Field-programmable gate array-based hardware architecture for high-speed camera with KAI-0340 CCD image sensor

Wang Hao\textsuperscript{a,}, Yan Su\textsuperscript{a,}, Zhou Zuofeng\textsuperscript{a,}, Cao Jianzhong\textsuperscript{a,}, Yan Aqia\textsuperscript{a,}, Tang Liniao\textsuperscript{a,}, Lei Yangjie\textsuperscript{a}
\textsuperscript{a}Xi’an Institute of Optics and Precision Mechanics, Chinese Academy of Sciences, Xi’an, P.R China, 710119

ABSTRACT

We present a field-programmable gate array (FPGA)-based hardware architecture for high-speed camera which have fast auto-exposure control and colour filter array (CFA) demosaicing. The proposed hardware architecture includes the design of charge coupled devices (CCD) drive circuits, image processing circuits, and power supply circuits. CCD drive circuits transfer the TTL (Transistor-Transistor-Logic) level timing Sequences which is produced by image processing circuits to the timing Sequences under which CCD image sensor can output analog image signals. Image processing circuits convert the analog signals to digital signals which is processing subsequently, and the TTL timing, auto-exposure control, CFA demosaicing, and gamma correction is accomplished in this module. Power supply circuits provide the power for the whole system, which is very important for image quality. Power noises effect image quality directly, and we reduce power noises by hardware way, which is very effective. In this system, the CCD is KAI-0340 which is can output 210 full resolution frame-per-second, and our camera can work outstandingly in this mode. The speed of traditional auto-exposure control algorithms to reach a proper exposure level is so slow that it is necessary to develop a fast auto-exposure control method. We present a new auto-exposure algorithm which is fit high-speed camera. Color demosaicing is critical for digital cameras, because it converts a Bayer sensor mosaic output to a full color image, which determines the output image quality of the camera. Complexity algorithm can acquire high quality but cannot implement in hardware. An low-complexity demosaicing method is presented which can implement in hardware and satisfy the demand of quality. The experiment results are given in this paper in last.

Keywords: High-speed camera, Color demosaicing, Auto-exposure, FPGA

*wanghao@opt.ac.cn; phone 86 029-88887821; fax 86 029-88887821

1. INTRODUCTION

Recently, cameras are designed based CMOS image sensors (CISs) and CCDs (charge coupled devices). The designs based CMOS image sensors process the advantages of low power, low cost, high level of integration, etc. But the image quality produced by a CIS is inferior to that of the CCD [1]. CCDs have extremely low noise, low dark current, high Dynamic Range, etc. These features give this sensor exceptional sensitivity and make it ideal for machine vision, scientific, surveillance, and other computer input applications. In this paper, we introduce a new design based CCDs.
Many designs of camera have been proposed, and most of them emphasize the timing sequence and image processing implemented by a digital signal processor (DSP) in a digital still camera (DSC) [2]. For the high frame frequency and large data stream, DSP cannot dispose in real time. In this paper, field-programmable gate array (FPGA) based hardware architecture is given in Fig.1.

![Fig.1 System Diagram of the CCD Electronics](http://proceedings.spiedigitallibrary.org/)

There is a FPGA, CCD sensor, and two analog to digital converters (ADC). The pulse generator produces square pulses of a frequency which is decided by the number of CCD pixels and the frame-rate. The clock driver receives the TTL (Transistor-Transistor-Logic) level pulses from the pulse generator and transfers them to the sensor after adjusting pulses to the voltage level required by the sensor input standard. From these timing pulses, the CCD image sensor produces the analog output through amplifiers situated in the end. The ADC converts the analog output to 10 bits digital image and sends the data to FPGA.

The paper is organized as the following. Section 2 describes the hardware design of camera. Section 3 describes the timing sequence. Section 4 gives Algorithm used in the camera. The results are given in Section 5.

### 2. HARDWARE DESIGN

According to KAI-0340 datasheet [3], the Clock Levels is given in Table 1.

#### Table 1 KAI-0340 Clock level

<table>
<thead>
<tr>
<th>Name</th>
<th>Minimum (V)</th>
<th>Nominal(V)</th>
<th>Maximum(V)</th>
<th>Maximum Frequency</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vertical CCD Clock High</td>
<td>9.5</td>
<td>10</td>
<td>10.5</td>
<td>10kHZ</td>
</tr>
<tr>
<td>Vertical CCD Clocks Midlevel</td>
<td>-0.2</td>
<td>0</td>
<td>+0.2</td>
<td>10kHZ</td>
</tr>
<tr>
<td>Vertical CCD Clocks Low</td>
<td>-9.5</td>
<td>9</td>
<td>+8.5</td>
<td>10kHZ</td>
</tr>
<tr>
<td>Horizontal CCD Clocks High</td>
<td>-0.5</td>
<td>0</td>
<td>+0.5</td>
<td>40MHz</td>
</tr>
</tbody>
</table>
In this system, FPGA produce TTL signals, and we change TTL sequence timing to the voltage which CCD needs. The process of sequence timing is produced is introduced in Section 3. In this section, we introduce the hardware of voltage change, and the test results are given.

2.1 Vertical Clock Circuit
Vertical Clock Circuit is the module that we change the TTL voltage to the voltage -9.0V to 0V. We use the MAX4427 as the core chip to realize this function. The schematic and test result is given as Fig.2.

2.2 Horizontal Clock Circuit
Horizontal Clock Circuit is the module that we change the TTL voltage to the voltage -5.0V to 0V. We use the EL7457 as the core chip to realize this function. The schematic and test result is given as Fig.3.
2.3 CCD Output Circuit

CCD Output Circuit is the module that we increase the current of CCD output. We use the NPN transistor as the core chip to realize this function. This is a typical application ofEmitter follower circuit. The schematic and test result is given as Fig.4.

Fig.4 (a) The circuit of CCD output; (b) the test result of circuit

3. TIMING SEQUENCES DESIGN

3.1 Horizontal Clock timing

The Maximum Horizontal Clock Speed of KAI-0340 is 40 M HZ. In this system, we use 40 M HZ as clock speed. H1 and H2 are two Horizontal timing which CCD needs, and the clock phase of these timings is reverse. The timings is produced by FPGA, and result of emulator is given.

Fig.5 The simulation of horizontal clock timing

3.2 Vertical Clock timing

The Maximum Horizontal Clock Speed of KAI-0340 is 40 M HZ. In this system, we use 40 M HZ as clock speed. H1 and H2 are two Horizontal timing which CCD needs, and the clock phase of these timings is reverse. The timings is produced by FPGA, and result of emulator is given.
4. ALGORITHMS IN FPGA

4.1 A fast auto-exposure algorithm

Getting proper image exposure is a critical component of a digital or cell-phone camera image pipeline. In professional photography, the right exposure is normally decided manually either by visual inspection or by using an external exposure meter. However, in point-and-shoot digital and cell-phone cameras, an auto exposure (AE) module is used to automatically set the exposure level without any user intervention.

As discussed in [4-6], bright value (BV) is proportional to exposure duration $T$ and is inversely proportional to the square of $f$-number $F$. Hence, we can use $BV$ denote the exposure duration $T$. The $BV$ can be denoted by the image gray level.

Due to the simple but effective idea of the histogram, we use the histogram of the image gray level to compute the exposure time. We compute the weighted average of the image level using the statistical information of the histogram of the image level. Then the weighted average is compare with a pre-defined threshold to control the exposure time.

$$t_{target} = t_{current} + t_{step}$$

$t_{target}$ is the exposure time which we want to set;

$t_{current}$ is the exposure time which is current value of camera.

$t_{step}$ is variable step between $t_{target}$ and $t_{current}$.

$$Gray_{target} = Gray_{current} + Gray_{step}$$

$$Gray_{step} = Gray_{target} - Gray_{current}$$
4.2 Demosaicing Algorithm

The Bayer CFA pattern is designed is KAI-0340. There are many demosaicing Algorithm in [7-10]. Considering the complexity of Algorithm which can be in real time system, we choose the ACPI Algorithm [11]. In ACPI [11], the green plane is handled first and the other color planes are handled based on the estimation result of the green plane. When the green plane is processed, for each missing green component in the CFA, the algorithm performs a gradient test and then carries out an interpolation along the direction of a smaller gradient to determine the missing green component. A pixel without green component in the Bayer CFA may have a neighborhood as shown in either Fig. 1(a) or Fig. 1(b). Without losing the generality, here, we consider the former case only. In this case, the horizontal $\Delta H_{ij}$ gradient and the vertical $\Delta V_{ij}$ gradient at position $(i, j)$ are estimated first to determine the interpolation direction as follows:

$$\Delta H_{ij} = \left(G_{i,j-1} - G_{i,j+1}\right) + 2\left(2R_{i,j} - R_{i,j-2} - R_{i,j+2}\right)$$  (4)

$$\Delta V_{ij} = \left(G_{i-1,j} - G_{i+1,j}\right) + 2\left(2R_{i,j} - R_{i-2,j} - R_{i+2,j}\right)$$  (5)

Where $R_{m,n}$ and $G_{m,n}$ denote the known red and green CFA components at $(m, n)$ position. Based on the values of $\Delta H_{ij}$ and $\Delta V_{ij}$, the missing green component $g_{ij}$ in Fig. 1(a) is interpolated as follows:

$$g_{ij} = \left(G_{i,j-1} + G_{i,j+1}\right)/2 + \left(2R_{i,j} - R_{i,j-2} - R_{i,j+2}\right)/4 \quad \text{if} \quad \Delta H_{ij} < \Delta V_{ij}$$  (6)
\[ g_{ij} = \frac{G_{i-1,j} + G_{i+1,j}}{2} + \frac{(2R_{i,j} - R_{i-2,j} - R_{i+2,j})}{4}, \text{ If } \Delta H_{ij} > \Delta V_{ij} \]  
(7)

\[ g_{ij} = \frac{(G_{i-1,j} + G_{i+1,j} + G_{i,j-1} + G_{i,j+1})}{4} + \frac{(4R_{i,j} - R_{i-2,j} - R_{i+2,j} - R_{i,j-2} - R_{i,j+2})}{8}, \text{ If } \Delta H_{ij} = \Delta V_{ij} \]  
(8)

The ACPI algorithm has the better performance in dealing with color artifacts, and lower difficulty in implementation with real-time hardware system. This algorithm is applicable for this system.

5. EXPERIMENT RESULTS

This CCD camera works under +28 V, and the current is 0.3 A. The photo is taken by this system is in Fig.9.

(a)  
(b)  
Fig.9 The image from this CCD camera

6. CONCLUSIONS

A field-programmable gate array (FPGA)-based hardware architecture for high-speed camera which has fast auto-exposure control and colour filter array (CFA) demosaicing is proposed in this paper. And the result of this system show that this camera can get a good quality of photo image.

REFERENCES


