International Journal of Ethics in Engineering & Management Education
Website: www.ijeee.in (ISSN: 2348-4748, Volume 1, Issue 9, September 2014)

CMOS Camera Implementation with CSI based on ARM Processor

P. Anusha
Electronics & Communication Engineering
Joginapally BR Engineering College
Hyderabad, India
anusha.pallothu@gmial.com

K. Ravinder
Electronics & Communication Engineering
Joginapally BR Engineering College
Hyderabad, India
kravinder@gmail.com

***Abstract*-This paper discuss about the implementation of a CMOS camera (OV5647) sensor with MIPI CSI on ARM processor in considered using Linux environment can be used to take high definition video, as well as stills photographs. In this paper the hardware structure, device driver design, key program implementation & transplantation are described in detail with the MIPI CSI & V4L2 structure on the ARM processor (BCM2835) embedded platform are emphasized.**

Keywords: BCM2835, RPi, OV5647 CMOS camera, MIPI CSI, V4L2 protocol, SCCB, differential serial transfer

I. INTRODUCTION

This paper discuss about the stream of embedded system which can be defined as is a combination of computer circuitry and software that is built into a product for purposes such as control, monitoring and communication without human intervention. Some embedded systems include an operating system which is referred to as an embedded operating system considered as software for the system. The software is highly dependent on the CPU & specific chips chosen. Most embedded engineers have at least a passing knowledge of reading datasheets & reading schematics for components to determine usage of registers.

An ARM (Advanced RISC Machine) is the industry's leading provider of 16/32-bit embedded RISC (Reduced Instruction Set Computer) microprocessor solutions. ARM designs microprocessor technology that lies at the heart of the advanced digital products, from mobile phones and digital cameras to automotive systems. ARM's architecture is compatible with all four major platform Symbian OS, Palm OS, Windows CE, and Linux. Broadcom's application processor BCM2835 (ARM1176JZ-F) is a cost-optimized, full HD, multimedia applications for advanced mobile and embedded applications that require the highest levels of multimedia performance. The ARM1176JZF-S processor is built around the ARM11 integer core. It is an implementation v of the ARMv6 architecture that runs the ARM, Thumb, and Java instruction sets.

A camera sensor is a chip with optics, some driving circuits & interfaces. And image processing uses two types of sensor technologies CCD sensor and CMOS sensor. A CCD (charge coupled device) has photo sites, arranged in a matrix; each comprises a photo diode which converts light into charge & a charge holding region. The charges are shifted out of the sensor as a bucket brigade. In CMOS sensor each pixel contains a photodiode which convert light into electrons; it consists of charge to voltage conversion section, reset & select transistor, and an amplifier. CMOS sensors are cheaper, mass produced in mobile phones, toys, and produced in high end applications.

Using different interface technologies for connecting host processor with the CMOS camera using USB is one of the traditional technologies. The majority of cameras in high volume consumer products, such as smartphones and tablets, use MIPI based sensors. The most commonly used interface for this type of image sensor is the CSI-2 specification. Here CMOS camera (OV5647) has been chosen with CSI by considering V4L2 structure on BCM2835 processor.

This paper organized as follows: section 2 describes the hardware components, section 3 describes how to design V4L2 driver for camera, section 4 explains the kernel image and driver loading onto the board, and section 5 shows the result and conclusion of the paper.

I. HARDWARE DESCRIPTION

The hardware block diagram of the project can be shown at figure 1. BCM2835 ARM11 microprocessor is the core of the raspberry pi board. Among the different development boards available to implement CSI our first choice, which has an extremely low power draw, small form factor, no noise, solid state storage.

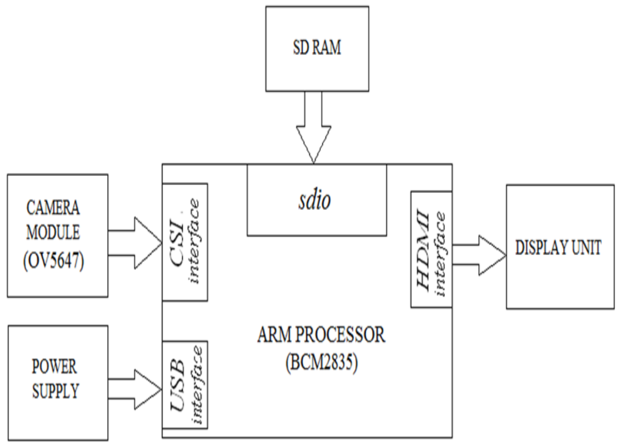
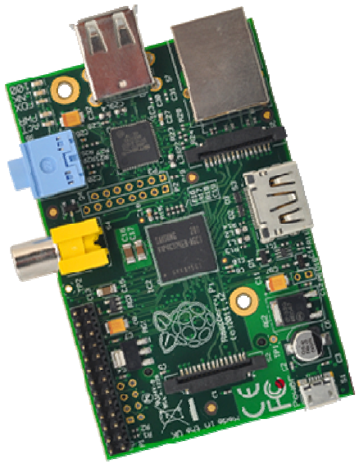


Figure 1: Block diagram of the hardware components

International Journal of Ethics in Engineering & Management Education
Website: www.ijeee.in (ISSN: 2348-4748, Volume 1, Issue 9, September 2014)

The Raspberry Pi device looks like a motherboard, with the mounted chips and ports exposed (something you'd expect to see only if you opened up your computer and looked at its internal boards), but it has all the components you need to connect input, output, and storage devices and start computing. The foundation picked a chip with ARM architecture for this reason (processor architecture commonly used for mobile phones and similar devices). The chip has 256 MB of RAM, runs at 700 MHz and includes a 1080p-capable GPU.



A. Interfacing CMOS camera with CSI to BCM2835

We took up interfacing, development and integration of Omnivision's OV5647 camera sensor to this board for increasing use cases with this low-cost computer. Raspberry Pi has its camera interface brought out on board through a 15-pin connector socket. Camera sensor interfaced using CSI2- MIPI interface. CSI-2 provides the mobile industry a standard, robust, scalable, low-power, high-speed, cost-effective interface that supports a wide range of imaging solutions for mobile devices. For interfacing the camera sensor we have used:

i. Two pair of differential data lines is used
ii. One pair of differential clock line is used.
iii. Two wire I2C control lines are used.

With appropriate ground lines this set constitutes the 15-lines from Raspberry Pi can be shown in figure 2.

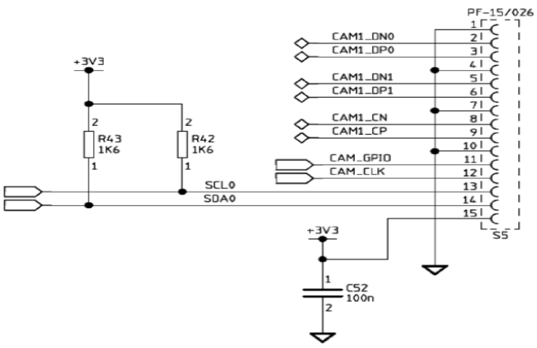


Figure 3: CSI lines from RPi

CPI uses 8 or 10-bit parallel data lines are used in this case. Maximum pixel clock supported is usually 96 MHz. CCI usually a two or three-wire interface used to control the sensor module. It usually confirms to the I2C standards. Whereas CSI uses Data and synchronization information is sent over differential serial lines. There could be up-to 4 data lane pairs plus 1 clock pair. The specification conforms to a standard set up by MIPI sensor module.

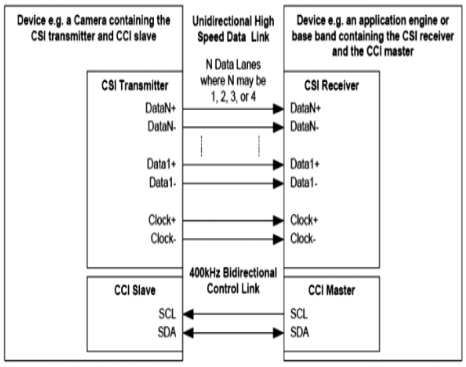


Figure 2: CSI transmitter and receiver block diagram

We have implemented CSI2 interface for connecting OV5647 (OV- Omni vision, 5-5 megapixel digital sensor, 6- color digital, 4- major iteration of chip, 7- minor iteration of chip) sensor to the Raspberry Pi development board.

B. OV5647 sensor

OV5647 is a low voltage, high performance, 5megapixel CMOS image sensor that provides 2592x1944 video output using OmniBSI technology. It provides multiple resolution raw images via the control of the serial camera control bus or MIPI interface. The OV5647 has an image array capable of operating up to 15 fps with user control of image quality, data transfer, camera functions, through the SCCB interface. The OV5647 sensor core generates streaming pixel data at a constant frame rate, indicated by HREF and VSYNC.

OmniBSI technology offers significant performance benefits over front side illumination technology, such as increased sensitivity per unit area, improved quantum efficiency, reduced cross talk and photo response non-uniformity, which all contribute to significant improvements in image quality and color reproduction. Additionally, omnivision CMOS image sensors use proprietary sensor technology to improve image quality by reducing or eliminating common lighting/electrical sources of image contamination, such as fixed pattern noise and smearing to produce a clean, fully stable color image.

International Journal of Ethics in Engineering & Management Education
Website: www.ijeee.in (ISSN: 2348-4748, Volume 1, Issue 9, September 2014)

II. SOFTWARE DESCRIPTION

The camera drivers are proprietary in the sense that they do not follow any standard APIs. That means that applications have to be written specifically for the Raspberry Pi camera. Under Linux, the standard API for cameras (including web cams) is V4L (Video for Linux), and a number of applications have been written that support any camera with a V4L driver.

A.Integrating camera sensor to BCM 2835 (Raspberry Pi)

Here CMOS camera implemented along with csi2 interface so the driver used is SH_CSI2_INIT(). So the csi driver calls the following functions [module_init, v4l2_device_register & video_register_device] to register with the v4l2 framework. It also provides the video frame grabber capability to the system. CSI driver provides implementation of the v4l2_ioctl_ops IOCTL operations for the device. These are the operations which get called from user-space. In the v4l2 framework, the driver needs to register itself as a subdev [v4l2_subdev_init]. More often than not sensor drivers are connected and detected using the i2c interface. The i2c slave address, bus number and other info (the power up sequence, pixel format, default resolution supported, the Hsync/Vsync polarity etc.) is provided in the board file (raspberrypi/linux/drivers/media/video/ov5642.c in this case).

Sensor platform data in the board file is entered through the structure ov5640_platform_camera.

After module_init() call it will read the V4L2_subdev_ops() which perform the operations driver probe(), bus_paramtr() to gather the information about pixel and image format like YUV(4:2:2), YUV(4:2:0), RGB565, RGB555 with respective pixel data information. And csi_hwinit() is used to enable MIPI clock and data lanes from the sensor with csi_client_config() functions.

III. SOFTWARE PROCEDURE

Initially the board must be loaded with the kernel image by using SD card in this project we have used raspbian as an image.

Step 1: Download the OS image Raspbian.
Step 2: Using windows source insight dump the image to the SD card.
Step 3: Now insert the SD card into the Raspberry pi board SD slot.
Step 4: Connect USB cable for power supply to the board.
Step 5: Using HDMI cable connect raspberry-pi to the monitor.
Step 6: And connect input devices to the target (like keypad, mouse).
Step 7: Now finally we can see the raspberry image on the host .

There are three applications provided, raspistill, raspivid and raspistillyuv. raspistill and raspistillyuv are very similar and are intended for capturing images, raspivid is for capturing video. All the applications are command line driven, written to take advantage of the mmal API which runs over OpenMAX. The mmal API provides an easier to use system than that presented by OpenMAX. Note that mmal is a Broadcom specific API used only on Videocore 4 systems. The applications use up to three OpenMAX(mmal) components - camera, preview and encoder. All applications use the camera component, raspistill uses the Image Encode component, raspivid uses the Video Encode component and raspistillyuv does not use an encoder, and sends its YUV output direct from camera component to file. The preview display is optional, but can be used full screen or directed to a specific rectangular area on the display. In addition it is possible to omit the filename option, in which case the preview is displayed but no file is written, or to redirect all output to stdout. Command line help is available by typing just the application name in on the command line.

After board bring up procedure interface the camera to raspi.

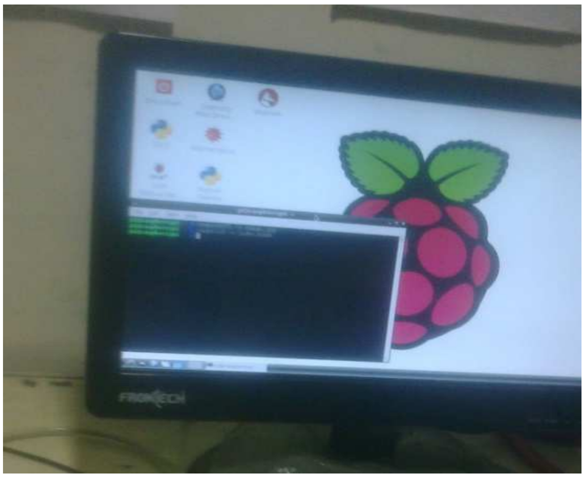
Step 1: Connect the camera to CSI port.
Step 2: Now use the commands for camera.
raspi-config
enable the camera then boot up
Step 3: #Capture a JPEG image
raspistill –o image.jpg
#Capture a 5s video in h264 format
raspivid -o video.h264

Take a capture at a different resolution.
raspistill -t 2000 -o image.jpg -w 640 -h 480

Record a 5s clip at a specified frame rate (5fps)
raspivid -t 5000 -o video.h264 -f 5

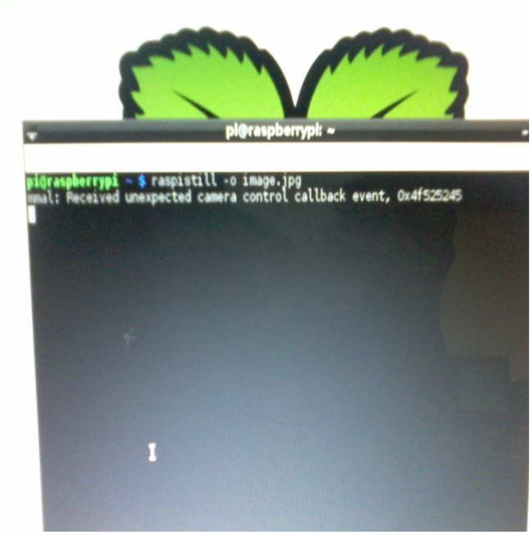
IV. RESULTS & CONCLUSION

CMOS camera (OV5647) driver on ARM processor (BCM2835), raspberry pi board is considered using embedded Linux environment.



International Journal of Ethics in Engineering & Management Education

Website: www.ijeee.in (ISSN: 2348-4748, Volume 1, Issue 9, September 2014)



\

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]. 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

[3]. 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.

[4]. Developing CMOS Camera and USB Device Drivers in Linux 2.6.32 CH. P. N. S. Sujitha Electronics & Communication Engineering Department, M.V.S.R. Engineering College, Osmania University, Hyderabad, A.P., India DVSR Sesidhar Assistant Professor, E.C.E. Department, M.V.S.R. Engineering College, Osmania University, Hyderabad, A.P., India