ISSN: 2664-9640

Moving object detection based on Zynq

Shuai Zhang[1], Qingsong Lin[2], Cunling Liu[2], and Hongze Yu[2]

[1]School of Information Engineering, Henan University of Science and Technology, Luoyang 471000, China;

[2]School of Information Engineering, Henan University of Science and Technology, Luoyang 471000, China;

Abstract

Moving target detection is an extremely important part of computer vision research, widely used in civil fields, industrial fields, military science and technology. With the development of imaging technology, moving target detection technology is limited by speed bottleneck. At present, the embedded hardware using PC and traditional serial processing can't solve the above problems, so a moving target detection system based on Zynq is proposed. A real-time image processing system is built on Zynq, and the inter frame difference algorithm is transplanted into the system. The system uses the idea of hardware and software co-design and modular design on Zynq to realize image acquisition, image preprocessing, frame difference calculation, HDMI display and other functions. The test shows that the system has the advantages of low power consumption, good real-time performance, and has a good application value.

Keywords

Moving target detection; Zynq; inter frame difference method.

1. Introduction

In this age of information overload, it's not enough to rely on people to process it. In this case, computer vision came into being. Moving object detection is a very important part of computer vision research [1], and it is the preliminary work of more advanced visual processing such as target recognition, target tracking, behavior analysis and scene understanding. Due to the development of imaging technology, the video is also improving in two directions of HD and high refresh rate, which greatly increases the amount of image data to be calculated. Due to the large amount of video image data, complex image processing algorithm and huge amount of calculation, moving target detection technology is limited by the speed bottleneck [2]. These are a big test for the platform computing power of the algorithm and the real-time performance of the algorithm itself.

At present, most moving target detection algorithms are based on PC, but PC has some disadvantages such as large volume, high power consumption and poor real-time performance [3]. Compared with the traditional sequential processing of graphic information by software, the hardware that can process in parallel can cope with the current situation of high-speed data transmission. With the development of micro-electronic information technology, the image processing system which uses special chip or large-scale integrated circuit to replace computer is getting more and more attention. Field Programmable Gate Array (FPGA) chip, with its advantages of high integration, fast parallel operation speed and low cost, is especially suitable for pixel-level operation in video image processing and can cope with the huge amount of data in video image processing. It is more and more used in video image processing. However, due to the poor flexibility of FPGA, some complex algorithms are too complicated and even difficult to be implemented on FPGA. ARM architecture has high flexibility and strong control

ISSN: 2664-9640

ability, and can run complex algorithms. However, due to the disadvantages of serial processing, it cannot meet the requirements of real-time processing due to the huge amount of data in video image processing. Zynq-7000 SOC(System on Chip), as a hardware platform for embedded System design, has the advantages of FPGA on PL side and ARM on PS side. It can not only meet the requirements of information interaction, fast data transmission and fast calculation among embedded devices. Compared with traditional embedded system hardware, it has obvious advantages in performance, cost and power consumption. Zynq-7000 device combines the latest semiconductor technology, computer technology and electronic technology [4], is a new generation of Programmable SoC platform created by Xilinx and ARM. By embedding dual-core Cortex-A9 processor into FPGA chip, the embedded platform has undergone a new revolution.

2. Overall system design

This system uses Xilinx company zynq-7020 development board, OV5640 camera, HDMI display to build a moving target detection system. The Zynq-7020 combines a dual-core ARM Cortex-A9 processor and a traditional live field programmable gate array (FPGA) logic unit, which can be both software and hardware programmable, making it fully programmable for both hardware and software. Therefore, the software and hardware co-design method can be adopted in Zynq development. The design of a system architecture usually takes a top-down approach [5]. Define the interfaces and parameters at the top level, and then the subsystems or functions at the bottom level. Because of the complexity of the design, some subsystems can be decomposed at a lower level. After the function unit is defined, the function is appropriately allocated to the software and hardware, and the system defines the necessary communication between each function module. In general, software (PS side) is suitable for some general sequential tasks, such as operating system, user application [6], PL side is more suitable for data flow calculation and other tasks. In addition, some software algorithms with parallel limitations can also be implemented on the PL side. This is similar to a coprocessor in that it frees the processor from heavy computing and parallelism to hardware processing, thus improving overall system performance.

The system is divided into software and hardware parts by using the idea of hardware and software co-design and modular design. Part PL mainly completes the image acquisition module, image processing module and image display module, which is achieved by Vivado 2018.3. The PS part completes the initialization of the camera, the allocation of memory, the control of DDR3, the invocation of each module, the driver of each IP and the general peripheral interface, which is realized by Software Development Kit (SDK).

2.1. Hardware platform construction

Vivado 2018.3 provided by Xilinx company was used to create the project, and IP such as Zynq and VDMA was configured to complete the construction of the hardware platform. Use IP integrators to create processing systems, Add ZYNQ7 Processing System, VDMA, Video in to AXI4-stream, Video Timing Controller, and Video Out IP addresses to the design and configure the parameters of each IP address.

2.2. Image acquisition and display system based on Zynq

2.2.1. Image collection

The OV5640 sensor output is RGB565 format, each pixel is 16 bits, the OV5640 output data on the rising edge of each PLCK is 8 bits, so the two clock cycles of the camera input interface are combined into one data [7]. And the subsequent image processing needs to be RGB888 format, using the way of low fill 0 to complete RGB565 to RGB88 data conversion. After each register configuration, the maximum time delay of 300ms is needed, that is, the time of 10 frames of

ISSN: 2664-9640

image output to output stable video stream. Therefore, the module counts frames and waits for 10 frames of data before starting data collection.

2.2.2. Image display

HDMI is backward compatible with DVI (digital video interface). In this module, driver logic of DVI interface is implemented, TMDS (Transition Minimized Differential Signaling) standard is used in interface protocol in physical layer transmission of audio and video data [8]. The TMDS connection used in DVI and HDMI video transmission is realized through four serial channels, among which three channels are respectively used to transmit the red, green and blue color components of each pixel in the video, and the fourth channel is the clock channel used to transmit the pixel clock. TMDS connections can be logically divided into two stages: encoding and concatenation. Therefore, the module is also divided into two sub-modules encoding and parallel conversion.

Encoder module is responsible for encoding data, Serializer module is responsible for serial conversion of encoded data, and finally OBUFDS into TMDS differential signal transmission. Encoder module uses TMDS coding algorithm in DVI specification to encode. The Serializer module converts encoded data in tandem. The Serializer module calls OSERDESE2, a dedicated hardware resource in series 7 FPGA, that is, the dedicated Serializer converter.

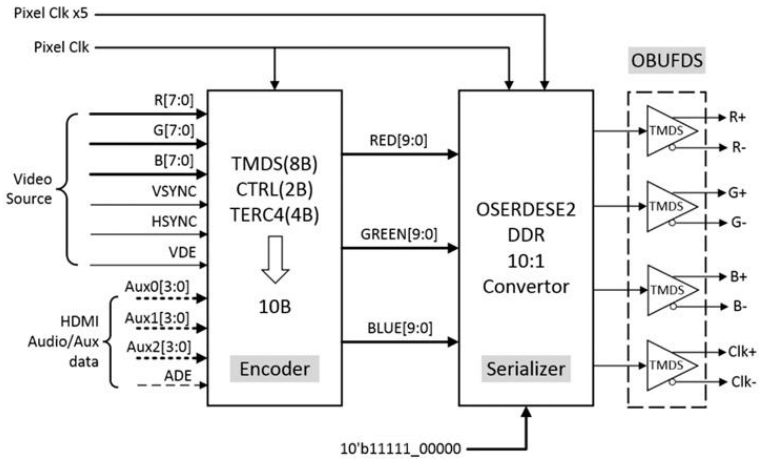

Fig. 1 Block diagram of RGB2DVI module

3. Zynq implementation of inter-frame difference method

3.1. Gray scale transformation

Under the action of the image acquisition module, the video images collected are in RGB888 format. In order to reduce the amount of computation and speed up the detection of moving targets, it is necessary to grayscale the video images collected [9]. Firstly, the color space conversion from RGB888 to YCbCr is completed, and the brightness component Y value in YCbCr is taken as the gray value.

The formula for RGB to YCbCr is as follows.

$$\begin{cases} Y = (77*R+150*G+29*B) >> 8 \\ Cb = (-43*R-85*G+128*B+32768) >> 8 \\ Cr = (128*R-107*G-21*B+32768) >> 8 \end{cases} \quad (1)$$

In the use of Verilog, the above formula is completed in three steps. The first step is to calculate the multiplication terms in brackets, the second step is to add the terms in brackets, and the third step is to move the calculation results to the right 8 bits. The whole process consumes three clock cycles, so it needs to delay 3 beats to synchronize data signals.

ISSN: 2664-9640

3.2. Median filtering

Median filtering algorithm can be implemented by bubble sorting method, selection sorting method and other multi-median methods, in order to find the median of nine pixel values in early neighborhood. However, using Verilog to achieve the above sorting algorithm will be very complex and the operation efficiency is not high, so the pipeline operation is selected to achieve median filtering [10].

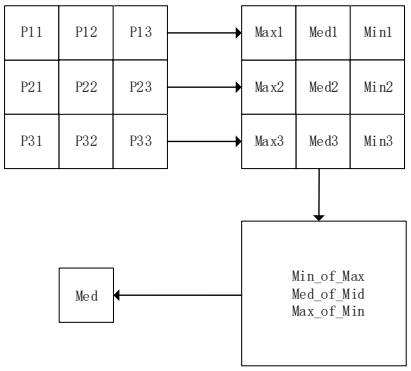

Fig. 2 Block diagram of median filtering algorithm

First, a 3x3 pixel array is generated, and then the three pixels of each row are sorted separately to get the maximum, intermediate and minimum values of each row. Next, extract the minimum of the three maximum values, the middle of the three intermediate values, and the maximum of the three minimum values. Finally, sort the three extracted values, take the intermediate value again, and use this value to replace the pixel value of the point.

In order to generate a 3x3 filtering template, RAM is used to store the data of the first two lines of the image, and the current input data is taken as the third line. When the third row of data is reached, the line_shift_RAM_8bit module is called to read the first two rows stored in RAM, thus obtaining a three rows and one column of pixel data.

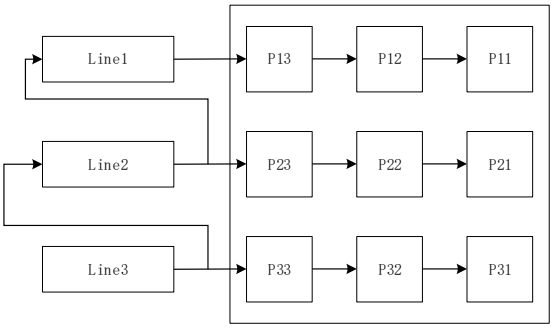

Fig. 3 module Schematic diagram of 3×3 matrix generation

3.3. Frame difference operation

The n and n-1 frames of the video image sequence are denoted as f_n and f_{n-1}, and the gray values of the pixels corresponding to the two frames are denoted as f_n (x,y) and f_{n-1} (x,y).Make the difference between the gray values of the pixels of the two adjacent frames and take their absolute values to obtain the difference image D_n:

$$D_n(x,y) = | f_n(x,y) - f_{n-1}(x,y) | \quad (2)$$

Frame Buffer is used to avoid the problem of rate mismatch between the input and output of the image. Since the memory capacity required to store a frame is larger than the on-chip BRAM, the external DDR3 memory is selected as the frame cache. Since DDR3 is the PS part of the storage interface, PL logic needs to access DDR3 through the AXI interface, so the system calls

ISSN: 2664-9640

the VDMA IP core to complete the frame cache. Set VDMA to Dynamic Genlock mode and frame cache to 3.

This system calls two VDMA, among which VDMA0's function is to output the processed image to the HDMI output module to solve the problem of the mismatch between the input and display device transmission rate, VDMA1's function is to cache the last frame data for the realization of the frame difference operation.

Frame difference calculation requires the pixel gray values of the same pixel coordinates of two adjacent frames to make difference [11]. In order to synchronize two frames of data, that is, pixel alignment (at the same time, the corresponding two pixels are always the same pixel coordinates), the module instantiates a FIFO (First In First Out) to write data from VDMA1, And when the start signal of the next frame arrives, it is read out of the FIFO to do frame difference calculation. When the FIFO is added, the FIFO is set to a synchronous FIFO and two signals, almost empty and almost full, are added to indicate that the FIFO is about to be empty and full, respectively. Write Verilog code to operate the read port and write port of FIFO to achieve data caching.

3.4. Binarization

The threshold T is set and the binarization processing is carried out on the pixel points after difference one by one to obtain the binarization image Rn. Where, the gray value of the foreground image is set to 255, and the gray value of the background image is set to 0.

$$R_n(x,y)=\begin{cases}255, D_{\mathrm{n}}(x,y)>T\\ 0, else\end{cases} \tag{3}$$

Manually set a parameter named THRESHOLD as the THRESHOLD for binarization and set a flag bit. When the gray value of the pixel is less than the threshold, the flag bit is set to 1; otherwise, it is set to zero. When I take the absolute value of the result of the frame difference operation.

Verilog cannot directly use the sign of absolute value to obtain absolute value. Instead, it adopts the method of conditional judgment, that is, first judge the size of two numbers, subtract the smaller number from the larger one, so as to avoid the problem of taking absolute value after negative number.

3.5. Morphological Filtering

The corrosion algorithm implementation requires a call to the previously completed Shift_RAM IP to generate a 3×3 sliding window. Carry out "and" operation on the nine values in the sliding window. Only when all the nine values are 1, the point of the output image can be outputed as 1; otherwise, it is 0. Verilog is used to implement the algorithm. In the first clock cycle, the three values of each row of the 3×3 matrix are summed and calculated to get three intermediate values. In the second clock cycle, the three intermediate values are summed and calculated again, and the final value is taken as the value of the point. The whole process consumes two clock cycles, and the synchronization signal is dealt with by two beats.

$$P=P_{11}\,\&\,P_{12}\,\&\,P_{13}\,\&\,P_{21}\,\&\,P_{22}\,\&\,P_{23}\,\&\,P_{31}\,\&\,P_{32}\,\&\,P_{33} \tag{4}$$

The implementation process of the inflation algorithm is similar to that of the corrosion algorithm. Shift_RAM IP is also called to generate a 3×3 matrix. It simply changes the "and" of the corrosion algorithm to "or". That is, nine values in the field are "or", as long as there is one of the nine values, the point output is 1. Consumption of two clock cycles, the synchronization signal to do two beat processing.

ISSN: 2664-9640

3.6. Obtain the position information of the moving target

The pixel coordinates of the moving target are obtained by counting the frame start signal and line end signal. All Xilinx IP related to video processing follows AXI Stream video streaming protocol. And tlast and tuser are given special meanings, where tuser represents the start of frame signal (SOF, start of frame) and tlast represents the end of line signal (EOL, end of line). Both signals are accompanied by pixel data transmission. Only one valid transmission time is maintained, i.e. the frame start signal is transmitted with the first pixel of a frame and the line end signal is transmitted with the last pixel of a row. Specifically, x_cnt and y_cnt are defined to count the x and Y directions respectively with a reg type variable. When the system is reset, initialize both variables to 0. When s0_axis_tvalid_dly comes, check whether the end of the line signal has arrived. If it arrives, the end of the line is indicated. Clear x_cnt and increment y_cnt. At the same time, if the start signal of the frame arrives, it indicates the arrival of a new frame data, x_cnt and y_cnt will be cleared. By doing this, the pixel coordinates after the frame difference are obtained.

Judge the distance between moving targets by judgment. When the distance is greater than a certain range, it is identified as another moving target. Manually set a parameter named MIN_DIST to determine if it is a moving object. First of all, if the upper, lower, left and right boundaries of each moving target plus the minimum spacing is not greater than the whole frame image, the upper, left and right boundaries are diffused to the minimum spacing around, which is called the neighborhood of the moving target. When the reset signal and frame start signal arrive, the moving target position parameter is initialized. Detection and marking takes two clock cycles. In the first clock cycle, the pixels marked as moving targets are found, and the elements in the moving target list are voted to judge whether it is a new moving target. If the data in the moving target list is invalid.

The element votes the input data as the new target; If the data in the moving target list is valid, it will judge whether the current pixel is located in the neighborhood of the element. If the coordinates of the point are beyond the scope of the element field, the element will vote to identify the pixel point as the new moving target. In the second clock cycle, candidate data is updated to the list of moving targets based on the voting results. If all votes are passed, a new moving target is judged to appear, and the current value of x and y is assigned to the new moving target positioning register. If pixels marked as moving objects appear, but fall within the neighborhood of an element in the moving objects, the list of moving objects is traversed to expand the boundary information of those elements not voted as new objects. After a frame of data statistics is finished, the location information of each moving target is recorded and output.

3.7. Adding an Enclosure Box

First, the line end signal of the frame start signal of the input signal is counted and its horizontal and vertical coordinates are obtained. Two clock cycles are required for detection and marking. The first clock cycle determines whether the current pixel is located on the border. In the second clock cycle, judge whether the frame is within the range of other moving targets. If so, it indicates that the frame overlaps, and the internal frame will no longer be displayed. Only the pixel coordinates of the point are on the border and not within the range of other moving targets will be displayed as the specified border color, otherwise the output is the original image data.

3.8. Software

After completing the hardware design, choose File > Export > Export Hardware on the menu bar to Export the hardware, and select Include bitstream files in the pop-up dialog box. Choose File > Launch SDK from the menu bar to Launch the SDK software, create an application project, and enter the design of the software part.

ISSN: 2664-9640

3.8.1. Initializing the OV5640 Camera

Before OV5640 can work normally, it is necessary to initialize the sensor, that is, to make it work in the expected working mode through the SCCB (Serial Camera Control Bus) Bus configuration register. Including OV5640 output format, resolution, pixel PCLK clock frequency and so on, and get better picture quality image. Since OV5640 initialization only needs to be performed once after power-on, this module is completed by the PS terminal.

PS side through the SCCB bus to configure the OV5640 camera register, so that its output pixel format is RGB565 format, and through the relevant parameters to adjust the resolution of the output image. SCCB ports are connected to PS through EMIO, and PS drives SCCB. The write transmission protocol of SCCB bus is almost the same as that of IIC, and the driver of IIC can be used to configure the camera directly. The EMIO is initialized, the SCCB port is set to output, the output is enabled, and the SCL and SDA of the SCCB are increased. Write corresponding functions to realize SCCB start signal, stop signal, send a byte, accept a byte, generate a reply signal. For example, the generation process of the initial signal of SCCB is as follows: Both SCL and SDA of SCCB are raised and delayed by 4us; SDA is pulled down and delayed by 4us, SCL is pulled down. These functions are called to form the functions of read and write operations on the SCCB bus.

The OV5640 can adjust the resolution of the output image. Inside the ov5640_init function, the first delay is 20ms (at least 20ms from the time when the OV5640 is powered on to the time when the SCCB configuration starts). Read register addresses 0x300B and 0x300A to obtain the camera ID of OV5640. If the obtained ID is not equal to 0x5640, return 1. Otherwise, the register input software is reset first to restore the initial value of the register. After the register software is reset, it needs to delay for 1ms to configure other registers. The register whose address is 0x3008 is written to 0x82 to complete the software reset and set the power sleep mode to normal mode. After 1ms delay, the register at address 0x3008 is written to 0x02 to restore it to normal mode and start to write corresponding mail

The OV5640 camera is configured in memory mode, and the final return value is 0.

3.8.2. VDMA configuration

Configure VDMA with the run_vdMA_frame_buffer () function, which comes from Xilinx's official VDMA template. Configuration includes frame cache address, read channel, read channel, and so on. The last parameter of the run_vdma_frame_buffer() function is ONLY_READ, which indicates that only read channels are enabled. ONLY_WRITE Indicates that only the write channel is enabled. BOTH indicates that BOTH the read and write channels are enabled. This system uses this function to configure VDMA0 and VDMA1.

```
int run_vdma_frame_buffer(XAxiVdma* InstancePtr, int DeviceId, int hsize,
        int vsize, int buf_base_addr, int number_frame_count,
        int enable_frm_cnt_intr,vdma_run_mode mode)
```

Fig. 4 Run_vdma_frame_buffer () function

3.8.3. Displaying initialization cores of related IP cores

In the hardware part, the dynamic clock controller AXI_dynclk IP core is added. The IP core is from the open source document of Digilent. Here, the C source code file for configuring the IP core is manually added (the C source code file is from the open source file), so as to control the IP core to generate clock signals of different frequencies. Includes VTC and dynamic clock configurations; The Display Set Mode function sets the resolution of VTC output; Finally, the Display Start() function starts VTC to work.

ISSN: 2664-9640

4. System performance test and analysis

4.1. Resource Utilization

After completing the PL section, Vivado reported resource utilization, as shown in the figure, which was no more than 25% for each resource, meaning that a significant amount of resources could be used to improve and expand the system.

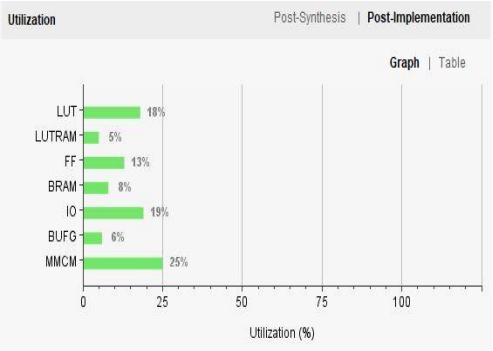

Fig. 5 Resource utilization

4.2. Power Consumption Evaluation

Similarly, Vivado evaluated power consumption, as shown here. The power is only 2.022W, which meets the requirements of low power consumption.

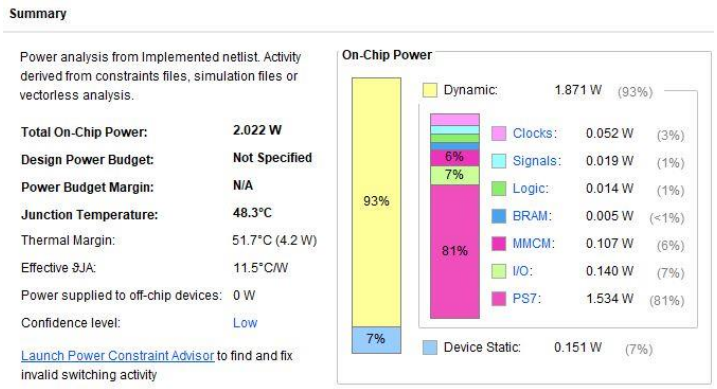

Fig 6 Power consumption assessment

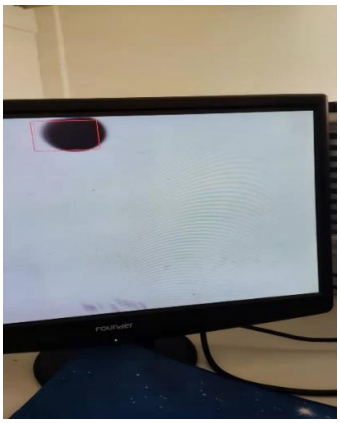

Fig. 7a

Fig. 7b

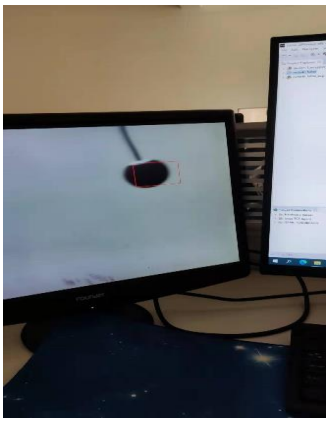

Fig 7c

4.3. Analysis of test results

Download software programs and bitstream files to the development board for board-level debugging in the SDK. The test results show that the system can detect moving objects. The results show that the system can detect single moving target and multiple moving targets in

ISSN: 2664-9640

real time with a resolution of 640*480, and the system has good robustness and strong adaptability to the environment.

5. Summarize

In this paper, the idea of hardware and software co-design gives full play to the advantages of fast FPGA speed and high flexibility of ARM, and adopts the method of hard part algorithm acceleration to design a real-time moving target detection system with good real-time performance, small volume and low power consumption.

References

[1] Yi Luo: Research on moving target detection and tracking algorithm(Master's Degree, Southeast University, China 2017).p. 1.
[2] Dongwei Wang Design of binocular stereo vision image acquisition and processing system based on FPGA. Aviation precision manufacturing technology.Vol.54 (2018) No. 1, p. 25-28.
[3] Yubin Deng: Research on Moving Target Detection Technology based on ZYNQ (Master's Degree, Nanjing University of Science and Technology, China 2018).p. 2.
[4] Jingjia Tan, Lesheng He, Jun Wang, et al. Design of AXI DMA data transmission and display based on Zedboard platform. TV technology. Vol 42 (2018) No. 06, p. 41-45.
[5] Xuliang Wu, Huifeng Yue, Fenli Zhou, et al. Integrated control combination based on ZYNQ platform. The electronic world. Vol.54 (2018). No. 5, p. 77-88.
[6] Information on: https://blog.csdn.net/xh116996/article/details/89218766.
[7] Song Wu, Lihau Zhang, Design and implementation of image processing system based on FPGA. Radiocommunication technology. Vol.44 (2018), No. 4, p. 407-410.
[8] Qingbo Ji et al. Real-time embedded object detection and tracking system in Zynq SoC. EURASIP Journal on Image and Video Processing, Vol. 1 (2021), No 1.
[9] Touzani Hajar et al. Co-design implementation of High Efficiency Video Coding standard encoder on Zynq MPSoC, International Journal of Circuit Theory and Applications, Vol. 49 (2021), No. 4, p. 1013-1027.
[10] Mengxue Sheng, Wanwan Hou, Juchao Jiang, et al. Implementation of Accelerating Video Preprocessing based on ZYNQ Platform Resource Management. Journal of Physics Conference Series, Vol.1544 (2020), No. 01, p. 012112.
[11] Liang Jia, Chen Wu. Target detection based on inter-frame difference method and FPGA implementation, Computers and information technology, Vol.29 (2021), No. 02, p. 20-23.