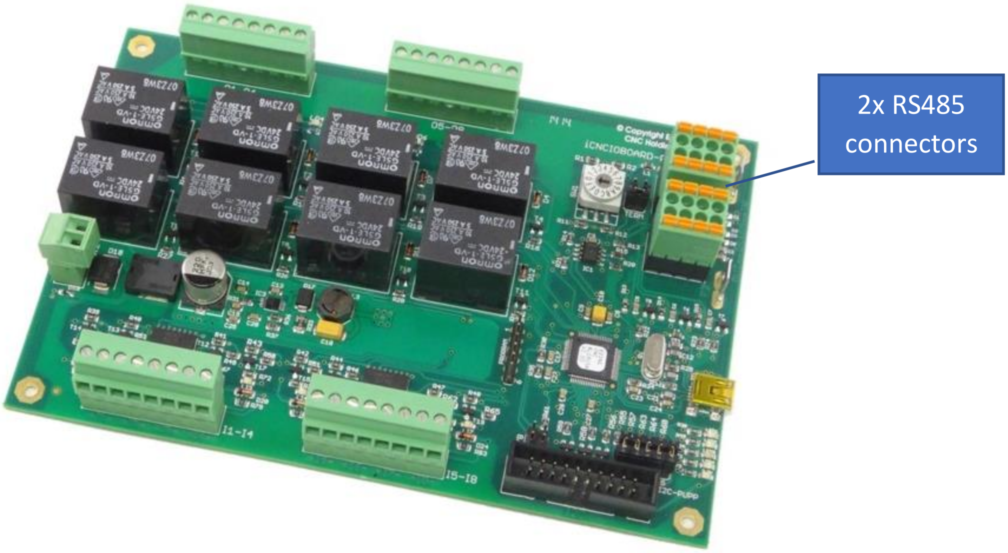
How-to connect RLY8IO

Rev.2

The RS485 interface on RLY8IO

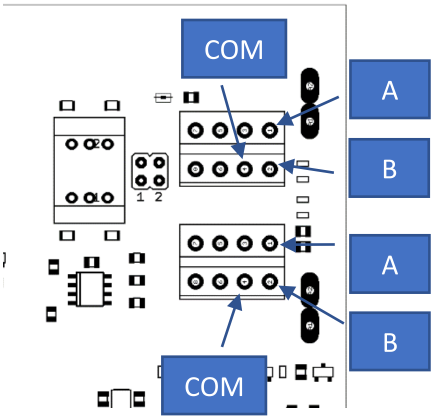
The RS485 interface is used to connect the RLY8IO to a controller but can also be used to daisy chain other boards with a RS485 interface. For this reason, the RLY8IO has two connectors. Both connectors have identical pinning.



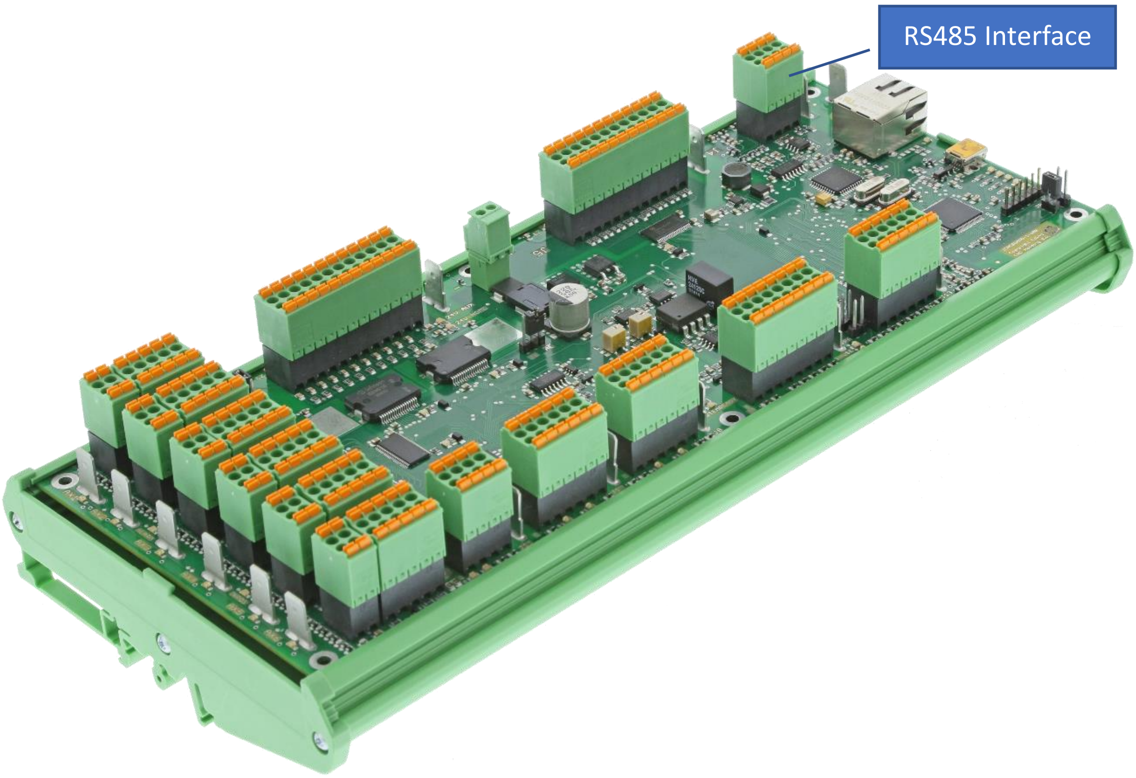
Please note, the last interface card that is daisy chained needs to have its termination resistors enabled. On the other boards this termination should be disabled.

The RS485 interface consists out of 2 signals called A and B, and one wire called common.

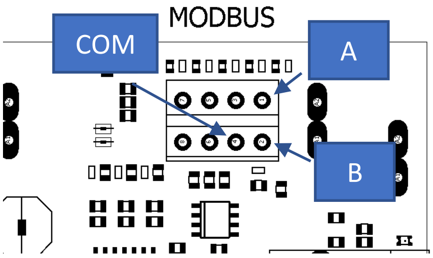
The wiring on the RLY8IO is as follows:



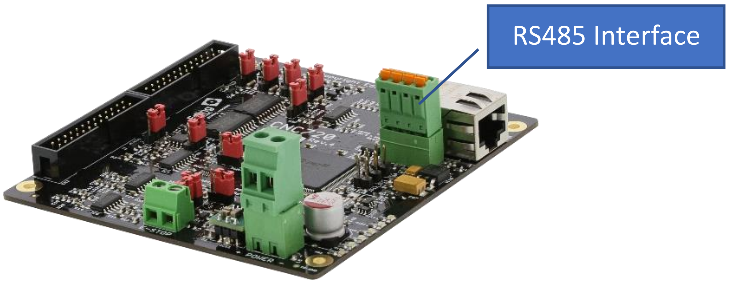
On the iCNC600 the RS485 connections are identical to the RLY8IO interface:



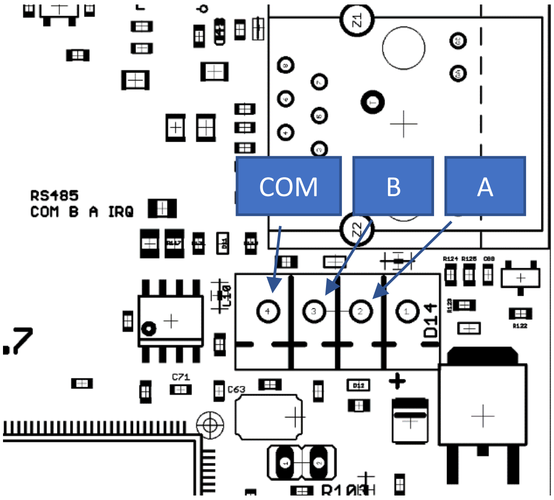
The image below shows the actual connection on the connector.



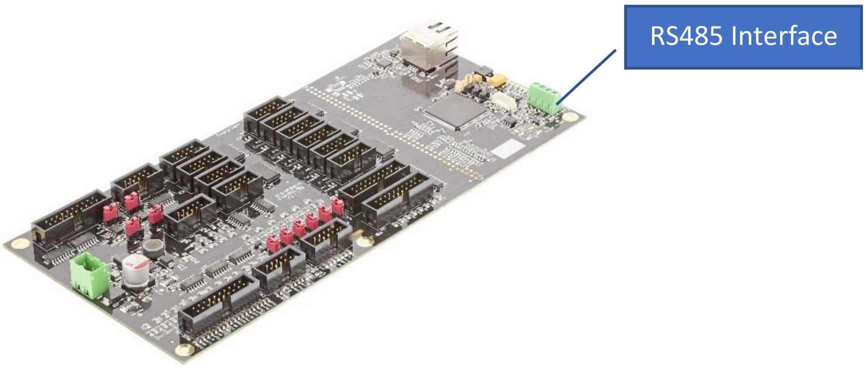
The CNC720 connections are shown in the image below.



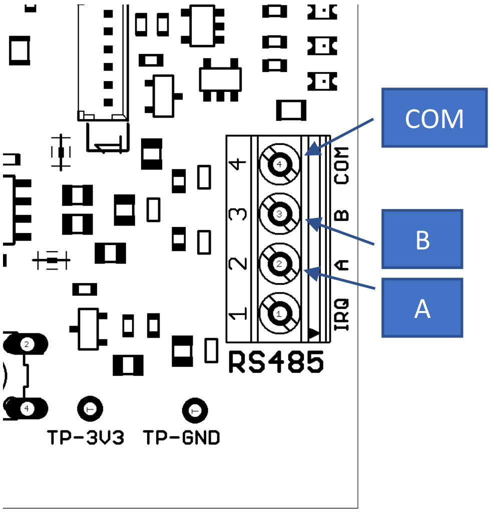
The image below shows the actual connections on the connector.



The CNC760 connections are shown in the image below.

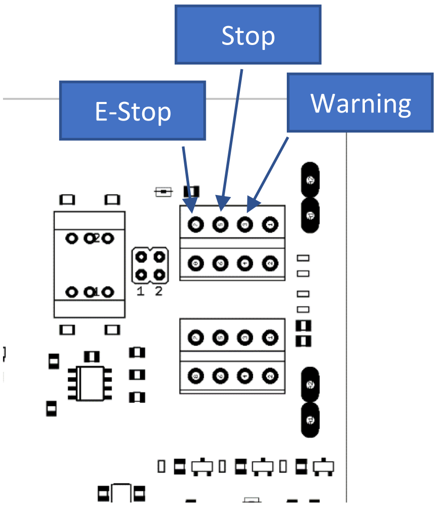


The image below shows the actual connections on the connector.

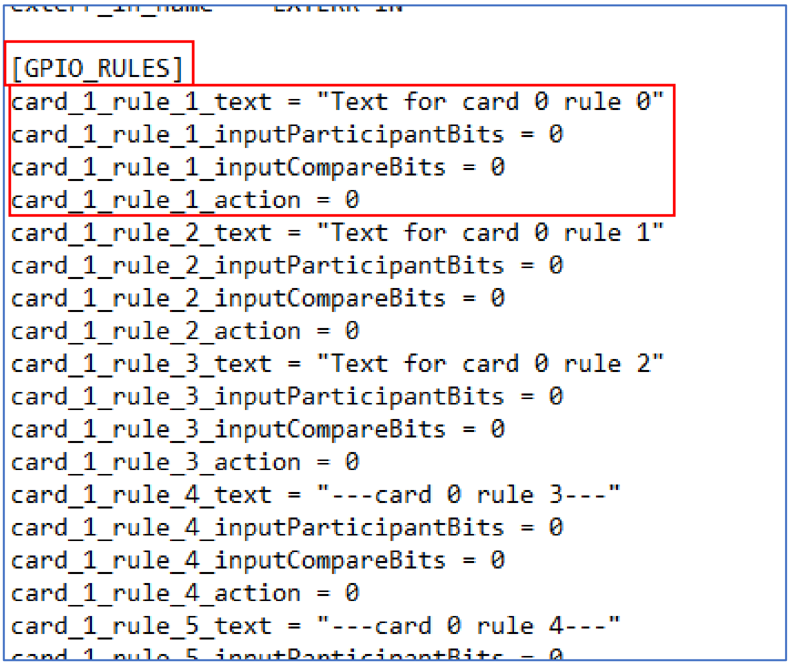


Using estop, stop, warning outputs

The RLY8IO has additionally some extra outputs that can indicate warning, stop or E-stop.



These outputs are open-collector type. They can be activated by settings in the CNC.INI file.
The settings can be found under the section [GPIO_RULES], below is an image showing a part of this.



```
exterr_in_name    = EXTERR IN

[GPIO_RULES]
card_1_rule_1_text = "Text for card 0 rule 0"
card_1_rule_1_inputParticipantBits = 0
card_1_rule_1_inputCompareBits = 0
card_1_rule_1_action = 0
card_1_rule_2_text = "Text for card 0 rule 1"
card_1_rule_2_inputParticipantBits = 0
card_1_rule_2_inputCompareBits = 0
card_1_rule_2_action = 0
card_1_rule_3_text = "Text for card 0 rule 2"
card_1_rule_3_inputParticipantBits = 0
card_1_rule_3_inputCompareBits = 0
card_1_rule_3_action = 0
card_1_rule_4_text = "---card 0 rule 3---"
card_1_rule_4_inputParticipantBits = 0
card_1_rule_4_inputCompareBits = 0
card_1_rule_4_action = 0
card_1_rule_5_text = "---card 0 rule 4---"
card_1_rule_5_inputParticipantBits = 0
```

For each rule 4 lines exist:

Text	The text displayed when the defined condition appears
inputParticipantBits	The bits that are to be monitored
inputCompareBits	The level to which it is compared low (0) or high (1)
Action	Which action to perform: 0 = No action 1 = Warning 2 = Stop 3 = E-stop

Text

The 'text' will be displayed in the operator screen if a monitored condition occurs.

inputParticipantBits

This value is a binary value indicating which bits needs to be monitored. Since there are 8 inputs there are 8 values:

# Input	Value
None	0
Input 1	1
Input 2	2
Input 3	4
Input 4	8
Input 5	16
Input 6	32
Input 7	64
Input 8	128

To monitor, for example, input #5 you would write:

```
inputParticipantBits = 16
```

However, you could also want to monitor input #5 and #8, in that case you would want to add these two values, so 16 + 128 = 144. So the result would be:

```
inputParticipantBits = 144
```

inputCompareBits

The next step is to indicate whether you want to compare the value for a high (1) or low (value). To indicate this the same values are used as for the 'inputParticipantBits'. Except, now they indicate the level you want to check for.

For example, if you want to check input #5 for being high, you would write:

```
inputCompareBits = 16
```

If you want to compare it to low, you would write:

```
inputCompareBits = 0
```

If you would want to monitor input #5 and #8 for both high:

```
inputCompareBits = 144
```

If you would want to monitor input #5 for low and #8 for high:

```
inputCompareBits = 128
```

Action

The last step is to indicate what kind of message is send, or output is activated.

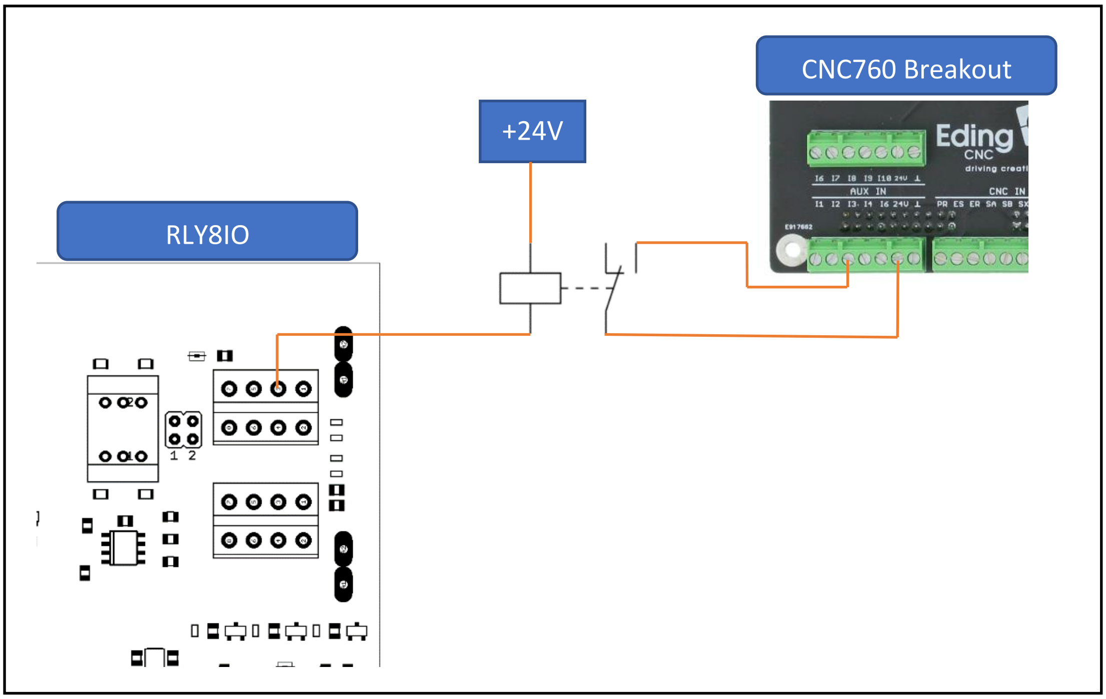
The outputs are reset when '**Reset'** is press on the operational screen.

Connecting RL8IO outputs to a CNC controller.

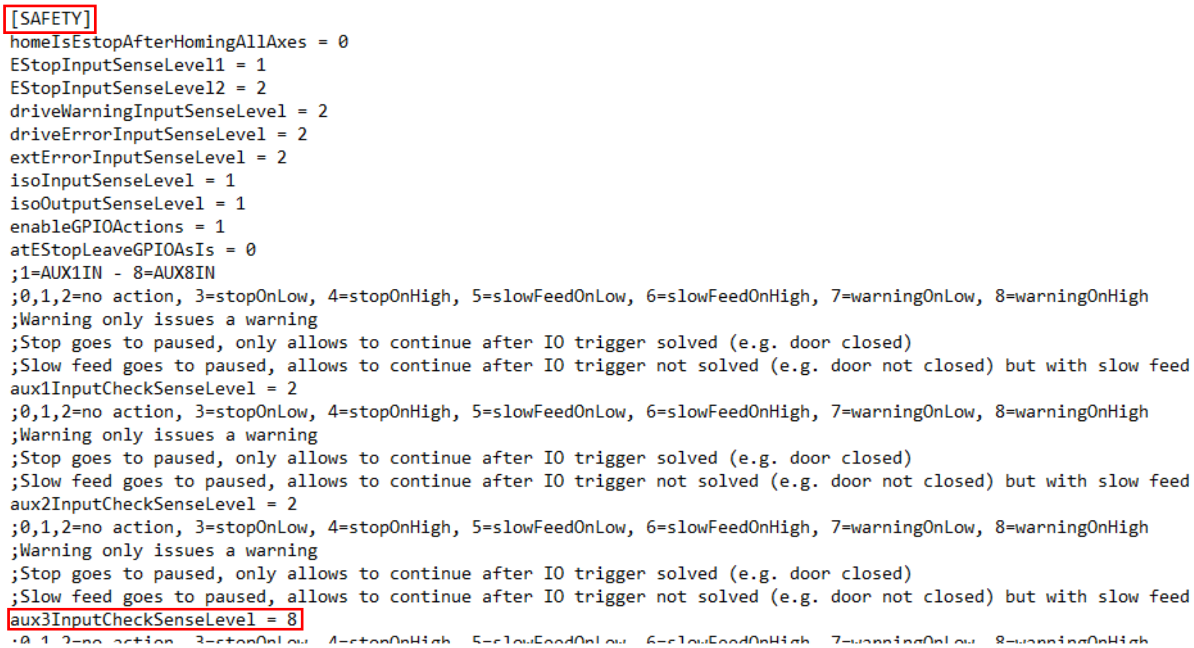
The outputs of the RLY8IO that indicated the warning, stop or e-stop signal are open-collector. This means when they output is activated the output will switch to ground. The controller can directly be connected to the iCNC600. Unfortunately the CNC760 does not have the inputs for the RLY8IO to connect directly to it. However, by using/configuring the AUX outputs similar behavior can be made.

CNC760

The easiest way to connect the RLY8IO status outputs is by a relay. Below is diagram that shows how to connect a relay tot to the 'Warning' output of the RLY8IO connector to, as example, AUX3 input.



If the warning output is activated it will show that AUX input #3 will also become high. The CNC760 input (in this case #3) needs to be configured to watch input 3. For the CNC760 this configuration will be:



```
[SAFETY]
homeIsEstopAfterHomingAllAxes = 0
EStopInputSenseLevel1 = 1
EStopInputSenseLevel2 = 2
driveWarningInputSenseLevel = 2
driveErrorInputSenseLevel = 2
extErrorInputSenseLevel = 2
isoInputSenseLevel = 1
isoOutputSenseLevel = 1
enableGPIOActions = 1
atEStopLeaveGPIOAsIs = 0
;1=AUX1IN - 8=AUX8IN
;0,1,2=no action, 3=stopOnLow, 4=stopOnHigh, 5=slowFeedOnLow, 6=slowFeedOnHigh, 7=warningOnLow, 8=warningOnHigh
;Warning only issues a warning
;Stop goes to paused, only allows to continue after IO trigger solved (e.g. door closed)
;Slow feed goes to paused, allows to continue after IO trigger not solved (e.g. door not closed) but with slow feed
aux1InputCheckSenseLevel = 2
;0,1,2=no action, 3=stopOnLow, 4=stopOnHigh, 5=slowFeedOnLow, 6=slowFeedOnHigh, 7=warningOnLow, 8=warningOnHigh
;Warning only issues a warning
;Stop goes to paused, only allows to continue after IO trigger solved (e.g. door closed)
;Slow feed goes to paused, allows to continue after IO trigger not solved (e.g. door not closed) but with slow feed
aux2InputCheckSenseLevel = 2
;0,1,2=no action, 3=stopOnLow, 4=stopOnHigh, 5=slowFeedOnLow, 6=slowFeedOnHigh, 7=warningOnLow, 8=warningOnHigh
;Warning only issues a warning
;Stop goes to paused, only allows to continue after IO trigger solved (e.g. door closed)
;Slow feed goes to paused, allows to continue after IO trigger not solved (e.g. door not closed) but with slow feed
aux3InputCheckSenseLevel = 8
;0,1,2=no action, 3=stopOnLow, 4=stopOnHigh, 5=slowFeedOnLow, 6=slowFeedOnHigh, 7=warningOnLow, 8=warningOnHigh
```

Please note, that input #3 can be configured as Warning, Stop, Low-Feed on the CNC760. For more information about configuring the AUX input please refer to the manual.