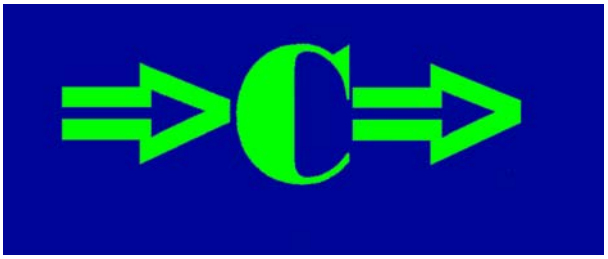
CyberSense



CyQ®701

OPERATORS MANUAL

Version 9.1

SUPPORT
support@cyq.com

CyberSense Inc.
1330 River Road Nicholasville, KY 40356-9649
www.cyq.com
800-942 9327

TABLE OF CONTENTS

INSTALLATION

1. Test

A. Plug the wall transformer into a mains (115 USA / 240 EUR) vac outlet. Plug the barrel connector into the back of the 701. While viewing the front panel LED's turn on the power switch (red handle, up is on). The blue power light should come on followed by a one-half second flash from each of the eight green LED's in order from input channels 1 to 4, followed by output channels 1 to 4. The yellow Tx LED should flash briefly. This confirms that power and microcontroller function are good.

HYPER TERMINAL

(Updates are on Cnet along with many similar programs such as Tera Term)

From windows
1). Click on programs
2). Click on Accessories
3). Select hyper terminal if available. If it does not appear on the list of accessory programs, then go to: How to install Hyper terminal.

HOW TO SET UP WINDOWS HYPER TERMINAL FOR A COM PORT

Most of the settings correspond to the defaults for a PC.

1) In hyper terminal create a "**new connection**" this can be done by clicking the **Hypertrm.exe** icon or via the menu within HyperTerminal "**File:NewConnection**".

2) Type in a name for the connection (for example, cyq).

3) Under "**connect using**" select "**Direct to Com 2**" if your available com port is com 2 otherwise select the appropriate com port. (Modems are relentlessly selfish about interrupts. So if a modem is present, then avoid the modem port and it's cousin. That is, if the modem is on com 1 then avoid both com 1 and 3. Com 1 and 3 share an interrupt while com 2 and 4 share a different interrupt; in this case avoid 2 and 4.

4) Set "**Bits per second**" to 9600.

5) Set "**Data Bits**" to 8.

6) Set "**Stop Bits**" to 1.

7) Set "**Flow control**" to None.

8) Click OK - the hyper terminal screen should appear.

9). Select **properties** from the menu bar.

8) Click on "**settings**", use default: **autodetect** and **back scroll 500 lines**.

9) For **“ASCII setup**” use
a). Sending - check: “**echo typed characters locally**” box. This lets you see the commands you type in. On some versions of windows this may not work. A “ck;” command to the CyQ module will enable keystroke echo (received characters will be echoed) by the module.

b). Receiving - check: “**append line feeds**” and “**wrap long lines**”. The emits a carriage return (CR), but no line feed. This keeps data from being overwritten on the screen.

10) Under "**File**" menu select "**Save As**" and save the configuration file to a convenient folder.

You can create a shortcut to the configuration file and put it on the desktop, this makes life a bit simpler. The configuration files have a naming convention of *.ht where star is the name you chose when setting up the new connection. For example, “cyq.ht”.

1. Connect a serial cable from the DB-9 on the back of the 701 to the serial COM port that you intend to use.

2. Apply power: the word “CyQ701” will be sent by the 701. That is why the yellow Tx LED blinks on power up.

Notes:
1. To change the baud rate DURING a session you must (after going through properties to change the rate) click the ‘call’ button then ‘disconnect’ and then ‘connect’.
2. The ANSI terminal option will display all ASCII characters including symbols for the unprintable ones.

HOW TO INSTALL HYPER TERMINAL

1). Click on **Control panel**
2). Click on **My Computer**
3). Click on **Install/Remove Programs**
4). Click on **Windows settings**
5). Click on **Communications**
6). Select **Hyper terminal** and click **apply** or OK at the bottom. You will be prompted for your Windows CDROM.
7). Done and out’ta there.

CyQ 701 COMMANDS

Introduction

Logic conventions for a dry contact interface:
To sense a switch closure at the input it is intuitive that the contacts CLOSED, state is logic TRUE (binary value 1). Therefore, it is intuitive to also define output logic TRUE as the output as relay contacts CLOSED as logic TRUE (binary 1). Else the LED's on the panel make no sense because the relay lights would be off and the input lights would be on when the relays contacts are open and the inputs contacts are open. However, note that from a voltage perspective the input logic is now inverted from the usual convention where high voltage is logic 1 and low voltage is logic 0.

A bit has a value of 1 or 0 corresponding to logical true or false states. The true or false states represent a contact condition in the 701, either closed (1 or true) or open (0 or false). In CyQ701 commands where a logical value, L, is needed true may be represented by either T or t or 1, your choice; false may be represented by F, f, or 0. The default is TRUE, that is, sending cr; is the same as sending crT;

Commands

Commands may be either upper or lower case characters. All commands **MUST end** with either a semi-colon '**;**' or \r (the ENTER key). The command is buffered, but not processed by the 701 until a ';' or '\r' (ENTER key) is received. The backspace key may be used for corrections.

For brevity we will use the following symbols for values in commands:
L is a logical value where TRUE is T or t or 1, and FALSE is F or f or 0.
X is a hexadecimal number.
N is an integer number.

Command Summary

w write outputs
r read inputs
s stop
g go
? status (command summary and current values).

ckL; configure keystroke (received character) echo on or off.
coft; configure format text
cofx; configure format hexadecimal.
cofr; configure format remote relay (command).

crL; configure read on change either on or off.
crcL; configure read on close on or off for all channels.
croL; configure read on open on or off for all channels.
cro#NL; configure channel N (1-4) read on contact open on or off.
crc#NL; configure channel N read on contact close on or off.
crsL; configure synchronized / asynchronized read on change modes.
crmX; configure lookup table read mask to value X.
ctts; or cttl; configure tick time interval, s=short (0.1 ms), l=long (1 ms).
ctdN; configure debounce time in ticks (0-250).
ctwN; configure wait time in ticks (0-250).

cwL; configure local write from lookup table control on or off.
cwmX; configure lookup table write enable mask to value X.
cwtXX; configure lookup table input entry X to cause output value X.

cq@X; configure baud rate to value X.
cq?; will cause “cyq701” version to be transmitted.

mss; memory save setup, causes current configuration to be saved.
mls; memory load setup, loads the saved setup configuration.
msd; memory save as default, saves the current configuration as the power up default.
mpd; purges the power up default settings.

NOTE: on power up, while the lights are flashing, pressing the Esc key ONCE will cause our original safe configuration to be loaded. Do NOT hold down the Esc key.

e turn error light off in “es;” error sticky mode. Default is “et;” error transient
$@R (case sensitive) will cause the module to restart (reset).

1. Write output command: w

The permanent defaults at power up are all relay channels are set to FALSE (binary 0). FALSE is open relay contacts; TRUE (binary 1) is closed relay contacts.

Single channel w commands

"wNL;" Single channel output control examples: The letter w followed by an output channel number between 1 and 4, followed by either true or false. That is wN1; or wN0 where N is a channel number between 1 and 4. wN; defaults to true.

"w1t;" or "w1;" Causes the relay contacts to close for channel 1 (the left first two pins on the lower row of the connector will be shorted; output 1 LED will light).

"w1f;" or "w10;" Causes the relay contacts to open for channel 1; output 1 LED will go off).

"w2T;" or w2; Causes the relay contacts to close for channel 2 (the first two pins on the lower row of the connector will be shorted; output 2 LED will light).

"w2F;" Causes the relay contacts to open for channel 2 (the first two pins on the lower row of the connector will be open or infinite resistance; output 2 LED will go off).

Byte wide output examples:

1). Text format

"wLLLL;" where L is logic true or false (T,t,1 or F,f,0)

2). Hexadecimal encoded integer format.

"wX;" where X is a hexadecimal character. In hexadecimal format the hexadecimal character (8 bit) is translated to the 4 bit binary representation of the hexadecimal digit. The channels are in bit order: bit 0 corresponds to channel 1, bit 1 corresponds to channel 2 ..... If bit 0 is 1 (True), then relay 1 closes, else if bit 0 is 0 (False) then relay 1 opens.

Note that in text format the order corresponds to the front panel 1234. It is customary in English to read increasing from left to right. In hex format a string of binary digits (ones and zeros, 0 is off and 1 on), such as 1101, are numbered from 0 for the least significant bit ascending to the left. For 1101: Bit 0 is 1, bit 1 is 0, bit 2 is 1, and bit 4 is 1. In binary the bits are numbered in order of increasing significance. The right most bit is the least significant (bit 0) while the left most (bit 7) is the most significant. This is the same as money; in price $427 the least significant digit is the 7, and the most significant digit is 4. The channels are bit mapped accordingly: channel 1 corresponds to bit 0, channel 2 is bit 1, channel 3 is bit 2, and channel 4 is bit 3.

Examples, hexadecimal:

"w7;" Relays 1,2 and 3 are on with contacts closed ; relay 4 is open. The binary representation of hexadecimal digit 7 is 0111

"wA;" Hexadecimal digit A is binary 1010, therefore, channels (relays) 1 and 3 are open, two and four are closed.

The following table may be helpful:

HEX	binary	Channel 4	Channel 3	Channel 2	Channel 1
0	0000	open	open	open	open
1	0001	open	open	open	closed
2	0010	open	open	closed	open
3	0011	open	open	closed	closed
4	0100	open	closed	open	open
5	0101	open	closed	open	closed
6	0110	open	closed	closed	open
7	0111	open	closed	closed	closed
8	1000	closed	open	open	open
9	1001	closed	open	open	closed
A	1010	closed	open	closed	open
B	1011	closed	open	closed	closed
C	1100	closed	closed	open	open
D	1101	closed	closed	open	closed
E	1110	closed	closed	closed	open
F	1111	closed	closed	closed	closed

2. Read Input Channel Command: r

At power up, with nothing connected, all input channels are set to FALSE (binary 0) by

active pullups (100K to 5 vdc). If the input pin is shorted to ground by a switch contact then the channel is TRUE(binary 1). The purpose of the inputs is to DETECT a contact closure by a switch attached to the input connector.

Read Examples:

a. Single channel input detection.

"**r1;**" If Pin 2 on input channel 1 (the leftmost two pins on the upper row of the connector) is low, then the string "1f" will be transmitted in text format. In hex format a "10" will be sent. If input 1 (PIN 2) is high, then an r1; command causes the string "11" to be transmitted in hex format, and "1T" in text format. Channel 1 panel LED on the input (top) row will be ON if input 1 is TRUE meaning switch closed, input low.

"**r4;**" If input channel 4 (the last two pins on the upper row) is low, then the text format string "4f" will be transmitted. In hex format a "10" will be sent. If input 4 (PIN 8) is high (PIN 7), then the either the hex string "41" or the text string "4T" is transmitted. Channel 4 LED on the input (top) row will be on if input 4 is TRUE.

The channel value is sent because the serial inputs and outputs are asynchronous The lag between computer command and computer response to an input from the 701 is unknown. Without the channel number it is quite easy to lose track of the relationship between outgoing commands and incoming data.

b. Byte wide read

"r;"

Text format: 4 characters are transmitted. The characters are either T or f corresponding to either closed or open input contacts. Input channel 1 is sent first followed 2, 3, and 4.
Therefore TfTf means that input 1 contacts are closed, input 2 open, input 3 closed, input 4 open.

Hex format: a byte is transmitted with bits 4 through 7 (the high nibble) set to 0; the low bits 0 through 3 contain the logic state of **input** channels 1 through 4. The resulting byte is an integer number that the computer converts to hexadecimal character format for transmission. Thus, the reception of 0b means that input channel 1 is closed (true), 2 is closed, 3 is open, and 4 is closed.

Configure

Output Format

"coft;" Configure output format text. The single channel write case is not affected; in the all channel case the write command must use be of the form "wLLLL;" where L is T,F or t,f or 1,0. Single channel read output will always use only T or F.

"cofx;" Configure output format hexadecimal encoded integer. The single channel case will not be affected. The all channel case must be of the form "wX; where X is is a hexadecimal number between 0 and F. Single channel read output will always use only 1 or 0.

"cofr;" Relay format, the read sends a command on the serial output port that will cause a remote 701 to set it's relays to the first 701's input. Used in stand alone. All other output messages are suppressed.

Tick Time

"ctts;" and "cttl;" The 701 runs in a loop checking inputs and setting outputs. The loop time (tick time) is set by this command. ctts = 's'hort and sets a tick time of 0.1 ms while cttl; = 'l'ong and sets a 1 ms tick time. The tick time unit is used by both debounce ("ctdN;")and sync wait ("ctwN;")commands.

Debounce Time

"ctdN;" Configure debounce value. N is an integer number from 0 to 250. It is the time in tenth of a millisecond that the input is allowed to settle after a change is detected. When all inputs are stable the current contact closures are read and transmitted. A switch contact bounces on closure. In read on change mode this could cause several hundred reads to be done and transmitted.

Wait Time

"ctsN;" This sits the time that the 701 waits with no contacts bouncing before declaring that a valid read on change has occurred. This reduces output traffic in async mode by combining closely spaced changes. In sync mode this compensates for inputs with short debounce times, but longer lags between individual inputs.

Synchronized Read

"crsL;" Sits synchronized read mode. In this mode all inputs must have stopped bouncing for a time equal to the sync wait time ("ctsN;") before a valid read is declared. In async mode and read on change the change is declared and the read sent as each input stops bouncing. Sync mode is useful when the inputs represent a 4 bit logic state, and are expected to all change at the same time. In practice this does not happen

because some input contacts either lag or bounce more than the others. Synchronized modes allows these differences to be corrected. Note that using sync mode may cause asynchronous changes to be missed.

Read on Change

"crL;" If L is either ';' or true (T,t,1) then read on change is activated. A change on any input will cause an all channel read. The initial change will be detected; after a delay equal to the debounce time the contacts will be read again. If the change is still present, then the read will be transmitted. If L is false (F,f,0) then read on change will be deactivated.

"cr(contacts)#NL; where contacts is either 'o' (letter oh) for open, or 'c' for closed. The channel number is 'N'.
Activates read on change for channel N for closed to open contacts, and open to closed contacts. Default is all channels and both changes.

cro#1t; Read when channel 1 contacts open, go from closed to open.
cro#1f; Disable read when channel one opens.
crc#2t; Enable read when channel two contacts close, go from open to close.
crc#21; Enable read when channel two contacts close, go from open to close.

To completely disable read on change for channel 1 enter:
'cro#1f;' and 'crc#1f;' to turn off both edges.
To Enable all channels for read on open enter:
cro; or croL; where L is t,T, or 1.
To Disable all channels for read on closed enter:
'crcf;' or 'crc0;'

When in text format output and read on change, the read will produce 4 logical values with True (T) indicating the channel(s) that changed, followed by a comma; this will be followed by the current logical values of the 4 channels.

In hex format the first hex digit will be a bit map of the channel(s) that changed and the second digit will be the current values of the 4 channels.

Local Lookup Table Write Control

"cwL;" Enable local write control.

The outputs are determined from the inputs by a look up table stored in memory. The look up table is programmable, and is saved by the memory save command. In this mode the 701 will control the outputs without the need for serial communication with a computer; it becomes a stand alone device. Handy way to create SPDT, DPDT, SP4T

switches and other logic.

"cw;" : enables local control.
"cwf;" : disables local control.

Set Read and Write Masks for look up table control
"crmX;" sets the read mask (a logical AND mask) to hexadecimal X (default is 0xF, 1111, use all inputs). For example:

"crmE;" corresponds to 1110 and channel 1 input is forced to zero,
"crm7;" corresponds to 0111 and channel 4 input is forced to zero.
This command reduces the number of entries needed in the look up table.

"cwmX;" sets the write enable mask. A zero bit blocks an individual output from being written to. These outputs may then be used for other purposes. For example:

"cwm7;" will block output 1 from being written to from the look up table.

Load Lookup Table
"cwtXX;" write table, loads lookup table location first X with second X. The first hexadecimal character X is the input state. The second X is the output state that is produced. For example:

"cwt9F" will cause all output relays to close when the input switches are closed on channels 3 and 4 and open on channels 1 and 2. In binary 1100 -> 1111.

Utility

Status

"?;" Causes a summary of commands and current values to be sent to the terminal.

Memory

"mss;" : Save setup. The present status is written to nonvolatile memory. This includes configure commands and baud rate. Useful for testing before saving as default. Allows multiple configuration settings to be switched on the fly with one command.

"mls; Loads the saved setup.
"msd;" Save default. Saves the current configuration as the power up default.
"mpd;" Purges the saved power up default. The module will boot as originally shipped.

Error

"**e;**" turns off the error LED. "es;" command causes errors to be sticky, and the error light to remain set until it is turned off by the "e;" command. The default sitting is "et;" errors transient mode; the error light is trned off on the next valid command.

Baud rate

"**cq@X**;" where X ranges from 0 to A.
This sets the transmit and receive baud rates for the serial output. **The DEFAULT rate at power up is 9600**. There is always 1 start bit, 8 data bits, and 1 stop bit; no parity; no handshaking. The computer baud rate must be changed to match **after** this command is executed (in hyper terminal you must click properties, and then configure. The baud rate will not change until you click disconnect, and then connect on the main toolbar).

The values are: 1=2400, 2=4800, 3=9600, 4=14,400, 5=19,200, 6=28,800, 7=38,400, 8=57,600, 9=115,200, A=230,400.

Restart

"**$@R**" will cause the module to do a power up reset. This command is CASE Sensitive, $@r will not work.
Note: Pressing the esc key during a restart will cause the original default values to be used instead of a saved default.

Version

"cq?;" Causes "cyqVERSION" to be transmitted.

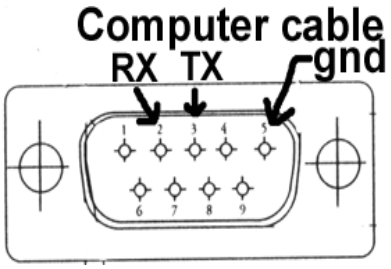
SERIAL HARDWARE INTERFACE

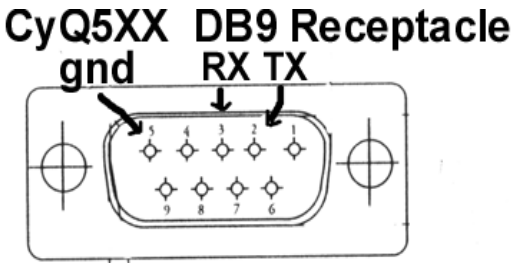
RS232C

This is a **standard serial cable** for DB9 interconnection. It is NOT a null modem cable. The crossover from transmit to receive is done at the receptacle on the A/D. The required condition is that transmit from the computer connects with receive at the peripheral (A/D) while transmit from the A/D connects to receive at the computer.

CyQ Host.....CyQXXX Module

TX PI 3.....Pin 3 RX
RX Pin 2.....Pin 2 TX
GND Pin 5.....Pin 5 GND



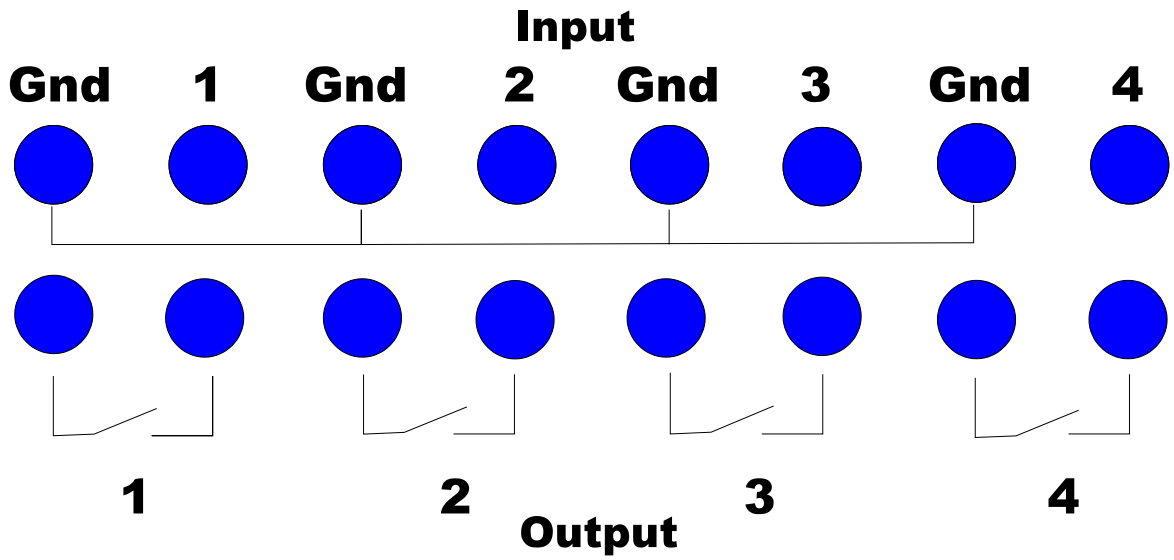


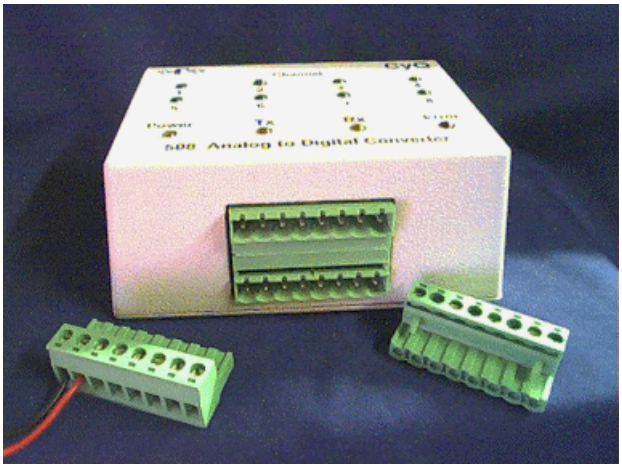
The interface is implemented with a MAX232A integrated circuit from Maxim Inc. It has been tested to 115.2 kb on a 6 foot cable. (9600 baud at 24 feet)

There is no hardware (washers/nuts) inside the case for the jackscrews on the DB9 receptacle. If the post comes loose, then simply screw it on out; remove it from the connector, and reinstall it. Do not over tighten.

INPUTS/OUTPUTS

SIGNAL





The input signal source needs to be able to sink 1.2 ma to force the input to 0 (due to the 10K pull up resistor to 12 vdc). Any dc voltage in excess of 4.0 vdc will produce a logic high (1 or true). A logic low input voltage must be less than 1.0 vdc. (Input is to a Schmidt trigger.) The input leakage current to the microcontroller input pin is typically 1 ua; this will produce a 10 mv drop across the 10K input protection resistor which is negligible.

Output relay contacts

The relay contacts are rated at 0.5 amps and 200 vdc for switching, and 1.25 amps for carrying current. The contact resistance is 0.200 ohms. Insulation resistance is 10 gigohms (10^{10}). Switching time is 1 ms for on and 0.5 ms for off.

POWER

A 7.5 to 12 vdc unregulated wall transformer with 100 ma or more current capability will work. This is a 2.1 mm barrel connector. Center positive. The red handle power switch is directly above. Up is on and down is off. This power input is diode protected against reverse polarity. It is varistor protected above 18 volts. Further protection is provided by a 500 ma self resettable fuse in series with varistor.

Error Codes

1. Receive buffer overrun. Too many instructions were sent too fast. Generally seen when a computer is issuing instructions. Insert a wait or delay instruction in your program.

2. Transmit buffer overrun. Too many messages being sent from the module. May be caused by switch changes occurring too rapidly when in read on change mode.

3. Command buffer overflow. The command exceeds 10 characters in length. May be due to a stuck key on the keyboard or failing to use ';' or cr to end commands.

4. Read command buffer overflow. Commands may arrive while another rad is waiting on debounce to complete. These are buffered. Too many reads were received during debounce or debounce time is set too long.

Actions by CyQXXX due to overflows: An error message is sent. The buffer is flushed and all pointers are set to zero.

10. Unrecognized top level ecommand. The first letter of the command is invalid.
11. Expecting a channel number.
12. Write command length is wrong.
13. A logical value is expected T,F or t,f or 1,0 or ;
14. Expecting an 'r' or 'w' in look up table write command.
15. Debounce number is too big.
16. Configure command is unrecognizable.
17. In read on change a '#' or t,f or 1,0 was expected.
18. Configure format hasn't heard of this format.
19. A hexadecimal character was expected.
20. Baud rate code is invalid.

31. Memory write command, what to write is wrong.
32. Attempt to load a saved setup when no setup has been saved.
33. Memory command is unknown; what to do, the second character.

701 SPECIFICATIONS

- Outputs: 4 sets of normally open relay contacts.
- Relay contacts are rated 200 vdc, 0.5 amp switching, and 1.25 amp carrying.
- Insulation resistance is 10,000,000,000 ohms.
- Contact closure time 1 ms, open time 0.5 ms.
- Inputs: 4 pin pairs for sensing of contact closure.
- Input pullups 100 K ohms to 5.0 vdc.
- Input protective 10K ohms to junction of 5.0 and Gnd connected protective diodes at IC Pin.
- Input pin current: 500 ua when shorted to ground; 2 ma at 25 vdc.
- Input logic TRUE (1): a dc voltage greater than 2.5.
- Maximum safe input voltage for electronics greater than 35 vdc
- Input logic FALSE (0): a dc voltage less than 0.75 volts.
- Serial communications RS232, connector: standard DB9 female. TX, RX, Gnd.
- Contact Interface connector: Phoenix Contact MDSTB2,5/G1-5.08 (1762431); Two removable wire connect plugs. Phoenix Contact MSTB2.5/8-ST-5.08
- Pins on contact interface connector UL/CSA rated Nominal 300 v/10 amp, surge 4 kv.
- **Power supply:** 7.5 to 12 vdc unregulated wall transformer, 300 ma, 2.1 mm barrel connector. **Dropout is 7 volts.**
- **Power input:** reverse polarity diode protected, 500 ma self resetting fuse, surge protected above 18 vdc to 600 amps.
- **Power supply current : 76 ma**
- **Size**: width 4 in.(10 cm), depth 3 in.(7.6 cm), height 2.0 in. (5 cm).
- **Weight**: 7.7 oz (218 g)**.**
- **Material**: ABS plastic case, textured gray, 4.49 X 3.25 inches (11.2X8.3 cm).

LEGAL STUFF

Limited Warranty

CyberSense warrants to the original purchaser or end-user complete satisfaction for 90 days - money back or exchange; we pay shipping within USA. This product is warranted to be free from defects in material or workmanship for a period of five years. During this period the product will be replaced or repaired without charge for either parts or labor.

This warranty is voided if the product is modified, misused, subjected to abnormal environments (examples: submergence in water or harsh chemicals, ovens, etc) or conditions (vaporized, parts, circuit board, etc due to application of mains to a signal input, etc). This warranty is void if the case is opened for any reason without contacting us. Specifically - if you attempt repairs then, at our discretion, you abandon this warranty and all benefits of this warranty. We will gladly assist with trouble shooting your setup by phone or email.

Fragile sensors such as force, low pressure, and 30 gauge thermocouples are excluded from this warranty, and are sold without warranty. You must test immediately upon receipt. We test immediately prior to shipping. You will know it is a fragile sensor because it will be packed with a picture of a broken wine glass and a trash can.

Purchase price refund including shipping (exception on shipping: international orders) for 90 days (30 days international) or repair and replacement for a period of 5 years (3 year international) are the ONLY REMEDY of the purchaser. This warranty is in lieu of all other warranties either express or implied; specifically, any warranties of implied merchantability or fitness for a particular use or purpose. CyberSense shall in no way be liable for indirect or consequential damages of ANY kind or nature.

Some jurisdictions do not allow exclusion or limitation of consequential or incidental damages, or how long implied warranties last; therefore, the above limitations may not apply to you. This warranty gives you specific legal rights. In some states you may have other rights.

Use in life support systems: NO!

Trademarks

CyQ is a registered trademark owned by CyberSense, Inc.; CyberSense, Inc. is a trade name. Any other trademarks, trade names, service marks, or service names owned or registered by any other company and used in this manual are the property of their respective companies.