

Using the FGC3 with the CERN LPC Power Converters

6 April 2020

Table of Contents

1	Introduction	1
2	USB interface	2
3	Connecting to the FGC3 terminal interface	2
4	Connecting to the FGC Spy interface	3
5	Boot program and Main program	4
6	Updating software in the boot program	5
7	Main program terminal modes	6
8	FGC Commands	6
9	FGC Properties	7
10	FGC Operational State	9
11	Resetting or power cycling an FGC3	10
12	LOAD model	11
13	LIMITS model	16
14	Regulation	23
15	FGC Power Converter State Machine	35
16	Configuration Properties	37
17	Reference Generator	38

1 Introduction

This guide is an introduction to using an FGC3 with Class 63 to control a small CERN LPC power converter (CUTE, CANCUN, MACAO), via the USB interface. It provides an overview of the features of Class 63, including some that are used with other larger converters.

2 USB interface

The FGC3 uses a 2-channel FTDI USB interface chip. The channels appear as consecutively numbered COM ports when connected to a computer with the FTDI drivers installed. The drivers are included in Windows 7 and Window 10 by default, MacOS and Linux. In Windows, the numbers of the COM ports can be seen in the Windows Device Manager and will depend on the number of other ports already present. An example is shown in Figure 1.

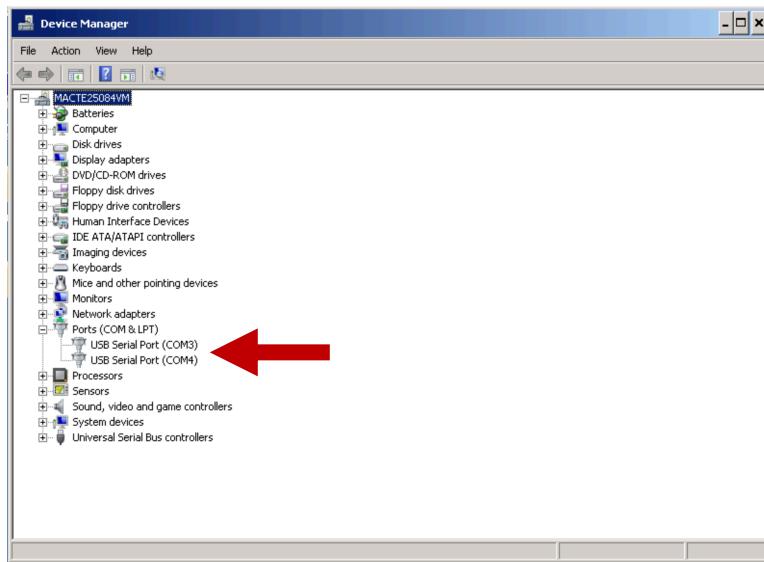


Figure 1 - Windows Device Manager showing FTDI USB serial COM ports

The lower numbered COM port provides the bi-directional terminal interface. This is limited to 960 characters per second because the older FGC2 had a physical RS232 port that runs at 9600 baud. This low speed limit is not really a problem for most short commands, but it requires patience when updating software or the programmable logic (this takes about 6 minutes). Furthermore, the terminal interface only supports ASCII command responses, which prevents the readout of logging properties that return binary data.

The higher numbered COM port is used for the uni-directional Spy interface. The FGC3 uses the Spy interface to transmits six floating point values per millisecond. They can be received using the fgcspy.py python program. This mitigates, to some extent, the inability to readout logging properties over the USB terminal interface.

3 Connecting to the FGC3 terminal interface

One MacOS and Linux, the screen command can act as a terminal emulator while on Windows, we recommend using teraterm4 to connect to the terminal COM port. A copy of teraterm4 is on the shared KT-FGC cernbox folder under Windows_Software/teraterm4. It does not need to be installed – just copy the teraterm4 folder somewhere on the C drive. It can be convenient to pin the application

(ttermpro.exe) to the taskbar or start menu. Connect the FGC3 and switch it on. Once the drivers are activated, you can choose the terminal COM port from the “Serial port...” option of the teraterm4 Setup menu as shown in Figure 2:

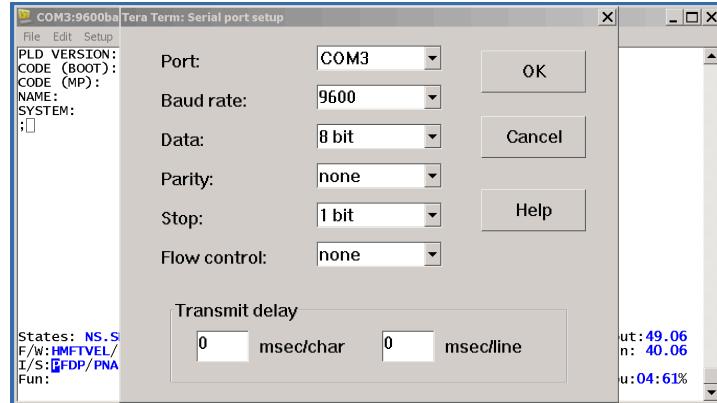


Figure 2 - Configuring the COM port in Tera Term

You can set the font size from the “Font...” option. Once the settings are correct, save them using the “Save setup...” option. If you use the default file name (“TERATERM.ini” in the teraterm4 folder) then when you start teraterm4 in future, it will use the same settings and connect to the COM port automatically, if it is active.

4 Connecting to the FGC Spy interface

The fgcspy.pl program can be launched to capture the 1 kspS stream of six signals from the FGC.

The choice of the six signals to send to the Spy interface is controlled by the property SPY.MPX. This can be changed at any time. The full list of available signals can be seen with the command “G SPY.MPX RANGE”.

The default signals are: I_REF, I_MEAS, V_REF, V_MEAS, I_A, I_B.

You can change the I_B signal to the current regulation error with the command “S SPY.MPX[5] I_ERR”. You can inform fgcspy.pl about this change:

```
fgcspy.pl I_ERR
```

The fgcspy.pl program can record the spy data in two ways:

1. Consecutive archive files with a specified number of seconds of data. The default is 10s. The file name includes the date and time of the first sample.
2. Rolling data file with a specified number of seconds of data, updated periodically. The default is 20s of data, updated once per second.

These files can be viewed using PowerSpy (<https://service-powerspy.web.cern.ch>). By using Chrome, the rolling data file will be reloaded automatically every time it is updated.

5 Boot program and Main program

The FGC3 has two different programs in its Flash memory:

- Boot program – this always runs first after a reset. It is a bit like the bios on a PC. It provides important low-level services:
 - Software updates from the USB or FGC_Ether interface, including updating itself
 - Programmable logic updates from the USB or FGC_Ether interface
 - Database updates from the USB or FGC_Ether interface
 - Self-tests
 - Starting the main program
- Main program – this controls a power converter:
 - Instantiates properties and supports set and get commands
 - Supports the terminal and Spy interfaces

The boot program will launch the main program automatically within a few seconds, if no faults are detected. The user can stay in the boot by pressing the ESC key during the countdown, or by running the command “S DEVICE.BOOT” in the main program.

The user can interact with the boot using numbered menus. For example, the root menu is:

```
Root menu
B:A G:1 T:1497109742 F:0200 W:0000 L:0000 ST:FO
0* RunMain
1 Run Log
2 Device status
3 Manual Tests
4 Self tests
5 Codes
6 Expert
7 Spy
8* Build info
```

Items with a “*” after the number are executable actions, while the other items are submenus. All the boot menus are documented on the FGC web site under the FGC3 platform.

Some actions require parameters. In this case, a leading comma is added automatically, and the user should enter the parameters separated by commas.

The ESC key will return to the root menu. The Delete key will return the parent menu.

6 Updating software in the boot program

In the boot program, root menu option 5 (Codes) allows the software and data structures to be updated. Option 0 of the codes menu lists the state of all the installed codes. For example:

```
5 Codes
B:A G:1 T:1497110733 F:0201 W:0001 L:0200 ST:F0
0* Show installed codes
1* Show advertised codes
2* Update codes from Fieldbus
3* Update code from Terminal
4* Toggle flash zone lock (4,<flash zone index>)
5* Erase Flash zone (5,<flash zone index>)
$0,50,Show installed codes
$5,FLASH ZONE|RESIDENT CODE|CLS:CODE:CALC:STATE|CODE AFTER UPDATE|
$2, 0:PLD |60_PLD_14_ANA101 | 60:696B:696B:Valid |OK |0000 |
$2, | 1455271244 | | | 0 |
$2, 1:BL27-24|60_BootProg | 60:7465:7465:Valid |OK |0000 |
$2, | 1496154728 | 30/05/2017 16:32:08 | | 0 |
$2, 2:BL23-17|63_MainProg | 61:6B33:6B33:Valid |OK |0000 |
$2, | 1496052532 | 29/05/2017 12:08:52 | | 0 |
$2, 3:C62_3 |31_IDProg | 31:F82A:F82A:Valid |OK |0000 |
$2, | 1467302040 | | | 0 |
$2, 4:C62_456|60_IDDB | 60:4627:4627:Valid |OK |0000 |
$2, | 1495009237 | | | 0 |
$2, 5:C62_1 |60_PTDB | 60:D708:D708:Valid |OK |0000 |
$2, | 1495009237 | | | 0 |
$2, 6:BL14 |60_SysDB | 60:FBF8:FBF8:Valid |OK |0000 |
$2, | 1496910933 | 08/06/2017 10:35:33 | | 0 |
$2, 7:BL16 |60_CompDB | 60:770F:770F:Valid |OK |0000 |
$2, | 1496910933 | 08/06/2017 10:35:33 | | 0 |
$2, 8:BL18 |60_DIMDB | 60:EE7F:EE7F:Valid |OK |0000 |
$2, | 1496910933 | 08/06/2017 10:35:33 | | 0 |
$2, 9:BL01 |NameDB | 0:3291:3291:Valid |OK |0000 |
$2, | 0 | | | 0 |
$2,Writable devices:0,Read-only devices:0
$3
```

Programs, programmable logic and databases (DBs) are packaged in “codes” which have a version number that is the Unix time when the code was created, for example, 1455271244. Codes are protected by a 16-bit CRC checksum.

A code can be updated through the terminal interface by running the code menu option 3 (Update code from Terminal) and then selecting the “Send file...” option of the File menu in teraterm4. The boot will wait up to 40 seconds for the code to be sent before giving up and returning to the menu. If this happens, just enter 3 again and try to be quicker. It is essential in teraterm4 to select “binary data” in the Send File dialogue when sending a code.

7 Main program terminal modes

The main FGC3 software class supports two modes for the terminal interface:

- Editor mode – for a human user
- Direct mode – for a computer program

Editor mode is enabled using the ESC key and is described in FGC_Terminal_interface.pdf, which is in the FGC_Documentation\FGC_Software cernbox folder. In editor mode, the FGC will echo the characters it receives, and will provide command line editing and history, much like a basic Unix shell. It also dedicates five lines at the bottom of the 80x24 character display to show information updated in real-time. Four lines have a fixed format, and one is selectable using the RTD property.

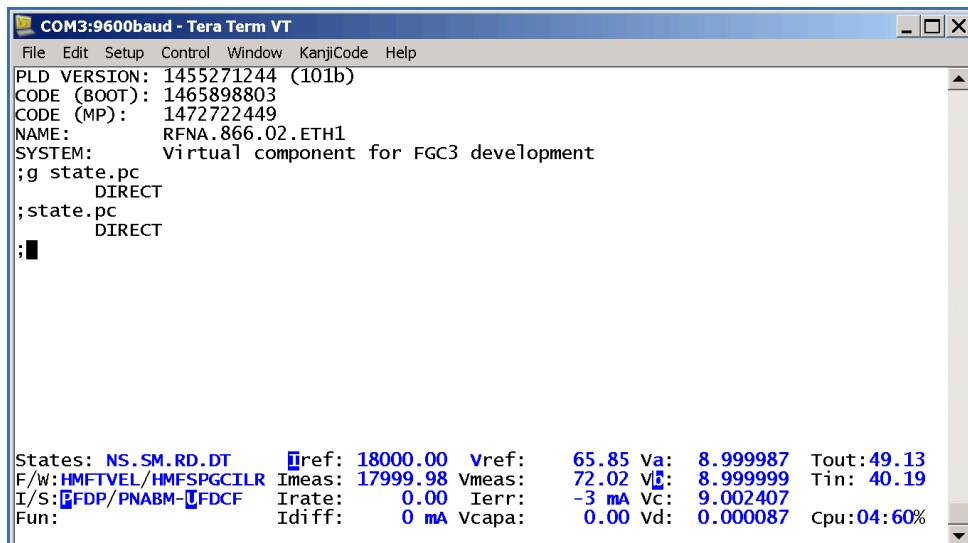
In direct mode, the Serial Command Response Protocol (SCRP) is used. Characters are not echoed. The protocol is described in the document FGC_Command-Response_Protocol.pdf. It is enabled with Ctrl-Z, or the start of command delimiter (!).

8 FGC Commands

The FGC3 main software class only supports two commands (note that commands and property names are not case-sensitive):

- S or s *Set property*
- G or g *Get property*

The get property command is the default when using **editor mode**, so if a property name is entered, the get command will execute and the value of the property will be displayed. An example is shown in Figure 3:



A screenshot of the Tera Term VT terminal window titled "COM3:9600baud - Tera Term VT". The window shows the following text output:

```
PLD VERSION: 1455271244 (101b)
CODE (BOOT): 1465898803
CODE (MP): 1472722449
NAME: RFNA.866.02.ETH1
SYSTEM: Virtual component for FGC3 development
;g state.pc
    DIRECT
;state.pc
    DIRECT
;■

States: NS.SM.RD.DT  Iref: 18000.00  Vref:  65.85  Va:  8.999987  Tout: 49.13
F/W: HMFTVEL/HMFSPGCLRL  Imeas: 17999.98  Vmeas:  72.02  Vb:  8.999999  Tin: 40.19
I/S: FDP/PNABM-TDFDCF  Irate:  0.00  Terr:  -3 mA  Vc:  9.002407
Fun: Idiff:  0 mA  Vcapa:  0.00  Vd:  0.000087  cpu: 04:60%
```

Figure 3 - Getting a property with the FGC terminal

In **direct mode**, there is no default command and the command must always be preceded by the start of command delimiter (!). For example, the direct command:

```
!G POLL\n                                (\n is a linefeed character, 0x0A)
```

will receive a response like this:

```
$TIME_NOW:7962295.486000\n
FAULTS:\n
WARNINGS:SIMULATION\n
ST_LATCHED:\n
ST_UNLATCHED:PC_PERMIT VS_POWER_ON NO_50HZ_SYNC\n
STATE_OP:SIMULATION\n
STATE_PC:DIRECT\n
ST_ADC_A:V_MEAS_OK\n
ST_ADC_B:V_MEAS_OK IN_USE\n
ST_ADC_C:V_MEAS_OK\n
ST_ADC_D:V_MEAS_OK\n
ST_DCCT_A:\n
ST_DCCT_B:\n
REF_I:1.7999999E+04\n
REF_V:6.5871591E+01\n
MEAS_I:1.8000003E+04\n
MEAS_V:7.2003908E+01\n
;
```

Command responses are preceded by the start of response delimiter (\$) and are terminated by the end of response delimiter (;).

The POLL property was defined specifically for programs that use the serial interface. It collects together the most important information under one top-level property. Each field in the POLL property is also available in an individually named property, but getting POLL is more efficient than getting 18 different properties.

9 FGC Properties

The list of properties in Class 63 is visible on the FGC web site on the KT-FGC cernbox at:

FGC_Website/fgc/gendoc/def/Class63_properties.htm

There are hundreds of properties, but only very few are important for the basic control of a non-cycling DC power converter. Table 1 provides a list of the most important properties for running the converter, once the FGC has been correctly configured.

Table 1 - Important FGC properties for running a power converter

Property	Set/Get	Behavior
MODE.PC	S/G	Desired power converter state.
STATE.PC	G only	Actual power converter state.
MODE.OP	S/G	Desired operational state.
STATE.OP	G only	Actual operational state.
POLL	G only	Useful properties concerning the state and status of the power converter.
DCCT.SELECT	S/G	Select DCCT A or B or the average of A and B, for the regulation.
REG.MODE	S/G	Set I or V to choose current or voltage regulation mode.
REF	S/G	Top-level property for reference generation.
REF.DIRECT.V.VALUE	S/G	Voltage reference when in DIRECT state and REG.MODE is V.
REF.DIRECT.I.VALUE	S/G	Current reference when in DIRECT state and REG.MODE is I.
REF.FUNC.TYPE	S/G	Function to arm in IDLE state.
REF.RUN	S/G	Time to start function in ARMED state (no value means now).
REF.ABORT	S/G	Abort function in RUNNING state (no value required).
SPY.MPX	S/G	Selector for the six signals that are emitted through the Spy interface.
DEVICE.RESET	S only	Reset the FGC3 and try to return to the main program
DEVICE.BOOT	S only	Reset the FGC3 and stay in the boot program
DEVICE.PWRCYC	S only	Power cycle the FGC3 and try to return to the main program

Most properties use symbol lists that link labels with the integer values used internally by the software. The list of symbols for a property can be displayed by using the “RANGE” get option. For example:

```
G MODE.OP RANGE
(NORMAL SIMULATION TEST)
SIMULATION
```

The possible symbols are listed in parenthesis, space separated, before the line reporting the value of the property.

For a numerical property like REF.DIRECT.I.VALUE, the RANGE get option will display the minimum and maximum values that can be set.

```
G REF.DIRECT.I.VALUE RANGE
(-1.00000E+06 1.00000E+06)
0.00000E+00
```

Some properties are arrays. The array index or range can be specified using square brackets:

```
S TEST.FLOAT[3] 1.234
S TEST.FLOAT[3,5] 3.0,4.0,5.0
```

10 FGC Operational State

All FGC software classes (“main” programs and the “boot” program), implement the operational state machine, visible through the STATE.OP property. This state machine is illustrated Figure 4 and the states are described in Table 2.

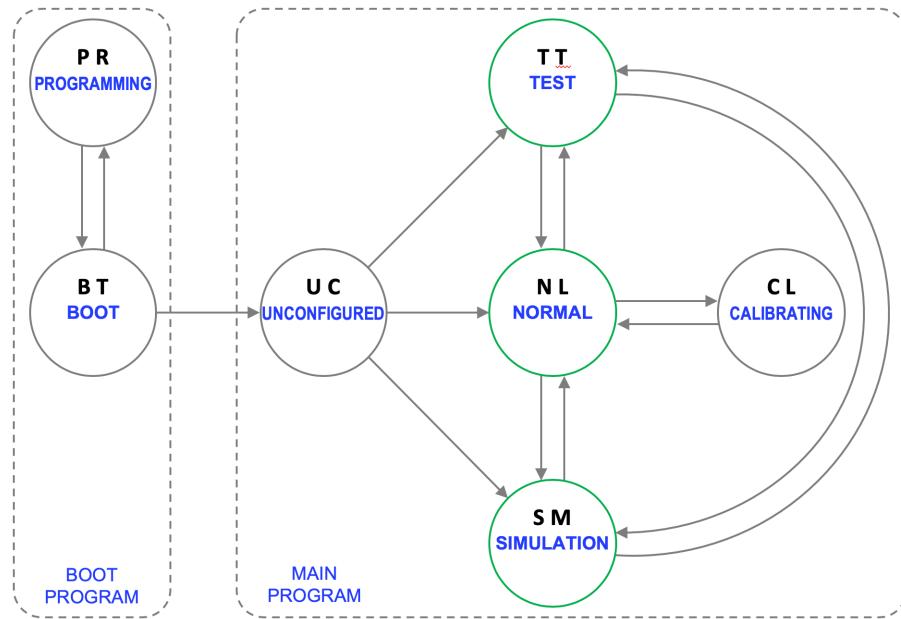


Figure 4 - Operational State Machine

The desire operational state for the main program is specified in MODE.OP. Three choices are possible:

- NORMAL – operate the power converter
- SIMULATION – simulate the power converter and, optionally, the load and measurement system
- TEST – allow tests of the analog interface

The boot program can be in two possible operational states, and the main program in five.

SIMULATION state is helpful for software development. It can simulate the converter state and the analog behavior of the load and measurement of signals, or just the converter state. In the latter case, an analogue load model can be used which means that the DAC and ADCs are used normally. This is possible with the portable FGC3 crate supplied by CERN. It has a circuit simulator card with an adjustable load time constant.

The operational state machine will be blocked in UNCONFIGURED state if any “configuration” properties were never set (see the section on Configuration Properties).

The desired operational state has no real meaning in the boot program.

Table 2 - States in the Operational State Machine

	State	Abbreviation	Program	Notes
0	UNCONFIGURED	UC	Main	At least one configuration property was never set. It is not possible to start the power converter when in UNCONFIGURED state.
1	NORMAL	NL	Main	The power converter can be operated.
2	SIMULATION	SM	Main	The power converter state (and optionally the load) are simulated. The SIMULATION warning will be active.
3	CALIBRATING	CL	Main	Either the DACs, ADCs or DCCTs are being calibrated. This is a transient state, lasting as long as the calibration process.
4	TEST	TT	Main	The power converter cannot be operated. This state is used to allow manual tests of the analog interface.
5	BOOT	BT	Boot	The boot program is running, but codes are not being updated.
6	PROGRAMMING	PR	Boot	The boot program is updating a code.

11 Resetting or power cycling an FGC3

It is possible to reset or power cycle an FGC3 manually and through the software.

11.1 Manual reset

The FGC3 can be reset manually by pressing the reset button on the front panel. Note that disconnecting the network address dongle will have the same effect as pushing the reset button. For as long as the button is pressed, the FGC3 will be held in reset and all the front panel LEDs will be blue.

11.2 Manual power cycle

The FGC3 can be power cycled manually, independently of the other cards in the crate, by pressing the reset and power cycle buttons on the front panel at the same time.

11.3 Software reset

The FGC3 can be reset in the boot by entering the boot menu command 621 (Global reset).

The FGC3 can be reset in the main program by either:

1. S DEVICE.RESET
2. S DEVICE.BOOT

The first option will try to return to the main program after the reset, while the second option will remain in the boot.

11.4 Software power cycle

The FGC3 can be power cycled in the boot by entering the boot menu command 620 (Power cycle).

The FGC3 can be power cycled in the main program running the command S DEVICE.PWRCYC.

12 LOAD model

12.1 First order model

The Class 63 load model is based on a simple first order magnet load model show in Figure 5.

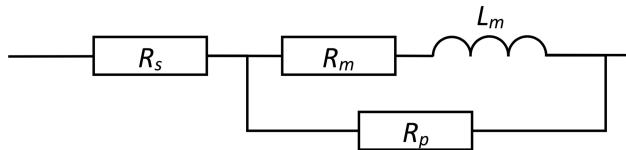


Figure 5 - First-order Load Model

The series, parallel and magnet resistances and magnet differential inductance are defined in the following properties:

- LOAD.OHMS_SER = R_s
- LOAD.OHMS_MAG = R_m
- LOAD.OHMS_PAR = R_p
- LOAD.HENRYS = L_m

So, if the magnet is superconducting, then LOAD.OHMS_MAG is zero and if the circuit it is just resistive, then LOAD.HENRYS is zero.

If there is no damping resistor then LOAD.OHMS_PAR should be set to at least 1.0E+08. In this case, only the sum of LOAD.OHMS_SER and LOAD.OHMS_MAG is significant.

The continuous-time transfer function from voltage to current for this model is:

$$G(s) = \frac{1}{R_s + \frac{1}{R_p + \frac{1}{R_m + sL_m}}}$$

This has a bode plot with the form shown in Figure 6.

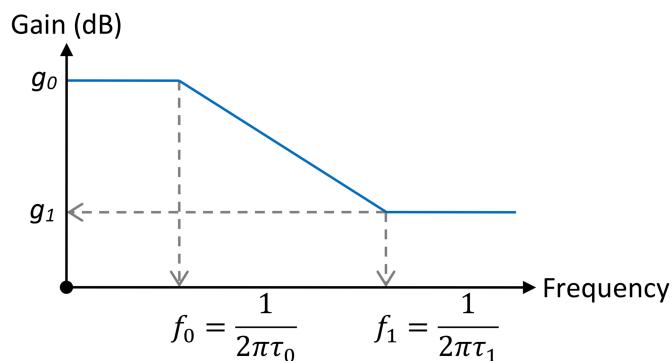


Figure 6 - Bode plot of first-order load model

Where the frequencies f_0 and f_1 are defined by:

$$f_0 = \frac{R_s + \frac{R_s}{1+R_s/R_p}}{2\pi L_m} \quad \text{and} \quad f_1 = \frac{R_m + R_p}{2\pi L_m}$$

and the DC gain g_0 and the high frequency gain g_1 are given by:

$$g_0 = \frac{1}{R_s + \frac{R_m}{1+R_m/R_p}} \quad \text{and} \quad g_1 = \frac{1}{R_s + R_p}$$

12.2 Magnet Saturation

Warm magnets typically saturate as the field approaches 1 tesla (10k gauss). As an example, Figure 7 shows a measurement of the differential inductance in Henrys (L_m) and magnetic energy in MJ (E) for the CERN proton synchrotron (CPS) main magnets, as a function of magnet current in amps (I_m).

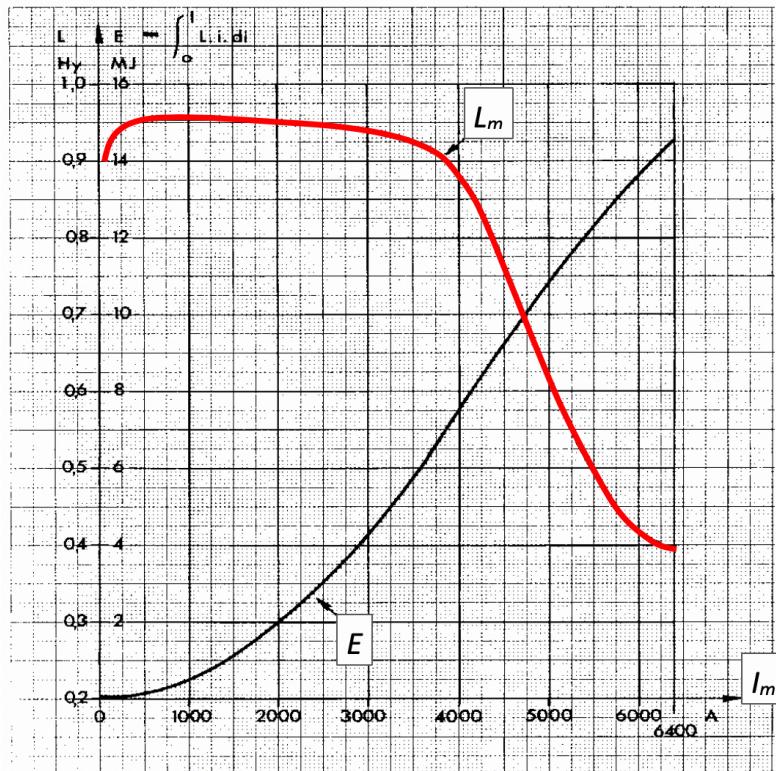


Figure 7 - Magnet Saturation in the CERN Proton Synchrotron Main Magnets

The magnet's differential inductance L_m declines dramatically as the current rises above 4000A and if this is not compensated, the performance of a current regulator will be impaired. Class 63 includes a simple compensation algorithm based on a linear-parabola-linear-parabola-linear model of the differential inductance illustrated in Figure 8. Note that when regulating field, magnet saturation is a second order effect, so no compensation is required.

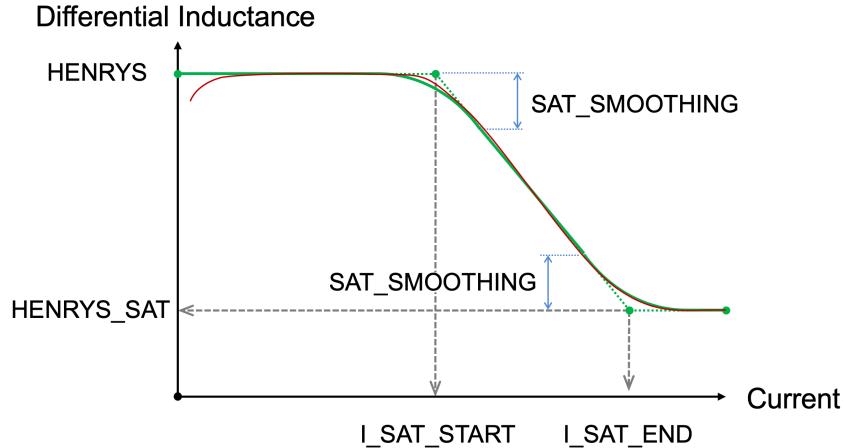


Figure 8 - Magnet Saturation Model

The saturation model parameters are HENRYS, HENRYS_SAT, I_SAT_START and I_SAT_END, which define three linear segments. It can then be smoothed with symmetric parabolic segments using the SAT_SMOOTHING parameter. This goes from 0 (no smoothing) to 0.5, in which case the two parabolas meet in the middle and there is no middle linear segment. Typically, a value of around 0.15 gives good results.

The properties that contain the saturation model parameters are:

- LOAD.HENRYS_SAT
- LOAD.I_SAT_START
- LOAD.I_SAT_END
- LOAD.SAT_SMOOTHING

Thus, the differential inductance for a given current can be derived from the normalized saturation model $f(I)$:

$$L_m(I) = f(I) \cdot L_m$$

The reduction in the differential inductance can be hidden from the RST algorithm by a simple equation that modifies V_{ref_reg} to derive V_{ref_sat} :

$$V_{ref_sat} = \{1 - f(I)\} IR + f(I) \cdot V_{ref_reg}$$

Where R is the resistance corresponding to the load pole:

$$R = R_m + \frac{R_s}{1 + R_s/R_p}$$

From this we can see that as $f(I)$ reduces from 1, a larger and larger proportion of V_{ref_sat} comes from Ohms law (IR). The regulator becomes less feedback and more feedforward. From experience, this still gives good performance as $f(I) \rightarrow 0.2$. But if $f(I) \rightarrow 0$, the regulator will stop rejecting perturbations.

12.3 LOAD SELECT

At CERN, many large converters in the Large Hadron Collider are installed in service tunnels parallel to the accelerator tunnel and separated by about 10m. Large water-cooled cables pass between the tunnels and are connected to the feed throughs on the cryogenic feeder boxes. At the end of each shutdown, while the cryostat is still not cold enough to power the magnets, a 24-hour heat run of every power converter is conducted. This is done by bolting the copper plates at the end of the cables together to create a circuit with the normal resistance but almost no inductance. Having tuned the circuit parameters and regulator to work with this “cable circuit”, the settings should be remembered so that they can be used after every shutdown.

This is achieved by making all the load-related properties arrays and using the property LOAD.SELECT as the array index to select the active value in every property.

The number of loads is four and the de-facto use of the four “slots” is:

- [0] – Normal load
- [1] – Cable circuit
- [2] – Short circuit
- [3] – Test

Index 2 (Short circuit) can be used to capture the circuit and regulator configuration that works with a short circuit at the output of the converter. This is normally only used for converter reception tests, but might be used again in the event of major repairs.

Index 3 (Test) is available for any other applications.

The full list of properties that are arrays of four elements where the index is LOAD.SELECT is available in Table 3.

Table 3 - Load-related array properties that use LOAD.SELECT

LIMITS.B.CLOSELOOP	REF.DEFAULTS.I.MIN_RMS
LIMITS.B.ERR_FAULT	REF.DEFAULTS.I.PRE_FUNC_MAX
LIMITS.B.ERR_WARNING	REF.DEFAULTS.I.PRE_FUNC_MIN
LIMITS.B.NEG	REF.DEFAULTS.PLATEAU_DUR
LIMITS.B.POS	REF.DEFAULTS.V.ACCELERATION
LIMITS.B.RATE	REF.DEFAULTS.V.DECELERATION
LIMITS.B.STANDBY	REF.DEFAULTS.V.LINEAR_RATE
LIMITS.I.CLOSELOOP	REF.DEFAULTS.V.MIN_RMS
LIMITS.I.ERR_FAULT	REF.DEFAULTS.V.PRE_FUNC_MAX
LIMITS.I.ERR_WARNING	REF.DEFAULTS.V.PRE_FUNC_MIN
LIMITS.I.LOAD	REG.B.EXTERNAL
LIMITS.I.LOAD.RMS_FAULT	REG.B.EXTERNAL_ALG
LIMITS.I.LOAD.RMS_TC	REG.B.EXTERNAL.CLBW
LIMITS.I.LOAD.RMS_WARNING	REG.B.EXTERNAL.MEAS_SELECT
LIMITS.I.LOW	REG.B.EXTERNAL.MOD_MARGIN
LIMITS.I.NEG	REG.B.EXTERNAL.SMOOTH_ACTUATION

LIMITS.I.POS	REG.B.EXTERNAL.TRACK_DELAY_PERIODS
LIMITS.I.RATE	REG.B.EXTERNAL.Z
LIMITS.I.STANDBY	REG.B.INTERNAL
LIMITS.I.ZERO	REG.B.INTERNAL.AUXPOLE1_HZ
LIMITS.V.NEG	REG.B.INTERNAL.AUXPOLES2_HZ
LIMITS.V.POS	REG.B.INTERNAL.AUXPOLES2_Z
LOAD.GAUSS_PER_AMP	REG.B.INTERNAL.MEAS_SELECT
LOAD.HENRYS	REG.B.INTERNAL.PURE_DELAY_PERIODS
LOAD.HENRYS_SAT	REG.B.INTERNAL.SMOOTH_ACTUATION
LOAD.I_SAT_END	REG.B.PERIOD_ITERS
LOAD.I_SAT_GAIN	REG.I.EXTERNAL_ALG
LOAD.I_SAT_START	REG.I.EXTERNAL.CLBW
LOAD.OHMS_MAG	REG.I.EXTERNAL.MEAS_SELECT
LOAD.OHMS_PAR	REG.I.EXTERNAL.MOD_MARGIN
LOAD.OHMS_SER	REG.I.EXTERNAL.SMOOTH_ACTUATION
LOAD.SAT_SMOOTHING	REG.I.EXTERNAL.TRACK_DELAY_PERIODS
REF.DEFAULTS.B.ACCELERATION	REG.I.EXTERNAL.Z
REF.DEFAULTS.B.DECCELERATION	REG.I.INTERNAL.AUXPOLE1_HZ
REF.DEFAULTS.B.LINEAR_RATE	REG.I.INTERNAL.AUXPOLES2_HZ
REF.DEFAULTS.B.MIN_RMS	REG.I.INTERNAL.AUXPOLES2_Z
REF.DEFAULTS.B.PRE_FUNC_MAX	REG.I.INTERNAL.MEAS_SELECT
REF.DEFAULTS.B.PRE_FUNC_MIN	REG.I.INTERNAL.PURE_DELAY_PERIODS
REF.DEFAULTS.I.ACCELERATION	REG.I.INTERNAL.SMOOTH_ACTUATION
REF.DEFAULTS.I.DECCELERATION	REG.I.PERIOD_ITERS
REF.DEFAULTS.I.LINEAR_RATE	

13 LIMITS model

Class 63 has an extensive list of limits properties presented in Table 4. Most are arrays of four elements based on LOAD.SELECT (see Table 3). The limits properties serve several important purposes:

- They inform the software if the power converter has 1, 2 or 4-quadrants
- They inform the software if field, current or voltage signals are not in use
- They define real-time fault limits on:
 - voltage, current and field measurements
 - filtered RMS measured current
 - filtered RMS rate of change of voltage reference
 - regulation error (field or current and voltage)
 - difference between channel A and B current measurements
- They define real-time warning limits on
 - filtered RMS measured current
 - regulation error (field or current and voltage)
 - difference between channel A and B current measurements
- They define real-time clip limits on
 - voltage, current and field references and power (via the voltage reference)
 - rates of change of voltage, current and field references
- They define real-time low and zero limits on measured field and current
- They define the current and field close loop thresholds for unipolar (1 and 2-quadrant) converters
- They define the standby current and field reference levels
- When attempting to arm a reference function, the function value and rate of change are checked and must respect the limits for the function to be armed

13.1 1, 2 and 4-Quadrant Converters

The four quadrants are defined in the classical way by the POS and NEG limits for current and voltage, as shown in Figure 9.

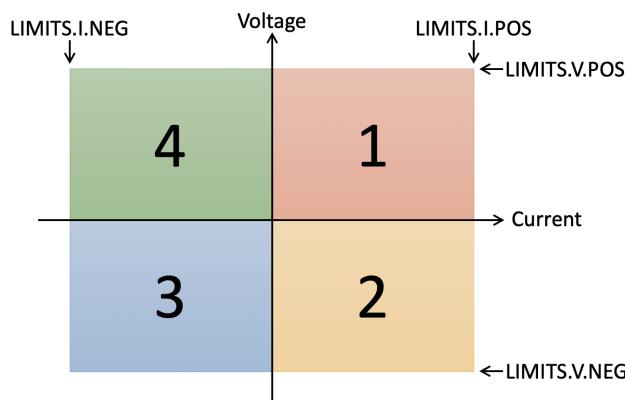


Figure 9 - Converter Quadrants

Table 4 - Voltage, Current, Field and Power limits properties.
[NL] indicates arrays of NUM_LOAD (4) elements.

Voltage Limits LIMITS.V.	Current Limits LIMITS.I.	Field Limits LIMITS.B.	Power Limits LIMITS.P.	Notes
POS[NL]	POS[NL]	POS[NL]	POS[NL]	Positive limit
NEG[NL]	NEG[NL]	NEG[NL]	NEG[NL]	Negative limit
CLOSELOOP[NL]	CLOSELOOP[NL]			Threshold for closed-loop feedback
STANDBY[NL]	STANDBY[NL]			Standby reference
RATE	RATE[NL]	RATE[NL]		Reference rate limit
	LOW[NL]	LOW[NL]		Measurement low threshold
	ZERO[NL]	ZERO[NL]		Measurement zero threshold
ERR_WARNING	ERR_WARNING[NL]	ERR_WARNING[NL]		Regulation error warning threshold
ERR_FAULT	ERR_FAULT[NL]	ERR_FAULT[NL]		Regulation error fault threshold
RATE_RMS_TC				Voltage ref rate RMS filter time constant
RATE_RMS_FAULT				Voltage ref rate RMS fault threshold
	RMS_TC			Converter current RMS filter time constant
	RMS_WARNING			Converter current RMS warning threshold
	RMS_FAULT			Converter current RMS fault threshold
	LOAD.RMS_TC[NL]			Load current RMS filter time constant
	LOAD.RMS_WARNING[NL]			Load current RMS warning threshold
	LOAD.RMS_FAULT[NL]			Load current RMS fault threshold
QUADRANTS41[2]	QUADRANTS41[2]			Voltage exclusion zones for quadrants 4 & 1
	I_MEAS_DIFF			Warning threshold for the difference between I_DCCT_A and I_DCCT_B
	HARDWARE			Set to inform software if a hardware threshold is set
	EARTH			Earth current threshold

The Class 63 software will infer the converter type from the POS and NEG limits, according to Table 5.

Table 5 - Definition of quadrants based on current and voltage limits

Converter type	Quadrants	LIMITS.I.NEG	LIMITS.I.POS	LIMITS.V.NEG	LIMITS.V.POS
1-Quadrant	1	0	> 0	0	> 0
2-Quadrant	1,2	0	> 0	< 0	> 0
4-Quadrant	1,2,3,4	< 0	> 0	< 0	> 0
1-Quadrant + polarity switch	1,3	0	> 0	0	> 0

The presence of a polarity switch is indicated by separate SWITCH properties. If a switch is present and is inverted, the 1-quadrant limits for quadrant 1 are rotated to define the limits of quadrant 3. So the current limits become [-LIMITS.I.POS, 0] and the voltage limits become [-LIMITS.V.POS, 0].

13.2 Signal availability

If a given measurement signal (current or field) is not available, then the POS and NEG limits should both be set to zero. By contrast, the voltage limits should not be set to zero, even if the voltage measurement is not available.

13.3 Measurement limits

The real-time fault limits on field, current and voltage measurements are set 1% beyond limits specified in the POS and NEG properties by the user. The standard response to exceeding a fault limit is an immediate trip of the power converter. In the case of voltage, this will only be the case if the FGC is regulating the voltage.

13.4 RMS Limits

Class 63 provides three filtered RMS limits, two for the measured current and one for the rate of change of voltage reference. In all cases, the software supports warning and fault thresholds. The warning threshold uses 10% hysteresis to avoid toggling. Setting a threshold to zero disables it.

13.4.1 Limits on the RMS current measurement

The current driven through the converter and the load will deposit heat energy in these components. The temperature of a component, such as the IGBT switches, cables or magnet coils will follow a first order response where the excitation will be proportional to the square of the current. Each component will have a temperature where damage will occur, corresponding to a steady current of a certain value. The temperature will rise with a characteristic time constant related to the thermal inertia of the component and the rate at which heat can enter or leave.

It is easy to model this temperature response using a first order filter on the square of the measured current. By defining the time constant that is a bit faster than the real response and an RMS current threshold that is a bit below the current that would cause damage, it is possible to protect the component.

This is especially important for cycling circuits where the peak current exceeds the maximum RMS current, which is very common.

There are two separate RMS current measurement limits because it may be necessary to protect the internal converter components (e.g. the IGBTs) and load components (e.g. the magnet coils). They will have very different thermal time constants and safe RMS current thresholds.

The first RMS current measurement limit is based on scalar properties:

- LIMITS.I.RMS_TC
- LIMITS.I.RMS_WARNING
- LIMITS.I.RMS_FAULT

It is there to protect the converter components, so the protection is not affected by the choice of load.

The second RMS current measurement limit is specified by array properties based on LOAD.SELECT:

- LIMITS.I.LOAD.RMS_TC[4]
- LIMITS.I.LOAD.RMS_WARNING[4]
- LIMITS.I.LOAD.RMS_FAULT[4]

The RMS current measurements for the two limits are visible in Figure 10, which shows a simulation of one supercycle containing four cycles in the CERN SPS main dipole circuit.

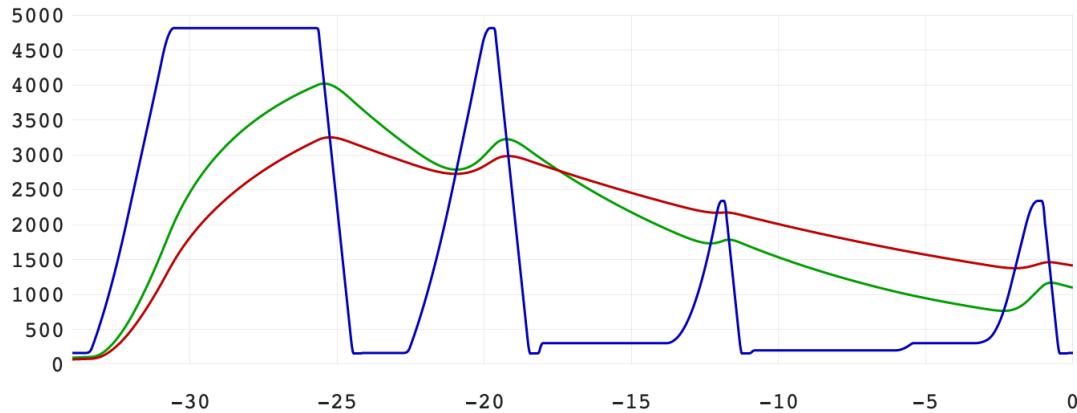


Figure 10 - I_RMS (green) and I_RMS_LOAD (red) with filter time constants of 5s and 10s

In this simulation, LIMITS.I.RMS_TC is 5s (green), while LIMITS.I.LOAD.RMS_TC (red) is 10s.

13.4.2 Limits on the RMS voltage reference rate

If the current or field regulator becomes unstable, the voltage reference may oscillate with a growing magnitude. The current or field will also oscillate, but if the inductance is very large, then this oscillation may be very small and may not exceed the regulation error fault limit. This could result in full-scale voltage oscillations that overheat the converter's output filter.

The software can detect this kind of instability using a filtered RMS of the rate of change of voltage. So the same function that provides the filtered RMS limits for the measured current can be reused to protect against regulator instability.

The default response to exceeding the RMS voltage reference rate limit is to switch off the converter, but this can be configured to execute an open-loop ramp down of the current. This can reduce the wear and tear on the converters circuit breaker.

13.5 Reference limits

The real-time fault clip limits on field, current and voltage references are set 0.1% beyond the user specified limits. The same limit is checked before allowing a reference function to be armed.

13.6 Unipolar converters

Converters that are unipolar in current (1 or 2-quadrant) usually have a non-linear voltage response at low current. This means that the software should not use feedback regulation for the current or field below a defined threshold. These thresholds can be set in the properties:

- LIMITS.B.CLOSELOOP
- LIMITS.I.CLOSELOOP

Below the threshold (which includes hysteresis to avoid toggling), the software will use a feedforward regulator and above it switches to feedback.

13.7 Recovered energy limits

If a 2 or 4-quadrant converter without energy recovery is used to drive a highly inductive load, then the stored circuit energy can be sufficient damage the output stage of the converter, if it is extracted from the circuit too quickly. This was the case for the 4-quadrant converters used in the Large Hadron Collider (LHC).

The software provides a protection by allowing the voltage to be limited as a function of the current. This is defined using two properties, each an array of two elements:

- LIMITS.I.QUADRANTS41[2]
- LIMITS.V.QUADRANTS32[2]

The limits defined for quadrants 4 and 1 are rotated and applied to quadrants 2 and 3. Examples are illustrated in Figure 11.

It is also possible to use the QUADRANTS41 limits to limit the power being injected into a circuit. This is used with the massive 1.5 GJ toroidal magnet circuit in LHC ATLAS experiment. The converter power needs to be limited in case one sub-converter is offline, so the voltage available is reduced from the nominal 18V to 16V between 18kA and 21kA. This respects the converter power limit of 328kW. In this case the QUADRANTS41 limit applies only in quadrant 1, as illustrated in Figure 12.

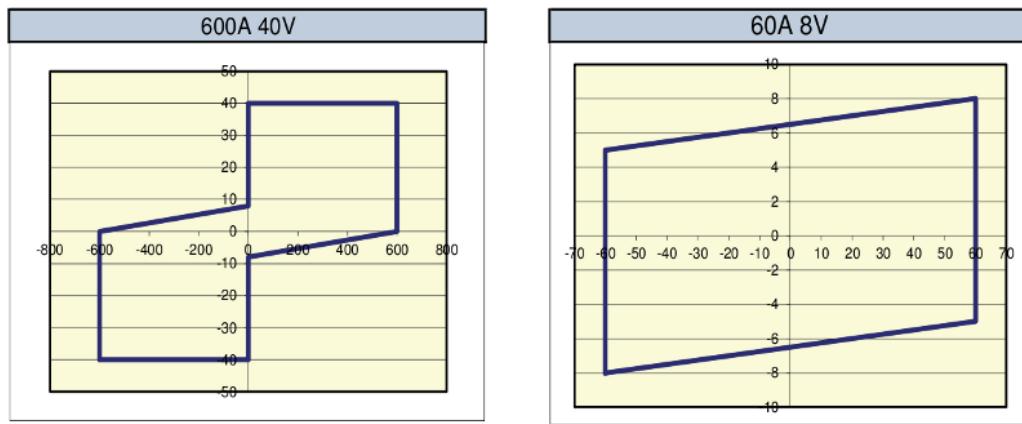
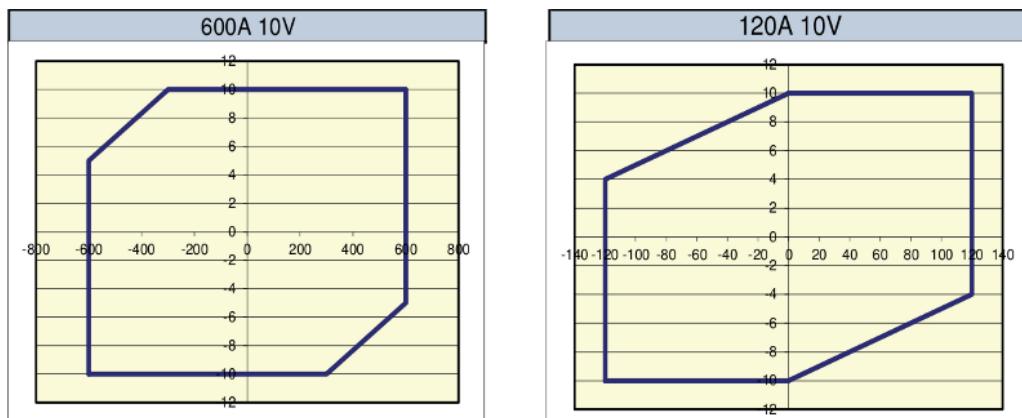


Figure 11 - Examples of QUADRANTS41 to limit recovered power from LHC circuits

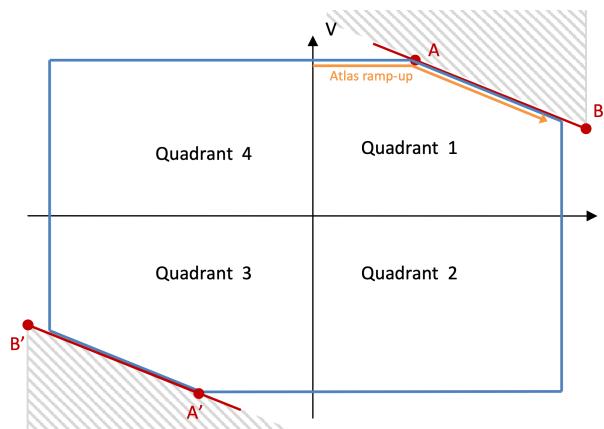


Figure 12 - Using QUADRANTS41 to limit the input power to the circuit

13.8 Power Limits

As an alternative to using the QUADRANTS41 limits, Class 63 allows the power to be limited by setting:

- LIMITS.P.POS
- LIMITS.P.NEG

This is useful for the CERN SPS accelerator main power supplies, because the main dipole power converter is rated to +/-150MW, however, the French grid cannot receive back more than 50MW.

Limits are also put on the smaller quadrupole circuits and an example is shown in the following figure.

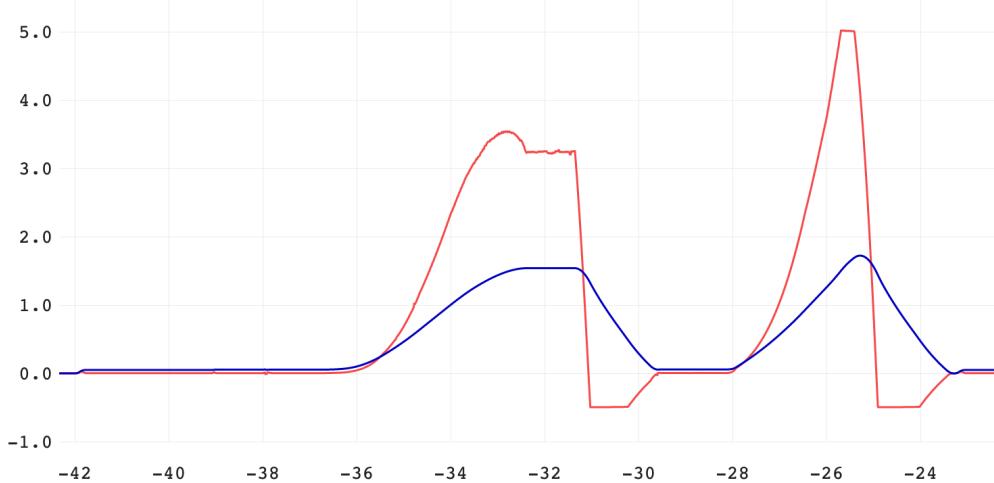


Figure 13 - Example of Power limits with an SPS main quadrupole circuit

The blue trace shows the current reference in kiloamps and the red trace shows the power in megawatts. LIMITS.P.POS was set to 5MW and LIMITS.P.NEG was set to -0.5MW. In this case, the positive limit was set just for the test, it is not needed operationally.

The power limits work by limiting the voltage dynamically and when limited, the circuit will not be able to follow the current reference. This is unimportant during the ramp down because the reference is calculated on the fly by the software and it tolerates the voltage being limited. But clipping the power on the ramp up will result in a regulation error which could result in a trip of the cycle if the error exceeds the LIMITS.I.ERR_FAULT threshold. So if a positive power limit is needed, the operator would need to change the shape of their current reference function to avoid hitting the limit.

14 Regulation

Figure 14 is a slightly simplified representation of the current regulation algorithm in Class 63, when driving an analogue voltage source using the DAC (i.e. the actuation is a voltage reference). Regulation of magnetic field is similar but magnet saturation compensation is not required and the measurement typically comes from a hall probe in the magnet.

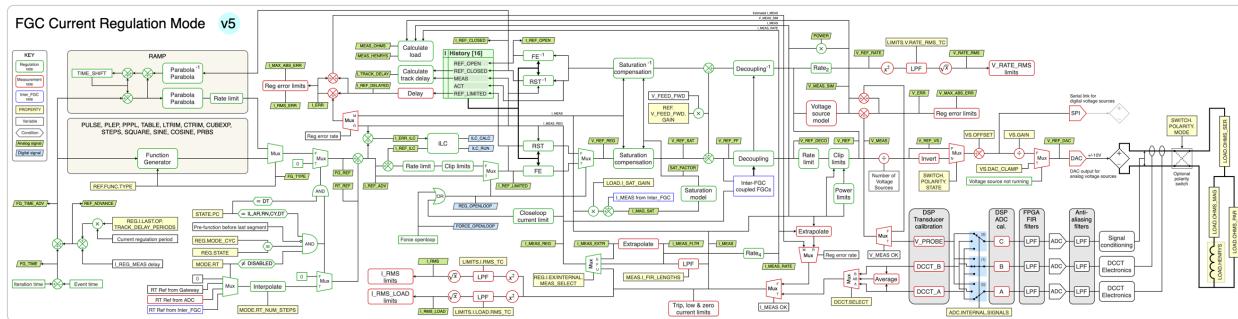


Figure 14 - Overview of current regulation with an analog voltage source

The software must choose whether to use the open-loop feedforward regulator (forward Euler) or closed-loop feedback regulator (RST). It will choose the feedforward regulator if:

- The converter is unipolar (1 or 2-quadrant) and the regulated signal (current or field) is below the CLOSELOOP limit
 - The converter is ramping down in SLOW_ABORT state with pending faults present (for example, a regulation error fault)
 - The converter is running a pre-function or post-function with OPENLOOP policy (this is beyond the scope of this guide, at the moment).

Both feedback and feedforward regulators are reversible. Normally the actuation is calculated from the reference, but if the actuation is limited, then the reference is back calculated from the limited actuation. It is this back-calculated reference that is stored in the history. This provides anti-windup protection for both feedback and feedforward regulators.

14.1 Choice of actuation reference

Class 63 supports three types of actuation reference, selected with the property VS.ACTUATION:

1. CURRENT_REF – the FGC is driving a current source and must supply the current reference. No regulation is required in the software, just function generation and limits.
 2. VOLTAGE_REF – the FGC is driving a voltage source and must regulate the current or field.
 3. FIRING_REF – the FGC is driving the firing angle and must regulate the voltage and output filter damping as well as the current or field.

VOLTAGE_REF is the most common case at CERN for switching converters. Thyristor converters use the FGC to regulate the voltage and output filter damping, so they use FIRING_REF actuation. An overview of the voltage regulation and filter damping is presented in Figure 15.

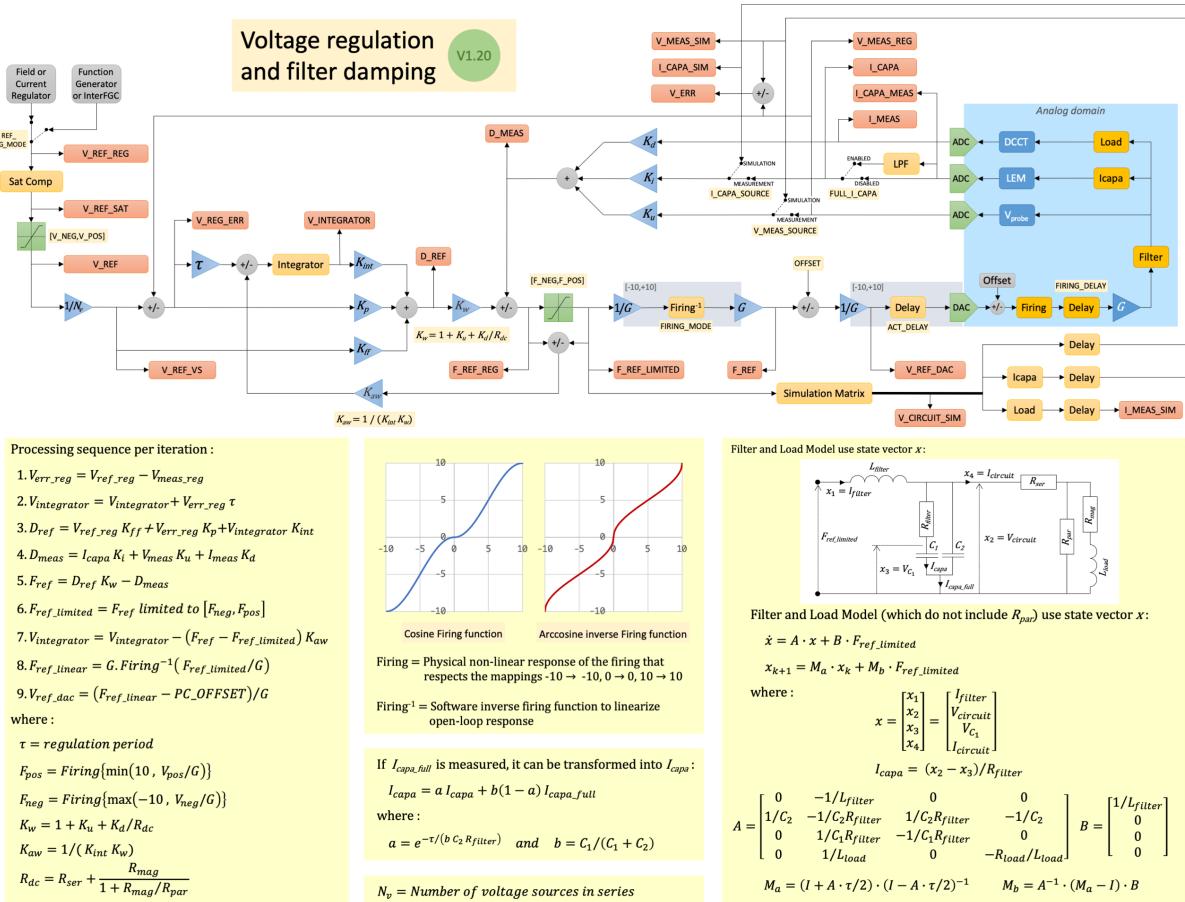


Figure 15 - Overview of voltage regulation and filter damping in Class 63

The regulation period for field and current regulators can be any integer multiple of the system's iteration period, which 100 μs for Class 63. The system iteration periods are known as "ITERS" and properties that are in units of the system iteration period have the suffix "_ITERS", for example, REG.I.PERIOD_ITERS.

Other properties are in units of the regulation period and these have the suffix "_PERIODS", for example, REG.I.INTERNAL.PURE_DELAY_PERIODS.

The regulation periods for the current and field regulators (if both are in use) may be different, unless the measurement delay for current and field are significantly different.

If a collection of circuits is being used as actuators for orbit or tune feedback, then aligning the phase of the regulators in all the circuits is essential. This is possible because the clocks in all the FGCs can be aligned to better than a microsecond using the FGC_Ether synchronization distribution system. To be

aligned, the regulation period must be chosen to divide exactly into twelve seconds. Table 6 lists all the periods that satisfy this criterion.

With all the regulation periods listed in the table, the software will ensure that the start of each UTC second where:

$$(Unix_time \% 12) = 0$$

will be a regulation iteration. So provided the feedback system also has access to synchronized Unix time, it can ensure that it sends the current reference packets timed to arrive just before each regulation period.

Table 6 - Regulation periods that can be phase-aligned because they divide exactly into 12 s

Period (ITERS)	Period (μs)	Rate (Hz)	Period (ITERS)	Period (μs)	Rate (Hz)	Period (ITERS)	Period (μs)	Rate (Hz)
1	100	10000.000	32	3200	312.500	200	20000	50.000
2	200	5000.000	40	4000	250.000	240	24000	41.667
3	300	3333.333	48	4800	208.333	250	25000	40.000
4	400	2500.000	50	5000	200.000	300	30000	33.333
5	500	2000.000	60	6000	166.667	320	32000	31.250
6	600	1666.667	64	6400	156.250	375	37500	26.667
8	800	1250.000	75	7500	133.333	400	40000	25.000
10	1000	1000.000	80	8000	125.000	480	48000	20.833
12	1200	833.333	96	9600	104.167	500	50000	20.000
15	1500	666.667	100	10000	100.000	600	60000	16.667
16	1600	625.000	120	12000	83.333	625	62500	16.000
20	2000	500.000	125	12500	80.000	750	75000	13.333
24	2400	416.667	150	15000	66.667	800	80000	12.500
25	2500	400.000	160	16000	62.500	960	96000	10.417
30	3000	333.333	192	19200	52.083	1000	100000	10.000

Table 7 – Current measurement delay for different choices of regulation measurement signal

REG.I.INTERNAL/ EXTERNAL.MEAS_SELECT	Delay	Notes
UNFILTERED	MEAS.I.DELAY_ITERS	No filtering in the FGC software. The 10 ksps acquisitions are only filtered in the FGPA.
FILTERED	MEAS.I.DELAY_ITERS + <i>current_filter_delay_iters</i>	Filter is 1 or 2 stage moving-average (boxcar) filter defined by MEAS.I.FIR_LENGTHS[2]
EXTRAPOLATED	0	Linear extrapolation of the filtered signal estimates the measurement with no delay

Table 8 – Field measurement delay for different choices of regulation measurement signal

REG.B.INTERNAL/ EXTERNAL.MEAS_SELECT	Delay	Notes
UNFILTERED	MEAS.B.DELAY_ITERS	No filtering in the FGC software. The 10 ksps acquisitions are only filtered in the FGPA.
FILTERED	MEAS.B.DELAY_ITERS + <i>field_filter_delay_iters</i>	Filter is 1 or 2 stage moving-average (boxcar) filter defined by MEAS.B.FIR_LENGTHHS[2]
EXTRAPOLATED	0	Linear extrapolation of the filtered signal estimates the measurement with no delay

14.2 Measured signal for regulation

Class 63 supports regulation of either a current or field measurement. In both cases, there is choice of three versions of the measurement that can be used for regulation. The measurement delay is different for each choice as indicated in Table 7 and Table 8.

14.2.1 Unfiltered Measurement

The most direct measurement is UNFILTERED. It is normally based on the ADC measurement of the either the DCCTs or the Hall probe(s). In both cases, it is possible to have two parallel acquisition channels (A and B) at 10 ksps. The unfiltered field or current measurement can be based on one or both of these measurements, according to the properties DCCT.SELECT and B_PROBE.SELECT. This is described in the Table 9.

Table 9 - Channel selector options for the unfiltered current and field measurements

DCCT.SELECT or B_PROBE.SELECT	Notes
A	Unfiltered measurement comes from Channel A only
B	Unfiltered measurement comes from Channel B only
AB	Unfiltered measurement is the average of Channels A and B, if both are valid. If a channel becomes invalid, then the software automatically switches to use the remaining channel.

14.2.2 Filtered Measurement

The software provides a very simple filter structure that can apply zero, one or two moving average FIR filter stages to the unfiltered field and current measurements. The filter runs at the iteration rate of 10 ksps. The lengths of the stages in iterations are defined in the two-element arrays MEAS.I.FIR_LENGTHHS and MEAS.B.FIR_LENGTHHS. Setting the length of a stage to zero or one has the effect of disabling that stage. The delay in ITERS of the current filter stages is:

$$current_filter_delay_iters = \sum_0^{i=1} \frac{(MEAS.I.FIR_LENGTHHS[i] - 1)}{2}$$

and of the field filter stages is:

$$field_filter_delay_iters = \sum_0^{i=1} \frac{(MEAS.B.FIR_LENGTHS[i] - 1)}{2}$$

Note that this formula works if the stage is disabled by setting the length to 1, but it does not work if the length is set to zero. Internally, the software converts a length of zero to be one to ensure that the delay is correctly calculated.

The filter is important for slow circuits that require high stability since it can be used to target 50Hz or 60Hz environmental noise. Each moving average filter provides a notch at the corresponding frequency and all multiples of that frequency. In some cases, residual DCCT modulator second-harmonic noise can be an issue that can also be targeted with the second moving average filter.

14.2.3 Extrapolated Measurement

The extrapolated signal removes the measurement delay but at the cost of an increase in noise. It works by simple linear extrapolation through the most recent filtered measurement and the filtered measurement from the regulation period in the past. This is run at 10 ksps. This increase in noise depends upon the degree of extrapolation, which is related to the regulation period and the measurement delay that must be overcome.

Being linear, the extrapolator cannot anticipate accelerations and deceleration of the current or field reference, so the extrapolator introduces an undershoot during accelerations and an overshoot during decelerations. It will also increase noise, even when the reference is stable. Despite this, sometimes the extrapolator is worth the cost of the extra noise as it will reduce the pure delay that the regulator will have to handle.

As an example, the next two figures show data from the FGC running the 20 kA 80 V power converter that drives the TRIUMF cyclotron main magnet.

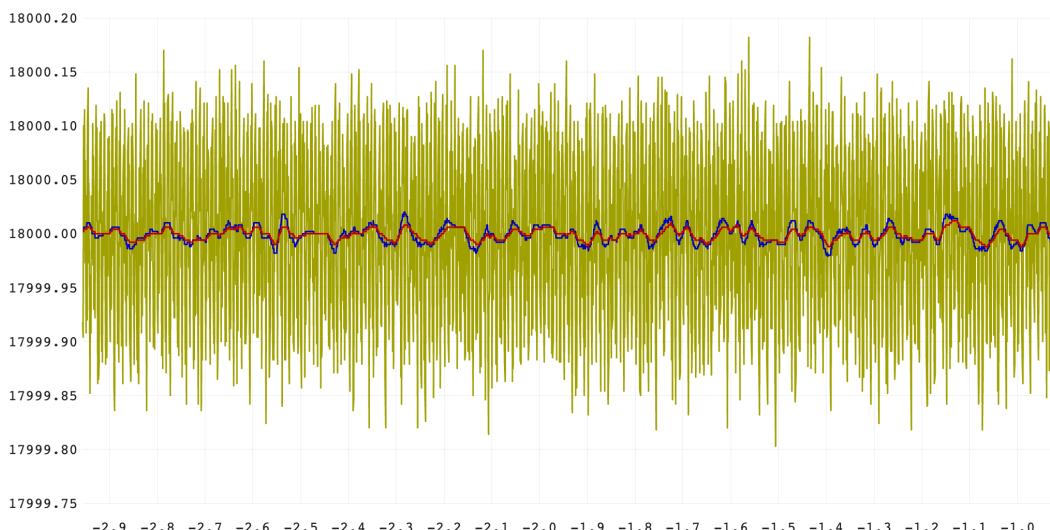


Figure 16 – Unfiltered (brown), filtered (red) and extrapolated (blue) current measurements with the TRIUMF main magnet circuit at 18 kA

Figure 16 shows a time series acquisition of the unfiltered (brown), filtered (red) and extrapolated (blue) current measurement with the TRIUMF main magnet circuit at 18 kA. Figure 17 shows the FFT of these measurements. The unfiltered signal shows big spikes at 60 Hz, 120 Hz, 240 Hz etc.. and the next largest is around 147. This is the second harmonic of the DCCT modulators. But choosing FIR lengths of 167 and 68, it was possible to put notches at multiples of 59.88Hz and 147.06 Hz. The FFT of the filtered signal (red) shows that this is very effective at eliminating these spikes. The extrapolated signal (blue) is a few dB noisier, but is still a huge improvement over the unfiltered signal and it is used for regulation.

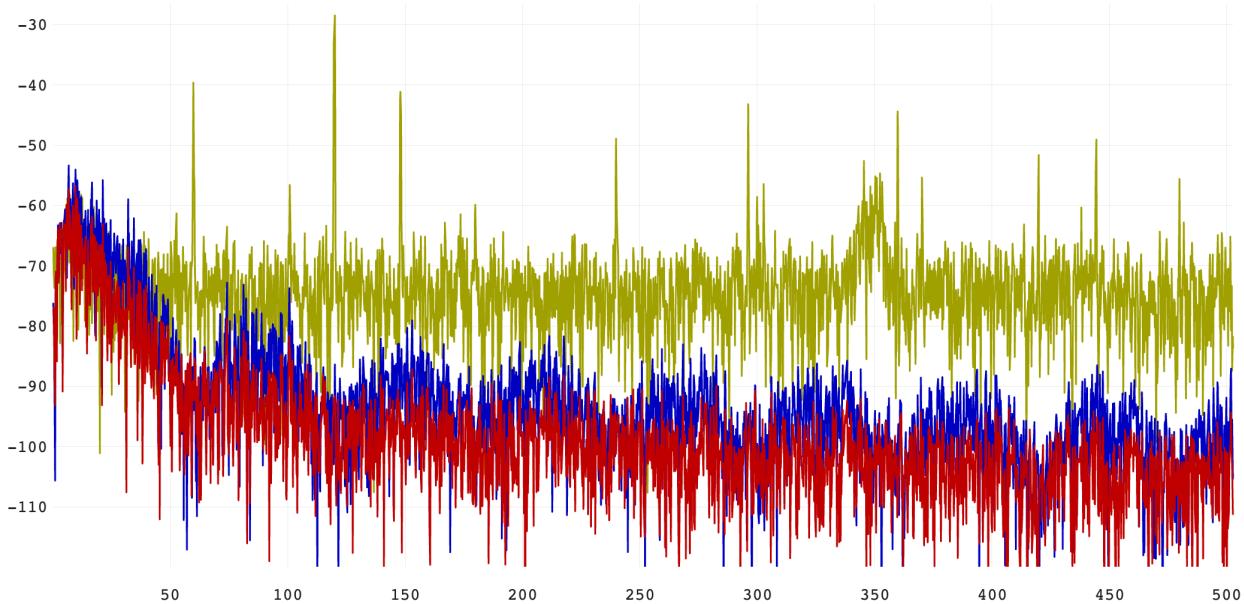


Figure 17 - FFT of the unfiltered (brown), filtered (red) and extrapolated (blue) current measurements with the TRIUMF main magnet circuit at 18 kA

The RMS and Peak-peak noise for the different signals and the regulation error is shown in the following table. The FIR filters are essential to achieving the desired long-term stability of 2 ppm (40 mA). Thanks to the filter, the short-term RMS noise is only 0.3 ppm of 20 kA.

Table 10 – Five second RMS and Peak-peak noise for the different current measurements with the TRIUMF main magnet circuit at 18 kA

Signal	RMS Noise (mA)	Peak-peak Noise (mA)
UNFILTERED	65	450
FILTERED	4	25
EXTRAPOLATED	6	40
REGULATION ERROR	6	35

14.3 Feedforward Regulator: Forward Euler

The feedforward regulator is based on solving numerically the governing differential equation for the circuit current in the load model presented in Figure 5. As a reminder, the continuous-time transfer function is:

$$G(s) = \frac{1}{R_s + \frac{1}{\frac{1}{R_p} + \frac{1}{R_m + sL_m}}}$$

The governing equation for the circuit current¹ is:

$$(R_p + R_m)V(t) + L_m \frac{dV}{dt}(t) = [R_s(R_p + R_m) + R_p R_m]I(t) + (R_p + R_s)L_m \frac{dI}{dt}(t)$$

This is solved numerically using the Forward Euler method: $(z - 1)/T \rightarrow s$

14.4 Feedback Regulator: RST

The feedback regulator is a 15th-order RST algorithm which is based on this equation:

$$\sum_0^N \{Act_i \cdot S_i\} = \sum_0^N \{Ref_i \cdot T_i\} - \sum_0^N \{Meas_i \cdot R_i\}$$

Where $i=0$ corresponds to the current sample, $i=1$ is the previous sample and so on. This notation was proposed by Landau², however in many text books the R and S polynomials are exchanged. By keeping the history of the previous N samples of the reference, measurement and actuation, it is easy to calculate the new actuation if you know the new reference and measurement:

$$Act_0 = \frac{\sum_0^N \{Ref_i \cdot T_i\} - \sum_0^N \{Meas_i \cdot R_i\} - \sum_1^N \{Act_i \cdot S_i\}}{S_0}$$

Equally, when the actuation is limited, it is equally easy to back-calculate the reference which, when combined with the new measurement will result in the limited actuation:

$$Ref_0 = \frac{\sum_0^N \{Act_i \cdot S_i\} + \sum_0^N \{Meas_i \cdot R_i\} - \sum_1^N \{Ref_i \cdot T_i\}}{T_0}$$

The benefit of the RST equation is that any linear regulator up to order N can be implemented by choosing the appropriate RST polynomial coefficients. Simple PI, PID, or PII controllers can be implemented as well as more complex higher order systems, *without changing the software*.

¹ Note that the circuit current splits between the magnet and the parallel resistor.

² I.D. Landau, "System Identification and control design", Prentice-Hall International, 1990

14.5 Internal RST coefficient synthesis algorithm

The FGC software includes an algorithm that can synthesize the RST coefficients for three different types of regulator:

Table 11 - Types of regulator than can be internally synthesized

Regulator type	Stability	Tracking	Use
PII	Limited for high performance	Excellent	Operation with any well-known loads (resistive or inductive) with fast voltage sources or modest bandwidth for the rejection of perturbations
PI	Good	Poor	Initial commissioning of poorly known resistive or fast inductive loads
I	Excellent	Terrible	Heat runs and commissioning on resistive loads

The PII is the normal choice because it provides good tracking. It is actually a collection of five algorithms that are used according to the pure delay in the closed loop. The pure delay can be entered directly in units of regulation periods, or it will be estimated by the software if you set the pure delay to zero.

To estimate the pure delay, the software starts with:

$$pure_delay_periods = \frac{vs_act_delay_iters + vs_rsp_delay_iters + meas_delay_iters}{reg_period_iters}$$

and if VS.ACTUATION is FIRING_REF then:

$$pure_delay_periods += \frac{vs_firing_delay}{reg_period}$$

where:

vs_act_delay_iters is set by the user in VS.ACT_DELAY_ITERS. It defines the pure delay between the start of a regulation iteration and the moment that the voltage starts changing.

vs_rsp_delay_iters is calculated by the software from the voltage source second order response. If VS.ACTUATION is VOLTAGE_REF, then this is known from VS.SIM.BANDWIDTH and VS.SIM.Z. If VS.ACTUATION is FIRING_REF, then it is known from REG.V.EXTERNAL.CLBW and REG.V.EXTERNAL.Z.

meas_delay_iters is calculated by the software from the voltage source second order response. If VS.ACTUATION is VOLTAGE_REF, then this is known from VS.SIM.BANDWIDTH and VS.SIM.Z. If VS.ACTUATION is FIRING_REF, then it is known from REG.V.EXTERNAL.CLBW and REG.V.EXTERNAL.Z.

Table 12 shows how the pure delay is used to select the type of algorithm. The boundary between Type 2 and 3 and between Type 4 and 5 depends on the load and the regulator parameters.

Table 12 - PII regulator synthesis algorithms based on the pure delay

Algorithm Index	Pure delay (periods)	Regulator Type	Track delay (periods)
1	0 – 0.4	Deadbeat 1 period	1
2	0.401 – ~0.9	Pseudo-deadbeat	1+pure_delay
3	~0.9 – 1.4	Deadbeat 2 period	2
4	1.401 – ~1.9	Pseudo-deadbeat	2+pure_delay
5	~1.9 – 2.4	Deadbeat 3 period	3

Note that only Algorithm 1 can be used for field regulators or for circuits with a significant parallel resistance, where significant means:

$$\frac{LOAD.OHMS_SER}{LOAD.OHMS_PAR} > 10^{-4} \text{ or } \frac{LOAD.OHMS_MAG}{LOAD.OHMS_PAR} > 10^{-4}$$

The performance of the regulator synthesized by the internal algorithms is defined by the auxiliary poles. The properties are under REG.I for the current regulator and REG.B for the field regulator. Table 13 and Table 14 present the selection of current and field regulators respectively.

14.6 External RST coefficient synthesis algorithm

The internal RST synthesis algorithms have the important limitation that they do not take into account the dynamics of the voltage source, no effects of eddy currents in the magnet yoke. If these effects are significant, then a more advanced synthesis algorithm must be run offline and the externally computed RST coefficients must be provided through separate properties.

Tools based on Matlab are used at CERN and are being ported to Python so that they can be used by other labs.

Table 13 - Current regulator selection

Current Regulation																
Operational or test RST	Internal or External RST synthesis algorithm	Regulator period	Type of regulator	Use of this type of regulator	Condition to select this type of regulator	RST synthesis algorithm	pure_delay in regulator periods <2>	track_delay in regulator periods <3>	Regulator specification	Load model	Voltage source model	Regulation measurement control	Smooth actuation control	Results from last attempt to set the RST (will be activated only if the RST ok)		
OPERATIONAL	INTERNAL REG.I. EXTERNAL_ALG [LOAD.SELECT] is DISABLED	dead-beat or pseudo dead-beat	I	Very robust control of resistive loads	LOAD.HENRYS[LOAD.SELECT] < 1E-10 & REG.I.INTERNAL.AUXPOLE2_HZ[LOAD.SELECT] == 0	20		The track delay is variable but will be assumed to be 1	REG.I.INTERNAL.AUXPOLE1_HZ[LOAD.SELECT]	LOAD.OHMS_SER[LOAD.SELECT]	VS.AC_DELAY_ITERS VS.SIM.BANDWIDTH VS.SIM.Z or VS.AC_DELAY_ITERS VS.SIM.NUM[] VS.SIM.DEN[]	REG.I. INTERNAL. MEAS_SELECT [LOAD.SELECT] and MEAS_I_DELAY_ITERS MEAS_I_FIR_LENGTHS[]	REG.I. INTERNAL. SMOOTH_ACTUATION [LOAD.SELECT]	REG.I.LAST.OP.STATUS REG.I.LAST.OP.MEAS_SELECT REG.I.LAST.OP.SMOOTH_ACTUATION REG.I.LAST.OP.RI[] REG.I.LAST.OP.SI[] REG.I.LAST.OP.TI[]		
			PI	Very robust control of fast loads	LOAD.HENRYS[LOAD.SELECT] >= 1E-10 & REG.I.INTERNAL.AUXPOLE2_HZ[LOAD.SELECT] == 0	10		The track delay is variable but will be assumed to be 1		LOAD.OHMS_SER[LOAD.SELECT] LOAD.OHMS_MAG[LOAD.SELECT] LOAD.HENRYS[LOAD.SELECT]				REG.I.LAST.OP.STATUS REG.I.LAST.OP.MEAS_SELECT REG.I.LAST.OP.SMOOTH_ACTUATION REG.I.LAST.OP.RI[] REG.I.LAST.OP.SI[] REG.I.LAST.OP.TI[]		
			Modest bandwidth and deterministic tracking with fast or slow loads without significant eddy currents	REG.I.INTERNAL.AUXPOLE2_HZ[LOAD.SELECT] > 0	1	< 0.401	1	REG.I.INTERNAL.AUXPOLE1_HZ[LOAD.SELECT] REG.I.INTERNAL.AUXPOLE2_HZ[LOAD.SELECT] REG.I.INTERNAL.AUXPOLE2_Z[LOAD.SELECT]		LOAD.OHMS_SER[LOAD.SELECT] LOAD.OHMS_MAG[LOAD.SELECT] LOAD.HENRYS[LOAD.SELECT]				REG.I.LAST.OP.STATUS REG.I.LAST.OP.MEAS_SELECT REG.I.LAST.OP.SMOOTH_ACTUATION REG.I.LAST.OP.MOD_MARGIN REG.I.LAST.PURE_DELAY_PERIODS REG.I.LAST.OP.TRACK_DELAY_PERIODS		
					2	>= 0.401 && < ~0.9	1 + pure_delay_periods			LOAD.OHMS_SER[LOAD.SELECT] LOAD.OHMS_MAG[LOAD.SELECT] LOAD.HENRYS[LOAD.SELECT]				REG.I.LAST.OP.STATUS REG.I.LAST.OP.MEAS_SELECT REG.I.LAST.OP.SMOOTH_ACTUATION REG.I.LAST.OP.MOD_MARGIN REG.I.LAST.PURE_DELAY_PERIODS REG.I.LAST.OP.TRACK_DELAY_PERIODS		
					3	>= ~0.9 && < 1.401	2			LOAD.OHMS_SER[LOAD.SELECT] LOAD.OHMS_MAG[LOAD.SELECT] LOAD.HENRYS[LOAD.SELECT]				REG.I.LAST.OP.STATUS REG.I.LAST.OP.MEAS_SELECT REG.I.LAST.OP.SMOOTH_ACTUATION REG.I.LAST.OP.MOD_MARGIN REG.I.LAST.PURE_DELAY_PERIODS REG.I.LAST.OP.TRACK_DELAY_PERIODS		
					4	>= 1.401 && < ~1.9	2 + pure_delay_periods			LOAD.OHMS_SER[LOAD.SELECT] LOAD.OHMS_MAG[LOAD.SELECT] LOAD.HENRYS[LOAD.SELECT]				REG.I.LAST.OP.STATUS REG.I.LAST.OP.MEAS_SELECT REG.I.LAST.OP.SMOOTH_ACTUATION REG.I.LAST.OP.MOD_MARGIN REG.I.LAST.PURE_DELAY_PERIODS REG.I.LAST.OP.TRACK_DELAY_PERIODS		
					5	>= ~1.9 && < 2.401	3			LOAD.OHMS_SER[LOAD.SELECT] LOAD.OHMS_MAG[LOAD.SELECT] LOAD.HENRYS[LOAD.SELECT]				REG.I.LAST.OP.STATUS REG.I.LAST.OP.MEAS_SELECT REG.I.LAST.OP.SMOOTH_ACTUATION REG.I.LAST.OP.MOD_MARGIN REG.I.LAST.PURE_DELAY_PERIODS REG.I.LAST.OP.TRACK_DELAY_PERIODS		
	EXTERNAL REG.I. EXTERNAL_ALG [LOAD.SELECT] is ENABLED	dead-beat or pseudo dead-beat	Maximum bandwidth and deterministic tracking with slow voltage sources	Option in FGCrn+ menu	model driven	Derived	Set by external algorithm in REG.I. EXTERNAL. TRACK_DELAY_PERIODS [LOAD.SELECT]	REG.I.EXTERNAL.CBW[LOAD.SELECT] REG.I.EXTERNAL.Z[LOAD.SELECT] REG.I.EXTERNAL.MOD_MARGIN[LOAD.SELECT]	LOAD.OHMS_SER[LOAD.SELECT] LOAD.OHMS_PAR[LOAD.SELECT] LOAD.OHMS_MAG[LOAD.SELECT] LOAD.HENRYS[LOAD.SELECT]	VS.AC_DELAY_ITERS VS.SIM.BANDWIDTH VS.SIM.Z or VS.AC_DELAY_ITERS VS.SIM.NUM[] VS.SIM.DEN[]	REG.I. EXTERNAL. MEAS_SELECT [LOAD.SELECT] and MEAS_I_DELAY_ITERS MEAS_I_FIR_LENGTHS[]	REG.I. EXTERNAL. SMOOTH_ACTUATION [LOAD.SELECT]	REG.I.LAST.OP.RI[] REG.I.LAST.OP.SI[] REG.I.LAST.OP.TI[]			
			Control of non-standard loads with significant eddy currents or inter-turn capacitance	Option in FGCrn+ menu	data driven from PRBS or SineFit TFA	Deduced			None - data driven from TFA results							
			Modest bandwidth and deterministic tracking with fast or slow loads without significant eddy currents	REG.I.INTERNAL.AUXPOLE2_HZ[LOAD.SELECT] > 0	1	< 0.401	1	REG.I.INTERNAL.AUXPOLE1_HZ[LOAD.TEST_SELECT] REG.I.INTERNAL.AUXPOLE2_HZ[LOAD.TEST_SELECT] REG.I.INTERNAL.AUXPOLE2_Z[LOAD.TEST_SELECT]	LOAD.OHMS_SER[LOAD.TEST_SELECT]				REG.I.LAST.TEST.STATUS REG.I.LAST.TEST.MEAS_SELECT REG.I.LAST.TEST.SMOOTH_ACTUATION REG.I.LAST.TEST.RI[] REG.I.LAST.TEST.SI[] REG.I.LAST.TEST.TI[]			
					2	>= 0.401 && < ~0.9	1 + pure_delay_periods		LOAD.OHMS_SER[LOAD.TEST_SELECT] LOAD.OHMS_MAG[LOAD.TEST_SELECT] LOAD.HENRYS[LOAD.TEST_SELECT]				REG.I.LAST.TEST.STATUS REG.I.LAST.TEST.MEAS_SELECT REG.I.LAST.TEST.SMOOTH_ACTUATION REG.I.LAST.TEST.RI[] REG.I.LAST.TEST.SI[] REG.I.LAST.TEST.TI[]			
					3	>= ~0.9 && < 1.401	2		LOAD.OHMS_SER[LOAD.TEST_SELECT] LOAD.OHMS_MAG[LOAD.TEST_SELECT] LOAD.HENRYS[LOAD.TEST_SELECT]				REG.I.LAST.TEST.STATUS REG.I.LAST.TEST.MEAS_SELECT REG.I.LAST.TEST.SMOOTH_ACTUATION REG.I.LAST.TEST.RI[] REG.I.LAST.TEST.SI[] REG.I.LAST.TEST.TI[]			
					4	>= 1.401 && < ~1.9	2 + pure_delay_periods		LOAD.OHMS_SER[LOAD.TEST_SELECT] LOAD.OHMS_MAG[LOAD.TEST_SELECT] LOAD.HENRYS[LOAD.TEST_SELECT]				REG.I.LAST.TEST.STATUS REG.I.LAST.TEST.MEAS_SELECT REG.I.LAST.TEST.SMOOTH_ACTUATION REG.I.LAST.TEST.RI[] REG.I.LAST.TEST.SI[] REG.I.LAST.TEST.TI[]			
					5	>= ~1.9 && < 2.401	3		LOAD.OHMS_SER[LOAD.TEST_SELECT] LOAD.OHMS_MAG[LOAD.TEST_SELECT] LOAD.HENRYS[LOAD.TEST_SELECT]				REG.I.LAST.TEST.STATUS REG.I.LAST.TEST.MEAS_SELECT REG.I.LAST.TEST.SMOOTH_ACTUATION REG.I.LAST.TEST.RI[] REG.I.LAST.TEST.SI[] REG.I.LAST.TEST.TI[]			
TEST	INTERNAL REG.I. EXTERNAL_ALG [LOAD.TEST_SELECT] is DISABLED	dead-beat or pseudo dead-beat	I	Very robust control of resistive loads	LOAD.HENRYS[LOAD.SELECT] < 1E-10 & REG.I.INTERNAL.AUXPOLE2_HZ[LOAD.SELECT] == 0	20		The track delay is variable but will be assumed to be 1	REG.I.INTERNAL.AUXPOLE1_HZ[LOAD.TEST_SELECT]	LOAD.OHMS_SER[LOAD.TEST_SELECT]	VS.AC_DELAY_ITERS VS.SIM.BANDWIDTH VS.SIM.Z or VS.AC_DELAY_ITERS VS.SIM.NUM[] VS.SIM.DEN[]	REG.I. INTERNAL. MEAS_SELECT [LOAD.TEST_SELECT] and MEAS_I_DELAY_ITERS MEAS_I_FIR_LENGTHS[]	REG.I. INTERNAL. SMOOTH_ACTUATION [LOAD.TEST_SELECT]	REG.I.LAST.TEST.STATUS REG.I.LAST.TEST.MEAS_SELECT REG.I.LAST.TEST.SMOOTH_ACTUATION REG.I.LAST.TEST.RI[] REG.I.LAST.TEST.SI[] REG.I.LAST.TEST.TI[]		
			PI	Very robust control of fast loads	LOAD.HENRYS[LOAD.SELECT] >= 1E-10 & REG.I.INTERNAL.AUXPOLE2_HZ[LOAD.SELECT] == 0	10		The track delay is variable but will be assumed to be 1		LOAD.OHMS_SER[LOAD.TEST_SELECT] LOAD.OHMS_MAG[LOAD.TEST_SELECT] LOAD.HENRYS[LOAD.TEST_SELECT]				REG.I.LAST.TEST.STATUS REG.I.LAST.TEST.MEAS_SELECT REG.I.LAST.TEST.SMOOTH_ACTUATION REG.I.LAST.TEST.RI[] REG.I.LAST.TEST.SI[] REG.I.LAST.TEST.TI[]		
			Modest bandwidth and deterministic tracking with fast or slow loads without significant eddy currents	REG.I.INTERNAL.AUXPOLE2_HZ[LOAD.SELECT] > 0	1	< 0.401	1	REG.I.INTERNAL.AUXPOLE1_HZ[LOAD.TEST_SELECT] REG.I.INTERNAL.AUXPOLE2_HZ[LOAD.TEST_SELECT] REG.I.INTERNAL.AUXPOLE2_Z[LOAD.TEST_SELECT]		LOAD.OHMS_SER[LOAD.TEST_SELECT] LOAD.OHMS_MAG[LOAD.TEST_SELECT] LOAD.HENRYS[LOAD.TEST_SELECT]				REG.I.LAST.TEST.STATUS REG.I.LAST.TEST.MEAS_SELECT REG.I.LAST.TEST.SMOOTH_ACTUATION REG.I.LAST.TEST.RI[] REG.I.LAST.TEST.SI[] REG.I.LAST.TEST.TI[]		
					2	>= 0.401 && < ~0.9	1 + pure_delay_periods	LOAD.OHMS_SER[LOAD.TEST_SELECT] LOAD.OHMS_MAG[LOAD.TEST_SELECT] LOAD.HENRYS[LOAD.TEST_SELECT]	REG.I.LAST.TEST.STATUS REG.I.LAST.TEST.MEAS_SELECT REG.I.LAST.TEST.SMOOTH_ACTUATION REG.I.LAST.TEST.RI[] REG.I.LAST.TEST.SI[] REG.I.LAST.TEST.TI[]							
					3	>= ~0.9 && < 1.401	2	LOAD.OHMS_SER[LOAD.TEST_SELECT] LOAD.OHMS_MAG[LOAD.TEST_SELECT] LOAD.HENRYS[LOAD.TEST_SELECT]	REG.I.LAST.TEST.STATUS REG.I.LAST.TEST.MEAS_SELECT REG.I.LAST.TEST.SMOOTH_ACTUATION REG.I.LAST.TEST.RI[] REG.I.LAST.TEST.SI[] REG.I.LAST.TEST.TI[]							
					4	>= 1.401 && < ~1.9	2 + pure_delay_periods	LOAD.OHMS_SER[LOAD.TEST_SELECT] LOAD.OHMS_MAG[LOAD.TEST_SELECT] LOAD.HENRYS[LOAD.TEST_SELECT]	REG.I.LAST.TEST.STATUS REG.I.LAST.TEST.MEAS_SELECT REG.I.LAST.TEST.SMOOTH_ACTUATION REG.I.LAST.TEST.RI[] REG.I.LAST.TEST.SI[] REG.I.LAST.TEST.TI[]							
					5	>= ~1.9 && < 2.401	3	LOAD.OHMS_SER[LOAD.TEST_SELECT] LOAD.OHMS_MAG[LOAD.TEST_SELECT] LOAD.HENRYS[LOAD.TEST_SELECT]	REG.I.LAST.TEST.STATUS REG.I.LAST.TEST.MEAS_SELECT REG.I.LAST.TEST.SMOOTH_ACTUATION REG.I.LAST.TEST.RI[] REG.I.LAST.TEST.SI[] REG.I.LAST.TEST.TI[]							
	EXTERNAL REG.I. EXTERNAL_ALG [LOAD.TEST_SELECT] is ENABLED	dead-beat or pseudo dead-beat	Maximum bandwidth and deterministic tracking with slow voltage sources	Option in FGCrn+ menu	model driven	Derived	Set by external algorithm in REG.I. EXTERNAL. TRACK_DELAY_PERIODS [LOAD.TEST_SELECT]	REG.I.EXTERNAL.CBW[LOAD.TEST_SELECT] REG.I.EXTERNAL.Z[LOAD.TEST_SELECT] REG.I.EXTERNAL.MOD_MARGIN[LOAD.TEST_SELECT]	LOAD.OHMS_SER[LOAD.TEST_SELECT] LOAD.OHMS_PAR[LOAD.TEST_SELECT] LOAD.OHMS_MAG[LOAD.TEST_SELECT] LOAD.HENRYS[LOAD.TEST_SELECT]	VS.AC_DELAY_ITERS VS.SIM.BANDWIDTH VS.SIM.Z or VS.AC_DELAY_ITERS VS.SIM.NUM[] VS.SIM.DEN[]	REG.I. EXTERNAL. MEAS_SELECT [LOAD.TEST_SELECT] and MEAS_I_DELAY_ITERS MEAS_I_FIR_LENGTHS[]	REG.I. EXTERNAL. SMOOTH_ACTUATION [LOAD.TEST_SELECT]	REG.I.LAST.TEST.RI[] REG.I.LAST.TEST.SI[] REG.I.LAST.TEST.TI[]			
			Control of non-standard loads with significant eddy currents or inter-turn capacitance	Option in FGCrn+ menu	data driven from PRBS or SineFit TFA	Deduced			None - data driven from TFA results							
			Modest bandwidth and deterministic tracking with fast or slow loads without significant eddy currents	REG.I.INTERNAL.AUXPOLE2_HZ[LOAD.SELECT] > 0	1	< 0.401	1	REG.I.INTERNAL.AUXPOLE1_HZ[LOAD.TEST_SELECT] REG.I.INTERNAL.AUXPOLE2_HZ[LOAD.TEST_SELECT] REG.I.INTERNAL.AUXPOLE2_Z[LOAD.TEST_SELECT]	LOAD.OHMS_SER[LOAD.TEST_SELECT] LOAD.OHMS_MAG[LOAD.TEST_SELECT] LOAD.HENRYS[LOAD.TEST_SELECT]				REG.I.LAST.TEST.STATUS REG.I.LAST.TEST.MEAS_SELECT REG.I.LAST.TEST.SMOOTH_ACTUATION REG.I.LAST.TEST.RI[] REG.I.LAST.TEST.SI[] REG.I.LAST.TEST.TI[]			
					2	>= 0.401 && < ~0.9	1 + pure_delay_periods		LOAD.OHMS_SER[LOAD.TEST_SELECT] LOAD.OHMS_MAG[LOAD.TEST_SELECT] LOAD.HENRYS[LOAD.TEST_SELECT]				REG.I.LAST.TEST.STATUS REG.I.LAST.TEST.MEAS_SELECT REG.I.LAST.TEST.SMOOTH_ACTUATION REG.I.LAST.TEST.RI[] REG.I.LAST.TEST.SI[] REG.I.LAST.TEST.TI[]			
					3	>= ~0.9 && < 1.401	2		LOAD.OHMS_SER[LOAD.TEST_SELECT] LOAD.OHMS_MAG[LOAD.TEST_SELECT] LOAD.HENRYS[LOAD.TEST_SELECT]				REG.I.LAST.TEST.STATUS REG.I.LAST.TEST.MEAS_SELECT REG.I.LAST.TEST.SMOOTH_ACTUATION REG.I.LAST.TEST.RI[] REG.I.LAST.TEST.SI[] REG.I.LAST.TEST.TI[]			
					4	>= 1.401 && < ~1.9	2 + pure_delay_periods		LOAD.OHMS_SER[LOAD.TEST_SELECT] LOAD.OHMS_MAG[LOAD.TEST_SELECT] LOAD.HENRYS[LOAD.TEST_SELECT]				REG.I.LAST.TEST.STATUS REG.I.LAST.TEST.MEAS_SELECT REG.I.LAST.TEST.SMOOTH_ACTUATION REG.I.LAST.TEST.RI[] REG.I.LAST.TEST.SI[] REG.I.LAST.TEST.TI[]			
					5	>= ~1.9 && < 2.401	3		LOAD.OHMS_SER[LOAD.TEST_SELECT] LOAD.OHMS_MAG[LOAD.TEST_SELECT] LOAD.HENRYS[LOAD.TEST_SELECT]				REG.I.LAST.TEST.STATUS REG.I.LAST.TEST.MEAS_SELECT REG.I.LAST.TEST.SMOOTH_ACTUATION REG.I.LAST.TEST.RI[] REG.I.LAST.TEST.SI[] REG.I.LAST.TEST.TI[]			

<1> The regulation period cannot change with the converter running, so the test RST regulator has to have the same period as the operational regulator.

<2> The pure delay can be specified in REG.I.INTERNAL.PURE_DELAY_PERIOD, but if it is zero, it will be estimated.

<3> The track delay for the internal algorithms can be specified in REG.I.INTERNAL.TRACK_DELAY_PERIODS but if it is zero, then the algorithm will compute the value as shown.

Table 14 - Field regulator selection

Field Regulation														
Operational or test RST	Internal or External RST synthesis algorithm	Regulator period	Type of regulator	Use of this type of regulator	Condition to select this type of regulator	RST synthesis algorithm	pure_delay_in regulator_periods <2>	track_delay_in regulator_periods <3>	Regulator specification	Load model	Voltage source model	Regulation measurement control	Smooth actuation control	Results from last attempt to set the RST (will be activated only if the RST ok)
OPERATIONAL	INTERNAL REG.B. EXTERNAL_ALG [LOAD.SELECT] is DISABLED	REG.B. PERIOD_ITERS [LOAD.SELECT] <1>	I	Very robust control of resistive loads	LOAD.HENRYS[LOAD.SELECT] < 1E-10 && REG.B.INTERNAL.AUXPOLES2_HZ[LOAD.SELECT] == 0	20		Variable	REG.B.INTERNAL.AUXPOLE1_HZ[LOAD.SELECT]	LOAD.OHMS_SER[LOAD.SELECT]	VS.ACT_DELAY_ITERS VS.SIM.BANDWIDTH VS.SIM.Z or VS.ACT_DELAY_ITERS VS.SIM.NUM[] VS.SIM.DEN[]	REG.B. INTERNAL MEAS_SELECT [LOAD.SELECT] and MEAS.B_DELAY_ITERS MEAS.B.FIR_LENGTHS[]	REG.B. INTERNAL SMOOTH_ACTUATION [LOAD.SELECT]	REG.B.LAST.OP.STATUS REG.B.LAST.OP.MEAS_SELECT REG.B.LAST.OP.SMOOTH_ACTUATION REG.B.LAST.OP.R[] REG.B.LAST.OP.S[] REG.B.LAST.OP.T[]
			PI	Very robust control of fast loads	LOAD.HENRYS[LOAD.SELECT] >= 1E-10 && REG.B.INTERNAL.AUXPOLES2_HZ[LOAD.SELECT] == 0	10		Variable		LOAD.OHMS_SER[LOAD.SELECT] LOAD.OHMS_MAG[LOAD.SELECT] LOAD.HENRYS[LOAD.SELECT]				
			dead-beat or pseudo dead-beat	Modest bandwidth and deterministic tracking with fast or slow loads without significant eddy currents	REG.B.INTERNAL.AUXPOLES2_HZ[LOAD.SELECT] > 0	1	< 0.401	1		LOAD.OHMS_SER[LOAD.SELECT] REG.B.INTERNAL.AUXPOLE1_HZ[LOAD.SELECT] REG.B.INTERNAL.AUXPOLES2_HZ[LOAD.SELECT]				
						2	>= 0.401 && < ~0.9	1 + pure_delay_periods		LOAD.OHMS_SER[LOAD.SELECT] LOAD.OHMS_PAR[LOAD.SELECT] <=> LOAD.OHMS_MAG[LOAD.SELECT] LOAD.HENRYS[LOAD.SELECT]				
						3	>= 0.9 && < 1.401	2		LOAD.OHMS_SER[LOAD.SELECT] LOAD.OHMS_MAG[LOAD.SELECT] LOAD.HENRYS[LOAD.SELECT]				
						4	>= 1.401 && < ~1.9	2 + pure_delay_periods		LOAD.OHMS_SER[LOAD.SELECT] REG.B.INTERNAL.AUXPOLE1_HZ[LOAD.SELECT] REG.B.INTERNAL.AUXPOLES2_HZ[LOAD.SELECT]				
						5	>= ~1.9 && < 2.401	3		LOAD.OHMS_SER[LOAD.SELECT] REG.B.INTERNAL.AUXPOLE1_HZ[LOAD.SELECT] REG.B.INTERNAL.AUXPOLES2_HZ[LOAD.SELECT] REG.B.INTERNAL.AUXPOLES2_Z[LOAD.SELECT]				
	EXTERNAL REG.B. EXTERNAL_ALG [LOAD.SELECT] is ENABLED	REG.B. PERIOD_ITERS [LOAD.SELECT] <1>	dead-beat or pseudo dead-beat	Maximum bandwidth and deterministic tracking with slow voltage sources	Option in FGCrn+ menu	model driven	Derived	Set by external algorithm in REG.B. EXTERNAL TRACK_DELAY_PERIODS [LOAD.SELECT]	REG.B.EXTERNAL.CLBW[LOAD.SELECT] REG.B.EXTERNAL.Z[LOAD.SELECT] REG.B.EXTERNAL.MOD_MARGIN[LOAD.SELECT]	LOAD.OHMS_SER[LOAD.SELECT] LOAD.OHMS_PAR[LOAD.SELECT] LOAD.OHMS_MAG[LOAD.SELECT] LOAD.HENRYS[LOAD.SELECT]	VS.ACT_DELAY_ITERS VS.SIM.BANDWIDTH VS.SIM.Z	REG.B. EXTERNAL MEAS_SELECT [LOAD.SELECT] and MEAS.B_DELAY_ITERS MEAS.B.FIR_LENGTHS[]	REG.B. EXTERNAL SMOOTH_ACTUATION [LOAD.SELECT]	REG.B.LAST.OP.STATUS REG.B.LAST.OP.MEAS_SELECT REG.B.LAST.OP.SMOOTH_ACTUATION REG.B.LAST.OP.MOD_MARGIN REG.B.LAST.OP.PURE_DELAY_PERIODS REG.B.LAST.OP.TRACK_DELAY_PERIODS REG.B.LAST.OP.R[] REG.B.LAST.OP.S[] REG.B.LAST.OP.T[]
										None - data driven from TFA results				
			dead-beat or pseudo dead-beat	Control of non-standard loads with significant eddy currents or inter-turn capacitance	Option in FGCrn+ menu	data driven from PRBS or SineFit TFA	Deduced	Set by external algorithm in REG.B. EXTERNAL TRACK_DELAY_PERIODS [LOAD.SELECT]	REG.B.EXTERNAL.CLBW[LOAD.SELECT] REG.B.EXTERNAL.Z[LOAD.SELECT] REG.B.EXTERNAL.MOD_MARGIN[LOAD.SELECT]	LOAD.OHMS_SER[LOAD.SELECT] LOAD.OHMS_PAR[LOAD.SELECT] LOAD.OHMS_MAG[LOAD.SELECT] LOAD.HENRYS[LOAD.SELECT]	VS.ACT_DELAY_ITERS VS.SIM.BANDWIDTH VS.SIM.Z or VS.ACT_DELAY_ITERS VS.SIM.NUM[] VS.SIM.DEN[]	REG.B. INTERNAL MEAS_SELECT [LOAD.SELECT] and MEAS.B_DELAY_ITERS MEAS.B.FIR_LENGTHS[]	REG.B. INTERNAL SMOOTH_ACTUATION [LOAD.SELECT]	REG.B.LAST.OP.STATUS REG.B.LAST.OP.MEAS_SELECT REG.B.LAST.OP.SMOOTH_ACTUATION REG.B.LAST.OP.MOD_MARGIN REG.B.LAST.OP.PURE_DELAY_PERIODS REG.B.LAST.OP.TRACK_DELAY_PERIODS REG.B.LAST.OP.R[] REG.B.LAST.OP.S[] REG.B.LAST.OP.T[]
										None - data driven from TFA results				
										None - data driven from TFA results				
										None - data driven from TFA results				
TEST	INTERNAL REG.B. EXTERNAL_TEST_SELECT [LOAD.TEST_SELECT] is DISABLED	REG.B. PERIOD_ITERS [LOAD.TEST_SELECT] <1>	I	Very robust control of resistive loads	LOAD.HENRYS[LOAD.SELECT] < 1E-10 && REG.B.INTERNAL.AUXPOLES2_HZ[LOAD.SELECT] == 0	20		Variable	REG.B.INTERNAL.AUXPOLE1_HZ[LOAD.TEST_SELECT]	LOAD.OHMS_SER[LOAD.TEST_SELECT]	VS.ACT_DELAY_ITERS VS.SIM.BANDWIDTH VS.SIM.Z or VS.ACT_DELAY_ITERS VS.SIM.NUM[] VS.SIM.DEN[]	REG.B. INTERNAL MEAS_SELECT [LOAD.TEST_SELECT] and MEAS.B_DELAY_ITERS MEAS.B.FIR_LENGTHS[]	REG.B. INTERNAL SMOOTH_ACTUATION [LOAD.TEST_SELECT]	REG.B.LAST.TEST.STATUS REG.B.LAST.TEST.MEAS_SELECT REG.B.LAST.TEST.SMOOTH_ACTUATION REG.B.LAST.TEST.R[] REG.B.LAST.TEST.S[] REG.B.LAST.TEST.T[]
			PI	Very robust control of fast loads	LOAD.HENRYS[LOAD.SELECT] >= 1E-10 && REG.B.INTERNAL.AUXPOLES2_HZ[LOAD.SELECT] == 0	10		Variable		LOAD.OHMS_SER[LOAD.TEST_SELECT] LOAD.OHMS_PAR[LOAD.TEST_SELECT] LOAD.OHMS_MAG[LOAD.TEST_SELECT] LOAD.HENRYS[LOAD.TEST_SELECT]				
			dead-beat or pseudo dead-beat	Modest bandwidth and deterministic tracking with fast or slow loads without significant eddy currents	REG.B.INTERNAL.AUXPOLES2_HZ[LOAD.SELECT] > 0	1	< 0.401	1		LOAD.OHMS_SER[LOAD.TEST_SELECT] REG.B.INTERNAL.AUXPOLE1_HZ[LOAD.TEST_SELECT] REG.B.INTERNAL.AUXPOLES2_HZ[LOAD.TEST_SELECT] REG.B.INTERNAL.AUXPOLES1_Z[LOAD.TEST_SELECT]				
						2	>= 0.401 && < ~0.9	1 + pure_delay_periods		LOAD.OHMS_SER[LOAD.TEST_SELECT] LOAD.OHMS_PAR[LOAD.TEST_SELECT] <=> LOAD.OHMS_MAG[LOAD.TEST_SELECT] LOAD.HENRYS[LOAD.TEST_SELECT]				
						3	>= 0.9 && < 1.401	2		LOAD.OHMS_SER[LOAD.TEST_SELECT] LOAD.OHMS_PAR[LOAD.TEST_SELECT] <=> LOAD.OHMS_MAG[LOAD.TEST_SELECT] LOAD.HENRYS[LOAD.TEST_SELECT]				
						4	>= 1.401 && < ~1.9	2 + pure_delay_periods		LOAD.OHMS_SER[LOAD.TEST_SELECT] LOAD.OHMS_PAR[LOAD.TEST_SELECT] <=> LOAD.OHMS_MAG[LOAD.TEST_SELECT] LOAD.HENRYS[LOAD.TEST_SELECT]				
						5	>= ~1.9 && < 2.401	3		LOAD.OHMS_SER[LOAD.TEST_SELECT] LOAD.OHMS_PAR[LOAD.TEST_SELECT] <=> LOAD.OHMS_MAG[LOAD.TEST_SELECT] LOAD.HENRYS[LOAD.TEST_SELECT]				
	EXTERNAL REG.B. EXTERNAL_ALG [LOAD.TEST_SELECT] is ENABLED	REG.B. PERIOD_ITERS [LOAD.TEST_SELECT] <1>	dead-beat or pseudo dead-beat	Maximum bandwidth and deterministic tracking with slow voltage sources	Option in FGCrn+ menu	model driven	Derived	Set by external algorithm in REG.B. EXTERNAL TRACK_DELAY_PERIODS [LOAD.TEST_SELECT]	REG.B.EXTERNAL.CLBW[LOAD.TEST_SELECT] REG.B.EXTERNAL.Z[LOAD.TEST_SELECT] REG.B.EXTERNAL.MOD_MARGIN[LOAD.TEST_SELECT]	LOAD.OHMS_SER[LOAD.TEST_SELECT] LOAD.OHMS_PAR[LOAD.TEST_SELECT] LOAD.OHMS_MAG[LOAD.TEST_SELECT] LOAD.HENRYS[LOAD.TEST_SELECT]	VS.ACT_DELAY_ITERS VS.SIM.BANDWIDTH VS.SIM.Z	REG.B. EXTERNAL MEAS_SELECT [LOAD.TEST_SELECT] and MEAS.B_DELAY_ITERS MEAS.B.FIR_LENGTHS[]	REG.B. EXTERNAL SMOOTH_ACTUATION [LOAD.TEST_SELECT]	REG.B.LAST.TEST.STATUS REG.B.LAST.TEST.MEAS_SELECT REG.B.LAST.TEST.SMOOTH_ACTUATION REG.B.LAST.TEST.R[] REG.B.LAST.TEST.S[] REG.B.LAST.TEST.T[]
										None - data driven from TFA results				
										None - data driven from TFA results				
										None - data driven from TFA results				
										None - data driven from TFA results				

<1> The regulation period cannot change with the converter running, so the test RST regulator has to have the same period as the operational regulator.

<2> The pure delay can be specified in REG.I.INTERNAL.PURE_DELAY_PERIOD, but if it is zero, it will be estimated.

<3> The track delay for the internal algorithms can be specified in REG.I.INTERNAL.TRACK_DELAY_PERIODS but if it is zero, then the algorithm will compute the value as shown.

<4> The track delay for the internal algorithms can be specified in REG.I.INTERNAL.TRACK_DELAY_PERIODS but if it is zero, then the algorithm will compute the value as shown.

14.7 Jury's test and modulus margin

Whether the RST coefficients are calculated internally or externally, the software will check them to see if the controller is likely to be unstable. The results are reported in the parent properties:

- REG.B.LAST.OP – Last update to the operational field regulator
- REG.B.LAST.TEST – Last update to the test field regulator
- REG.I.LAST.OP – Last update to the operational current regulator
- REG.I.LAST.TEST – Last update to the test current regulator

In each case, the child properties are listed in Table 15.

Table 15 – REG.B/I.LAST.OP/TEST properties

Property	Notes
STATUS	Status of the regulator
MEAS_SELECT	UNFILTERED, FILTERED or EXTRAPOLATED
MOD_MARGIN	Modulus margin – below 0.4 will result in a warning
PURE_DELAY_PERIODS	Pure delay in regulation periods
TRACK_DELAY_PERIODS	Track delay in regulation periods
R	R coefficients
S	S coefficients
T	T coefficients
A	Load numerator coefficients (internal algorithm only)
B	Load denominator coefficients (internal algorithm only)
ASBR	A.S+B.R coefficients (internal algorithm only)

The REG.B/I.LAST.OP/TEST.STATUS properties will report status according to Table 16.

Table 16 – RST Status

OK	Regulator is OK
LOW_MOD_MARGN	Warning: Low modulus margin (< 0.4)
OHMS_PAR_SMAL	Fault: The parallel resistance is too small.
PURE_DLY_BIG	Fault: Pure delay is too big
R0_IS_ZERO	Fault: $ R[0] $ is less than 10^{-7}
S0_NOT_POS	Fault: $S[0]$ is less than 10^{-7}
T0_NOT_POS	Fault: $T[0]$ is less than 10^{-7}
SUM_S_IS_NEG	Fault: Sum(S) is negative
SUM_S_ERROR	Fault: Sum(Even S) less than Sum(Odd S)
S_UNSTBL_POLE	Fault: Unstable pole in S (Jury's test)

15 FGC Power Converter State Machine

Class 63 supports the power converter state machine presented in Table 17. It is visible in the read-only STATE.PC property which is also included in the POLL property. The user can set the desired state in the property MODE.PC. This is a general policy in the FGC software. A STATE property reports the actual state, while a MODE property allows the users to set their desired state.

The power converter state machine has many states, but not all are used with a DC converter that is not cycling under the control of a timing system. The states and transitions are illustrated in Figure 18.

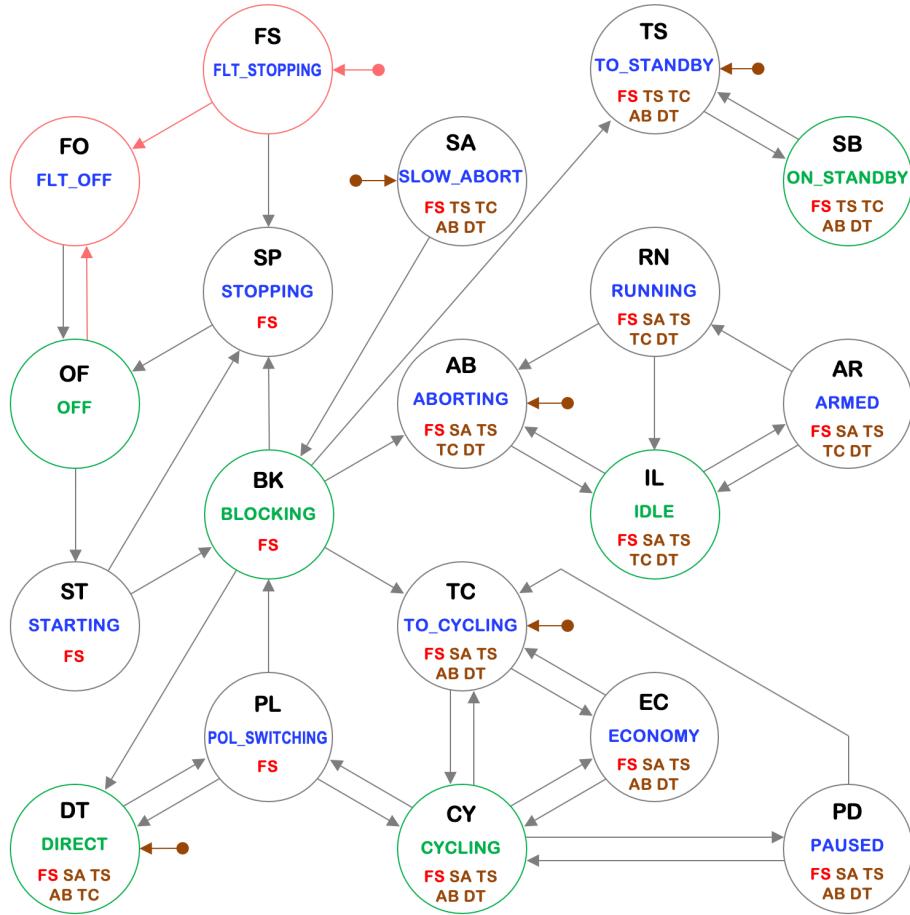


Figure 18 - Power Converter State Machine

Arrows between states indicate transitions which have transition conditions. All the conditions for the transitions leaving the current state are evaluated every 5 ms in priority order, to decide if the state will change. There are six states with “anonymous” entry transitions, for example, XXtoFS leads to FLT_STOPPING from fifteen other states. Anonymous transitions are indicated by a dot and an arrow entering the state ($\bullet \rightarrow$).

If a state can transition via an anonymous transition to another state, then the abbreviated name of the target state is included inside the state’s circle in the state diagram. For example, for the IDLE state,

there are four anonymous transition, XXtoFS, XXtoSA, XXtoDT and XXtoTS, leading to FLT_STOPPING, SLOW_ABORT, DIRECT and TO_STANDBY respectively.

Table 17 - Power Converter State Machine

	State	Abbreviation	Notes
0	FLT_OFF	FO	Converter is not powered and at least one fault was latched.
1	OFF	OF	Converter is not powered and no faults are latched. It can be started by setting MODE.PC to ON_STANDBY, IDLE or DIRECT.
2	FLT_STOPPING	FO	Converter is shutting down and at least one fault was latched.
3	STOPPING	SP	Converter is shutting down and no faults are latched. This follows a command to set MODE.PC to OFF.
4	STARTING	ST	Converter is starting. It waits in this state until the start-up is complete and the voltage regulation is running.
5	SLOW_ABORT	SA	Converter is ramping down the current before turning off. This can be following a user command to set MODE.PC to SLOW_ABORT or OFF, or the removal of the PC_PERMIT interlock signal.
6	TO_STANDBY	TS	Converter is ramping to the LIMITS.I.MIN current level, using the default acceleration and rate of change.
7	ON_STANDBY	SB	The converter is running with the reference locked at the LIMITS.I.MIN current level.
8	IDLE	IL	The converter is running with the reference holding the last value. It is possible to arm a change in the reference value by setting the REF property.
9	TO_CYCLING	TC	Not needed for the TRIUMF converter.
10	ARMED	AR	The converter is running with the reference holding the last value and is ready to run a reference change that was previously armed.
11	RUNNING	RN	The converter is running a change in reference using the previously armed function.
12	ABORTING	AB	The converter is smoothly slowing down a reference change before the end of the function. It decelerates using a parabolic function of time.
13	CYCLING	CY	The converter is responding to timing events to know the timing and user number of the reference function to play on each cycle.
14	POL_SWITCHING	PS	The polarity switch is changing position.
15	BLOCKING	BK	The converter is powered, but the output is blocked.
16	ECONOMY	EC	The converter is saving energy in the case where no beam is present.
17	DIRECT	DT	The converter will ramp immediately to the reference value.

More information is included in the web page on the FGC web site on the KT-FGC cernbox at:

FGC_Website/fgc/gendoc/def/SymListPC.htm

A symbol list is like an enumerated type, linking text labels to integers.

16 Configuration Properties

A subset of properties are stored in non-volatile memory. They are known as configuration properties, and they define the characteristics of the power converter, the load, the regulator and the scaling and calibration errors for the analog measurement devices (DCCTs, ADCs, voltage references).

Three properties can help with managing the configuration properties:

Table 18 - Configuration management properties

Property	Command	Behavior
CONFIG.SET	G	Lists all configuration properties and their values.
	S	Cancel a previous set of CONFIG.UNSET.
CONFIG.UNSET	G	Lists all configuration properties that were never set. If STATE.OP is UNCONFIGURED, then at least one property will be in this list. Once the last configuration property is set, the list will be empty and STATE.OP can leave UNCONFIGURED.
	S	Resets a flag so that after the next reset of the FGC3, the boot program will erase all configuration properties values. Once the main program starts, STATE.OP will be UNCONFIGURED. This flag can be cleared by setting CONFIG.SET.

A file can contain the set commands to set all the configuration properties. The FGC can be configured using teraterm4 by dragging and dropping the configuration file onto the teraterm4 window.

It is a good idea to have separate files to configure the different analog components, so the correct calibration errors are not lost or overwritten by mistake.

In an operational system, a configuration manager program runs all the time and watches for FGCs that need to be configured automatically following a reset, or on demand by a user. At the time of writing, this program is written in Perl and only works within CERN, but it will be ported to python and adapted to run at other labs.

Configuration properties are divided into three groups:

Table 19 - Types of configuration properties

Type of configuration property	Linked to	Examples
SYSTEM	Name of the FGC Device	Circuit parameters (resistance, inductance), regulation parameters, limits
EQUIPMENT	Complete barcode of the component	Calibration errors of the ADCs on an analogue interface card.
TYPE	Type of a component, identified by the first part of the barcode	Scaling of DCCT heads

17 Reference Generator

In this section, example commands are shown for terminal editor mode in upper case. As a reminder, commands, property names and property values are not case sensitive. To use a command in terminal direct mode, prefix the start of command delimiter (!) and end with a linefeed (\n). For example:

```
S MODE.PC IDLE
```

in direct mode becomes:

```
!S MODE.PC IDLE\n
```

17.1 Regulation mode

The most common regulation mode is current. When REG.MODE is I, the reference generator value will be treated as a current reference in amps and the current regulator will calculate the voltage reference based on the current reference and the measured current.

During converter commissioning or tests, it may be useful sometimes to directly set the voltage reference. This is possible by changing the regulation mode from current to voltage using the property REG.MODE:

S REG.MODE V	<i>Switch regulation mode to voltage</i>
G REG.STATE	<i>Read back the regulation state</i>
S REG.MODE I	<i>Switch regulation mode to current</i>
G REG.STATE	<i>Read back the regulation state</i>

When REG.STATE is V, the reference generator value will be treated as a voltage reference in volts.

Note that after a reset of the FGC3, REG.MODE will be initialized to the value in the configuration property called REG.MODE_CYC, so this will normally be set to "I".

17.2 DIRECT power converter state

When a power converter operates with a DC reference, DIRECT power converter state is the most useful (note, this has nothing to do with direct mode for the terminal communications). In DIRECT state, the required current is set using the REF.DIRECT.I.VALUE property or it can be supplied from an external source, such as the orbit feedback system. When REF.DIRECT.I.VALUE is being used, the FGC3 software is responsible for smoothly ramping the current to the new reference value, while respecting the current, voltage and power limits. While ramping, STATE.PC remains DIRECT and REF.DIRECT.V.VALUE can be set at any time, even while the current is changing.

For example:

S MODE.PC OFF	<i>Send a reset to clear any faults</i>
G POLL	<i>Get POLL to check states and faults</i>
S REF.DIRECT.I.VALUE 10	<i>Set reference to initial current level</i>
S MODE.PC DIRECT	<i>Start converter into DIRECT state</i>

G POLL	<i>Get POLL to check states and status</i>
S REF.DIRECT.I.VALUE 20	<i>Ramp to 20A</i>
G POLL	<i>Get POLL to check states and status</i>
S MODE.PC OFF	<i>Ramp down in SLOW_ABORT and then stop</i>

When using editor mode, G POLL isn't needed because the states and status of the converter are visible in the bottom four lines of the terminal window. In direct mode, the controlling program can repeatedly get POLL to update its knowledge about the states and status of the converter.

If the converter is operated in VOLTAGE regulation mode in DIRECT state, then REF.DIRECT.V.VALUE should be set to ramp the voltage reference.

The acceleration, deceleration and linear rate will be taken from the default configuration properties:

Property	Behavior
REF.DEFAULTS.I.ACCELERATION	Default current acceleration (A/s ²)
REF.DEFAULTS.I.DECELERATION	Default current deceleration (A/s ²)
REF.DEFAULTS.I.LINEAR_RATE	Maximum rate of change during current ramp (A/s)
REF.DEFAULTS.V.ACCELERATION	Default voltage acceleration (V/s ²)
REF.DEFAULTS.V.DECELERATION	Default voltage deceleration (V/s ²)
REF.DEFAULTS.V.LINEAR_RATE	Maximum rate of change during voltage ramp (V/s)

In terminal editor mode only, if a number is entered in DIRECT state, then the command "S REF.DIRECT.I.VALUE" or "S REF.DIRECT.V.VALUE" will be added as a prefix, according to the regulation mode. So it is sufficient to enter the current for the FGC3 to ramp to that current (or voltage if in voltage regulation mode).

17.3 IDLE power converter state

When commissioning the converter, it may be better to operate in IDLE/ARMED/RUNNINGS states. In these states, the current can be ramped using the REF property:

S MODE.PC OFF	<i>Send a reset to clear any faults</i>
G POLL	<i>Get POLL to check states and faults</i>
S MODE.PC IDLE	<i>Start converter into IDLE state</i>
G POLL	<i>Get POLL to wait for IDLE state</i>
S REF NOW,15	<i>Arm and run a ramp to 15A</i>
G POLL	<i>Get POLL to wait for ramp to return to IDLE</i>
S MODE.PC OFF	<i>Ramp down in SLOW_ABORT and then stop</i>

S REF NOW,{current} will try to arm a ramp to the requested current level and if successful, it will start the ramp after a delay of 1 second. The state will go IDLE to ARMED to RUNNING and then back to IDLE, once the reference arrives at the final value.

Note that it is only possible to arm a new reference change in IDLE state, so if S REF NOW is executed in ARMED or RUNNING, the request will be rejected.

As with DIRECT state, the acceleration/deceleration and linear rate will be taken from the default configuration properties. The default values can be overruled by adding parameters after the final reference value. For example:

S REF NOW,-20,1	<i>Arm and run a ramp to -20 A with accelerator/deceleration of 1 A/s²</i>
S REF NOW,15,,2	<i>Arm and run a ramp to 15A with max Linear rate of 2 A/s and default acceleration/deceleration</i>
S REF NOW,20,2,5	<i>Arm and run a ramp to 20 A with max linear rate of 5 A/s and acceleration/deceleration of 2 A/s²</i>

In terminal editor mode only, if a number is entered in IDLE state, then the command “S REF NOW,” will be added as a prefix. So it is sufficient to enter the current for the FGC3 to ramp to that current (or voltage if in voltage regulation mode):

20	<i>Arm and run a ramp to 20 A with default accelerator/deceleration and Linear rate</i>
20,1	<i>Arm and run a ramp to 20 A with accelerator/deceleration of 1 A/s²</i>
20,,10	<i>Arm and run a ramp to 20 A with max Linear rate of 10 A/s and default acceleration/deceleration</i>
20,5,50	<i>Arm and run a ramp to 20 A with max linear rate of 50 A/s and acceleration/deceleration of 5 A/s²</i>

In IDLE state, it is also possible to arm other types of functions. Some are useful for evaluating the performance of the converter. For example:

```
S REF SINE,{amplitude},{num_cycles},{period}
```

Only “S REF NOW” will arm *and* start the function. All other functions will be armed only. A second command is needed to change state to RUNNING, where the function will be executed. For example:

S REF SINE,2,10,3	<i>Arm a sine wave of +/-1 Amp, for 10 cycles of 3 seconds</i>
G REF.INFO	<i>Read back the meta data for the armed function</i>
S REF.RUN	<i>Run the armed function</i>

“S REF.RUN” without a parameter will start the function immediately. If a parameter is supplied, it can specify the Unix time when the function should start running.

It is also possible to set the function's properties individually, and then arm the function using REF.FUNC.TYPE. For example, the same function as given above can be armed as follows:

S REF.TEST.AMPLITUDE 2	<i>Set the amplitude to 2 Amps peak-peak</i>
S REF.TEST.NUM_CYCLES 10	Set number of cycles to 10
S REF.TEST.PERIOD 3	Set period to 3 seconds
S REF.FUNC.TYPE SINE	Try to arm the sine wave
G REF.INFO	Read back the meta data for the armed function
S REF.RUN	Run the armed function

Three other test functions use the same REF.TEST properties: COSINE, STEPS and SQUARE. Consult the web site for more details. Two other TEST properties are significant for SINE and COSINE (but are ignored for STEPS and SQUARE). These are TEST.WINDOW and TEST.EXP_DECAY.