



Networked IO Board

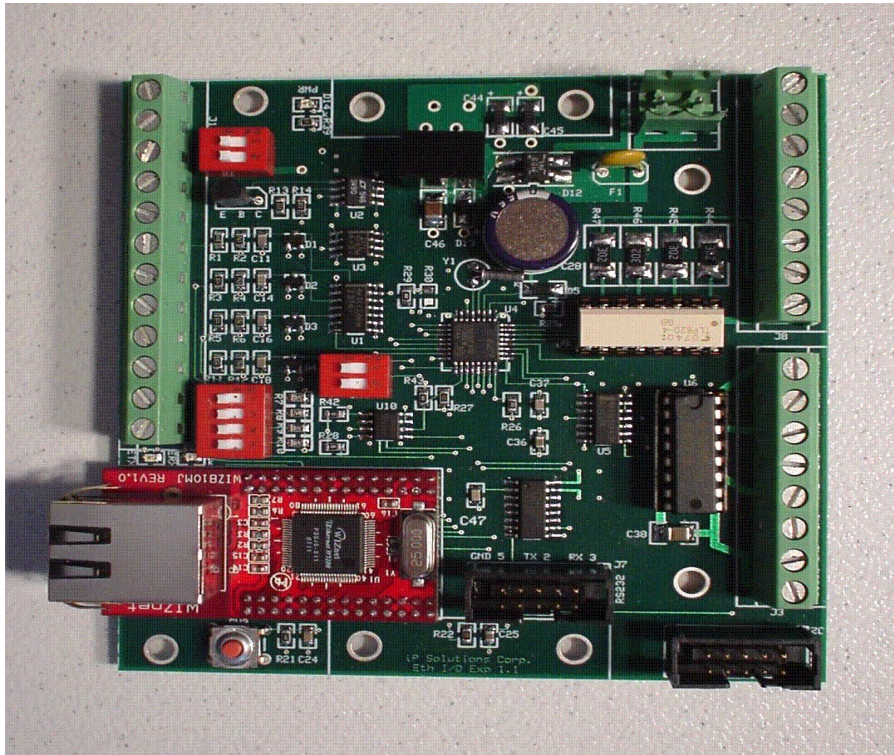
Features

- 4 V/I Analog Inputs 0-11V/0-44mA
- 1 V/I DAC Output 0-10V/0-20mA
- 4 Opto-isolated Digital Inputs
- 3 Relay Driver Digital Outputs
- RTC with 64 bytes SRAM, 1 week backup with super capacitor
- On-Chip 8051 Temperature Measurement
- On-Board Temperature Measurement
- 2-Pin User Input Switch
- 50MIPS 8051 with 32KB Flash, 2304 Byte SRAM
- 512Kbytes Storage Flash
- Programming APIs for C, C++, and Visual Basic as well as a .NET component interface
- Modbus TCP and Modbus UDP support
- 1 RS232 Interface with 10 pin header
- 1 10/100 Ethernet Port with RJ45 Connector
- Up to 4 Frequency or Period Measurement Channels
- Up to 3 Output Pulse or Frequency Output Channels
- DIN Rail or Panel Mounting
- 4" X 4" Board Size

Description

The iP Solutions Networked IO board is a 10/100 Ethernet enabled data acquisition and control board useful in networked embedded applications. It can be used to add analog and digital I/O or to expand I/O over an Ethernet network.

A simple TCP/IP client command protocol executed over an Ethernet 10/100 network is used to rapidly program analog and digital outputs on the board as well as configure the board resources. Monitored analog and digital data is sent back to a host using a UDP client at user defined update rates. Easy to use programming APIs are available for C, C++ and Visual Basic languages. A ready to use .NET component is also available. Additionally, the board supports Modbus TCP and Modbus UDP.





Networked IO Board

DC Specifications T=25 degrees C

Parameter	Description	Minimum	Typical	Maximum	Units	See Notes
Vsup	Min-Max supply voltage	12		24	Volts	
Isup	Nominal power supply		300		mA	
Ain	Voltage input mode range	0		11.0	Volts	1,3
Ain	Voltage input mode impedance		300K		Ohms	1,3
Ain	Current input mode	0		44.2 (see	mA	1,2,3
Ain	Current input mode Z	246.5	249	251.5	Ohms	2,3
Din	Digital input voltage range	0		30	Volts	
Din	ViL	0		1	Volts	
Din	VIH	1	1.7	30	Volts	6
Dout	Output voltage range	0		30	Volts	
Dout	Output current range	0		400	mA	
Dout	Output Vol @ 100 mA	1	1.5	2	Volts	
DAC	Voltage output mode	0		10	Volts	4,5
DAC	Current output mode	0		20	mA	4,5

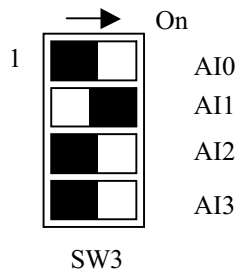
Notes:

1. Basic measurement resolution is 12 bits. This is 2.686mV per step for voltage mode and 10.788uA per step for current mode.
2. Limit input current to $\leq 20\text{mA}$ to prevent excess power dissipation in input termination resistor.
3. Set Switch SW3 input channel position to ON for voltage input mode and OFF for voltage input mode.
4. Basic programmable output resolution is 12 bits. This is 2.442mV per step for voltage mode and 4.884uA per step for current mode.
5. Set position 1 of SW1 to ON for current mode and OFF for voltage mode.
6. If DICOM is tied to a high voltage then VIH is limited to no greater than the DICOM level.

Pin Descriptions

Analog ADC inputs

Four analog inputs are available, AI0 – AI3. Each analog input can be configured as either a 0–11.0 volt input or as a 0–44.2 mA input. Configuration is accomplished by switching in a 249-ohm resistor from input to ground using DIP SW3. For example, channel AI1 is put into current mode measurement by switching SW3 position 2 ON (See the example below). The other SW3 switch positions similarly switch any of the other AI channels to the current measurement mode. Due to input termination resistor power dissipation limits, the input must be restricted to 0–20 mA for this mode.



All analog input measurements have a basic 12-bit resolution. This results in a 2.686 mV resolution for voltage measurements and a 10.788- μ A resolution for current measurements.

Analog Output

One DAC output is available. This output can be switched between a 0–10 volt output or a 0–20 mA output. Use SW1 to change this mode. Switch SW1 position 1 to the ON position to put the DAC into the voltage output mode. Both modes have a basic programmable resolution of 12 bits. This results in a 2.442 mV resolution for voltage mode and a 4.884- μ A resolution for current mode.

Digital Inputs

Four optically isolated digital inputs are available for general-purpose on/off inputs or for measuring frequency or period of an input waveform. These inputs are used in conjunction with the DICOM input. DICOM may be connected to either ground or a high voltage such as V_p . Other voltages up to 30 volts may be used instead of V_p . When DICOM is grounded, a V_{IH} level on the input turns on the channel and the processor will read a “0” level. When DICOM is at a high voltage and the input is at a V_{IH} level, the channel is off and the CPU will read a “1”. The level chosen for DICOM therefore can invert the logic. Inputs may range as high as 30 volts but should be no higher than DICOM if DICOM is connected to a high voltage. V_{IL} is approximately 0 to 1 volts and V_{IH} is approximately 1.7–30 volts.

The inputs default to simple on/off operation, however, one or more of the digital inputs can be configured to measure frequency or period of an input waveform.

Digital Outputs

Three digital outputs are available for driving relays or for pulsing the output at some frequency. These outputs are open collector with a voltage range of 0–30 volts and up to 400 mA per output. If used in the pulse or frequency mode, a resistor must be used to pull up the output collector to a supply voltage such as 5V or 12V (30V maximum).

Use the DIODE pin to clamp the high voltage that the open collector outputs are pulled up to. For example, if the output is tied to 24 volts through a relay, then the DIODE pin could be connected to 24 volts. If an output is pulled up to a different level, then each its output should be clamped with an external diode from the output to the pull-up voltage to prevent excessive overshoot.



Networked IO Board

The outputs default to simple digital on/off operation, however, one or more of the outputs may be configured as a frequency or pulsed output.



Networked IO Board

Programming and Configuration

The following section describes the communications protocol between a host and an Ethernet I/O Expansion board. For those who wish to program at a higher level, programming APIs for C, C++ and Visual Basic are available as well as a .NET component.

Ethernet Configuration

Each board has a unique Ethernet address that is stored in flash memory.

Socket Configuration

Socket 0 of the I/O board is configured as a TCP/IP server set to respond to remote host commands. This server listens on local IP xxx.xxx.xxx.xxx on local port 60250. The IP address is configured through an RS232 link or changed via a TCP/IP command. The default IP is set to 192.168.1.1. A user switch will reset the default IP if switched to the on position during a reset or power on sequence.

Socket 1 of the I/O board is configured as a UDP client that sends packets to the “remote host” IP at port 59750. Socket 1 communicates the current status of the inputs and other parameters when enabled with the “Start UDP packets” command as described below. The remote host IP is automatically obtained from the remote host that issues this command.

Command Syntax

Commands are “put” to Socket 0 of the I/O Expansion board. The command syntax is short and precise in order to minimize the “byte fetches” by the I/O board. This will enable quick turn around times for cmd-response.

- All commands start with a command byte specified in the table below.
- Subsequent bytes are command specific.

Command	Command byte value
Start UDP packets	0x01
Stop UDP packets	0x02
Set Analog Output	0x03
Set Individual Digital Outputs	0x04
Set All Digital Outputs	0x05
Set All Outputs	0x06
Format Digital Inputs	0x07 and up

Start and Stop Sending UDP Status Packets

As stated above, Socket 1 is used to communicate the status of inputs in UDP packets. The UDP packet format is described below. The commands for starting and stopping UDP packet sending are as follows.

Start UDP packets

The “Start UDP packets” is a two-byte command, which consists of the command byte (0x01) followed by a byte value that specifies the interval between UDP packets in increments of 0.1 seconds.

Example: 0x01 0x0a



Networked IO Board

This example starts the UDP packets transmitting at one-second intervals. The IP address to which UDP packets are sent is the same address from which the command is sent.

Stop UDP packets

The “Stop UDP packets” command is a single byte command, 0x02

Setting the State of Outputs

The following commands are used to set the state of the analog and digital outputs.

Set Analog Output

The “Set Analog Output” command is a three-byte command which consists of the command byte (0x03) followed by a two-byte, little Endian, value, which specifies the value of the output DAC.

Example: 0x03 0x3b 0x62

This example sets the analog output DAC to the value of 0x623b.

Set Individual Digital Outputs

The “Set Individual Digital Output” command is a two-byte command, which consists of the command byte (0x04) followed by a byte that specifies both the output index as well as output state. The high-order nibble (4 bits) of the “index/state” byte specifies the digital input index by setting an “on” state (1) in the corresponding upper nibble bit position. The low-order nibble specifies the state, on (1) or off (0).

Example: 0x04 0x21

This example sets digital input 1 on.

Set All Digital Outputs

All of the digital outputs can be set at once with one command. The “Set All Digital Outputs” command is a two-byte command, which consists of the command byte (0x05) followed by a byte that specifies all of the digital outputs. The three low-order bits of the “value” byte specify the state of the outputs. The bit position of each directly corresponds to the output index of each output.

Example: 0x05 0x05

This example sets digital inputs 0 and 2 on, while setting digital input 1 off.

Set All Outputs

All outputs, analog and digital, can be set with one four-byte command. The “Set All Outputs” command consists of the command byte (0x06) followed by the two-byte, little Endian, value for the analog DAC value, and finally by a single byte that specifies the digital output states in the same format as above in the “Set All Digital Outputs” command.

Example: 0x06 0x4a 0x5c 0x04

This example sets the analog output DAC value to 0x5c4a, the digital output 2 on, and digital outputs 0 and 1 off.



Networked IO Board

Monitoring Analog and Digital Signals

UDP Packet Format

The UDP packet format representing the current status of the I/O board inputs is as follows:

Data	Number of Bytes
Version	1
Analog input 0	2
Analog input 1	2
Analog input 2	2
Analog input 3	2
Digital input 0	1
Digital input 1	1
Digital input 2	1
Digital input 3	1
Chip Temperature	2
Board Temperature	2
Chip Vcc	2
Digital input 0 frequency or period value	2
Digital input 1 frequency or period value	2
Digital input 2 frequency or period value	2
Digital input 3 frequency or period value	2

The byte order for the analog values is little Endean. The two-byte analog values are the 16-bit ADC values (actual resolution of the ADC is 12-bits so the upper 4 bits should be zero). The values of frequency and period are also little Endean. The digital input single bytes are binary 0 or 1.

Unless otherwise specified, all analog channel data sent to the remote host is in raw format. Raw format means that the 2 byte data is a binary count from 0 – 4095 that represents either a 0 – 11 volt input or a 0 – 44.2 mA input depending upon the setting of mode switch SW3. Switch an AI channel to the current input mode by switching the corresponding SW3 switch position to the ON position.

For example if data sent back from the board is: 100 1 223 2 then the converted voltage value would be $((1 * 256 + 100)/4096) * 11 = (356/4096) * 11 = 0.956$ volts for channel AI0. Similarly, channel AI1 is $(735/4096) * 11 = 1.974$ volts. If the channel is configured as a current input value, then instead of multiplying by 11 volts, multiply by 0.0442 amperes.



Networked IO Board

On-Chip Temperature Data

On-chip temperature output voltage measurement is referenced to the on-chip voltage reference which is approximately 2.2 volts so when calculating this temperature, use 2.2 volts as the multiplying factor. The formula for temperature in degrees C is: $T_c = (V_{\text{meas}} - 0.78)(416)$.

For example, if the on-chip temperature bytes are ... 32 7 ... then the voltage value is $((7 * 256) + 32)/4096 * 2.2 = 0.98$ volts, rounded. So, $T_c = (0.98 - 0.78)(416) = 83$ degrees C, rounded.

On-Board Temperature Data

On-board temperature output voltage measurement is also referenced to the on-chip voltage reference of 2.2 volts. Output voltage of the temperature sensor is 480 mV at 0 degrees C. The slope of the output is 15.1 mV/degree C. So, $T_c = (V - 0.480)/0.0151$.