

## RCP02 V22 Release Notes

These notes cover changes made to the RCP02 code since release V21 of 8 November 1999. If you are upgrading from an earlier release, please read those notes also.

### New Features

1. The RCP02 now supports eight analog voltage input lines that can be used to directly monitor continuously varying waveforms within the user's environment. ***No additional hardware is required to use this new feature***; only the addition of a few wirewrap wires on the Main Board and Backplane to bring out the external connections.

The analog input lines can accept voltages in the range  $-12\text{V}$  to  $+12\text{V}$ , and their input impedance is  $1\text{M}\Omega$ . Channels 0 through 3 are single ended (a single input relative to ground), and channels 4 through 7 are differential (difference of two inputs). The 12-bit A/D convertor operates in the  $-5\text{V}$  to  $+5\text{V}$  range for all channels. Thus, the single ended inputs must stay within  $-5\text{V}$  to  $+5\text{V}$  to be sampled properly; and the difference voltage on the differential channels must be likewise constrained.

The analog signal inputs are available on the following (already existing) pins of the RCP02/Main board.

Channel	+Input	-Input
-----	-----	-----
0	H2,3	N/A
1	H2,7	N/A
2	H2,11	N/A
3	H2,15	N/A
4	H2,17	H2,19
5	H2,21	H2,23
6	H2,27	H2,29
7	H2,31	H2,33

2. The new command "Monitor ADC" may be used to view the sampled voltage on each of the eight analog input lines. An internal loop-back measurement of the AZ and EL drive output voltages is also included. A sample printout is shown below; all values are displayed in Volts.

```
RCP> monitor adc
Analog Input Lines (Volts relative to GND)
  0      1      2      3      4      5      6      7  AzRef ElRef
-----
 0.92  4.04 -3.12  0.00  0.00  0.00  0.00  0.00  1.22  0.00
```

### Setup Changes

1. You may now configure Boolean variables whose values are based on comparison tests of the eight analog voltage input lines. In this way, the analog inputs can be thresholded and used as additional inputs to logic equations within the RCP02. Up to sixteen such variables may be defined, i.e., you may have, on the average, two threshold tests for each input line.

To setup the analog input variables, use the “Control ADC” command to define the following information for each voltage comparison test that you need:

```
RCP> control ADC
Analog Input Test Variable Definitions
-----
A/D Logic Variable #0 is defined: YES           New Value:
Description of A00 variable: 'HiTemp '         New Value:
Input summation term #1: A0                     New Value:
Input summation term #2: -A4                     New Value:
Input summation term #3: Zero                    New Value:
Input summation term #4: Zero                    New Value:
Test for ( A0-A4 > 3.55 Volts ) :               New Value:
```

This example defines a new Boolean status variable named “a00\_HiTemp”. This variable name will appear in the “?v” list of available status variables within the equation editor, and the variable may be used on the right side of any logic equation. The descriptive suffix is intended to make the variable meaningful and readable within the text of the logic equations. You may choose any 8-character name that does not contain spaces or punctuation other than '.', '\_', and '-'. The descriptive suffix can be omitted (not recommended) by entering a space at the prompt, but your logic equations will then become less readable.

The comparison test operates by summing the voltages on one or more input channels, and then testing whether that sum is greater than a specified voltage. If the test passes, then the variable is TRUE; otherwise it is FALSE. Moreover, the input channels can be either added or subtracted when computing the sum.

In the above example, “a0\_HiTemp” will be TRUE whenever the difference of the voltages on channels 0 and 4 is greater than 3.55 volts. If you wish to create variables with a negated sense, you may reverse the signs of the comparison tests. For example, we could have created “a1\_LowTemp” by defining the variable as:

```
A/D Logic Variable #1 is defined: YES           New Value:
Description of A01 variable: 'LowTemp '         New Value:
Input summation Term #1: A4                     New Value:
Input summation Term #2: -A0                     New Value:
Input summation Term #3: Zero                    New Value:
Input summation Term #4: Zero                    New Value:
Test for ( A4-A0 > -2.55 Volts ) :              New Value:
```

These could then be combined in a logic equation as follows:

```
EQ00: # V0 will be TRUE whenever the temperature is normal
\--: v0 = !(a0_HiTemp | a1_LowTemp)
-->
```