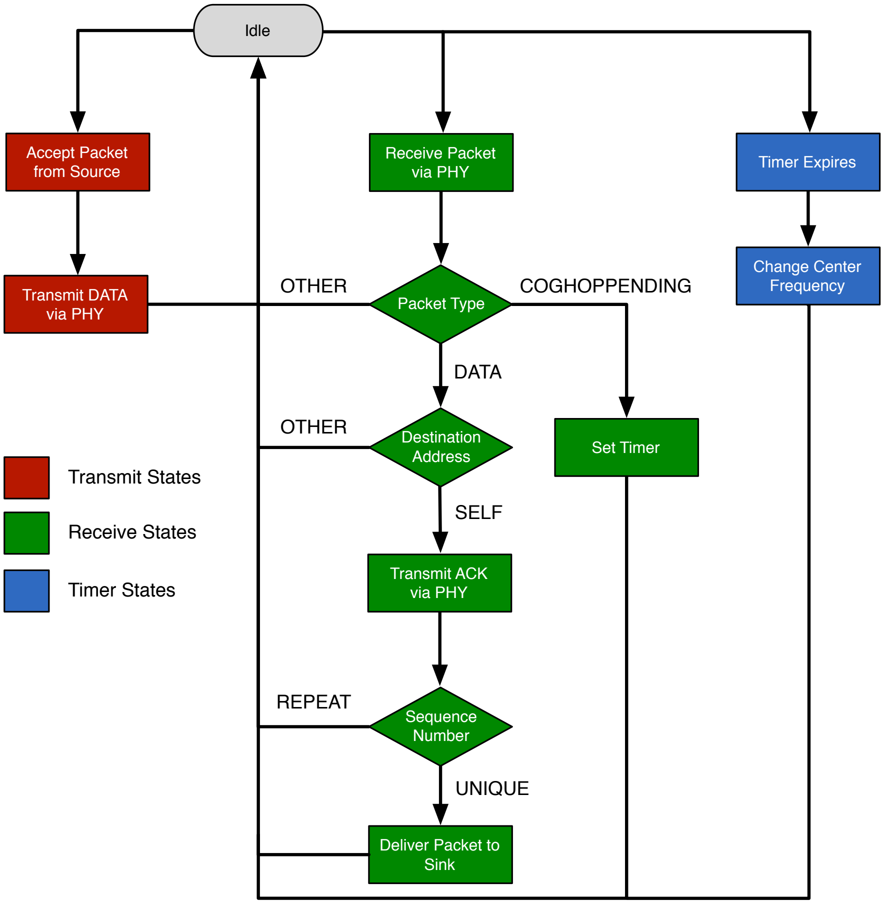

Figure 1: *cogMac State Diagram*

Test your cogMac receive and timer states before proceeding with the next section.

1.2 Channel Change Requests

In addition to advertising upcoming channel changes, the cogMac server also accepts requests from clients for preferred channels. Implement the following functionality:

1. Use the UART callback to listen for input from the PC keyboard.
2. When the user types a number in [0,9], send a wireless packet to the cogMac server with header values:
   - `header.pktType`: `PKTTYPE_COGREQ`
   - `header.destAddr`: `0` (ID of cogMac server)
   - `header.srcAddr`: `myID`
   - `header.cogParam`: Your desired channel (integer in [1,10])
3. Listen for a response from the server with type `PKTTYPE_COGRESP`. This response will be addressed to your node (i.e. the `srcAddr` field of your `PKTTYPE_COGREQ`). In this response the server will indicate the status of your request in the value of the `header.cogParam` field:
   - `COGRESP_SUCCESS`: Request accepted; your channel will be used next
   - `COGRESP_TOOLATE`: Request denied; another user has already requested a channel
   - `COGRESP_ILLEGALCHAN`: Request denied; the requested channel is unavailable
4. Continue monitoring for `PKTTYPE_COGHOPPENDING` packets and schedule the appropriate channel changes.

2 Testing your MAC

As in the previous lab, use VLC to test your cogMac implementation. If it works, your video stream should continue uninterrupted across frequency changes.