International Journal of Control and Automation
Vol. 7, No. 12 (2014), pp. 47-60
http://dx.doi.org/10.14257/ijca.2014.7.12.05

The Design and Implementation of Novel Intelligent Terminal Driving Recorder

Yanshu Jiang and Jia Lv

School of Automation, Harbin University of Science and Technology, Harbin,China
E-mail: jiangyanshu0@163.com

Abstract

This thesis designs a new type of driving recorder, utilizing intelligent terminal as the client, driving recorder as a serve. In data processing, information display and data storage. Utilizing intelligent terminal such as a smartphone or tablet itself processor, a display and a memory card to replace the traditional corresponding feature of driving recorder. In the aspect of data transmission, utilize WIFI for information interaction between driving recorder and intelligent terminal, realizing wireless communications. Through the intelligent terminal to realize the data preview and playback function, driving recorder become more portable.

Keywords*: driving recorder, H.264, RTSP, RTP, Hi3516*

1. Introduction

Drive recorder can record and store data during vehicle driving process, the basic composition of drive recorder includes camera, sensor, image processor, encoder, CPU, memory card, Wireless transmission module, and so on. In the process of driving, drive recorder can record the road conditions circularly in every second. For investigating the responsibility of traffic accident [1].

People are becoming more and more like drive recorder, as it can record the process of driving. It can protect the driver's personal property safety, also to record moments of trip. Abroad, utilization rate is as high as more than 90%. When the European and American use drive recorder, the traffic accident reduce significantly. In China, an increasing number of drivers use drive recorder [2-3].

The traditional driving recorder has screen of 3 inches, pixel is low and the picture is not clear, volume is big and equipment cannot upgrade. With the development of smartphone and tablet, intelligent terminal depends on excellent man-machine interface and ubiquitous wifi module, contributed to the perfect combination of intelligent terminal and drive recorder [4-6]. Intelligent terminal replace end fitting of traditional drive recorder, could reduce cost and improve processing speed. Storage using SD card, if you want to analysis traffic data on computer, don't need a peripheral interface, just use the card reader can be implemented. Use wifi for the communication of driving record system to realize wireless transmission. You only need to download APP into your intelligent terminal, APP connect to driving recorder, you can preview real-time video or playback video, also can store picture or video in process of driving [7-8].

ISSN: 2005-4297 IJCA

2. Implementation Scheme

2.1 Designing of the Whole Structure

Drive recorder mainly consists of three parts, server, transmission network and client. Video server is the data sources in the driving record system. Complete video collection, compression and package sent to the network. As one of the core of drive recorder: transmission network, all data is inseparable from the network transmission, the innovation of this design is drive recorder itself can launch WIFI signal, at the same time developed Android and IOS software platform, after the intelligent terminal connected to the drive recorder, you can watch real time video in the intelligent terminal, playback the recorded video and other function. You can also watch the images and related video taken by remote control. Overall system block diagram is shown in Figure 1.

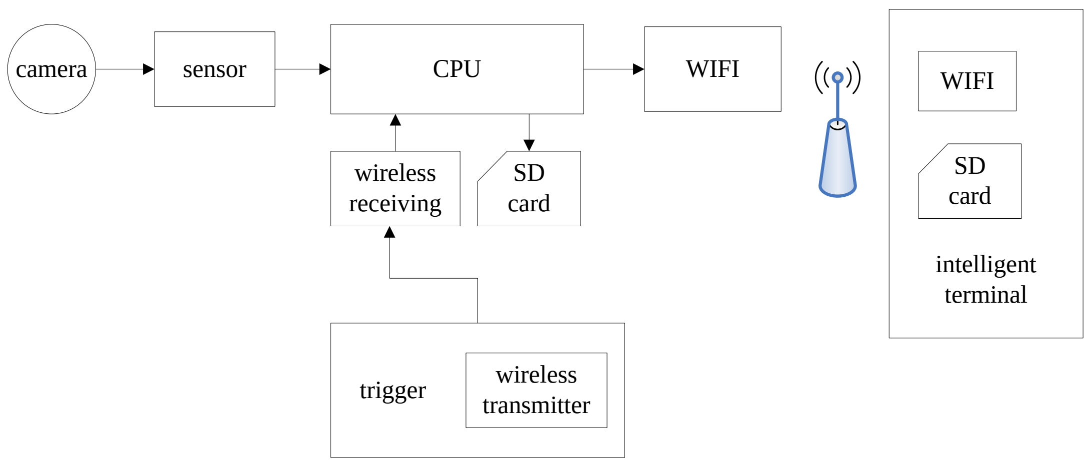

Figure 1. Overall Structure

2.2 Hardware Design

The processors of network video server generally adopt DSP or ARM, This scheme adopts the combination of video acquisition chip and embedded processors, Choosees Hi3516 chip as embedded processors of video server. Hi3516 is a professional high-end SOC chip which Haisi semiconductors developed for HD IP Camera product application. H.264 stream coding performance of 1080p@30fps, excellent ISP and the quality of video encoding, high-performance features such as intelligent speed engine, meet product features, performance and image quality of IP camera, at the same time, can greatly reduce the cost.

1. Processor system

Hi3516 used ARM926 as CPU of the whole system, processor embedded 16 KB instruction cache and data cache 16 KB, working frequency can up to 440 MHZ.

2. Video and image processing

Support 3D denoising, image enhancement, edge enhancement, de-interlace processing capacity, support video, graphical output resistance to flicker, support video zoom function of 1/16 ~ 8 x, support graphical zoom function of 1/2 ~ 2 x, eight areas OSD superposition of video pre-processing, two layers image superposition of video post-processing hardware.

3. Video coding processing capacity

H. 264 coding can support maximum resolution is 2M Pixel, Supports H. 264 and JPEG encoding and decoding, Supports bit rate control of H.264, *etc.*,

4. External memory interface

The system provide controller to support DDR3RAM, the maximum capacity to support 2 GbitDDR3 SDRAM.

5. Peripheral interface

Support the RMⅡ and MⅡ mode; Support 10/100 Mbit/s full-duplex or half duplex mode, RM Ⅱ interface providing PHY clock output.

6. Video interface

Support 8/10/12 bit RGB Bayer input , Clock frequency up to 74025 MHz, support BT. 601, BT. 656,support CVBS output 1 road, support load detection automatically , provide a video output interface of BT. 1120, used for outside enlarge HDMI or SDI interface, support 1080 p@ 30 fps output.

7. Audio interface

Integrates audio codec, supports 16 bit input and output of voice.

8. Other interfaces

Hi3516 provides a USB2.0 HOST interface, three UART interface, three PWM interface, supports POR, integrated high precise RTC and two channels SAR – ADC.

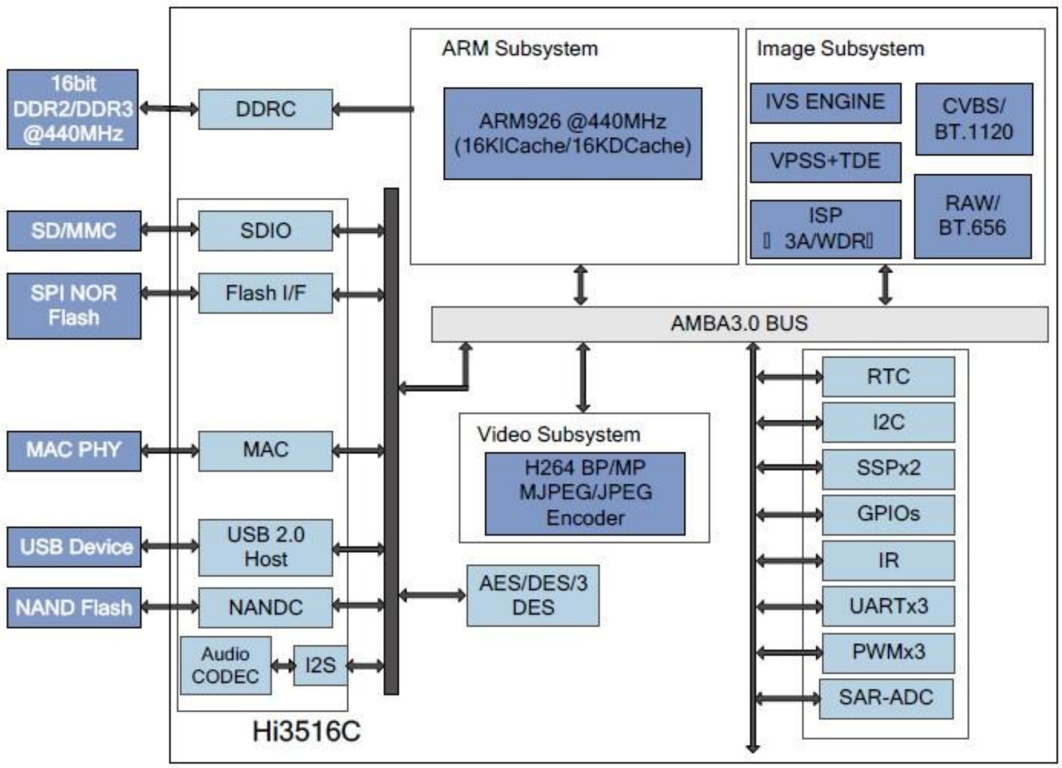

Figure 2. Chip Logic Diagram

This system increased 2.4 G wireless remote control for taking photos and local WIFI module. It shows as follow.

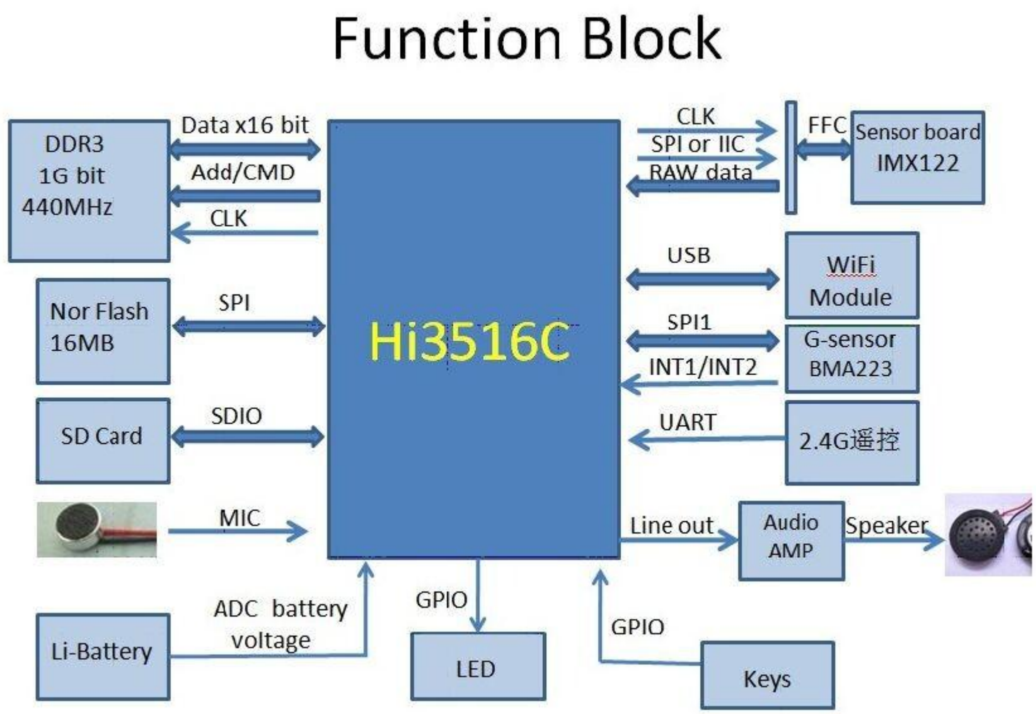

Figure 3. Hardware Block Diagram

2.3 Software Service Platform Design

1. Software hierarchy

Using the Linux operating system, Linux operating system completely free, compatible with POSIX standards, good interface, support for multiple platforms and almost all the standard network protocol, as well as the embedded TCP/IP network protocol stack. Software hierarchy as shown in the Figure below.

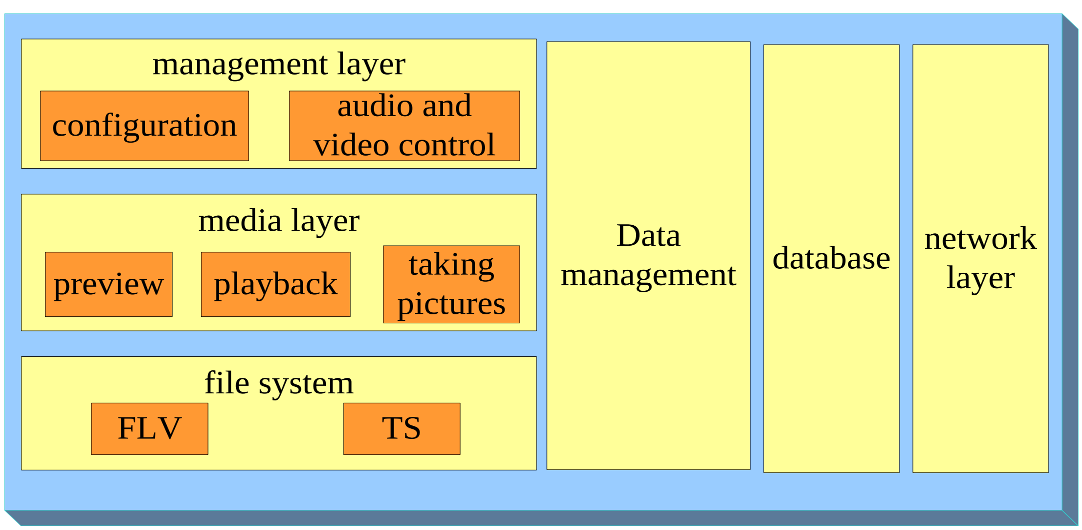

Figure 4. Video Server Software Hierarchy

2. Cross-compilation environment

Cross-compilation environment, simply said is that on a platform to generate another platforms executable code, it commonly comprising integrated development environment by compiler, connector and interpreter. Cross compile tools are mainly binutils gcc and glibc, *etc.*, Sometimes for the sake of reduce the size of the libc library, you can also use other c library instead of glibc, such as uClibc, dietlibc and newlib.

Development environment: The host PC running Red Hat9.0, target plate uses Hi3516 (ARM9) embedded processors. To establish an ARM cross-compilation environment for the main process:

(1)Before Set up the Shared directory, copy haisi SDK to the shared directory.

(2)Copy haisi SDK to working directory: cp Hi3516_SDK_V1.0.7.1.tgz /work/share/platform/.

(3)Use the command: tar xzvf Hi3531_SDK_V1.0.7.1.tgz. Unzip the file.

(4)Enter the file: cd Hi3531_SDK_V1.0.7.1 Perform an installation package command ./sdk.unpack.

(5)Install the cross compiler

Enter the installation directory: cd arm-hisiv100nptl-linux/

Modified to executable file: chmod +x cross.install

Run the setup command: ./cross.install

(6)Perform source /etc/profile, Installation of cross compiler configuration script environment variable can take effect, or please login again.

(7)Verify the installation is successful or not

Write a simple test. C file

Use the following command to compile arm-hisiv100nptl-linux-gcc test.c -o kk

If you can compile, installation is successful.

3. Video on Demand System Design

Real-time video on demand function is that the real-time video preview, playback, pause, stop, etc. It is based on streaming media transport protocol. Streaming media refers to "flow" in the network transmission to transmit audio, video and other multimedia files. Streaming transmission way is wait for video and audio multimedia files compressed into a zip file, by the server to the client continuously in real time transfer. Mainly includes three parts: coding part, network transmission parts and decoding part.

3.1 Video Coding

Coding is based on a certain protocol or format converts analog information to the bit stream. Coding can not only guarantee the quality of the original audio and video file, but also to compress the original file to facilitate data transmission. This system adopts the H. 264 compression technology; H.264 has a low bit rate, high quality images, strong fault tolerance, strong adaptability of network and other significant advantages.

Specific coding design is as follows:

(1)Initialize

H.264EncoderInit ();

This function is mainly to do the following work:

Obtain configuration file information: CHAR* pcCfgFile = "H.264Cfg.txt";

Open the h. 264 encoder: m_stH264 = H264Open(pcCfgFile)；

Assign each frame data encoded cache: m_pEncoded = H264Malloc(m_lDstSize)；

(2)Compression coding

INT iResult = H264Encode(m_stH264, pOriginalData, m_pEncoded, m_lDstSize)；

3.2 Network Transmission

Network transmission is based on streaming media transmission protocol RTP/RTCP, RTSP, *etc.*, so as to realize information interaction between encoding and decoding terminal. In essence is the session between server and client. This system adopts RTSP (real time streaming protocol) to interact information between server and client.

RTSP message interaction process:
C means client, S means server
First step: Query the server available methods
C->S:OPTION request
S->C:OPTION response
Second step: The media description information
C->S:DESCRIBE request
S->C:DESCRIBE response
Third step: Establish a RTSP session
C->S:SETUP request
S->C:SETUP response
Fourth step: Request to transfer starting
C->S:PLAY request
S->C:PLAY response
Fifth step: Data broadcast
S->C: Send streaming data
Sixth step: Close the session, exit.
C->S:TEARDOWN request
S->C:TEARDOWN response

1）Delay Problems and Solutions in the Process of Network Transmission

RTSP is launched, put an end to the streaming media. While RTP is streaming data, RTCP and RTP, cooperate with each other, RTCP control RTP media stream. RTCP itself does not transmit data, but collaborates with RTP, package and send multimedia data. Provide feedback for quality of service of RTP. In the process of information transmission of the actual drive recorder system, will inevitably encounter delays, the so-called time delay, that is between driving in the process of real-time traffic and real-time video player has time difference. It will affect the authenticity of driving records. Especially in the event of traffic accident, can't record accurate real-time record of the accident. There are many factors that can lead to time delay, such as the choice of transport protocol is not suitable, cache frame set value is not reasonable, *etc.,* In view of the defects such as caused by the delay in the process of transmission. This system provides the solving strategies to reduce delay, to make the play more smoothly. RTSP and RTP diagram is as shown in Figure 5.

Time delay problem solution: The first is the choice of transport protocol, the transport layer has two joint agreements UDP and TCP, the speed of TCP protocol transmission is slow, it has lost packet retransmission mechanism, and it is ordered. The speed of UDP protocol transmission is fast may be lost package, and it is disorderly. The protocol allows a range of error. As a result of the video frame is continuous, there is strong correlation between the frame and the frame, so on the transmission adopts the UDP protocol can meet the demand. Use the UDP protocol will be encoded data sent to the client, the process is as follows:

(1)Use the UDP protocol to send encoded data to the client:

Sendto(socketfd, pBuf, ullen, 0, (struct sockaddr *)destAddr, sizeof(struct sockaddr));

(2)The client receives the encoded data:

Recvfrom (slSocket, pBuf, ulTotal, 0, (struct sockaddr *)&addrRemote, &slFromlen);

The second, after select the protocol, thread processing data reception and decoding, at the same time to set the cache frame below 3 frames. If this time appear the phenomenon of the decoding slowly, in order to ensure the real-time video transmission, will be considered lost frames, but reference frame can't lose. In the whole process of receiving, decoding and drawing, to reduce the number of replication can also be effective in reducing latency.

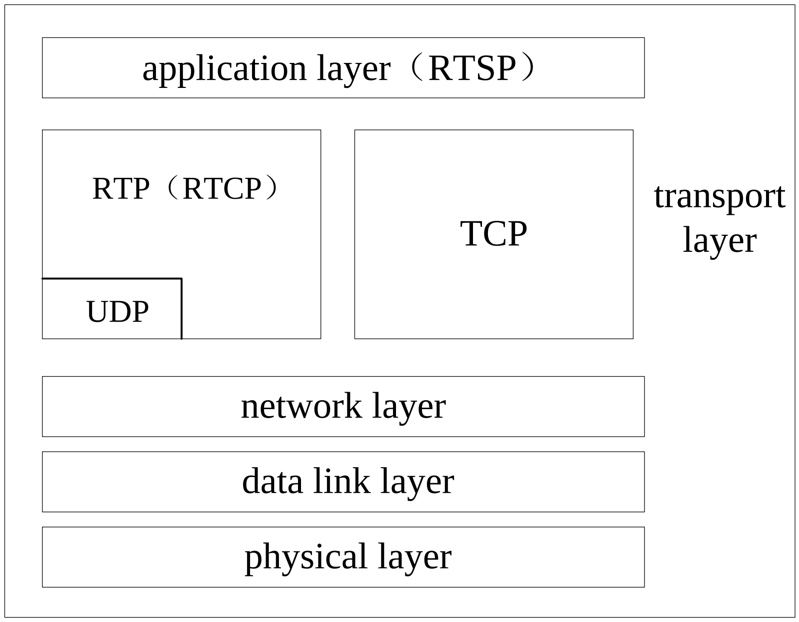

Figure 5. RTSP and RTP Diagram

2) Audio and Video Synchronization Issues and Solutions

Play multimedia files has a high requirement on the synchronization. But in practice, it will appear audio and video data are not synchronized. Audio and video have time lag, it was a serious impact on the quality of play, so to solve this problem is imperative.

There are two kinds of solution for the audio and video synchronization, the one is the sender to solve, and the other is receiver to solve. In the audio and video package has a timestamp. In order to achieve complete synchronization of audio and video, after the arrival of audio and video then play in the corresponding time. In this way to realize audio and video synchronously .The sender need to keep sending the corresponding audio data and video data synchronously, and the receiving end need to cache data, determine the synchronous data arrival began to play, otherwise in the buffer, receiving end according to the timestamp of received audio data and video data and then Arrange to play.

Audio collection has time mechanism. If the time required for sampling data is 1s, the audio can be played at their own speed; the number of frames per second is fixed. If the speed of capture is 30pfs, 1/30 s refresh a frame images by calculation, this way can ensure the audio and video collection is synchronized.

After the sender capture the audio and video data, based on RTP protocol, the time stamp on the data, by UDP (user datagram protocol) sent out. For receiving audio data send to sound card, while at the same time to send time of audio packets timestamp to video package. And then compare video package timestamp with audio package timestamp. If not reach the reference time, decode to play. If reach the reference time, video appear lag phenomenon, Will need to discard the frame, to receive the next frame video, move in circles, until the end of the command.

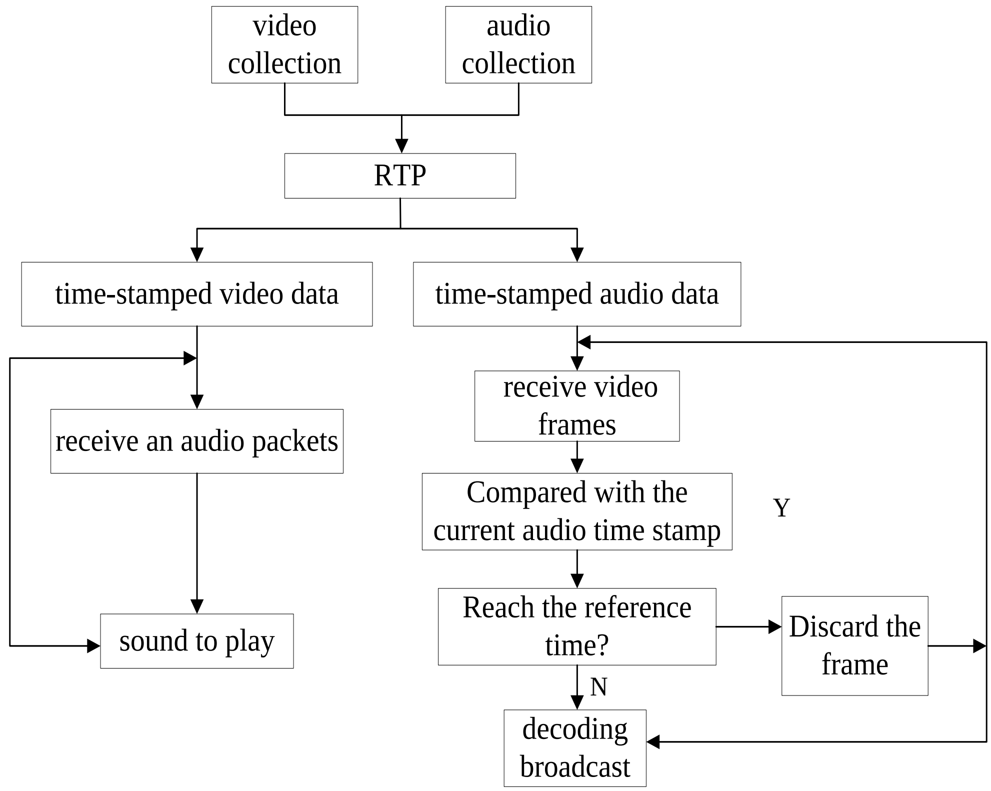

Figure 6. The Flow Chart of Audio and Video Data Synchronization

3.3 Decoding

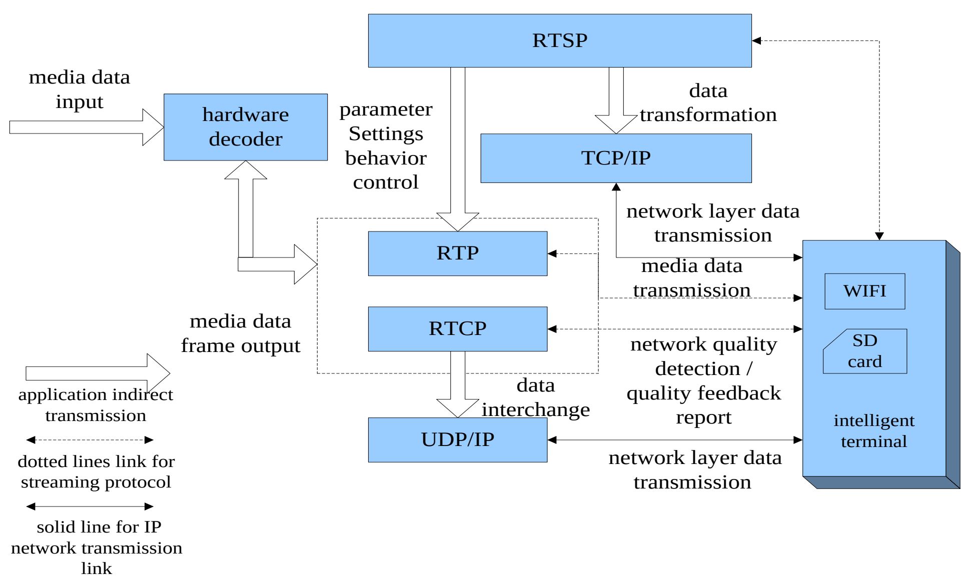

Figure 7. Streaming Protocol Stack

Used for receiving of audio and video stream, received the compression file after H.264 coding, if you want to read the file, you need to decode, through the player can be realized , and then you can preview real-time video or playback video, and so on.

When the drive recorder powered on, camera captures the light signal, after collection procedures to stay digital video data of stay coding, then after H.264 coding of Hi3516 chip and data package of RTP protocol. At last, through the network transmission to the client. The client will receive RTP packets parsing decoding and playing.

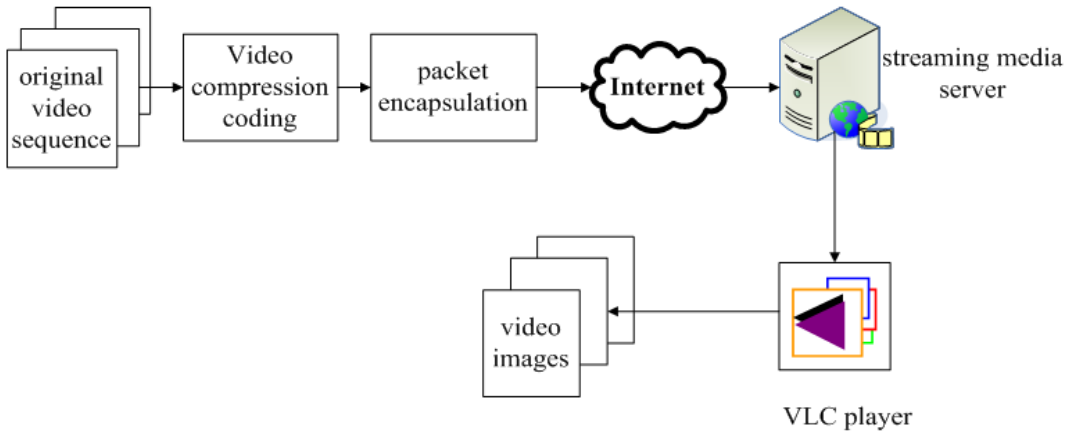

Figure 8. Data Flow Diagram

Preview: When the client send preview request to server, Server requests to establish response through RTSP streaming, and when the session is successful, can enter the preview mode.

Playback：Playback process is the compressed file which stored in the memory card decompression process. Then read and watch.

Taking pictures: If you want to record the key events, you can press the external wireless remote control button, the wireless remote control using 2.4G module communicate with drive recorder. Key events of images and video in 5s before and after the events will be automatically stored in the smart phone terminal.

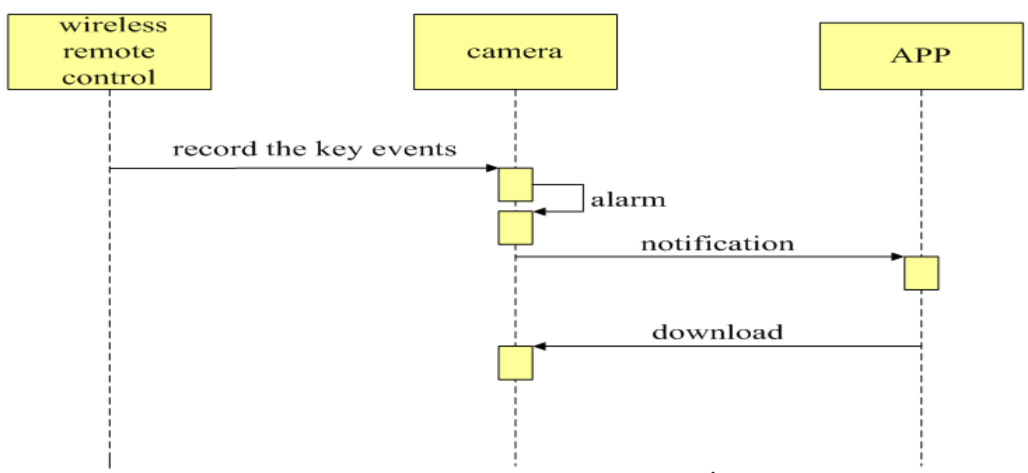

Figure 9. Key Event Log Process

In the driving record system, Set the preview resolution is 854 * 854 pixels; there are three kinds of the resolution of the playback, 1080p, 720p and 960p. In the process of switch video preview and playback, it will meet crash. Once appear crash, need to restart the APP. The situation of real-time video viewing will cause interference, therefore, to guarantee the resolution switch without a crash is necessary. Switch the resolution, there are two kinds of situations, intelligent terminal APP actively switch and camera actively switch.

In the first case of the solution, before APP actively switch, firstly stop the decoder, then through the HTTP protocol to send commands to the camera business processing layer, the camera receives the command line immediately empty the original data cache. Then another resolution of data sent to the RTSP cache. After APP waiting for a period of time then open the decoder, get data from RTSP.

In the second case of the solution, if camera actively switch resolution, need to stop the current resolution code flow, no longer send RTSP stream, then through HTTP protocol notice APP that need to switch the code flow, close the current decoder after APP received the notice, through HTTP protocol send request to camera of other resolution. After receive request, send the new resolution into the RTSP cache, APP waiting for a period of time then open the decoder, get data from RTSP.

The above scheme can guarantee not crash when switching resolution.

4. System Operation and Software Testing

Through the front on the driving recorder design of the whole system, including hardware design and software design. This chapter will test the whole function of video surveillance system. This system camera adopts SONY sensor，Support the resolution of 1080 p. Client is intelligent terminal use apple mobile phones, the server is the driving recorder, Software testing interface as follows.

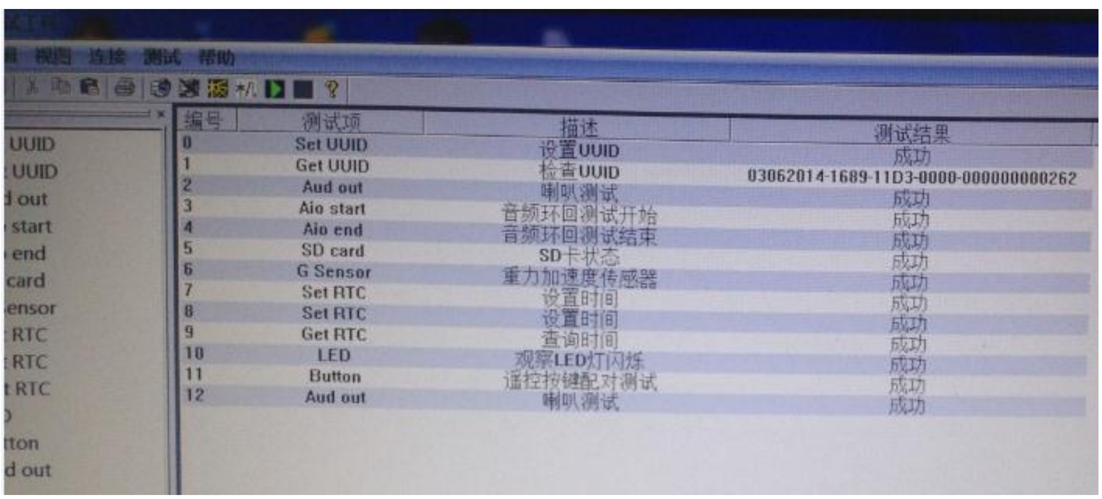

Figure 10. Software Testing Interface and Results

Intelligent terminal APP interface:

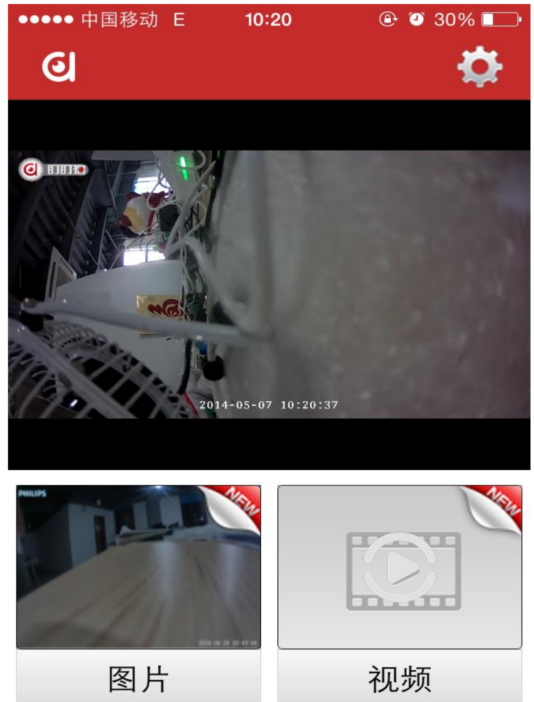

Figure 11. Equipment not Connected

Figure 12. Equipment connection is successful

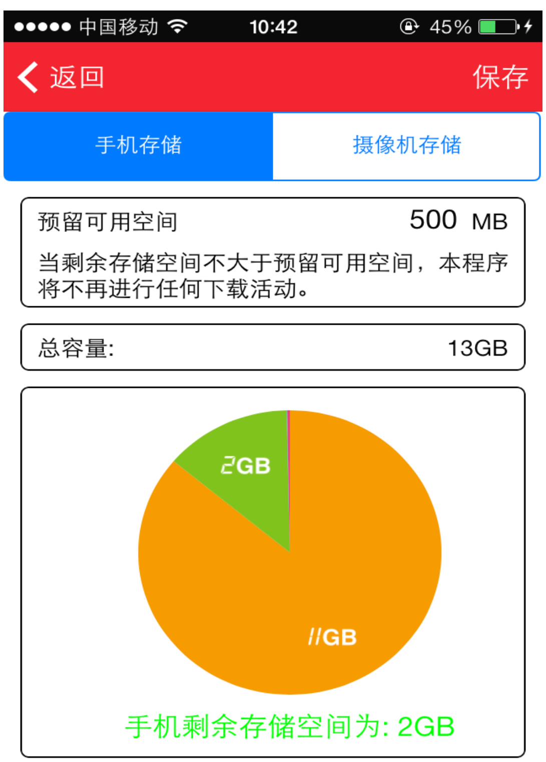

Figure 13 APP store interface **Figure 14 APP Settings interface**

In the process of actual driving scene：

Figure 15. Haze Weather Driving Record Scene

In the process of the actual walking scene:

Figure 16. Sea Scene

This chapter tested the whole function of video surveillance system, confirms the feasibility of video surveillance system of intelligent terminal platform.

5. Conclusion

This system was designed and implemented the intelligent terminal for the development of driving recorder, and producted, Compared with the previous driving recorder. Intelligent terminal drive recorder cost low and easy to promote. In the intelligent terminal APP, realize preview or playback of data. It also can record key event through wireless remote control. The recorder SD card can directly communicate with the computer without a peripheral interface, practical and convenient, compare with the traditional drive recorder, is very different. Therefore it will have the very good development prospect.

References

[1] F. Matusek, S. Sutor, F. Kruse, K. Kraus and R. Reda, "NIVSS: A Nearly Indestrucrible Video Surveillance System", The Third International Conference on Internet Monitoring and Protection, Bucharest, (**2008**) July, pp. 98-102.

[2] Y. Li, D. Han and J. Yan, "Design and Implementation of a Wireless Video Surveillanee System Based on ARM", 3rd International Conference on Digital Image Processing, (**2011**).

[3] J. Dudak, S. Pavlikova, G. Gaspar and M. Kebisek, "Application of open source software on arm platform for data collection and processing", 14th International Conference on Mechatronics, MECHATRONIKA, **(2011),** pp.77.

[4] J. Nightigale,W. Qi and C. Grecos, "Optimised transmission of H.264 scalable video streams over multiple paths in mobile networks", IEEE Transactions on Consumer Electronics, vol. 56, no. 4, **(2010),** pp. 2161-2169.

[5] W. Wenjie, C. Hyunseok, A. Goodman, *et al.*, "Managing Digital Rights for P2P Live Broadcast and Recording on the Internet", IEEE Transactions on Multimedia, vol. 14, no. 6, pp. 1538-1545.

[6] Z. Zhen and C. Ruchun, "Remote Intelligent Monitoring System Based on Embedded Internet Technology" IEEE International Conference on Automation and Logistics, **(2007),** pp. 2665-2669.

[7] K. Seajin, L. Byungjin, J. Jaewon, *et al.*, "Multi-object tracking coprocessor for multi-channel embedded DVR systems", IEEE Transactions on Consumer Electronics, vol. 58, no.4 **(2012)**, pp. 1366-1374.

[8] X. Haibo, L. Quanhu, C. Jialong, *et al.*, "The research of real-time video transmission based on 3G network in Linux", 9th International Conference on Fuzzy Systems and Knowledge Discovery(FSKD), (**2012**), pp. 2274-2277.