RCC01 Programming Guide

To program the RCC01 fit the jumper provided onto the programming link, connect to a suitable RS232 terminal, such as MS Windows Hyper Terminal with a straight 9 way M-F lead, configure your terminal to 8 Data 2 Stop No Parity. Now apply power to the RCC01 – between 7 and 15 volts will be OK.
Your RCC01 should respond with :-

Remote contact closure over RS232 by Richard Drabble
Type string to be sent when local inputs change.
Type 2 x DC1 (Ctrl-Q) characters to mark the position of hex data
containing the input port information
DC1 characters may be anywhere in the string but must be contiguous
Maximum of 32 characters

In my example I will be setting up the units to work using a DMS 'short message' between unit 1000 and unit 1001 both units belonging to fleet 100. I will show programming for the RCC01 connected to unit 1001.
From the DMS spec published by Kenwood then I type:-

<Ctrl-B>F1001000<Ctrl-Q><Ctrl-Q>01<Ctrl-C>

When an input on this unit changes then the unit will initiate a short message call to unit 1000 (F1001000) containing 2 hex digits carrying the port data (marked by the Ctrl-Q characters) with a sequence number of 01.
Notice that any string of up to 32 characters is acceptable, with 2 of these being taken up by the actual port data. This allows plenty of flexibility to use other short message bearers provided by other manufacturers.
The RCC01 will now respond with:-

Type string that will be received when remote inputs change
Type 2 x DC1 (Ctrl-Q) characters to mark position of hex data
containing the input port information
DC1 characters may be anywhere in the string but must be contiguous
Type DC2 (Ctrl-R) characters as wildcards
Maximum of 32 characters

Here you are being asked for the format of the message you expect to receive from the remote unit when its inputs change state. Again with reference to the Kenwood DMS spec I enter:-

<Ctrl-BF1001000<Ctrl-Q><Ctrl-Q>01<Ctrl-C>

This tells the RCC01 where to find the port data in the incoming text, again notice that although I'm using Kenwood DMS the flexibility to use other bearers is there.
The RCC01 will now respond with :-

Is a positive acknowledgement confirming correct transmission expected Y/N

In order to confirm successful transmission the RCC01 can be programmed to look for a positive acknowledgement to confirm successful transfer of the data, alternatively the RCC01 can be programmed to simply hope for the best
In my example I will be using a positive ack.

Y

The RCC01 now responds with:-

Type string that will be received to confirm successful transmission
Type DC2 (Ctrl-R) characters as wildcards

Here you are being asked for format of the positive ack message; you may define wildcard characters if required.
Although I'm using Kenwood's DMS protocol, which it's self provides an ack which could be used I will use an ack generated by the remote RCC01, this has the advantage that is proves 'end-to-end' that the message got to the remote RCC01. If the unit was set up to use the Kenwood DMS ack it would only confirm delivery to the remote radio – not the remote RCC01.

<Ctrl-B>F1001000RXOK01<Ctrl-C>

The RCC01 now responds with:-

Input how long to wait for positive ack before retransmitting 15-255 (seconds)

If a positive ack is not received before this time expires the RCC01 will assume a failure and resend the original message. I will allow 30 seconds which should be plenty.

30

The RCC01 will now respond with:-

Should this unit send a positive ack when valid data received Y/N

In order to confirm that a valid message has been received the RCC01 can generate an ack message. Many data bearers, such as Kenwood's DMS will produce a positive ack for you, however some don't and in any case an ack from the bearer only proves delivery as far as the radio (or whatever) at the remote end, not the remote RCC01.
In my example I will setup end-to-end ack to prove delivery all the way to the remote RCC01

Y

The RCC01 will now respond with:-

Type string that will be sent to confirm valid data has been received
Maximum of 32 characters

Here the RCC01 is asking for the positive ack string, again sticking with Kenwood DMS my ack string is:-

<Ctrl-B>F1001000RXOK01<Ctrl-C>

This completes RCC01 programming.
You should now remove power and the programming jumper and connect the RCC01 serial port to the RS232 port used to access your data bearer, keeping in mind that the RCC01 is configured as DCE, so if you are connecting to another DCE you will need a crossover lead.

Operation of the RCC01 is simple, when an input changes state the change will be sent to the remote RCC01, the corresponding output on the remote unit will then also change state to mirror the local input.