

# 基于 ZYNQ 的高清图像显示及检测系统设计

林振钰, 张志杰, 刘佳琪

(中北大学 仪器科学与动态测试教育部重点实验室, 太原 030051)

**摘要:** 针对当前基于 ARM 和 DSP 的嵌入式图像处理系统前端采集速度慢和图像处理算法不易加速的缺点, 设计了一种基于 HDMI 接口的全高清(分辨率 1920×1080)实时视频采集与图像处理系统; 采用 500 万像素级别 CMOS 摄像头作为前端数据源, 主芯片内部采用 ARM+FPGA 的异构架构, 兼备 FPGA 的并行处理能力与 ARM 处理器任务调度功能; 基于 AXI 协议设计了自定义数据存储传输的 IP 核, 实现了处理速度与带宽最大化; 利用 HLS 工具将图像预处理算法快速打包生成 IP 核, 在 FPGA 中实现图像算法的硬件加速, 完成图像处理系统平台原型机的设计; 与传统的 PC 机和相机的机器视觉平台相比, 该系统运行平均耗时在 10 ms 以内, 实时检测效果令人满意, 有效解决了低功耗与高数据带宽和处理速度之间的矛盾, 为后端结果分析和边缘加速提供了良好支持。

**关键词:** ZYNQ 芯片; 图像处理; 实时性; ARM 处理器; FPGA; Vivado HLS

## Design of High-definition Image Display and Detection System Based on ZYNQ

Lin Zhenyu, Zhang Zhijie, Liu Jiaqi

(Ministerial Key Laboratory for Instrumentation Science and Dynamic Measurement, North China University, Taiyuan 030051, China)

**Abstract:** Aiming at the shortcomings of the current ARM and DSP-based embedded image processing system, such as the front-end acquisition speed is slow and the image processing algorithm is not easy to accelerate, a full HD (resolution 1920×1080) real-time video acquisition and image processing system based on the HDMI interface is designed. A 5 million pixel CMOS camera is used as the front-end data source. The main chip uses an ARM+FPGA heterogeneous architecture, which has both the parallel processing capability of the FPGA and the task scheduling function of the ARM processor. Based on the AXI protocol, a custom data storage and transmission IP core is designed to maximize processing speed and bandwidth. Use the HLS tool to quickly package the image pre-processing algorithm to generate an IP core, implement hardware acceleration of the image algorithm in FPGA, and complete the design of the image processing system platform prototype. Compared with the traditional PC and camera combined machine vision platform, the system takes an average of less than 10 ms to run, and the real-time detection effect is satisfactory, effectively solving the contradiction between low power consumption and high data bandwidth and processing speed. Provides good support for back-end result analysis and edge acceleration.

**Keywords:** ZYNQ chip; image processing; real time performance; ARM processor; FPGA; Vivado HLS

## 0 引言

图像包含大量的信息, 对于人类而言视觉信息利用率最高, 人类对数字图像显示的效果的判断会受到各种外部环境和主观意识等因素的影响。随着信息技术的飞速发展, 在工农业生产中通过加入图像系统提取有用信息进行处理<sup>[1]</sup>, 从而输出检测结果或者控制信号, 来提高系统的智能化和自动化。在某些特定场景下, 如交通运输、精密测控、显微医学操作、工业探伤以及农田作业等领域<sup>[2]</sup>, 对图像有实时性和高精度检测要求, 而在图像显示及处理的过程中要占用大量资源和时间, 例如, 尺寸为 1 920×1 080

的 8/24 bit 的 BMP 图像大小为 5.93 MB<sup>[3]</sup>, 连续的图像数据频带较宽, 数据量大, 像素相关性大。传统的 DSP 或 ARM 处理器采样速率较低, 且系统使用浮点, 很难对其进行直接的处理, 且成本较高。FPGA 适用于高速采样频率且任务比较固定或重复的情况, 系统使用定点, 可灵活定制样机。本文基于 Xilinx 公司 ARM + FPGA 体系结构的 Zynq 高性能芯片, 在 AXI 协议框架下设计了自定义数据存储传输的 IP 核, 实现了处理速度与带宽最大化。利用 HLS 工具将图像预处理算法快速打包生成 IP 核<sup>[4]</sup>, 为各种中端图像处理算法提供了可扩展的 FPGA 硬件加速通道, 利用

收稿日期: 2020-06-29; 修回日期: 2020-07-15。

作者简介: 林振钰(1995-), 男, 山西吕梁人, 硕士研究生, 主要从事仪器与测试技术、图像识别与处理等方向的研究。

张志杰(1965-), 男, 山西忻州人, 博士, 教授, 主要从事仪器与测试技术、动态测试技术与应用、信号处理理论与技术等方面的研究。

引用格式: 林振钰, 张志杰, 刘佳琪. 基于 ZYNQ 的高清图像显示及检测系统设计[J]. 计算机测量与控制, 2021, 29(2): 30-34, 39.

软硬协同设计的方法完成了 HDMI 接口的高清图像显示系统平台的原型机设计。有效解决了低延时低功耗与高数据带宽和处理速度之间的矛盾<sup>[5]</sup>, 为后端结果处理与应用提供了良好支持。

## 1 系统硬件电路框架

基于 ZYNQ 的实时视频高速采集处理系统原理如图 1 所示, 系统平台主要由图像采集模块、数据存储模块、硬件加速模块、内部传输互联模块、HDMI 高清显示模块组成。Zynq-7000SoC 可扩展平台作为整个实时处理系统的核心, 上方矩形为 FPGA 可编程逻辑 (PL, programmable logic) 部分, 下方矩形为 ARM 处理系统 (PS, processing system)。其中 PS 部分的高性能 DDR3 存储单元通过 AXI 内部互联与 PL 端进行大吞吐量的高速数据通信, 其片内带宽足够大, 消除了芯片间互联存在的带宽瓶颈问题<sup>[6]</sup>。

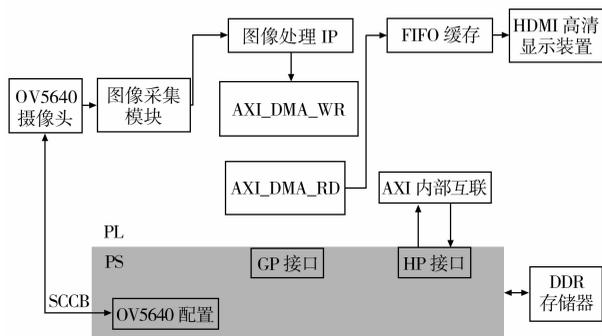


图 1 系统结构原理框图

根据系统读取外界视频数据信息的数据流向, 从软硬件功能层面具体划分为:

1) 采集部分: 视频通过位于 PL 侧的 500 万像素的 cmos 摄像头进行采集, PS 侧引出 EMIO 引脚通过软件模拟 SCCB 协议对 cmos 摄像头进行配置。将采集到的图像数据送入 PL 中使用 Verilog HDL 编程封装的图像采集模块将数据位宽整合成 24 位。

2) 硬件加速部分: 将 RGB 数据格式传送到图像预处理模块, 实现对 RGB 像素信息的各种预处理, 如灰度化、滤波、边缘化等, 此模块利用 Xilinx 的 HLS 工具, 通过 C 语言编程实现各种图像预处理 IP 核的快速定制。在 PL 端实现硬件加速功能, 提高图像处理的实时性。

3) 存储传输模块: 自定义 AXI\_DMA\_WR 写模块将数据通过 HP 口与 PS 端的 DDR 进行高数据交互, 将处理后的视频数据传输至 DDR3 内存芯片中存储, 最后由 PS 端发出读信号控制 AXI\_DMA\_RD 读模块将 DDR3 中的图像数据读出并缓存到 PL 端的 FIFO 模块。

4) 高清显示模块: 实现 HDMI 接口驱动逻辑, 完成图像显示功能。根据差分信号高速数据传输技术 TMDS (transition minimized differential signaling), 编码驱动图像输出到 HDMI 接口, 实现较传统 VGA 接口更为高清的图

像实时显示功能。

## 2 系统关键部分实现

### 2.1 视频采集模块

视频采集模块选用 OmniVision 公司 OV5640 摄像头作为数据采集前端, 它是一款 1/4 英寸单芯片图像传感器, 其感光阵列  $2\,592 \times 1\,944$  (即 500 W 像素), 能实现最快 15 fps QSVGA ( $2\,592 \times 1\,944$ ) 或者 90 fps VGA ( $640 \times 480$ ) 分辨率的图像采集。具有高灵敏度、低串扰和低噪声的优质性能。传感器内部集成了图像处理的功能, 包括自动曝光控制 (AEC)、自动白平衡 (AWB) 等。同时该传感器支持 LED 补光、MIPI (移动产业处理器接口) 输出接口和 DVP (数字视频并行) 输出接口选择、ISP (图像信号处理) 以及 AFC (自动聚焦控制) 等功能<sup>[7]</sup>。本系统中选择 DVP 接口模式<sup>[8]</sup>, 在 PS 端通过 SCCB 协议将传感器视频分辨率配置为  $1\,920 \times 1\,080$ , 视频帧率配置为 30 f/s。时钟频率为 74.25 MHz ( $2\,200 \times 1\,125 \times 30$ )。

为了匹配适应显示模块 HDMI 接口输入端 RGB888 格式, 在 PL 端使用 Verilog HDL 描述图像采集模块实现将 8 位原始图像数据转换为 24 位 RGB 数据的功能。摄像头采集模块在像素时钟的驱动下将传感器输出的场同步信号、行同步信号以及 8 位数据转换成读写控制模块的写使能信号和 24 位写数据信号, 完成对 CMOS 传感器图像的采集。对图像采集模块进行测试仿真效果如图 2 所示, 在帧同步信号激励下 RGB 颜色分量各占 8 位位宽输出。



图 2 图像采集模块测试仿真图

### 2.2 数据读写模块

数据读写模块即 FPGA 与 ARM 的数据交换, 不同工艺特征的处理器与 FPGA 融合在一个芯片上之后, 片内处理器与 FPGA 之间的互联通路就成了 ZYNQ 芯片设计的中中之重。如果 Cortex-A9 与 FPGA 之间的数据交互成为瓶颈<sup>[9]</sup>, 那么处理器与 FPGA 结合的性能优势就不能发挥出来。在 ZYNQ-7000AOSOC 器件中, PS 和 PL 之间的主要连接是通过一组 AXI (Advanced extensible Interface) 接口, AXI 协议是基于猝发式传输机制的一种高性能、高带宽、低延迟的片内总线, 总线由 5 个独立的通道构成: 1) 读地址; 2) 读数据; 3) 写地址; 4) 写数据; 5) 写响应。5 个通道都有各自的源端产生的 VALID 信号和目的端产生的 READY 信号组成的握手机制, 如图 3 所示。

在地址通道上, 每个交易有地址和控制信息, 这些信息描述了需要传输的数据性质。在主从设备之间传输数据, 分别使用从设备的写数据通道和到主设备的读数据通道。在主设备到从设备的写数据交易中, AXI 有一个额外的写响应通道。从设备通过响应通道向主设备发出信号表示写

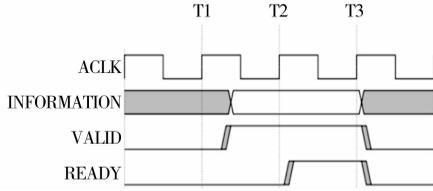


图 3 AXI 协议的握手信号对

交易完成。

### 2.2.1 AXI\_DMA\_RD 读模块

由于实时连续图像数据量大，本设计利用片内提供的 HP 口即 AXI 的高性能端口 (high performance ports) 进行数据交互，对 DDR3 进行直接存储映射访问，其可设 FIFO 缓冲来提供“批量”读写操作，数据宽度是 32 或 64 位，适用于大块数据的高性能传输，带宽高。吞吐量最高，由于 4 个 HP 接口中 PL 都是做主机的，故创建一个 master 接口的 AXI\_DMA\_RD 读模块。主机首先在地址通道给出读地址和控制信号，然后从机由读数据通道返回读出的数据，如图 4 所示。突发读操作主机只给出一个地址，从该地址连续突发读出数据。

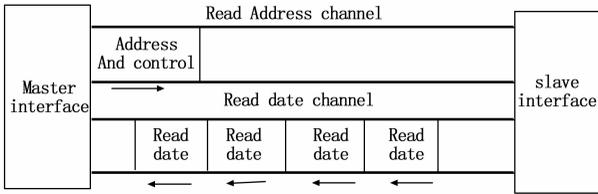


图 4 突发读操作过程

通过编辑 IP 保留模块接口信息，将 AXI\_DMA\_RD 读模块突发长度 M\_AXI\_LAST 改为 256，数据位宽设为 64，从而提高 AXI 访问 DDR3 的效率，将带宽做到最大，通过 AXI 总线把 DDR3 的数据读出来进入 FIFO 缓存。

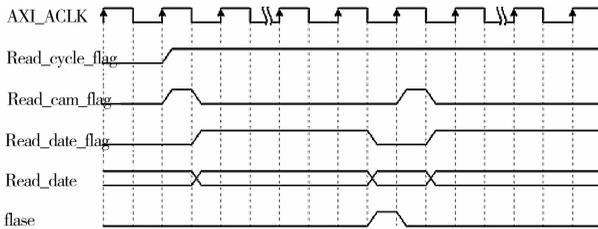


图 5 AXI\_DMA\_RD 读模块时序图

具体设计时序如图 5 所示，在 PS 端控制读数据初始化脉冲 TXN 后，进入读数据阶段，使 read\_cycle\_flag 拉高并一直保持。Read\_cmd\_flag 开始发送读地址给从机，从机反馈数据 read\_data\_flag，数据接收完毕后产生一个 flase 结束信号。256 个数据读完之后读结束 read\_data\_flag 拉低，地址清零，每个像素点占 4 个字节，则有结束地址 READ\_END\_ADDR 为 8 294 400 (1 920 \* 1 080 \* 4)。在 read\_cycle\_flag 一直保持高电平状态下循环再读。内部生成位宽为 64，深度为 4 096 的异步 FIFO。为了保证 FIFO 内部一直有数据，在 FIFO 缓存接口添加 count 计数

器，当 FIFO 数据小于 2 047 就开始从 DD-R 读数据。由于突发长度为 256，故将地址递增参数 burst\_size\_bytes 设为 2 048 (256 \* 8)。最后通过添加例化 RGB 时序模块，将该模块封装成以 RGB 时序输出的模块，显示模块给内部 FIFO 读使能，当 FIFO 内数据量大于 2 047 时候读使能信号拉高，将读出数据缓存到 FIFO 送给显示器。

### 2.2.2 AXI\_DMA\_WD 写模块

同理，在编辑 AXI\_DMA\_WD 写模块时例化一个位宽 32，深 8 192 的 FIFO。图像采集模块帧同步信号作为内部 FIFO 写使能，读端口则设置为 64 位位宽，当 FIFO 读 counter 大于 2 047 时把 W\_axi\_flag 拉高开始写数据。

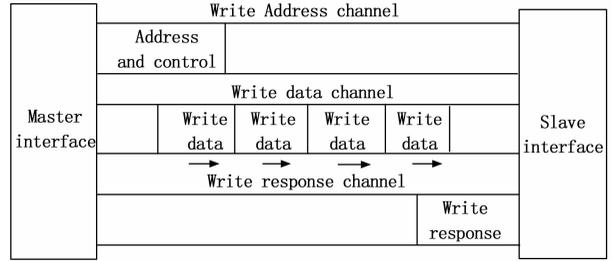


图 6 写传输过程

写传输过程如图 6 所示，它用到了写地址、写数据和写响应 3 个通道。主机在写地址通道给出写地址和控制信号，然后在写数据通道连续突发写 4 个数据。从机在接收数据之后，在写响应通道给出响应信号。

### 2.2.3 读写传输测试

在 VIVADO 工具中创建一个 BD 文件，添加 ZYN-Q7 处理系统 IP，通过配置 ZYNQ7 处理系统 IP 进行读写传输功能测试，引出一个 GPIO 接口作为 AXI\_DMA\_RD 读模块的启动信号，在 PS 端设定一个 TXN 启动脉冲信号，内存基地址即访问 DDR3 的起始地址设为 0X01000\_000。PS 端通过 C 代码模拟生成从 1 开始累加的测试数据写入 DDR3 并读出，PL 端添加逻辑分析仪监测 IP 检测，从而实现通过软件触发硬件运行，抓取片内实时信号的自回环测试功能，如图 7 所示。

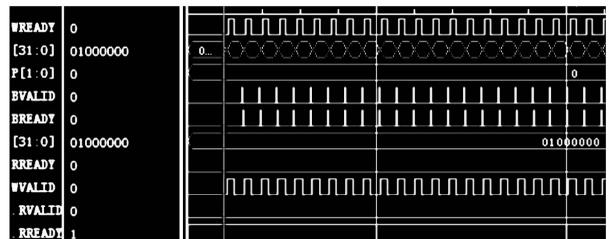


图 7 逻辑分析仪抓取的实时信号

在 SDK 工具中添加 Memory 监视器，可实时查看 DDR 内存测试数据动态，如图 8 所示。结果表明，该模块可实现预期功能。

### 2.3 高清显示模块

显示终端采用带有 HDMI 接口的设备，HDMI 既高清

01000000	00000001	00000002	00000003	00000004
01000010	00000005	00000006	00000007	00000008
01000020	00000009	0000000A	0000000B	0000000C
01000030	0000000D	0000000E	0000000F	00000010

图 8 DDR3 中的实时数据

晰度多媒体接口 (high definition multimedia inter - face), 是新一代的一种数字音视频接口, 较传统的 V - GA 接口有数据传输带宽高、色彩度高及传输速度快等显著优点。本设计仅使用 HDMI 接口显示图像, 不传输音频, 因此只实现 DVI 接口驱动逻辑完成图像显示功能。DVI 和 HDMI 接口协议在物理层都使用 TM - DS 标准传输音视频数据。TMDS (transition minimized differential signaling) 是一项使用差分信号高速数据传输技术<sup>[10]</sup>。

由于每个像素点的颜色深度为 24 位, 即 RGB 每个颜色分量各占 8 位, 通过添加一个 RGB2DVI 模块, 将 RGB888 格式的视频图像转换成 TMDS 数据输出。每个通道上的颜色数据通过一个 8B/10B 的 Encoder 编码器来转换成一个 10 位的像素字符。10 位的字符通过 Serializer 并串转换器转换成串行数据, 最后经过信号输出时的 OBUFDS 差分信号缓冲器由 TMDS 数据通道发送出去<sup>[11]</sup>, OBUFDS 模块在 FPGA 内通过添加例化原语实现, 将内部逻辑信号转换成差分信号输出。上述过程的具体实现流程如图 9 所示。

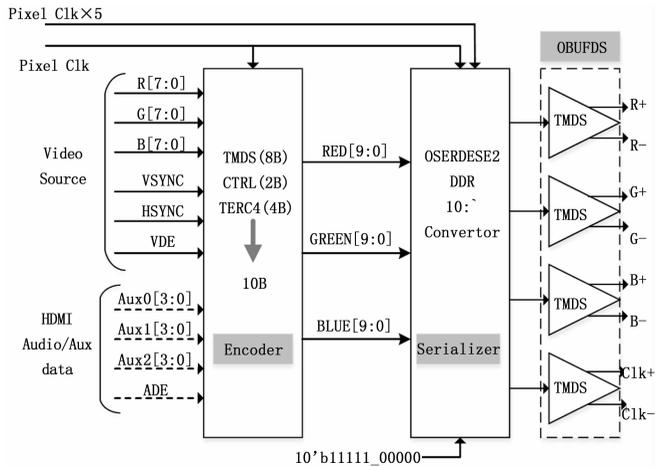


图 9 RGB2DVI 模块框图

需要注意的是, 10 : 1 的并转串过程所生成的串行数据速率是实际像素时钟速率的 10 倍, 实现的是 10 : 1 的转换率。整个系统需要两个输入时钟, 一个是视频的像素时钟 Pixel Clk (74.25 MHz), 另外一个时钟 Pixel Clk x5 (371.25 MHz) 的频率是像素时钟的 5 倍。在此频率基础上实现双边沿采样, 即在 5 倍时钟频率的基础上又实现了双倍数据速率。

在支持 HDMI 接口的 1 080 p 分辨率的显示器上对高清显示模块进行测试, 测试结果如图 10 所示。测试过程中, 系统内部产生三基色、棋盘格图像数据源, 结果表明, 显示效果符合系统要求。

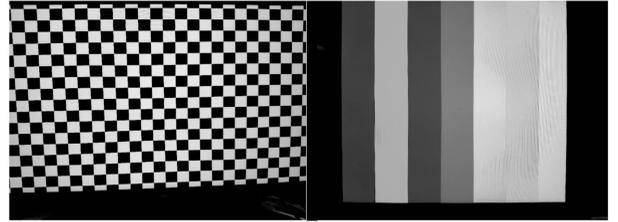


图 10 高清显示模块测试效果图

### 2.4 图像特征提取 IP 核的实现

对图像预处理的算法常见的有中指滤波, 均值滤波, 高斯滤波, 直方图均衡化等<sup>[12]</sup>。在工业生产中, 图像的特征信息显得尤为重要。工业电子显微镜单一的图像显示对焦, 不能实时迅速地对其进行处理与特征提取。通过某种方式对图像中的每一个像素点进行遍历, 判断每个像素点是一个图像特征, 从而减少冗余信息, 可直观地评估图像为后续提供有效信息。

边缘检测即可标识数字图像中亮度变化明显的点, 大幅度地减少数据量, 剔除不相关的信息, 保留图像重要的结构属性。在边缘检测算法中, Sobel 边缘检测算子采用 2 个方向的 3 \* 3 模板检测水平和垂直方向的边缘, 如表 1、2 所示, 其计算简单、速度快, 满足实时性的要求。

表 1 滤波模板

-1	0	+1
-2	0	+2
-1	0	+1

表 2 滤波模板

+1	+2	+1
0	0	0
-1	-2	-1

将上述 3 \* 3 滤波模板与图像作平面卷积, 即可分别求得横向和纵向的亮度差分近似值。以 A 代表原始图像, G<sub>x</sub> 和 G<sub>y</sub> 分别代表经横向和纵向边缘检测的图像灰度值<sup>[13]</sup>, 其卷积因子计算公式如下:

$$G_x = \begin{bmatrix} 1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{bmatrix} * A \quad G_y = \begin{bmatrix} +1 & +2 & +1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} * A \quad (1)$$

以 f(x,y) 表示图像 (x,y) 点的灰度值, 则有:

$$G_x = [f(x+1,y-1) + 2 \times f(x+1,y) + f(x+1,y+1)] - [f(x-1,y-1) + 2 \times f(x-1,y) + f(x-1,y+1)] \quad (2)$$

$$G_y = [f(x-1,y-1) + 2 \times f(x,y-1) + f(x+1,y-1)] - [f(x-1,y+1) + 2 \times f(x,y+1) + f(x+1,y+1)] \quad (3)$$

图像的每一个像素点的横向和纵向灰度值均通过以下公式来计算：

$$G = \sqrt{G_x^2 + G_y^2} \quad (4)$$

通过将数值与提前设定好的合理阈值比较来判定该点是否为边缘信息点。

Xilinx 推出的高层次综合工具 HLS 中包含大量的视频库函数，大部分图像处理模块函数和 OpenCV 库函数一致。本文利用视频处理库中的 sobel 函数，通过 C++ 代码综合、仿真生成 RTL 电路，将其快速封装生成 IP 核，在 PL 端对图像处理算法进行硬件并行处理加速。

其具体开发流程中的关键语句如图 11 所示。首先，将视频流格式数据转换成 mat 类型，mat 类型的 RGB888 格式的彩色图像再转换成单通道的灰度图像，用灰度数据与 Sobel 算子卷积；其次，将灰度数据转换成 3 个通道的灰度图像；最后，将 mat 类型的图像数据转换成视频流格式输出，完成处理功能。经仿真、综合完成后，将其打包生成 IP 核在 PL 中调用。在视频处理功能和数据结构方面，被综合后的代码与 OpenCV 基本对应，该方法极具灵活性、可扩展性，加快了图形处理模块设计的开发时间。显著提高了设计效率。

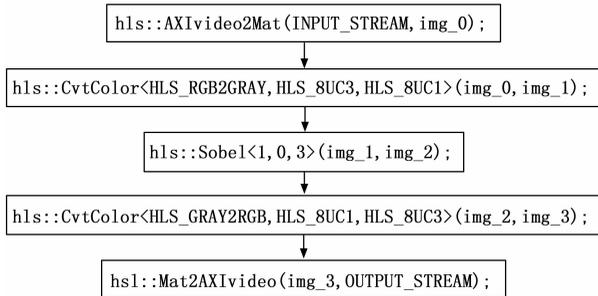


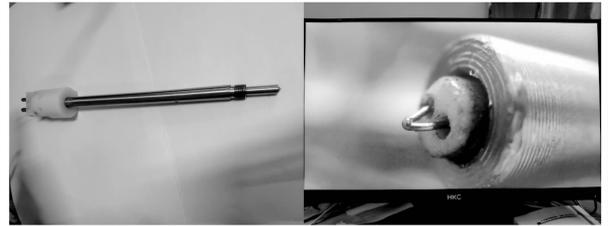
图 11 基于 HLS 实现 Sobel 算子的设计流程

### 3 实验结果与分析

用 Vivado 工具的 IP 集成器将上述生成的 IP 核添加到系统 Block Design 中，对本设计进行最后的综合、实现、并生成比特流文件。导出最新的硬件平台即 BSP 板级支持包后启动 SDK 开发平台进行设计，利用 C 语言编程依次配置 GPIO 输入输出端口信息，设定系统初始化脉冲 TXN。并通过 SCCB 协议配置摄像头参数信息。在 SDK 中对其进行单步调试，直到程序进入 while 主循环，周而复始地对 DDR 中起始地址 0×1000000 开始不断进行读写操作，处理连续的完整图像数据。

功能测试：采用两种不同场景对系统平台进行功能测试。

1) 关闭图像处理通道，显示原始高清图像，采用一支偶丝直径为 0.25 mm 的高温热电偶对系统平台的高清显示功能进行测试，测试结果如图 12 所示。结果表明，该系统可直接对焦热电偶结点，清晰地显示肉眼不方便观察的偶结状态，对热电偶头部的损坏程度做出评估。



(a) 热电偶 (b) 经系统显示的偶结

图 12 原始图像高清显示的测试结果

2) 开启图像预处理硬件加速通道，采用带有激光烧蚀痕迹和受电火花侵蚀而形成槽形缺陷的特殊板材奥氏体钢对系统平台的图像实时处理功能进行测试。结果表明，通过调整检测阈值，可有效辨别出激光烧蚀痕迹与板材裂缝，如图 13 (c) 中的小圆点为激光烧蚀痕迹，图 13 (d) 为调整阈值后板材裂缝缺陷的检测效果，该结果可为板材的完好性评估提供有效参考价值。



(a) 测试样本 (b) 经系统显示的钢材表面



(c) 较小阈值检测效果 (d) 较大阈值检测效果

图 13 图像处理功能测试结果

性能测试：在两种不同的平台上对系统平台的性能进行对比测试，即将在 ARM 处理器上通过软件的方式直接调用由 OpenCV 自带函数处理的结果和在本系统硬件平台上进行处理的结果进行速度比对。其性能比对效果如表 3 所示。

表 3 软硬件边缘检测耗时比较

测试次数	硬件平台实现/ $\mu\text{s}$	OpenCV 软件实现/ $\mu\text{s}$	提升倍数
1	7 369	43 856	5.95
2	7 406	44 087	5.95
平均耗时	7 387.5	43 971.5	5.95

对系统占用资源进行汇总，片内资源使用情况如表 4 所示。

(下转第 39 页)