IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 23 March 2024, accepted 22 April 2024, date of publication 30 April 2024, date of current version 7 May 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3395502

RESEARCH ARTICLE

A Custom RISC-V Based SOC Chip for Commodity Barcode Identification

SIJIE LIN, RENPING WANG, TING CAI, AND YUNZE ZENG
College of Physics and Information Engineering, Fuzhou University, Fuzhou, Fujian 350108, China
Corresponding author: Renping Wang (rpwang@fzu.edu.cn)

This work was supported in part by the Natural Science Foundation of Fujian Province under Grant 2023J01398.

ABSTRACT The Internet of Things (IoT) is a crucial component of the contemporary information industry and represents a significant advancement in information technology aimed at enhancing both human productivity and daily existence. Their applications are extensive and far-reaching. However, the present state of research on the design of low-cost IoT SoC chips leveraging open-source instruction set architectures lacks the requisite depth and breadth. To meet the requirements of low-cost IoT system-on-chip (SoC) development, this study presents a commodity code recognition SoC chip based on the RISC-V instruction set architecture, which is capable of performing image acquisition and barcode recognition. The proposed system comprises two main components: a low-power RISC-V processor and an image recognition module. This study initially enhanced the speed, accuracy, and area efficiency of the hardware design of a commodity barcode image-recognition module. Subsequently, the image recognition control module was developed using the RISC-V processor and CMOS image sensor OV7670, and the outcomes of image recognition were accessed through interrupts. The processing speed for collecting and identifying 640 × 480 images on the FPGA board was 11.4FPS, and the image recognition rate was 99.5%. The chip was taped-out using the UMC55n process, which successfully decoded the barcodes and output the results at a working frequency of 40 MHz.

INDEX TERMS Barcode, MPW, RISC-V, SoC chip.

I. INTRODUCTION

With the advancement of very large-scale integration (VLSI), integrated circuits (IC) are progressively transitioning towards Integrated Systems (IS), heralding the emergence of the SoC paradigm. SoC denotes a comprehensive system or product composed of amalgamating multiple integrated circuits with distinct functions onto a single chip, encompassing complete hardware systems along with their embedded software. Embedded platforms such as ARM and DSP cannot satisfy the requirements of fast use, owing to their limited storage capacity, low processing speed, and poor scalability [1]. With the development of the Internet of Things and information processing technology, system-on-chip with microprocessors as the core, with the advantages of low cost, small size, high integration, and low power consumption, has gradually become the mainstream of today's integrated circuits, providing a broad market and development opportunities for the integrated circuit industry [2].

These advantages are combined in different fields [3] using SoC chips to provide a good solution for the implementation of data information systems that require high real-time performance and a fast processing speed [4]. IoT devices typically consist of one or more SoCs. In the market, most electronic products are based on processors primarily from Intel (x86 architecture) and ARM. Intel's x86 instruction set architecture dominates in desktop PCs and server domains, whereas ARM instruction set architecture stands out in embedded mobile handheld devices and similar fields. However, after years of development, traditional x86 or ARM instruction set architectures have become exceedingly complex, with a growing number of instructions, thus increasing the difficulty for developers to learn and utilize them. Moreover, the utilization costs of x86 or ARM instruction set architectures are prohibitively high, far exceeding the economic feasibility of ordinary companies. In recent years, the emergence of

The associate editor coordinating the review of this manuscript and approving it for publication was Libo Huang.

open-source hardware represented by RISC-V has overcome the limitations imposed by traditional hardware architecture licensing, further fostering innovation and development in IoT devices [5]. The traditional x86 architecture and ARM instruction sets have become extremely complex after years of development, and contain an increasing number of instruction sets, which significantly increases the difficulty of learning and use. In contrast, the RISC-V architecture has the advantages of instruction set reduction, modularity, open source, scalability, and arbitrary portability of calling soft cores [6], resulting in low cost and a low technical threshold to meet user customization needs [7], which meets the processor requirements of the Internet of Things (IOT) [8] and embedded systems [9].

Currently, common commodity code identification devices typically use a laser scanner to obtain the level signal corresponding to the barcode through linear scanning. It is necessary for people to manually perpendicular the commodity barcode to the scanner to correctly read the commodity barcode information and then decode it using a computer [10] In contrast, there are already barcode recognition devices based on image processing in the market that can identify barcodes with quality defects such as tilts and folds, but they are always expensive. Therefore, it is of great commercial value to research and develop cost-effective barcode recognition products based on image processing [11]. With the popularity of barcodes as tagging systems, significant prior work has been conducted on reading barcodes using computer vision-based methods [12].

The implementation of RISC-V SoC has realized practical IP (Intellectual Property) sharing, fostering the establishment and development of hardware open-source technology platforms and shared open-source ecosystems. Currently, most academic research on processors focuses primarily on the processors themselves, but there is still insufficient research on the application of the entire RISC-V processor in the IoT field. RISC-V, as an open instruction set architecture, has also shown tremendous potential in the IoT domain. Reference [13] delves into the challenges faced in developing SoCs for IoT edge devices, proposing a security and privacy-based architecture, which provides valuable insights for the development of IoT edge systems. Reference [14] introduces a refined RISC-V Instruction Set Architecture (ISA) extension scheme for efficient FFT (Fast Fourier Transform) operations. This research further expands the application of RISC-V in the IoT domain, offering new possibilities for performance optimization and feature enhancement of IoT edge devices. Additionally, a study [15] designed a novel RISC-V compatible coprocessor based on the SM2 and SM3 algorithms, representing a significant advancement in RISC-V development.

It can be observed that significant progress has been made in the design of hardware acceleration modules on the RISC-V platform [16]. With improvements in chip integration, integrated circuit chips for digital signal processing are no longer simple dedicated hardware circuits but are combined with special processing modules and CPUs to complete more complex tasks through software and hardware collaboration. However, there is still some deficiency in research related to software-hardware co-design. The approach of software-hardware co-design approach can fully utilize existing software and hardware resources, maximizing development efficiency [17]. Numerous studies combining software and hardware have been conducted. For instance, [18] proposes a brain-inspired Hierarchical Interactive Memory Computing (IMC) system aimed at addressing computational efficiency and performance issues in video emotion analysis. This system employs a software-hardware co-design approach to achieve emotion analysis of multimodal information through efficient modal interaction and semantic elimination. Additionally, [13] introduces a lightweight Continuous Monitoring System (CMS) based on software-hardware co-design, which can communicate with RISC-V to identify any anomalies in its operational behavior.

The advantages of the RISC-V architecture in terms of low cost and low barriers have facilitated its application in various fields such as the Internet of Things (IoT), mobile terminals, and edge computing [19]. However, the application of the RISC-V instruction set architecture in the embedded domain is insufficient, and there is still a lack of integration between hardware and software in the relevant research. To leverage the advantages of RISC-V in terms of power consumption and cost, enrich the ecosystem of RISC-V, and expand its application in the embedded domain, this study focuses on the application of barcode recognition. A low-power, real-time barcode recognition system was designed based on the RISC-V instruction set architecture processor. It collects and recognizes CMOS image sensor data, with the recognition results transmitted via a serial port to a PC. The operation of this barcode recognition system is simple and conducive to software debugging.

II. OVERALL SOC CHIP DESIGN

Based on the RISC-V core with commonly used peripheral modules, such as GPIO, UART, and SPI, an image recognition control peripheral is added to the commodity code recognition SoC chip module to complete the reset and enable the image recognition module. The system architecture of the commodity code recognition SoC chip, comprising a low-power RISC-V processor and an image recognition module, is illustrated in Figure 1. The system operates at a clock frequency of 40 MHz, considering the balance between the performance and power consumption. Building upon the RISC-V processor, the image-control module is designed to handle the reset and enablement of the image recognition module, as well as to retrieve results, such as connected domain positioning and decoding, from the RISC-V processor.

The working principle is as follows. After the system is powered on, the MCU system is configured using an image-control module mounted on the ICB bus, and the image sensor register is configured using the camera configuration module

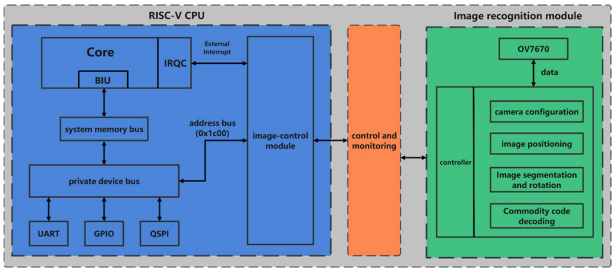

FIGURE 1. System architecture of product code recognition SoC chip.

based on the standard SCCB (Serial Camera Control Bus) protocol to complete the system initialization. The original image data were collected in frames using the image acquisition model, and the data were then written into the storage area while completing image monochromaticization. When the image data acquisition frame was completed, the image acquisition module was stopped. The image acquisition module, image positioning module, image segmentation and rotation module, and commodity code decoding module in this system were executed until the corresponding end signal was sent to the image-control module. The recognition result was transmitted to the RICS-V processor, the acquisition of a new image frame was restarted, and the data flow was executed in this manner.

III. IMAGE PROCESSING SECTION

The image recognition flow is illustrated in Figure 2, and is primarily composed of a localization-cropping module and a rotation-decoding module. After the original image is positioned and segmented, the effective barcode area in the original image can be accurately cut, thereby effectively reducing the data processing of the subsequent rotation angle calculation and improving the calculation accuracy of the decoding operations. To implement image recognition in the hardware, it is necessary to optimize the image algorithm for hardware acceleration. This paper introduces the hardware implementation of image recognition with a focus on optimizing speed, accuracy, and area.

A. AREA OPTIMIZATION

1) EFFICIENT SRAM-BASED IMAGE PROCESSING

In the process of image recognition, there is a need to cache a large amount of image data, which often involves consumption of storage space. This may lead to an increase in chip area and higher costs. In engineering practice, larger blocks of SRAM are commonly preferred over multiple smaller-capacity SRAMs, because the former contributes to a smaller overall chip area. Considering the use of the UMC55n1P6M process in subsequent fabrication, with the Memory Compiler in the process library capable of generating a maximum of 32K $\times$ 8bit SRAM, a thorough assessment of the chip area, processing efficiency, and cache data size influenced the decision to construct the SRAM using groups of 10 chips.

In this design, the image-processing module operates in a serial manner, requiring buffering of the original image and waiting for barcode localization before initiating the recognition processing. To address the potential issue of excessive data retention during the image-processing stage, we propose a method that employs a dual-buffer storage structure, as illustrated in Figure 3. We determined the specific address signals based on the state machine of the image processing module. Because the image-processing module follows a pipeline structure and executes tasks sequentially, careful attention is paid to the constraints on shared resources, which prohibit multiple processes from simultaneously writing data. To address this issue, we devised an arbitration mechanism that utilizes the start and end signals from each module as inputs, facilitating data flow and mitigating potential conflict. The simulation waveform is shown in Figure 4. This design aims to comprehensively consider the efficiency of image processing, utilization of storage space, and overall system performance, thereby providing an effective solution for storage management in image-data processing.

2) EFFICIENT REGISTER-BASED IMAGE PROCESSING

The process of executing digital image processing tasks demands real-time responsiveness, simultaneous reading and writing of multiple data, and the caching of a substantial amount of intermediate data. It is suitable to use registers for storage; however, the excessive use of register resources consumes significant hardware resources, thereby increasing the chip area. To address this issue, this design combines a shared storage group strategy, including edge detection, image dilation, connected-component localization, and barcode decoding, using shared cache space along with a cache arbitration module to reduce the number of registers. As the image-processing workflow in this design operates in a pipelined manner, it establishes a prerequisite for sharing the cache space through time-division multiplexing. By delving into the storage space requirements for caching intermediate data in processes such as edge detection, image dilation, connected-component localization, and barcode decoding, the design optimizes the data storage and access schemes. This optimization is achieved by applying the shared cache space through the cache arbitration module. The structure of the shared cache space is illustrated in Figure 5.

The area sizes before and after optimization were compared using the logic synthesis tool Design Compiler based on the UMC 55 nm process, as shown in Figure 6. The area before optimization was approximately three times larger than the optimized area, measured in square millimeters.

B. SPEED OPTIMIZATION

1) OPTIMIZATION OF PARALLEL STORAGE FOR EDGE DETECTION RESULTS

This design features an 8-bit width for the SRAM. When employing the Sobel operator to detect image edges, the edge detection output for eight pixels is combined and then packed

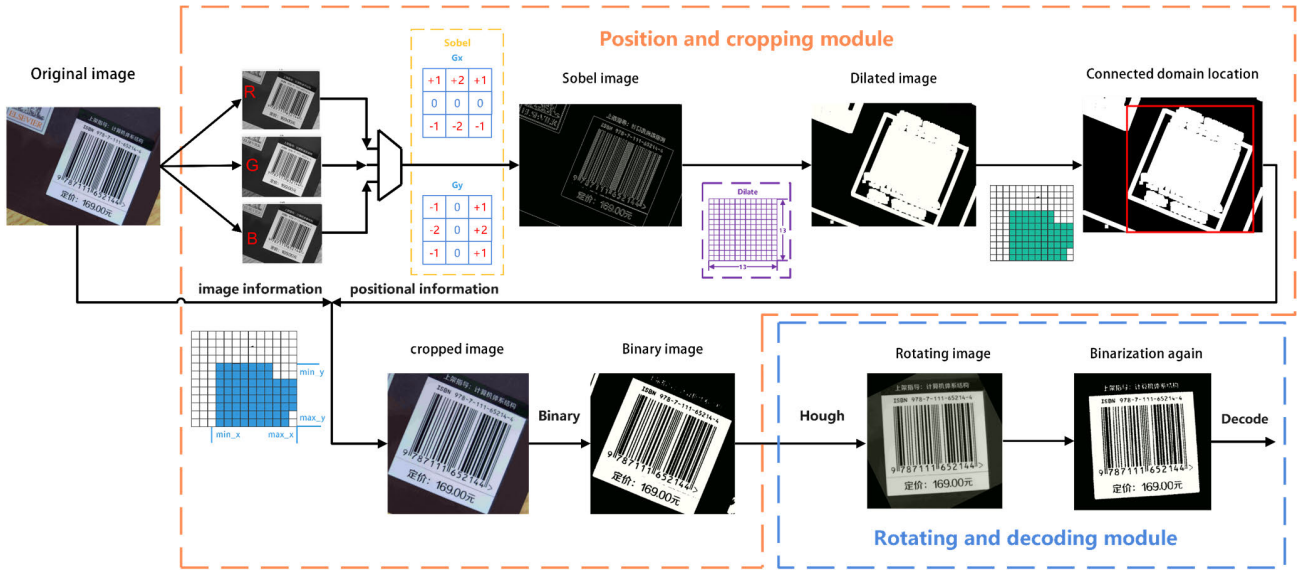

FIGURE 2. System architecture of product code recognition SoC chip.

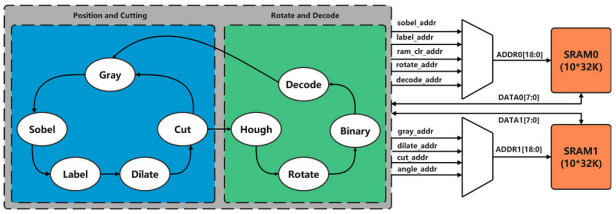

FIGURE 3. Double buffering storage structure.

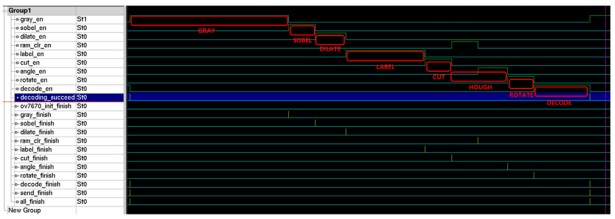

FIGURE 4. State transition waveform diagram of the image processing module.

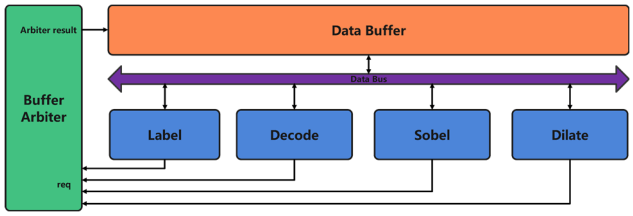

FIGURE 5. Shared cache space structure in image processing module.

into the SRAM. The approach of combining edge detection results through pixel aggregation enhances the effective information density, establishing a prerequisite for efficient implementation of the dilation algorithm.

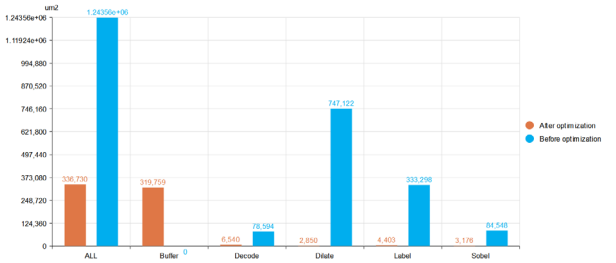

FIGURE 6. Comparison of area sizes before and after optimization. (UMC 55nm Process).

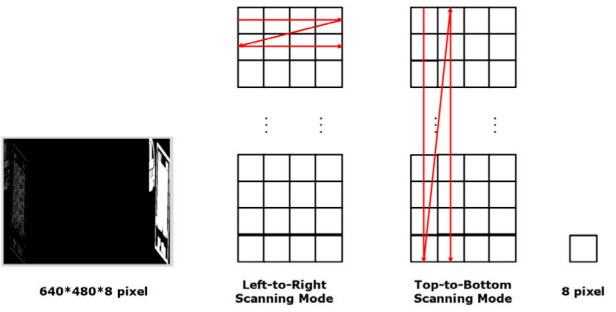

FIGURE 7. Effectiveness of row storage optimization.

The approach of combining edge detection results through pixel aggregation is instrumental in enhancing effective information density. This establishes a crucial prerequisite for efficient implementation of dilation algorithms. The effects of the compressed image are shown in Figure 7. In contrast to the conventional left-to-right and top-to-bottom image processing approaches, in which 560 clock cycles are required to buffer data for 12 rows, this design proposes a top-to-bottom and left-to-right approach. This method requires only

80 clock cycles to buffer the data for a single column, which significantly reduces image processing time.

2) OPTIMIZATION OF PARALLEL STORAGE FOR EDGE DETECTION RESULTS

The foundation of line detection in the Hough transform lies in coordinate system conversion, specifically mapping Cartesian coordinates to polar coordinates. Each Cartesian discrete point corresponds to 180 discrete points in polar coordinates. In this approach, the processing of a single pixel requires 180 clock cycles, which contributes to substantial time consumption during the entire image scanning process. After barcode localization, the crucial information is typically located in the central region of the image. Recognizing the inherent parallel line features of barcodes, this design introduces an innovative optimization strategy to focus line detection efforts solely on the central 40 × 40 region of the cropped barcode image, as illustrated in Figure 8. This not only effectively reduces the computational complexity but also enhances the accuracy of line detection.

FIGURE 8. Untrimmed line detection (Left) Line detection after cropping (Right).

Extensive testing validates the practical effects of this optimization strategy. The average size of the cropped images is 350 × 300 pixels. Theoretically, this results in an average reduction of 103,400 clock cycles. A comparison of the clock cycles before and after optimization for the line detection module is presented in Table 1, demonstrating a significant improvement in performance. This satisfies the requirements of real-time applications measured during the clock cycles.

TABLE 1. Clock cycles used before and after optimization of line detection module.

Average Uncropped Clock Cycles	18900000
Clock Cycles Used After Cropping	288000
Saved Clock Cycles	18612000
Saved Clock Cycles Ratio	98.4%

C. ACCURACY OPTIMIZATION

1) COLLABORATIVE DESIGN OF IMAGE EDGE DETECTION AND CONNECTED COMPONENT LOCALIZATION

The edges exhibit noticeable variations in the intensity of an image. Edges provide boundaries between different areas in an image [20]. Therefore, the edge information of a barcode in an image can be used to determine its position. After the first convolution operation of the Sobel operator, the monochromaticized image data were used again to perform the convolution operation to complete the image dilation to compensate for the loss of image edge information caused by the fixed threshold selection of the Sobel algorithm and complete the merging of adjacent regions. In the ArTe-Lab barcode database, barcode images with tilt, glare, wrinkles, and bends are selected to complete the attempts of convolution kernels of various sizes and shapes. The experimental results show that the 13 × 13 two-dimensional matrix convolution kernel has the highest application rate, and that the effective information, including the barcode position, is the most complete. The experimental results show that the image-positioning segmentation module can complete the successful positioning and cropping of barcodes with tilt, strong light exposure, folds, and bending, as shown in Figure 9.

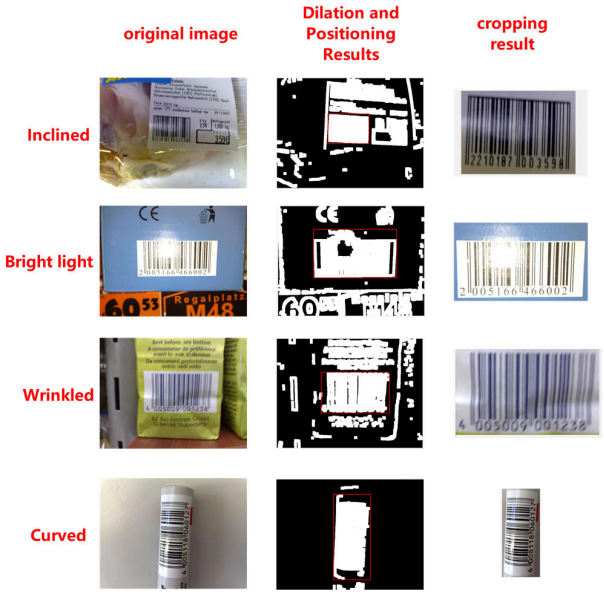

FIGURE 9. The results of localization and cropping for barcodes in low-quality images.

2) DESIGN OF MULTI-THRESHOLD DECODING MODULE

The EAN-13 product code consists of a start symbol, a stop symbol, five standard middle separators, six left-side data symbols, five right-side data symbols, and a checksum symbol, totaling 95 standard bars to form complete data information. To improve the decoding accuracy, optimizations were applied to determine the effective rows for the bar width, calculate the averages, and select the binarization thresholds. Enhancing decoding accuracy, especially in handling barcode defects, involves algorithm improvements that traverse each row of data rather than relying solely on a single row for computation. By scanning the entire image row by row, rows with 60 empty bars were recorded as effective, whereas black

borders introduced by rotation were recording 62 flips as effective rows. This method remains equally applicable in cases of barcode contamination, breakage, or defects because images with barcode defects do not have 60 empty bars of 60.

The fixed-threshold binarization method is simple and requires minimal resource consumption during the hardware implementation. However, it cannot guarantee suitability across various scenarios. To enhance versatility, this design proposes the use of an 8-level threshold for the barcode width calculation and decoding. The flow of the modified algorithm for the decoding module is shown in Figure 10.

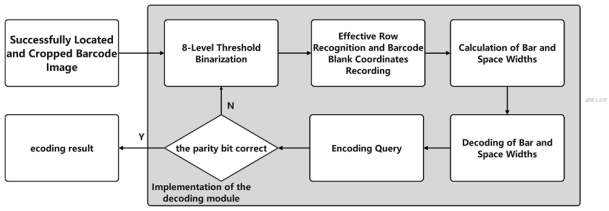

FIGURE 10. Decoding module implementation process flowchart.

Flexible threshold selection enhances adaptability to different scenarios, making the system more robust. The optimal threshold selection for binarization varies across different scenes. The test results, as shown in Figure 11, illustrate our method compared with the fixed-threshold method. Utilizing eight threshold levels for barcode decoding, as opposed to the single-threshold binary method, has proven to be more suitable for barcode recognition in complex scenarios, particularly in environments with intense lighting.

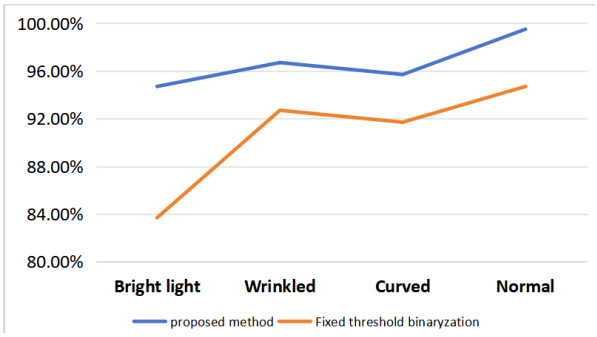

FIGURE 11. The comparison results between multi-threshold and single-threshold methods under different scenarios.

IV. HARDWARE-SOFTWARE CO-DESIGN

A. RISC-V INSTRUCTION SET

The most significant difference in the RISC-V instruction set architecture is modularity. Different instruction modules can be combined by users according to application scenarios, and each module is represented by an English letter. The basic integer instruction set is the most basic and only instruction set that RISC-V must contain, represented by the letter I, which can support 32-bit, 64-bit or 128-bit, and users can also choose to use other instruction sets according to their needs [21]. In this study, an RISC-V core was designed using a 32-bit architecture with 32-bit architecture with 32-bit general-purpose register width. It also supports compressed instructions with 16-bit encoding length. According to the naming rules of RISC-V architecture, the instruction set used in this design can be expressed as RV32EC. The RV32E instruction set is a subset of RV32I, which is specially designed for low-power embedded scenarios. Through this modular feature, the RISC-V can be miniaturized with low power consumption.

B. RISC-V CORE

The pipeline structure of a processor is one of the most basic elements of the processor microarchitecture. Each stage of the pipeline is composed of registers, and the additional stages of the pipeline structure require additional registers. The increase in the number of pipeline stages results in additional overhead for the chip area and power consumption. On the premise that CPU performance is sufficient to meet the small area and low power consumption requirements, the CPU core adopts a two-stage variable-length pipeline architecture, as shown in Figure 12. The second-level pipeline is the execution module (EXU) of the instruction, which is responsible for decoding, execution, memory access, and write-back of the instructions. This can balance the power consumption and performance, thereby realizing a low-power embedded processor core.

C. IMAGE-CONTROL MODULE

The data transmission structure between the image recognition module and the CPU is shown in Figure 13. The CPU sends a write request to the image-control module through the command channel of the ICB bus. After receiving the request, the image-control module waits for and receives the read data from the image recognition module, and then sends the read data back to the CPU through the return channel. For example, the CPU writes the corresponding parameter value into the camera_parm register, and the image recognition module reads the value of the camera_parm register to complete operations such as camera initialization and image acquisition mode configuration. The ICB bus has both high speed and ease of use, its protocol control is simple, and only two independent channels are required to complete data transmission.

V. SYSTEM VERIFICATION

A. VERIFICATION PLATFORM

The system structure designed in this study can be divided into two parts: software debugging, and hardware downloading. The software debugging component was primarily composed of debugging modules, including a software development platform comprising the Hummingbird debugger, OpenOCD, and related SDK installed in the Linux system [22]. The hardware included the RISC-V product code recognition SoC based on the RTL design completed by Vivado in a Windows environment. After synthesis, layout,

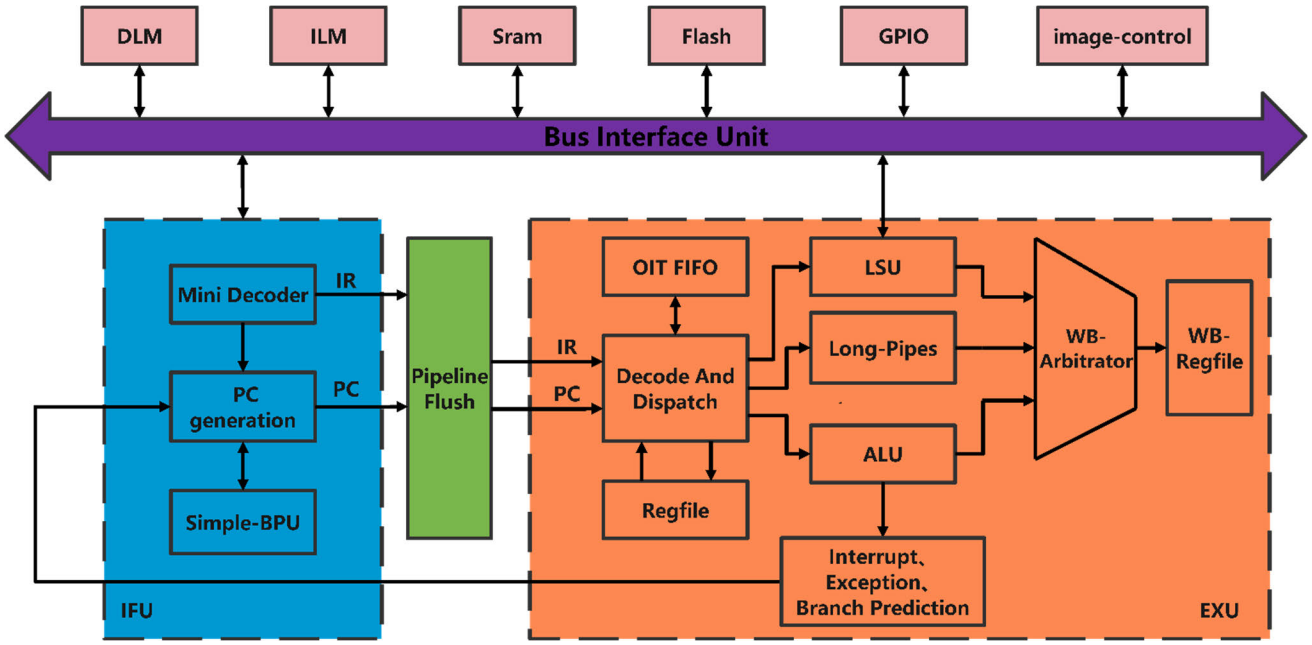

FIGURE 12. The block diagram of RISC-V core.

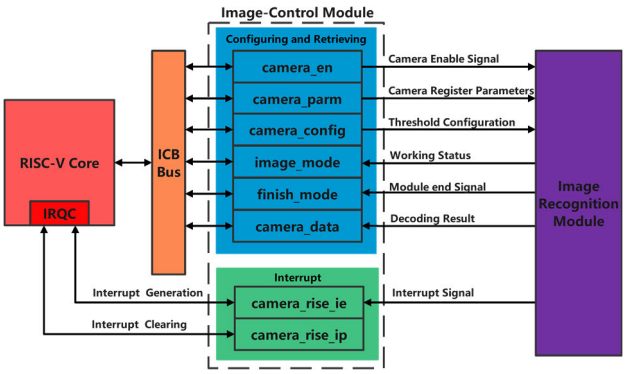

FIGURE 13. The block diagram of image-control module.

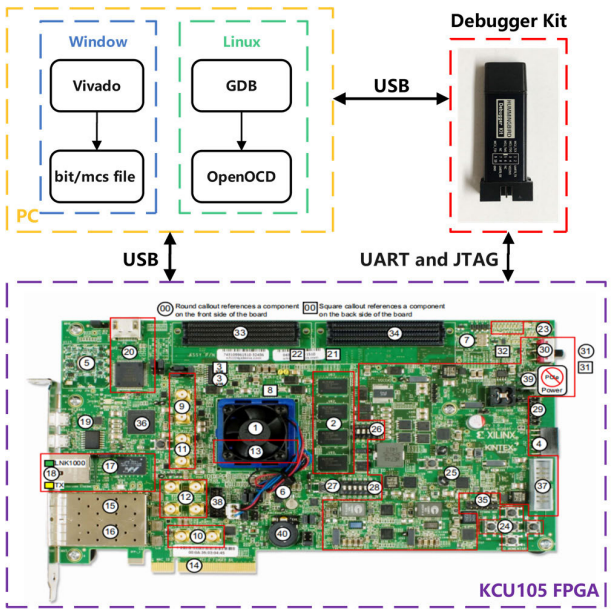

FIGURE 14. System verification platform structure.

and implementation, bit files are generated and burned into the FPGA. The verification structure of this system is shown in Figure 14, which is based on the KCU105 FPGA development board of Xilinx Company for the realization of the system.

B. TEST ENVIRONMENT

Validation was conducted using the Xilinx KCU105 development board, employing Vivado 2018 for synthesis and layout. Resource utilization is outlined in Table 2. The camera data are transmitted to the FPGA through the FMC interface. After data processing is completed in the image processing module, real-time image data are output to the display via the HDMI interface. The HDMI display section primarily consists of the I2C bus selection chip TCA9548 and video codec chip ADV7511. Based on the I2C protocol, after configuring the registers of ADV7511, image output is achieved by configuring the line and frame signals. The final display is realized by connecting the HDMI data transmission lines to the monitor.

The system triggers interruptions based on the decoded completion signal. When a frame of image decoding is completed, the RISC-V Core triggers an interrupt, enters interrupt mode, reads and prints decoding results, connects component localization results, etc., clears the interrupt source, and exits the interrupt. This awaits the next interruption. Real-time signal monitoring of the decoding results using an Integrated

TABLE 2. The utilization of hardware resources on the FPGA development board.

Resource	Utilization	Available	Utilization%
LUT	81407	242400	33.58
LUTRAM	1039	112800	0.92
FF	37345	484800	7.70
BRAM	176	600	29.34
DSP	22	1920	1.15
IO	79	520	15.19

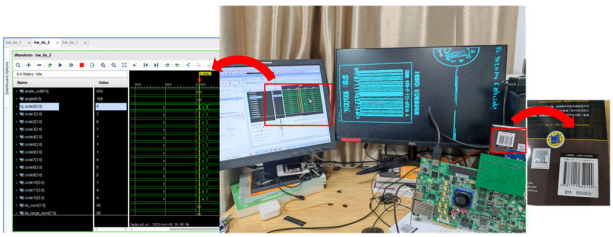

FIGURE 15. Test environment and test results.

Logic Analyzer (ILA) on the FPGA board is shown in Figure 15. Operating at a frequency of 40 MHz, the image processing speed for capturing and recognizing a 640 × 480 image was 11.4FPS, with an image recognition rate of 99.5%.

VI. EVALUATION AFTER LAYOUT DESIGN

This design leverages the Synopsys Design Compiler tool, employing the UMC 55 nm technology 1P6M1T0F process. The entire SoC chip was synthesized at the maximum system clock frequency of 40 MHz. The physical design was carried out using ICC2, and formality verification was performed using the formality tool. The extraction of RC parasitic parameters is accomplished with the StarRC tool, coupled with Prime Time in the DMSA mode to address multiple scenario timing repairs, ensuring timing convergence. The physical verification tasks, including Design Rule Check (DRC) and Layout vs. Schematic (LVS), are executed using Calibre, a tool from Mentor Graphics. Following the successful design stages, the GDS (Graphic Data System) for Multi-Project Wafer (MPW) fabrication is sent to the United Microelectronics Corporation (UMC) in Taiwan, as illustrated in Figure 16(a).

The achieved specifications encompass the following parameters: a working frequency of 40 MHz, a chip area totaling 7330580 square micrometers, and a power consumption of 51.1539 milliwatts. Furthermore, to guarantee circuit stability and reliability, it is crucial to ensure that the cumulative voltage drop and ground voltage rebound do not exceed 5% of the chip's supply voltage, VDD. Following the completion of chip fabrication, we opted for a 40-pin Quad Flat No-leads (QFN) package, illustrated in Figure 16(b).

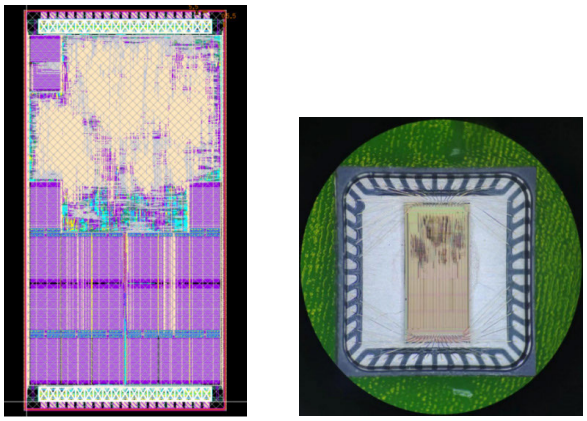

FIGURE 16. Chip layout design and chip packaging. (a) The chip layout. (b) The chip packing.

VII. CONCLUSION

This SoC chip was fabricated using the UMC55nm 1P6M1T0F process for Multi-Project Wafer (MPW) production. It was packaged in a 40-pin quad flat no-lead (QFN) package. After packaging, the chip underwent PCB testing at a frequency of 40 MHz, as depicted in the figure. During testing, the IO interface configuration of the software program was modified and the board-level configuration of OpenOCD in the SDK software development kit was adjusted. The software program was compiled using the RISC-V gcc toolchain and downloaded to the finalized chip via the MCU_JTAG interface. The test results, as shown in Figure 17, indicate correct translation outcomes, affirming the functionality of the SoC chip.

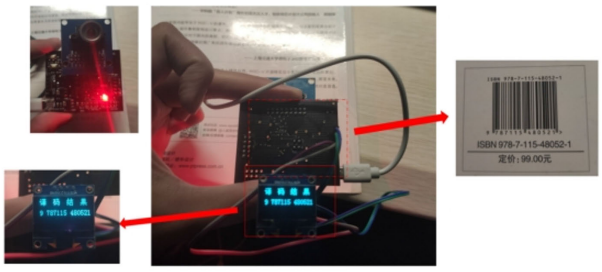

FIGURE 17. Test results of the finished chip.

This SoC chip has advantages such as a small footprint, low cost, straightforward operational procedures, and ease of software debugging. However, it exhibits limitations in accurately recognizing the conditions of excessive or insufficient illumination, blurry images, and severe damage to the barcodes. Further optimization of the barcode recognition algorithm is required to address these challenges. However, the system still has room for improvement. In scenarios with excessively strong or dim lighting, blurred images, or severely damaged barcodes, the recognition accuracy is

not ideal. Further optimization is required in the barcode recognition algorithm to address these limitations.

REFERENCES

[1] N. Dutt, A. Jantsch, and S. Sarma, "Toward smart embedded systems: A self-aware system-on-chip (SoC) perspective," *ACM Trans. Embedded Comput. Syst.*, vol. 15, no. 2, pp. 1–27, Jun. 2016.

[2] J. Hu, X. I. E. Lihong, Z. O. U. Wanghui, Z. H. A. N. G. Lei, and H. U. Xiaodong, "Design of miniature image acquisition system based on low power system on chip," *J. Hunan Univ., Natural Sci.*, vol. 46, no. 2, pp. 86–91, 2019, doi: 10.16339/j.cnki.hdxbzkb.2019.02.012.

[3] E. Gholizadehazari, T. Ayhan, and B. Ors, "An FPGA implementation of a RISC-V based SoC system for image processing applications," in *Proc. 29th Signal Process. Commun. Appl. Conf. (SIU)*, Jun. 2021, pp. 1–4.

[4] I. Bravo-Muñoz, J. L. Lázaro-Galilea, and A. Gardel-Vicente, "FPGA and SoC devices applied to new trends in image/video and signal processing fields," *Electronics*, vol. 6, no. 2, p. 25, Mar. 2017, doi: 10.3390/electronics6020025.

[5] H. Y. Yan, L. Chen, and X. Tong, "Design of SoC security check architecture for Internet of Things applications," *Comput. Eng.*, vol. 47, no. 2, pp. 152–159, 2021, doi: 10.19678/j.issn.1000-3428.0057460.

[6] G. Soulard, G. P. Lachance, É. Boisselier, M. Boukadoum, and A. Miled, "RISC-V based processor architecture for an embedded visible light spectrophotometer," in *Proc. IEEE Can. Conf. Electr. Comput. Eng. (CCECE)*, Sep. 2022, pp. 360–363, doi: 10.1109/CCECE49351.2022.9918259.

[7] R. Höller, D. Haselberger, D. Ballek, P. Rössler, M. Krapfenbauer, and M. Linauer, "Open-source RISC-V processor IP cores for FPGAs—Overview and evaluation," in *Proc. 8th Medit. Conf. Embedded Comput. (MECO)*, Jun. 2019, pp. 1–6.

[8] J. X. Sui and Y. H. Ji, "Design of Internet of Things SOC platform based on RISC-V processor," *Mod. Electron. Technol.*, vol. 45, no. 3, pp. 118–123, 2022.

[9] M.-R. Roberto and A. Alfredo, "An energy consumption benchmark for a low-power RISC-V core aimed at implantable medical devices," *IEEE Embedded Syst. Lett.*, vol. 184, no. 7, pp. 1106–1112, Jun. 2022.

[10] C. H. Wei, R. P. Wang, and R. I. Chen, "Design of EAN-13 bar code chip module based on image recognition," *J. Guangxi Univ., Natural Sci. Ed.*, vol. 2022, no. 4, p. 47, 2022.

[11] G. R. Narayan and M. V. James, "Barcode recognition from video by combining image processing and Xilinx," *Proc. Eng.*, vol. 38, pp. 2140–2146, Jan. 2012.

[12] L. Xu, V. R. Kamat, and C. C. Menassa, "Automatic extraction of 1D barcodes from video scans for drone-assisted inventory management in warehousing applications," *Int. J. Logistics Res. Appl.*, vol. 21, no. 3, pp. 243–258, May 2018.

[13] P. Haribabu, G. Sasirekha, M. Rao, J. Bapat, and D. Das, "RISC-V core for ethical intelligent IoT edge: Analysis & design choice," in *Proc. IEEE/ACM 23rd Int. Symp. Cluster, Cloud Internet Comput. Workshops (CCGridW)*, Bangalore, India, May 2023, pp. 110–117, doi: 10.1109/CCGRIDW59191.2023.00031.

[14] S. Jiang, Y. Zou, H. Wang, and W. Li, "An FFT accelerator using deeply-coupled RISC-V instruction set extension for arbitrary number of points," in *Proc. IEEE 34th Int. Conf. Appl.-Specific Syst., Archit. Processors (ASAP)*, Porto, Portugal, Jul. 2023, pp. 165–171, doi: 10.1109/asap57973.2023.00036.

[15] Y. Chen, Y. Pei, X. Lu, and M. Shi, "Point multiplication ZigZag: An optimized co-processor architecture design for SM2/3 based on RISC-V," in *Proc. IEEE Int. Conf. Commun.*, Rome, Italy, May 2023, pp. 3793–3798, doi: 10.1109/ICC45041.2023.10279122.

[16] D. Z. Li, K. N. Cao, M. Qu, and F. Wang, "Co-simulation verification of five-stage pipelined RISC-V processor with software and hardware cooperation," *J. Jilin Univ., Inf. Sci. Ed.*, vol. 35, no. 6, pp. 612–616, 2017, doi: 10.19292/j.cnki.jdxxp.2017.06.006.

[17] Z. Dong, Y. Han, C. S. Lai, G. Zhou, and D. Qi, "EMSN: An energy-efficient memristive sequencer network for human emotion classification in mental health monitoring," *IEEE Trans. Consum. Electron.*, vol. 69, no. 4, pp. 1005–1016, Nov. 2023, doi: 10.1109/TCE.2023.3263672.

[18] X. Ji, Z. Dong, Y. Han, C. S. Lai, and D. Qi, "A brain-inspired hierarchical interactive in-memory computing system and its application in video sentiment analysis," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 33, no. 12, pp. 7928–7942, Dec. 2023, doi: 10.1109/TCSVT.2023.3275708.

[19] R. Serrano, M. Sarmiento, C. Duran, K.-D. Nguyen, T.-T. Hoang, K. Ishibashi, and C.-K. Pham, "A low-power low-area SoC based in RISC-V processor for IoT applications," in *Proc. 18th Int. SoC Design Conf. (ISOCC)*, Oct. 2021, pp. 375–376, doi: 10.1109/ISOCC53507.2021.9613880.

[20] A. Roy and S. Clara, "A FPGA based implementation of Sobel edge detection," *Comput. Rev.*, vol. 59, no. 12, p. 656, 2018.

[21] S. P. Pan and Y. Y. Liu, "Overview of RISC-V microprocessors and commercial IP," *Appl. Microcontrollers Embedded Syst.*, vol. 20, no. 6, p. 5, 2020.

[22] W. Wu, D. Su, B. Yuan, and Y. Li, "Intelligent security monitoring system based on RISC-V SoC," *Electronics*, vol. 10, no. 11, p. 1366, Jun. 2021.

SIJIE LIN received the B.S. degree from the College of Physics and Information Engineering, Fuzhou University, Fuzhou, China, in 2022, where he is currently pursuing the M.S. degree in integrated circuit engineering. His research interests include digital image processing and RISC-V.

RENPING WANG received the B.S. degree from China Textile University, Shanghai, China, in 1994 and the M.S. degree from Fuzhou University, Fujian, China, in 2008. From 1994 to 2005, he worked at Beijing Yanshan Petrochemical Company and Fujian Qiaoxing School. He joined Fuzhou University in 2008, where he is currently an Associate Professor with the Department of Microelectronics Science and Engineering. His research interests include digital integrated circuit system design and full custom design.

TING CAI received the B.S. degree from the College of Physics and Information Engineering, Fuzhou University, Fuzhou, China, in 2021, where he is currently pursuing the M.S. degree in integrated circuit engineering. His research interests include digital system design and RISC-V.

YUNZE ZENG received the B.S. degree in electronics information science and technology from Jiangxi University of Science and Technology, Jiangxi, China. He is currently pursuing the M.S. degree in integrated circuit engineering with the School of Fuzhou University, Fuzhou, China.