

**NAME**

microfast – Oxford/Tennelec/Nucleus Microfast MCA

**DESCRIPTION**

The Microfast is an ISA (PC card) multichannel analyzer available with 2048, 4096, 8192 or 16384 32-bit channels. The card can be used for counting to live-time, real-time or gross-counts presets. The card also includes an on-board discriminator.

**FUNCTIONS**

`mca_get(arr [, roi_beg [, roi_end]])` or `mca_get(g, e [, roi_beg [, roi_end]])` – Gets data from the currently selected MCA-type device, and transfers it to the array `arr` or to element `e` of data group `g`. Use of arrays rather than data groups is recommended. The native data type for the Microfast is `ulong`.

`mca_par("disable" [, arg])` – With no arguments, returns nonzero if the selected MCA device has been disabled by the user, otherwise returns zero. If `arg` is 1, disables the MCA. If `arg` is 0, turns off the disabled mode. When the device is disabled, `spec` will not access the hardware. On startup, and after the standard `config` macro or the `reconfig` command is run, disabled mode is off.

`mca_par("auto_run" [, arg])` – With no arguments, returns nonzero if the selected MCA device has auto-run mode set, otherwise returns zero. If `arg` is 1, enables auto-run mode. If `arg` is 0, turns off auto-run mode. When auto-run mode is set the device is started and stopped with the counting functions `tcount()`, `mcount()`, etc. When not set, the counting functions are ignored, but the device can be controlled with the "run" and "halt" options described below. In addition, the device can be halted with the `stop()` function and will be halted with `^C`. On startup, and after the standard `config` macro or the `reconfig` command is run, auto-run mode is on.

`mca_par("soft_preset" [, arg])` – With no arguments, returns nonzero if the selected MCA device has soft-preset mode set, otherwise returns zero. If `arg` is 1, enables soft-preset mode. If `arg` is 0, turns off soft-preset mode. When set, and if auto-run mode is enabled (see above), the MCA device is set to count for the time preset given as an argument to the `tcount()` function. The `wait()` function will wait until both the timer and the MCA device have counted to their respective presets. When soft-preset mode is not set, but auto-run mode is, the device is programmed for continuous data acquisition and will be stopped when the timer's preset is reached. Currently, when both soft-preset and auto-run modes are in effect, when counting to monitor (using `mcount()`) rather than to time, the device is also programmed for continuous data acquisition, as above.

`mca_par("run")` – Programs the MCA device with the appropriate parameters and starts acquisition.

`mca_par("halt")` – Halts the MCA device.

`mca_par("clear")` – Clears the data in the current group.

`mca_par("group_size" [, arg])` or `mca_par("gain" [, arg])` – With no argument, returns the current group size. Otherwise sets the group size to `arg`. Valid arguments are 256, 512, 1024, 2048, etc., up to the total number of channels. On the Microfast card, setting the gain is the same as setting the group size.

`mca_par("select_group" [, arg])` – With no argument, returns the currently active group. Otherwise, sets the active group to `arg`.

`mca_par("preset" [, arg])` – With no arguments, returns the current preset value. Otherwise, sets the preset to `arg`. The preset value is in seconds for live-time and real-time modes or number of counts for integral mode. If the preset is zero, the device will

programmed for continuous run, except when the soft-preset mode is in effect, as described above.

`mca_par("live")` – Sets the MCA device to live-time mode.

`mca_par("real")` – Sets the MCA device to real-time (or true-time) mode.

`mca_par("integral")` – Sets the MCA device to a mode where it runs until the counts in a region of interest reach the preset.

`mca_par("elapsed_live")` – Returns the elapsed live time in seconds.

`mca_par("elapsed_real")` – Returns the elapsed real time in seconds.

`mca_par("dead")` – Returns the dead-time percentage, calculated as  $100 \times (\text{real\_time} - \text{live\_time}) / \text{real\_time}$ .

`mca_par("gross_counts")` – Returns the total counts in all channels.

`mca_par("adc_lld" [, arg])` – With no argument, returns the current lower-level discriminator setting. Otherwise, sets the lower-level discriminator to *arg*. The default value is 32. Legal values are from 0 to 4095.

`mca_par("adc_uld" [, value])` – With no argument, returns the current upper-level discriminator setting. Otherwise, sets the upper-level discriminator to *value*. The default value is 4095. Legal values are from 0 to 4095.

`mca_par("adc_zero" [, value])` – With no argument, returns the current zero-level discriminator setting. Otherwise, sets the zero-level discriminator to *value*. The default value is 2047. Legal values are from 0 to 4095. Changing the zero-level shifts the spectrum to the left or right.

#### SEE ALSO

*arrays mca*