

一种广覆盖的实验室安全监控系统设计

李嘉曾^{1,2} 韩利峰¹ 李丹清^{1,3} 黄文博¹ 吴丽梅^{1,2} 张立园^{1,2} 梁子薇^{1,2} 夏汇浩¹

1 (中国科学院上海应用物理研究所 上海 201800)

2 (中国科学院大学 北京 100049)

3 (上海交通大学 上海 200240)

摘要 为了保障钍基熔盐堆核能系统(Thorium Molten Salt Reactor Nuclear Energy System, TMSR)项目的实验安全运行,需要构建一套实验室安全监控系统,该系统需要解决以下问题:实验室地域覆盖广、环境差异大、信号种类繁多、软件接口复杂。现有的商业监控系统很难满足以上需求,因此开发了一套基于Node-Web技术的实验室安全监控系统。展示了该系统的结构框架,并阐述了基于视频接口进行软件开发的流程。该系统实现了EPICS (Experimental Physics and Industrial Control System)信号采集系统和海康威视(Hikvision)视频采集系统的集成,通过Node-Web前端可视化接口的开发,实现了监测界面中EPICS变量和视频流联动的功能。系统具有分布式、网络化、扩展性好、低成本的特点,可适用于建设大型科研机构实验室安全监控系统。

关键词 分布式实验室安全监控, EPICS, Node.js, 实时 Web 监控

中图分类号 TL99

DOI: 10.11889/j.0253-3219.2018.hjs.41.110605

Design of safety monitoring system for wide coverage laboratories

LI Jiazeng^{1,2} HAN Lifeng¹ LI Danqing^{1,3} HUANG Wenbo¹ WU Limei^{1,2}
ZHANG Liyuan^{1,2} LIANG Ziwei^{1,2} XIA Huihao¹

1(Shanghai Institute of Applied Physics, Chinese Academy of Sciences, Shanghai 201800, China)

2(University of Chinese Academy of Sciences, Beijing 100049, China)

3(Shanghai Jiao Tong University, Shanghai 200240, China)

Abstract [Background] The present commercial system for safety monitoring can hardly meet the complex requirements of thorium molten salt reactor (TMSR) Nuclear Energy Project's laboratory safety to deal with wide geographical coverage, distinct environmental difference, various signals, various software interfaces, etc. [Purpose] This study aims to build a laboratory safety monitoring system which can fulfil these requirements. [Methods] A laboratory safety monitoring system was developed by using Node.js, experimental physics and industrial control system (EPICS) and Hikvision video acquisition system. A function to link videos with EPICS signals was implemented. It possesses distributed structure, network distribution, well scalability and optimum economical efficiency. [Results] Various signals such as toxic and harmful gases, temperature and smoke, etc. are monitoring in real-time, so as to ensure the system to run efficiently and stably. [Conclusion] This system can be referenced for the development of security monitoring system in large-scale institutions and enterprises.

Key words Distributed laboratory safety monitoring, EPICS, Node.js, Real-time Web monitoring

中国科学院战略先导科技专项(No.XDA02010300)资助

第一作者: 李嘉曾, 女, 1994年出生, 2016年毕业于兰州大学, 现为硕士研究生, 研究方向为电子与通信工程

通信作者: 韩利峰, E-mail: hanlifeng@sinap.ac.cn

收稿日期: 2018-05-28, 修回日期: 2018-07-20

Supported by Strategic Pilot Science and Technology of Chinese Academy of Sciences (No.XDA02010300)

First author: LI Jiazeng, female, born in 1994, graduated from Lanzhou University in 2016, master student, major in electronic and communication engineering

Corresponding author: HAN Lifeng, E-mail: hanlifeng@sinap.ac.cn

Received date: 2018-05-28, revised date: 2018-07-20

中国科学院钍基熔盐堆核能系统(Thorium Molten Salt Reactor, TMSR)战略先导科技专项, 致力于研发第四代裂变反应堆核能系统。TMSR 核能专项近期的科技目标是建成钍基熔盐实验堆, 并形成支撑未来发展的技术研发能力^[1-2]。项目初期开展了放射化学、熔盐化学、高温材料、同步辐照、电解制氢、集热传热等专业学科研究实验, 在建设和运行过程中实验室的安全问题不容小觑, 需加紧规划构建实验室安全集中监控系统。经过对安全监控系统的设计需求详细评估后发现有限预算前提下开发相应的监控系统极具难度: 首先涉及到的 34 间实验室在园区中处于较为分散的分布状态, 地域覆盖广; 其次, 实验环境各异, 涉及材料、熔盐物、放化等实验室包含的信号类型繁多, 对视频、温度、烟雾、有毒有害气体、水流量、漏电保护等信号均有涉及; 软件接口同时涉及信号采集模块、火灾报警控制器、视频监控等多种接口; 最终要能实现在统一、可视化的软件平台下融合所有的信号系统^[3]。基于上述需求本文对信号类型进行了分类并相应各自分配经济的监测方案, 如输入/输出(Input/Output, IO)信号以开关量或模拟量的形式由探测器输出, 适合用分布式的网络 IO 模块实现采集。火灾报警信号的监测和记录因涉及到行业技术规范要求^[4], 适合直接接入园区内安防系统的火灾报警器。视频监控和记录适合采用商用低成本的视频系统。

针对上述系统特点, 综合网络及分布式方案, 我们选用 Node.js 平台。Node.js 采用 JavaScript 独特的事件系统以及非阻塞的 I/O 模型, 十分适合将实时数据密集型 web 应用部署在分布式设备中。该系统整合了视频监控、火灾监控和其它安全信号监控。采用本地和中控报警相结合的方式, 开发了一套广覆盖的实验室安全系统。该系统特点在于报警信号与视频调用相结合, 可以快速定位事件^[5-6], 并结合了历史数据库的访问, 为快速回调事故前期状态提供了可行性。本系统为 TMSR 系统的安全提供了可靠的保障, 且减少了值班人员冗余, 保障实验室监控状态的同时大幅度节省了人力成本。

1 系统结构

如图 1 所示, TMSR 实验室安全监控系统是一个整体网络化的结构, 具有专用的监控数据网络进行数据通讯。总值班室的客户端通过以太网获取服务器的数据更新。火灾烟雾探头通过通讯回路汇总到火灾控制器然后联入监控网络, 其它类型的信号探测器通过分布式的小型 IO 模块接入到监控网络。新式摄像头直接通过本地交换机接入到监控网络, 原有老式摄像机汇总到原有视频终端机也可以网络访问。服务器接入网络与设备进行通讯, 把实时数据发送到客户端, 并存储相应信号的历史数据。

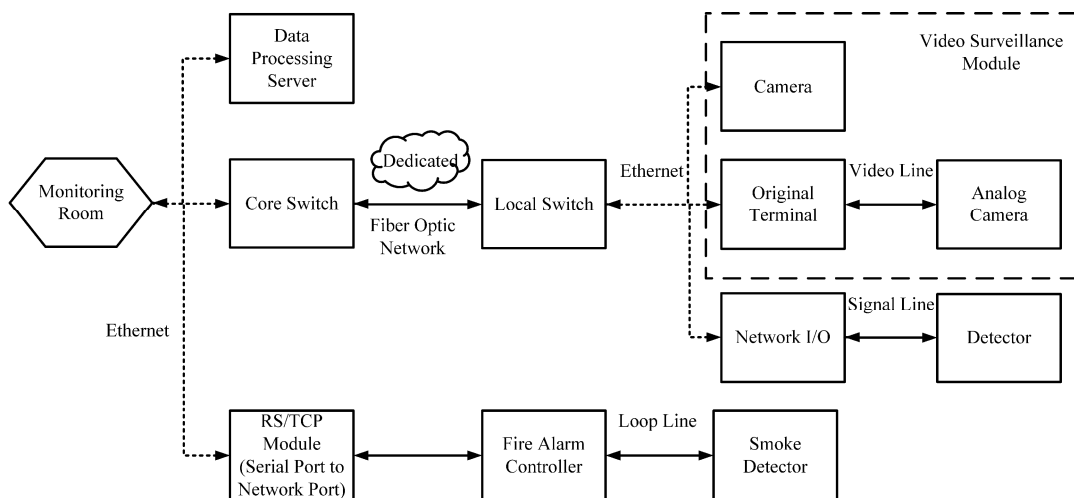


图 1 TMSR 实验室安全监控系统网络结构示意图

Fig.1 Schematic diagram of network structure of laboratory safety monitoring system in TMSR

1.1 系统软件架构

图 2 为 TMSR 实验室安全监控系统软件架构图。本系统整体采用 Node.js 软件平台, 将 Web 端、EPICS 系统和视频监控系统相融合, 使用 Node-express 模块架设 web 网站^[7], 利用 EPICS

(Experimental Physics and Industrial Control System) 接口模块实现与 EPICS 信号监控系统通讯^[8-10]、使用 Node-serialport 与实现火灾报警系统的通信, 通过 Node-ffi 自行封装所用软件的 SDK 并发布了视频摄像头的 API 接口模块实现视频采集。通过使用

Node-epics 调用 libCA 库实现与 EPICS 实时监控系统的对接；使用套接字 Socket 事件编程实现前后端视频图像的安全传输。系统在集中监控界面采用一

系列基于 JavaScript 的框架进行数据可视化的开发，使得界面更具备可读性。

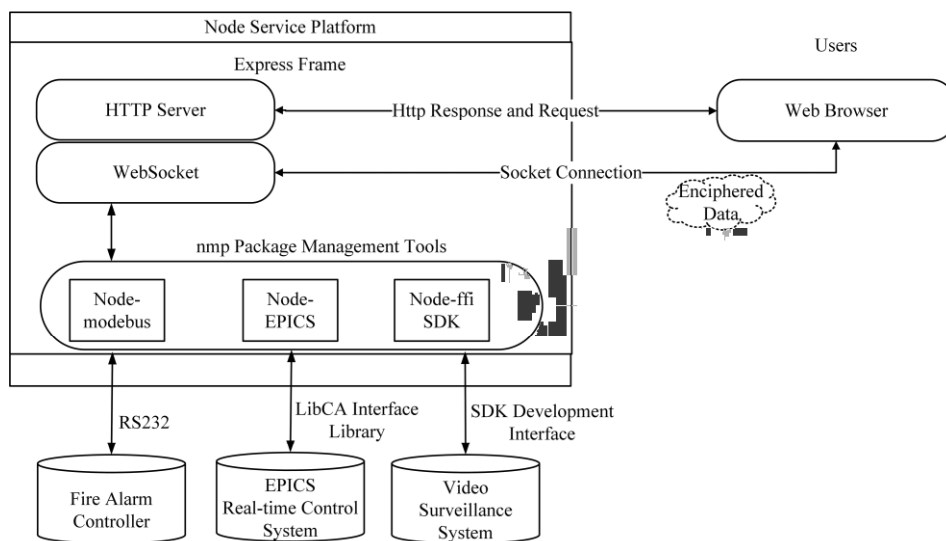


图2 TMSR 实验室安全监控系统软件架构
Fig.2 Software architecture of laboratory safety monitoring system in TMSR

1.2 系统前后端通讯

本系统的前后端通信过程如图 3 所示。系统采用基于 Node.js 的 socket.io 建立通信连接。该连接建立了可以双向传送数据的通道，在一次连接后可以复用。后端首先实例化 socket.io，利用其提供的 API 进行监听，前端在初始化 Web 通过浏览器输入 IP 地址访问服务器，由服务器发送页面触发前端页

面渲染执行脚本，并向后端 Socket 模块发送连接请求，连接完成之后，Socket 模块分别与 Node 中的 EPICS 模块、SDK 接口模块及 RS232 串口通讯模块相连，由它们分别向 Socket 模块发送 Socket 事件，当变量数值发生变化时，Socket 模块发送变量更新事件到前端，当前端监听到服务器发送的事件后执行回调函数，更新前端页面数据，实现实时监控。

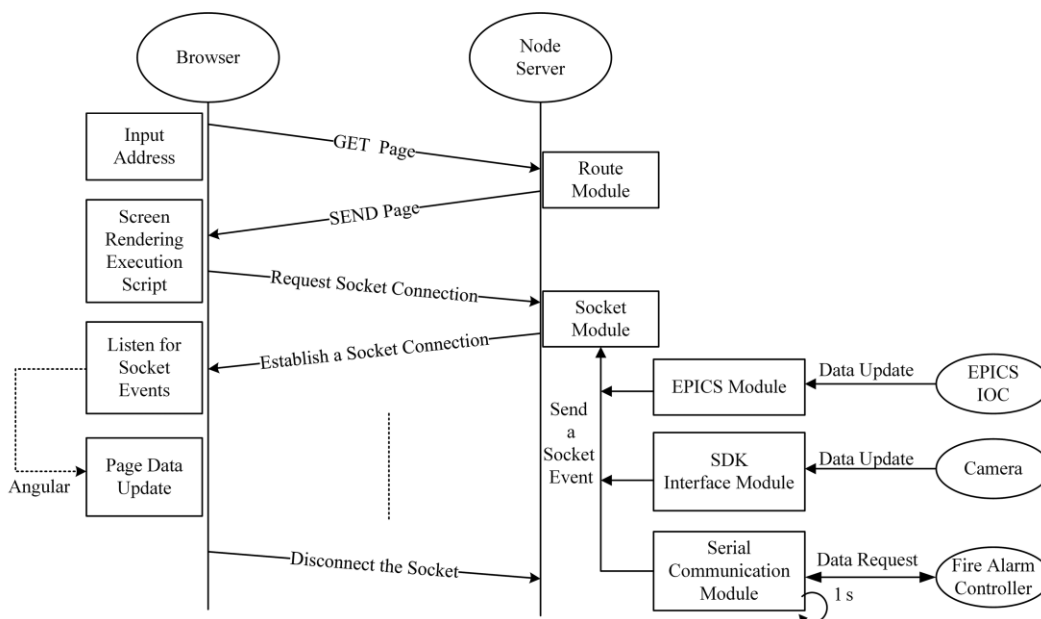


图3 网页前后端数据通讯流程图
Fig.3 Data communication flow chart of front and back end of web page

2 信号采集及监控过程

2.1 安全信号采集系统

信号采集模块连接有标准信号探测器用于实验室综合安全环境相关参数的检测,如:实验环境温度、火灾烟雾、有毒气体含量、加热炉温度、实验设备冷却水流量、可燃气体含量、毒物泄露、电器控制箱漏电报警、放射性剂量等;温度信号使用 Pt100 铂电阻的 RTD 温度传感器进行采集,气体分别采用 Mt500 固定式可燃性气体探测器和专用的 KD700HF 探头,流量监测采用 DN50 水流量计。各信号检测器采集的信号通过接入远程以太网 IO 设备 Moxa ioLogik E2160 接入监控网络。

该模块上层软件结合 TMSR 采用 EPICS 作为非安全级分布式控制系统的情况,为了保持统一协调且便于维护采用基于 EPICS 的软件开发框架。并采用紧凑型网络来实现园区较大范围分布的信号采集。监测数据的传输依赖 EPICS 的 IOC 事件驱动机制以 Node-EPICS 模块实现。信号通过交换机作为信息传输媒介再将信息传送到 EPICS 系统的 IOC (Input Output Controller)层进行广播,在寻找到目标 IOC 后建立点对点连接再从 EPICS IOC 端通过 CA 协议发送到后端服务器^[8-10]。

2.2 火灾报警系统

由于火灾报警控制器应答的信息长度不确定,根据实际发生报警的探头打包信息,不太适合使用固定信息长度的 EPICS 软件进行通讯^[11],所以直接调用 Node-serialPort 模块和火灾报警器进行通讯,调用方式参见文献^[12]。实验室内的烟感探头,通过总线回路统一接到火灾报警控制器。火灾报警器通过 RS232/TCP 模块接入到监控网络。服务器首先发起请求,火灾报警控制器确认主机的请求并进行响应,即采用查询/回应的通信方式进行对报警器的分布式远程控制。火灾报警控制器产生火灾报警信号时,显示终端软件会触发声音报警模块发出声音报警。

2.3 视频监控系统

视频采集模块主要通过安装的 50 个新式摄像头及原有摄像头全部接入基于 TCP 传输协议的以太网,其中摄像头采用 POE 供电。图 4 为本系统基于 Node.js 的实时视频监控系统架构图。

本系统使用 Node-ffi 去封装设备网络软件开发工具包(Software Development Kit, SDK)中的动态库和播放库 SDK 中的动态库,使其与 Node.js 兼容。

为了提高 Node.js 对视频数据的调用和处理性能,系统使用 Node.js 的 buffer 技术将其编码为 base64 格式的字符串,并启用 SDK 的播放库相关函数 (PlayM4_GetJPEG)抓图成 JPEG 格式,基于封装好的应用程序编程接口 (Application Programming Interface, API)建立 JavaScript 库 (HKload.js)来加载实时预览流程,输出视频帧图像数据。且使用 WebSocket 协议即套接字 (Socket.io) 传送视频数据可以保证实时性,在显示端使用 HTML5 (Hypertext Markup Language 5) 的 Canvas 标签,该标签通过绘制上下文来进行绘图及图像显示,我们使用 Canvas2D 在显示端实时显示视频帧图像^[13-14]。

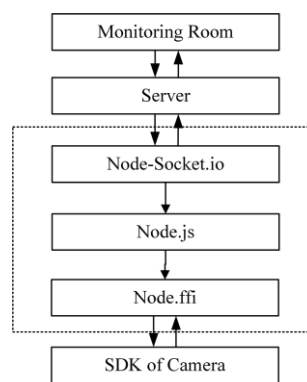


图 4 基于 Node.js 的实时视频监控系统架构图
Fig.4 Architecture diagram of real-time video surveillance system based on Node.js

本安全监控系统为了满足较高的带宽需求,采用单端口 $100 \text{ Mbit} \cdot \text{s}^{-1}$, 上行带宽 $1 \text{ Gbit} \cdot \text{s}^{-1}$ 的配置;监控系统采用的单个 400 万像素摄像头的视频数据经过 H265 标准编码后的消耗带宽为 $1.5 \text{ Mbit} \cdot \text{s}^{-1}$, 60 个摄像头的消耗带宽低于 $100 \text{ Mbit} \cdot \text{s}^{-1}$ 。且监控系统采用独立网络,其他数据采集模块的信号通讯占用带宽不足 $1 \text{ Mbit} \cdot \text{s}^{-1}$ 。因此系统的网络性能远高于消耗带宽,可保证系统的稳定运行。

2.4 数据归档功能

历史数据对于安全系统而言也至关重要的,一方面历史数据可以用来查看过去某段时间系统运行的各个参数,帮助进行安全隐患的判断及定位,另一方面可以提供过去事故的时间节点及频率等,有利于对实验室布局的优化,提高警惕性并提供可靠的经验^[15]。对于监控系统,每个子系统都具备数据归档功能,方便事故回放与分析。

对于流量、气体、温度等安全信号监控系统,本系统的数据处理服务器与 EPICS 的历史归档引擎插件 RDB Archive Engine 相结合将实时参数数据保存到关系型数据库 postgresql 中,并在系统中连接

到了 Navicat 数据库管理工具，可快速直观的访问到历史数据库。根据时间戳可在数据库查询对应时间的相应信号数据。对于火灾报警控制器而言，由于没有与 EPICS 系统进行对接，故直接将数据记录在控制器中按时间的倒序记录警报，最多可以记录 1000 个监管历史数据。

对于视频监控系统而言，若在数据库存储只能将视频转化为二进制存储在表中，调用查看时需要再将二进制转换回 jpg 格式，由于数据量的庞大，查询性能将大大降低。DVR 存储是目前最常见的一种存储模式，解编码器设备直接挂接硬盘，使用方便，技术发展成熟，故本系统采用数字硬盘录像机 (Digital Video Recorder, DVR) 即内置硬盘的方式进行图像存储，容量为 3000 G，存储周期达 3 周。

3 系统联动

现有的实验室安全系统通常将设备信号监控及视频监控独立开来，没有相互结合，多个实验室监控影像被集中于一台设备上排列显示，并同时轮动，警报发生时很难快速预测及定位到事故发生的地点，为了方便安全人员快速准确的定位警报事件，本系统在 Node.js 整体软件框架下将系统进行了联结。通过开发摄像头的 Node.js 接口函数，订阅视频硬件数据更新，利用 WebSocket TCP/IP 通道通知数据更新，并使用 HTML5 Canvas 标签进行视频流的展示^[16]。警报发生时，Node.js 服务器将报警信号加工后发送到客户端，此时客户端将弹出新的界面，新的浏览界面将发送 Get 信号到后端，后端接收后

通过 WebSocket 协议再发送到 Web 端，从而触发视频页面的调用，使得安全监控操作更加简便快速，大大降低实验室损坏的风险。

4 集中监控界面

系统集中监控界面如图 5 所示，本系统将所有信号以网页的形式直观的提供给监控人员。对于显示安全信号及火警信号的前端页面，页面程序采用 HTML5、CSS、JavaScript 等技术，为了展示信号趋势，引入了 echart 图表及 DataTable 技术，并结合园区实况引入地图 API 等，将数据汇总可视化。

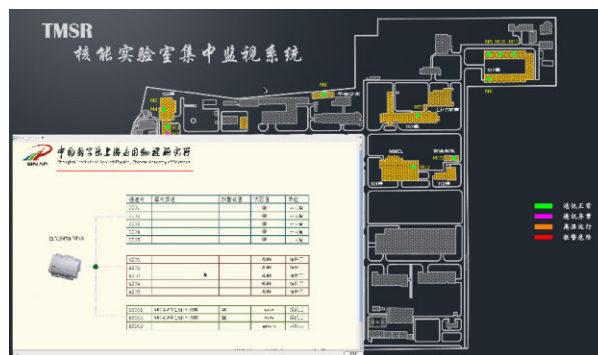


图 5 集中监控系统界面
Fig.5 Centralized monitoring interface

图 6 为系统集中监控报警界面，监控系统运行时一旦某个监测点发生异常状态，将在园区地图对应位置闪烁警示，并在该监测点地图上跳动显示标注图标、弹出消息框及响起警报。



图 6 集中监控系统报警界面
Fig.6 Centralized monitoring alarm interface

对于视频监控界面来说，采用 HTML5 的 Canvas 标签，Canvas 主要提供画布的功能，无需插

件，且对于不同设备上的浏览器有更好的兼容性，更加适合于移动设备以及跨平台设备的开发。视频

界面的设计首先创建图像对象来接受来自服务端的视频数据, 将通过图像源的方式将 base64 编码格式的图像数据嵌入到图像对象中, 考虑实时视频监控本身特点, 将图像数据通过 Canvas2D 的绘制上下文进行显示。

5 结语

本文紧密结合了实验室环境实况, 综合使用了 EPICS、Node.js、web 技术开发了一套实验室安全监控系统。该系统特点在于将警报及视频页面的调用相结合, 开发了相应的 web 界面提高了直观性, 并具备历史数据归档功能, 综合解决了实验室分布分散、安全信号复杂等问题, 实现了高效的集中管理, 降低了人力成本, 为大型实验室的安全监控系统构建提供参考。

参考文献

- 1 蔡翔舟, 戴志敏, 徐洪杰. 钍基熔盐堆核能系统[J]. 物理, 2016, **45**(09): 578–590.
CAI Xiangzhou, DAI Zhimin, XU Hongjie. Thorium molten salt reactor nuclear energy system[J]. Physics, 2016, **45**(09): 578–590.
- 2 江绵恒, 徐洪杰, 戴志敏. 未来先进核裂变能——TMSR 核能系统[J]. 中国科学院院刊, 2012, **27**(3): 366–374.
JIANG Mianheng, XU Hongjie, DAI Zhimin. Advanced fission energy program-TMSR nuclear energy system[J]. Bulletin of Chinese Academy of Sciences, 2012, **27**(3): 366–374.
- 3 雷蕾, 韩利峰, 徐海霞, 等. EPICS 环境下的软件规范管理[J]. 核技术, 2015, **38**(6): 060501. DOI: 10.11889/j.0253-3219.2015.hjs.38.060501.
LEI Lei, HAN Lifeng, XU Haixia, *et al.* Software management in EPICS environment[J]. Nuclear Techniques, 2015, **38**(6): 060501. DOI: 10.11889/j.0253-3219.2015.hjs.38.060501.
- 4 GB 4717-2005: 火灾报警控制器[S]. 北京: 中国标准出版社, 2005.
GB 4717-2005: Fire alarm controller[S]. Beijing: Standards Press of China, 2005.
- 5 李玉, 刘铁根, 王绍俊, 等. 全光纤分布式视频联动长距离周界安防监控系统[J]. 光电子·激光, 2013, **24**(9): 1752–1757.
LI Yu, LIU Tiegeng, WANG Shaojun, *et al.* All fiber distributed long-distance perimeter security monitoring system with video linkage function[J]. Journal of Optoelectronics-laser, 2013, **24**(9): 1752–1757.
- 6 Gao H, Yuan L, Wang T. The family safety monitoring system based on ethernet and Bluetooth[C]. Intelligent Control and Automation, 2010: 4281–4285.
- 7 Brown E. Web development with node and express: leveraging the JavaScript stack[M]. O'Reilly Media, Inc., 2014.
- 8 Han L F, Chen Y Z, Cai J, *et al.* The application of EPICS in TMSR radiation protection and access control system[J]. Nuclear Science and Techniques, 2016, **27**(2): 41. DOI: /10.1007/s41365-016-0040-7.
- 9 Hanks J. Realtime display of EPICS data using HTML5 and Websockets[C]. EPICS Collaboration Meeting, Oak Ridge National Laboratory, US, 2016.
- 10 Uchiyama A, Furukawa K, Higurashi Y. EPICS channel access using WebSocket[C]. Proceedings of the 9th International Workshop on Personal Computers and Particle Accelerator Controls, US, 2012.
- 11 韩利峰, 陈永忠, 周大勇, 等. HTS 熔盐实验回路分布式控制系统设计[J]. 核技术, 2013, **36**(9): 090603. DOI: 10.11889/j.0253-3219.2013.hjs.36.090603.
HAN Lifeng, CHEN Yongzhong, ZHOU Dayong, *et al.* Design of the distributed control system for HTS molten salt test loop[J]. Nuclear Techniques, 2013, **36**(9): 090603. DOI: 10.11889/j.0253-3219.2013.hjs.36.090603.
- 12 Tom Igoe. Serial Communication with Node.js[EB/OL]. 2014-10-03[2018-05-05]. <https://itp.nyu.edu/physcomp/labs/labs-serial-communication/lab-serial-communication-with-node-js/>.
- 13 房峰, 高美凤. 基于 Node.js 的移动视频监控系统[J]. 计算机系统应用, 2017, **26**(10): 281–285.
FANG Feng, GAO Meifeng. Mobile video monitoring system based on Node.js[J]. Computer System and Applications, 2017, **26**(10): 281–285.
- 14 Cantelon M, Harter M, Holowaychuk T J, *et al.* Node.js in action[M]. Manning Publications, 2017.
- 15 Hu Z, Mi Q R, Zheng L F, *et al.* EPICS data archiver at SSRF beamlines[J]. Nuclear Science and Techniques, 2014, **25**(2): 020103. DOI: 10.13538/j.1001-8042/nst.25.020103.
- 16 Zhao S M, Xia X L, Le J J. A real-time web application solution based on Node.js and WebSocket[J]. Advanced Materials Research, 2013, **816**: 1111–1115. DOI: 10.4028/www.scientific.net/AMR.816-817.1111.