



# Effective Transfer Function Approach for Decentralized Controller

M.Bharathi

Asst. Professor, Department of E&I, Bharath University, Chennai, India

**ABSTRACT:** Effective transfer function approach is a novel method for decentralized control system design of multivariable interactive processes. An extension of the effective transfer function approach by taking into consideration all the interactions is proposed. The simplicity and effectiveness of the method is based on the incorporation of the interaction frequency directly in the controller design. This approach ensures that all the necessary information of the gain and interaction frequency changes are provided. The advantage of this method is more significant when applied to higher dimensional processes with complicated interaction modes. Since this is an extension of the ETF approach, it can also be easily integrated into an auto-tuning control structure.

**KEY WORDS:** Multivariable system, Effective transfer function, decentralized control.

## I. INTRODUCTION

The video monitoring system is a integrated system with strong guard ability. Depending on its intuitive, convenient, rich information the video monitoring system widely applied in many kinds of situations. In recent years, with the rapid development of computer, network and the imagery processing, lots of embedded video monitoring system emerged. The scene monitoring has become the hot spot of research. What this article designs is one kind of embedded system which can gather image and the processing rapidly. According to the characteristic of the system, such as small size, low power consumption, quick speed and so on, we design one kind of embedded image acquisition system with strong versatility.

## II. SYSTEM OVERALL DESIGN

This system adopts the embedded system technology MPEG-4 image coding technology as well as the network Transmission technology. The system uses the USB interface Camera to capture video under the embedded Linux System Platform which based on S3C2440 micro controls chip. Transport the gathered data to the development board by USB Interface and carries on network transmission, then display it through PC after complete the MPEG-4 compressed encoding processing, thus realizes the video monitoring function. Video capture system design as shown in Fig.1

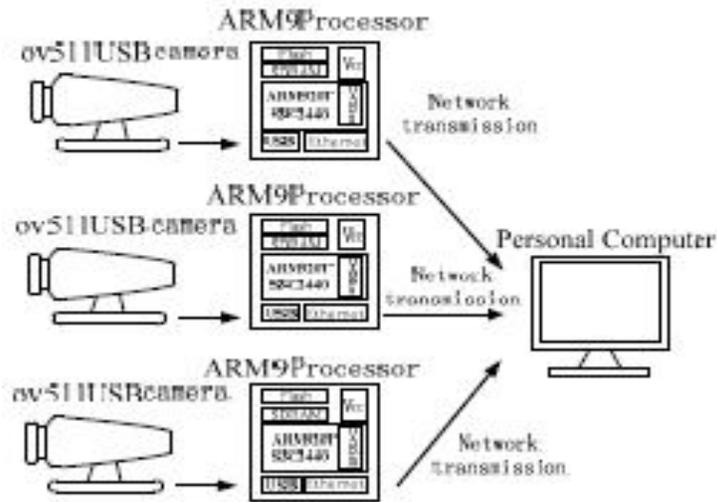


Fig-1: Structure of video Capturing System

### III. DESIGN OF HARDWARE FRAMEWORK

The S3C2440 that has been adopted in this system is a 16/32 RISC (Reduced Instruction Set Computer) embedded microprocessor based on ARM920T kernel. The chip includes full performance MMU (Memory Management Unit), the 128M flash memory which is used for solidifying operating system, graphical user interface and imagery processing algorithm and so on. 32M×2 SDRAM (Synchronous Dynamic Random Access Memory) is used to running the system program and user program.

RS-232 is used to develop Linux host machine and the Ethernet is used to system's network transmission. S3C2440 mainly oriented to the handhold equipment as well as the application of high cost performance and low power loss. At present it's generally used in micro controllers of the multimedia terminals. The image input module of the system selects the web eye V2000 product, the lens use the CMOS (Complementary Metal Oxide Semiconductor) photosensitive part, it has quick response, and lower cost compared to CCD (Charge-coupled Device), digital signal processing chip (DSP) was ov511, the Linux kernel usually brings the driver of this model of chips so as to transplant facilitate. This gross structure diagram of the hardware platform as shown in Fig.2.

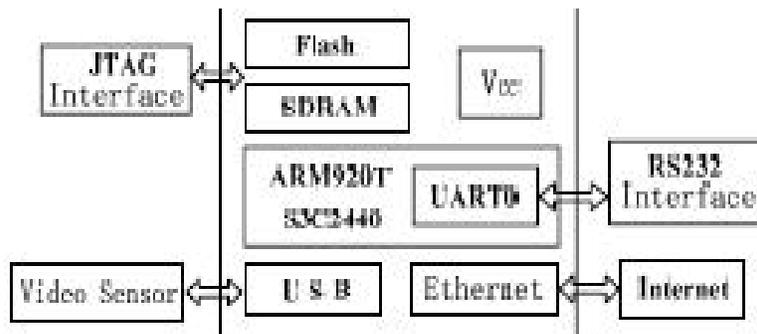


Fig-2: Structure of Hardware platform

## IV. SYSTEM SOFTWARE DESIGN

### a. Transformation of Linux Operating System

The transplant of Linux operating system is related with the hardware. Its essence is making the essential modification to the Linux operating system according to the concrete hardware platform to make it running on this hardware platform very well. The system's kernel edition is 2.6.12. The Linux operating system's transplant needs to complete three works: boot loader transplant, Linux kernel transplant and filing system transplant. Boot loader is running before the operating system kernel, the mainly role is initializing hardware equipment(including I/O, the special function register), establishing the memory space map and bringing the environment of the system's hardware and software to an appropriate state. The Linux operating system's kernel can provide good support to the ARM processor and manage most of components which connect to the periphery of the processor.

### b. Working with USB Video Capture Driver

Video4Linux is video equipment's kernel driver which is under embedded Linux. It provides a series of interface functions for the programming application of video Equipment under Linux. The driver of USB interface camera needs to provide the realizations of basic I/O operational functions, interrupting handling, memory mapping function and IOCTL functions of I/O channels, control interface, and define them in the struct file operations. Thus when the application programme carry on system calls such as open and close operations, the Linux kernel will visit the function which is provided by the driver through the struct file operation.

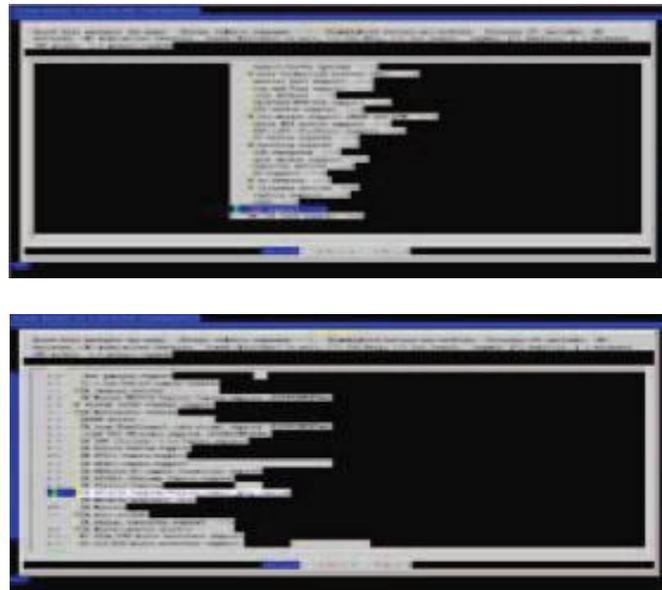


Fig-3. Loading of USB driver module

C .USB driver module to load:



In order to drive the USB Camera in the system platform, first we compile the USB Controller driver module into kernel dynamically to make The platform support the USB interface. Then input compile Command “make menuconfig” in the list of kernel which is Under Linux, choose “Video for Linux” which under “Multimedia device--->”, load the video4linux module. As shown in Fig.3, press “Enter” to enter “usb support--->” and enter “<\*>support for usb” which is under the list again, then choose “usb camera ov511 support”, thus in the kernel add Driver support for USB camera which adopt the ov511 Connection chip, save the configuration and exit. Make dep; make image; make modules then generate the driver which involve ov511 under /driver/usb, at the same time the creating zImage will be placed in the /tftpboot. Finally, start board with the new kernel and load it successfully by insmod ov511.o. Write the kernel to flash by DNW after successfully testing it. Insert the camera, the system will prompt discovering the ov511 camera, it indicates that the driver is loaded Successfully.

#### **D. The control word and the structure provided by V4L:**

It provides many data structure and control command in Video4linux *Standard*; the programme controls the device By calling IOCTL Function to finish the task of video collection. Major Device’s control command in the function and data Construction application supported by Video4linux

a) **VIDIOCGCAP:** VIDIOCGCAP is used for obtain video device information; the gained information is stored in data construction video capability.

b) **VIDIOCSWIN and VIDIOCGWIN:** VIDIOCSWIN and VIDIOCGWIN are used for set and inquiry the display window information, these information is stored in video window.

c) **VIDIOCGCAPTURE & IDIOCSCAPTURE:**

VIDIOCGCAPTURE and VIDIOCSCAPTURE are used for Sub-zone gathering information inquiring and setting. This Information is stored in video capture.

d) **VIDIOCGPICT & VIDIOCSPICT:** VIDIOCGPICT, VIDIOCSPICT are used for inquiring and setting image Attribute.

e) **VIDIOCMCAPTURE:** VIDIOCMCAPTURE is used for obtain video image.

f) **VIDIOSYNC:** VIDIOSYNC is used for judge whether Video image intercepts successfully.

g) **Video\_channel:** The video\_channel is about various Signal source attributes.

h) **Video\_window:** The video\_window contains Information about capture area.

I) **Video\_mbuf:** The video\_mbuf is the frame Information mapped by mmap.

j) **video\_mmap:** The video\_mmap is used in mmap.

#### **E. Video capture process based on V4L:**

The flow chart of video capture based on V4L as shown in Fig.4, the corresponding device files of cameras in the system is /dev/video0, open video device with call function “vd->fd =open(“/dev/video0”,O\_RDWR)”,and function“IOCTL(vd->fd,VIDIOCGCAP&(vd->capability))” to read the corresponding information of camera in video\_capability. Information is copied from the kernel space to member variables of user program space vd->capability, use the function “IOCTL(vd->fd, VIDIOCGPICT, &(vd->picture))” to read video picture information in image buffer. After get video picture information, if it is need to set the information, we can first assign a new value for each component, then call “ioctl(vd->fd, VIDIOCSPICT, &(vd->picture))” to reset these information. after the initialization of the above equipment, we can use V4L to video intercept. This system intercept video data by mean of memory mapping through mmap(), a period of memory should be mapped for buffer through mmap interface, the driver carry on image gathering through the VIDIOCMCAPTURE control word after mapping device files into the memory. The mmap() system call makes processes realize sharing memory by mapping the same ordinary file. The same block of physical memory is mapped to different processes address space. Different processes can see data updates in sharing memory between each other. Process can visit files as common as memory access rather than calling files operation function, so the access speed of using mmap() memory mapping is much faster than standard file I/O when dealing with large files.

First use the function “IOCTL(vd->fd, VIDIOCGMBUF, &(vd->mbuf))” to obtain frame information which in camera storage buffer, then modify video\_mmap and current setting of frame state. Bind mmap with video\_mbuf, thus map the

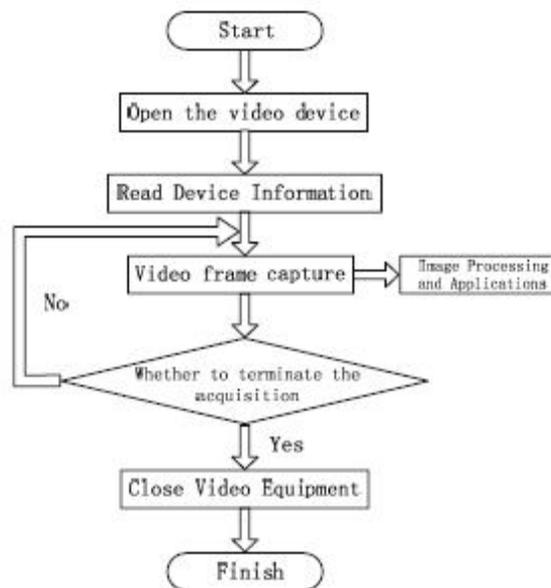
device files which is corresponded to the video equipment into memory by using the function `vd->map=mmap(0, vd->mbuf.size, PROT_READ|PROT_WRITE, MAP_SHARED, vd->fd, 0)`”.

The function “`IOctl(vd->fd, VIDIOCMCAPTURE, &(vd-> mmap))`” to finish a frame data intercepting. We can judge whether the video current frame capture was finished by function “`IOctl(vd->fd, VIDIOCSYNC, &frame)`”.

In order to obtain continuous frame video images, it is necessary to set the number of `vd-> mmap.frames` so as to determine cycle times of capturing frame data. Use the sentence of “`vd->map+vd->mbuf.offsets[vd->frame]`” to obtain frame address, each captured frame data is stored in the style of documents, then copy the captured data to compressed buffer in MPEG-4, the capture will stop when the captured frame number achieves the set value of `vd-> mmap.frames`.

## F. Video Information Compression and Network Transmission

Video data must undergo the pre-treatment before carrying on transmission on the Ethernet due to network bandwidth limits. This system uses a new generation of MPEG-4 which based on objected coding standards; it has the huge superiority in interactive, anti-code nature and the highly effective compression. XVID is the newest MPEG-4 codec, and is also the MPEG-4 encoder which is published by GPL agreement and the first real opening source. In order to make the embedded system has MPEG-4 decoding function, we must transplanted the software codec into embedded system first, then write specific application program according to function it provided to realize video coding functions. Here we choose version of `xvidcore1.1.2`, the specific method of realization is as follows:



**Figure.4 Video capture process**

## G. Configure the kernel:

Configure the kernel of XVID through the following command: `[root@localhostgeneric]#./`

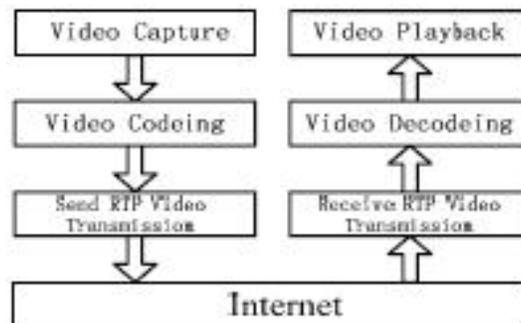
configure --prefix=/home/123 CC=arm-Linux-gcc -host=arm-Linux .The configuration is mainly platform system, host and target system, the compiler type, some default file format, the necessary header files.

#### H. Compile the source code:

Compile the source code, Make some general source file generate target file, then pack Dynamic link storehouses according to these object files. The Dynamic link storehouse is some compiled code pieces, when needs its program ran, lib/ld.so will load it dynamically. The dynamic link storehouse saves the floppy disk and the memory Space and the compiling time.

#### I. Write application programme:

Copy the storehouse Document libxvidcore.so which is produced by cross Compiling to libs subdirectory of cross compiler working Directory. This library files provides programming interface for other module of the system. Write our own application Programme with the actual need The system transmits video data in real time by using RTP (Real-time Transport Protocal)/RTCP (Real-time Transport Control Protocal) protocol made by IETF sound video work team. Compressed encoding video data is processed by group module of RTP, it packs the RTP package by adding the masthead and transmits by the transmission module. In order to satisfy the latency and the drop package request of the network video data live transmission, we need cooperate RTP and RTCP to providing data real-time transmission, which contain statistical information like the number of sending package and lost package, the server can change the transmission rate dynamically according to the statistical information.



**Figure.5 Data transmission Process**

Now there has been lots of open source libraries provide the realization of network protocol RTP/RTCP function. The Jrtp lib is an object-oriented RTP library which follows the RFC1889 design completely, it's a RTP storehouse which Realizes by the C language. In order to realize network Transmission of the video data by jrtp lib, we must first configurate and compie jrtp lib source to generate the jrtp lib library of embedded environment. After producing the static storehouse, we can write application programme by calling Static libraries of jrtp lib to realize coding image transmission.

Program is divided into the sender and receiver, the video Collection part of sender collects video information and sends Data to encoder for MPEG-4 coding, then adds on the RTP Masthead before the coded VOP (Video Object Plane) data, Realizes the network transmission by jrtp lib network transmission module. The receiver recives RTP video data, Removes the RTP masthead and sends it to MPEG-4 decoder for decoding again.



System uses the MPEG-4 compression algorithm so it has high compression ratio, the general bandwidth transmission Speed is 8 frames/s in general bandwidth, and the video displayed by the system is smooth and stable. Video collection system display effect as shown in Fig.6. In the follow-up work, if we use MPEG-4 video encoding chip, the system can achieve better video monitoring results

## V.CONCLUSION

Video collection system chooses S3C2440 32-bit Embedded micro-controller chip with higher frequency and V2000 camera based on ov511, combines V4L video interface Technology and MPEG-4 video coding and decoding Technology and streaming video transmission technology, it Can realized the rapid video acquisition and real-time transmission well. This video collection system has stable performance and lower cost. In the follow-up work, this system still needs improving in video capture and V4L standards, if necessary, it can add the image processing algorithms. This article has certain research value in the video Image's application aspect.

## REFERENCES

- [1] Huiming Liu, & Rulin Wang. Development Philosophy and Practice of the Video monitoring system[J]. Intelligent Building. 2004, 3
- [2] Weihua Ma. Embedded Systems Principles and Applications[M]. Beijing University of Posts and Telecommunications Press, 2006
- [3] Tianze Sun, Wenju Yuan and so on. Embedded Design and Linux Driver Develop Guidance[M]. Beijing:Electronic Industry Press, 2005.
- [4] Alan Cox. Video4Linux Programming [EB / OL]. <http://kernelbook.Sourceforge.Net>,2000
- [5] Yuzhuo Zhong, Qi Wang, Yuwen He, Object-based Multimedia Data Compression and Coading International Standard MPEG-4 and Calibration Model [M]. Beijing: Science Press, 2000
- [6] XVID core API overview. <http://www.xvid.org>
- [7] Jori Liesenborgs.JRTP LIB 3.5.2.March 26, 2006