

Controller Parameters User Guide

Table of Contents

Introduction.....	3
Overview.....	3
Importing Parameters.....	4
Parameter List.....	4
Exporting Parameters.....	6
Parameter Rules.....	6
Rule Editor.....	8
Rule Item Editor.....	8
Rule Validation.....	10
Revision History.....	12

Controller Parameters User Guide

Introduction

The Controller Parameters utility was written to allow viewing or editing the contents of parameter files used by Hexagon Leitz protocol controllers and the new DC line of controllers. The parameters can be exported to a text file and merged into the original if this utility is used as an editor.

The key feature of this utility is the ability to create a set of parameter rules. The rules can be used to test almost any parameter against a constant or product from any combination of other parameters.

Overview

The Controller Parameters utility has three main sections:

- 1) The rule editor. This editor can be detached from the main program.
- 2) The list of parameter names and variables used by the rule editor.
- 3) The parameter viewer/editor. Parameters can be edited if needed.

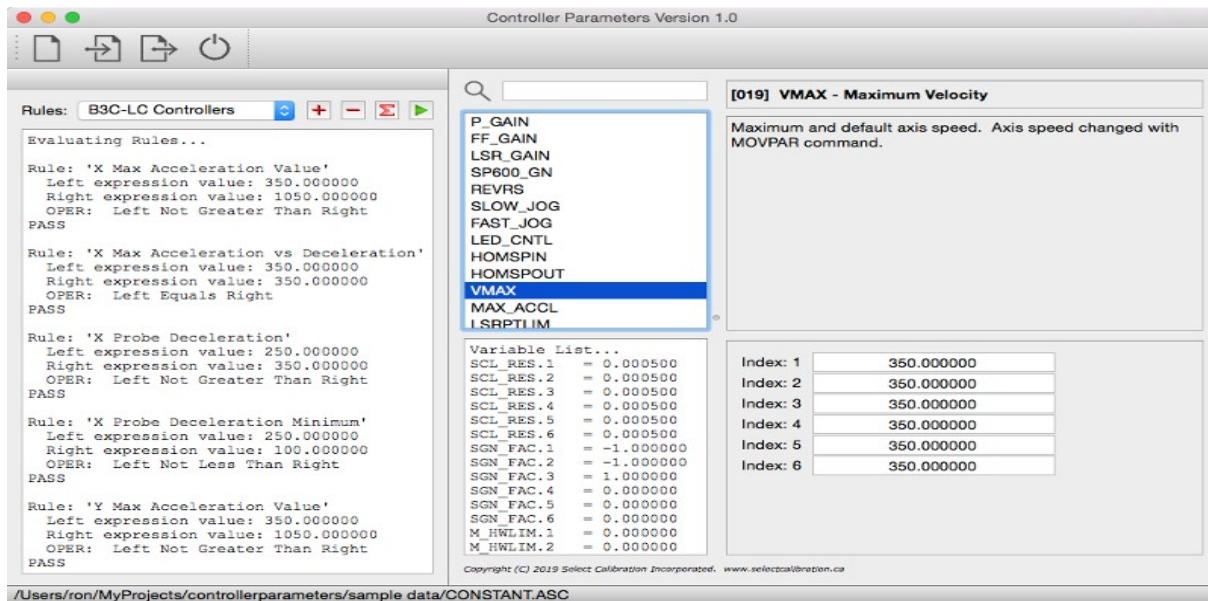


Illustration 1: Controller Parameters utility showing the parameter viewer / editor and rule evaluation.

Controller Parameters User Guide

Importing Parameters

Parameter data that is supported by the Controller Parameters utility can be loaded by one of two methods:

- Select *File - Open* from the main menu or by using the toolbar shortcut.
- Drag and drop the parameter file onto the Controller Parameter utility.

The currently supported parameter files is CONSTANTS.ASC from Leitz protocol controllers (FBPC, FB2, B3C-LC, ...) or any of the XML files from the DC controllers (DC800, DC240, DC241, RC1, ...).

Parameter List

When a parameter file is loaded the parameter list section shows all the entries found in the file. Illustration 2 shows an example of the axis data from a DC controllers XML file.

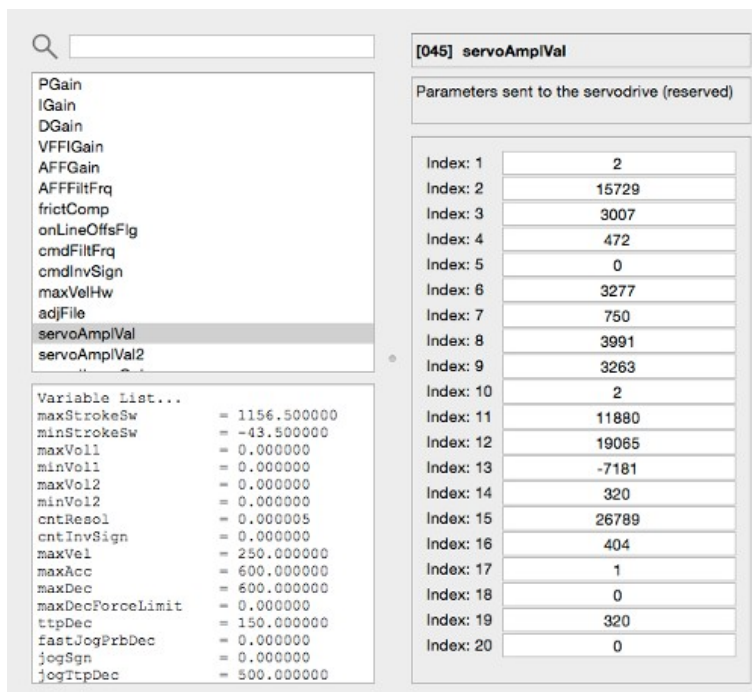


Illustration 2: Parameter List display section.

Parameter entries can be searched by entering all or part of a parameter name. The search will hide all entries that do not contain the search text.

Selected entries will activate an editor for viewing or modification of the selected parameter value. Illustration 3 shows an example of editing the bitfield parameter CMMCONFIG for a B3C-LC controller.

Controller Parameters User Guide

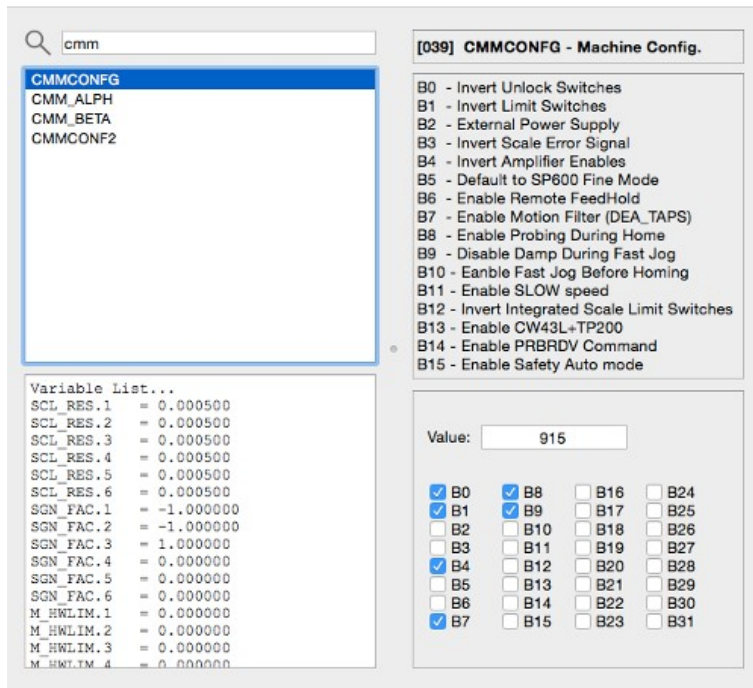


Illustration 3: Bitfield editor and search example.

The variable list below the parameter list shows all items that can be used in rule expressions. Illustration 4 shows an example of the variables created from the parameter VMAX.

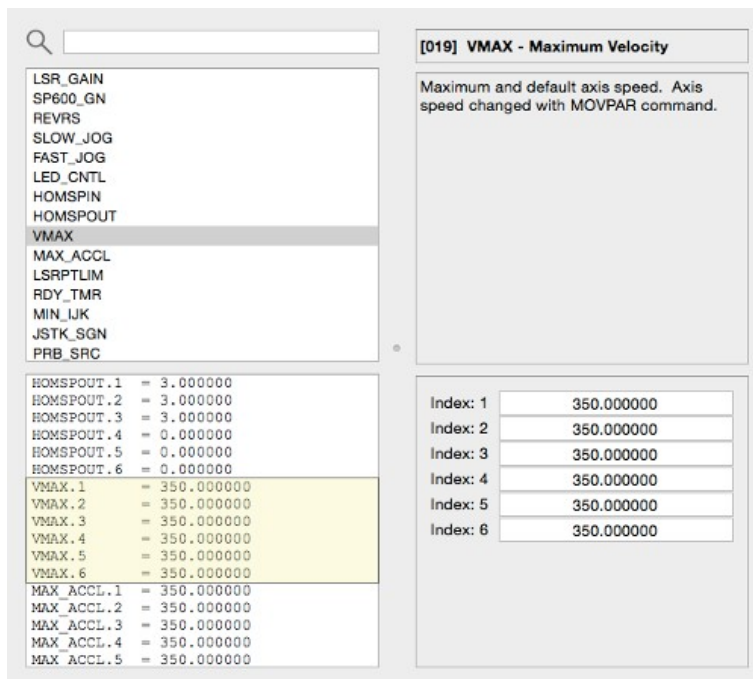


Illustration 4: Example of variables from the parameter VMAX.

Controller Parameters User Guide

Parameters can have a single entry or an array of data. Single entry parameters will have a corresponding variable with the same name as the parameter while parameter arrays will have variables created with an extension of `<dot><index>`. For example, the variable for the third index value of `VMAX` is `VMAX.3` (representing the Z axis).

Exporting Parameters

Parameters that have been modified can be exported to a text file by the Controller Parameters utility. The formatting of the original input parameter file is too quirky to re-create in its entirety so it was decided to save to a simple text file instead and not overwrite the original input file.

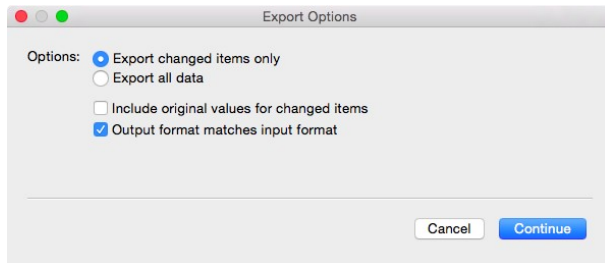


Illustration 5: Export options.

The format of the exported file can be similar to the format of the input file allowing easier transfer to the source parameter file. The exported file can be set to contain only parameters that have been modified or all of the data. The options when exporting parameter files are described in the following table:

<i>Option</i>	<i>Description</i>
Export changed items only	The output file will only contain entries that were changed. Unless extensive changes have been made the output file will be very small in this case.
Export all data	The output file will contain all parameters.
Include original values for the changed items	This applies only to parameters that have been changed. The output with this option checked will show the original value and the final value.
Output format matches input format	The format of the output file will mimic that of the input file if this option is checked otherwise a generic format is used.

Parameter Rules

A typical parameter file contains many entries and is therefore subject to many potential problems. There has been numerous examples of parameter values that have caused subtle problems with machines (particularly retrofit machines). Testing each parameter configuration manually is difficult and time consuming. The automated testing using the parameter rules option allows many of the mundane and easily overlooked entries to be validated.

Controller Parameters User Guide

Illustration 6 shows the parameter rule section which can be detached from the main program for convenience.

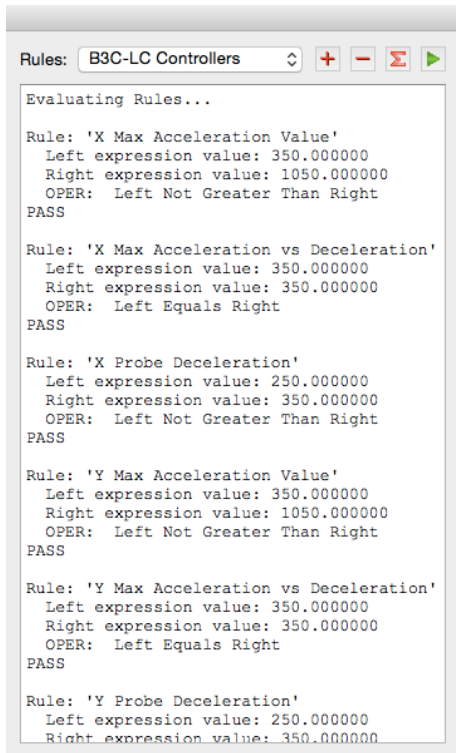






Illustration 6: Parameter rule utility.

Option	Description
Rules: <input type="text" value="B3C-LC Controllers"/>	Selection list for rule groups. In this example the rule group is named based on the type of controller it is intended for.
	Create a new rule group.
	Remove the selected rule group. This also removes all rules associated to the rule group.
	Edit the rules for the selected rule group.
	Evaluate the loaded parameter files using the rules in the selected rule group.

Controller Parameters User Guide

Rule Editor

Illustration 7 shows an example of the rule editor and all rules assigned to the selected rule group. Each individual rule entry is a specific test that is performed on the parameter data.

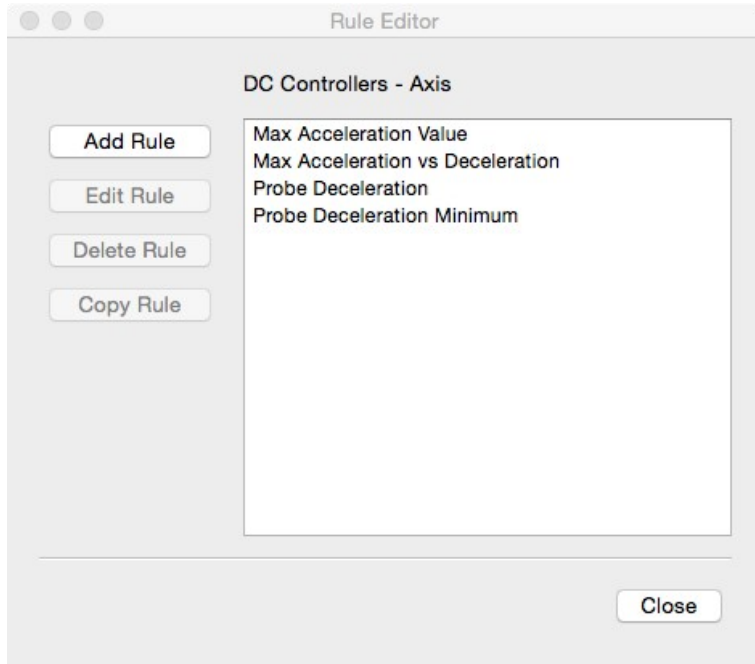


Illustration 7: Rule editor.

Options:

<i>Option</i>	<i>Description</i>
Add Rule	Create a new rule entry.
Edit Rule	Modify the selected rule entry. Double clicking on a rule entry will automatically open the rule for editing.
Delete Rule	Delete the selected rule(s).
Copy Rule	Copy the selected rule(s) to another rule group. Rules that exist in the target group are replaced.

Rule Item Editor

Rule items are created or edited using the rule item editor. Illustration 8 shows an example of the rule item editor.

The data in the rule item editor is not cleared from the previous state when adding new rule entries. This is useful when creating several rules that are similar but only differ by minor changes (axis name for example) as all the previous entries are left intact.

Controller Parameters User Guide

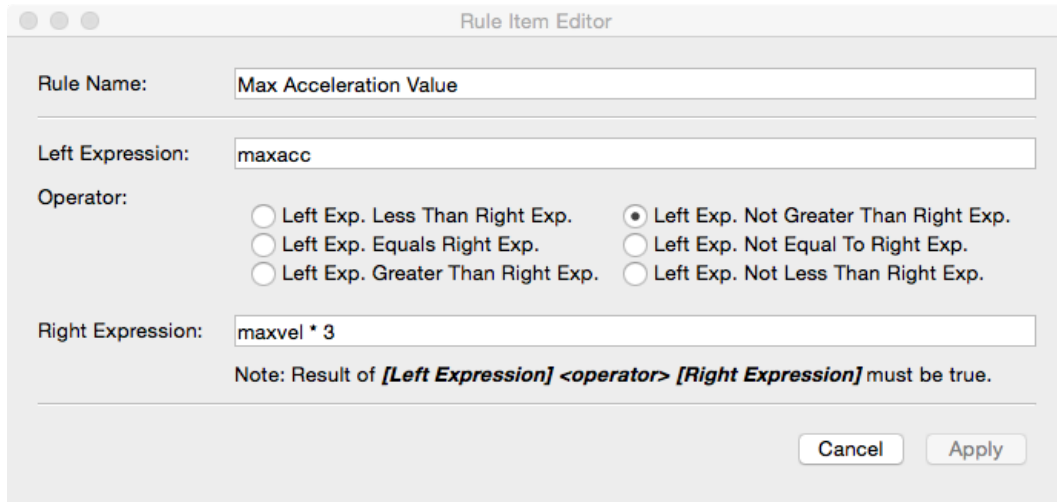


Illustration 8: Rule item editor

Options:

<i>Option</i>	<i>Description</i>
Rule Name	Name of the rule entry. The name can be between 1 and 64 characters in length. The rule name must be unique for any particular rule group.
Left Expression	Left side of the expression. The expression can contain constants, variables, and include standard operators (+-*/).
Operator	Operator for comparison between the left and right expression values.
Right Expression	Right side of the expression. The expression can contain constants, variables, and include standard operators (+-*/).

Each rule item is a logical test that produces a true or false result. In order for the rule item to pass the returned result must be true.

Result = Left_Expression [Operator] Right_Expression

Expression Operators:

<i>Operator</i>	<i>Description</i>
Left Exp. Less Than Right Exp.	The value of the left expression must be less than the right expression.
Left Exp. Equals Right Exp.	The left and right expressions must be the same.
Left Exp. Greater Than Right Exp.	The value of the left expression must be greater than the right expression.

Controller Parameters User Guide

<i>Operator</i>	<i>Description</i>
Left Exp. Not Greater Than Right Exp.	The value of the left expression must not be greater than the right expression.
Left Exp. Not Equal To Right Exp.	The left and right expressions must not be the same.
Left Exp. Not Less Than Right Exp.	The value of the left expression must not be less than the right expression.

The operators with *NOT* are the logical reverse of the corresponding operators without *NOT*. The order the operators are placed in the rule item editor group these ideas into rows. The following scenarios shows the logic behind the placement of the operators:

Left Expression : V1
Right Expression: V2

Scenario: V1 must not be greater than V2. V1 can equal V2.
Operator: *Left Exp. Not Greater Than Right Exp.*

Scenario: V1 must be the same as V2.
Operator: *Left Exp. Equals Right Exp.*

Scenario: V1 must be greater than V2. V1 cannot equal V2.
Operator: *Left Exp. Greater Than Right Exp.*

Rule Validation

The rules testing is performed by pressing the *Execute* button. The result from each rule item tested is displayed as a text result.

Example from running a set of rule tests:

Evaluating Rules...

Rule: 'Max Acceleration Value'
Left expression value: 600.000000
Right expression value: 750.000000
OPER: Left Not Greater Than Right
PASS

Rule: 'Max Acceleration vs Deceleration'
Left expression value: 600.000000
Right expression value: 600.000000
OPER: Left Equals Right
PASS

...

Example of a failed rule:

Controller Parameters User Guide

```
Rule: 'Probe Deceleration Minimum'  
Left expression value: 50.000000  
Right expression value: 100.000000  
OPER: Left Not Less Than Right  
* FAIL *
```

There are five lines generated from each rule item that is executed. The five lines contain the following information:

<i>Option</i>	<i>Description</i>
Rule:	The name of the rule.
Left expression value:	The value of the left side expression. Variables are substituted with actual values and operators are evaluated (+-* /).
Right expression value:	The value of the right side expression. Variables are substituted with actual values and operators are evaluated (+-* /).
OPER:	Description of the operator used for comparison of the left and right expressions.
PASS / FAIL	Indicator if the rule passed (result was true) or failed (result was false).

Controller Parameters User Guide

Revision History

<i>Date</i>	<i>Version</i>	<i>Changes</i>
Aug 9, 2019	1.0	New Program