sensors

rcb. sensors

`new sensors()`

Sensor interface functions.

Source: rcbApi.js, line 914

Methods

`(static) averageResultsArray(resultsArray) → {object}`

Helper function that averages an array of 'results'. Must be an array of 'results', where 'results' is obtained from the read function.

Parameters:

Name	Type	Description
`resultsArray`	array	An array holding multiple results

Source: rcbApi.js, line 1102

Returns:

A single averaged results object structure. See the read function for more details on this object.

Type `object`

`(static) read(callback, averageQty`$_{opt}$`)`

Gets new sensor readings. Automatically averages a few readings for reducing noise. IMPORTANT: use the result.print() function to see the structure of the result variable. This structure will vary depending on the hardware (1520 or 1580), if there are accessories connected, or if debug mode is active. See the

example below for using the print() function. Note: each entry has 'working' and 'display' sections. 'working' will always remain the same, while 'display' will follow the user's display unit preferences. Use 'display' if reporting to the user, and use 'working' if performing calculations.

Parameters:

Name	Type	Attributes	Default	Description
callback	readSensorsReady			The function to execute when readings are ready.
averageQty	integer	<optional>	5	The number of samples to average before returning the result.

Source: rcbApi.js, line 947

Examples

```
//This sample script prints the content of the structure
//returned by the rcb.sensors.read callback
rcb.sensors.read(callback);

function callback(result){
    //print structure content
    result.print();
    rcb.endScript();
}
```

```
//Read 10 samples averaged, and print thrust on screen
rcb.sensors.read(callback,10);

function callback(result){
   var thrust = result.thrust.displayValue;
   var unit = result.thrust.displayUnit;
   rcb.console.print("Thrust: " + thrust.toPrecision(3) +
   rcb.endScript();
}
```

(static) readOhm(callback*opt*)

Reads the ohmmeter. If verbose mode is active, the reading will be displayed on the console.

Parameters:

Name	Type	Attributes	Description
callback	readOhmReady	<optional>	The function to execute when the reading is ready.

Source: rcbApi.js, line 991

Example

```
//Gets the ohmmeter reading
rcb.sensors.readOhm(callback);

function callback(reading){
    rcb.console.print("Ohm reading: " + reading.toPrecisi
    rcb.endScript();
}
```

(static) setMotorPoles(numberOfPoles)

Changes the number of motor poles. The correct number of poles is required to obtain a correct rpm reading.

Parameters:

Name	Type	Description
numberOfPoles	integer	The motor number of poles. Must be an multiple of 2.

Source: rcbApi.js, line 1089

Example

```
rcb.sensors.setMotorPoles(6);
```

(static) setSafetyLimit(sensorId, min, max)

Changes the safety limit for a sensor. Units are internal working units (A, V, RPM, g, and N·m) regardless of the user display units. It is not possible to set limits beyond hardware limits (values will automatically be trimmed).

Parameters:

Name	Type	Description
sensorId	string	"current", "voltage", "rpm", "thrust", or "torque".
min	number	Minimum sensor value before cutoff activates.
max	number	Maximum sensor value before cutoff activates.

Source: rcbApi.js, line 1061

Example

```
rcb.sensors.setSafetyLimit("current",10,20);
//rcb.endScript -> the safety cutoff will prevent motor f
```

(static) tareCurrent(callback*opt*)

Performs a tare function on the current sensor (helps overcome Hall effect hysteresis). Only supported on Hall effect current sensors, such as the ones used in the Series 1780. If trying to tare the current on other proucts, this function will have no effect other than calling the specified callback.

Parameters:

Name	Type	Attributes	Description

Name	Type	Attributes	Description
callback	tareCurrentComplete	<optional>	The function to execute when the tare is complete.

Source: rcbApi.js, line 1037

Example

```
//Simple script that only tares the current and finishes
rcb.sensors.tareLoadCells(rcb.endScript);
```

(static) tareLoadCells(callback*opt*)

Performs a tare function on the load cells.

Parameters:

Name	Type	Attributes	Description
callback	tareLoadCellsComplete	<optional>	The function to execute when the tare is complete.

Source: rcbApi.js, line 1014

Example

```
//Simple script that only tares the load cells and finish
rcb.sensors.tareLoadCells(rcb.endScript);
```

Documentation generated by JSDoc 3.6.3 on Wed Aug 21 2019 09:49:26 GMT-0400 (Eastern Daylight Time) using the Minami theme.