Alarms Testing

# Case FF conformance

The test is specified by AN-010 2.0 “Common Lab Integration Test Requirements”, #2.6

# Case FD\_SIMULATE

FD\_SIMULATE to be tested per definition in FF-891 “Function Block (Part 2)”, #5.3

This parameter allows the conditions to be manually supplied when simulation is enabled. When simulation is disabled both the diagnostic simulate value and the diagnostic value track the actual conditions. The simulate jumper is required for simulation to be enabled.

In SVI FF, the jumper is emulated with local UI (SIMULATN menu).

While FD\_SIMULATE is enabled the recommended action will show that simulation is active.

Verify that the mapping follows this scheme:



# Case Alert Mapping

Verify that each TB alert is or is not reported in FD alerts depending on the setting of TB.ALERT\_ACTION.

Verify that configurable alerts trigger the correct type of FD alert.

Verify that alerts are published if FD priority >=2

# Case TB UPDATE\_EVT

Per FF-891 “Function Block (Part 2)”, #2.1, TB.ST\_REV must increment by more than 1 on

* RB.RESTART = “Restore TB to factory”
* Exiting TB LO mode

because more than one parameter could have been updated.

# Case WRITE\_LOCK

Test must verify conformance with FF-891 “Function Block (Part 2)”, #3.1.3.

# Advanced options

Enable all TB alerts

Trigger the conditions for alerts (really easy is temperature alert, then come pressure alerts)

Modify TB “Advanced” keys to prohibit one or more of these alerts

;TRAVEL\_ACCUMULATION\_A\_ALERT

;TRAVEL\_ACCUMULATION\_B\_ALERT

;TRAVEL\_ACCUMULATION\_TREND

;CYCLE\_COUNTER\_A\_ALERT

;CYCLE\_COUNTER\_B\_ALERT

;CYCLE\_COUNTER\_TREND

;POSITION\_ERROR\_TREND

;NEAR\_CLOSED\_ALERT

;SUPPLY\_PRESSURE\_HI\_ALERT

;SUPPLY\_PRESSURE\_LO\_ALERT

;SUPPLY\_PRESSURE\_LOLO\_ALERT

(Ask for keys for your device, or use Standard key to prohibit all)

Verify that the prohibited alerts become disabled, and previously triggered alerts are no longer active.

Verify that the prohibited alerts do not trigger the bits in FD\_SIMULATE

Verify that the user can’t re-enable the prohibited alerts.

Enable an alert (e.g. temperature) and configure it so that the alert condition is met.

Prohibit the alert in “Advanced” options.

Verify that the alert is disabled and is not active

Allow the alert again in “Advanced” options.

Verify that the alert is enabled and active

# Alerts log

Generate a non-configurable alert from an APP fault.

Verify that it can be found in TB.ALERT\_LOG (in circular reading, perhaps).

# Appendix: Quotes

**FF-891 FS 1.10**

FD\_SIMULATE  
This parameter allows the conditions to be manually supplied when simulation is enabled. When simulation is disabled both  
the diagnostic simulate value and the diagnostic value track the actual conditions. The simulate jumper is required for  
simulation to be enabled and while FD\_SIMULATE is enabled the recommended action will show that simulation  
is active.

5.5.8

With the FD\_xxx\_Active parameters you know the result after the filtering by FD\_xxx\_Map. Prior to filtering by FD\_xxx\_Map, the detected conditions may be observed in the FD\_Simulate.Diagnostic\_Value. Which also serves as the filtered results of  
FD\_Extended\_Map\_x where the FD\_Extended\_Active\_x serves as the observable detected conditionsbefore filtering. The  
vendor must fix the definition of all bits so that DD technology may label them.

2.1

**No alert will be generated while a block is in Out of Service mode**, so that downloads will not generate many update alerts. ST\_REV will

be incremented for each change to static data that occurs while the block is in the O/S mode. On the transition out of O/S mode, an update

alert may be generated if the value of ST\_REV for the block does not match that of the last reported alert.

**3.1.3 Features**

If the resource supports alert reports, the Reports option will be set in the features bit strings. If it is not set, the master must poll for alerts.

Clearing WRITE\_LOCK will generate the discrete alert WRITE\_ALM, at the WRITE\_PRI priority. Setting

WRITE\_LOCK will clear the alert, if it exists. When the soft write lock bit is not true in the features bit strings, writes to the parameter WRITE\_LOCK shall be rejected by the device.

5.3

BLOCK\_ALM

The block alarm is used for all configuration, hardware, connection failure or system problems in the block. The cause of the

alert is entered in the subcode field. The first alert to become active will set the Active status in the Status attribute. As soon as

the Unreported status is cleared by the alert reporting task, another block alert may be reported without clearing the Active

status, if the subcode has changed.

BLOCK\_ERR

This parameter reflects the error status associated with the hardware or software components associated with a block. It is a

bit string, so that multiple errors may be shown.

BLOCK\_ERR\_DESC1 to 9

These parameters are used by a device to report more specific details regarding persistent errors that are reported through

BLOCK\_ERR.

WRITE\_ALM

This alert is generated if the write lock parameter is cleared.

**5.5.1 BLOCK\_ERR\_DESC\_*n***

Care should be taken by the device developer to avoid unnecessarily reporting a given error through more than one

mechanism. For example, a transducer problem should only be reported through its transducer block’s BLOCK\_ERR

and not also be reported through its associated AI block’s BLOCK\_ERR.