

4 Configuration

The "real" user interface is provided by the LCI configuration XML files. These files are compiled into binary configuration files for upload to the LAT using the `LCI_parser` executable. The ID of this binary file is one of the parameters to the `calibrate` command. A configuration file may contain multiple configurations, which are executed in succession. The first such configuration establishes a base and should specify all the available calibration elements. The common elements must be specified, as no defaults are taken. The sub-system specific elements may be omitted, and default to the values established by the LAT configuration. Subsequent configurations need only specify incremental changes from the previous configuration, and can be null.

The parser takes two or three arguments: an optional string identifying the subsystem configuration being specified ("ACD", "CAL" or "TKR"), the XML file name, and the output binary file name.

The contents of a binary configuration file may be displayed using the `LCI_report` executable. It takes one or two arguments: the name of the binary file and the optional name of the output file. If the latter is not specified, the output is directed to the terminal.

4.0 XML File Structure

It is recommended, but not required, that the first two lines of a configuration XML file be an XML declaration followed by a document type declaration. These lines have the form:

```
<?xml version='1.0' standalone='yes' ?>
<!DOCTYPE LCI_XXX_XML SYSTEM
"/afs/slac/g/glast/flight/ICS/source/LCI/prod/LCI/xxx.dtd">
```

in which "XXX" and "xxx" are to be replaced by the name of the subsystem being calibrated.

The last part of the document type declaration is the name of the document type definition (DTD) file defining the structure of the XML file. In this case it is not actually used by the parser, but is included for documentation purposes.

The document type name, "LCI_XXX_XML", normally specifies the subsystem being calibrated, but if the subsystem name is supplied to the `LCI_parser` command, it has precedence, and the name in the document type declaration, if present, must match.

The remainder of an XML file is enclosed within the root tag pair, whose name must be the same as the document type, i.e. "LCI_ACD_XML", "LCI_CAL_XML", or "LCI_TKR_XML".

Comments may be placed anywhere within an XML file. These are delimited by the strings "`<!--`" and "`-->`" (note the presence of the space in each).

4.1 XML Vocabulary

Within the root element, there are one or more configurations, each contained in a `<configuration>` element. As previously stated, the first configuration must contain all the common elements, and it is good practice to specify all the elements pertaining to the given calibration type.

Element (tag) names must be supplied exactly as shown.

Element values are either numbers or keywords. Numbers may be specified in either decimal or case-insensitive hexadecimal (leading "0x") notation. Keywords are also case-insensitive. With one exception, all numbers used must be unsigned. The parser checks that each value is either a recognized keyword or a valid number, and such a number must lie within an acceptable range.

Many of the quantities specified in the configuration can be iterated over as indicated by the *iterate* comment next to the tag in the description below. In this case the XML element should contain either the elements `<initial>`, `<delta>` and `<count>`, or the single element `<constant>`, or a keyword. The keyword LATC is always acceptable, and indicates that LCI should not load any value for the parameter. Other acceptable keywords are described below where applicable. The value of the `<delta>` element is the only one which may be negative. Note that the `<count>` value is the number of steps from the initial value; hence the total number of values is one greater than this.

When more than one quantity is being iterated over, the iteration order is fixed by the LCI code. This order is the same as the order in which the iterated quantities are presented in the following descriptions, with the first presented being the first one iterated over (i.e. In the innermost loop).

4.1.0 Common Elements

The following elements are used in all configuration files.

`<number>` : Number of events to collect.

`<period>` : Clock ticks (in 50 nsec units) between calibration pulses.

DEFAULT gives a value of 20000 for a pulse rate of 1 Khz.

If a value less than 20000 is entered then 20000 will be used.

`<delay>` : Delay (in 1/100 secs) inserted after the configuration phase of the calibration cycle.

Large changes in DAC value can require a settling period.

`<latc_delay>` : Additional delay (in 1/100 secs) when a LATC configuration is required.

Performing a LATC configuration generally requires even more settling time afterwards.

<**strobe**> : ON or OFF.

It can be interesting to set up the LAT for calibration, but not pulse the front-end electronics, i.e. send a TACK only. The OFF keyword specifies that the TAM **should not** have the calstrobe bit set. The ON keyword specified that the TAM **should** have the calstrobe bit set.

<**zero_suppress**> : ON or OFF.

ON enables zero-suppression, OFF disables zero-suppression.

4.1.1 Calorimeter Elements

The follow elements are used in calorimeter configuration files.

<**auto_range**> : ON or OFF.

If ON, calorimeter auto-ranging is used and a single range returned. If OFF, all four ranges are returned.

<**first_range**> : Sets the state of the USE_FRST_RNG bit and FRST_RNG bits.

LATC indicates that LCI should not set these bits.

OFF Indicates that the USE_FRST_RNG bit should be cleared.

A number from 0 to 3 indicates the the USE_FRST_RNG bit should be set, and the FRST_RNG bits should be set to the value.

<**gain**> : Gain selection

<**low**> : Low energy gain selection. Three bit field.

<**high**> : High energy gain selection. Four bit field.

<**range-enab**> : Range enable bits.

<**low**> : Low energy range enable. ON or OFF.

<**high**> : High energy range enable. ON or OFF.

<**trig-enab**> : Trigger enable bits.

<**low**> : Low energy trigger enable. ON or OFF.

<**high**> : High energy trigger enable. ON or OFF.

<**calib-enab**> : Calibration enable bits.

<**low**> : Low energy calibration enable. ON or OFF.

<high> : High energy calibration enable. ON or OFF.

<calib-gain> : Calibration gain to use. HIGH or LOW.

<inject> : *Iterate* : Size of the calibration charge. Twelve bit DAC value.

<column> : *Iterate* : Calorimeter column selection, 0 - 11.

FOREACH indicates that LCI should iterate over all the columns in all the towers.

ALL indicates that LCI should enable all columns of all the towers.

<log_accept> : *Iterate* : Log accept threshold. Seven bit DAC value.

<trigger> : Trigger threshold

<low> : *Iterate* : Low energy trigger threshold. Seven bit DAC value.

The range select bit is just treated as the MSB.

<high> : *Iterate* : High energy trigger threshold. Seven bit DAC value.

<tack> : *Iterate* : Eight bit trigger sequence TACK delay.

<range_uld> : *Iterate* : Range upper level discriminator threshold. Seven bit DAC value.

4.1.2 Tracker Elements

The following elements are used in tracker configuration files.

<split> : The splits values to use.

<low> : Low side split value, i.e. the number of low-talking GTFEs, 0 - 24.

<high> : High side split value, i.e. the number of high-talking GTFEs, 0 - 24.

<inject> : *Iterate* : Size of calibration charge. Seven bit DAC value.

<threshold> : *Iterate* : Tracker trigger threshold. Seven bit DAC value.

<tack> : *Iterate* : Eight bit trigger sequence TACK delay.

<channel> : *Iterate* : Tracker channel selection, 0 - 1535.

FOREACH indicates that LCI should iterate over all the channels in all towers with one channel enabled per layer.

ALL indicates that LCI should iterate over all the channels on a TFE with one channel enabled per TFE.

4.1.3 ACD Elements

The following elements are used in ACD configuration files.

<**range**> : Range setting. Possible values are HIGH and LOW.

<**inject**> : *Iterate* : Size of calibration charge. Six bit DAC value.

<**hld**> : *Iterate* : High level discriminator. Six bit DAC value.

<**pha**> : *Iterate* : PHA threshold. Twelve bit DAC value.

<**veto_vernier**> : *Iterate* : Veto vernier. Six bit DAC value.

<**veto**> : *Iterate* : Veto threshold. Six bit DAC value.

<**bias**> : *Iterate* : Bias voltage. Six bit DAC value.

<**hold_delay**> : *Iterate* : Hold delay. Seven bit DAC value.

<**hitmap_delay**> : *Iterate* : Hitmap delay. Five bit DAC value.

<**channel**> : *Iterate* : ACD channel selection, 0 - 215.

FOREACH indicates that LCI should iterate over all the channels, enabling one at a time.

ALL indicates that all channels are enabled.