Implementation of a RTSP streaming based on ARM processor

Sujal Patel[1]
[1]M.E. Student
[1] VLSI & Embedded System Design Department
[1] GTU PG School, Ahmedabad.

Abstract— this paper presents a design and implementation of RTSP (Real Time Streaming Protocol) streaming server. It utilizes arm BCM2835 and Linux operating system as a platform and develops network streaming media on the arm BCM2835 through rtsp streaming server and VLC media player. BCM2835 is a cost optimized, full HD and multimedia processor. A real time streaming media and some basic function of Real Time Streaming Protocol is implemented in this paper. This implementation proves to stream audio and video data smoothly. So we can conclude that this RTSP server can be embedded into many multimedia applications.

Keywords: RTSP, arm BCM2835, vlc player.

I. INTRODUCTION

With the development of internet and multimedia technologies, Streaming of multimedia over internet is present and future of the web, and is currently facilitate a huge demand for online customers. The difference between streaming multimedia and traditional multimedia is that the former can acquire smooth data flow directly and need not to download the data to the local machine. So we call the transmission of getting smooth data stream from network streaming transmission. Streaming media is audio or video content sent in compressed form over the internet and played immediately, rather than being saved to the local machine. With streaming media, a user does not have to wait to download a file to play it. In this paper a RTSP server is embedded on to Linux platform which uses arm BCM2835 hardware. RTSP server uses RTSP and RTP API's and utilities.

II. THE STRUCTURE OF THE SYSTEM

Host machine and embedded arm board is connected with a serial port and/or local network through Ethernet cable. Running the super terminal as ARM embedded board display terminals on the host PC, by the serial port, Uboot, Zimage and file system can be downloaded into the board. In this way, the authors can debug ARM embedded board in the host PC through hard drive cross-compiled program. The tests are implemented in the local area network. Streaming media server and embedded development environment diagram is shown in figure 1

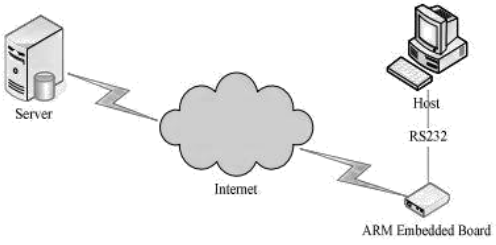

Fig. 1: System Diagram

A. RTSP Streaming Server

Streaming Server is an open source project intended for developers who need to stream QuickTime and MPEG-4 media on alternative platforms such as Windows, Linux, and Solaris, or those developers who need to extend and/or modify the existing streaming server code to fit their needs.It allows you to send streaming media to clients across the Internet through the streaming media protocols. As shown in figure 2, in RTSP streaming, the presentation description file may be obtained by the client using HTTP or other means such as email and may not necessarily be stored on the media server.

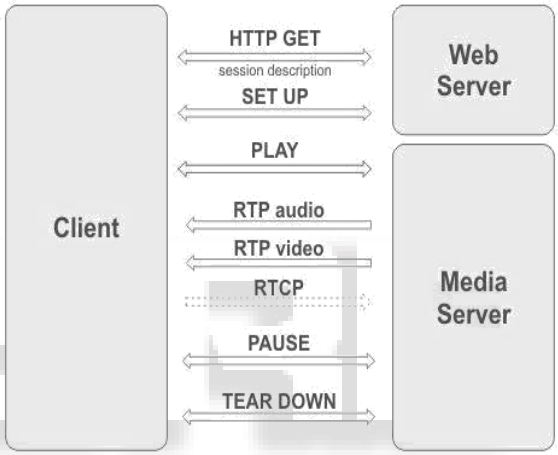

Fig. 2: Working of RTSP Streaming Server

B. ARM Processor

The BCM2835 is a cost-optimized, full HD, multimedia applications processor for advanced mobile and embedded applications that require the highest levels of multimedia performance. Designed and optimized for power efficiency, BCM2835 uses Broadcom's Video Core® IV technology to enable applications in media playback, imaging, camcorder, streaming media, graphics and 3D gaming [4].

The ARM chips have the general characteristics of the RISC systems. Moreover, some special advanced technologies are adopted in the processor by engineers to ensure the high performance with the minimized chip size and reduced power consumption.

C. Streaming Protocols

The main protocols used to perform real-time data stream are usually based on the following three protocols: Real-Timing Stream Protocol (RTSP), Real-time Transport Protocol (RTP) and the Session Description Protocol (SDP).

1) RTSP:

The Real-Timing Streaming Protocol (RTSP) acts a control protocol to media streaming servers. RTSP provides an extensible frame to enable controlled, on-demand delivery of real-time data [3]. It establishes connection between two

end points of the system and manages the streaming media data. This protocol is intended to control multiple data delivery sessions, provide a means for choosing delivery channels such as UDP, multicast UDP and TCP, and provide a method for choosing delivery mechanisms based on RTP.

2) RTP:

The Real-timing Transport Protocol (RTP) defines a standardized packet format for delivering audio and video over IP networks [2]. It also manages compensation and detection of wrong data sequence, which could occur occasionally during transmission over TCP network. RTP is applied extensively in embedded systems and intelligent terminals, such as PDA, television applications and video services. In addition, RTP also supports multicast transmission of streaming data.

3) SDP:

The Session Description Protocol (SDP) is intended for describing multimedia communication sessionsfor the purposes of session announcement, session invitation, and parameter negotiation [1]. It describes the initialization parameters for the media session. This protocol does not deliver media itself but it is normally used to setup the session and facilitate the playback of the streaming data. SDP is intended to be feasible to support new video and audio formats.

III. THE REALIZATION OF STREAMING PROTOCOL

In order to realize RTSP streaming through an RTSP Server it is necessary for us to cross-compile and install library on the host. It supports open standards such as RTP/RTCP, RTSP and SIP for streaming. The libraries are well-suited for use within embedded systems. It is designed to support many video and audio formats we usually use such as MPEG/H.264 video.

The Schedule of RTSP Server is as follows: [6]

(1) Create Task Scheduler and Basic sag class
(2) Get access to streaming media address, and create RTSP Client object
(3) RTSP Client object sends option commands and parse the response of the server side, accesses the command set that can be used.
(4) RTSP Client object sends the DESCRIBE commands, and accesses streaming media to describe the SDP string in the server-side when feedback is received.
(5) Create Media Session object to parse the SDP string, RTP and RTCP are used for communication during this process including the choice of protocol and port
(6) Create specific instance of the Source and Sink object according to the different types of streaming media
(7) RTSP Client object sends SETUP and PLAY commands, after that the server begins to transfer streaming data
(8) Task Scheduler starts event processing loop, process the streaming data by monitoring data packets and calling the registered function.

IV. TEST PROCEDURE AND RESULTS

To start streaming Audio and Video data we need to setup hardware configuration and software testing before it. First of all we need to power up the ARM board and connect it to the local area network or with serial cable. After that open the application on the board and start the streaming server by putting command line as shown in figure 3 ./rtprtspdemo gives the command line in which we need to provide complete information for streaming those are, server ip, port of server for RTSP, number of session to create, name of the session, type of codec and file path of the codec from which a server fetch file and streamed it to the destination.

Fig. 3: commend line for streaming

After doing this procedure server will start and we need to open multimedia player like vlc, media player, mplayer etc. in that particular multimedia player there is an option for playing a file through network is available. By using that feature we can stream our media file over network by putting network URL in to the network protocol box e.g. rtsp: //server ip:server port name/ file name. For example in vlc media player how to enter url is shown in figure 4, after putting url into the given box just push "play".

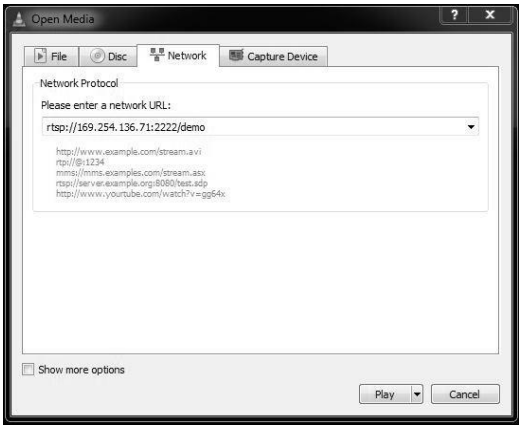

Fig. 4: playing file through RTSP network streaming media player

We devise a form to test some important parameters of the network streaming media player. The test results are shown in TABLE I.

Requirements of Performance	Test Results
Streaming	Hold Out

Command of play and Suspension	Hold Out
Network delay	Small
Resolution	1080*960
Audio and Video Synchronization	Synchronous

Table. 1: The performance of streaming server

V. CONCLUSION

This paper describes the implementation of a network streaming server based on the ARM processor BCM2835, and the system makes use of RTSP protocol. In addition it uses open source media player in which play, pause, stop and other functions of streaming are available. At last the performance results show that it completely satisfies the embedded users demand and performs well.

REFERENCE

[1] Handley, M. and V. Jacobson, "RFC 2327 SDP: Session Description Protocol", April 1998.
[2] H. Schulzrinne, S. Casner, R. Fredernick, and V. Jacobson, "RFC-1889 RTP: A Transport Protocol for Real-Time Applications", January 1996.
[3] H. Schulzrinne, A. Rao and R. Lanphier, "Real Time Streaming Protocol (RTSP)", April 1998.
[4] http://www.broadcom.com/products/BCM2835
[5] http://www.raspberrypi.org/wp-content/uploads/2012/02/BCM2835-ARM-Peripherals
[6] http://en.wikipedia.org/wiki/LIVE555
[7] Yan liu, Yu liu, "THE RESEARCH OF STREAMING MEDIA MUTUAL DIGEST AUTHENTICATION MODEL BASED ON RTSP PROTOCOL", International Conference on Wavelet Analysis and Pattern Recognition, Hong Kong, 30-31 Aug. 2008.

