

**嵌入式系统开发学期**

**实训报告**

项目名称 智慧校园仿真应用系统

|  |  |
| --- | --- |
| 作者姓名 | 杨雨婷 |
| 学号 | 2017115050614 |
| 所在院系 | 计算机与信息工程学院 |
| 学科专业名称 | 物联网工程 |
| 班级 | 1706班 |
| 导师 | 霍浩，胡克伟 |
| 答辩时间 | 2019年12月 29 日 |

目 录

[1 前言 1](#_Toc28038)

[2 需求分析 1](#_Toc24580)

[3 总体设计 2](#_Toc30443)

[3.1 系统目标 2](#_Toc19073)

[3.2 构建开发环境 3](#_Toc7312)

[3.3 系统功能结构 4](#_Toc16091)

[4 数据库设计 4](#_Toc20933)

[4.1 数据库概要说明 4](#_Toc8726)

[4.2 数据库E-R图 5](#_Toc29350)

[4.3 数据表结构 5](#_Toc26209)

[5 接口设计 6](#_Toc7455)

[5.1软件接口 7](#_Toc15494)

[5.2硬件接口 10](#_Toc26399)

[6 程序主要系统开发 15](#_Toc4574)

[6.1 系统框架 16](#_Toc817)

[6.2 Web应用层设计 16](#_Toc18166)

[6.3 中控服务系统设计 17](#_Toc12057)

[6.4 数据采集系统设计 18](#_Toc11335)

[7 系统出错处理设计 18](#_Toc25644)

[8 项目总结 19](#_Toc32165)

智慧校园仿真应用系统

杨雨婷

（湖北师范大学计算机与信息工程学院 中国 黄石 435002）

# 1 前言

随着物联网技术的发展，以及移动应用市场的重大变革，现在的生活渐渐发展为以移动为中心的智能多元化生活，人们可以通过手中的移动设备或互联网随时随地监控家里的状况、远程管理和控制家用电器或设备，智能化的生活逐渐被更多的人所接受和使用。

正是在此大的行业背景下，结合功能强大的Castwise-Z7嵌入式全可编程应用开发实训平台，提出了《智慧校园仿真应用系统》这一项目，旨在学习和实践新兴的物联网技术，掌握新的行业动向。

# 2 需求分析

程序主界面如图2-1所示。为登录界面。

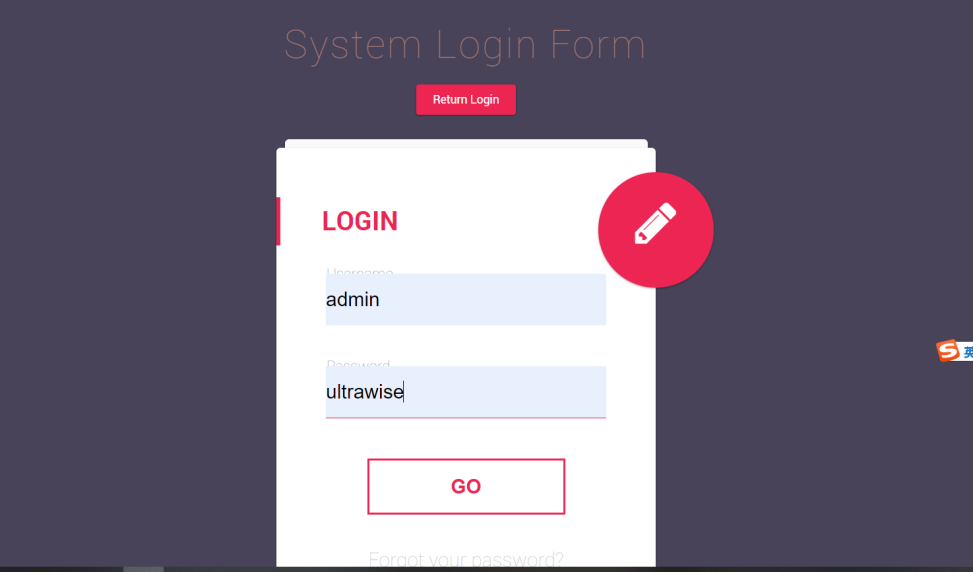


图2-1 仿真系统登录界面

介绍界面：

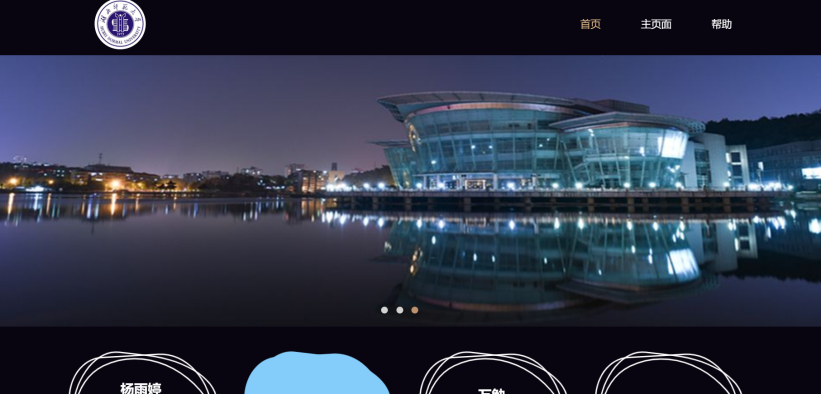


图2-2 登录主介绍界面

我的介绍界面：

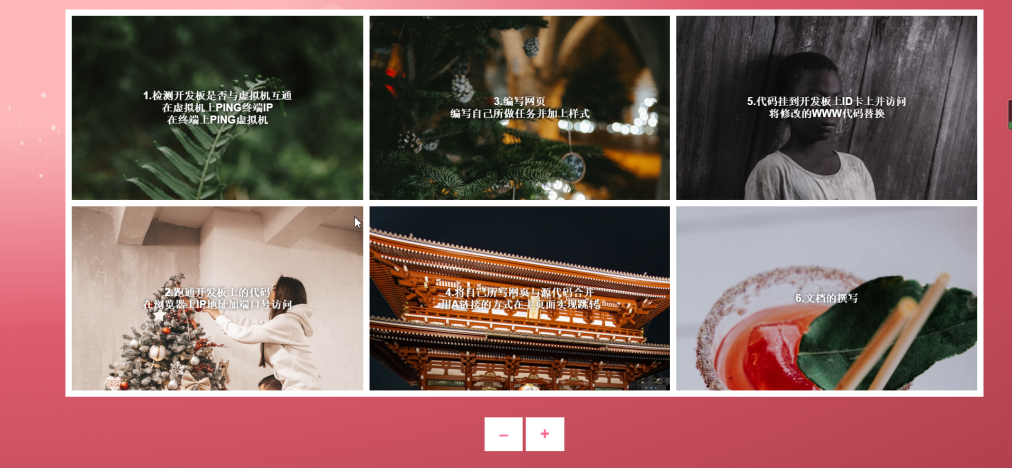


图 2-3 我的介绍界面

系统软件界面：



图 2-4 系统软件界面

# 3 总体设计

## 3.1 系统目标

该项目是基于Castwise-Z7嵌入式全可编程应用开发实训平台的数据采集和环境监控系统，系统应具有如下功能：

（1）环境监测

通过收集智能传感器和Zigbee无线网络采集的环境数据，以Web页面方式显示。主要包括：

1）温度监测

通过温度传感器采集环境的温度，经过数据处理后以页面方式显示。

2）湿度监测

通过湿度传感器采集环境的湿度，经过数据处理后以页面方式显示。

3）风速监测

通过风速传感器采集风速仪的转速，经过数据处理后以页面方式显示。

4）风向监测

通过风向传感器采集风向仪的方位值，经过数据处理后以页面方式显示。

（2）视频监控

通过网络摄像头实时采集视频数据流，当用户通过客户端访问视频监控页面时可调用摄像头查看所监控的环境情况。

1. 条码操作

1）二维码操作

在二维码操作页面上输入任意字符，可生成二维码图片，并能够保存和打印。

2）条形码操作

在条形码操作页面上输入任意数字，可生成条形码图片，并能够保存和打印。

## 3.2 构建开发环境

### 2.3.1硬件配置

**Web服务端：**

1. 设备名称：Castwise-Z7嵌入式全可编程应用开发实训平台
2. CPU处理器：ARM Cortex-A9 1GHz
3. 内存：1GB DDR3 SDRAM
4. 存储器：4GB eMMC高速Flash
5. 显示分辨率：1280\*720P

**PC客户端：**

1. CPU处理器：Intel Pentium 4 1.6GHz以上
2. 内存：1GB以上
3. 硬盘：80GB以上
4. 显示器分辨率：1280\*720P以上

### 2.3.2软件配置

**Web服务端：**

1. 操作系统：Linux 3.5.0
2. Web服务器：www

**PC客户端：**

1. 操作系统：Windows XP/7/8兼容操作系统
2. 浏览器：Google Chrome/Mozilla Firefox兼容浏览器
3. 播放器：VLC Media Player 2.1.3

## 3.3 系统功能结构



图 1 系统功能结构

# 4 数据库设计

## 4.1 数据库概要说明

在设计数据库时，考虑到系统主要应用于嵌入式ARM开发平台，且从易于开发和管理的角度考虑，系统选择采用SQLite作为后台数据库。

根据系统的功能需要，设计了一个info数据库，在其中设计了4个关系表：user\_info, sys\_info, style\_config, monitor\_config 分别对应系统的不同功能。

## 4.2 数据库E-R图

## 4.3 数据表结构

在SQL Server 2008 R2中创建数据库，

**（1）用户信息表**

|  |  |  |
| --- | --- | --- |
| user\_info表 | | |
| 列名 | 数据类型 | 允许空 |
| user\_id | int |  |
| name | nvarchar(255) | √ |
| password | nvarchar(255) | √ |
| email | nvarchar(255) | √ |
| permission | int | √ |

表1：用户信息表

user\_info表，对应于系统登陆login和用户注册register中间件。此表设计的目的是存储系统中远程用户的注册信息，包括了表中具有唯一性的user\_id值（主键）、用户名称，用户密码，电子邮件地址和权限。当远程用户登陆到本系统的登陆页面时，系统会要求其登陆用户名和密码，用户输入后会将所得信息与该表的name和password对比，若一致则通过，否则将显示错误信息。

**（2）系统信息表**

|  |  |  |
| --- | --- | --- |
| sys\_info表 | | |
| 列名 | 数据类型 | 允许空 |
| machine\_name | nvarchar(255) | √ |
| product\_id | nvarchar(255) | √ |
| hardware\_id | nvarchar(255) | √ |
| software\_id | nvarchar(255) | √ |
| serial\_num | nvarchar(255) | √ |

表2：系统信息表

sys\_info表，对应于系统设置setup中间件。此表设计的目的是用来存储系统信息，包括主机名称、产品型号、硬件ID、软件ID和序列号

**（3）界面风格表**

|  |  |  |
| --- | --- | --- |
| style\_config表 | | |
| 列名 | 数据类型 | 允许空 |
| page\_style | int |  |

表3：用户信息表

style\_config表，对应于系统设置setup中间件。此表设计的目的是用来存储系统界面风格的配置信息

**（4）监控配置表**

|  |  |  |
| --- | --- | --- |
| monitor\_config表 | | |
| 列名 | 数据类型 | 允许空 |
| video\_1 | nvarchar(255) | √ |
| video\_2 | nvarchar(255) | √ |
| video\_3 | nvarchar(255) | √ |
| video\_4 | nvarchar(255) | √ |
| video\_5 | nvarchar(255) | √ |
| video\_6 | nvarchar(255) | √ |

表4：用户信息表

monitor\_config表，对应于系统设置setup中间件。此表设计的目的是用来存储视频监控的配置信息（视频监控地址参数）

表4-1 菜单表

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名称 | 字段类型 | 键/索引 | 备注 |
| id | int | 主键 | 编号 |
| MenuName | varchar(50) |  | 菜单名称 |
| MenuCaption | nvarchar(50) |  | 菜单标题 |
| MenuId | int | 索引 | 菜单编号 |
| MoudleId | int | 外键 | 模块编号 |

# 5 接口设计

## 5.1软件接口

1. CGI命令控制程序

1）解析数据函数

函数：int ParseData(char \*filename, struct data \*js\_data)

作用：从Web应用层CGI请求中获取并解析JSON格式的设备操作数据

2）增加命令队列函数

函数：int addCmdQueue(queue\_data\* p\_map, struct data \*js\_data)

作用：把设备操作数据包添加到共享内存队列中，并作新增操作标志，即向命令解析服务器发送新增操作命令

3）获取返回结果函数

函数：int getResultQueue(int pos, queue\_data\* p\_map, struct redata \*re\_data)

作用：轮询等待设备操作后的返回结果数据

4）返回结果数据函数

函数：int ReturnData(struct redata \*re\_data)

作用：解析返回的结果数据，并按照Web应用层的要求分割返回数据，并打包成JSON格式返回给应用层

1. parseServer命令解析程序
2. 初始化共享队列函数

函数：int InitializeQueue(queue\_data\* q\_map)

作用：初始化共享内存队列

1. 打开串口函数

函数：int open\_port(int com\_port)

作用：打开串口

1. 设置串口参数函数

函数：int set\_com\_config(int fd,int baud\_rate, int data\_bits, char parity, int stop\_bits)

作用：设置串口参数，实现中控服务系统与数据采集系统之前的数据通信

1. 队列管理线程函数

函数：void \*QueueManager(void \*arg)

作用：管理3个队列管理线程

（CleanupManager、TimeoutCounter、TimeoutManager）

1. 清除队列线程函数

函数：void \*CleanupManager(void \*arg)

作用：清除共享内存队列中超时未完成的操作数据包

1. 超时计数器函数

函数：void \*TimeoutCounter(void \*arg)

作用：循环检测共享队列，并启动超时计数器

1. 队列超时管理函数

函数：void \*TimeoutManager(void \*arg)

作用：循环监测共享队列，管理超时的数据包，并相应标志状态

1. common公共函数库

自定义操作数据包结构体

typedef struct

{

int used\_stat; //队列状态

int sandtable\_id; //沙盘ID

int sandtable\_node\_id; //沙盘节点ID

char operate\_code[128]; //操作命令码

char operate\_data[128]; //操作命令参数

char result\_code[128]; //返回状态码

char result\_data[128]; //返回数据

int cmd\_input\_stat; //输入态

int cmd\_send\_stat; //发送态

int cmd\_result\_stat; //结果态

int cmd\_finish\_stat; //完成态

int cmd\_remove\_stat; //删除态

int time\_out; //超时计数器

}queue\_data;

1. login系统登陆中间件
   1. 检查用户名和密码函数

函数：int check\_user\_and\_pwd(sqlite3 \*db,char \*name ,char\* pwd)

作用：查询当前用户名和密码是否与数据库中的记录相匹配

1. register用户注册中间件
   1. 创建数据库函数

函数：int db\_create(sqlite3 \*db,char \*pTable\_name,int column\_num,COLUMN column[])

作用：创建用户记录数据库文件

* 1. 新增用户记录函数

函数：int db\_insert(sqlite3 \* db, char \* pTable\_name, int column\_num, COLUMN column[])

作用：在数据库中插入一条用户记录

* 1. 检查用户名记录函数

函数：int check\_name\_exist(sqlite3 \*db,char \*name)

作用：判断当前输入用户名是否已存在数据库中

1. setup系统设置中间件
   1. 获取系统信息函数

函数：int get\_system\_info()

作用：查询数据库，获取当前系统信息

* 1. 设置系统信息函数

函数：int set\_system\_info()

作用：获取应用层输入的系统参数，修改当前系统信息，并保存于数据库

* 1. DHCP动态配置网络函数

函数：int set\_dhcp\_network()

作用：采用DHCP方式动态配置网络，并保存网络信息到配置文件

* 1. 手动配置网络函数

函数：int set\_manual\_network()

作用：获取应用层输入的网络参数，采用手动方式配置网络，并保存网络信息到配置文件

* 1. 获取网络配置函数

函数：int get\_network\_config()

作用：从当前系统中动态获取网络配置信息，并返回应用层

* 1. 获取界面风格函数

函数：int get\_style\_config()

作用：从数据库中获取当前界面设置参数，并返回应用层

* 1. 设置界面风格函数

函数：int set\_style\_config()

作用：获取应用层输入的界面参数，修改当前系统界面风格，并保存参数到数据库

* 1. 获取视频监控配置函数

函数：int get\_monitor\_config()

作用：从数据库中获取视频监控配置参数，并返回应用层

* 1. 设置视频监控配置函数

函数：int set\_monitor\_config()

作用：获取应用层输入的视频监控配置参数，修改视频监控地址，并保存参数到数据库

## 5.2硬件接口

### 5.2.1 嵌入式硬件平台

Castwise-Z7嵌入式全可编程应用开发实训平台采用基于Xilinx Zynq-7000 AP SoC 系列芯片Z-7020（XC7Z20-1CLG400）作为实训平台的核心处理器，该处理器采用最新ARM-Cortex™-A9双核和Artix™-7 FPGA架构，其FPGA可编程逻辑功能使得整个系统拥有丰富的可拓展能力。处理器含有ARM双核Cortex™-A9处理器（主频高达1GHz），并且支持NEON™ Processing / FPU Engines，其拥有强大的音视频处理能力。处理器提供多种外围设备I/O接口，如USB、SPI、SDIO、CAN、UART和I2C等接口。

Castwise-Z7开发平台的功能原理图如下：



图：Castwise-Z7原理框图

### 5.2.2 硬件平台技术参数

|  |  |
| --- | --- |
| **核心资源** | |
| 处理器 | * Zynq™-7000 AP SoC XC7Z020-1CLG400 * 采用最新ARM Cortex™-A9 双核+ Artix™-7 FPGA架构 * 主频高达1G * 支持NEON™ Processing / FPU Engines * 支持VFPU 浮点运算 * 256KB的片上存储空间 * FPGA 包含85K逻辑单元 * FPGA 包含 560K BANK RAM * FPGA 支持2路 2X12bit 的ADC |
| 存储器 | * 1GB DDR3 SDRAM (1066 Mbps) * 128Mb QSPI Flash * 4GB eMMC Flash * 4 GB SD 卡 |
| 音视频 | * 24-bit立体声音频编解码器 * HDMI 输出 (支持1080p60 + audio) * 支持VGA (12 bit color video output)输出 * 支持I2C Audio Codec，立体声 48K录制和播放音频，采样频率范围8KHz~96KHz，支持Mic In，Line In,Line Out，Headphone 输入输出 |
| 模拟输入 | * 支持 4 路模拟信号输入通道（2 路差分 / 2路单通道） |
| 网口 | * 一路10/100/1000 Mbps 以太网口 |
| 配置与调试 | * 一路USB OTG接口 * 一路JTAG调试接口 * 一路UART RS232 DB9串口 * 启动模式配置开关 |
| 电源与指示 | * 电源开关 * 复位开关 * 电源指示灯 * 编程指示灯 |
| 外围设备I/O接口 | * 支持I2C接口 * 支持SPI接口 * 支持SDIO接口 * 支持CAN接口 * 支持UART接口 |
| **功能模块** | |
| 传感器模块 | 该模块包含如下传感器：   * 光强度传感器（测量范围1-70000lx，精度1lx） * 数字温度传感器（测量范围 －55℃～+125℃，测量精度低至0.0625℃） * 温湿度传感器（测湿度范围20－90％RH 精度 ±5％RH，测温范围0－50℃ ，精度 ±2℃ ） * 脉冲测湿度传感器（测量范围20…95%RH精度 ±5%RH） |
| 模数转换模块 | 该模块包含：   * 12位精度内部ADC * 测量范围0-1V,测量精度244uv * 一路PT100湿度采集模块，采集范围-200℃-660℃ * 一路滑动变阻器电压采集模块，采集范围0-2.5v |
| 键盘字符模块 | 该模块包含如下：   * OLED显示屏（分辨率128 x 32，0.91寸） * 4\*4矩阵键盘 * 1个4位数码管7段数码管 |
| 通用接口模块 | 该模块包含如下：   * 4个LED * 4个拨码开关 * 4个按键 * 1个蜂鸣器 |
| 摄像头模块 | * 30万 CMOS摄像头，最大分辨率1,600 x 1,200 pixels，支持RawRGB、RGB(GBR4:2:2、565/555/444)、YUV(4:2:2)和YCbCr(4:2:2)输出格式 |
| 电容触摸显示屏模块 | * 24位7寸LCD显示屏，分辨率800\*480 * 电容触摸屏 |
| 通信模块 | 该模块包含如下通信模块：   * CAN通信接口，支持ISO 11898 -1，传输速度高达1 Mb/s * RS232 DB9 串口通讯接口 * 蓝牙模块，支持蓝牙3.0 / 4.0+ EDR ，支持蓝牙4.0 BLE通讯协议与3.0 SPP 串行通讯协议 |
| 电机模块 | 该模块包含如下：   * 板载电机驱动电路，预留了接口，方便外接其它电机模块 * 一个直流电机带红外测速模块 |
| 面包板模块 | * 将Zynq重要的资源管脚引到面板上，方便用户做实验 |
| 多路开关模块 | * 12路LED * 12路拨码开关 |
| WIFI模块 | * 支持USB接口 * 网络标准：IEEE 802.11n，IEEE 802.11 * 传输速率：300Mbps * 频率范围：单频(2.4-2.4835G) |
| 无线传感器传输模块 | * 支持Zigbee协议栈 * 基于IEEE802.15.4标准的低功耗[局域网](http://baike.haosou.com/doc/3165868.html" \t "_blank)协议 * 支持温湿度、光强度、压强等传感器 |
| 串口模块 | * 支持USB接口 * 支持RS232 DB9接口 |

表1：Castwise-Z7硬件平台技术参数

**（3）Zigbee协调器/控制器**

负责Zigbee数据采集设备之前的数据传输，通过Zigbee无线网络方式。



图：Zigbee无线协调器/控制器

**（4）Zigbee温湿度仪**

采集温湿度数据，并通过Zigbee无线网络方式传输。



图：Zigbee温湿度仪

**（5）Zigbee风速仪**

采集风速数据，并通过Zigbee无线网络方式传输。



图：Zigbee风速仪

**（6）Zigbee风向仪**

采集风向数据，并通过Zigbee无线网络方式传输。



图：Zigbee

# 6 程序主要系统开发

## 6.1 系统框架

智能数据采集系统的系统框架共划分为三个层次，从上至下分别为：Web应用层、中间件层和数据采集层，系统框图如下：



图：智能数据采集系统-系统框架

## 6.2 Web应用层设计

Web应用层的主要任务是响应客户端浏览器发送过来的HTTP和CGI请求；提交从中间件层所获取的设备层的传感器采集数据；提供用户登陆、用户注册、设备操作、视频监控、条码操作、系统设置等界面；将用户信息和系统配置写入数据库等。网页前端层是直接与用户交互的一层。



图：Web应用层-系统架构

## 6.3 中控服务系统设计

中间件层，即中控服务系统层。包括Appweb服务器、CGI命令控制器和命令解析服务器。其主要任务是解析Web应用层发送过来操作命令，并传递给Zigbee网络层；从网络层获取设备层采集的数据，打包成JSON格式的字符串，实现对设备层采集数据的处理，并按应用层兼容的格式进行打包，然后返回给Web应用层显示。

Appweb是一个嵌入式HTTP Web服务器，它可以直接集成到客户的应用和嵌入式设备上，便于开发和部署基于Web的应用程序和设备。本项目中的Web应用层就是基于Appweb服务器运行的网页前端应用。

CGI命令控制器的作用是响应客户端浏览器发送过来的CGI命令请求，解析命令请求，打包成操作数据包，并把它加入共享内存队列，便于命令解析服务器使用。

命令解析服务器从共享内存队列中获取操作数据包并解析成数据采集层能够识别的格式，发送至串口，与数据采集层交互通讯。

视频监控模块直接采用视频播放插件与Web页面交互，即在客户端浏览器上安装VLC Media Player播放器插件后，在Web应用层的视频监控页面上直接播放视频。

系统登陆、用户注册和系统设置这3个模块分别对应相应的中间件软件，且底层采用SQLite嵌入式数据库作为存储机制，实现完整的系统交互。



图：中控服务系统--系统框图

## 6.4 数据采集系统设计

数据采集层包括由Zigbee协调器和Zigbee控制器组成的无线网络和嵌入在Zigbee数据采集设备中的智能传感器共同组成。每个Zigbee控制器对应一个或一组传感器设备。

Zigbee协调器作为无线网络中的路由器，它上层对接中控服务系统，从串口接收中控发来的命令并广播发送至Zigbee网络，使设备节点可以接收命令；另一方面，从Zigbee网络接收设备节点返回的操作结果数据，并通过串口回送给中控服务系统。

Zigbee控制器监听网络上的命令请求，并根据设备节点号匹配属于自己的操作请求，并根据命令执行对应的设备操作：数据采集、设备重启、检测设备状态等，然后获取智能传感器的返回数据或设备状态，并再次通过Zigbee网络回送给协调器。



图：数据采集系统--系统框图

# 7 系统出错处理设计

智能数据采集系统中设置了一些常见错误现象和其处理方式，如下表所示：

|  |  |  |
| --- | --- | --- |
| **错误类型** | **错误现象** | **处理** |
| 客户端问题 | 浏览器无法正确加载网页 | 更换兼容的浏览器版本 |
| 软件问题 | 浏览器无法加载更新数据 | 自动刷新处理 |
| 软件问题 | 监控视频无法播放 | 播放窗口显示黑色 |
| 设备错误 | 设备操作异常/失效 | 提示设备操作失败 |

表1：出错处理设计

# 8 项目总结

《智慧校园仿真应用系统》项目运行在Linux系统中，所以在硬件方面需要底层硬件板卡的支持，在硬件板卡中需求配置网口，USB接口，音频接口等外部接口设备。并且在其中还的配有专门的温度，湿度，风速、风向等传感器硬件设备。在软件方面，需要对Linux系统进行定制移植。在Linux系统中需要移植小型的Web服务器，开发Web前端应用层。