# 目 录

[第一章 面向IPv6无线传感网的信息收集平台概述 2](#_Toc398038692)

[1.1面向IPv6无线传感网的信息收集平台需求性分析概述 2](#_Toc398038693)

[1.1.1 功能需求 2](#_Toc398038694)

[1.1.2 界面需求 2](#_Toc398038695)

[1.2 面向IPv6无线传感网的信息收集平台的实现概述 3](#_Toc398038696)

[第二章 面向IPv6无线传感网的信息收集系统硬件环境 6](#_Toc398038697)

[2.1无线传感器网络节点 6](#_Toc398038698)

[2.2 Smeshlink IPv6 MXG300智能网关 8](#_Toc398038699)

[2.3 USB无线网卡(STICK) 12](#_Toc398038700)

[2.4 JTAG介绍与使用 13](#_Toc398038701)

[2.5 MXP540 USB串行功能扩展板介绍与使用 15](#_Toc398038702)

[第三章 面向IPv6无线传感网的信息收集系统软件环境 17](#_Toc398038703)

[3.1 Contiki Studio集成环境开发平台 17](#_Toc398038704)

[3.2 WinAVR和AVR studio软件开发平台 20](#_Toc398038705)

[3.3面向IPv6无线传感网的信息收集系统开发工具 21](#_Toc398038706)

[3.3.1 Microsoft Visual Studio 2005介绍 22](#_Toc398038707)

[3.3.2 C#编程语言介绍 23](#_Toc398038708)

[3.3.3 mysql数据库介绍 23](#_Toc398038709)

[3.4面向IPv6无线传感网的信息收集系统设计 25](#_Toc398038710)

[3.4.1系统分层架构设计 25](#_Toc398038711)

[3.4.2系统功能 25](#_Toc398038712)

[3.4.3基本功能需求 26](#_Toc398038713)

[3.4.4系统主要功能的流程图 26](#_Toc398038714)

[3.4.5数据库逻辑结构设计 29](#_Toc398038715)

[3.4.6 数据库结构设计表 30](#_Toc398038716)

[3.5面向IPv6无线传感网的信息收集系统具体实现 31](#_Toc398038717)

[3.5.1 系统界面设计 31](#_Toc398038718)

[3.5.2系统各模块实现代码 34](#_Toc398038719)

[3.5.3 数据库实现 44](#_Toc398038720)

[第四章 面向IPv6无线传感网的信息收集平台运行与测试 45](#_Toc398038721)

[4.1平台的搭建 45](#_Toc398038722)

[4.2 节点监测客户端运行与测试 46](#_Toc398038723)

[第五章 总结 48](#_Toc398038724)

第一章 面向IPv6无线传感网的信息收集平台概述

1.1面向IPv6无线传感网的信息收集平台需求性分析概述

1.1.1 功能需求

系统的开发主要实现以下几个目标：(1)界面设计友好(2)数据存储安全(3)信息分类清晰(4)查询速度快(5)具有易操作性和可维护性。面向IPv6无线传感网的信息收集软硬件集成平台具有账户系统、节点信息访问、节点信息存储、节点历史信息管理、节点历史信息查询、系统帮助等功能。用户可以通过本系统浏览节点上的温度、湿度信息，并且可以获取通过连接在节点上的摄像头获取图像信息，实时拍摄所监测位置的图像。

在Contiki系统的基础上，基于uIPv6START KIT无线网络开发套件，通过加入和数据库的交互，把历史数据存入数据库中，从而在本系统中不仅可以访问传感器节点上的实时数据，还能查看历史数据，以便于分析问题。

1.1.2 界面需求

软件界面是人与计算机之间的媒介。用户通过软件界面来与计算机进行信息交换。软件界面的质量，直接关系到应用系统的性能能否充分发挥，能否使用户准确、高效、轻松、愉快地工作，所以软件的友好性、易用性对于软件系统至关重要。目前国内软件开发者在设计过程中很注重软件的开发技术及具有的业务功能，而忽略了用户对软件界面的需求，影响软件的易用性、友好性；对界面设计的研究也集中在界面设计技术、设计手段上面。软件开发人员在设计时以经验为参考依据，缺乏对实际用户需求的了解。而软件的友好性、易用性同用户特征紧密相联，同样的软件界面，不同用户可能有绝然相反的评价。因此分析用户特征、了解用户需求和操作习惯，是开发软件界面的必有步骤，必须引起足够重视。

在一个大型系统中，需求分析人员面对的用户只能是众多单一的用户个体，他们的需求千奇百怪。只有明确了用户角色，需求分析人员才能在纷乱复杂而又不甚明了的用户要求中理出脉络，依据用户角色不同的优先级别，平衡众多用户需求中的矛盾。考虑使用信息收集平台的用户一般为生疏用户。对于软件使用和功能并没有太多的了解，所以在进行界面设计时间要充分的考虑到这些，把软件尽可能的做的操作方便、简单明了。

1.2 面向IPv6无线传感网的信息收集平台的实现概述

面向IPv6无线传感网的信息收集系统使用的硬件平台是美信凌科公司的IPv6智能网关(MXG300)、MX231CC节点、USB无线网卡(STICK)和JTAG下载器。系统的硬件平台配置和硬件平台如图1-8所示。软件平台是WinAVR和AVR studio（如图1-9），用于向节点和USB网卡中下载程序。软件开发工具选择的是Microsoft Visual Studio 2005，数据库的开发使用的是mysql数据库开发工具，如图1-10。

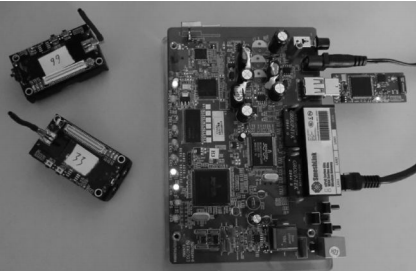
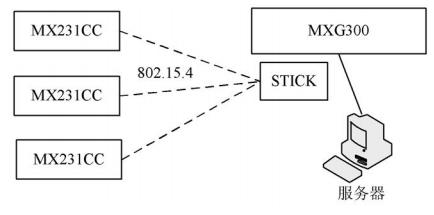


图1-8系统的硬件平台及连接示意图

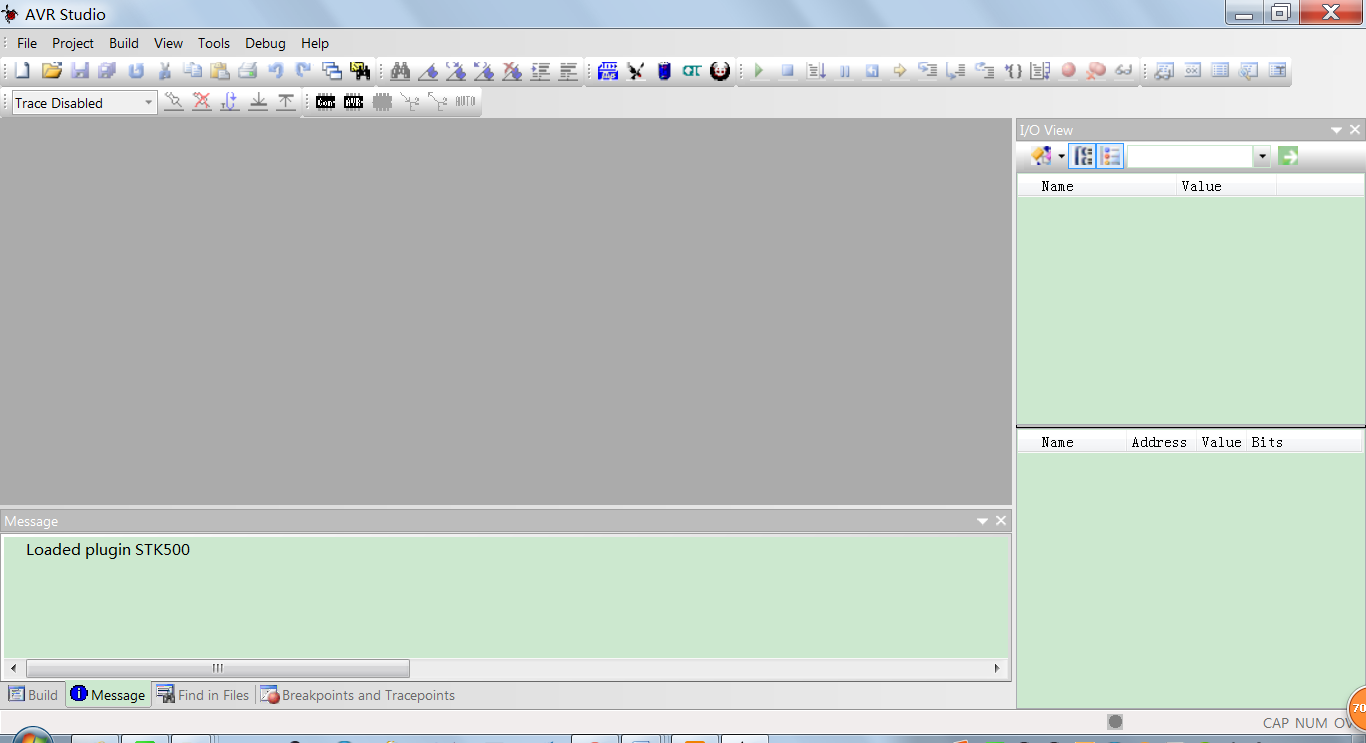


图1-9 AVR studio

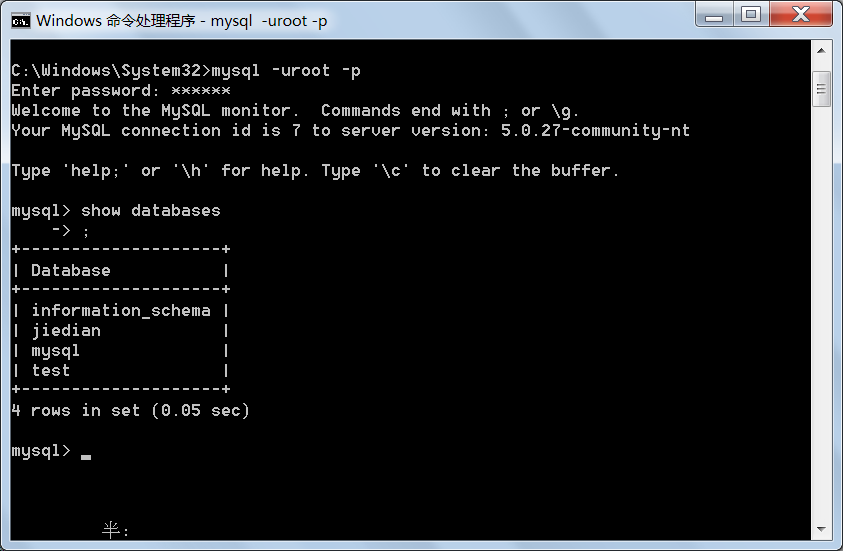
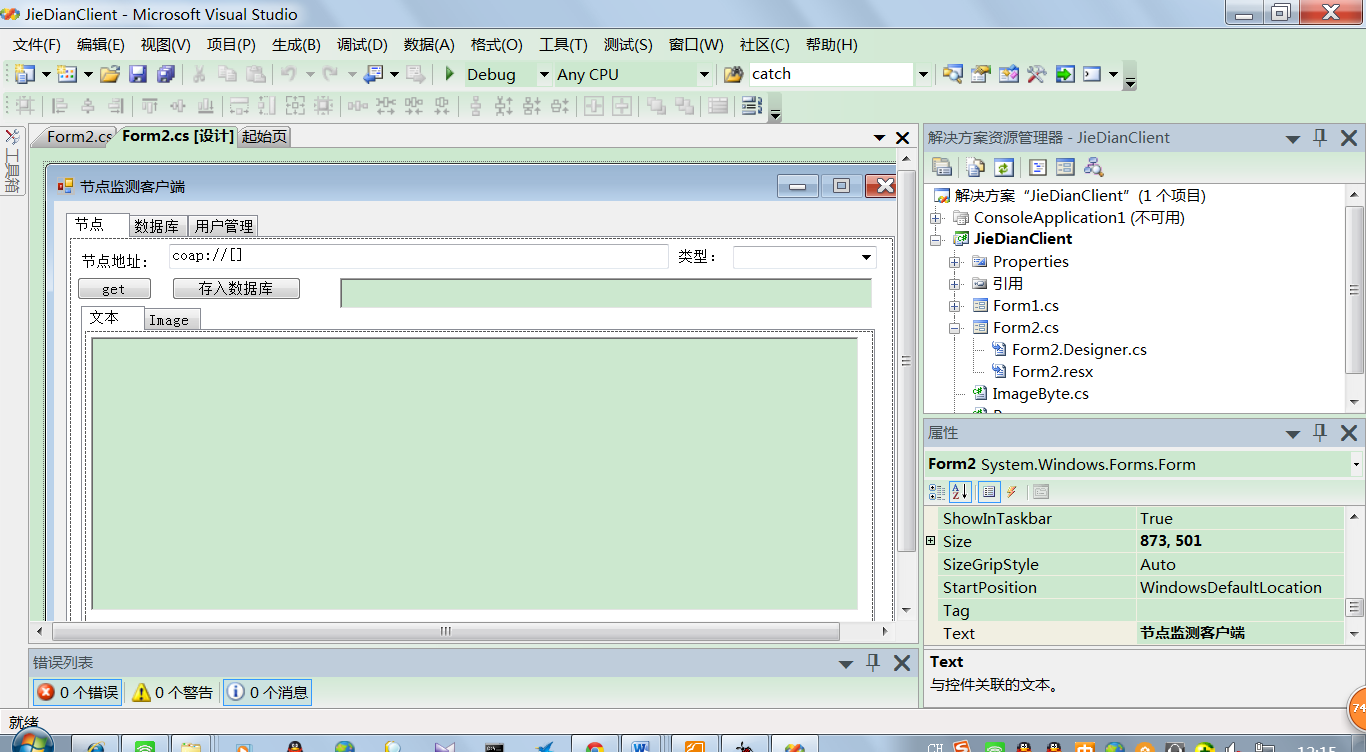


图1-10 Microsoft Visual Studio 2005及mysql

面向IPv6无线传感网的信息收集系统开发工具选择的是Microsoft Visual Studio 2005和mysql数据库开发工具。在系统构建的时候，使用了三层架构，分别是用户界面层、业务实体层、数据层。用户界面层，是系统的UI部分，负责使用者与整个系统的交互，为用户提供友好的操作界面和帮助信息，因此包含了许多控件和相关UI组件。业务实体层是应用的本体，它将业务处理逻辑编入程序代码中，接受用户请求。从数据层获取数据、通过应用逻辑进行处理并将结果交给用户界面层显示。数据访问层负责管理和操作数据库，它必须能够快速的执行大量的数据检索和更新操作。 三层架构设计如下图1-11所示。

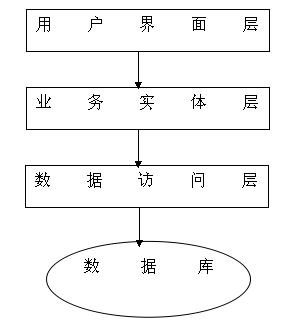


图1-11三层架构设计图

第二章 面向IPv6无线传感网的信息收集系统硬件环境

2.1无线传感器网络节点

传感器节点通常是一个微型的嵌入式系统，它的处理能力、存储能量和通信能量相对较弱，通过携带能量有限的电池供电。从网络功能上看，每个传感器节点兼顾传统网络节点的终端和路由器双重功能，除了进行本地信息收集和数据处理外，还要对其他节点转发来的数据进行存储、管理和融合等处理，同时与其他节点协作完成一些特定任务。汇聚节点的处理能力、存储能力和通信能力相对比较强，它连接传感器网络与Internet等外部网络，实现两种协议栈之间的通信协议转换，同时发布管理节点的监测任务，并把收集的数据转发到外部网络上。汇聚节点既可以是一个具有增强功能的传感器节点，有足够的能量供给和更多的内存与计算资源，也可以没有监测功能仅带有无线通信接口的特殊网关设备。

无线传感器网络节点根据其具体应用场合的不同，设计也各不相同，但其基本结构均由传感器模块、处理器模块、无线通信模块和能量供应模块四部分组成，如图2-1所示。传感器模块负责检测区域内信息的采集和数据转换；处理器模块负责控制整个传感器节点的操作、存储和处理本身采集的数据以及其他节点发来的数据；无线通信模块负责与其他传感器节点进行无线通信，交换控制信息和收发采集数据；能量供应模块为传感器节点提供运行所需的能量，通常采用微型电池。

由于传感器节点通常需要在被监控区域放置半年甚至更长时间不能更换电池，因此节能是传感器节点的最主要的要求，上层协议及应用软件的设计均应将能耗作为重要考虑因素。

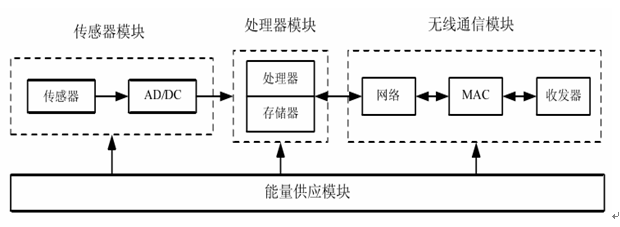


图2- 1无线传感器网络节点组成结构

面向IPv6无线传感网的信息收集平台收集信息时使用的节点是MX231CC节点，节点上的主要芯片是ATmega1284P，它具有128 KB的可编程闪存，4 KB的E2PROM，16 KB的片内SRAM，JTAG接口，优化的功耗和处理速度。节点上运行Contiki系统。节点上还有51Pin无线传感器网络经典接插件、光敏传感器、室内温度传感器、三色LED指示灯等。如图2-2

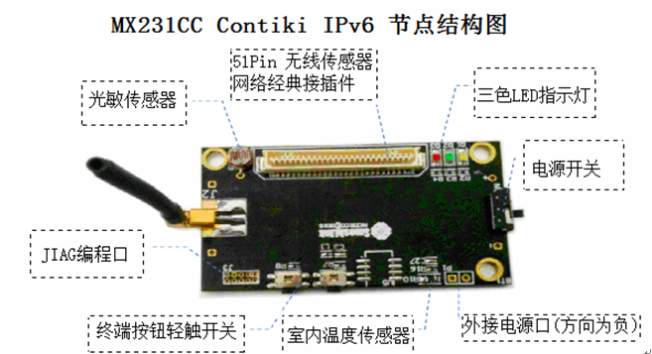


图2- 2 MX231CC Contiki IPv6 节点结构图

MX231CC节点上的主要芯片是ATmega1284P其电路图如图2-3所示

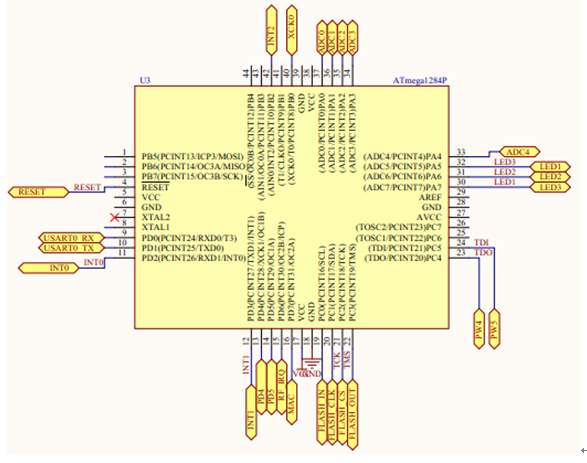


图2- 3 ATmega1284P芯片引脚图

Atmega1284芯片主要特点：

该ATmega1284P提供了以下特点：128K字节的系统内可编程Flash(具有在写的过程中还可以读的能力) 4K字节的EEPROM ，16K字节SRAM ，32个通用构成的I / O口线， 32个通用工作寄存器，实时计数器（ RTC ），三个灵活的具有比较模式和PWM的定时器/计数器，2个USART ，面向字节的两线串行接口面，一个8通道10位ADC，具有可选的可编程增益，具有片内内部振荡器可编程看门狗定时器，一个SPI串行端口， IEEE标准。 1149.1兼容的JTAG测试接口，也可用于访问片上调试系统和编程和六个软件选择的省电模式。在空闲模式时CPU停止工作，而允许SRAM，定时器/计数器，SPI端口以及中断系统继续工作。

MX231CC节点上的51Pin无线传感器网络接插件其电路图如图2-4所示

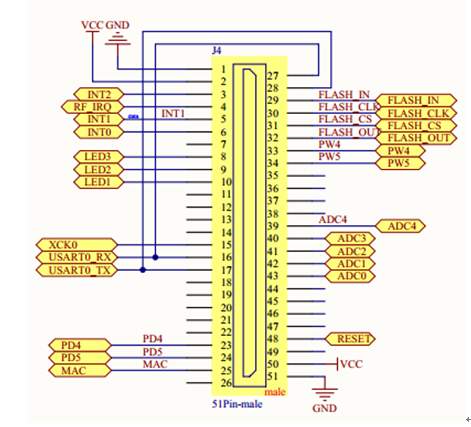


图2- 4 51Pin无线传感器网络接插件引脚图

2.2 Smeshlink IPv6 MXG300智能网关

IPv6智能网关上的主要芯片有：BCM 6358UKFBG支持多用户以太网功能，具有高度优化的32 MIPS CPU和标准的EJTAG调试器；BCM53 25EKQMG集成了5个收发器，具有128 KB的数据包缓冲区，最多可以支持2K的MAC地址，支持地址自动学习，提供真正的即插即用连接，而且是低功耗的；SIGe2521A60提供2．4～2．5 GHz的无线工作频段范围，应用于ISM 2．4．GHz的无线解决方案

图2-5是IPv6智能无线网关的接口布局，它是基于OPENWRT系统定制完成的。具备3个局域网口，1个广域网口，1个802．11a／b／g WiFi无线网络接口，1个标准USB口和1个可选的串口调试口。该智能无线网关除具备通用无线路由器的功能以外，还可以实现基于Contiki操作系统的USB UIP网络和普通IP网络之间的IPv6互连，同时还支持有能力的系统在OPENWRT的基础上开发自己的应用软件包，实现更复杂的应用。

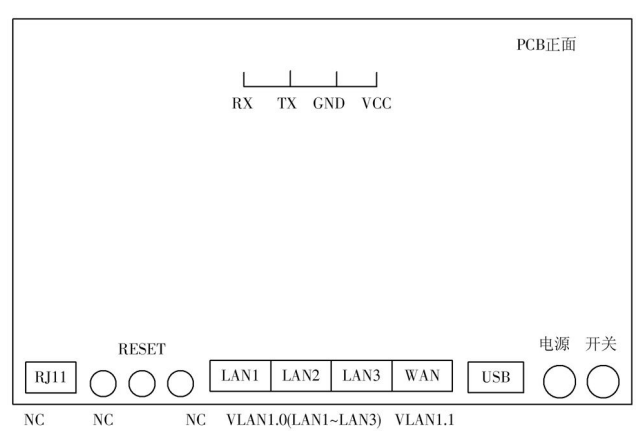


图2-5 网关接口布局

OPENWRT是一个开源的Linux版本。主要应用于嵌入式系统。网关和节点上同时装有Contiki系统，它提供宏定义和RESTful网络服务实例。

Smeshlink IPv6 MXG300智能网关网络地址的配置：

网关在出厂之前已经完成了所有地址的配置，默认地址如下，注意，子网掩码区能省略：

LAN：192.168.1.1 2001:1::1/64

USB0:2001:2::1/64

如果想更改地址，需要更改两个文件，一个是/etc/config/networks，用来更改 IPV6 地址，另外 1 个是/etc/config/radvd，用来路由信息的宣告。为了改变地址信息，用户可以直接通过访问 http://192.168.1.1，用 root admin 登陆后进行更改，也可以通过 ssh 客户端登陆以后进行更改，两者的效果是一样的，但 radvd 文件的修改只能通过 putty 这样的客户端完成。网页修改比较简单，请用户自行完成即可。如果我们要把 usb0 的地址改成 aaaa::1/64，那么通过 putty 登陆以后，执行vi /etc/config/networks

文件修改后内容如下：

config 'switch' 'eth1'

option 'reset' '1'

option 'enable\_vlan' '1'

config 'switch\_vlan'

option 'device' 'eth1'

option 'vlan' '0'

option 'ports' '0 1 2 5\*'

config 'switch\_vlan'

option 'device' 'eth1'

option 'vlan' '1'

option 'ports' '3 5\*'

config 'interface' 'loopback'

option 'ifname' 'lo'

option 'proto' 'static'

option 'ipaddr' '127.0.0.1'

option 'netmask' '255.0.0.0'

config 'interface' 'lan'

option 'type' 'bridge'

option 'ifname' 'eth1.0'

option 'proto' 'static'

option 'ipaddr' '192.168.1.1'

option 'netmask' '255.255.255.0'

option 'nat' '1'

option 'defaultroute' '0'

option 'peerdns' '0'

option 'ip6addr' '2001:1::1/64'

config 'interface' 'wan'

option 'ifname' 'eth1.1'

option 'proto' 'dhcp'

config 'interface' 'usb0'

option 'proto' 'static'

option 'ifname' 'usb0'

option 'defaultroute' '0'

option 'peerdns' '0'

#option 'ip6addr' '2001:2::1/64'

option 'ip6addr' 'aaaa::1/64'

改动完成后需要重新启动网络。/etc/init.d/network restart

然后执行vi /etc/config/radvd

文件修改后内容如下：

config interface

option interface 'usb0'

option ignore 0

option AdvSendAdvert 1

option AdvLinkMTU 1280

option AdvCurHopLimit 128

option AdvReachableTime 360000

option MinRtrAdvInterval 100

option MaxRtrAdvInterval 150

option AdvDefaultLifetime 200

config prefix

option interface 'usb0'

#option prefix '2001:2::/64'

option prefix 'aaaa::/64'

option AdvOnLink 1

option AdvAutonomous 1

option AdvPreferredLifetime 4294967295

option AdvValidLifetime 4294967295

option ignore 0

config rdnss

option interface 'usb0'

option addr ''

option ignore 1

config interface

option interface 'lan'

option AdvSendAdvert 1

option AdvManagedFlag 0

option AdvOtherConfigFlag 0

option ignore 0

option MinRtrAdvInterval 5

option MaxRtrAdvInterval 10

option AdvDefaultPreference low

config prefix

option interface 'lan'

option prefix '2001:1::/64'

option AdvOnLink 1

option AdvAutonomous 1

option AdvRouterAddr 1

option ignore 0

config rdnss

option interface 'lan'

option addr ''

option ignore 1

即和前面的地址网络号要保持一致，其它都不用变，然后执行以下命令，重起 radvd 即可：/etc/inid.d/radvd restart

这样就完成了智能无线网关的配置工作。网关即可正常使用。

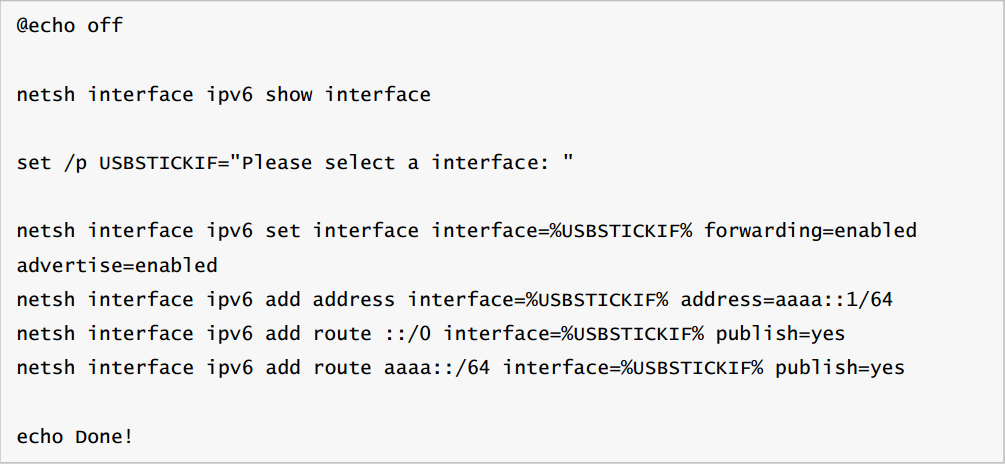
2.3 USB无线网卡(STICK)

用户使用面向IPv6无线传感网的信息收集系统时既可以通过连接IPv6智能网关访问节点资源也可以可以通过在计算机上插USB无线网卡(STICK)访问节点资源。通过在节点上烧写不同的程序用户可以通过HTTP协议、COAP协议访问MX231CC节点。这里的USBStick 是用作无线传感网络与其他网络连接的网关，用于连接节点与传感网络外部设备。

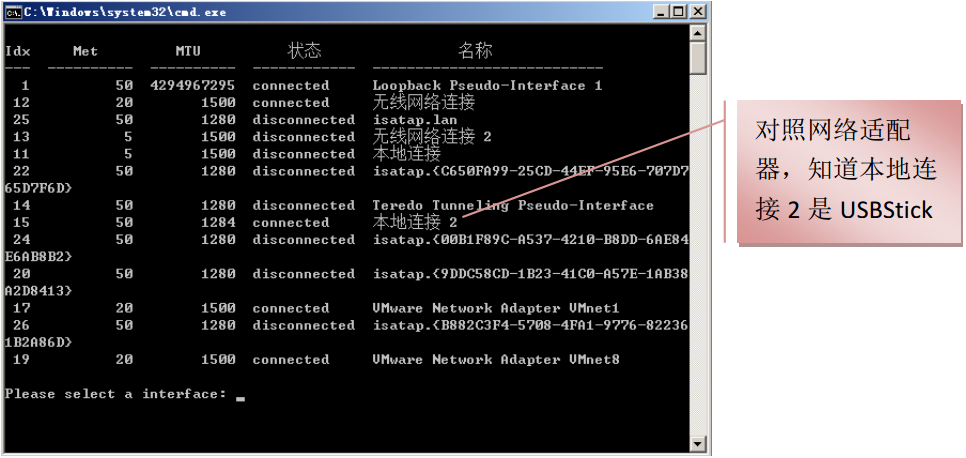
若想USBstick正常工作还要对其进行配置。将 USBStick 连接至 PC（pc系统为win7系统），在网络连接（网络->属性->更改适配器设置）中可以看到 USBStick 代表的网络：



在win7系统上运行以下.bat 批处理文件：



以管理员身份运行该批处理文件，运行结果如下图所示：



本地连接 2 的 interface（第一列代表的数字）是 15，所以输入 15，按下回车完成配置。

2.4 JTAG介绍与使用

JTAG接口定义：

JTAG(Joint Test Action Group ，联合测试行动小组 ) 是一种国际标准测试协议，主要用于芯片内部测试及对系统进行仿真、调试， JTAG 技术是一种嵌入式调试技术，它在芯片内部封装了专门的测试电路 TAP （ Test Access Port ，测试访问口），通过专用的 JTAG 测试工具对内部节点进行测试。目前大多数比较复杂的器件都支持JTAG协议。标准的JTAG接口为4线制：TMS、TCK、TDI和TDO，分别为测试模式选择、测试时钟、测试数据输入和测试数据输出。

JTAG接口提供了在线程序调试功能，如图2-6所示为JATG调试节点及接口图。

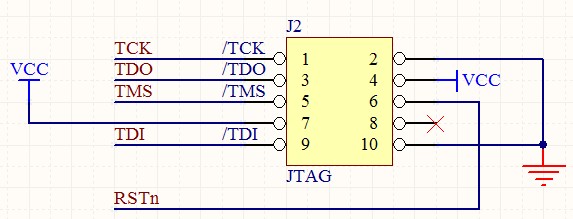


图2- 6 JATG调试节点及接口图

**JTAG引脚定义：**

JTAG接口主要包括四个引脚下：TMS TCK TDI 和TCO及一个可选配的引脚TRST，用于驱动电路模块和控制执行规定的操作。各引脚的功能如下：

TCK：JTAG测试时钟，为TAP控制器和寄存器提供测试参考。在TCK的同步作用下通过TDI和TDO引脚下串行移入或移出数据及指令。同时，TCK为TAP控制器状态机提供时钟。

TMS：TAP控制器的三项式输入信号。TCK的上升沿时刻TMS的状态确定TAP控制器即将进入的工作状态。通常TMS引脚具有内部上拉[电阻](http://www.dzsc.com/product/searchfile/294.html) ，以保证该引脚在没有驱动时处于逻辑1状态。

TDI：JTAG指令和数据寄存器的串行数据输入端。TAP控制器的当前状态以及保持在指令寄存器中的具体指令决定对于一个特定的操作由TDI装入哪个寄存器。在TCK的上升沿时刻，TDI引脚状态被除数采样，结果送到JTAG寄存器组。

TRST：测试复位输入信号，低电平有效，为TAP控制器提供异步初始化信号。



图2-7 编程接口和无线节点连接示意图



图2-8 编程接口和USBstick连接示意图

2.5 MXP540 USB串行功能扩展板介绍与使用

节点采集图像信息时需要用到MXP540 USB串行功能扩展板。MXP540既可以做串口打印输出的调试板（使用其中的USB转串口功能），也可以做摄像头的转接板（使用其中的5V输出和UART转接功能），其具体使用方法如下：

摄像头转接板

首先需要把跳线拿开，或者让RX1和RX短接，TX1和TX短接，其目的是让单片机的UART0不再和USB转串口芯片连通，然后按照下图连接摄像头和转接板，同时还需要5V外接电源才能工作:



具体就是黄线/绿线和RX0相连，绿线/黄线和TX0相连，红线和5V相连，黑线和GND相连。（黄线RX0，绿线TX0），连接完成后通过节点访问摄像头，采集图像信息。

串口打印调试板

只需通过跳线让RX0和RX短接，TX0和TX短接即可，这样用户就可以使用电脑查看MX231CC的串口输出了。

第三章 面向IPv6无线传感网的信息收集系统软件环境

3.1 Contiki Studio集成环境开发平台

继美信凌科（Smeshlink）和法国电信中国研发中心（French Telecom Beijing FT-Group）合作研发基于 IEEE802.15.4 物理层的IPv6实验系统后，美信凌科面向研发用户提供完整的Contiki IPv6开发系统。

开发系统包含最新的MX231CC无线节点、最新IPv6 WiFi智能网关以及 Contiki Studio集成开发环境。套件系统支持Windows平台开发编程。该开发套件是目前国内唯一完整支持Contiki系统的开发套件。

Contiki Studio集成开发环境是美信凌科（SmeshLink）推出的基于Window操作系统的Contiki集成开发环境。Contiki Studio采用一站式安装服务，将Contiki 系统的编译、辅助工具和开发环境集成在一起，为高效开发Contiki应用程序创造了有利条件。

Contiki Studio集成开发环境支持：

1) 一站式Windows标准安装

2) 系统资源消耗少，界面统一

3) 关键字、变量和函数高亮显示

4) 变量名、函数名编辑过程智能显示

5) 括号及分号规则检测

6) 支持Contiki应用程序工程化管理

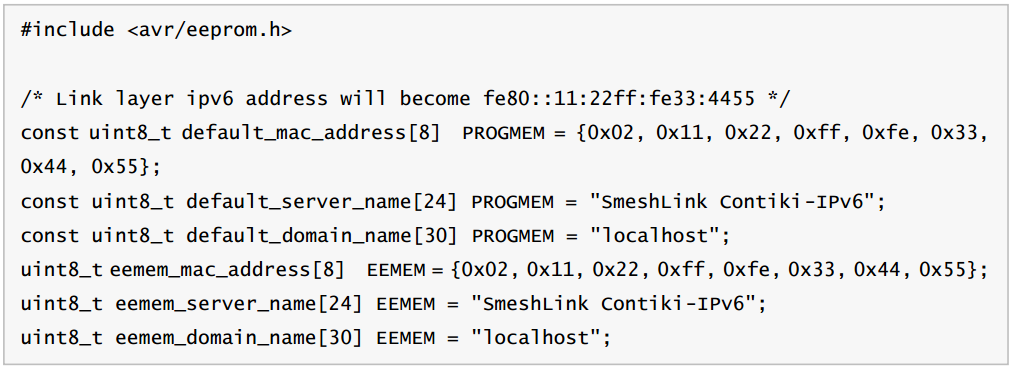
7) 支持工程编译和分平台编译

8) 支持用JTAG ICE MKII编程器直接上载程序

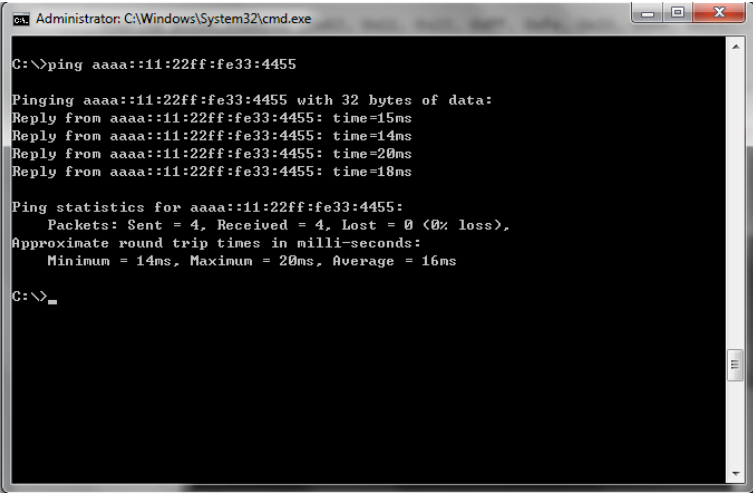
9) 提供了多个由浅入深的示例程序，快速实现6LowPAN应用网络

Contiki Studio集成开发环境示例程序介绍：

Contiki Studio集成开发环境提供了一些示例程序，通过在节点上烧写不同的程序，可以实现用户的不同的功能。比如Contiki Studio集成开发环境中的mx-web示例程序是在 mx2xxcc 节点上的一个小型 Web 服务器，终端可以访问节点上的 Web 资源。将示例程序 examples/02-mx-web 编译并烧写到无线节点上。示例程序 mx-web中，节点地址定义位于 examples/02-mx-web/ipv6\_smeshlink/makefsdata.h 中，内容如下：



其中 eemem\_mac\_address 与 default\_mac\_address 均定义的是节点地址。如果无线节点的 IPv6 地址就是 aaaa::11:22ff:fe33:4455。



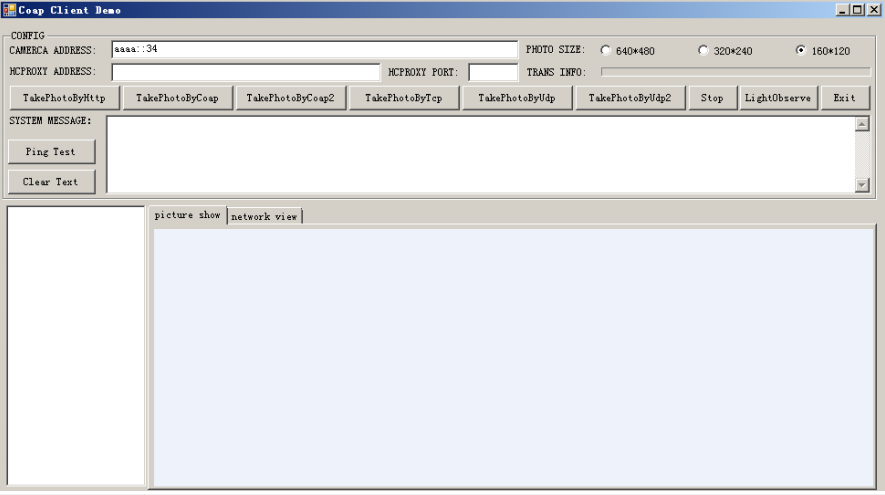
打开浏览器，在地址栏输入 http://[aaaa::11:22ff:fe33:4455]/，可以浏览运行在节点上的 Web 网站。



将示例程序 examples/08-camera/camera-rest编译并烧写到节点上可以实现通过节点访问摄像头的功能。将示例程序 examples/08-camera/camera-rest 编译并烧写到无线节点上，配置好 USBStick 网关，并连接好摄像头、扩展板和无线节点：



打开节点开关，启动 Wireless Video Monito：



在 CAMERCA ADDRESS 中输入摄像头节点的 IP 地址。点击 Ping Test，如果在右侧显示 Node IP Works （如上图所示） ，则节点开始工作。点击 TakePhotoByCoap即可获得图像。

3.2 WinAVR和AVR studio软件开发平台

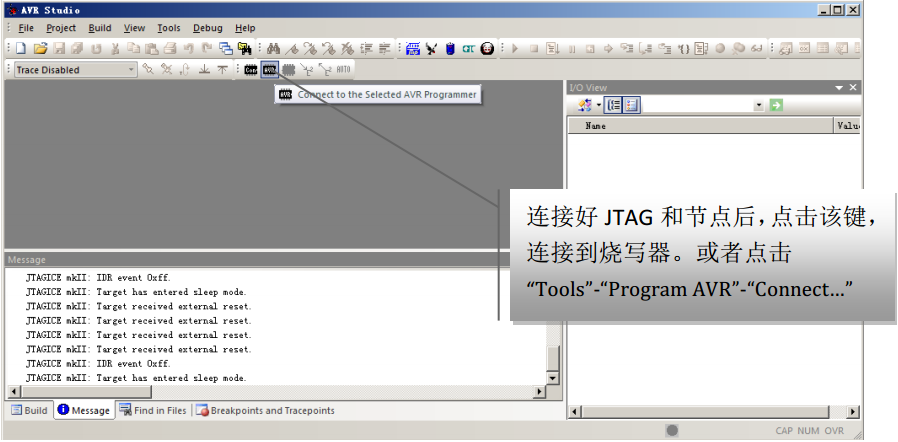
AVR Studio是ATMEL官方针对AVR系列单片机推出的集成开发环境，它集开发调试于一体，有很好的用户界面，很好的稳定性。AVRSdudio是免费的。AVRStudio本身可以开发汇编程序，如果希望使用c语言开发，则需要安装c编译器（WinAVR等）。

WinAVR是GNU组织推出的AVR单片机的gcc编译器，该编译器的编译效率极高。gcc编译器是开源的，是免费的。使用gcc编译器，可以掌握标准c的语法等。

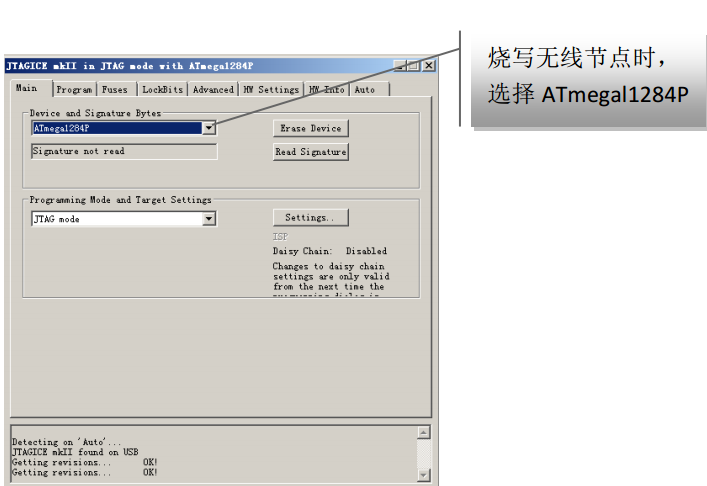
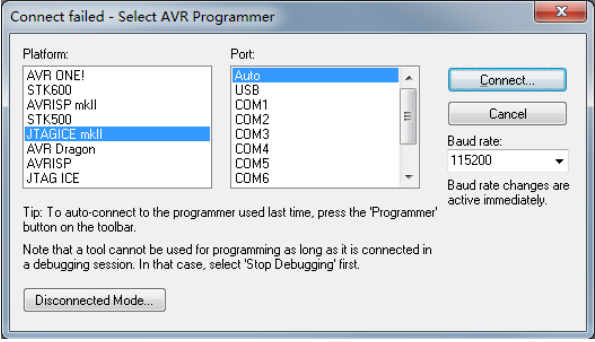
WinAVR和AVR studio软件开发平台的使用：

下载安装 WinAVR 和VR studio软件完成后，即可通过WinAVR和AVR studio软件开发平台将编译好的程序烧写进节点中，具体做法为将 JTAG 通过串口线连接到 PC 的 USB 接口上，烧写过程中节点与 USBStick 和 JTAG 编程接口的连接如图 2-7、图 2-8 所示。设备连接后，操作如下：

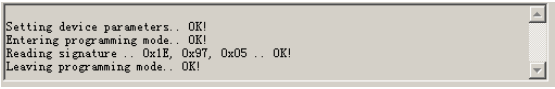
1）打开AVR studio 4



2）选择 Platform: JTAGICE mkII，Port: Auto，Baud rate: 115200，点击“Connect…”。



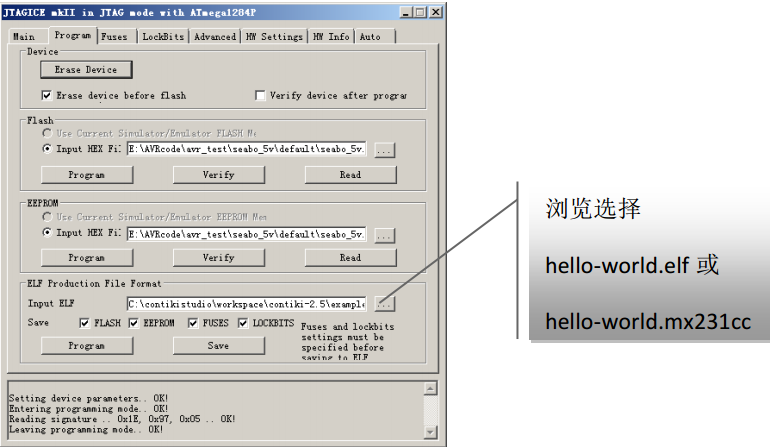
点击 Read Signature，在下方出现：



说明节点连接成功。注意，烧写前，将节点装好电池，打开开关，才能顺利烧写。如果 JTAG

经过改装，可以直接供电，则节点不用电池供电也可烧写。

3) 点击“Program”选项卡，在“ELF Production File Format”部分，“Input ELF”选择examples/01-hello-world/hello-world.mx2xxcc 文件，确认设备连接无误后，点击“Program”按钮。



4) 上载成功后断开 JTAG 连接，将看到节点启动后，程序运行，红、绿、黄三个 LED 灯会依次闪烁。如果连接了开发调试附件，可以看到串口不断有数据输出。

3.3面向IPv6无线传感网的信息收集系统开发工具

面向IPv6无线传感网的信息收集系统开发工具选择的是Microsoft Visual Studio 2005和mysql数据库开发工具。

3.3.1 Microsoft Visual Studio 2005介绍

Visual Studio .NET 是一套完整的开发工具，用于生成 ASP Web应用程序、XML Web services、桌面应用程序和移动应用程序。Visual Basic .NET、Visual C++ .NET、Visual C# .NET 和 Visual J# .NET 全都使用相同的集成开发环境 (IDE)，该环境允许它们共享工具并有助于创建混合语言解决方案。另外，这些语言利用了.NET Framework的功能，此框架提供对简化ASP Web应用程序和XML Web services开发的关键技术的访问。

Visual Studio .NET是用于快速生成企业级ASP.NET Web应用程序和高性能桌面应用程序的工具。Visual Studio包含基于组件的开发工具（如 Visual C#、Visual J#、Visual Basic和 Visual C++），以及许多用于简化基于小组的解决方案的设计、开发和部署的其他技术。

Visual Studio支持Microsoft .NET Framework，该框架提供公共语言运行库和统一编程类；ASP.NET使用这些组件来创建ASP.NET Web应用程序和XML Web services。Visual Studio是微软公司推出的开发环境。是目前最流行的Windows平台应用程序开发环境。Microsoft Visual Studio使开发人员能够快速创建高质量、用户体验丰富而又紧密联系的应用程序，充分展示了 Microsoft 开发智能客户端应用程序的构想。借助 Visual Studio 2005，采集和分析信息将变得更为简单便捷，业务决策也会因此变得更为有效。任何规模的组织都可以使用Visual Studio 2005快速创建能够利用 Windows Vista™ 和 2007 Office system 的更安全、更易于管理并且更可靠的应用程序。Visual Studio 2005可以高效开发 Web应用，集成了ASP.NET AJAX 1.0，包含ASP.NET AJAX项目模板，它还可以高效开发Office 应用和Mobile应用。Visual Studio 2010同时带来了NET Framework 4.0，并且支持开发面向Windows 7的应用程序。除了Microsoft SQL Server，它还支持 IBM DB2和Oracle数据库。

信息收集系统使用的是Visual Studio .NET。Microsoft .NET Framework，该框架提供公共语言运行库和统一编程类；ASP.NET使用这些组件来创建ASP.NET Web应用程序和XML Web services。VS2005结合了ASP.NET语言完成了信息收集平台，包括对节点资源的访问（包括光照、温度、电压）、存储、查询、删除管理；该开发工具还有优秀实用的Chart控件，能够实现对节点历史曲线视图的查看；能够支持用户忘记密码发送邮件、导出节点信息的历史数据到Excel文件，以便查看和分析等功能。该开发工具能够满足平台开发的全部需求，极为方便。

3.3.2 C#编程语言介绍

C#是微软公司发布的一种面向对象的、运行于.NET Framework之上的高级程序设计语言。并定于在微软职业开发者论坛(PDC)上登台亮相。C#是微软公司研究员Anders Hejlsberg的最新成果。C#看起来与Java有着惊人的相似；它包括了诸如单一继承、接口、与Java几乎同样的语法和编译成中间代码再运行的过程。但是C#与Java有着明显的不同，它借鉴了Delphi的一个特点，与COM（组件对象模型）是直接集成的，而且它是微软公司 .NET windows网络框架的主角。

C#是一种安全的、稳定的、简单的、优雅的，由C和C++衍生出来的面向对象的编程语言。它在继承C和C++强大功能的同时去掉了一些它们的复杂特性（例如没有宏以及不允许多重继承）。C#综合了VB简单的可视化操作和C++的高运行效率，以其强大的操作能力、优雅的语法风格、创新的语言特性和便捷的面向组件编程的支持成为.NET开发的首选语言。

C#是面向对象的编程语言。它使得程序员可以快速地编写各种基于MICROSOFT .NET平台的应用程序，MICROSOFT .NET提供了一系列的工具和服务来最大程度地开发利用计算与通讯领域。

C#使得C++程序员可以高效的开发程序，且因可调用由 C/C++ 编写的本机原生函数，因此绝不损失C/C++原有的强大的功能。因为这种继承关系，C#与C/C++具有极大的相似性，熟悉类似语言的开发者可以很快的转向C#。

3.3.3 mysql数据库介绍

MySQL是一个开放源码的小型关联式数据库管理系统，开发者为瑞典MySQL AB公司。MySQL被广泛地应用在Internet上的中小型网站中。由于其体积小、速度快、总体拥有成本低，尤其是开放源码这一特点，许多中小型网站为了降低网站总体拥有成本而选择了MySQL作为网站数据库。

与其他的大型数据库例如Oracle、DB2、SQL Server等相比，MySQL自有它的不足之处，但是这丝毫也没有减少它受欢迎的程度。对于一般的个人使用者和中小型企业来说，MySQL提供的功能已经绰绰有余，而且由于MySQL是开放源码软件，因此可以大大降低总体拥有成本。Linux作为操作系统，Apache和Nginx作为Web服务器，MySQL作为数据库，PHP/Perl/Python作为服务器端脚本解释器。由于这四个软件都是免费或开放源码软件（FLOSS)，因此使用这种方式不用花一分钱（除开人工成本）就可以建立起一个稳定、免费的网站系统，被业界称为“LAMP“组合。

MySQL由MySQL AB开发、发布和支持。MySQL AB是一家基于MySQL开发人员的商业公司，它是一家使用了一种成功的商业模式来结合开源价值和方法论的第二代开源公司。MySQL是MySQL AB的注册商标。MySQL是一个快速的、多线程、多用户和健壮的SQL数据库服务器。MySQL服务器支持关键任务、重负载生产系统的使用，也可以将它嵌入到一个大配置(mass-deployed)的软件中去。MySQL网站(http://www.mysql.com)提供了关于MySQL和MySQL AB的最新的消息。

　　MySQL是一个数据库管理系统，一个数据库是一个结构化的数据集合。它可以是从一个简单的销售表到一个美术馆、或者一个社团网络的庞大的信息集合。如果要添加、访问和处理存储在一个计算机数据库中的数据，你就需要一个像MySQL这样的数据库管理系统。从计算机可以很好的处理大量的数据以来，数据库管理系统就在计算机处理中和独立应用程序或其他部分应用程序一样扮演着一个重要的角色。

　　MySQL是一个关系数据库管理系统，关系数据库把数据存放在分立的表格中，这比把所有数据存放在一个大仓库中要好得多，这样做将增加你的速度和灵活性。“MySQL”中的SQL代表“Structured Query Language”（结构化查询语言）。SQL是用于访问数据库的最通用的标准语言，它是由ANSI/ISO定义的SQL标准。SQL标准发展自1986年以来，已经存在多个版本：SQL-86，SQL-92，SQL:1999，SQL:2003，其中SQL:2003是该标准的当前版本。

　　MySQL是开源的，开源意味着任何人都可以使用和修改该软件，任何人都可以从Internet上下载和使用MySQL而不需要支付任何费用。如果你愿意，你可以研究其源代码，并根据你的需要修改它。MySQL使用GPL(GNU General Public License，通用公共许可)，在hpt://www.fsf.org/licenses中定义了你在不同的场合对软件可以或不可以做什么。如果你觉得GPL不爽或者想把MySQL的源代码集成到一个商业应用中去，你可以向MySQL AB购买一个商业许可版本。

　　MySQL服务器是一个快的、可靠的和易于使用的数据库服务器

　　如果这是你正在寻找的，你可以试一试。MySQL服务器还包含一个由用户紧密合作开发的实用特性集。你可以在MySQL AB的http://www.mysql.com/it-resources/benchmarks/上找到MySQL服务器和其他数据库管理系统的性能比较。

　　MySQL服务器原本就是开发比已存在的数据库更快的用于处理大的数据库的解决方案，并且已经成功用于高苛刻生产环境多年。尽管MySQL仍在开发中，但它已经提供一个丰富和极其有用的功能集。它的连接性、速度和安全性使MySQL非常适合访问在Internet上的数据库。

3.4面向IPv6无线传感网的信息收集系统设计

3.4.1系统分层架构设计

系统采用三层结构设计，三层结构设计是在逻辑上的划分方式，以往的设计是把页面表现的内容和业务逻辑内容放在一起，错综复杂，没有清晰的逻辑结构，很可能只有开发者本人了解自己开发的东西，别人需要使用的时候，费时费力，难以辨认，三层结构就是要求把以往的具体业务逻辑从页面展示的内容中提取出来，单独作为一个层次，分为了表示层(USL)、业务逻辑层(BLL)、数据库持久层(DAL)，表示层只负责页面的渲染和数据内容的展示，具体怎么实现的展示结果就交给了业务逻辑层了，业务逻辑层负责业务需求的处理，从数据库中读取数据，按照要求把结果集返回给表示层页面，数据访问层也是介于持久层与业务层之间的一个层次，用于连接数据库和业务层的数据访问问题。三层结构的软件，如果设计的合理，每一层都是单独的存在，对彼此的依赖度也较小，如果需要升级，更换某个层次的内容也较为简单的实现，不再像过去那样一发而动全身。

微软的三层架构示意图，如图所示：

表示层

业务层

数据层

图3- 1 三层架构示意图

本平台之所以使用三层架构，是因为三层架构清晰的逻辑结构，表示层、业务逻辑层、数据库层每一层都是单独的存在，对彼此的依赖度也较小，如果需改动或升级，更换某个层次的内容也较为简单的实现，不再像过去那样一发而动全身，容易维护。

3.4.2系统功能

面向IPv6无线传感网的信息收集软硬件集成平台具有账户系统、节点信息访问、节点信息存储、节点历史信息管理、节点历史信息查询、个人账户管理、系统帮助等功能。主要分为普通用户和管理员两个部分。如图3-2所示。



图3- 2面向IPv6无线传感网的信息收集软硬件集成系统结构图

3.4.3基本功能需求

用户功能需求:

(1)账户系统。用户登录，退出，找回密码（发邮件）。

(2)对节点内部资源（包括光照、温度、电压）的访问，并返回节点信息列表；提供节点的实时曲线视图

(3)对节点历史收集数据的存储、查询、管理等功能，导出数据到Excel文件

(4)个人账号管理：修改密码，修改信息

(5)系统帮助：关于

管理员功能需求

(1)账户系统。登录，退出，找回密码，修改信息

(2)用户账户管理：添加用户

(3)对节点内部资源的访问，并返回节点信息列表；提供实时曲线视图

(4)对节点历史收集数据的存储、查询、管理等功能，导出数据到Excel文件

(5)个人账号管理：修改密码，修改信息

(6)系统帮助：关于

3.4.4系统主要功能的流程图

添加新用户和登录功能流程图，如图3-3所示：

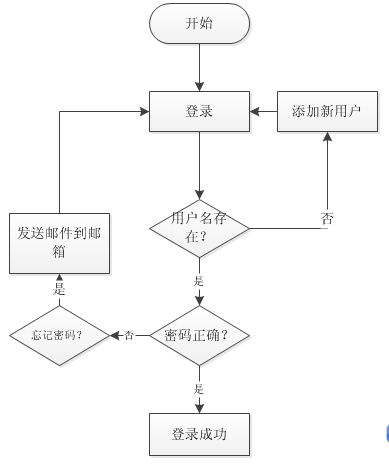


图3-3登录流程图

访问节点资源信息（光照、温度、电压）、存储功能流程图，如图3-4所示：

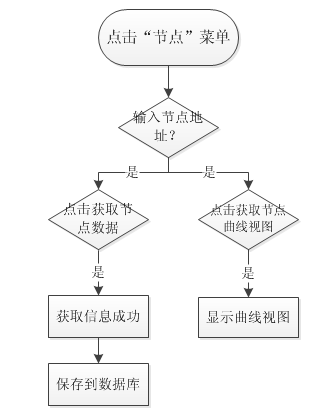


图3-4节点存储曲线视图查看

节点历史记录查询流程图，如图3-5所示：

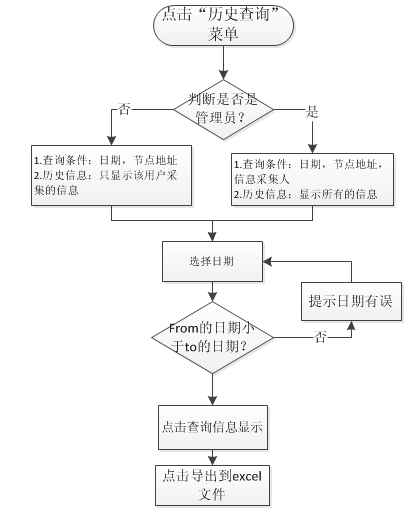


图3- 5节点历史记录查询流程图

用户账户删除流程图，如图3-6所示：

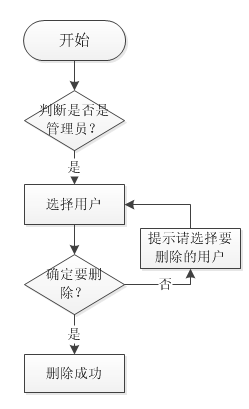


图3-6用户账户删除流程图

修改密码功能流程图，如图3-7所示：

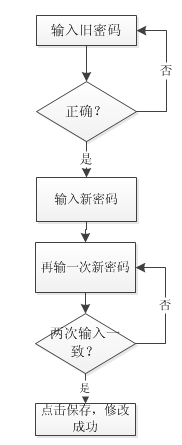


图3-7修改密码流程图

3.4.5数据库逻辑结构设计

数据库概念结构是独立于实际数据模型的信息结构，必须将其转化为逻辑结构后才能进行数据库应用的设计,也就是要将概念上的结构转化为数据库系统所支持的实际数据模型。面向IPv6无线传感网的信息收集平台中要访问节点，节点信息是系统的核心部分，因此要创建节点信息实体，用来保存节点的详细信息。节点信息实体属性图如图3-8所示。

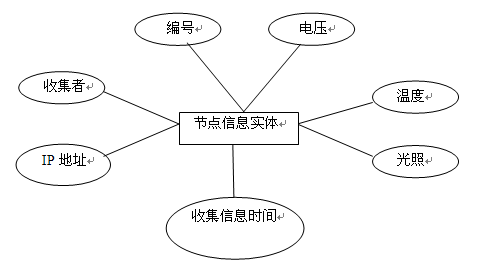


图3- 8 节点信息实体属性图

用户是面向IPv6无线传感网的信息收集软硬件集成平台的重要组成部分。这里创建了一个用户信息实体，用来存储用户的详细信息。所有用户信息实体属性图如图3-9所示。

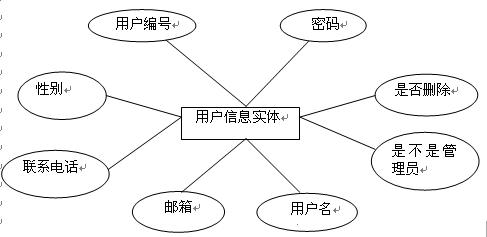


图3- 9 所有用户信息实体属性图

3.4.6 数据库结构设计表

本系统使用的数据库管理系统为mySQL。数据库命名为：jiedian，共有3张表，分别是：

（1）用户信息表：记录用户的用户名、密码、享有的权限等信息。

（2）节点信息表：记录访问的节点的信息和IP地址等信息。

（3）图像信息表：记录访问的节点上的图像信息。

表3- 1用户信息表 jd\_user

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段 | 类型 | 长度 | 允许空值 | 说明 | 描述 |
| userID | int | 11 | 否 | 主键 | 用户编号 |
| username | nvarchar | 50 | 否 | 主键 | 用户名 |
| password | nvarchar | 50 | 否 | - | 密码 |
| isdel | int | 11 | 否 | - | 是否删除  0----未删除 1------删除 |
| ismanager | nvarchar | 50 | 否 | - | 是不是管理员 |
| sex | nvarchar | 32 | 是 | - | 性别 |
| telphone | nvarchar | 50 | 是 | - | 联系电话 |
| email | nvarchar | 50 | 是 | - | 邮箱 |

表3- 2节点信息表 jd\_info

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段 | 类型 | 长度 | 允许空值 | 说明 | 描述 |
| infoID | int | 11 | 否 | 主键 | 节点编号 |
| voltage | nvarchar | 50 | 是 | - | 电压 |
| temperature | nvarchar | 50 | 是 | - | 温度 |
| light | nvarchar | 50 | 是 | - | 光照 |
| IP | nvarchar | 50 | 是 | - | 节点地址 |
| username | nvarchar | 50 | 否 | - | 收集者姓名 |
| time | datetime | 50 | 否 | - | 收集时间 |

表3-3图像信息表jd\_image

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段 | 类型 | 长度 | 允许空值 | 说明 | 描述 |
| ip | Varchar | 225 | 否 | 主键 | 节点地址 |
| time | Datetime |  | 否 | 主键 | 图像时间 |
| Image | Mediumblob |  | 是 |  | 图像 |

3.5面向IPv6无线传感网的信息收集系统具体实现

3.5.1 系统界面设计

PC端插入USBStick，打开节点，运行数据收集平台。如图3-10所示。



图3- 10 数据收集平台运行

1. 登录界面，用户名密码正确可以登录成功；如果用户名不存在，管理员需添加新用户，如果用户名存在，密码错误可点击找回密码，将会将新密码发送到该账户的邮箱，即可再登录。
2. 运行系统，首先出现登录界面（用户名密码均不可为空，输入不正确会提示错误），如图3-11所示；如果无账户登录系统，需要管理员添加新用户，如图3-12所示；如用户名存在忘记密码，可发送邮件找回密码，如图3-13所示；登录成功进入主界面，如图3-14所示：

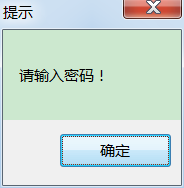
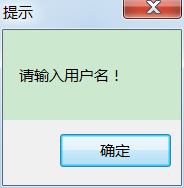
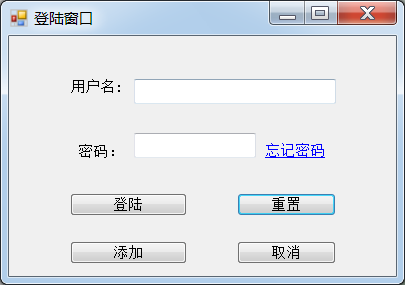


图3- 11 登录界面及错误提示



图3- 12添加新用户

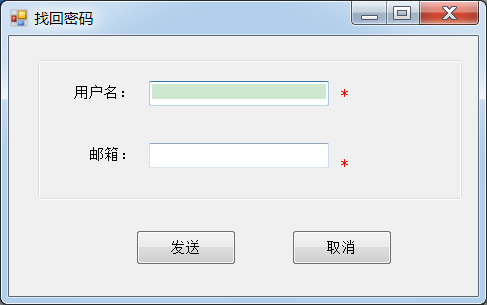


图3- 13发送邮件找回密码

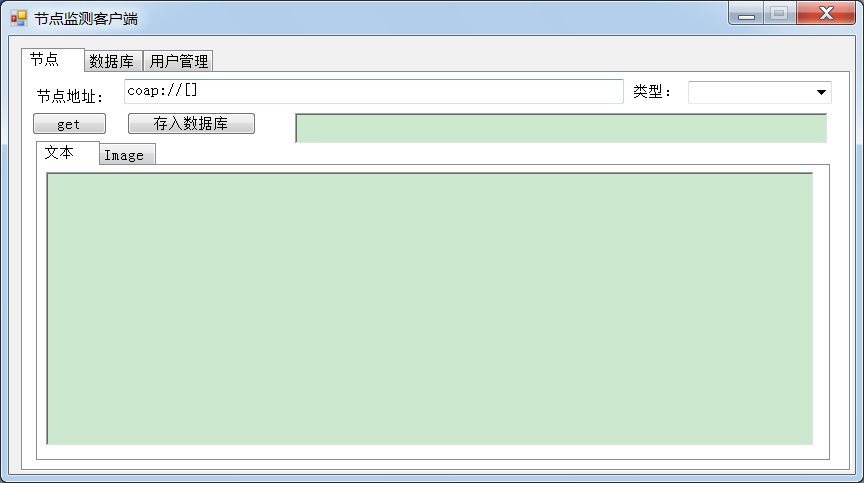


图3- 14主界面

3.5.2 系统各模块实现代码

列举三层架构实现添加用户的具体代码实现：

1. 数据访问层DAL.cs：

public bool ExecuteSQL(string sql)

{

SqlConnection con = new SqlConnection(DAL.ConnectionString);

SqlCommand cmd = new SqlCommand(sql, con);

try

{

con.Open();

cmd.ExecuteNonQuery();

return true;

}

catch (Exception e)

{

return false;

}

finally

{

con.Close();

con.Dispose();

cmd.Dispose();

}

}

1. 添加用户的业务逻辑层方法UserDataOperation.cs：

public static bool insertUser(UserData data)

{

string sql = "insert into [users](username,password,ismanager,sex,telphone, email) values("+ "'" + data.username + "'," +

"'" + data.password + "'," +

"'" + data.ismanager + "'," +

"'" + data.sex + "'," +

"'" + data.telphone + "'," +

"'"+ data.email+"')";

return dal.ExecuteSQL(sql);

}

1. 界面表示层AddUser.cs：

if(UserDataOperation.insertUser(data))

{

MessageBox.Show("添加成功！", "提示");

Login login = new Login();

this.Hide();

login.Show();

}

系统登录界面主要代码：

private void Btn\_DengLu\_Click(object sender, EventArgs e)

{

if (Tb\_UserName.Text == null || Tb\_UserName.Text.Trim().Equals(""))

{

MessageBox.Show("请输入用户名！", "提示");

this.Tb\_UserName.Focus();

return;

}

if (Mtb\_MiMa.Text == null || Mtb\_MiMa.Text.Trim().Equals(""))

{

MessageBox.Show("请输入密码！", "提示");

this.Mtb\_MiMa.Focus();

return;

}

MySqlConnection sqlconnection = new MySqlConnection();

sqlconnection.ConnectionString="server=localhost;user id=root;password=271828;database=jiedian";

sqlconnection.Open();

String ps = "'and password='";

String ed="';";

String sql = "select \* from jd\_user where username='" + Tb\_UserName.Text + ps + Mtb\_MiMa.Text + ed;

MySqlCommand mySqlCommand = new MySqlCommand(sql,sqlconnection);

MySqlDataReader reader = mySqlCommand.ExecuteReader();

try

{

if (reader.HasRows)

{

MessageBox.Show("恭喜你登陆成功！！");

this.Hide();

Form2 f2 = new Form2();

f2.Show();

}

else

{

MessageBox.Show("用户名或密码错误！！");

}

}

catch (Exception)

{

MessageBox.Show("读取失败！");

}

finally

{

reader.Close();

}

}

系统主界面主要代码

1）：获取节点信心代码：

private void button4\_Click(object sender1, EventArgs ee)

{

#region

this.Text = "获取节点信息！";

pictureBox1.Image = null;

richTextBox1.ResetText();

String payload = null;

Request request = new Request(Method.GET);

String re\_uri = textBox1.Text + "/" + comboBox1.Text;

request.URI = new Uri(re\_uri);

MessageBox.Show(re\_uri);

request.SetPayload(payload);

request.Timeout += new EventHandler(request\_Timeout);

try

{

request.Respond += OnResponse;

request.Send();

}

catch (Exception ex)

{

MessageBox.Show("Failed executing request: " + ex.Message);

}

#endregion

//Thread th = new Thread(ThreadChild);

//th.Start();

}

void OnResponse(Object sender, ResponseEventArgs e)

{

if (this.InvokeRequired)

{

this.Invoke(new EventHandler<ResponseEventArgs>(OnResponse), sender, e);

return;

}

Response response = e.Response;

//response.ContentType == MediaType.TextPlain, MediaType.ImageJpeg

if (response == null)

{

MessageBox.Show("Request timeout");

}

else

{

MessageBox.Show("Time (ms): " + response.RTT);

richTextBox1.Text = response.PayloadString;

if (response.Payload == null) MessageBox.Show("response.Payload == null");

else if (MediaType.IsImage(response.ContentType))

{

pictureBox1.Image = CoapClient.ImageByte.ByteToImage(response.Payload);

}

}

//if (!response.IsEmptyACK && !loop)

// Environment.Exit(0);

}

void request\_Timeout(object sender, EventArgs e)

{

MessageBox.Show("Request timeout");

}

static void ThreadChild()

{

MessageBox.Show("这是子线程执行的代码");

}

2）储存节点信息代码：

private void button1\_Click(object sender, EventArgs e)

{

if (pictureBox1.Image != null)

{ //将图片对象image转换成缓冲流imageStream

MemoryStream imageStream = new MemoryStream();

pictureBox1.Image.Save(imageStream, System.Drawing.Imaging.ImageFormat.Jpeg); //获得图片的字节数组imageByte

byte[] imageByte = imageStream.GetBuffer(); //建立数据库连接

MySqlConnection conn = new MySqlConnection("Server=localhost;Uid=root;Password=271828;Database=jiedian");

conn.Open(); //设置命令参数

string insertStr="insert into jd\_image(ip,time,image) values(?IP,?TIME,?IMAGEBYTE)";

MySqlCommand comm = new MySqlCommand();

comm.Connection = conn;

comm.CommandText = insertStr;

comm.CommandType = CommandType.Text; //设置数据库字段类型MediumBlob的值为图片字节数组imageByte

String ip1 = textBox1.Text.Trim();

comm.Parameters.Add(new MySqlParameter("?IP", MySqlDbType.VarChar)).Value = ip1;

String timenow = DateTime.Now.ToString();

comm.Parameters.Add(new MySqlParameter("?TIME", MySqlDbType.Datetime)).Value = timenow;

comm.Parameters.Add(new MySqlParameter("?imageByte", MySqlDbType.MediumBlob)).Value = imageByte; //执行命令

try {

comm.ExecuteNonQuery();

String str1 = "存入图片的时间为：";

MessageBox.Show(ip1);

rTB\_log.Text = str1+timenow +"\n存入图片的IP为："+ip1;

}

catch (Exception ex)

{

MessageBox.Show(ex.ToString());

}

comm.Dispose();

conn.Close();

conn.Dispose();

}

}

3）查询存储在本地的图片信息：

private void button3\_Click(object sender, EventArgs e)

{

pictureBox2.Image = null;

MySqlConnection sqlconnection = new MySqlConnection();

sqlconnection.ConnectionString = "server=localhost;user id=root;password=271828;database=jiedian";

sqlconnection.Open();

//String ps = "'and password='";

//String ed = "';";

//String sql = "select \* from jd\_user where username='" + Tb\_UserName.Text + ps + Mtb\_MiMa.Text + ed;

//string insertStr = "insert into jd\_image(ip,time,image) values(?IP,?Time,?Image)";

//MySqlCommand comm = new MySqlCommand();

//comm.Connection = conn;

//comm.CommandText = insertStr;

//comm.CommandType = CommandType.Text;

//comm.Parameters.Add(new MySqlParameter("?Image", MySqlDbType.MediumBlob)).Value = imageByte;

//comm.Parameters.Add(new MySqlParameter("?Time", MySqlDbType.Datetime)).Value = DateTime.Now.ToString();

//comm.Parameters.Add(new MySqlParameter("?IP", MySqlDbType.VarChar)).Value = textBox1.Text.Trim();

String chaxun = "select image from jd\_image where ip=?IP and time=?Time";

MySqlCommand comm = new MySqlCommand();

comm.Connection = sqlconnection;

comm.CommandText = chaxun;

comm.CommandType = CommandType.Text;

comm.Parameters.Add(new MySqlParameter("?Time", MySqlDbType.Datetime)).Value = tb\_time.Text.Trim();

comm.Parameters.Add(new MySqlParameter("?IP", MySqlDbType.VarChar)).Value = tb\_ip.Text.Trim();

MySqlDataReader dr = comm.ExecuteReader();

if (dr.Read())

{ //读出图片字节数组至byte[]

byte[] imageByte = new byte[dr.GetBytes(0, 0, null, 0, int.MaxValue)];

dr.GetBytes(0, 0, imageByte, 0, imageByte.Length); //将图片字节数组加载入缓冲流

MemoryStream imageStream = new MemoryStream(imageByte); //从缓冲流生成图片

Image image = Image.FromStream(imageStream, true);

pictureBox2.Image = image;

}

dr.Dispose();

comm.Dispose();

sqlconnection.Close();

sqlconnection.Dispose();

}

4）用户管理主要代码：

private void button2\_Click(object sender, EventArgs e)

{

try

{

MySqlConnection sqlconnection = new MySqlConnection();

sqlconnection.ConnectionString = "server=localhost;user id=root;password=271828;database=jiedian";

sqlconnection.Open();

String sql = "select \* from jd\_user";

MySqlCommand mySqlCommand = new MySqlCommand(sql, sqlconnection);

MySqlDataAdapter sda = new MySqlDataAdapter(mySqlCommand);

DataSet ds = new DataSet();

sda.Fill(ds, "用户表");

dataGridView2.DataSource = ds;

dataGridView2.DataMember = "用户表";

dataGridView2.AutoGenerateColumns = true;

for (int i = 1; i < this.dataGridView1.ColumnCount; i++)

{

this.dataGridView1.Columns[i].DefaultCellStyle.SelectionBackColor = Color.White;

this.dataGridView1.Columns[i].DefaultCellStyle.SelectionForeColor = Color.Black;

this.dataGridView1.Columns[i].ReadOnly = true;

}

sqlconnection.Close();

}

catch (Exception ee)

{

MessageBox.Show("错误：" + ee.Message, "错误");

}

}

private void button7\_Click(object sender, EventArgs e)

{

try

{

MySqlConnection sqlconnection = new MySqlConnection();

sqlconnection.ConnectionString = "server=localhost;user id=root;password=271828;database=jiedian";

sqlconnection.Open();

String sql = "select \* from jd\_user where username='"+textBox2.Text+"';";

MySqlCommand mySqlCommand = new MySqlCommand(sql, sqlconnection);

MySqlDataAdapter sda = new MySqlDataAdapter(mySqlCommand);

DataSet ds = new DataSet();

sda.Fill(ds, "用户表");

dataGridView2.DataSource = ds;

dataGridView2.DataMember = "用户表";

dataGridView2.AutoGenerateColumns = true;

for (int i = 1; i < this.dataGridView1.ColumnCount; i++)

{

this.dataGridView1.Columns[i].DefaultCellStyle.SelectionBackColor = Color.White;

this.dataGridView1.Columns[i].DefaultCellStyle.SelectionForeColor = Color.Black;

this.dataGridView1.Columns[i].ReadOnly = true;

}

sqlconnection.Close();

}

catch (Exception ee)

{

MessageBox.Show("错误：" + ee.Message, "错误");

}

}

3.5.3 数据库实现

数据库建立、连接及其数据库表设计

建立一个数据库jiedian及jd\_user表、jd\_image表、jd\_info 3张数据库表。

连接数据库

应用程序要访问数据库，就必须现在应用程序和数据库之间建立一个连接。我们可以为同一个数据库建立很多连接，不同的应用程序可以连接存取数据库里的数据。通过MySql.Data.MySqlClient命名空间下的MySqlConnection可以建立一条操作Mysql数据库的连接。

要建立数据库连接就必须先构造数据库连接字符串，连接字符串包括：

⊙ server：后面接数据库服务器的名称，表示本机的写法有“.”、“（local）”等。如果不是本机可以写对方数据库服务器的IP或服务器名称。

⊙ database：你要访问数据库的名称。

⊙ uid：登录数据库帐户的名称。

⊙ pwd：登录数据库帐户的密码。

有了数据库连接字符串我们就可以直接建立数据库连接对象了，如下：MySqlConnection conn =new MySqlConnection("Server=localhost;Uid=root;Password=271828;Database=jiedian");

建立的数据库连接之后，如果要通过该连接操作数据库里面的数据，还需要打开该连接：

conn.Open();

如果该数据库连接使用完了，可调用Close方法关闭该连接，如：

conn.Close();

要注意每个数据库能接受的连接是有限的，每个连接都可以为一个应用程序服务，所以数据库连接式非常宝贵的资源。

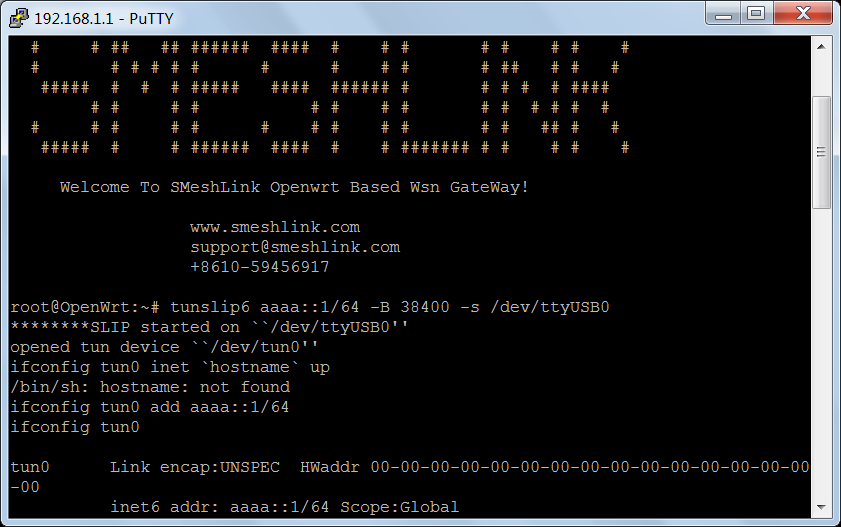
第四章 面向IPv6无线传感网的信息收集平台运行与测试

4.1平台的搭建

使用WinAVR和AVR studio软件开发平台把Contiki Studio集成环境开发平台编译好的camera示例程序烧写到节点上并按照图1-8连接好相关硬件。摄像头、扩展板和无线节点的连接如下图所示。



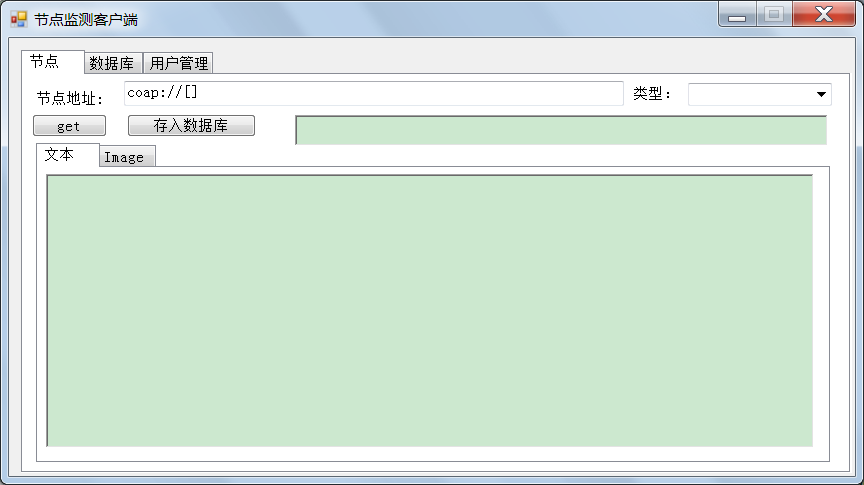
此时还需要对网关进行配置，使用putty登陆到网关并输入命令如图所示：



配置完成后使用cmd ping节点，显示可以ping通节点，配置完成。平台搭建完成。

4.2 节点监测客户端运行与测试

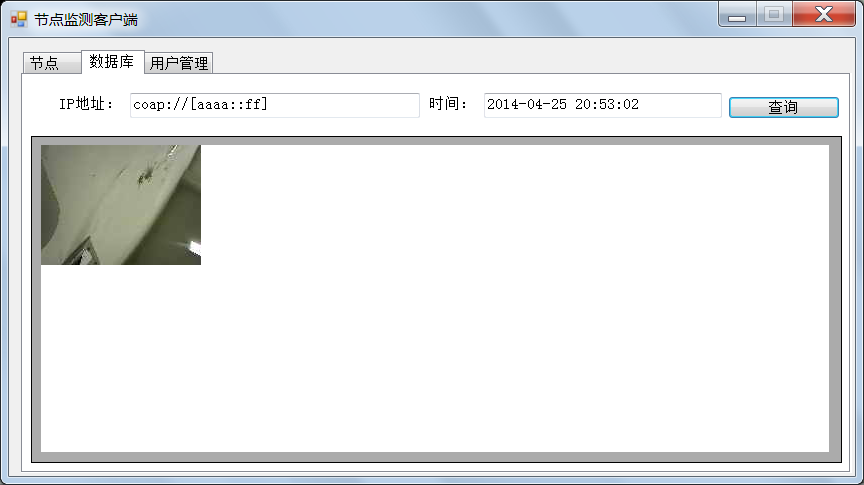
在计算机端启动系统，首先是登陆界面，输入用户名和密码登陆到系统，登陆后显示的系统界面如下图所示。



测试能否通过本系统访问到节点上的图像信息：节点在节点地址栏里输入相应的节点地址，并在类型栏里选择相应的节点的类型，点击get按钮即可获取相应的节点上的信息。



点击存入数据库按钮即可将相应的数据存入到数据库中，存入到数据库中的历史数据可通过节点地址和时间查询获得，上图即为从节点地址为aaaa::1122上获取图像的界面。



通过IP和时间查询图像信息

在地址栏输入不同的节点地址均可获取到相应的数据，说明系统可以正常运行。

第五章 总结

随着计算机网络技术的不断发展，越来越多的设备需要接入网络。无线传感器网络综合了传感器技术、嵌入式计算技术、现代网络与无线通信技术，其协作地感知、采集、处理、发布信息的功能特点，以及无需固定设备支撑、快速部署、易于组网的性能优势，使得传感器网络作为人类信息采集能力的延伸，成为当前信息技术领域的一项重要的前沿技术。

6LoWPAN 技术使得无线传感器网络的节点自由接入互联网成为了可能，它精简了 IPv6协议的功能，对必需的部分进行了保留和修改，对非必需的部分进行了裁剪和删除。它实现了 IPv6 网络与基于 IEEE 802.15.4 标准的传统无线传感器网络的无缝连接。对于实现物联网和基于 IPv6 的互联网的全面融合具有划时代意义。在未来五到十年里，相互融合的物联网和互联网会形成真正的全 IP 网络， 这样任何人会在任何时间，任何地点，通过任何终端都能获取所需要的任何信息，这是未来信息社会的美好前景。而 6LoWPAN 技术的采用将从根本上解决物联网感知延伸层传感器节点接入互联网的问题，实现传感器网络末梢节点的 IP 化。IPv6传感器网络是IPv6技术与无线传感器网络的融合，具有两者各自部分特征，同时也具有其独特性。这种独特性，决定了要采用 6LoWPAN 技术重新构建基于IPv6的传感器网络体系结构。互联网工程任务组的6LoWPAN工作组致力于将IPv6应用在IEEE802.15.4之上。从而实现无线传感器网络和IP网络之间的无缝连接。

本平台完成了对节点资源的访问，并且实现了对历史信息的管理操作等功能。