**GENERAL TEST MOD/ARI**

# Process Setup

Asks the user what the product (MOD) and communication module (COMM) are under test and the type of alimentation. The sequence saves this information in FileGlobals variables that will be used in all the other sequences.

# PreUUT

Initialize the power supply (Omicron) with the parameters chosen in [Process Setup](#_Process_Setup) and turns on the MOD. Set some default values of the DUT. If present, initialize the communication with the COMM and set the default values.

# MainSequence

Asks the user what sub-sequences want to execute.

# General and Architecture and LED

This sequence checks all the main functionalities of the device. It checks, in each possible configuration of the device, the possibility of manual opening/closing, but also the LED and the procedure to Lock the device.

1. Check Manual Open/Close and LED at power off
2. Check Manual Open/Close and LED in Not-Active state
3. Check Manual Open/Close and LED in Active state

## MOD

1. Check Manual Open/Close and LED in Lock State

## ARI

1. Check Manual Open/Close and LED in Standby-mode and Lock State
2. Check reset of the number of reclosing attempts
   1. During Power Fail
   2. During Manual Open/Close
   3. During Switch turn
3. Check Manual Open/Close and LED with continuous trip

# Input/Output

This sequence tests all the functionalities of the Inputs and Outputs in each configuration of the device.

1. Check inputs are not working in Not-Active state
2. Check inputs and outputs are working in Active State in each lever position (Opne, Half-Open, Close)

## MOD

1. Check Outputs in Lock state
2. Check Input closing command works only after the dead time or after a power fail

## ARI

1. Check outputs during Stand-by mode and Lock state
2. Check input closing in stand-by mode does not work

# Power Outage

This sequence tests the behaviour of the device when a power fail occurs during a manoeuvre.

1. Check that, if a power fail occurs during the closing manoeuvre, the device opens automatically at power on.

## ARI

1. Check that, if a power fail occurs during the Auto-reclosing manoeuvre, the device opens and after closes automatically at power on.

# Motor Driver

This sequence measures with a multimeter the power consumption at rest and during a manoeuvre.

1. Measure of the power consumption at rest
2. Measure of the power consumption during opening manoeuvre
3. Measure of the power consumption during closing manoeuvre

## ARI

1. Measure of the power consumption during Auto-reclosing manoeuvre

# Modbus Register Check

Check the functionalities of all the registers of the Modbus map.

## INTERNAL COMMUNICATION FAILURE

Check that the communication between MOD and COMM works.

## COMMAND BREAKER

1. Check the opening and closing command via communication.

### MOD

1. Check that the command via communication does not work during dead time

### ARI

1. Check that the command via communication does not work during stand-by mode and lock state

## ENABLE/DISABLE REMOTE COMMANDS

1. Check that both input and communication remote command can be simultaneously disabled via communication
2. Check that only the communication remote command can be disabled
3. Check that both input and communication remote command can be restore via communication
4. Check that only the input remote control can be disabled

## ARI SETTINGS

Check that the auto-reclosing settings of the ARI can be modified via communication and work properly:

* Reclosing Attempts: number of auto-reclosing procedures before the device enter the Lock state.
* Waiting time: time that the ARI wait in the stand-by mode before the auto-reclosing procedure.
* Neutralization time: amount of seconds beyond which the next trip will be considered as the first.

ATTENTION: the neutralization time written via communication in the Modbus register is not the real neutralization time of the device, because the real neutralization time used in the firmware is always Real\_Neutralization\_time = Neutralization\_Time + Waiting\_time in order to have always Real\_Neutralization\_Time > Waiting\_time.

## STATUS BREAKER

Check that the position of the lever of the MOD is detected properly (in each possible state of the device) by reading the Modbus register of the status breaker.

## TRIPPED

Check that the trip of the MPD is detected properly.

## TIME TRIPPED

1. Read the time of the last trip
2. Trip
3. Read the time of the last trip
4. Check that the two times just read are different

## Manoeuvre parameters

Read the parameters (PEAK CURRENT, MOTOR TORQUE, TIME TO MAX TORQUE, PEAK TORQUE TOGGLE ANGLE AND CLOSE/OPEN TIME) of the manoeuvre during multiple opening and closing manoeuvres and display the graphs.

## NUMBER OF MANOEUVRE

1. Call a parallel sub-sequence “Modbus\_read\_loop” in order to detect if the Modbus register MANOEUVRE ATTEMPT ONGOING indicates that a manoeuvre is running during an opening/closing procedure.
2. Check that the registers of the number of manoeuvre and of the number of closing manoeuvre are updated correctly when an opening/closing procedure is completed.

### ARI

1. Check that the registers of the number of manoeuvre and of the number of closing manoeuvre are updated correctly when an Auto-reclosing procedure is completed.

## INPUT/OUTPUT

Check that the input and output registers work properly in each possible state of the device.

## TIME POWER FAIL

1. Read the time of the last power fail
2. Power fail
3. Read the time of the last power fail
4. Check that the two times just read are different

## LAST DEMANDED COMMAND FAILED

1. Call a parallel sub-sequence “Modbus\_read\_loop\_power\_fail\_search” that can detect if the Modbus register LAST DEMANDED COMMAND FAILED indicates that the last manoeuvre is failed due to a power fail.

### ARI

Check that the register works if the Auto-reclosing procedure fail.

## DIAGNOSTIC REGISTER

Check of the diagnostic register in each state of the device.

## MOD DUT INFORMATION

Read and display in the report the information of the MOD and COMM.

# Double Input

This sequence check that the double command has no effect on the device.

1. Modify the setup in order to send the input opening/closing command simultaneously to the two inputs of the MOD
2. Loop of repeated double-input command with a parallel sub-sequence “Omicron read loop output 1” that counts if a manoeuvre is executed.
3. Returns to the separate output setup

# Eprom Writing

This sequence checks the stability of the Eprom writing process with repeated power fail events during the eprom writing process.

1. Repeated power fails just before an Opening/Closing manoeuvre
2. Repeated power fails just after an Opening/Closing manoeuvre

# Modbus Settings

This sequence tests the modbus communication by varying baudrate, parity and address.

1. Check the communication at each address
2. Check the communication at each parity
3. Check the communication at each baudrate

# COMM LED & Button

This sequence tests the LED and the button of the COMM module.

1. Check the LED of the COMM when is not connected to the MOD
2. Check the LED of the COMM when the address is not set
3. Check the LED of the COMM when the address is set through micro switch
4. Change the Address via Modbus and verify the new communication parameters
5. Use the button to reset the default settings and verify

# Firmware Upgrade

This sequence tests the Firmware Upgrade procedure of the MOD via COMM module.

1. Read FW upgrade register and check that is not ongoing
2. Send FW Upgrade command
3. Check the LED during the FW Upgrade and wait
4. Reboot the MOD
5. Read new FW version and display in the report (compared to the old one).

# PostUUT

Power off of the Omicron.