

Camera Device Driver Development Based on Windows CE

Woonchul Ham and Hyunsok Choi

Electronics and Informatic Division, Chonbuk National University, Jeonju, Chonbuk, Korea

Tel: 82-063-270-3561; Fax:82-063-270-2394;

Tel: 82-063-270-2399; Fax:82-063-270-2394;

Abstract: Nowadays, camera is an one of most commercial products that are usually found in the market. It means that camera products are popular product and these popularity is likely to remain continue in the near future. In this paper, we describe a method of constructing camera device driver of Windows CE on embedded system. Used product for test is that MBA2440 board of AIJI system company, S3C2440A CPU of Samsung company, PO2030N camera of Pixelplus company. We used Windows CE operating system to operate the embedded board and the camera is operated with a camera device driver of Windows CE. First we describe firmware program which operate camera under a microprocessor level, then we describe a method of constructing camera device driver of Windows CE.

Key-Words: Camera, Device Driver, Windows CE, S3C2440, Embedded System, IIC

1 Introduction

It is easy to see various camera products in every palce today. It is due to a drop in prices of the camera modules as scientific and technical advanced. The use of a camera in various areas is increasing these days and it means that technique for operating camera needs in many areas. So in this paper, we describe a method of operating a camera on embedded system. The reason we choose embedded system is that the number of people interested in embedded system by its many advantages. We used Microsoft Windows CE which is increasing its use for operating system of embedded system. To operate a camera module, we set the register of camera module by using IIC interface. To capture a image, Camera module signals YCbCr image data and S3C2440A(ARM core) CPU must transform YCbCr data into RGB data through a its inside camera controller. We use camera interrupt of Windows CE to receive right image from the camera. The used embedded board for test is MBA2440 of AIJI system company which use SAMSUNG's S3C2440A 16/32-bit RISC microprocessor. And we used Pixelplus's PO2030N camera module which use VGA(640x480) CMOS image sensor. We use Microsoft Windows CE platform builder 4.2 for developing a platform. This work was supported by the second stage of Brain Korea 21 Project and Mechatronic Research Center in 2006.

We describe the way to make the camera device driver based on the Windodows CE. The camera used in test is PO2030N which is manufactured by Pixelplus company. It is never been made the PO2030N'S camera device driver based on Wnidows CE before. It's a first try.

2.1 Overall Operation of the camera

Basis operation is as following .First camera controller provide a camera module with system clock. After at this point, camera module operate in properly. camera module gets a image data from CMOS sensor and transmits image data to the board. The board get image data from the camera with synchronized pixel clock, vertical sync, horizontal sync signals. Image data in the board is transfered to system memory through data

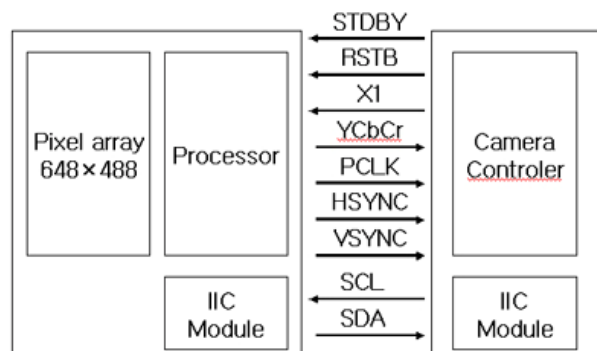


Fig.1 Overview of the camera block diagram

2 Developing camera device driver

bus lines. Then cpu display the image to lcd panel or store image data to the storage device. Most digital

camera operate like to this routine.

2.1.1 specific operation of the camera module

Camera module is composed of Fig.2 block diagram, and has 648*488 pixel CMOS sensor that convert image information coming from lens to electronic signal. Then image data is transmitted to Image Signal Processing block. After image Signal Processing Block converts image data, it transmits image data with synchronized signal to external port. PO2030N camera module supports ITU-R BT 601/656. In this test, we use ITU-R BT 601/656 type. To change camera control register, we use IIC interface. There are three external lines in camera module for IIC. The lines are composed of three parts. Each line is that clock line, communication line, control line.

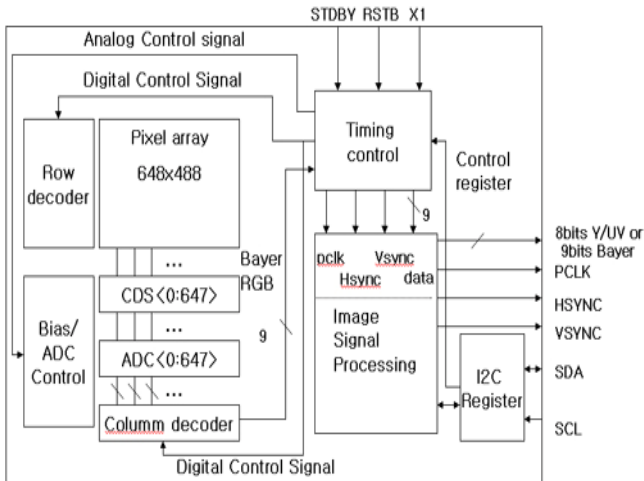


Fig.2 Block diagram of the camera module

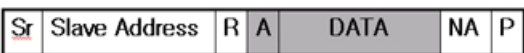
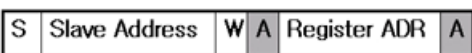
2.1.2 IIC communication

IIC interface was proposed PHILIPS company first.

Single Write Mode Operation



Single Read Mode Operation



□ From Master to Slave ■ From Slave to Master

Fig.3 Data packet of the IIC

It is composed of master and slave. More than two slaves communicate with master through an one data

line. IIC communication is simple because of using only SDA and SCL lines for communication. Also clock was supplied by master. Therefore there is no need to synchronize with slave device. Fig.3 shows data packet for IIC to communicate with camera and CPU.

2.1.3 Camera controller

S3C2440 cpu has camera controller inside itself. Which controller is consist of seven parts. Each part is that pattern mux, capturing unit, preview scaler, codec scaler, preview DMA, codec DMA, SFR. The maximum input size of image is 4096x4096 pixel and camera controller supports 2048x2048 pixel size for scaling. Fig .4 show each parts of the controller. Look into the data follow, T_patternMux receive image data from camera module to synchronize with pixel clock, then transmittes image data to a CatchCam. Then CatchCam transmits received image data to the preview scaler or the codec scaler. Each scaler resize the image and transmits it to DMA. The DMA store a system memory with image data, and cpu store storage device with image data or display image on LCD panel. Camera controller supports two view mode which is Preview mode and Codecview mode. In the Previ-

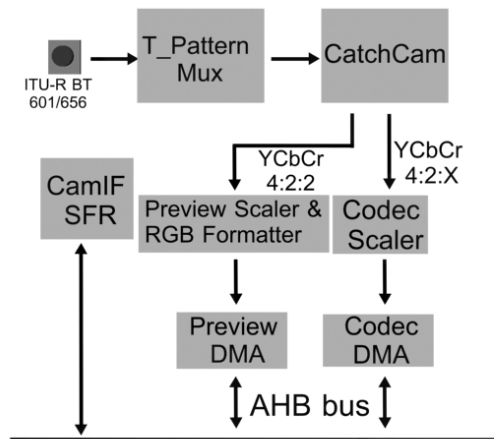


Fig.4 Block diagram of the camera

ew mode, hardware converts YCbCr image data into RGB image data automatically. In the Codecview, YCbCr image data which received from camera is converted to RGB image data by cpu for more quality image. If converting process should implemented in hardware, image frame rate should be drop. Preview mode receive only YCbCr(4:2:2) image data format. In contrast, the Codecview mode receive

YCbCr(4:2:x). It means that Codecview mode can receive several sampling frequency of image data. 4:2:2 means sampling frequency when analog signal is converted to digital signal. Following expression is converting expression to convert YcbCr data to RGB data in Codecview mode.

$$\begin{aligned}
 Y &= 0.299R + 0.587G + 0.1148B \\
 Cb &= 0.564(B-Y) + 128 \\
 Cr &= 0.713(R-Y) + 128
 \end{aligned}
 \tag{1}$$

$$\begin{aligned}
 R &= Y + (Cr-128)/0.713 \\
 B &= Y + (Cb-128)/0.713 \\
 G &= (Y - 0.299R - 0.114B)/0.587
 \end{aligned}
 \tag{2}$$

2.2 Image capture in the firmware

In firmware program, cpu is used like a microprocessor. Firmware program is easy to operate the camera device. Therefore we had been tested the camera on the firmware program first. This firmware program is very important for developing camera device driver of operating system. Because device driver program is

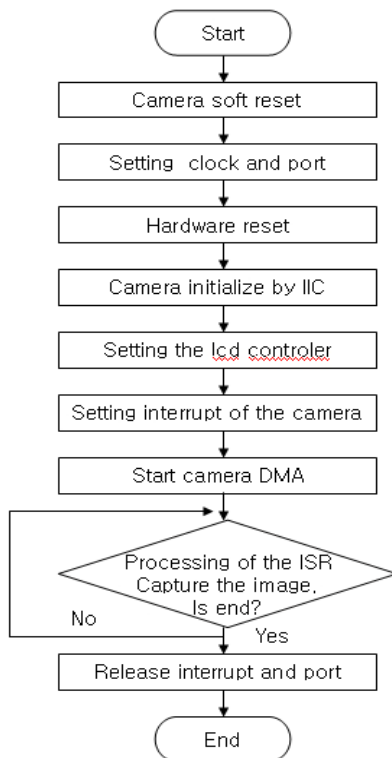


Fig.5 Flow char of display image

same to firmware program to hardware controlling in many aspects. Fig.5 sequence diagram shows the operation of the camera overallaly. This sequence will be used in camera device driver of the Windows CE.

In the sequence, the part of IIC communication initialize the camera register to YCbCr data format. This setting of camera control register is that select the image size and assigned system memory to the DMA for a pingpong memory. Interrupt of the camera is divided into between Preview interrupt and Codecview interrupt and it cannot be excuted at the same time.

2.3 Developing the camera device driver

Camera device driver is different when compared to firmware program. Because device driver must run cooperated with operating system.

2.3.1 Device driver model

Device driver of Windows CE divide into build-in driver and installable driver. The build-in dirver is loaded to the system memory when operating system is starting and it become runnable status. On the contrary, installable driver is loaded when specific device driver are needed. The camera device driver is former case. Also device driver can be dived into layered device driver and monolithic device driver. Layered device driver is composed of a class structure which dived into MDD and PDD class. On the contrary, Monolithic driver can be excuted on stand alone. The camera device driver is Monolithic driver. Monolithic dirver has the advantage of fast processing than layered device driver relatively. Because it process camera routine without passing through the upper class.

2.3.2 component of the file

In order to make camera device driver, we linked coredll.lib and drvlib.lib and camera progrma that was compiled. After compiled and linked, camera.dll file will be generated. camera.dll process capturing image from camera and display image to the LCD panel. Before make a camera.dll, we have to make source file that configure a compiling setting. Also we needs *.def file that configure external function that called by application.

2.3.3 Create device driver

In this paper, we through windows CE platform builder 4.2 that integrated development tool for the windows CE build the windows CE operating system. First we add the camera device driver to the catalog of

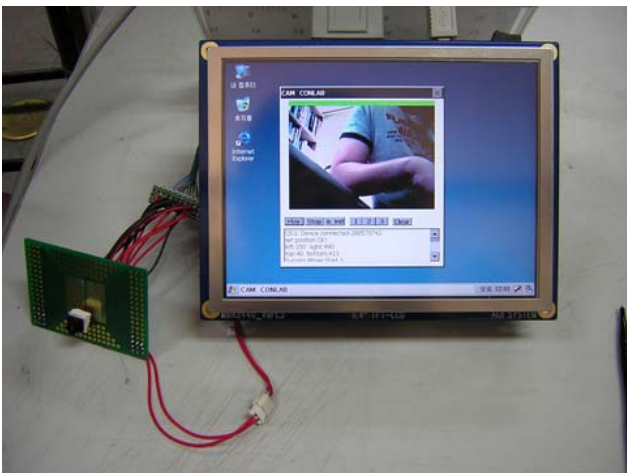
the platform for adding camera device driver to the platform. Then we put the camera module into the current platform. Also we registered camera device driver to platform.reg file. Following sample shows registration of register file.

```
[HKEY_LOCAL_MACHINE\Drivers\BuiltIn\Camera]
"Prefix"="CIS"
"Dll"="camera.dll"
"Index"=dword:1
"Order"=dword:1
```

Finally build the platform, the platform which include camera module will be created.

2.3.4 Inside of the camera device driver

When the operating system is starting, camera device driver is loaded to the system's memory by device manager. And initialize the camera device driver. The part of initialize is the same with the firmware program in every aspects. But there are some different processes in the operating system. It is use of virtual address which different with physical address. Also we can only use the memory by a assignation of the operating system. We don't need to initialize display routine in the camera device driver, because initialization of display driver is performed by display driver. Different like the firmware firmware, device driver only use interrupt by requesting interrupt to operating system. If device driver request for interrupt from the operating system, then operating system creates event related with interrupt. Then program waits event until receive the interrupt. In cases like a camera, the interrupt is occurred by a vertical sync signal which occurred when received an one frame



image

Fig.6 Image capturing from the camera

data from the camera. When interrupt is occurred , interrupt service routine copy the memory that an one frame image data to frame buffer of DMA display in the Preview mode. In the Codecvew, a YCbCr image data in the memory is converted into RGB first, then it transferred to the graphic frame buffer. we implemented camera device driver that was made we explained, then it worked properly. Fig 6 shows an application which play the capturing image.

3 Conclusion

Till We had been made camera device driver of Windows CE that we described so far, and it implemented on embedded system. Image frame rate was high when Preview mode is operated. On the contrary, we found image frame rate low when Codecvew mode is operated, as compared, to the Preview mode, because YCbCr image data has to be transformed into RGB image data by software. Camera device driver capture a image after interrupted. There are two methods to display captured image, an one method is that display captured image to display panel directly and another method is that display captured image through the GWES of Windows CE. In the case of direct display method, device driver processes display part, so frame rate is high as DirectX. But many process should be processed in device driver. In the case of using GWES, frame rate is a little bit low than direct display method, because another imaging process is needed additionally. But it has the advantage of that device driver is simple. And later on, we will apply this camera system to some applications.

References:

- [1] "Developing a Board Support Package for Microsoft Windows CE.NET," Microsoft, 2002.
- [2] Byungchan Ahan, "Porting Windows CE Operating System to ARM based board device," ICCAS, 2003.
- [3] Sang hyung, Kim, "Windows API conquest," Kanamsa, 2001.
- [4] "S3C2440A 32-bit RISC Microprocessor User's Manual," SAMSUNG Electronics, 2004.
- [5] "PO2030N 1/4.5 Inch VGA Single Chip CMOS IMAGE SENSOR", Pixelplus, 2005