

# TiGW1000 可编程边缘计算网关开发指南

## 目 录

### TiGW1000 可编程边缘计算网关开发指南

目 录

产品介绍

准备开发环境

    安装TiStudio

    创建TijOS Application工程

    编码

    下载、运行

TiDevManager设备管理器

TiGW1000 编程开发说明

    TiGW1000 Java类使用说明

        外设访问

            RS485

            RS232

            可控LED

            继电器

            数字量输出DO

            数字量输入DI

TiGW1000 外设访问组件

    TiSerialPort 串口类主要方法使用说明

    TiLED灯主要方法使用说明

        开灯turnOn

        关灯turnOff

        灯闪

MODBUS RTU组件

    主要接口

        参数说明

    调用过程

    例程

        代码调用过程

网络访问

    LAN网络设置 - TiLAN

云端接入

    云平台接入组件

    云端接入例程

        TijOS JDK 常用功能例程

技术支持

更多资源

联系方式

## 产品介绍

TiGW1000是钛云物联基于钛极OS(TiJOS)物联网操作系统开发的可编程边缘计算网关，用户可通过Java语言开发控制器内部的应用和控制逻辑以及与云平台交互过程。

以下介绍如何在TiGW1000网关进行应用开发, 更多内容可参考相关例程及钛OS(TiJOS)文档中心 <http://doc.tijos.net>

## 准备开发环境

### 安装TiStudio

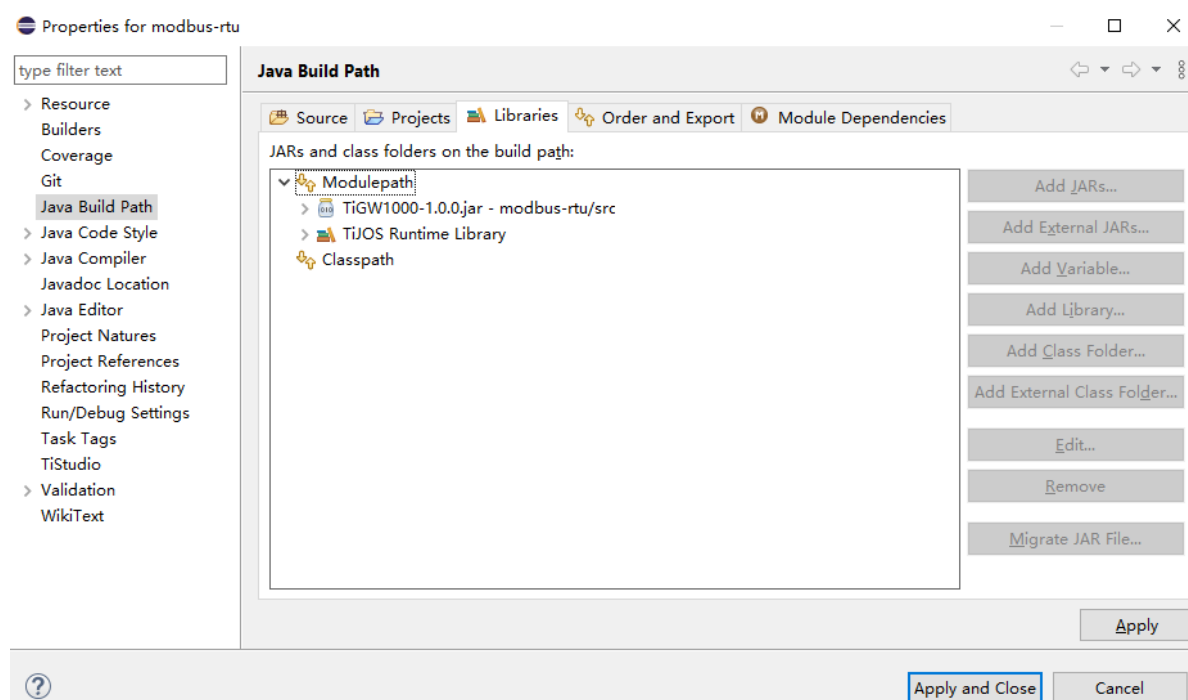
在进行开发之前，请先安装Eclipse开发环境及TiStudio开发插件，请参考如下链接进行开发环境的安装。

[开发环境搭建与开发板测试 - 文档中心 \(tijos.net\)](#)

## 创建TiJOS Application工程

TiGW1000提供了相关例程，用户可直接使用Eclipse打开例程进行修改或者新建一个TiJOS Application工程，具体过程请参考[欢迎来到TiJOS世界—新建工程Hello TiJOS - 文档中心](#)

在新建工程后，将TiGW1000-1.0.x.jar加入到工程中，将在工程属性中将该jar包加入到Java Build Path中，如下图所示：

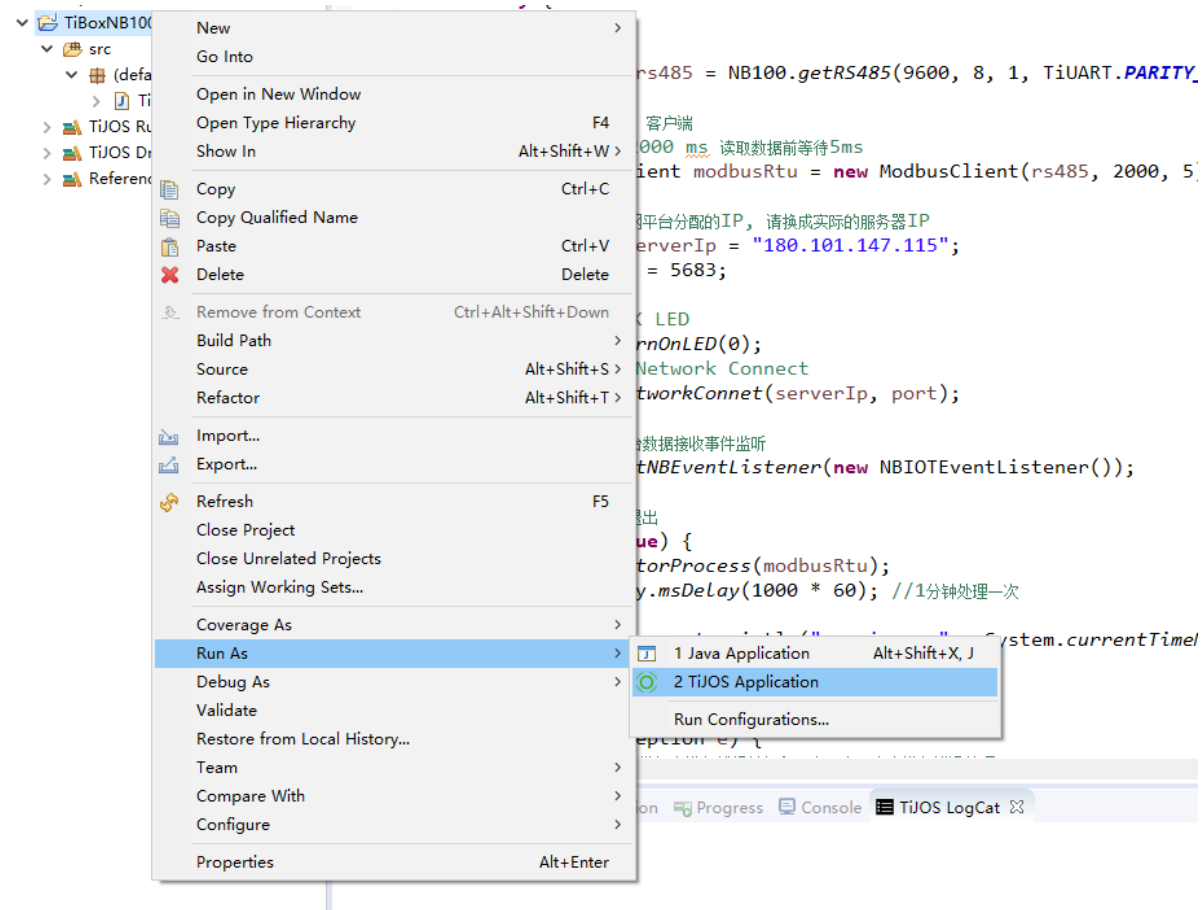


## 编码

此时，即可在Eclipse中进行相应的代码编写。

## 下载、运行

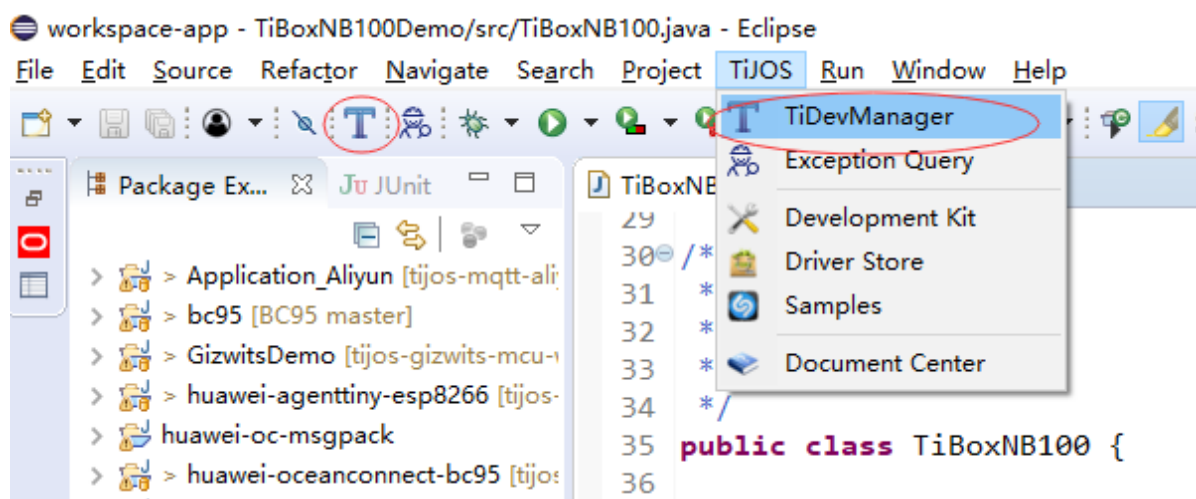
代码无误后，可通过Run As菜单选择"TiJOS Application"运行，在运行之前请确保已正确连接在TiBOX的USB编程口，可从TiJOS LogCat中查看日志或打开TiDevManager查看日志。



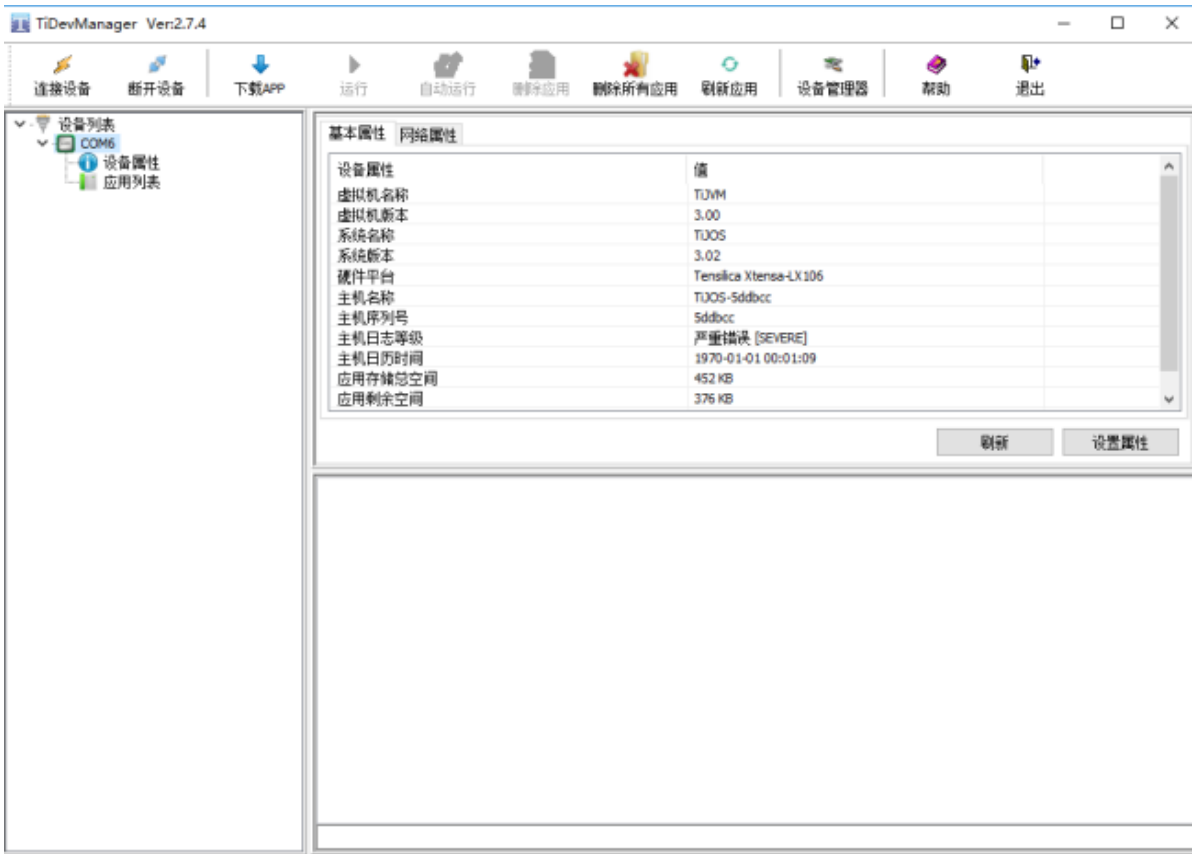
## TiDevManager设备管理器

TiDevManager设备管理器是钛极OS(TiJOS)开发套件TiStudio的组成部分，用于查看设备信息及应用管理的工具，也可单独运行，详细使用方法请参考[初识TiDevManager设备管理器 - 文档中心\(tijos.net\)](http://tijos.net)

TiDevManager可通过Eclipse的菜单启动。



启动后，可连接设备查看设备及应用信息



## TiGW1000 编程开发说明

TiGW1000 内置钛极OS(TiJOS) 操作系统，支持通过Java语言进行应用开发，可通过钛极OS(TiJOS) 开发工具链IDE进行应用开发，钛极OS(TiJOS)在线文档可参考 [doc.tijos.net](http://doc.tijos.net)

## TiGW1000 Java类使用说明

### 外设访问

tigateway.TiGW1000类提供了TiGW1000所支持的硬件资源访问，包括RS485,RS232,继电器，LED等，同时内置看门狗，用户可通过在TiStudio中进行简单的开发即可支持各种应用，同时提供了的MODBUS协议类，可以很方便地与支持MODBUS RTU协议的设备进行数据交互。

#### RS485

支持1路RS485, 支持最大驱动32个RS485从设备

#### RS232

支持1路RS232

#### 可控LED

提供1个红色电源指示灯和2个可编程控制LED灯，对应于蓝色和绿色，方便用户在代码中指示设备的当前状态，建议蓝色灯用于指示网络通讯状态，绿色灯用于指示设备通讯状态。

继电器

支持两路继电器

数字量输出DO

支持4路数字量输出DO

数字量输入DI

支持4路数字量输入DI

## TiGW1000 外设访问组件

为了方便用户在程序中访问外设，TiGW1000提供了JAVA类开源组件

方法	说明
<b>RS485</b>	
TiSerialPort getRS485(int baudRate, int dataBitNum, int stopBitNum, int parity)	获取RS485接口， 参数：波特率，数据位，停止位，校验位
<b>RS232</b>	
TiSerialPort getRS232(int baudRate, int dataBitNum, int stopBitNum, int parity)	获取RS232接口， 参数：波特率，数据位，停止位，校验位
<b>LED灯</b>	
TiLED blueLED()	获取蓝色灯对象
TiLED greenLED()	获取绿色灯对象
<b>继电器</b>	
void relayControl(int id, int ctl)	继电器开关控制，id 0或1号继电器，ctl: 0关 1开
<b>数字输出控制DO</b>	
void digitalOutput(int id, int ctl)	数字输出，id 0或1，ctl: 0关 1开
<b>数字输入DI</b>	
int digitalInput(int id)	数字输入，id 0-3 返回 0关 1开

## TiSerialPort 串口类主要方法使用说明

通过getRS485/getRS232获取串口后，即可对串口进行读写操作

方法	说明
<code>void write(byte [] buffer ,int start ,int length)</code>	写入数据到串口 buffer: 待写入数据 start 缓存区开始位置 length 写入长度
<code>boolean readToBuffer(byte[] buffer, int start, int length, int timeOut)</code>	从串口读取指定长度数据 buffer: 读入数据缓存区, start 缓存区开始位置, length 读取长度, timeOut超时, 单位毫秒
<code>byte [] read(int msec)</code>	从串口读数据, msec 最大毫秒数, 当有数据时从串口指定时间的数据返回, 如果没有数据则返回null

## TiLED灯主要方法使用说明

通过blueLED/greenLED可以获取蓝灯和绿灯对角, 并进行如下操作

### 开灯turnOn

指定的灯亮

### 关灯turnOff

指定的灯灭

### 灯闪

可通过turnOn/turnOff组合实现灯闪效果, 为不影响主程序运行, 也可启动一个专门的线程来实现闪烁的效果。

## MODBUS RTU组件

Modbus 一个工业上常用的通讯协议, 其中MODBUS-RTU在支持串口的设备中最为常用, 目前市面上很多设备和传感器都提供MODBUS-RTU的协议支持。

TiGW1000提供了相应的MODBUS RTU组件方便用户在代码中直接访问MODBUS RTU设备。

## 主要接口

函数	说明
ModbusClient(TiSerialPort serialPort, int timeout)	初始化, serialPort: 串口通讯对象, timeout: RS485设备通讯超时时间
<b>初始化寄存器请求</b>	根据参数构造MODBUS寄存器请求
initReadCoilsRequest(int deviceId, int startAddress, int count)	初始化Read Coils 请求
initWriteCoilRequest(int deviceId, int coilAddress, boolean value)	初始化WRITE COIL register 请求- 单寄存器操作
initWriteCoilsRequest(int deviceId, int startAddress, boolean[] values)	初始化WRITE MULTIPLE COILS registers 请求- 多寄存器操作
initReadHoldingRegistersRequest(int deviceId, int startAddress, int count)	初始化READ HOLDING REGISTERs 请求
initReadDiscreteInputsRequest(int deviceId, int startAddress, int count)	初始化READ DISCRETE INPUT REGISTERs 请求
initReadAnalogInputsRequest(int deviceId, int startAddress, int count)	初始化READ INPUT REGISTERs 请求
initWriteRegisterRequest(int deviceId, int regAddress, int value)	初始化WRITE SINGLE REGISTER 请求 - 单寄存器操作
initWriteRegistersRequest(int deviceId, int startAddress, int[] values)	初始化WRITE MULTIPLE 请求 - 多寄存器操作
<b>执行请求</b>	发送MODBUS请求到设备并获取返回数据
int execRequest()	执行MODBUS 请求并获得MODBUS响应结果
int getResponseAddress()	获取返回的MODBUS寄存器开始地址
int getResponseCount()	获取返回的MODBUS寄存器个数
boolean getResponseBit(int address)	获取指定地址COIL寄存器值, 对应于 InitReadCoilsRequest/InitReadDiscreteInputsRequest
int getResponseInt16(int address, boolean unsigned)	获取指定地址InputRegister/HoldingRegister的值, unsigned: 返回值 为无符号或有符号整数
int getResponseInt32(int address, boolean bigEndian)	获取指定地址InputRegister/HoldingRegister的连续2个寄存器值组成1个32位有符号整数, bigEndian:是否为大端格式

函数	说明
float getResponseFloat(int address, boolean bigEndian)	获取指定地址InputRegister/HoldingRegister的连续2个寄存器值组成1个32位浮点数， bigEndian:是否为大端格式

### 参数说明

- deviceId** - MODBUS 设备地址
- startAddress** - MODBUS 寄存器开始地址
- count** - 寄存器个数
- regAddress** - MODBUS 寄存器地址

### 调用过程

MODBUS 组件的调用过程一般为：

1. 指定串口实例化MODBUS对象， ModbusClient(TiSerialPort serialPort, int timeout)
2. 初始化MODBUS寄存器读写操作请求， initXXXRequest
3. 执行MODBUS请求并返回结果 execRequest
4. 根据设备协议中定义的数据类型获取返回的寄存器值  
getResponseBit/getResponseInt16/getResponseInt32/getResponseFloat

### 例程

**场景：**  
设备通过RS485连接到TiGW1000， 通讯MODBUS RTU协议进行数据交互

#### 设备通讯参数

参数	值
设备 ID	1
波特率	9600
数据位	8
停止位	1
校验位	无

**寄存器：** INPUT REGISTER (03)



寄存器地址	内容	操作权限	数值范围
0x0000	空气湿度	只读	0x00(0)--0x03E7(999) 对应 0%--99.9% 数值放大了10倍
0x0001	空气温度	只读	0x8190(-400)--0x0320(800) 对应 -40℃--80℃ 负数

## 代码调用过程

1. 打开RS485并获取TiSerialPort对象

```
TiGW1000 gw1000 = TiGW1000.getInstance();  
//获取RS485  
TiSerialPort rs485 = gw1000.getRS485(9600,8,1,0);
```

2. 创建MODBUS协议对象并挂接RS485

```
//MODBUS 客户端  
//通讯超时2000 ms  
ModbusClient modbusRtu = new ModbusClient(rs485, 2000);
```

3. 通过MODBUS协议读取寄存器数据

```
// MODBUS device address 设备地址  
int deviceId = 1;  
// Input Register 寄存器开始地址  
int startAddr = 0;  
// Read 2 registers from start address 读取个数  
int count = 2;  
  
//初始读取Holding Register参数， 设备地址， 寄存器开始地址， 个数  
modbusRtu.initReadHoldingRequest(deviceId, startAddr, count);  
int result = modbusRtu.execRequest();  
  
//读取成功进行数据解析  
if (result == ModbusClient.RESULT_OK) {  
    //获取第1个寄存器值 - 温度  
    int temperature = modbusRtu.getResponseInt16(startAddr, false);  
    //获取第2个寄存器值 - 湿度  
    int humidity = modbusRtu.getResponseInt16(startAddr + 1, false);  
}
```

## 网络访问

TiGW1000支持 以太网网络接入云平台。

用户可通过钛极OS运行库中TiLAN来进行网络的控制， 在程序运行开始通过startup来启动网络即可。

# LAN网络设置 - TiLAN

TiLAN中包含了所有与LAN网络相关的操作和设置  
TiLAN为单例，在操作网络时可通过getInstance获得实例并调用相应的方法。

主要方法如下：

方法	说明
TiLAN getInstance()	获取LAN实例
void startup(int timeout)	启动LAN并连接路由器, timeout以秒为单位
void startup(int timeout, INetworkEventListener lc)	启动LAN并连接路由， 当连接或断开时通过事件通知，断开后会自动尝试重新连接
void shutdown()	关闭LAN, 一般不调用
String getLocalIP()	获取IP
String getMAC()	获取MAC
String getNetMask()	获取子网掩码
String getGateway()	获取路由网关IP
void setStaticIP(String ip, String netmask, String gateway)	设置静态IP

TiLAN类中他方法的技术说明请参考TijOS Framework说明文档。

## 云端接入

TiGW1000内置钛极OS(TijOS)物联网操作系统, 不仅支持Java标准socket，同时也提供各种云平台接入专用组件, 包括MQTT, CoAP, 阿里云，腾讯云等等， 同时也可接入其它支持相关技术的云平台，可参考相应的例程。

## 云平台接入组件

tijos.framework.networkcenter

TijOS 网络中心包括如下包：

包名称	说明
tijos.framework.networkcenter.mqtt	MQTT客户端支持包，支持MQTT3.1.1标准
tijos.framework.networkcenter.http	HTTP 客户端
tijos.framework.networkcenter.coap	COAP 客户端
tijos.framework.networkcenter.lwm2m	LWM2m客户端
tijos.framework.networkcenter.alibaba	阿里云物联网平台客户端
tijos.framework.networkcenter.tencent	腾讯云IoT Explore物联网平台客户端

## 云端接入例程

钛极OS(TiJOS)提供了各种例程方便用户快速接入各种平台，相关例程请参考如下链接

[软件应用相关例程 - 文档中心\(tijos.net\)](http://tijos.net)

## TiJOS JDK 常用功能例程

TiJOS JDK与标准Java JDK兼容，本目录下列出了一些基于标准Java类的常用例程，方便用户参考


目录	说明
helloworld	Hello TiJOS
datatype\BitOperation	java下的位操作及移位操作例程
math	常用数据函数例程，加减乘除，三角函数等等
file	文件操作例程，文件创建删除，读写等等
thread	多线程例程，经典生产者消费者例程
network\NTP	基于NTPUDPClient的网络时间协议例程
network\tcp_client	基于SOCKET的TCP 客户端例程
network\tcp_server	基于ServerSocket和Socket的TCP服务器例程
network\udp	基于DatagramSocket的UDP例程
network\dns	基于InetAddress的DNS例程
base64	BASE64编码解码例程
json	JSON字符串生成和解析例程
MQTT	基于MqttClient的MQTT客户端例程
Alibaba	阿里云物联网平台例程
Tencent	腾讯云物联网平台例程

## 技术支持

如果您有任何技术问题，可通过电话，QQ群等方式与我们联系，同时钛云物联可提供产品定制，通讯协议开发，云端接入，技术培训等多种服务。

## 更多资源

TiGW1000是钛云物联的钛极OS(TiJOS)物联网操作系统的一个典型应用，关于钛极OS(TiJOS)物联网操作系统可参考如下资源：

资源	url
钛极OS官网	<a href="http://www.tijos.net">www.tijos.net</a>
钛极OS文档中心	<a href="http://doc.tijos.net">doc.tijos.net</a>
微信公众号 - 钛极OS	

## 联系方式

北京钛云物联科技有限公司

商务合作：13911058165

品牌热线：010-86462928

公司网址：[www.tijos.net](http://www.tijos.net)

电子邮件：[tijos@tijos.net](mailto:tijos@tijos.net)

在线购买：<https://shop423269048.taobao.com/>