WIRELESS OPEN-ACCESS
RESEARCH PLATFORM

Lab 6: lemmingMac

Chris Hunter

Rice University
CMC Lab

Document Revision 1.1

March 22, 2007

1 Introduction

The WARP networking lab is intended to introduce the tools required for MAC development on WARP. In previous labs, code on the PowerPC was used to drive cores running in the FPGA fabric. We will make extensive use of the WARPMAC framework (http://warp.rice.edu/trac/wiki/WARPMAC). The framework will handle the hardware interaction, and we will program a basic MAC.

In the previous lab, a central transmitter streamed UDP audio to each of the participants' PCs in a topology shown by Figure 1.

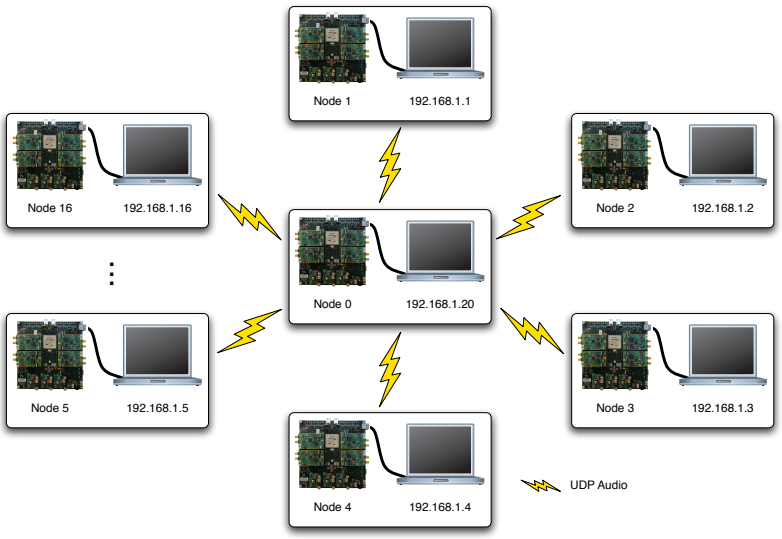

Figure 1: *Lab Topology*

This lab will have an identical configuration. The twist here, however, is that the transmitter will periodically hop to a different 802.11 channels (1 through 11). Before doing so, it will broadcast a special "FOLLOW" packet to notify users that the transmitter is about to hop and to what channel. If your MAC algorithm is coded properly, your node should hop frequencies to follow the central transmitter in its journey through the spectrum. This behavior is shown in Figure 2. At the application level, the audio stream should be maintained, and the frequency hopping should be indiscernible.

Figure 2: *User nodes should follow the central transmitter as it moves to a new part of the spectrum.*

The user's code is responsible for all of the behavior of the previous lab. You can use your completed code, or you can use the completed code provided. Additionally, you are responsible for the following behavior:

- If the packet was sent to the broadcast address (255:255:255:255:255:255), and it was a "FOLLOW" type, switch to the channel specified in the payload of the packet. The critical elements of the FOLLOW Macframe are shown in Figure 3

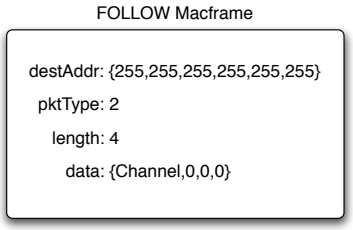

Figure 3: *The FOLLOW Macframe structure.*

The full software state of the client node is shown in Figure 4.

It is important to note that, while the transmitter has a MAC capable of retransmissions, the user's MAC will not. Because of the unidirectional nature of UDP, the only traffic that needs to be send to Node 0 is an ARP reply to establish the ARP table on both computers.

The WARPMAC_API will be required throughout this lab exercise. Skeleton code will be given that should compile without user modification and perform the behavior of the previous lab. By default, the project will blink user LEDs on the board upon the reception of packets that pass the checksum. Within the given code will be comments describing the code that needs to be written.

2 Extensions

- Gather statistics on a per-channel basis. Can you infer channel conditions using aggregate statistics at the MAC layer?
- If your node misses the FOLLOW packet, it will wait in its channel until the transmitter happens to jump back in to that channel. Try to infer when this occurs by listening for packets. If no packets have been received for some time, assume you lost sync. Can you sweep through the spectrum and try to lock back on?

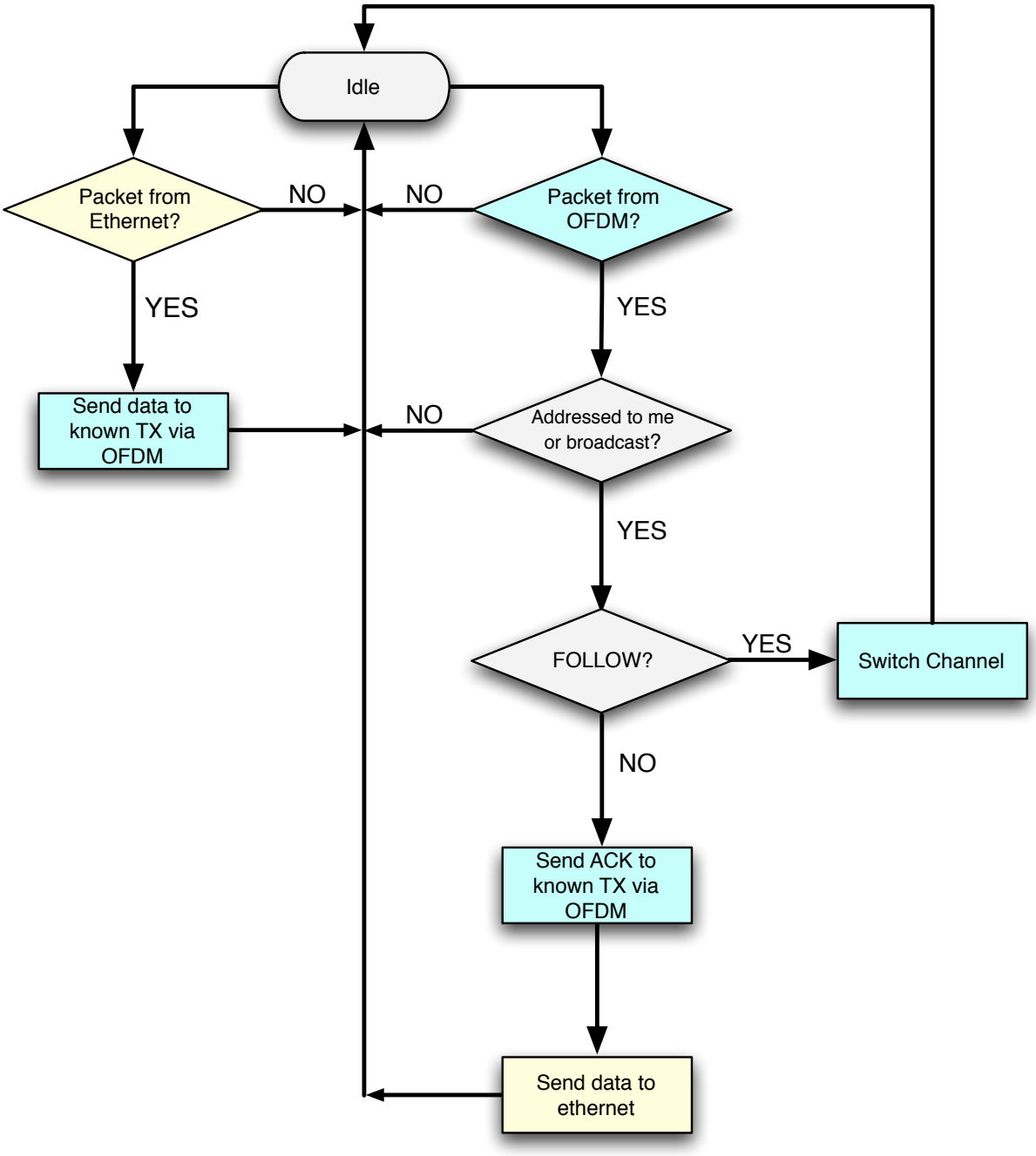

Figure 4: *State diagram for frequency-hopping, LemmingMAC*