

Streaming Audio Player

EDA385 - Advanced Embedded Systems Design

Project Proposal

Anil Kumar Metla

Anusha Gundarapu

Ramu Akarapu

1. Introduction

The project streaming audio player involves playback of audio which is streamed over the network. Figure 1 shows the block diagram of the streaming audio player. The user controls are displayed on the LCD and the keyboard acts as input device. The requests for play/pause/stop is issued via the keyboard, based on the input and current state the streaming application interacts with the lower layers takes appropriate action. For example in case of a play the user inputs the selection as play via the keyboard, then the streaming application connects to the server via the network and starts the streaming of a song/playlist over the network. The other end of the network is currently expected to be a PC. The streamed data is decoded via the audio decoder and played through the amplifier. While the song is being played, current song information and user controls are displayed via the LCD. Currently the RF is expected to be the physical medium of transmission and reception for the network, with the communication protocol being zigBee protocol stack. The PC is also expected to be using a similar setup running a zigBee stack. However based on the availability of the RF USB dongle the network the protocol stack and its related modules might change.

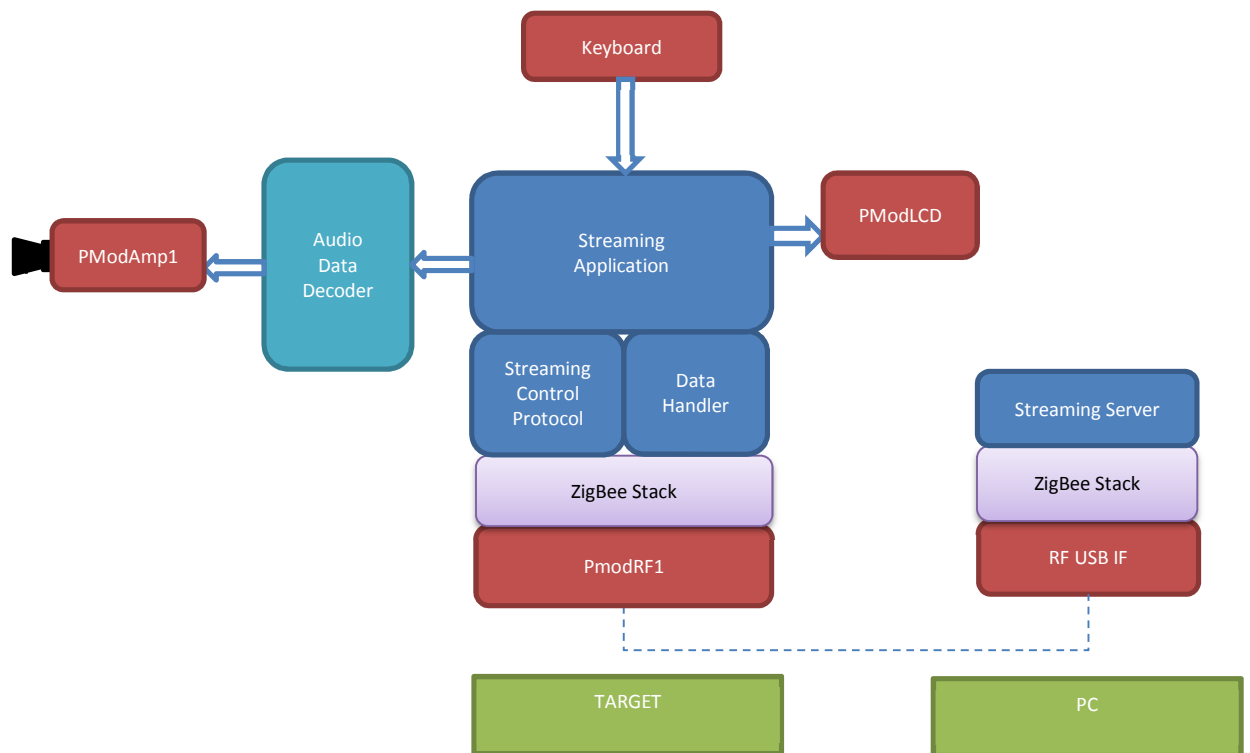


Figure 1: Block Diagram

2. Detailed Description

The streaming audio player system is divided into following key tasks.

1. Peripheral modules and Controllers for peripheral modules
2. Mp3 hardware decoder
3. Streaming Application
4. Streaming framework
5. Network stack
6. Streaming Server configuration

Peripheral modules and controllers

The implementation of the streaming audio player requires following peripheral modules pMods, and controllers to communicate with them.

They are

- a. PmodAmp1
- b. PmodLCD
- c. PmodRF1/PmodNIC

Along with these the keyboard via the PS2 interface requires the implementation of the hardware controllers.

PmodAmp1: This peripheral module requires a controller to feed data into its left and right channels. As the peripheral module has no inbuilt memory and any other control circuit, the controller has to be smart enough to feed the data to the amplifier all the time.

PmodLCD: PModLCD is a 16x2 LCD display and can be interface via the UART.

PmodRF1/pMod NIC: This Pmod requires a controller and a stack on top for networking purposes and the final selection of the module depends on the availability of the RF dongle at the PC side.

PS2: Keyboard is interfaced via the PS2 interface and a controller is required to interact with it.

Mp3 hardware decoder

A hardware mp3 decoder is implemented as part of the system to speed-up the decoding as well as to reduce load on the processor. The mp3 is the only medium of audio that is currently planned to be supported for streaming.

Streaming Application

A streaming Application will be implemented in software to control the various controllers, network interface, and finally acts as user interface. The streaming application takes input from key board and then based on the input it performs various operations such as streaming and playing of a song, changing the info on the LCD etc. This is the core of the entire system.

Streaming Framework

Streaming framework is implemented in the software and takes care of the implementation of the streaming protocols required. Realtime Streaming protocol (RTSP) is selected as control protocol and the Real time transport protocol (RTP) will be used for data transport. The framework is expected to be transparent to underlying network stack.

Network Stack

Based on the network physical medium of transport the network stack can either be a zigBee on RF or TCP/IP on NIC.

Streaming Server Configuration

The streaming server on the PC has to be setup according to the usage of the network protocol, and also it has to be configured to support the mp3 streaming as well as play list streaming. The entire PC configuration is covered under this task.

3. Project Plan

As the tasks are already identified the project is planned for parallel implementation. The task distribution will be dynamic and distributed among the team equally.

Task	Duration	Week
Controllers for peripheral modules	2 weeks	Week 3 – week 4 – week 5
Mp3 decoder	2 weeks	Week 3 – week 4 – week 5
Streaming framework	2 weeks	Week 3 – Week 4 – week 5
Streaming application	1 week	Week 3 – week 4 – week 5
Streaming server configuration	1 week	Week 3
System integration and testing	2 weeks	Week 6 – week 7
Bug fix	2 weeks	Week 7 – week 8
Report	2 weeks	Week 3 – week 4 – week 5 – week -6 –week 7 – week 8 – week 9