OCEAN CONTROLS

8 Relays with 10 A, 250 VAC rated contacts

4 Optically isolated digital inputs up to 30 VDC

Connections via pluggable screw terminals

3 Analog inputs, 0 to 5 V or 0 to 20 mA jumper selectable

Power indicator and serial transmit and receive LEDs

All enclosed in professional looking plastic case

Arduino Compatible – can be reprogramed with the Arduino IDE

Accepts Arduino shields (Ethernet / XBEE)

FTDI232 based USB virtual serial interface

RS-485 serial interface

Can be used as a USB to RS-485 converter

Multiple units can be connected in an RS-485 bus

The KTA-223 (12 VDC powered) and KTB-223 (24 VDC powered) are USB or RS-485 controlled IO modules for interfacing PCs to real world applications, such as controlling lights and sprinkler systems, reading sensors and monitoring switches and other digital signals.

The controller ships programmed as an Arduino with a custom sketch loaded that listens for commands on the USB or RS-485 port and responds by switching relays or returning the state of inputs. The sketch is available on the Ocean Controls website as an example of Arduino programming for the controller.

The custom sketch speaks a simple ASCII protocol that allows control from Windows/Mac/Linux using either USB virtual serial drivers or RS-485. The controller acts as a USB to RS-485 converter, so multiple devices can be connected to one RS-485 bus, allowing control of many devices from one USB port.
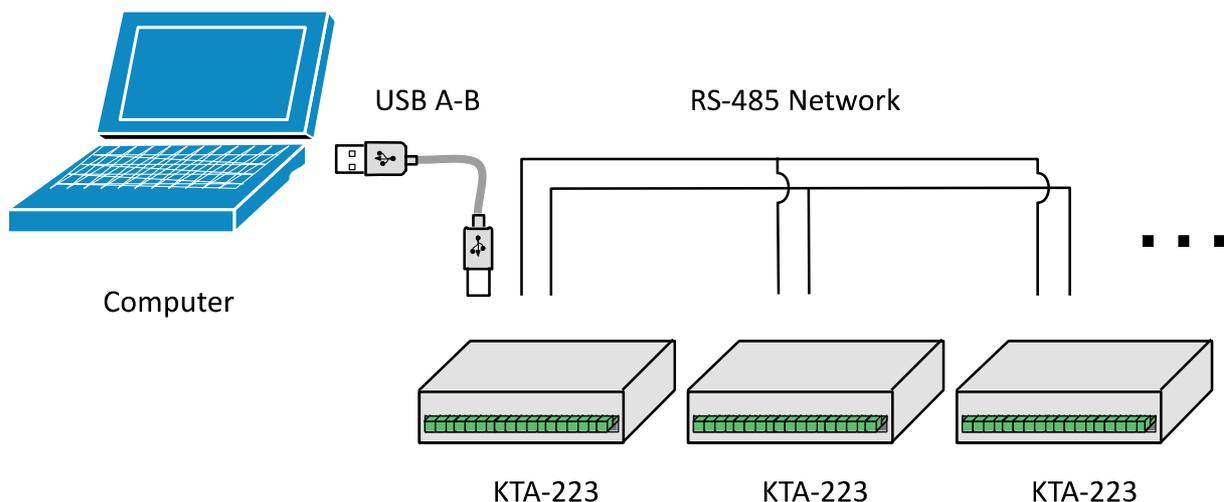

Figure 1 - Connecting multiple controllers with a RS-485 network

The controller is based on the hardware of the Arduino physical computing controllers. It can be programmed as a stand-alone controller using the free, open source Arduino environment. Internally,

the controller is "shield-compatible", allowing the use of many extension boards designed for the Arduino Deumilanove or Uno.

The controller comes in two versions: a 12 VDC KTA-223 and a 24 VDC KTB-223. The only difference between the two versions is the coil voltage of the relays loaded to the board.

The relays are rated to 10 A and 250 VAC. At the maximum current the PCB tracks would rise approximately 80º so the controller can only handle the full rated relay current for short periods of time.

The analog inputs are naturally 0 to 5 VDC. Each input has a precision 250 Ω resistor that can be selected by moving jumpers inside the unit. Installing these jumper shunts allow the measurement of 0 to 20 mA (including 4 to 20 mA) signals. The analog inputs have 4.7 kΩ inline resistors which will protect the microcontroller from damage for accidental input voltages up to approximately 30 V.

Connections:

Label	Description
+	Opto-Isolated Input Positive
-	Opto-Isolated Input Negative
ANx	Analog Input x
COM	Common Connection (Ground)
D+	RS485 Data+
D-	RS485 Data-

Label	Description
5VO	5V Output for Sensors
V+	Power Supply Positive Input
COM	Common Connection (Ground)
USB	USB B-type connection to PC
NO	Relay Normally Open Contact
C	Relay Common Contact
NC	Relay Normally Closed Contact

Table 1 - Connections

Specifications:

Power Supply V+ and COM:

KTA-223: 9 to 16 VDC (12 V Nominal) ~200 mA + External 5 V drain

KTB-223: 18 to 32 VDC (24 V Nominal) ~200 mA + External 5 V drain

Analog Input ANx:

0 to 5 V: ~500 kΩ effective resistance with no jumper installed

0 to 20 mA: 250 Ω effective resistance with jumper installed

Opto-Isolated Input: 0 to 30 V, ~1 kΩ effective resistance. ≤2.5 VDC activation threshold.

Relay Outputs: SPDT relays rated to 10 A (resistive). 250 VAC / 30 VDC

5 V Auxiliary Supply: 100 mA

Jumper Settings:

The Analog inputs of the KTx-223 can be set for 0 to 5 VDC or 0 to 20 mA operation. Opening the case and inserting jumper shunts in the positions J1, J2 or J3 will set the associated analog inputs to 0 to 20 mA operation. Removing the shunts will set the analog inputs to 0 to 5 VDC operation.

When the jumper labeled AUTO is installed the board will reset each time a serial connection is made to the USB serial port. This should only be installed when reprogramming via the Arduino Environment, or the device will reset each time a serial connection is made to the unit.

2

Using the Controller:

The controller requires 12 VDC or 24 VDC to operate, connected to V+ and COM. This can come from a plug-pack, bench top power supply or battery. The controller has screw terminals for the connection of power. Plug-pack power supplies often come with a plug on the end of the lead. The plug can be cut off and bare wires exposed for the screw terminals on the controller.

Connect the power supply positive connects to the V+ terminal and negative to the COM terminal next to it. The POWER LED should light. A diode protects the controller and prevents it from operating with power connected in reverse polarity. If the LED does not light, ensure your supply is delivering sufficient voltage and is connected correctly.

Connect the controller to a computer using a USB A male to USB B male cable. When the power is turned on your computer may prompt you to install drivers. The drivers required are the FTDI Virtual Serial (COM) Port Drivers the latest versions for all systems are available from http://www.ftdichip.com/Drivers/VCP.htm
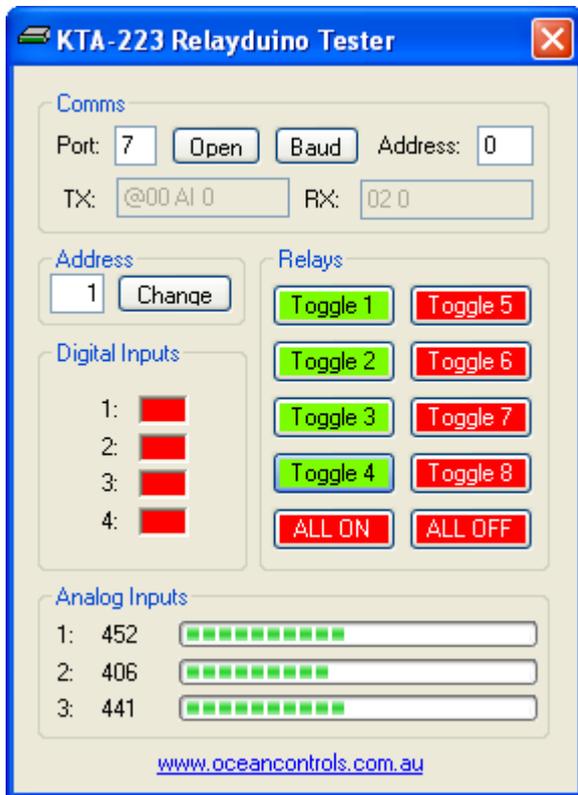
Test Utility:

The main window of the Windows test utility is shown here.

If the Address of the unit you wish to control is known put it in the "Address" text box, if not, use 0 for the address and any unit will respond.

Enter the COM Port number in the "Port" text box, if this is not known it can be found in the device manager under ports. The quickest way to run device manager is by clicking Start->Run and then typing "devmgmt.msc".

Once the device is communicating, Relays can be turned on or off by clicking the buttons in the Relays group and the status of the Digital and Analog Inputs are shown in their relevant groups.

The source code for this program is available from Ocean Controls and is written in Visual Basic Express 2008 which is available free from Microsoft.

Ocean Controls can also supply a similar example program with source code for Visual Basic 6.

Communicating with the Controller:

The Address and Baud Rate of the unit can be set and are stored in the controller's memory. By default the controller is listening for serial data at 9600 baud, and has address 00. The controller will always use 1 Stop Bit, 8 Data Bits and No Parity.

The commands the controller uses are in the form

@AA CC X<CR>

The @ symbol is used to define the start of a command.

AA is the address of the unit from 00 to 99.

CC is a two letter command used to determine the command type.

X is a one or more characters which determines the parameter for the command.

<CR> is the carriage return character. This is ASCII character 13, or 0x0d.

Each time a valid command is received the unit will respond with #AA followed by any values that are requested from the unit.

Note that 00 is the Wildcard Address. All controllers respond to address 00 regardless of what their address has been set to. If a command has 00 as the address, all devices on the bus will respond as if they have been individually addressed.

Command Set:

Letter	Command	Parameters
ON	Turn Relay On	1-8: Turn Relay 1-8 On Individually 0: Turn All Relays On at Once
OF	Turn Relay Off	1-8: Turn Relay 1-8 Off Individually 0: Turn All Relays Off at Once
WR	Write to all Relays at once	The parameter is a number which determines which of the relays should be turned on or off.
TR	Turn Relay on for Set Time	2 parameters, First is 1-8: Relay Number, Second 001-255: Time in tenths of seconds to turn ON for
KA	Keep Alive	0: Turn Off Keep Alive 1-255: Time in Seconds
RS	Status of Relays	1-8: Returns Status of Relays 1-8 Individually 0: Returns Status of All Relays
IS	Status of Inputs	1-4: Returns Status of Inputs 1-4 Individually 0: Returns Status of All Inputs
AI	Read Analog Input	1-3: Read Value of Analog Input 1, 2 or 3 0: Returns Value of All Analog Inputs
SS	System Status	Must be 0, Returns status of all Relays, Analog and Digital Inputs
SA	Set Address	01-99: Sets the Address of the unit in Memory

SB	Set Baud Rate	1-10: Sets the Baud Rate	
		1: 1200 baud	6: 19200 baud
		2: 2400 baud	7: 28800 baud
		3: 4800 baud	8: 38400 baud
		4: 9600 baud (default)	9: 57600 baud
		5: 14400 baud	10: 115200 baud

Table 1 - Command set

ON: Relay On Command

This command is used to turn a single relay on. E.g.: @44 ON 1 will turn relay 1 on for the unit with address 44. It can also be used to turn all the relays on, this occurs when the parameter value is 0.

OF: Relay Off Command

Similar to the on command this command will turn relays off in the same manner. E.g.: @44 OF 1 will turn relay 1 off for the unit with address 44, @44 OF 0 will turn all relays off.

WR: Write Relays Command

The write relays command is used when more than one relay is to be turned on or off at once. The parameter is a decimal number which, in binary, represents the on and off status of the 8 relays. The least significant bit of this value controls relay 1. The most significant bit of the parameter value controls relay 8. A set bit (1) turns the relay on, a cleared bit (0) turns the relay off. Example: To turn relays 1, 2 and 6 on (and others off) the binary value required is 00100011. In decimal this is 35. (2^(1-1) + 2^(2-1) + 2^(6-1) = 35). To issue this to a controller with address 44, the required command is @44 WR 35

TR: Timed Relay Command (Software v1.1 and Later)

The timed relay command is used to turn a single relay on for a set period of time, after which the relay automatically turns off. Two parameters are needed, separated by a space, the first is the relay number (1-8) and the second is the time to turn the relay on for in tenths of seconds (001-255) the second parameter must have 3 digits. E.g.: @44 TR 1 050 will turn relay 1 on for 5 seconds.

KA: Keep Alive Command (Software v1.2 and Later)

This command is used as a watchdog timer. If the parameter is 0 the watchdog is turned off. If the parameters is between 1 and 255 then if that time in seconds has elapsed without the unit receiving another Keep Alive command then all the relays will turn off. This is used for a fail safe to ensure if the host software can no longer communicate with the KTA-223 then the relays turn off.

IS: Input Status Command

This command will return the status of the inputs. If the parameter is between 1 and 4 then the controller will return a 0 or 1 corresponding to that input. E.g.: @44 IS 1 will return #44 1 if the input is on, or #44 0 if the input is off.

If the parameter is 0 then the unit will respond with the status of all the inputs, in similar form as the Write Relays command. E.g.: If inputs 1 and 2 for the unit are on then @44 IS 0 will return #44 3. 3 is 0011 in binary, and each bit represents each input from 4 down to 1.

RS: Relay Status Command:

Much the same as the input status command, this command will return the status of the relays. If the parameter is between 1 and 8 then the unit will return with a 0 or 1 corresponding to that relay. E.g.: @44 RS 1 will return #44 1 if the relay is on, or #44 0 if the relay is off.

If the parameter to this command is 0 then the unit will respond the same way as the input status command, but return the status of the relays.

AI: Analog Input Command

The analog input command will read the status of the analog input defined by the parameter and return it as a value between 0 and 1023. E.g.: @44 AI 1 will return #44 512 if the analog input is reading 50%.

SS: System Status (Software v1.1 and Later)

The system status will requires a parameter of 0 and returns the status of all relays and inputs, 5 values each separated by a space. The first value returned is the same as is returned for an RS 0 command, the second value returned is the same as is returned for an IS 0 command and the last 3 values returned are the same as returned from and AI 0 command.

SA: Set Address

Addresses are valid from 01-99. A unit will only respond if its address in memory is the same as that of the command sent, or if the address of the command sent is 00. The address is saved to non-volatile memory inside the controller, meaning it will be preserved even after power is disconnect from the controller.

SB: Set Baud

Parameters from 1 to 10 are valid, corresponding to values shown in Table 3.

1: 1200 baud	6: 19200 baud
2: 2400 baud	7: 28800 baud
3: 4800 baud	8: 38400 baud
4: 9600 baud (default)	9: 57600 baud
5: 14400 baud	10: 115200 baud

Table 1 - Baud rate selection

The baud rate is saved to non-volatile memory inside the controller, meaning it will be preserved even after power is disconnect from the controller.

Using the Controller as an Arduino:

The KTA-223, as supplied, is an Arduino compatible board with Arduino bootloader and a custom sketch loaded that responds to the serial commands listed above. The source code of this is available from Ocean Controls and can be modified in the Arduino environment to suit your purpose.

The Arduino programming environment can be downloaded for Windows, Mac OS X and Linux from http://www.arduino.cc/

When using the KTA-223 with the Arduino Environment select "Arduino Duemilenove w/ ATmega328" from the "Tools->Board" menu, and install the "AUTO" jumper on the PCB for ease of programming.

The hardware has been designed to accept the Arduino compatible Shields. The cover may not be able to be installed when using larger shields. Some shields may require removal or modification of the back panel to fit over-hanging components (The Libelium XBee shield fits with XBee modules using chip antennae, but SMA antenna connections conflict with the back panel)

The PCB versions before revision 6 do not locate the 6-pin ICSP in the same position as the Arduino Deumillanove. Some shields (including the Arduino Ethernet shield) take 5V power, ground or other signals from the ICSP header instead of from the shield headers. On KTA-223v5 and earlier controllers, jumper wires or connections might be required to use such shields.

For example, the the Libelium XBee shield must be supplied with 5V power by connecting 5V on the shield to K6 Pin 2 and GND on the shield to K6 Pin 6.

Space is provided on the PCB to install the SparkFun Real Time Clock module (SparkFun SKU: BOB-00099). The intention is to allow the controller to operate in stand alone situations that require more timing flexibility than the stock controller can provide. The PCB connects the RTC module SDA to Arduino Digital 12 and SCL to Arduino Digital 13. Installing this unit may prevent proper operation of other modules or shield that rely on these pins (for example, the Ethernet shield cannot be used with the RTC module.)

Solder jumpers J5 and J6 connect 4.7 kΩ pull up resistors to the D12 and D13 lines. These pull ups are required for I2C communication with the RTC module. Solder jumper J7 connects the square wave output of the RTC to D11 to allow use of the RTC output as an interrupt.

The RTC module is not connected to the hardware I2C module of the microcontroller. Communicating with the RTC requires the use of a software I2C library such as SoftI2CMaster by William Greiman.

Hardware revisions 8 and later include a three-way header connection suitable for connecting a DS18S20 or DS18B20 temperature sensor bus. Solder jumper J8 connects a 4.7 kΩ pull up resistor to D10, which is connected to the centre pin of the header.

The RS-485 transceiver is connected in parallel with the FTDI USB to Serial converter and ATMega328 UART pins. This transceiver allows half-duplex serial communication over 2 or 3 wires. The transceiver requires a TX Control signal to enable the transmit or receive line driver. When transmitting, the TX Control line must be asserted (driven high). To receive, the line must be left low.

The FT232RL USB to Serial converter provides a TXEN signal for RS-485 Transceivers. When data is received from the USB port by the FT232RL, it asserts the TX Control line, putting the RS-485 transceiver in Transmit mode. The serial data is then transmitted to the ATMega328 and onto the RS-485 network.

Using the RS-485 transceiver from custom Arduino code requires that your code drive the TX Control line high at the beginning of data transmission and returns it low at the end of the transmission. The TX Control line is connected to Digital 19. The Ocean Controls sketch provides an example of how to do this.

Table 4 shows the mapping of Arduino pins to the inputs and outputs of the controller.

KTA-223 IO	Arduino Pin	AVR Port.Pin	KTA-223 IO	Arduino Pin	AVR Port.Pin
Relay 1	Digital 2	PORTD.2	Analog In 1	Analog 6	ADC6
Relay 2	Digital 3	PORTD.3	Analog In 2	Analog 7	ADC7
Relay 3	Digital 4	PORTD.4	Analog In 3	Analog 0	PORTC.0
Relay 4	Digital 5	PORTD.5	RX Data	Digital 0	PORTD.0
Relay 5	Digital 6	PORTD.6	TX Data	Digital 1	PORTD.1
Relay 6	Digital 7	PORTD.7	RS-485 TX Control	Digital 19 / Analog 5	PORTC.5

Relay 7	Digital 8	PORTB.0		Ethernet Shield	Digital 10	PORTB.2
Relay 8	Digital 9	PORTB.1		Ethernet Shield	Digital 11	PORTB.3
Opto-In 1	Digital 15 / Analog 1	PORTC.1		Ethernet Shield / RTC SDA	Digital 12	PORTB.4
Opto-In 2	Digital 16 / Analog 2	PORTC.2		Ethernet Shield / RTC SCL	Digital 13	PORTB.5
Opto-In 3	Digital 17 / Analog 3	PORTC.3				
Opto-In 4	Digital 18 / Analog 4	PORTC.4				

Table 1 - Arduino Pin Mapping

KTA-223 Wiring Examples: Inputs

The opto-isolated inputs allow for a range of connection possibilities. The figures below show the wiring for a dry-contact switch, NPN and PNP-type sensor.
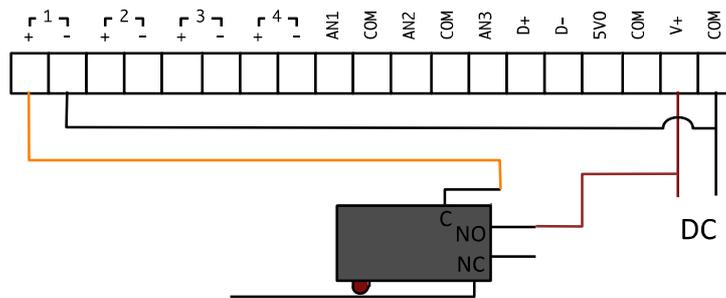
Figure 1 - Wiring a dry contact switch
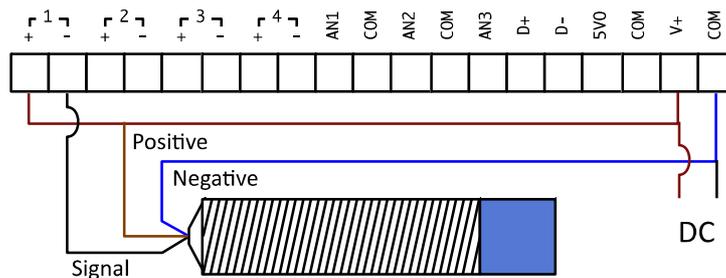
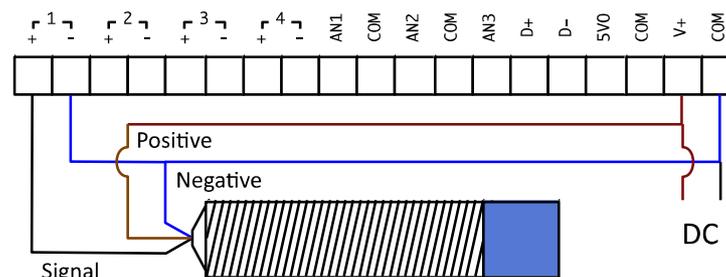Figure 1 - Wiring an NPN-type sensor

Figure 1 - Wiring a PNP-type sensor

Analog inputs can be wired for 0 to 5 V or 0 to 20 mA signals, depending on the position of the input jumper inside the unit. A regulated 5 VDC output is provided for the convenience of wiring analog sensors like potentiometers.
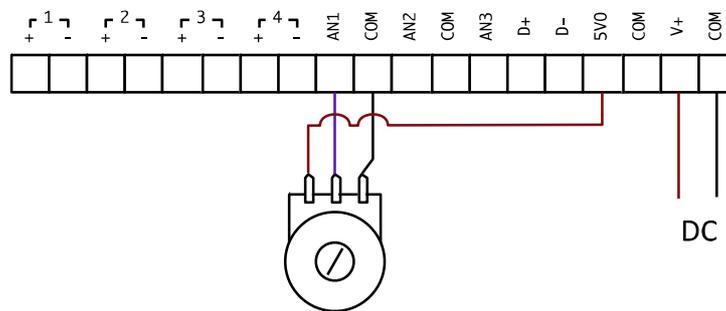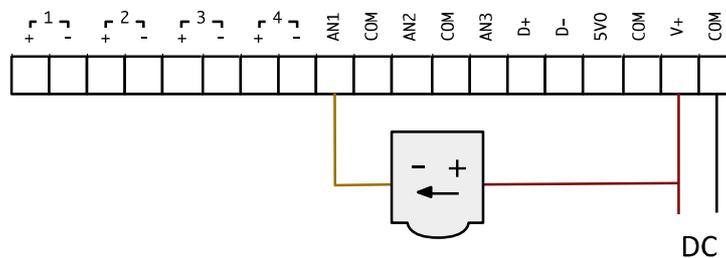
Figure 1 - Wiring a potentiometer

Figure 1 - Wiring a 4 to 20 mA, loop-powered sensor

Wiring Examples: Outputs

The relay outputs on the KTA-223 can be wired to DC or AC loads.
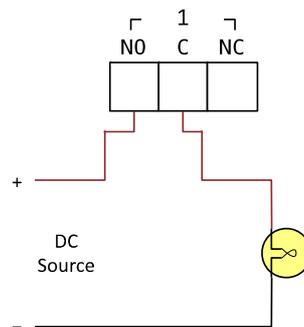
Figure 1 - Wiring a basic DC load

Inductive loads at high currents cause large voltage spikes when turned on or off, and this can disrupt sensitive electronics. For large inductive loads, a snubber is recommended. A DC load can be bypassed with a circulation diode. An AC load requires an RC snubber across the relay contacts or load. Ensure that diodes, resistors and capacitors used for snubbers are correctly rated for the load and voltage being switched.
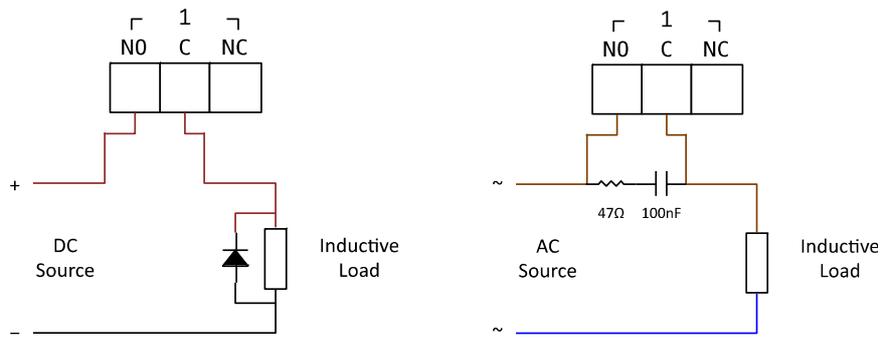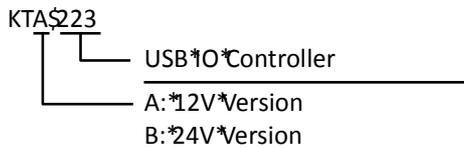
9

Figure 1 - Wiring Snubber Circuits

Selection Guide:

KTA-223

— USB*IO*Controller

A:*12V*Version
B:*24V*Version

Hardware Revision History:

KTA-223v1:   Initial release.

Erratum: R8 is not connected to ground. This was corrected on shipped boards by the addition of a jumper wire to the PCB. This fix was applied at the factory.

KTA-223v2:   Fix for R8

KTA-223v5:   Changed to a LM2841 switching regulator for easier assembly

KTA-223v9:   Added 3.3 V linear regulator to take load off FT232RL chip,

Added TX and RX LEDs, using a lightpipe and 556 to extend pulse time

Added FET and LED on D13,

Added pull up resistors and jumpers for RTC,

Changed current shunt resistors to precision 250 Ω

Changed regulator to LM2954,

Changed diodes and pull up resistors on TX/RX/TXEN lines to logic gates for better performance,

Changed to relays with higher coil to contact isolation rating

Added filter caps from USB shield to ground for better noise immunity

Added footprint, pull up and jumper for 1-Wire sensor

Changed position of ICSP header to match Arduino Uno

Added extra shield header pins to match Arduino Uno

KTA-223v12:   Changed serial logic gates to larger packages for more reliable assembly

Changed jumpers to a single 2x4, surface-mount block for easier assembly

Changed LEDs to side-mount units to remove light pipe

All other versions were internal prototypes.

Software Revision History:

The KTA-223 comes pre-programmed as an Arduino with a sketch loaded. The sketch can be upgraded using the Arduino IDE.

Software v1.0:   Initial Release.

Software v1.1:   Added TR and SS commands.

Software v1.2:   Added KA command.

Licensing:

The KTA-223 is derived from the Arduino Deumilanove and the schematics and CAD files are available under Creative Commons Attribution Share-Alike licenses. Contact info@oceancontrols.com.au for more information.

SEM.Test Compliance Service Co., Ltd.
3/F, Jinbao Commerce Building, Xin'an Fanshen Road,
Bao'an District, Shenzhen, P.R.C. (518101)

CERTIFICATE OF CONFORMITY

Certificate No.: SEM12105854

The following product has been tested by SEM.Test Compliance Service Co., Ltd. with the listing standards and found in conformity with the **EC Council Directive of 2004/108/EC**. It is possible to use CE marking to demonstrate the conformity with this **EMC Directive**.

Report No.	:	STR12108054E
Applicant	:	Ocean Controls
Address	:	Factory 3/24 Wise Ave, Seaford, Vic, Australia
Manufacturer	:	Ocean Controls
Address	:	Factory 3/24 Wise Ave, Seaford, Vic, Australia
Description of Product	:	Relayduino
Model No.	:	KTA-223
Test Standards	:	EN 61000-6-3: 2007+A1: 2011
		EN 61000-6-1: 2007

The referred test report(s) show that the product complies with the essential requirements in the above listed standards. The applicant is authorized to use this certificate in connection with the EC declaration of conformity according to Annex 1 of the Directive.

Test Laboratory

CE

Jandy So
Supervisor
Date of Issue: Oct. 19, 2012

This certificate of conformity is based on a single evaluation of the submitted sample(s) of the above mentioned product. It does not imply an assessment of the whole production and other relevant Directives have to be observed.

Tel.: +86-755-33663308   Fax.: +86-755-33663309   E-mail: sem@semtest.com.cn   Website: www.semtest.com.cn

DECLARATION OF CONFORMITY

This Product is in conformance with AS/NZS 61000.6.3 for residential, commercial and light-industrial environments.

Product Type:	Relayduino
Model:	KTA-223
Trade Name:	/
Responsible Party:	Ocean Controls
Address:	Factory 3/24 Wise Ave, Seaford, Vic, Australia
Contact Person/Title:	Peter Simmonds / Managing Director
Telephone:	+6 1397825882
Fax:	+6 1397825517

EUT Certification Summary

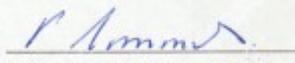|	
Equipment Class:	AS/NZS 61000.6.3
Report Number:	STR12108056E
Issuance Date:	Oct 19, 2007
Tested by:	SEM Test Compliance Service Co., Ltd.
Authorized Signature:	

Jandy so / PSQ Manager

We, the responsible party;

Ocean Controls

declare that the product

Relayduino

Was tested to conforms to the above mentioned Australia/New Zealand Standards(s).

Signature of Authorized Person	25/10/2012
	Date
PETER SIMMONDS	Managing Director
Print Name	Title
Signature of Authorized Person	Date
Name	Title

VERIFICATION OF CONFORMITY
According to FCC Part 15B

Certificate No.: SEM12103066

Responsible Party's Name	:	Ocean Controls
Address	:	Factory 3/24 Wise Ave, Seaford, Vic, Australia
Manufacturer	:	Ocean Controls
Address	:	Factory 3/24 Wise Ave, Seaford, Vic, Australia
Description of Product	:	Relayduino
Model No.	:	KTA-223
Report No.	:	STR12108055E-3

Compliance With Part 15B of FCC Rules.

This device complies with Part 15 of the FCC Rules. Operation is subject to the following two conditions: (1) This device may not cause harmful interference, and (2) This device must accept any interference received, including interference that may cause undesired operation.

Responsible Party:

OCEAN CONTROLS
Web: www.oceancontrols.com.au
Ph: +61 3 9782 5882
Post: PO Box 2191 Seaford BC
VIC 3198 Australia

Responsible Signature: _____

Name / Title: Peter Simmonds Managing Director.

Date: 25/10/2012

Tested By:

SEM.Test Compliance Service Co., Ltd.
3/F, Jinbao Commerce Building, Xin'an Fanshen Road, Bao'an District, Shenzhen, P.R.C.

Issued By: _____

Name / Title: Jandy So / PSQ Manager

Date of Issue: Oct 19, 2012

The Certification of Verification shows that the tested sample technically compliances with the FCC Part 15. The certification applies to the tested sample above mentioned only and should not implied an assessment of the whole.

Tel.: +86-755-33663308 Fax.. +86-755-33663309 E-mail: sem@semtest.com.cn Website: www.semtest.com.cn