ISSN:0975-9646

M.Gurunadha Babu et al, / (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 4 (5) , 2013, 664-667

Driver Development for CMOS Camera on ARM11 using Embedded Linux

M.Gurunadha Babu*, M.Venugopal#, M.Phaniraj Kiran**

**Professor & HOD,ECE dept.*
Chilkur Balaji Institute of Technology,Hyderabad

***Consultant*

#Managing Director R&D,
Emebedded software labs,Hyderabad,India

Abstract— **With the advent of VLSI/CMOS technology, using CMOS technology video acquisition is emphasised. Presently CMOS camera chip is not supported by the latest Linux kernel available in the market. In this paper, designing of CMOS camera (OV9650) driver on S3C6410 developing board is considered using embedded Linux environment and design of remote monitoring system. Serial camera control bus (SCCB) is a distinguishing feature of OV series CMOS chips. This paper emphasise on Serial camera control bus (SCCB) on ARM11 embedded platform. S3C6410 provides a camera interface, and the camera driver is designed based on it using embedded Linux platform. The system has the functions of video signal acquisition, compression, transmission over the internet for remote video monitoring. The advantage of using the Linux is that the kernel size is small. The outputs and the result show that the driver works well for video viewing and response is proper in remote operating mode. In our experiment the video quality is good. The driver can also be ported on to any other embedded boards.**

Keywords— **camera device driver --- OV9650, SCCB, S3C6410, Image capture.**

I. INTRODUCTION

CMOS image sensors uses CMOS technology [2] which consists of Image sensor core to capture the image information, single clock for timing the data transfer for read and write operation between the CMOS camera and the Microprocessor, sequential logic and programmable functions to support the data transfer functions, and A/D converter to convert the image data in analogue form into the digital form for processing. The image sensor core comes with the pixel array in form of grid and the above peripheral in a chip can be used to support CMOS camera interface. The advantage of using CMOS image sensors is small size, light weight, power consumption is low. At present the CMOS image sensors [3] are widely used for general purpose image capturing applications. The camera interface facility is provided by the ARM11 Microprocessor [4] which is a RISC machine. S3C6410 is provided with camera interface and application development.

Image retrieving is done using embedded operating systems. Embedded Linux is one of the OS which has features like its open source. Embedded Linux [4] is widely used in the embedded field. The advantage of using Linux is its small kernel size. The porting procedure of Embedded Linux on to the Microprocessor includes the development of cross compile environment, the compilation of boot loader, porting of Linux kernel and the construction of root file system. In this paper, the method of designing the Omni CMOS Camera driver [1] based on S3C6410 developing board with the embedded Linux environment is introduced. With the advent of multimedia technology, Video compression technology, surveillance of Video information has improved. Video monitoring through internet is used in various fields for example video conferencing. In this paper video signal acquisition, compression, transmission, remote video monitoring via Internet is implemented.

II. LITERATURE SURVEY

Mr. S. M. Gramopadhye, Prof. R. T. Patil, Mr. A. N. Magdum, Mr. R. A. Chaugule discussed in their paper the transplantation of the Linux operating system as well as implementation of CMOS device driver based on the mini2440 development board. The author explained that, the transplantation method of Embedded Linux includes the development of cross compile environment, the compilation of boot loader, porting of Linux kernel and the construction of root file system. Min Zhang ,Jin-guang Sun ,Shi Wang, et all has done Research and Implementation of the CMOS Camera Device Driver Based on S3C2440. Yuan Weiqi, Tang Yonghua discussed in their paper about SCCB (Serial Camera Control Bus) bus protocol to achieve the image sensor OmniVision series initialization and ensure their normal working hypothesis. Based on the combination of SCCB bus protocol the author described in detail on the basis of the DSP through HPI (Host Interface) programming SCCB bus protocol ideas and methods. The Functional specification by Omini vision technologies explains Serial Camera Control Bus(SCCB). Version:2.1, 2003,21042

III. HARDWARE

All The RISC ARM Microprocessor S3C6510 is interfaced to the OV9650 camera. The CMOS camera is a colour image sensor with 1.3 mega pixels. The maximum frame rate in VGA format is 30 fps. The interface between the CMOS camera and ARM processor is shown in the figure 1. S3C6410 is a Samsung processor which uses 16 and 32 and the core is ARM920T. The different frequencies used are 300,400 and 533MHz. It has 64MB Flash, USB, SDRAM and LCD controller, camera interface. The CMOS camera support is given by cam-interface. The data transmission through this camera interface is of two types. DMA transfer where the image data sampling from the camera interface into RGB format and the transfer to SDRAM using DMA transfer. In the second case the image data is transmitted to the SDRAM in YCbCr in 4:2:0 or 4:2:2 format.

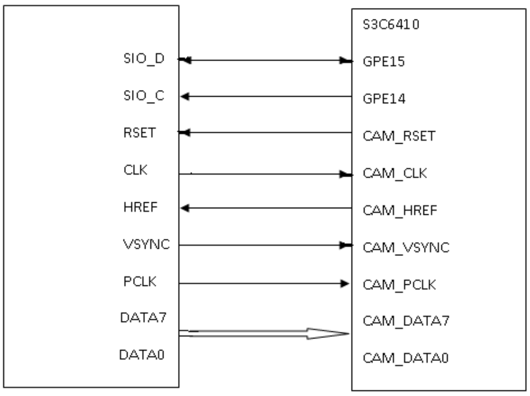


Figure 1. Interface between ARM and CMOS Camera

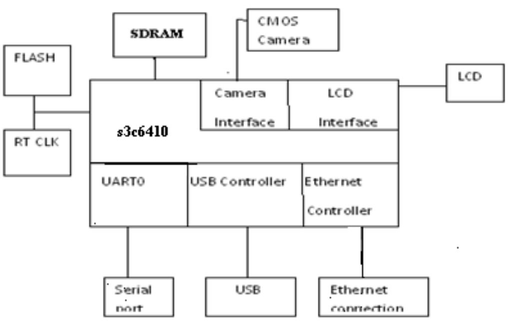


Figure 2: ARM11 Architecture

IV. Research on Device Drivers

In any operating systems, the functionality of device driver provides access for data structures, and Hardware. The Hardware details are shielded from the user for his application. Usually the OS treats everything in the form of a file. So Linux 2.6 version kernel treats the hardware also has a file. The user will be provided with the systems calls and the drivers are linked by file operations. The driver development for SCCB has to include initialization, device registration, and specification of APIs for an applications. The device driver frame work is shown in the figure. The S3C6410 does not provide sccb interface, so the driver has to include a sccb driver. The driver completes the device registration, initialization, API for applications. Device driver frame work is shown in Figure 3.

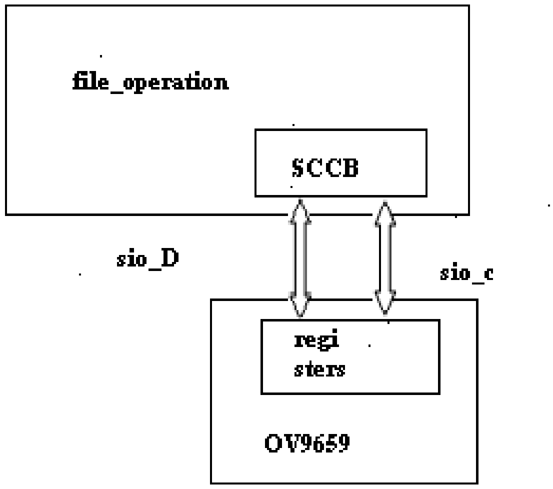


Figure 3: Driver frame work

V. Implementation of SCCB Driver

The serial camera control bus (SCCB) [3] is customized by the CMOS camera company. The control bus is used to read and write the registers with the control bits to control the output of the camera. The handshaking signals are used between the CMOS camera and the ARM processor to transmit the image data. Here we consider ARM processor has data terminal equipment (DTE) and the CMOS camera has data communication equipment (DCE). Initially the DTE sends the start signal, clock signal and the stop signal. Based on the interface the SCCB has address. This supports 100Kb/s or 400 Kb/s transmission speed. The address modes are ID address (8bit) and it is divided has read and write addresses. The MSB is active high and it is a chip select signal while LSB is active low and it is used for R/W control bit. The interface uses I/O mapped I/O technique and the read address is 0x61 and the write address is 0x60. The internal register unit address is to determine which internal register to operate. The SCCB do not have multi byte read and write support. The pin connecting with SIO_C is always set to output, while the pin connecting with SIO_D needs to dynamically change the input/output modes. In the SCCB bus protocol defines three operations i.e. the three-phase write operation, two-phase write operation and two-phase read operation. Three-phase write operation is used to write data into register. Two-phase write operation has only the first two steps of three-phase write operation. Two phase-read operation is used to read data in internal register has to include a three-phase write operation to provide the address of the register to operate.

Here, S3C6410 is the Data terminal equipment, OV9650 [4] is the Data communication equipment. The important function: in SCCB driver are sccb_write(u8 IdAddr, u8 SubAddr, u8 data) which is used to write data into internal register i.e. (control register) of OV9650 and u8 sccb_read(u8 IdAddr, u8 dceAddr) which is used to read data from internal register (control register) of OV9650.

TABLE 1. COMPARISON OF SCCB READ AND WRITE OPERATION

Operation	***Steps***
Thee phase write operation	Write the write address of the DCE (IDW,8Bit) and the "Dont care" * Write the register address (8Bit) and the "Dont care". * Write the data (8Bit) and the "Dont care".
Two-phase write	Write the write address of the DCE (IDW,8bit) and the "Don't care" *writer the register address (8bit) and the "Don'tcare"
Two-phase Read	Write the read address of the DCE (IDR,8bit) and the bit "Don't care" *read the data in register and NA

The code for sccb_read(u8 IdAddr, u8 dceAddr) is mentioned below.

```
u8 sccb_read(u8 IdAddr, u8 dceAddr)
{
u8 data;
down(&bus_lock);
sccb_start();
sccb_write_byte(IdAddr);
sccb_write_byte(SubAddr);
sccb_stop();
sccb_start();
sccb_write_byte(IdAddr|0x01);
data = sccb_read_byte();
sccb_stop();
up(&bus_lock);
return data;
}
```

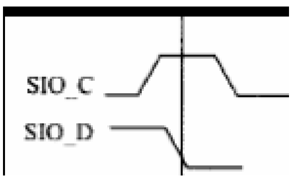
The timing diagram for sccb_read_byte() and sccb_write_byte() is to complete 8 bit read and write.

------- --

Status Timing

-------- --

Start



transmission

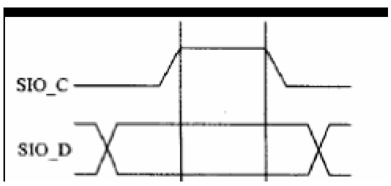


Figure 4: Timing diagram for read and write byte

Algorithm flow chart to reading 8 bit is shown in figure 5.

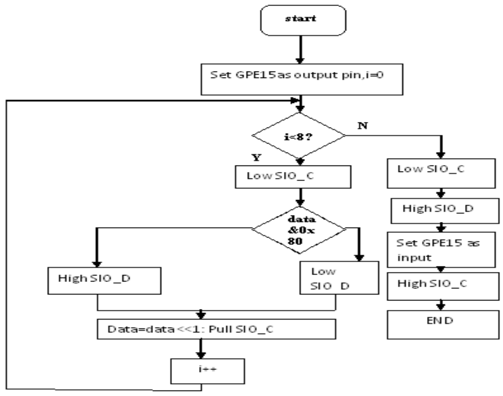


Figure 5: Algorithm flow chart for 8 bit read/write

VI. Implementation of the Driver Registration and Initialization

Here register the camera as a miscdevice. All the miscdevice share one major device number: 10. Using minor= MISC_DYNAMIC_MINOR to get a sub device number. All the miscdevices form a list. Calling the function misc_register() to add the device to the list, and registration is completed. Initialization includes:

1) Using s3c2410_ gpio_cfgpin to set GPIO to camera mode. Part code is as follows: s3c2410 _gpio_cfgpin(S3C2440_GPJ8,S3C2440_GPJ8_CAMPCLK);

2) Using request_mem_ region() to ask for memory resources for camera. Using ioremap-nocache () to map the memory resources to kernel space.

3) Initialize SCCB. Set SIC_C and SIC_D to high, which is the waiting status.

4) Initialize camera. Write the parameters (output format, AGC, sampling clock) into an array, and then using a cycle function to write parameters into the internal register of OV9650 through SCCB one by one.

Practical compilation instructions for camera port:

'#mkdir_install' #/configure -prefix=/opt/Embed/apps/SDL.1.2.13/_install -disable-video-nonx
-enable-video-qtopia –disable-video-photon –disable-video-photon –disable-video-direct –disable-video-ggi
-disable-video-svga –disable-video-aalib –disable-video-dummy –disable-video-dga –disable-arts –disable-esd
-disable-alsa –disable-video-x11 –disable-nasm –disable-degug –disable-joystick-amigaos –disable-joystick-beos –disable-joystick-bsd –disable-joystick-darwin –disable-joystick-dc –enable-joystick-linux –disable-joystick-macos –disable-joystick-mint –disable-joystick-win32 –disable-joystick –host=arm.linux –build=i386"
#/configure-help"

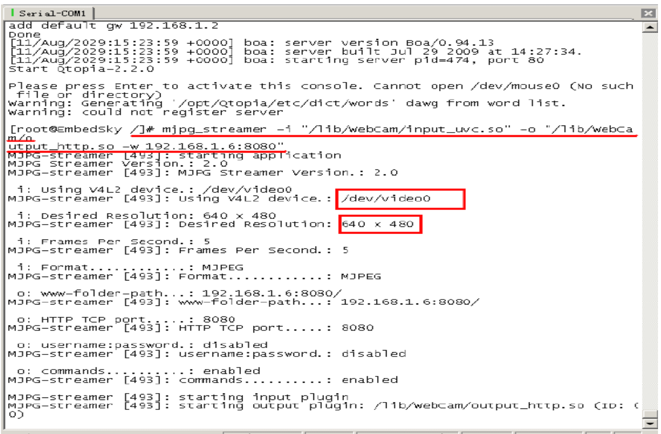
VII. Development Environment

Host computer: Founder PC
Ubuntu 9.04 Linux Operating System
nfs-kernel-server
Target board: ARM Development Board 6410
Embedded Linux-2.6.30.4
Cross compiler: arm-linux-gcc-4.4.3

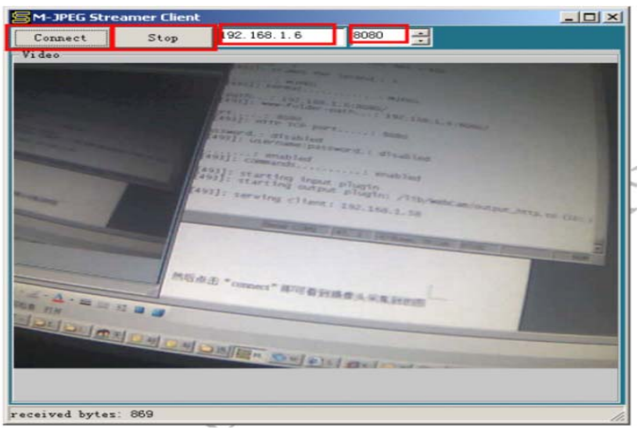
Menu configuration selection:

Device Driver →
Multimedia devices →
Multimedia core support
<*>Video for Linux
[*] Enable Video For Linux API 1 (DEPRECATED)
[*]Video capture adapters →
<*> OV9650 Drivers for EmbedSky SKY2440/TQ2440 Board
[*] V4LUSB devices →
--- V4L USB devices
<*> USB Video Class (UVC)
[*] UVC input events device support
<*> USB ZC0301[P] Image Processor and Control Chip support

Image captured information passed on to the system (streamed output):



On-board display from camera:



VIII. Conclusions

The CMOS camera driver design is developed on S3C6410 ARM11 processor. Embedded Linux environment is the platform for the driver development. SCCB features is being used for camera CMOS chip. The SCCB is realized using ARM11 development board. The output shows the driver is successfully ported and the captured camera output is displayed on the LCD display. The quality of the picture is as per the expectations based on the resolution of the camera. With the advent of VLSI integration, onto a single SOC and with a high resolution camera interface, on a high capacity bandwidth for a camera serial interface (which can be integrated on to a processor), for a mobile technology applications, this driver development can be enhanced to make a commercial hand held device.

References

[1] IOSR Journal of Electronics and Communication Engineering (IOSR-JECE) ISSN: 2278-2834, ISBN: 2278-8735, PP: 32-36 www.iosrjournals.org Second International Conference on Emerging Trends in Engineering (SICETE) 32 | Page Dr.J.J.Magdum College of Engineering, Jaysingpur Image Processing Based on Embedded Linux 1Mr. S. M. Gramopadhye, 2Prof. R. T. Patil, 3Mr. A. N. Magdum, 4Mr. R. A. Chaugule 1,2(RIT Sakharale, 3SGI Atigre, 4JJMCOE Jaysingpure).

[2] Research and Implementation of the CMOS Camera Device Driver Based on S3C2440 Min Zhang ,Jin-guang Sun ,Shi Wang, 2010 International Conference on Intelligent Computation Technology and Automation.

[3] Yuan Weiqi, Tang Yonghua. Application of SCCB of OmniVision camera in DSP[J]. Chinese Journal of Scientific Instrument. 2006_27_6__1687-1688. (in Chinese

[4] OmniVision Serial Camera Control Bus(SCCB) Functional Specification. OminiVision Technologies. Version:2.1, 2003,2

[5] Samsung Electronics S3C6400/6410 Scenario Based APPLICATIONS User's Manual (Linux)