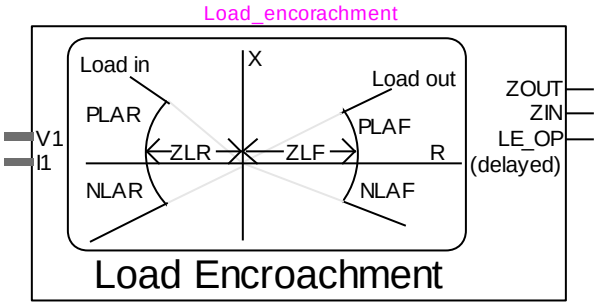
Protection: Load Encroachment functions



Protection: Load Encroachment functions 1
1 Introduction 1
2 Input data of Under voltage functions 1
3 Flags available in the tripping function and the output bundle of the relay 2
4 Scopes 2
5 Modifications 2

Henry Gras, Jean Mahseredjian, 8/16/2016 4:48:00 PM

1 Introduction

This function is intended to supervise the distance elements by defining a load impedance zone. When the measured impedance is inside this zone, the protection functions are blocked. For example, in the following "Tripping logic" (see the data input tab) selection
(21_P OR 21_G) AND NOT (LE_OP)
the distance protection will become inhibited if the impedance falls within the defined zone.

2 Input data of Under voltage functions

The R-X diagram presented in Figure 2-1 contains details for data entered below.

- **V_{min}**: Minimum voltage to enable this function, below this threshold, the functions is disabled.
- **ZLF**: Lower limit of the impedance magnitude of the load zone ZOUT.
- **ZLR**: Lower limit of the impedance magnitude of the load zone ZIN.
- **PLAF**: Upper limit of the positive impedance angle of the load zone ZOUT.
- **NLAF:** Upper limit of the negative impedance angle of the load zone ZOUT.
- **PLAR**: Upper limit of the positive impedance angle of the load zone ZIN.
- **NLAR**: Upper limit of the negative impedance angle of the load zone ZIN.
- **Pickup delay**: Delay of the in-zone detection.
- **Reset delay:** Delay to reset after a pickup when not in the zone anymore.

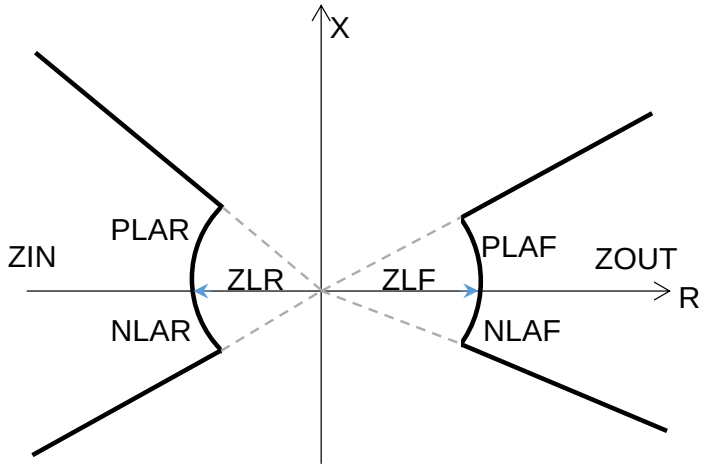


Figure 2-1 Load-encroachment scheme.

3 Flags available in the tripping function and the output bundle of the relay

- ZOUT: in-zone detection in the load zone ZOUT
- ZIN: in-zone detection in the load zone ZIN
- LE_OP*:* ZIN or ZOUT with the pickup and reset delays.

4 Scopes

The following scopes are located in the subcircuit: *RelayName/Control/Console*

- ZOUT: in-zone detection in the load zone ZOUT
- ZIN: in-zone detection in the load zone ZIN

5 Modifications

The protection functions are updated automatically. For example, for memory usage and computational speed considerations, if an entire element is disabled, the subcircuits associated to its functions are replaced by empty subcircuits with the same inputs and outputs. The outputs will be forced to zero or one. When enabled, the subcircuits can take different architectures considering the user choices. Some elements can be excluded if not enabled in the mask.

The updates are performed immediately after entering the parameters and clicking the OK button. The user should wait for the completion of tasks.

If the user wants to modify the subcircuit manually (for example, when adding new scopes), using in the GUI, and avoid the automatic updates of contents, the attribute DeviceVersion has to be set to “none” as shown below. To access to this attribute, right click on the desired device, then go to Attributes and select DeviceVersion (see Figure below).

To allow the automatic updates again, just remove the “none*”* string.

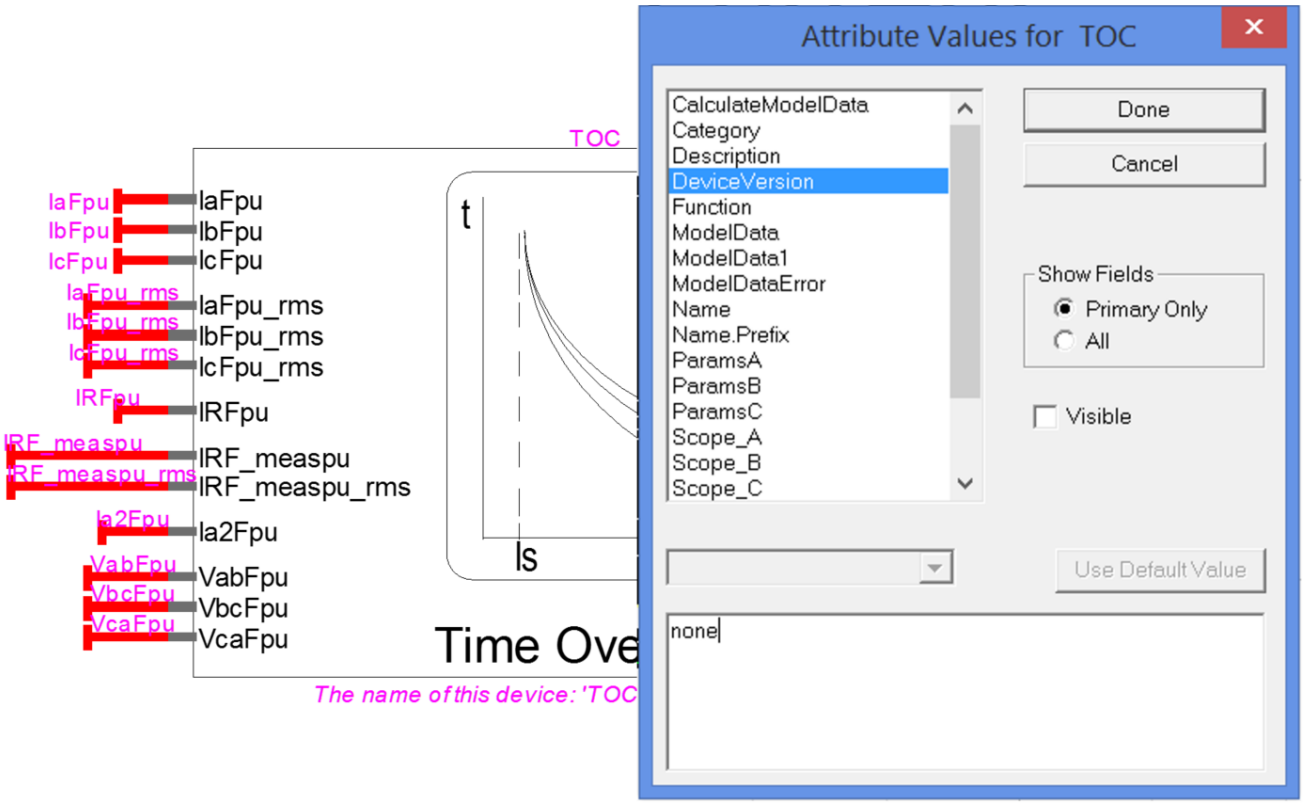


Figure 5-1 How to set the DeviceVersion attribute of the TOC element to allow modifications.