



HYDRONIC_IOs_Reordering_Misconfig_Cases

1.

1	Overview	3
2	Reordering & misconfiguration cases	3
2.1	Preconditions	3
2.2	Cases Analysis	3
2.2.1.	Bad or no communication	3
2.2.2.	Swapping slaves of same type.....	3
2.2.3.	Swapping slaves of different type.....	4
2.2.4.	Replacement of slave of same type	4
2.2.5.	Replacement of slave of different type	4
2.2.6.	Adding slaves at the end.....	4
2.2.7.	Adding slaves in between	4
2.2.8.	Missing Slaves.....	4

This document describes the system behavior of the HYDRONIC project according to the physical Slave connection to the RS485 communication line and analyses all possible cases of reordering or misconfiguration of the slave devices.

2 Reordering & misconfiguration cases

2.1 Preconditions

For all of the following cases a correct system configuration is assumed. Mainly that the IO addressing has been executed successfully based on the IO definitions and thus the serial numbers appear in the IO Configuration panel. In that case, the IO Configuration declarations match exactly one-by-one the physically connected slaves. Considering that this IO configuration is static, with no further changes, the following cases are identified.

Warning: For all of the following cases except 2.2.1, the user **must first power off the master device**, perform the desired replacement or swapping actions that are described below and then power on again the master device.

2.2 Cases Analysis

2.2.1. Bad or no communication

With an active project, the communication may be corrupted and lost at some slave device and beyond.

All the slaves, communicating or not, shall be set to their default deactivation (safety) values. The python code shall continue its execution, but without affecting any IO behavior during corrupted communication state. The master shall try to reconnect with the slaves and restore the communication, making multiple iterations through the communication bus, identifying each connected slave. The system shall provide a time frame of approximately 1-2 minutes for the missing slaves to reconnect and if they do so, then all the slaves' channels shall be reconfigured (activation values, channels, sensors, PWM, etc). In case the communication is not restored inside this time frame, the project shall get deactivated and the master shall auto-reset and attempt only once to reactivate the project. Upon reactivation, if the communication is restored and no further problems exist (e.g. IOs misconfiguration), the project shall run normally. In case that errors are detected, the project shall get deactivated again and the programmer must activate it manually through the UI.

2.2.2. Swapping slaves of same type

Independently of an active project or not, two slaves of the same type get swapped.

In that case, the system shall update automatically the IO configuration list and shall continue normally its operation.

2.2.3. Swapping slaves of different type

Independently of an active project or not, two slaves of different type get swapped.

As in the previous case, the system shall update automatically the IO configuration list and shall continue normally its operation.

2.2.4. Replacement of slave of same type

Independently of an active project or not, a slave gets replaced with a new slave of the same type.

In that case, the system shall update automatically the serial number of the old slave entry with the new one and shall continue normally its operation.

2.2.5. Replacement of slave of different type

Independently of an active project or not, a slave gets replaced with a new slave of different type.

In that case, the system shall detect an IOs **misconfiguration** status and shall alert the user.

2.2.6. Adding slaves at the end

Independently of an active project or not, additional slave devices are added at the end of the pre-existing connected slaves.

In that case, the system shall not be affected at all and shall continue its operation normally.

2.2.7. Adding slaves in between

Independently of an active project or not, additional slave devices are added among the pre-existing connected slaves.

In that case, the system shall consider this a reordering case and shall continue its operation normally.

Note: If the precondition at the beginning of this section about IO addressing is not met, the system understands that the user had not executed a successful IO addressing for the pre-existing connected slaves and this is not considered as a reordering case. Instead the system shall detect IOs **misconfiguration** status and alert the user accordingly.

2.2.8. Missing Slaves

Independently of an active project or not, some slave devices are missing (total number of connected slaves is less than what is configured in IO Configuration).

i) When for any reason some devices in the bus are missing upon an IO addressing check, either manual, or automatically performed, the user shall be alerted for the specific **missing** slave devices.

- For example, if the user has declared 10 slave devices in IO Configuration, the first 5 match one-by-one the physically connected, but the next 5 are missing, then the system shall alert the user that these last 5 slave devices are missing.
- ii) When for any reason some devices in the bus are missing and at the same time others are not connected as expected in IO Configuration, the user shall be alerted only for the **misconfigured** slave devices, since the misconfiguration case is considered of higher importance and must be solved first. A second IO addressing will then inform for the missing ones.
- For example, when the user has declared 10 slave devices in IO Configuration and from 1 to 5 they match one-by-one the physically connected slave devices, if from 6 to 8 they are not of the correct type (misconfigured) and the 9-10 are missing, then the system shall alert the user that these 6 to 8 slave devices are misconfigured, according to 2.2.5 case.