

## Design of Intelligent Monitoring and Alarm System Based on Domestic FPGA

LIU Jiyuan, SONG Changjun, TANG Yongming\*

(School of Electronic Science & Engineering, Southeast University, Nanjing Jiangsu 210096, China)

**Abstract:** In order to solve the problems of upgrading and intelligent functions of traditional monitoring, an intelligent monitoring and alarm system is designed based on domestic FPGA. This system consists of FPGA hardware, embedded SoC core, and supporting host computer software. The SoC core adopts the transplanted Cortex-M1 soft core scheme, and realizes TF card read and write acceleration, 800 Mbps storage bandwidth and 990 Mbps network after self-optimizing peripheral drivers. It can support video capture, video processing, and video upload in 1 024×768p@60 Hz video format. The system collects, stores, and processes real-time video through the SOC control hardware terminal, uses the UDP protocol to connect to the network to upload video data, and receives instructions from the host computer through the network, and performs voice alarms, video monitoring and other tasks. It has good practical value.

**Key words:** video uploading; video monitoring; voice alarm; FPGA

EEACC: 7210G doi: 10.3969/j.issn.1005-9490.2023.03.001

## 基于国产FPGA的智能监测报警系统设计

刘济源, 宋长骏, 汤勇明\*

(东南大学电子科学与工程学院, 江苏南京 210096)

**摘要:** 为了解决传统监控在升级换代、智能功能上的问题,设计了一种基于国产FPGA的智能监测报警系统。本系统由FPGA硬件、嵌入SoC内核、配套上位机软件三大部分组成,SoC内核采用移植Cortex-M1软核方案,在自主优化外设驱动后实现TF卡读写加速、800 Mbps存储带宽和990 Mbps网络带宽,可支持1 024×768p@60 Hz视频格式下的视频采集、视频处理、视频上传。应用系统功能包括:通过SoC控制硬件终端采集、存储、处理实时视频,采用UDP协议连网上传视频数据,并接收上位机通过网络发出的指令,执行语音报警、视频监控等任务,具有良好的实用价值。

**关键词:** 视频上传; 视频监控; 语音报警; FPGA

中图分类号: TP277

文献标识码: A

文章编号: 1005-9490(2023)03-0585-07

随着计算机、集成电路等学科的进步,社会进入大数据时代,智慧城市成为新的应用需求<sup>[1]</sup>。而作为面向城市终端的最后一公里的监控设备,就显得尤为重要。

一般的监控设备,作为传统的家用、道路/商场/企业监控、天网系统使用,仅仅需要视频传输功能即可。而传统应用场景,对于视频的清晰度、实时性、存储性、可编程性并没有太大的需求<sup>[2]</sup>。

但是随着智慧城市的试点、普及,网络传输带宽的提高,监控设备的功能必然将迎来新的变化,也将对监控设备带来更多的要求,可分为以下几个部分:

①从成本角度来看,大面积铺开新型终端监控设备,功在当下、利在千秋,但科技进步带来的换代成本过高。

②从终端要求角度来看,城市主机系统需要进行各种视频处理,包括人脸识别、流量分析<sup>[3]</sup>等等,视频清晰度和实时性越高越好。

③从实际应用角度来看,在新冠疫情、消防安全、国防安全等的检验下,监控设备不再局限于监控功能,还必须拥有语音提示、语音指挥功能,使得城市监管部门能在关键时刻执行交通疏散、疫情防控、紧急救援等指挥任务。

而监控设备使用的控制部分,可由定制化的ASIC芯片、高性能单片机、FPGA三者之一组成。但是定制化的ASIC芯片设计和换代成本很高,批量化换代更新不方便;高性能的单片机,价格高昂,且无法应对高清视频的实时处理和转发;FPGA芯片成本适中,因其可编程性、高速视频信号处理的优势,最适合作为监控设备的控制器<sup>[4]</sup>。

因此,本文依托国产化FPGA硬件平台,搭载DDR3存储颗粒、TF卡、摄像头,通过设计音频PCB扩展板、网络通讯系统及配套的电脑端处理软件,实现硬件高速视频采集、存储自检、语音报警、语音采集、远程网络遥控的功能,从而成功实现了智能监测报警系统。

## 1 系统整体设计结构

### 1.1 系统硬件结构

系统硬件端,采用紫光同创国产 PGL22G FPGA 芯片<sup>[5]</sup>,硬件板卡搭载 2 颗 256MB 的 DDR3 存储颗粒、千兆以太网卡槽、TF 卡槽、摄像头接口、HDMI 接口。

系统硬件结构如图 1 所示。

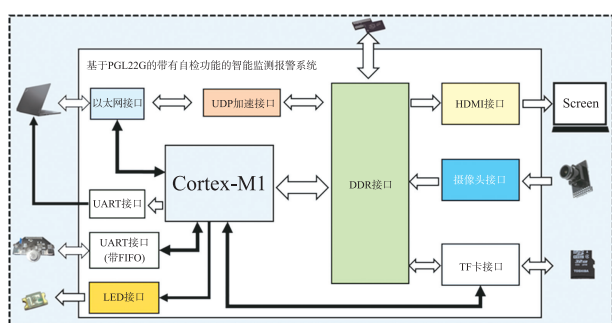


图 1 系统硬件结构框图

针对设计目标,需要实现 PC 端与硬件端的网络通信、硬件 TF 卡存储操作。纯硬件设计的控制系统,很难完成网络视频的传输、TF 卡文件系统的实现。因此,必须引入 SoC 内核。

硬件平台 PGL22G 芯片的主要硬件资源包括 17 536 个 LUT 查找表、26 304 个 D 触发器,48 块定制化 BRAM。可以计算出,仅仅够移植 ARM 公司的 Cortex-M1 和 M0 内核。M0 内核是低功耗内核,不适合操作实时高速视频的网络传输。而 M1 内核是 ARM 公司面向 FPGA 优化后的,主频可达 100 MHz~200 MHz,完全满足应用需求。

因此,系统采用 M1 内核嵌入至 FPGA 芯片的方案,并通过运行 FreeRTOS 操作系统<sup>[6]</sup>,对各种外设处理任务做调度。其中,DDR 存储器、TF 卡存储器、摄像头、以太网、音频接口、HDMI 接口等外设处理任务,需要通过硬件描述语言 VerilogHDL 实现,并挂载在 AHB 高速总线及 APB-To-AHB 桥路上<sup>[7]</sup>。

硬件端,主要执行软件端发送的指令,以及各种监控数据的采集、存储、自检等任务。

### 1.2 系统软件结构

系统软件端上位机,需要配合硬件端,通过以太网通信进行数据交互,包括获取硬件终端采集的视频数据、TF 卡存储文件、语音数据等,进行特定指令发送等。

为了方便调试和应用使用,上位机需要设计成可视化操作界面,并支持高带宽的网络实时通信。针对可视化 UI 界面的开发,采用 QT 框架实现最为合适。而 QT 框架,可通过 Python 或者 C++ 进行设

计,但是针对需求的高带宽的网络实时通信,以 Python 语言的处理方式,比 C++ 语言慢 10 倍以上,很难满足设计需求。

因此,软件方案采用 C++ 语言、QT 框架,进行界面 UI 开发、网络通讯协议设计,借助 WinAPI 接口进行千兆以太网数据解包、打包,从而实现配套上位机的设计。

软件上位机主要搭载登录界面、视频监控界面、TF 卡监测界面三大部分,同时可根据客户需求进行视频处理功能的扩展,例如人脸识别、行人识别等。上位机可提供的是终端监控采集的原始信息,后续可进行二次应用开发。

## 2 系统的硬件 RTL 层级实现

板卡搭载的系统层级 RTL 架构,如图 1 所示,主要包含 5 大部分:锁相环视频时钟发生器、SOC 接口、HDMI 接口、摄像头接口、APB 从机接口。其连接交互方式如图 2 所示。

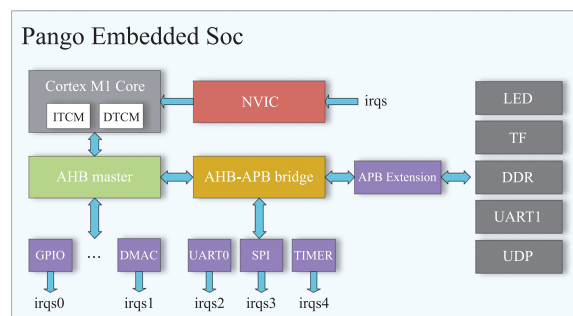


图 2 系统层级 ARM 架构

### 2.1 系统层级时钟树设计

为了保证系统运行时钟能够对以太网数据进行实时的解包打包处理,将 Cortex-M1 内核的系统运行时钟设置为与以太网收发时钟一致,均为 125 MHz。

同时,针对硬件设计的模块连接关系及物理连线中的时钟带负载能力,设计了如图 3 所示的时钟树。

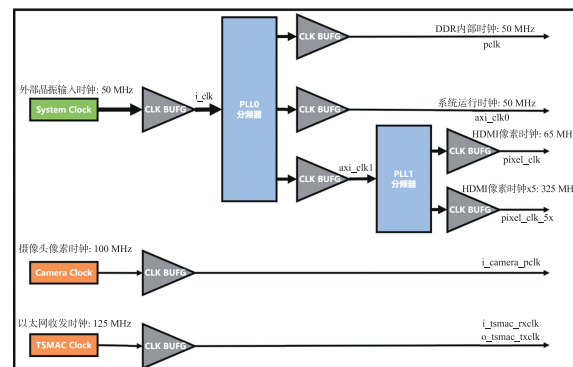


图 3 系统时钟树

其中主时钟树的根节点为输入的系统时钟 (50 MHz),其他时钟树,包括摄像头像素时钟 (100 MHz)、TSMAC 以太网收发时钟 (125 MHz)。

主时钟树下辖分支:AXI\_CLK0、AXI\_CLK1、pclk

①AXI\_CLK0:DDR3 IP 产生的 APB/AHB 总线时钟,时钟频率为 125 MHz。

②AXI\_CLK1:DDR3 IP 产生的 AXI4 总线时钟,时钟频率为 100 MHz。

③pclk:DDR3 IP 内部使用的时钟,时钟频率为 50 MHz。

AXI\_CLK0 分支下辖二级分支:SD\_Clock,TF 卡读写时钟,约束时钟频率为 50 MHz(该时钟可变,受 ARM 端控制,因此下图未标出)。AXI\_CLK1 分支下辖二级分支:HDMI\_Pixel、HDMI\_Pixel5x,HDMI/摄像头视频时钟,约束时钟频率为 65 MHz、325 MHz。

## 2.2 硬件 APB 从机接口设计

为了将外设包括 DDR 读写控制器、摄像头视频控制器、TF 卡读写控制器、UART 带 FIFO 缓冲控制器、LED 控制器等挂载在总线上,并利用好有限的可供用户使用的译码地址,对 AHB 总线地址进行二次译码,从机接口均通过一个译码器进行片选信号译码,分时复用。

## 2.3 硬件 APB 总线 TF 卡接口设计

针对外设 SPI 接口 TF 卡的读写驱动设计,可分为纯软件、软件为主硬件为辅、软件调度硬件为主的三种方式,具体介绍如下:

①纯软件的方式,即代表用 Cortex-M1 内核通过 IO 接口输出软件模拟的 SPI 数字信号来进行 TF 卡读写。

②软件为主硬件为辅的方式,即把 SPI 硬件驱动从软件中剥离开来,通过 Cortex-M1 内核调用硬件 SPI 接口进行 TF 卡读写。

③软件调度硬件为主的方式,即把 TF 卡读写驱动通过硬件实现,仅通过 Cortex-M1 内核进行 TF 卡文件系统的操作,文件系统发指令的操作全部由硬件完成。

其中,方案①纯软件的方式,虽然实现难度低,但是读写速度最慢。方案②软件为主硬件为辅的方式,实现难度适中,但是读写速度在 190 kB/s~1 MB/s 左右。读写速度最快的是方案③,相当于 DMA 直接操作,实现难度高,速度可达 2 MB 以上。

为了满足视频图片保存功能,让 1 024×768 格式的图片尽可能地快速保存,最终系统选择方案③

实现。实现过程中参考了 TF 卡国际标准命令协议<sup>[8]</sup>、闪迪 TF 卡数据手册,编写了硬件底层 TF 卡接口,包括上层 TF 卡读写控制器、TF 卡扇区读写驱动、TF 卡命令读写驱动、底层 SPI 主机控制器。

具备功能如下:

①TF 卡设备的 CID、CSD 信息读取,计算 TF 卡容量。

②TF 卡单/连续多扇区读写。

③总线数据读写。

④外部控制数据读写。

具体的设计结构,如图 4 所示。最终实测读写高速卡速率可以达到 2.96 MB/s。

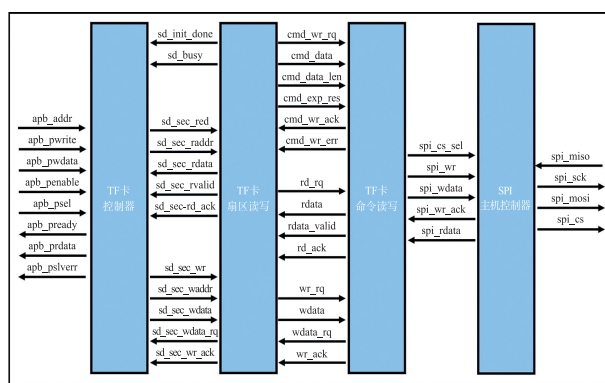


图 4 APB TF 卡接口

## 2.4 硬件 APB 总线 DDR 接口设计

由于所用 FPGA 平台的硬件 BRAM 资源不足以存储 Cortex-M1 内核的指令信息,因此只能将其用 D-Cache 的方式,通过预加载外部 FLASH 存储的指令 DTCM 数据到 DDR 中。这就意味着,M1 内核需要时刻读取 DDR 中指令信息。而设计目标中针对的视频实时网络传输,由于摄像头采集时钟与最终的网络输出不同步,必须先将其写入 DDR 中,再读出来发送。这样就会对 DDR 的读写造成影响,需要合理地分配和调度 DDR 的读写控制。

因此,系统需要设计一个 DDR 的仲裁接口,来进行数据的拥塞控制。根据 FPGA 厂商紫光同创的 DDR IP 说明,其接口协议定为 AXI4,所以读写接口就是实现一个 AXI4 总线的从机读写<sup>[9]</sup>。

这里给出 DDR 仲裁接口的设计原理,其主要是作为  $n$  个外设存取 DDR 数据中间的一个仲裁器,起到数据拥塞控制的作用<sup>[10-11]</sup>。也即,如果外设  $m$  需要存取数据,则首先向仲裁器发出申请,仲裁器处于空闲状态则通过该申请,并为其分配 DDR 读写通道;如果此时仲裁器处于忙状态,则该申请按照预先设定的优先级列表,进行队列 List 等待,当前面没有更高优先级的外设申请时,即可轮到外设  $m$  占用 DDR 总线。

### 3 系统的软件上位机实现

根据系统设计需求,上位机软件需要包括网络通讯、视频显示、文件树显示等功能。其中,主要需要解决的问题难点在于实时的网络通讯。

本系统设计需求不同于一般的网络通讯。系统对于网络 UDP 传输的解包有着较高的实时性要求,需要针对 1 024×768@ 60Hz 的视频进行数据解包接收,理论带宽传输压力位为 12 Mbps/每帧。传统的方式对于网络通讯的实现,通常调用 Socket 库,摆脱对于原始数据的处理,直接获取网络中的应用层数据。但是在本系统中,需要千兆级别带宽网络数据实时处理,Socket 调库方式效率低下,容易丢包,无法抓包高速、大数据量、连续的 UDP 数据。

经过研究和查阅 Windows C++开发手册,发现可以依托于 Windows 底层驱动接口 WinAPI<sup>[12]</sup>来进行网口数据获取。因此,针对于最重要的网络解析和发送设计,系统最终采用底层驱动解析方案,并结合 PCAP 库来开发网卡抓包、数据解析的 C++程序,最终实现实时抓包千兆网卡。

#### 3.1 上位机视频监控界面

上位机视频监控部分,根据系统设计需求,最终实现了自动扫描计算机网络接口、网卡接口选择、1 024×768@ 60 Hz 视频的实时图传、硬件视频分辨率调整和语音警报信息发送接口。



图 5 上位机视频监控界面

#### 3.2 TF 卡监测界面

上位机 TF 卡监测界面,根据系统设计需求,最终实现了自动扫描计算机网络接口、网卡接口选择、视频图传、视频截图(单张/连续多张连拍)、TF 卡文件目录树、TF 卡文件读写/重命名/删除/移动/创建、TF 卡坏区检测和语音警报信息发送接口。

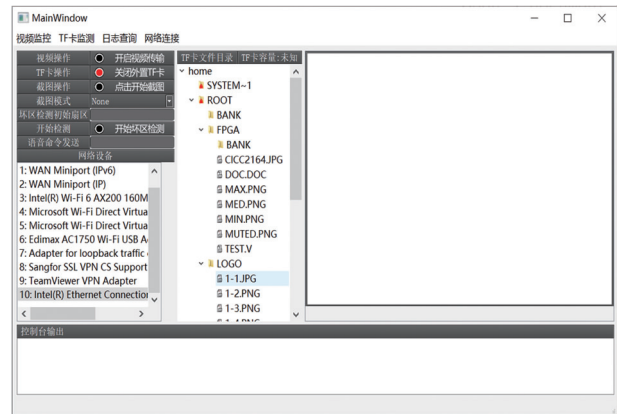


图 6 上位机 TF 卡监测界面

### 4 系统搭建与测试结果

#### 4.1 系统搭建说明

系统采用 OV5640 作为视频采集设备,TF 卡作为存储设备,音频扩展板作为音频采集和输出设备,通过软件上位机控制 FPGA 板卡执行具体的任务。

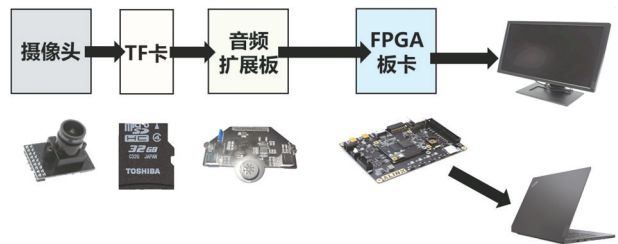


图 7 测试平台示意图

#### 4.2 系统功能时序仿真结果

为了验证系统的稳定性,需要在仿真时确认系统的时序指标符合设计要求<sup>[13]</sup>。FPGA 的时序指标结果,主要为关键路径的建立时间、保持时间的 slack、时序裕度  $F_{max}$ 。时序裕度是指在不违背内部建立时间和保持时间要求下可以达到的最大时钟频率,可通过时序裕度进行宏观的时序稳定性评判。

本系统对于时序指标的测试,通过在 FPGA 厂商提供的 EDA 工具 PDS 上进行时序的仿真,获取时序结果报告得到。

经过仿真,本系统只有运行时钟 AXI\_CLK0 为 107.124 MHz,没有达到约束要求 125 MHz,其他时钟的  $F_{max}$  全部达标,按照时序分析结果,本系统的时序路径,基本上达到设计要求。

Find:	Clock	Phase	Requested Frequency	Slack
1	TDM2_CLK	164.94208	125.00000MHz	1.938
2	Camera_Clock	140.64708	100.00000MHz	2.890
3	AXI_CLK0	77.42080	60.00000MHz	7.068
4	AXI_CLK0	107.12400	125.00000MHz	21.133
5	AXI_CLK1	124.02308	100.00000MHz	1.937
6	MEM2_Flat1	117.07808	65.00000MHz	6.642
7	MEM2_Flat1a	864.94408	125.00000MHz	0.479
8	System_Clock[SoC_Interface_Instr/DDR3_Interface_Instr/DDR3_Instr_u_p11_50_400/r_p11_m1/CLK0UT0_Inferred]	679.34008	500.00000MHz	0.264

图 8 时钟裕度仿真结果

除此之外,关键信号的扇出数决定了超长信号链的带负载能力,经过 CLKBUFG 时钟增强,将主时钟树上扇出树多的分支进行导流,减小扇出及时钟负载,如图 9 所示,最大扇出数为 Cortex\_M1 内核的系统时钟,仅有 2 609 个。

综上所述,本系统硬件部分的时序指标,基本满足应用需求。

Rank	Device	Fanout
1	SoC_InnerFace_Sat/Cortex_M1_InnerFace_Sat/F110P_R0F0	2609
2	SoC_InnerFace_Sat/DMC_InnerFace_Sat/F110P_R0F0	2104
3	SoC_InnerFace_Sat/DMC_InnerFace_Sat/rst_c180F0	1121
4	APB_R0F0	760
5	APB_R0F0	759
6	SoC_InnerFace_Sat/DMC_InnerFace_Sat/rst_p11_R0_40/rst_p11_p1	712
7	SoC_InnerFace_Sat/AMT_R0F0	671
8	F110P_R0F0	285
9	SoC_InnerFace_Sat/Cortex_M1_InnerFace_Sat/rst_inh_well_011	234
10	SoC_InnerFace_Sat/Cortex_M1_InnerFace_Sat/rst_inh_well_012	234
11	SoC_InnerFace_Sat/Cortex_M1_InnerFace_Sat/rst_inh_well_013	234
12	SoC_InnerFace_Sat/Cortex_M1_InnerFace_Sat/rst_inh_well_014	234
13	clocking_0	203
14	SoC_InnerFace_Sat/Cortex_M1_InnerFace_Sat/phy_ahb_dcache_0_mem0_ahb_dcache001_1	141
15	SoC_InnerFace_Sat/Cortex_M1_InnerFace_Sat/phy_ahb_dcache_0_mem0_ahb_dcache001_2	137
16	SoC_InnerFace_Sat/Cortex_M1_InnerFace_Sat/phy_ahb_dcache_0_mem0_ahb_dcache001_3	136
17	SoC_InnerFace_Sat/Cortex_M1_InnerFace_Sat/phy_ahb_dcache_0_mem0_ahb_dcache001_4	134
18	F110P_R0F0	140
19	SoC_InnerFace_Sat/Cortex_M1_InnerFace_Sat/phy_ahb_dcache_0_mem0_ahb_dcache001_7	129
20	SoC_InnerFace_Sat/Cortex_M1_InnerFace_Sat/phy_ahb_dcache_0_mem0_ahb_dcache001_8	128
21	SoC_InnerFace_Sat/Cortex_M1_InnerFace_Sat/phy_ahb_dcache_0_mem0_ahb_dcache001_9	124
22	SoC_InnerFace_Sat/Cortex_M1_InnerFace_Sat/phy_ahb_dcache_0_mem0_ahb_dcache001_10	124
23	SoC_InnerFace_Sat/Cortex_M1_InnerFace_Sat/phy_ahb_dcache_0_mem0_ahb_dcache001_11	123
24	SoC_InnerFace_Sat/Cortex_M1_InnerFace_Sat/phy_ahb_dcache_0_mem0_ahb_dcache001_12	115
25	SoC_InnerFace_Sat/Cortex_M1_InnerFace_Sat/phy_ahb_dcache_0_mem0_ahb_dcache001_13	113
26	SoC_InnerFace_Sat/Cortex_M1_InnerFace_Sat/phy_ahb_dcache_0_mem0_ahb_dcache001_14	113

图 9 高扇出信号仿真结果

### 4.3 系统功能布局布线优化结果

由于板卡资源并不是很充足,仅仅 Cortex-M1 内核就已经占据了 50%左右的硬件资源,对于其他外设的应用和功能设计造成了很大的资源限制。当 FPGA 芯片的硬件资源利用率超过 70%,软件工具将面临布线困难问题。

国产 EDA 工具 PDS 在处理布局布线问题时,其布线输出的 Log 信息和布线时间,反应了时序收敛的情况。因此,需要针对迭代信息和迭代时间绘制曲线图,来获取当前运行过程中的收敛情况,并根据结果做出优化调整。以下是针对布局布线优化时序前的布局布线迭代收敛图。

实验通过以下三种优化策略:①外设挂载,采用时钟 CLKBUFG 导流,将时钟树分叉,降低时钟负载与对应的扇出数,提高时序求解裕度。②优化逻辑设计,减少模块之间紧密的逻辑性,降低布线难度。③调节 PDS 软件参数,通过设定严格的参数,获得极佳的性能。但是过度的苛刻要求,会使得部分路径过长,因此严格设定之后,再进行适当的宽松设

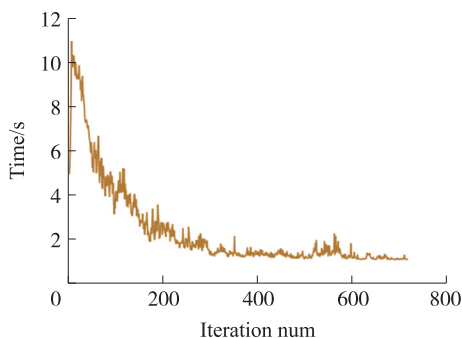


图 10 没有优化的迭代收敛图

计,让布线参数互相匹配,可以获得最优解。

经过优化之后的迭代收敛图如图 11 所示。

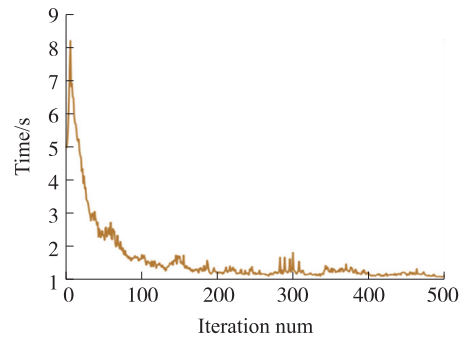


图 11 优化后的迭代收敛图

可以发现优化之后,布局布线迭代收敛的速度很快,迭代曲线波动小,收敛用时短,100 次左右可以收敛。

### 4.4 系统功能上板实验结果

按照 4.1 节所示搭建硬件系统,硬件实物图如图 12 所示。确认各设备连接正确后,硬件系统上电,打开软件上位机,通过上位机操作监控设备进行功能测试。

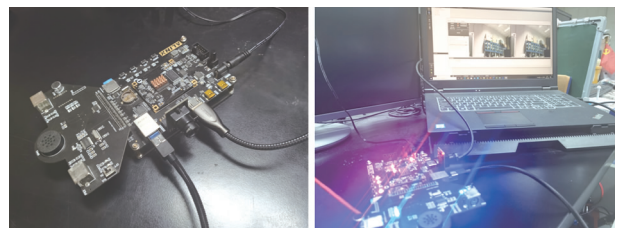


图 12 硬件实物连接图

视频监控功能测试,在上位机视频监控界面,通过选择不同的视频传输模式,来检查视频传输的正确性和稳定性。从图 13 的测试结果来看,系统可实现预期的监控视频设计功能。

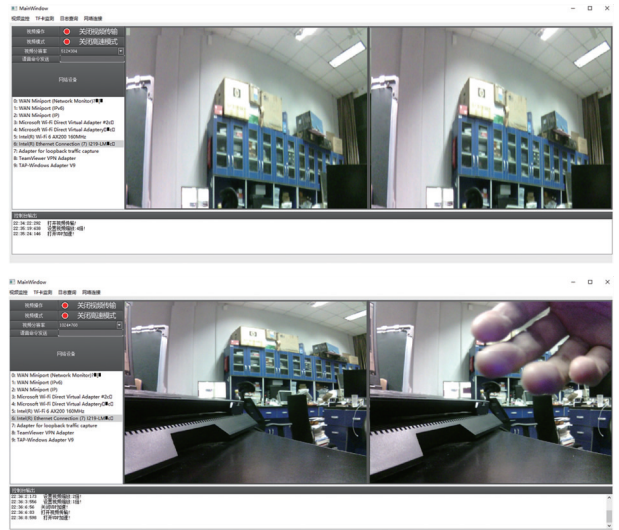


图 13 视频监控测试结果图

TF 卡监测功能测试,在上位机 TF 卡监控界面,通过操作 TF 卡和文件目录文件,来检查 TF 卡的保存图片功能、存取功能、自检功能等。从图 14、图 15 的测试结果来看,系统可通过加载 TF 卡文件,生成目录树,并且能够完成截图保存、文件存取、坏区自检操作,设计功能符合预期。

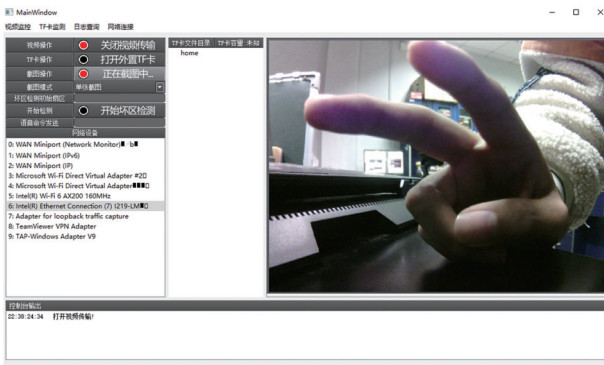


图 14 TF 卡监控界面截图存储测试结果图

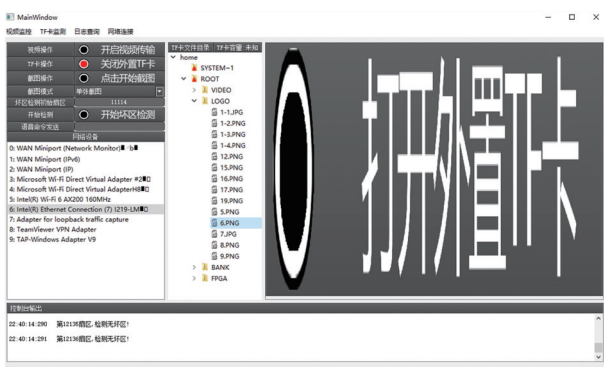


图 15 TF 卡监控界面文件读取和坏区检测测试结果图

### 4.5 系统功能资源占用结果

在经过对 ARM-Cortex-M1 进行内核裁剪、硬件工程优化之后,系统对 FPGA 资源占用率达到 65% 左右,使用了 PGL22G 芯片内部 64.58% 的 LUT、31.20% 的 FlipFlop 和 63.54% 的 BRAM,表 1 具体给出了系统资源使用。

表 1 系统资源使用情况

Resource	Utilization	Available	Utilization%
LUT	11325	17536	64.58
LUTRAM	96	1260	7.62
FlipFlop	8207	26304	31.20
BRAM	30.5	48	63.54
IO	113	240	47.08
PLL	2	6	33.33

## 5 结论

本文依托 FPGA 硬件平台设计的基于国产 FPGA 的智能监测报警系统,采用紫光同创 PGL22G 国产 FPGA 芯片作为主控,通过内嵌 SoC 控制硬件

终端采集、存储、处理实时视频,连网上传视频数据,并接收上位机通过网络发出的指令,执行语音报警、视频监控等任务。在经过测试后,系统的硬件设计具有良好的时序指标,满足应用需求,配套的上位机可扩展视频处理接口,使得本系统具有良好的商业监控实用价值。本设计为 2021 年第五届全国大学生集成电路创新创业大赛紫光同创杯竞赛命题,经过以上优化设计和应用开发,最终获得全国一等奖。

### 参考文献:

- [1] 朱庆. 三维 GIS 及其在智慧城市中的应用[J]. 地球信息科学学报,2014,16(2):151-157.
- [2] 王升哲,唐中和,郭航,等. 基于 SOC 架构的智能图像处理和外设控制系统设计[J]. 计算机测量与控制,2021,29(4):6.
- [3] Miklař M, Olszewski P, Nowosielski A, et al. Pedestrian Traffic Distribution Analysis Using Face Recognition Technology[C]//13th International Conference on Transport Systems Telematics, Katowice-Ustron, Poland, Springer, 2013:303-312.
- [4] Kuon, Ian, Jonathan Rose. Measuring the Gap Between FP-GAs and ASICs [J]. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 2007, 26(2):203-215.
- [5] 吴磊,王永甲,江先阳. 基于紫光现场可编程门阵列的无操作系统 LwIP 高效移植[J]. 计算机应用,2021,41(S2):183-186.
- [6] 杨斐文,张达,杨小聪,等. 基于 Cortex-M4 和 FreeRTOS 的矿用三维激光扫描测量系统设计[J]. 中国矿业,2014,23(S2):316-318,343.
- [7] Nawrath R, Czerwinski R. FPGA-Based Implementation of APB/SPI Bridge[J]. AIP Conference Proceedings, 2040(1):080011
- [8] Li P, Zhou Q L, Liu Y, et al. A Contactless Mobile Payment Method Based on Security TF Card and NFC Technology[J]. Advanced Materials Research, 2011:317-319:1769-1772.
- [9] Bhaktavatchalu R, Rekha B S, Divya G A, et al. Design of AXI Bus Interface Modules on FPGA[C]//2016 International Conference on Advanced Communication Control and Computing Technologies (ICACCCT), Ramanathapuram, India, IEEE, 2016:141-146.
- [10] Wang Y, Chen X L. Performance Analysis of DDR SDRAM in High Speed Image Data Acquisition[J]. Proceedings of SPIE, 2019, 11337:113370Z.
- [11] 尹春梅,陈钱,顾国华,等. DDR3 仲裁控制器设计及 FPGA 验证[J]. 计算机工程与设计, 2015, 36(8):2083-2089.
- [12] Sharma H, Kant S. Early Detection of Ransomware by In-

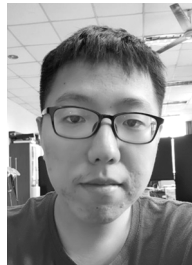
icator Analysis and WinAPI Call Sequence Pattern [C]//3rd International Conference on Information and Communication Technology for Intelligence Systems,

Ahmedabad, India, Springer, 2019:201-211.

[13] 胡凯,虞健,周洋洋,等. 自主FPGA芯片软件时序参数提取方法[J]. 电子与封装,2021,21(7):53-58.



刘济源(1998—),男,汉族,江苏徐州人,东南大学电子科学与工程学院硕士研究生,主要研究方向视频图像处理,FPGA应用、高速视频接口,erie@seu.edu.cn;



宋长骏(1998—),男,汉族,江苏徐州人,东南大学电子科学与工程学院硕士研究生,主要研究方向视频图像处理,可编程计算,songcj@seu.edu.cn;



汤勇明(1973—),男,汉族,江苏苏州人,东南大学电子科学与工程学院教授,通信作者,主要研究方向显示电子技术,电路与系统,嵌入式系统设计,视频图像处理,可编程计算,tym@seu.edu.cn。