INTEGRATING USB IN IMAGE SENSING APPLICATIONS

By Gopalakrishnan Vijayakumar, Applications Engineer Sr., and Hridya Valsaraju, Applications Engineer Sr., Cypress Semiconductor Corp.

USB has created its own market space by replacing interfaces such as PS/2, UART and parallel port, and today's laptops are mostly devoid of these legacy interfaces. It is a rare sight today to see a PS/2 keyboard or mouse being used on a laptop as they have been replaced by more flexible USB keyboards and mice. USB, however, is not just for consumer electronics devices. It also has a wide variety of applications in image sensing market, including cameras, biometric applications like fingerprint-based USB drives, fingerprint desktop authentication, and user finger print enrollment, to name a few.

Universal Serial Bus

USB 2.0 was the latest until the advent of USB 3.0 specification. USB 2.0 covered physical signaling rates of 1.5 Mbits/s (Low Speed), 12 Mbits/s (Full Speed) and 480 Mbits/s (High Speed). USB 3.0 was introduced to increase the physical signaling rate (up to 5 Gbit/s), to decrease power consumption, to increase power output and to be backwards-compatible with USB 2.0. The USB 3.0 standard supports backwards-compatible by implementing the USB 2.0 bus in parallel with the USB 3.0 bus. Many semiconductors companies have started sampling USB 3.0 chips. For example, Cypress has sampled the industry first flexible general-purpose USB 3.0 family of controllers, the EZ-USB FX3™.

Throughput Analysis of USB 2.0

Throughput is one of the major reasons for designing USB into any application. This is true in the case of image sensing applications as well. To achieve smooth uninterrupted video, the USB 2.0 specification support a dedicated bandwidth of 24 Mbytes/s via isochronous transfer and the bulk transfers can provide higher throughputs in excess of 40 Mbytes/sec based on factors like system bandwidth allocation for the USB Host Controller, operation system overhead, the presence of other USB devices, and so on. Choosing an efficient high-speed controller ensures the maximum frame rate optimization and easy upgrading to accommodate the ever-expanding offering of image sensors. However, high-speed USB becomes a bottleneck above VGA resolution. Theoretically, USB delivers uncompressed video at 30FPS @ VGA and 15FPS @ 1.3MP. USB 3.0 should be able to deliver maximum frame rates beyond 1.3MP.

Imaging Applications:

The two main Image sensing applications into which USB is being widely integrated are webcams and biometric applications. Webcams and PC cameras are creating a visual revolution for both business and personal purposes by users worldwide. Some of the market areas which are addressed by webcams are:

- Videoconferencing
- Video instant messaging
- Social networking services

Biometric applications comprise methods for uniquely recognizing individuals based upon one or more intrinsic physical or behavioral traits. Increased focus on securing access control has led to a huge growth in the area of biometrics.

The physical traits used for uniquely identifying individuals are those of fingerprint, face, iris, and palm. There are two modes in which a biometric system operates. In verification mode, the system makes a one-to-one comparison of the scanned template with the stored template. An example for this mode will be fingerprint desktop authentication. Here the user of the desktop enrolls his or her fingerprint and this enrolled template is compared with the fingerprint of any person who tries to access the system.

In identification mode, the system performs one-to-many comparisons with the stored template in the biometric database. For example, a system designed to control entry to a secured office area will utilize this mode. Here, the scanned fingerprint template of an employee will be compared with the templates of the group of employees who are allowed access into the area. USB finds its application in the initial stage of finger print enrollment, where the finger prints are scanned, processed, and stored in the biometric database.

Published in Industrial Control DesignLine (http://www.eetimes.com) August 2011

Biometric security mechanisms are widely used for applications like

- Personal computer / workstation security.
- Network / enterprise security.
- Banking and financial security systems.

USB-Image Sensor (IS) Interface:

In both of the above image sensing applications, the key step in the implementation is the transfer of captured images (with/without processing) to the PC. There are four different methods of implementing this interface:

USB+FPGA+IS:

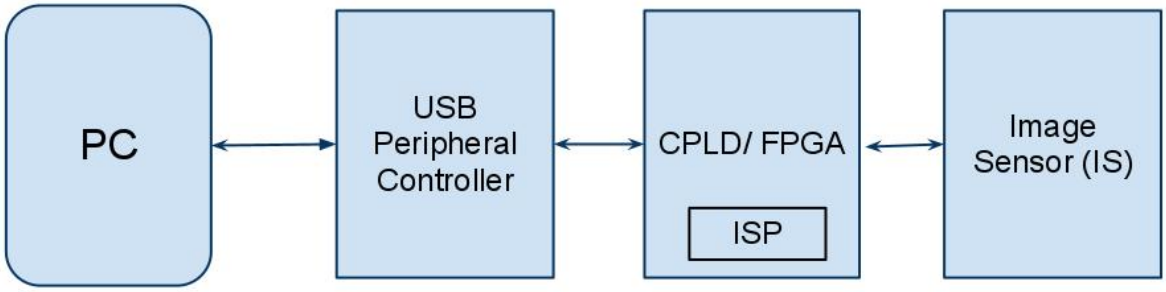

Figure 1: USB + FPGA + Image Sensor Interface

The design of a USB-based Image sensor interface is shown in **Figure 1**. Here, an FPGA is used to configure the image sensor, give control signals to sensors which do not have capability to supply their own control signals, and to perform the role of the Image Sensing Processor (ISP). The image sensor outputs the raw image data and this raw image data is sent to the FPGA where processing, compression, and/or encryption of the image is completed. The processed image data is then sent to the PC by the USB peripheral controller.

USB+IS (with ISP):

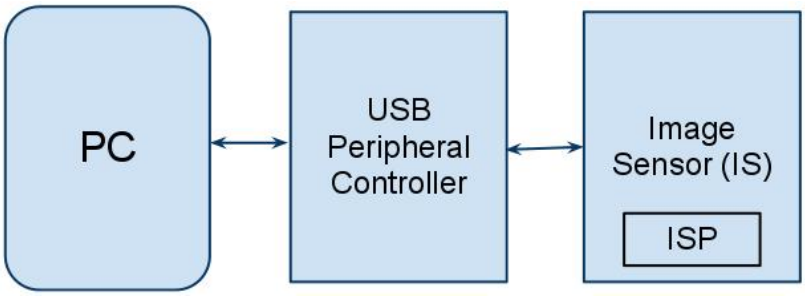

Figure 2: USB + Image Sensor (with ISP) interface

The second method interfaces a USB peripheral controller to an image senor with an integrated ISP. In this implementation, image processing is done within the image sensor IC. The processed image data is sent to PC through the USB peripheral controller.

IS (with ISP+USB):

Some image sensor manufacturers have integrated the USB controller with the ISP onto a single chip. One drawback with this method is that the programmability of the integrated USB controller is limited.

USB+IS (without ISP):

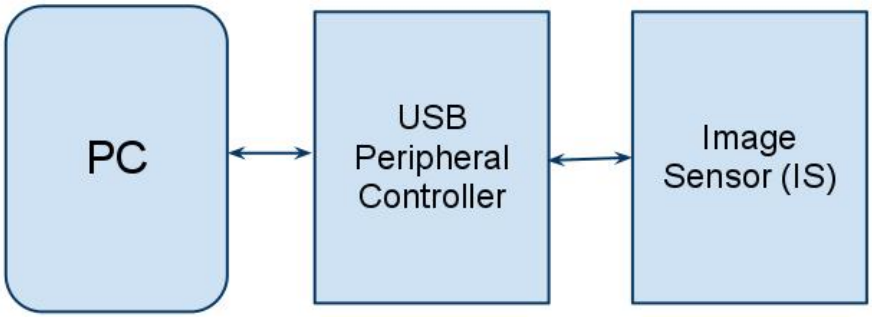

Figure 3: USB + Image Sensor (with ISP) interface

A PC with sufficient processing power can be used for image processing instead of an FPGA/ISP-based SoC. The interface becomes simple, as shown in the Figure 3, and effectively reduces the board space required. The USB peripheral controller acts as a data bridge between the PC and image sensor and transmits the raw image data to the PC. If a programmable USB peripheral controller is used, it can configure the image sensor over I2C, insert a header to each frame of the raw image data, and supply the master clock to the sensor thus eliminating the need for an extra crystal to clock the Image Sensor. This design is more cost effective than the other design techniques discussed above.

Practical Implementation of High Speed USB-IS Interface:

The fourth approach utilizes a programmable high-speed USB controller with embedded microcontroller, such as the Cypress EZ-USB FX2LP™. The controller handles most of the USB 1.1 and 2.0 protocols in hardware, thereby freeing the embedded microcontroller to perform application-specific functions as well as decreasing the overall development time required to ensure USB compatibility. The controller can be configured to work in the relatively simple Slave FIFO mode for this application.

The basic signals of most image sensors are Master Clock (Clock to the Image sensor), Configuration interface (e.g. I2C), Pixel clock (PCLK), data lines, control signal (VSYNC or frame sync, HREF or line sync). A typical example of VSYNC and HREF lines for an image data with resolution 640x480 is given below (see Figure 4).

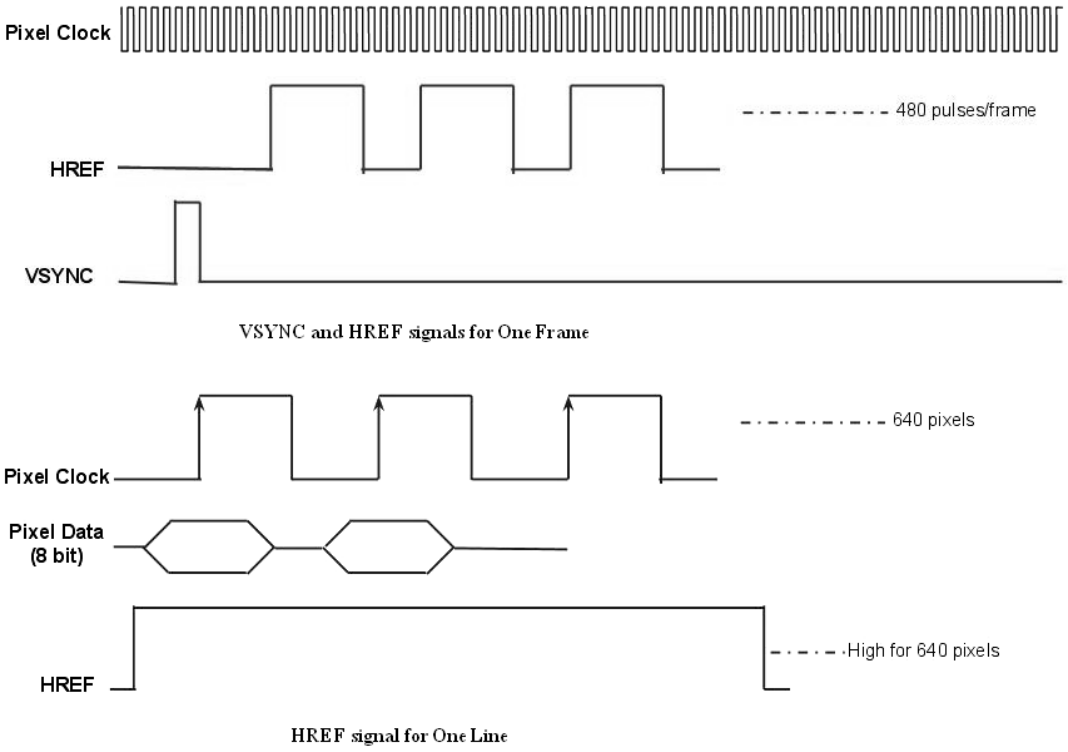

Figure 4: VSYNC and HREF signals for one line and one frame

The VSYNC can either be a short pulse or continuously high for one frame based on the configuration of the image sensors. As shown in figure, for an image with resolution of 640x480, 640 bytes are available on every high of HREF with respect to the PCLK and the HREF has 480 pulses per frame. Figure 5 shows the block diagram of this implementation.

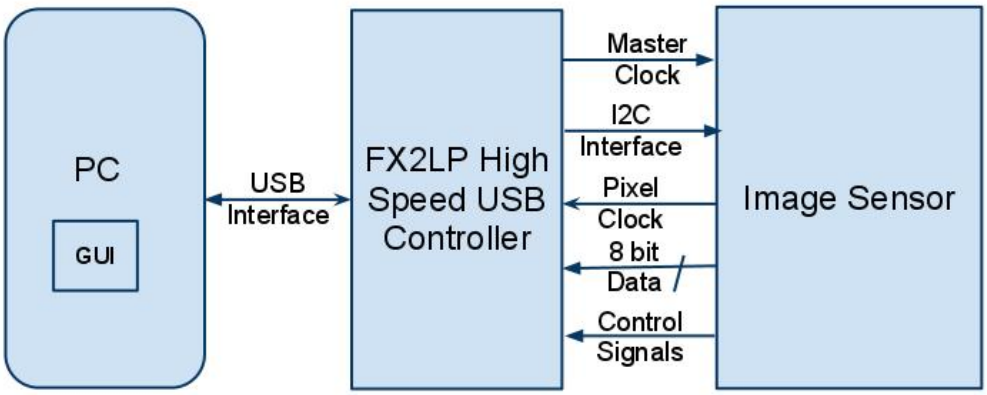

Figure 5: Block diagram of USB high speed controller + Image Sensor interface

The USB Controller supplies the master clock for the sensor. The configuration data is written to the image sensor configuration registers using the I2C module of FX2LP. Once the image sensor has been configured, it outputs the raw image data with reference Pixel Clock (PCLK). VSYNC is used by the USB controller to insert frame header information for the image data to isolate the frames. HREF and PCLK are used as control signals to transfer the data. The data from the image

sensor is sent to the FIFO in the USB Controller, which is then transmitted to the USB Host in the PC. A suitable application on the PC can process the Image data received from the USB Controller.

The value to the flourishing image sensor market segment can be immensely increased by integrating USB to existing image sensor design. The addition of USB will increase the device functionality and provide an additional method to control and upgrade devices in the field.

Cypress Semiconductor
198 Champion Court
San Jose, CA 95134-1709
Phone: 408-943-2600
Fax: 408-943-4730
http://www.cypress.com