Solution to redmine 7209

**TISSUE 1845 SwModKey seems to overwrite the CheckCondition.interlockCheck**

May 28, 2025

The SwModKey is intended to bypass the interlock conditions for local control (typically via the display on the IED, without the Operate service). Considering there is no Check conditions for the local control the SwModKey will indicate if the interlock check shall be performed or ‘bypassed’.



sCtl7 need to be updated to verify that SwModKey has NO impact on the normal Operate service from the remote/station level (orCat=2 or 3). The expected result is exactly the same.

|  |  |  |
| --- | --- | --- |
| **sCtl7** | **Check conditions** | [ ]  Passed[ ]  Failed[ ]  Inconclusive |
| IEC 61850-7-2 Subclause 20.5.2.5IEC 61850-8-1 Subclause 20PIXIT: Ct8 |
| Expected result1. DUT returns
2. DOns: Operate response- with optional AddCause = “Blocked-by-interlocking”
3. SBOns: Select response+ and Operate response- with optional AddCause = “Blocked-by-interlocking”
4. DOes: Operate response- with AddCause = “Blocked-by-interlocking”
5. SBOes: SelectWithValue response- with AddCause = “Blocked-by-interlocking” OR

 SelectWithValue response+ and Operate response- with AddCause = “Blocked-by-interlocking”1. DUT returns Select/SelectWithValue/Operate response+ OR (PIXIT)

a) DOns: Operate response- with optional AddCause = “Not-supported” or “Blocked-by-interlocking”b) SBOns: Select response+ and Operate response- with optional AddCause = “Not-supported” or “Blocked-by-interlocking”c) DOes: Operate response- with AddCause = “Not-supported” or “Blocked-by-interlocking”d) SBOes: SelectWithValue response- with AddCause = “Not-supported” or “Blocked-by-interlocking” OR  SelectWithValue response+ and Operate response- with AddCause = “Not-supported” or  “Blocked by interlocking”1. DUT returns Select/SelectWithValue/Operate response+
 |
| Test descriptionPrecondition: origin.orCat=2 or 3 and LLN0.SwModKey=F or absent1. Test engineer forces CILO.EnaOpn/EnaCls = FALSEa) DOns: Client sends correct Operate request with Check Interlock setb) SBOns: Client sends correct Select and Operate request with Check Interlock setc) DOes: Client sends correct Operate request with Check Interlock setd) SBOes: Client sends correct SelectWithValue and on response+ Operate with Check Interlock set2. Test engineer forces CILO.EnaOpn/EnaCls = FALSEa) DOns: Client sends correct Operate request with Check Interlock not setb) SBOns: Client sends correct Select and Operate request with Check Interlock not setc) DOes: Client sends correct Operate request with Check Interlock not setd) SBOes: Client sends correct SelectWithValue and Operate with Check Interlock not set3. Test engineer forces CILO.EnaOpn/EnaCls = TRUEa) DOns: Client sends correct Operate request with Check Interlock setb) SBOns: Client sends correct Select and Operate request with Check Interlock setc) DOes: Client sends correct Operate request with Check Interlock setd) SBOes: Client sends correct SelectWithValue and Operate with Check Interlock set4. When LLN0.SwModKey is present, change the value to T and repeat step 1, 2 and 3 |
| Comment |