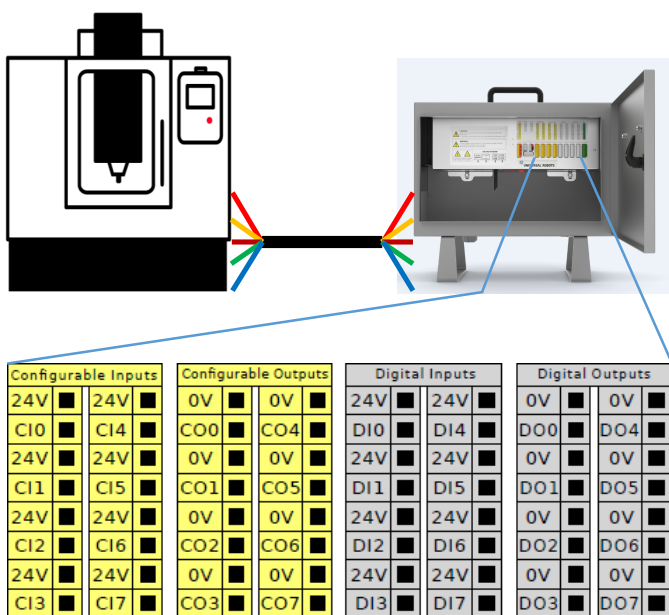


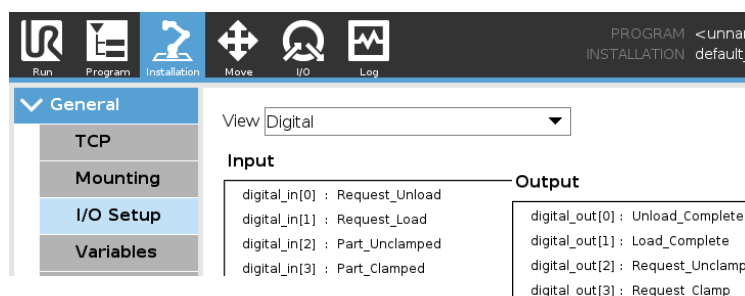
## Connection

Connect the machine to the robot using a multicore cable, where every signal will have its own wire. Use electrical isolation devices such as relays when the robot and the machine use different power supplies for the I/O systems.



## I/O Setup

Here you can give meaningful names to the signals connected to the Controller I/O to make it easier to use them in the Robot Program. Use the filter **View Digital/Configurable**.



**ProTip!** Configure an action for the signals **Unload\_Complete** and **Load\_Complete** to change to **Low when [the program is] not running**.

Rename

Action in program

## Signal Exchange

The most basic signal exchange requires only one signal to be sent from the machine to the robot in order to load or unload it.

However, including some additional signals to the machine can make the system more robust.

### Digital Input

- ☐ Request\_Unload **Indicates when the machine requests to be unloaded.**
- ☐ Request\_Load **Indicates when the machine requests to be loaded.**
- ☐ Part\_Unclamped **Indicates when the part is unclamped in the machine.**
- ☐ Part\_Clamped **Indicates when the part is clamped in the machine.**

### Digital Output

- ☐ Unload\_Complete **Confirms that the unload request is complete.**
- ☐ Load\_Complete **Confirms that the load request is complete.**
- ☐ Request\_Unclamp **Requests for the machine unclamp the part.**
- ☐ Request\_Clamp **Requests for the machine to clamp the part.**

Your machine may also require status or enabling signals. In some cases it's possible to trigger these with actions configured in the installation instead of setting them in the program.

The output **Robot\_Ready** indicates when the robot is ready for automatic loading and unloading. An I/O action could be configured to set it to **High when [the program is] running-low when stopped**.

The input **Machine\_Ready** indicates when the machine is ready for automatic loading and unloading and it could have an action to **Start Program**.

The output **Out\_Interf** indicates when the robot is outside the machine and it could be set using **Safety Planes** and **I/Os** using the **Normal** and **Reduced** modes.

When the **Enable\_Interference** input signal is high, the robot is allowed to move within the interference area inside the machine, but not when it is low. This can be handled by an additional thread in your program.