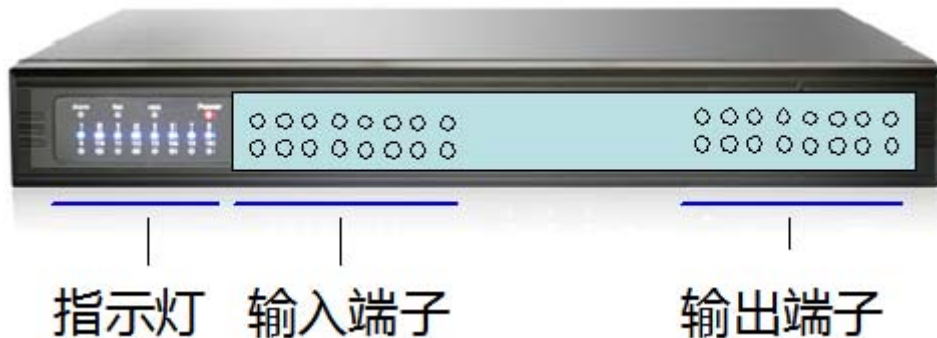


Visual NetIO快速使用手册

1 概述

Visual NetIO（可视网络IO）—— 可视网络远程控制开关、开关量信号回传设备。



本产品支持网络（目前为以太网，可定制WiFi/3G/GPRS等）方式控制开关、回传开关量信号。同时可以将现场图像（图片）回传，**实现可靠、可视的控制**。在高可靠性的同时也极大地方便用户监控工作现场、远程维护、排查故障等。

1.1 应用领域

安全监控系统、工业控制应用系统、室内娱乐系统、温湿度环境监控、楼宇智能控制、电力监控系统、智能家居。

1.2 特点

组合最多可达24路输入或输出通道，包括DI,DO,AD,DA,温湿度等

系统具有视频功能，可以实时查看你的设备。

设备联网，随时随地监测和控制你的设备

可跨越网关，真正实现互联网操作

内置HTTP协议，打开网页即可实现自由观察和控制

完善的密码验证机制，确保只有授权用户才能操作

可定制网页、LOGO

可内置硬盘

支持普通的串口服务器功能，TCP与RS232实现透明传输

同时支持TCP client和TCP server模式

两级认证模式——用户组、用户

Visual NetIO

多重可靠性设计，更强的防浪涌和抗干扰能力

内置看门狗

高速CPU，专用网卡芯片，保障系统稳定性和可靠性

可定制支持wifi、GPRS、3G等各种链接方式。

支持远程升级

支持定时控制

支持设备自主运行，断电后重启并恢复停电时的控制状态。

模块化设计，方便扩展。

1.3 规格

系统模块化设计，分为主控模块和接口模块，接口模块采用全光电隔离等可靠设计，以4路输入和输出为一组，可以按需扩展，如不受机箱限制，可以扩展至百路以上，且模块间互不影响。

电源电压：12V

电源电流：最大1A

1. 输入接口

光电隔离

2. 输出接口

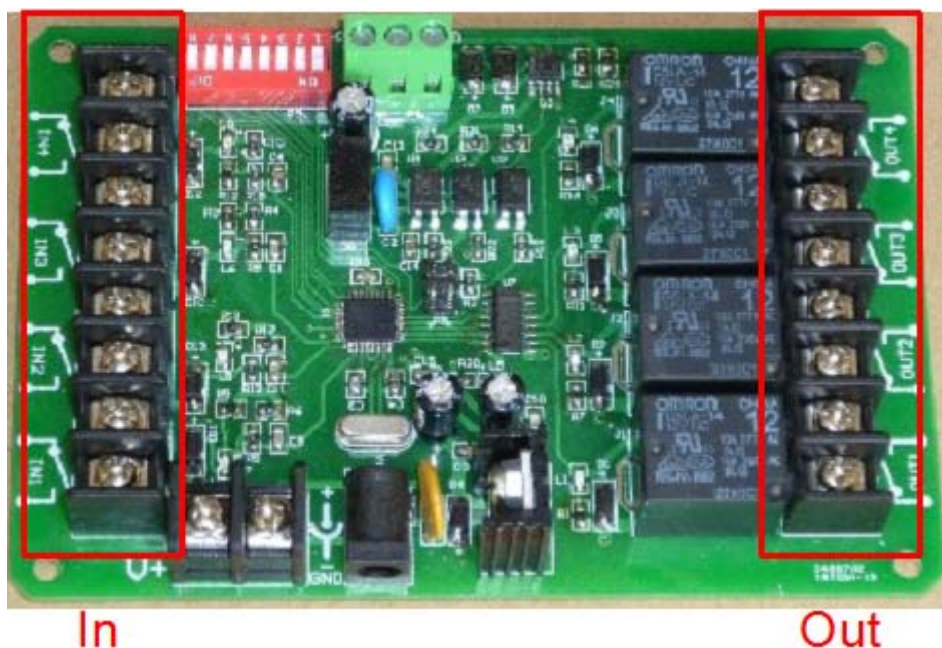
触点容量：AC250V10A，DC30V10A

2 使用

2.1 接线

本系统分文接口模块和主控模块，接口模块同主控模块在出厂时已经过严格测试，用户只需按接口板上的指示进行接线即可。即输入量

可以直接在前面板处接线；如果为了安全，也可打开机箱，直接接在接口板上，然后关好机箱，这样您的系统更加安全可靠。接口板如下：



2.2 WEB方式

正在开发——目前请使用dvriper工具查看控制效果。

系统默认IP: 192.168.100.62, IO控制端口: 8080

<http://192.168.100.62:8080>

2.3 TCP Server方式

正在开发——目前请使用dvriper工具查看控制效果。

2.4 TCP Client方式

正在开发——目前请使用dvriper工具查看控制效果。

3开发

为方便集成入您的系统，提供多种平台的SDK和使用协议的示例源码。

3.1 SDK

正在开发

提供windows、linux 版本的静态和动态库

3.2 使用协议——示例代码

Visual NetIO

3.2.1 示例工具说明

命令行下:

```
dvriper.exe [ip] [port]
[ip] dvr的ip
[port] dvr的tcp端口
```

软件正常运行后，会不断修改输出继电器的状态，可以听到噼啪噼啪的继电器工作声音。

默认的ip和端口在代码（Dvriper.cpp）中定义为:

```
//default
#define DEFAULT_IP "192.168.100.62"
#define DEFAULT_PORT 9001
```

主要文件说明:

Dvriper.cpp: main

DvrCmd.cpp: 功能函数实现

其他文件为辅助和定义文件。

3.2.2 获取端口状态

设备支持[查询](#)和有变化之后[主动回传（订阅）](#)端口状态两种方式来获取端口状态

1. 查询模式示例

```
// 获取端口 变化情况 ok
// Test_GetExtIO(0, 0);
```

查询:

客户端正常登录设备之后，调用接口。

```
void cmd_Analisys(void *pdip, size_t len)
else if (REQ_COMM_SEND == pDvrrip->dvrrip_cmd)
{
    printf("on extio out\n");
    if (pDvrrip->dvrrip_extlen == sizeof(EXT_IO_STATUS_T))
```

2. 订阅模式示例

```
// 订阅端口变化情况
Test_BookExtIO(0, 0);
```

```
__ip_head.dvrrip.dvrrip_p[0] = 1; // io模块类型， NIO系列产品固定为1
__ip_head.dvrrip.dvrrip_p[1] = 5; // 5-- 查询当前io状态操作码
```

3.2.3 控制输出端口

一次可以控制一个端口、多个或者全部输出端口

Visual NetIO

示例：

```
void Test_SetExtIO(int sockfd, int times)

//数据格式： 通道||开(1)关(0)&&通道||开(1)关(0) ....
//      控制各个通道之间用&&分割， 一次可以传输一个或者多个通道
//      通道从0开始， 其后的数字为1表示让继电器通（既俗称打开开关状态），
//      为0 表示让继电器断开（既俗称关掉开关状态）
//      当通道号填为'-1'（如下）， 即实现一次控制所有通道的功能， 其后的数字的
//      二进制的每一位来对应每一个继电器
//      如 0x05： 二进制为： 0000 0101， 则一次将第0和第2个继电器吸合， 其
//      他路继电器松开
```

如有任何问题，请留言或发邮件：faceaip@gmail.com 我们会及时解答您的问题。

Catalog 目录

- 1 概述 1
 - 1.1 应用领域 1
 - 1.2 特点 1
 - 1.3 规格 2
- 2 使用 2
 - 2.1 接线 2
 - 2.2 WEB方式 3
 - 2.3 TCP Server方式 3
 - 2.4 TCP Client方式 3
- 3 开发 3
 - 3.1 SDK 3
 - 3.2 使用协议的示例代码 3
 - 3.2.1 示例工具说明 4
 - 3.2.2 获取端口状态 4
 - 3.2.3 控制输出端口 4

Revision record 修订记录

日期	修订版本	修改章节	描述	作者
2011-11-15	1.0	All	创建	



Manual version 1.00
Last update: 2011-12-02 13:51:33
<http://www.faceaip.com>
faceaip@gmail.com