CS 466 LAB PROJECT
Wireless LAN System

Project Specification

Introduction

This document presents the project specification for the Wireless LAN System lab project. The remaining lab experiments using the Motorola MCORE processor will all be concentrated on implementing this project. The basic objective of this document is to give an overview of the project, how it will work and how it will be divided into weekly labs.

Basic Project Description

The project very simply consists of a single base-station and 12 workstations. The 12 workstations will attempt to communicate with the central base-station by using a well-defined wireless network. The communication will be in the form of wireless serial bit-stream data containing messages. For a description of these messages as well as their required format, refer to the Protocol Specification.

Project Requirements

This project requires that every student implement a workstation in the Wireless LAN System that will communicate with the central base-station through serial bit-stream message passing.

Project Outline

To implement a workstation in this Wireless LAN System, a control program must be written to run on the MCORE processor that will control message passing with the common base-station. The control program will require specific modules to do this. Note: These modules can be combined in any manner that each student chooses.

**Bit Receiver**
The Bit Receiver module will watch the data line of the receiver for the first ‘one’. When it receives this, it will begin the Bit Sampler, the Bit Timer and the Preamble Recognizer modules.

**Bit Sampler**
The Bit Sampler module will sample the data returned from the receiver and record the value of the bits received.
**Bit Timer**
The Bit Timer module will keep track of each bit period and trigger the Bit Sampler as to when it needs to sample the data line. This should be written using the PWM module as a general-purpose bit-period timer.

**Preamble Recognizer**
The Preamble Recognizer module watches the bits recorded from the Bit Sampler for the specified preamble value (0xCCCC). Once it detects an entire preamble sequence, it first triggers the Message Decoder to begin and then turns itself off (or at least stops looking for a preamble sequence).

**Message Decoder**
The Message Decoder will be used to decode the remaining bits of the transmitted message. It must first collect the eight bits that specify the workstation ID# and determine whether to keep the message or discard it. If the message must be discarded, the Message Decoder must reset the Preamble Recognizer to begin checking for a new preamble and then wait until it is triggered again. If the message is kept, the Message Decoder collects the next four transmitted bits and determines the control type of the message. This value will determine the exact length of the transmitted message. The Message Decoder must collect the rest of the bits and then trigger the Execution module. Once this is complete, it must turn itself off.

**Execution**
The Execution module must determine the contents of the message collected by the Message Decoder and determine the appropriate action to take in response to the message. This is determined according to the Protocol Specification.

**Transmitter**
The Transmitter module must first compose a message according to the requirements specified in the Protocol Specification, and then transmit the message.

**Project Timeline**
The timeframe for completion of this project is six weeks. The program modules are divided into weekly labs to help minimize overall congestion.

**Project Grading**
This project dictates almost half of the lab requirement for this class. Let’s just say it is worth a lot of points. Good Luck!