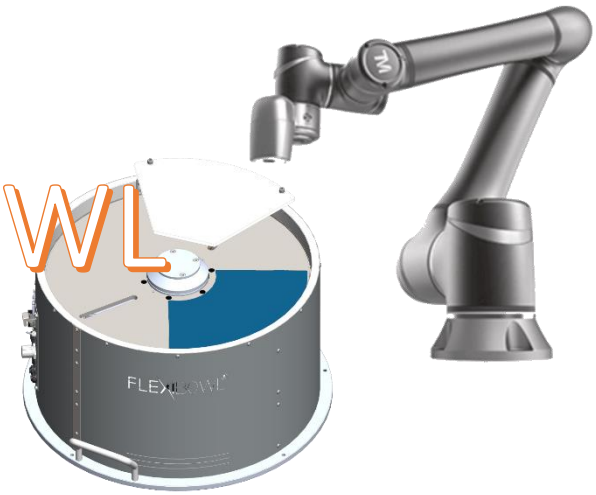
TM FLEXIBOWL PLUGIN



This Plugin was developed with the idea of communicating quickly and safely with the flexibowl through TM robots, using TMFLOW 1.76 or later software
The Plugin does not require any license to operate

FlexiBowl®

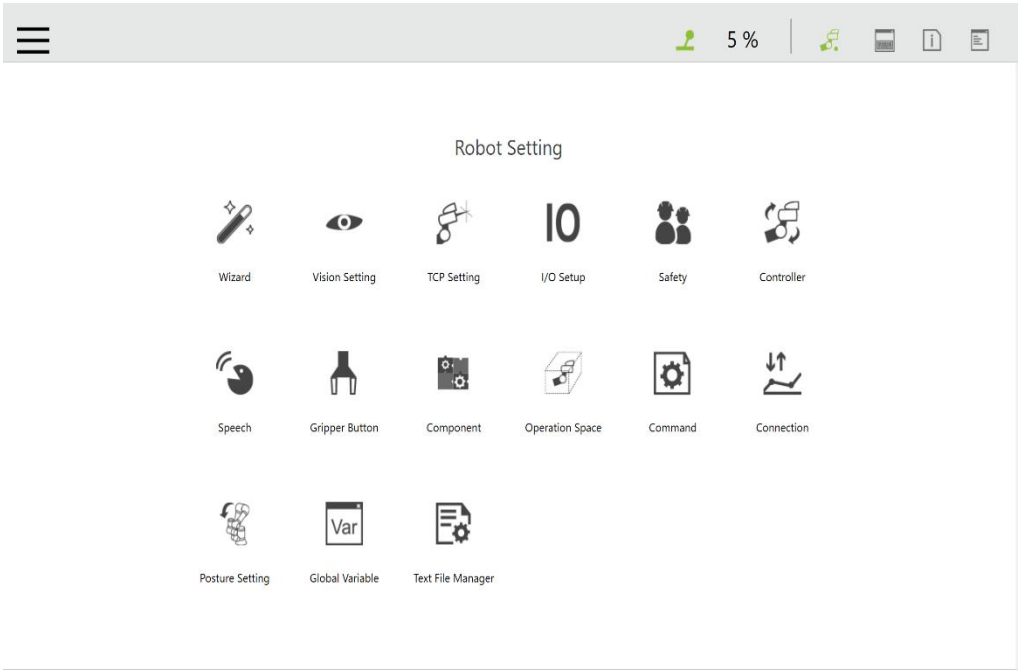
TM
FLOW

The Plugin is in the USB\TMROBOT folder, formed by two global variables and a Flexibowl Plugin program for Flexibowl movement. From here, the Plugin can be imported or created from scratch with the documents we will provide.
We will describe the operation of the Plugin and the creation process below.

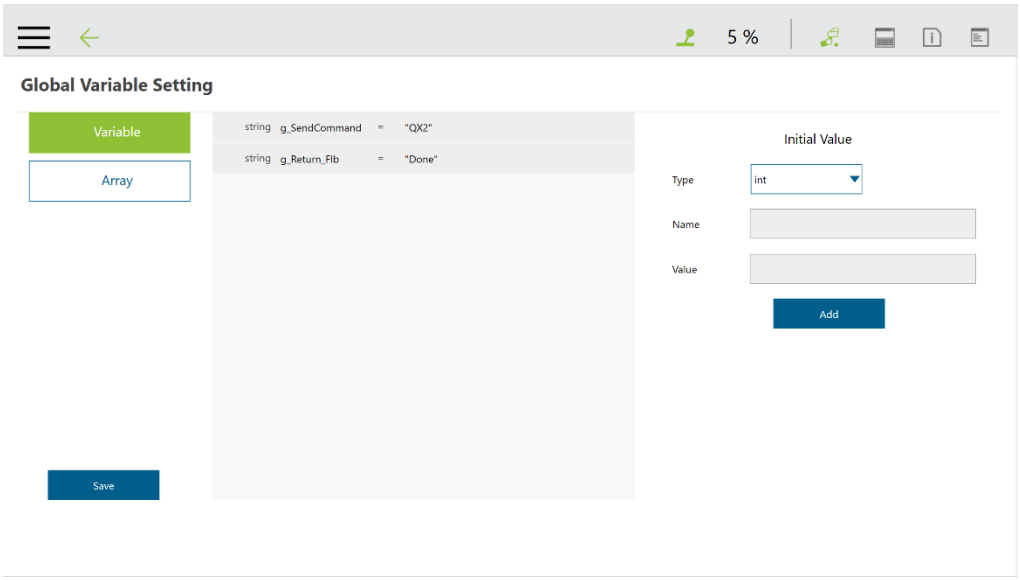
STEP 1:

Creation of two global variables for operation.
We will create two global variables
1)*Send_Command* to send the command to the Flexibowl
2)Return_Flb to receive the return string from the Flexibowl

Go to the Robot Setting➔Global Variable by means of the TmFlow menu
And create two string variables as shown in the image



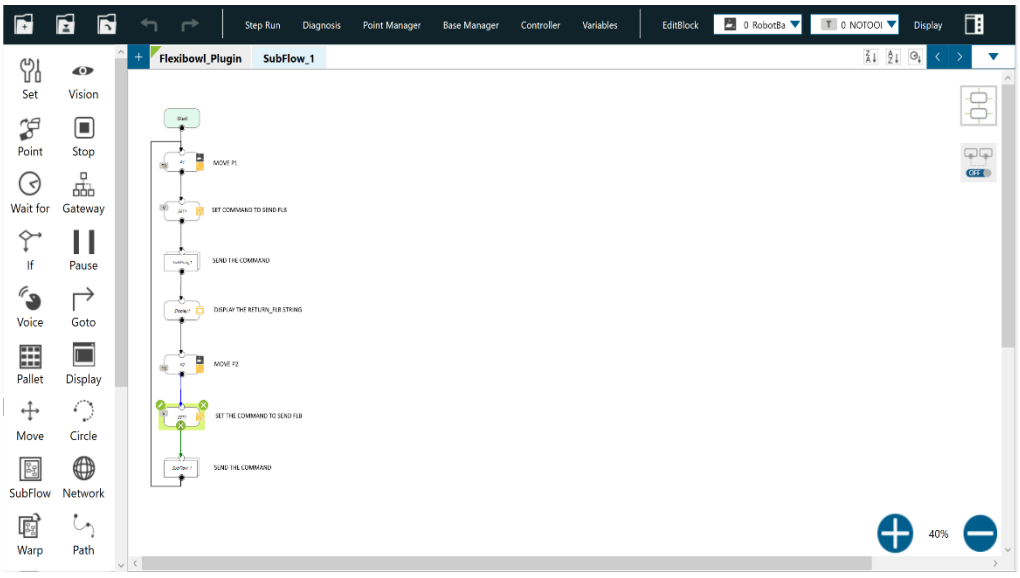
TM
FLOW

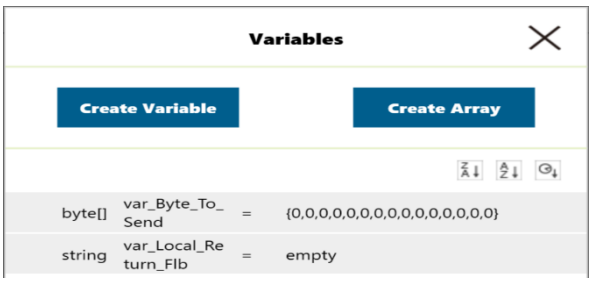


STEP 2:

Creation of a simple program to handle the robot and activate the Flexibowl.
Two local support variables will be created here

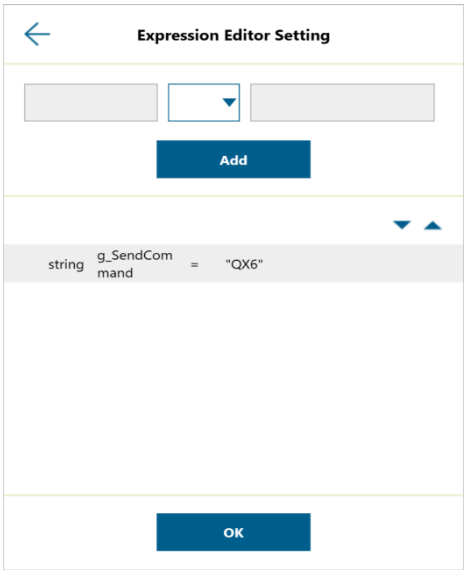
1)BYTE [] ARRAY *Byte_To_Send* from 15 positions, to send the command
2)String *Local_Return_Flb*





TM
FLOW

Let’s see how to set the Flexibowl command variable for movement



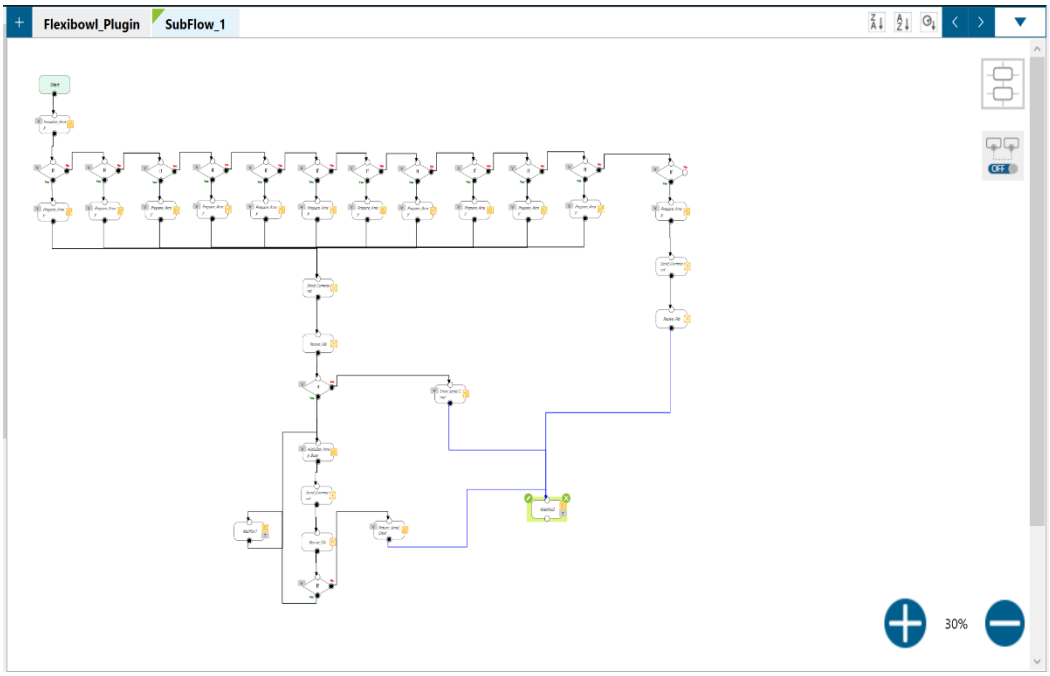
Command	Description
QX2	Move
QX3	Move - Flip
QX4	Move - Blow - Flip
QX5	Move - Blow
QX6	Shake
QX7	Light on
QX8	Light off
QX9	Flip
QX10	Blow
QX11	Quick Emptying Option
QX12	Reset Alarm
AL	Status Alarm

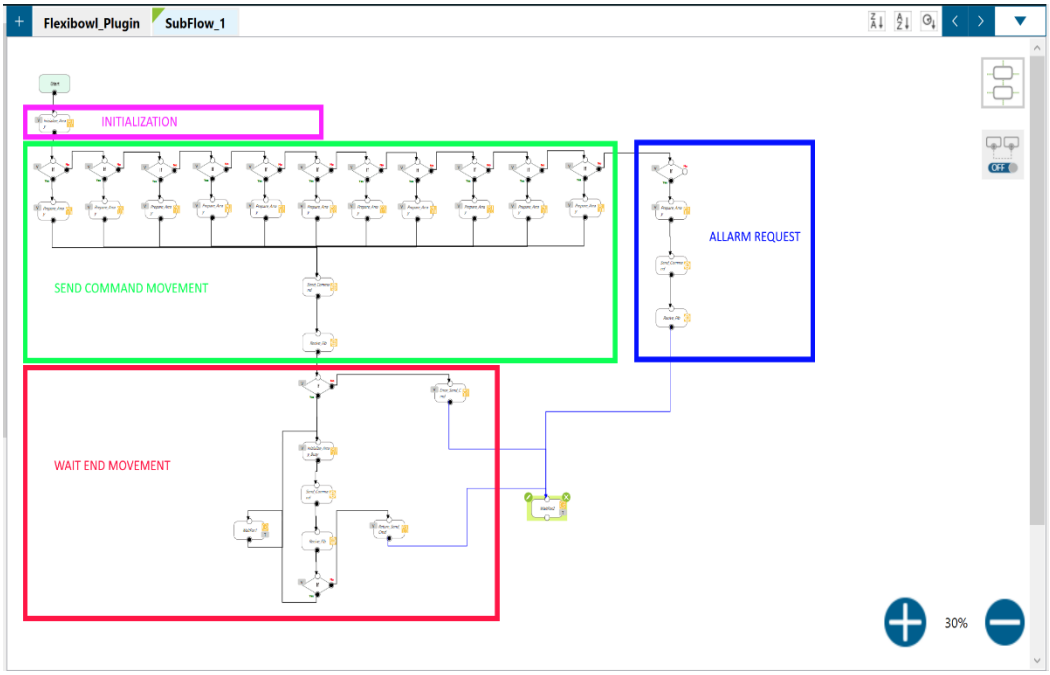
Action	Description
MOVE	Moves the feeder the current parameters.
MOVE-FLIP	Moves the feeder and activates Flip simultaneously
MOVE-BLOW-FLIP	Moves the feeder and activates Flip and blow simultaneously
MOVE-BLOW	Moves the feeder and activates Flip simultaneously
SHAKE	Shakes the feeder with the current parameters
LIGHT ON	Light on
LIGHT OFF	Light off
FLIP	Flip
BLOW	Blow
QUICK_EMPTYING	Quick Emptying Option
RESET_ALARM	Reset Alarm and enable the motor

TM FLOW

STEP 3:

Now we will analyse the SUBFLOW for Flexibowl movement

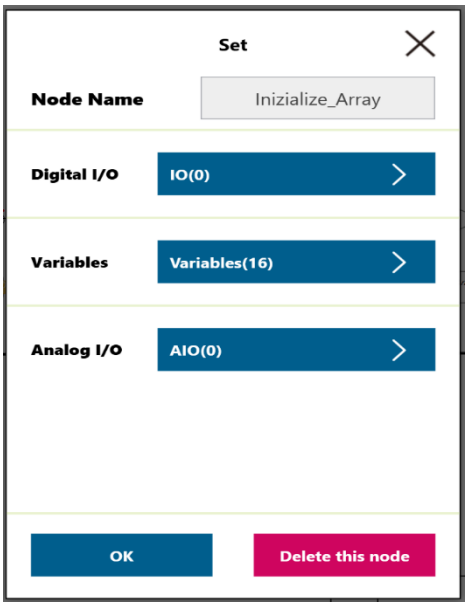


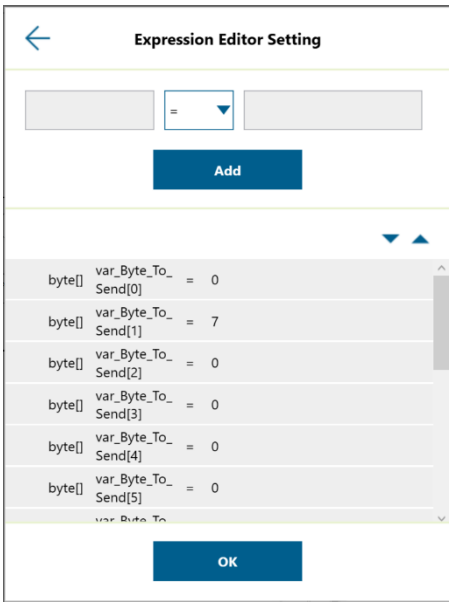


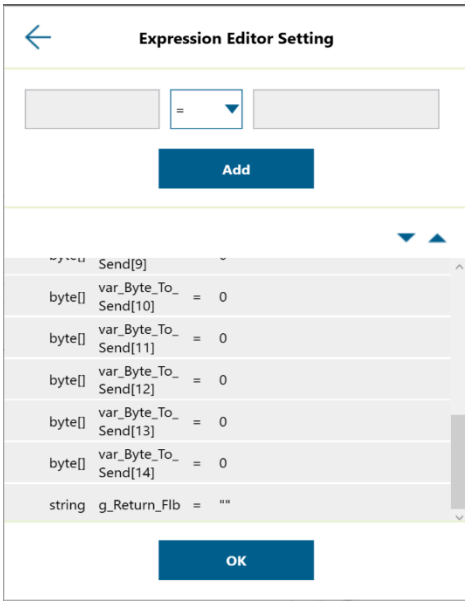
TM FLOW

STEP 4:

Initialisation:
The two local variables, the byte array and the Flb response will always be initialised here







IMPORTANT!
The default IP addresses is 192.168.1.10 in class B (Subnet Mask 255.255.0.0). The TCP/IP port is 7776 and the UDP port is 7775.

The correct syntax for each pack is:

Header		Description	Footer
Chr(0)	Chr(7)	Command (ASCII character vector)	Chr(13)

For each string sent to the Flexibowl®, an ECHO of the command received will be returned. If the string is interpreted correctly, the ECHO will be:

Header		ECHO	Footer
Chr(0)	Chr(7)	%	Chr(13)

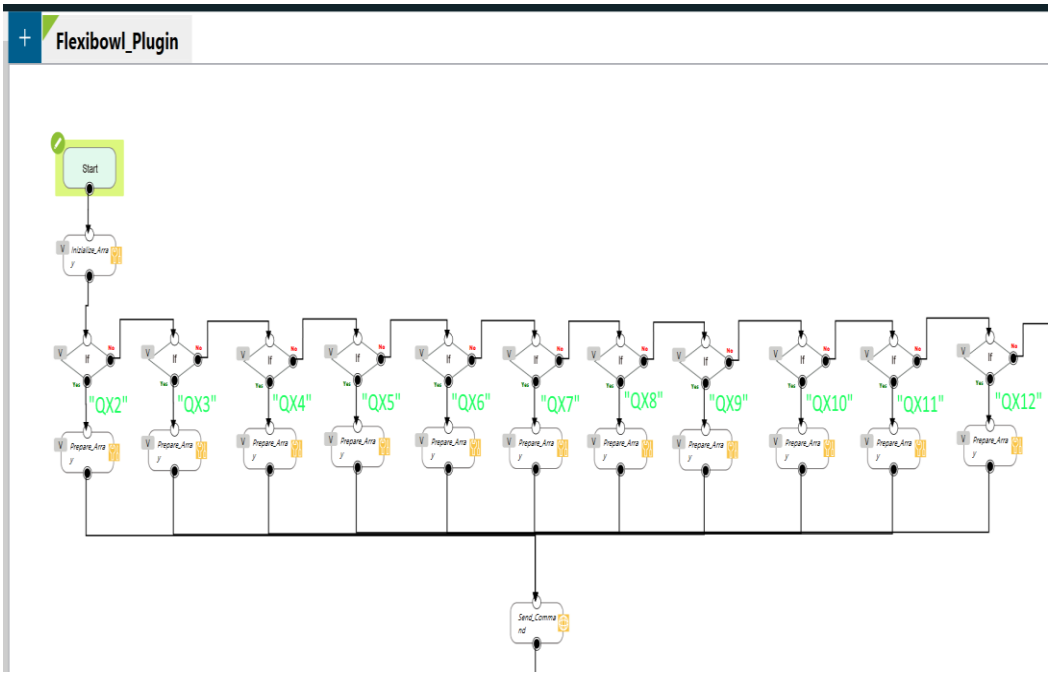
Otherwise if the string is not interpreted correctly, the ECHO will be:

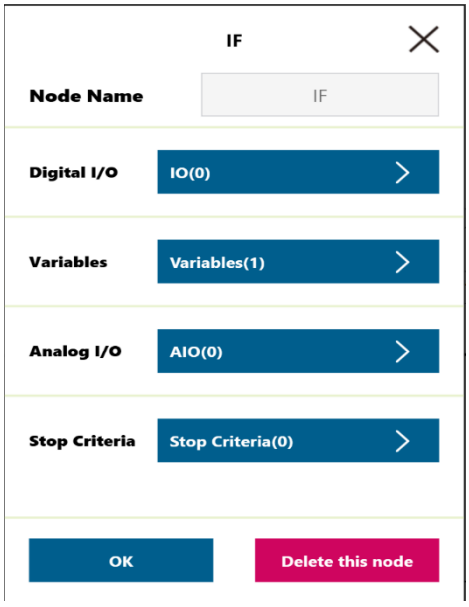
Header		ECHO	Footer
Chr(0)	Chr(7)	?	Chr(13)

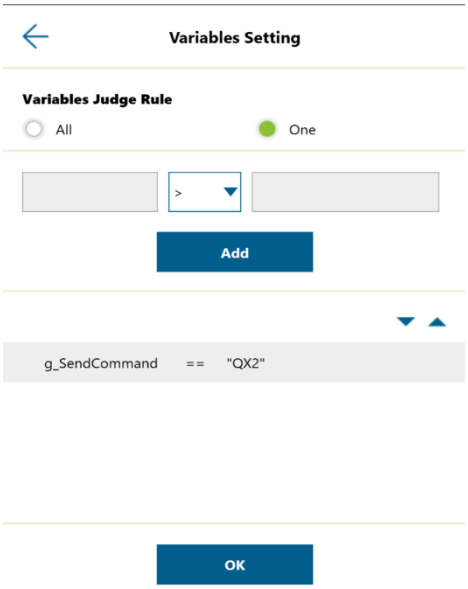
TM
FLOW

STEP 5:

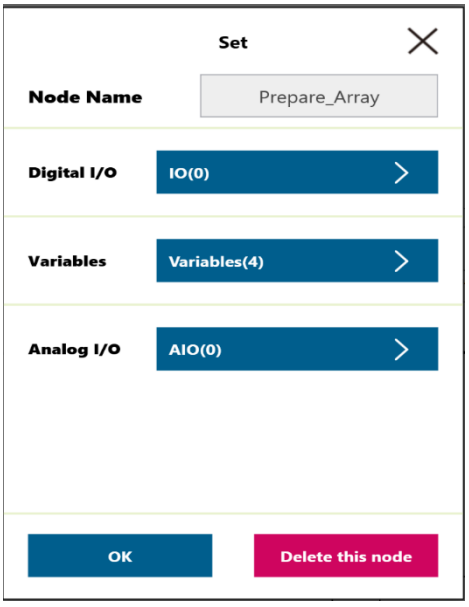
Send command movement:
The array to be sent to the Flb is prepared here according to the global variable where movement is set.

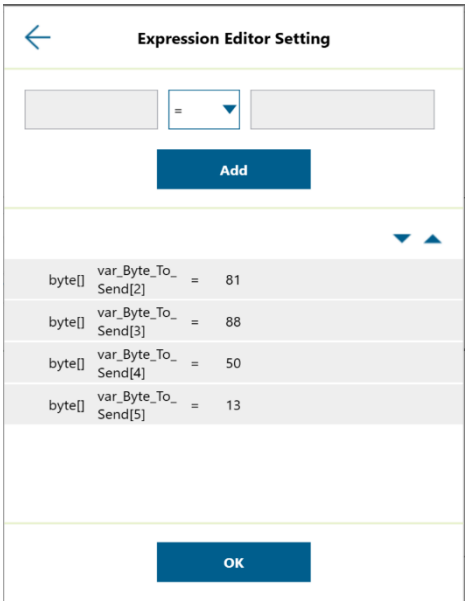


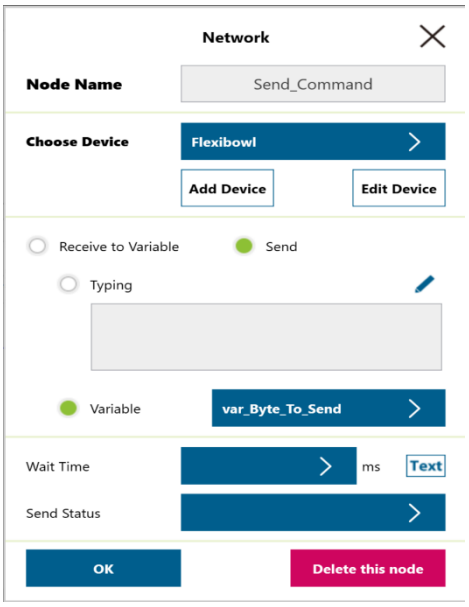


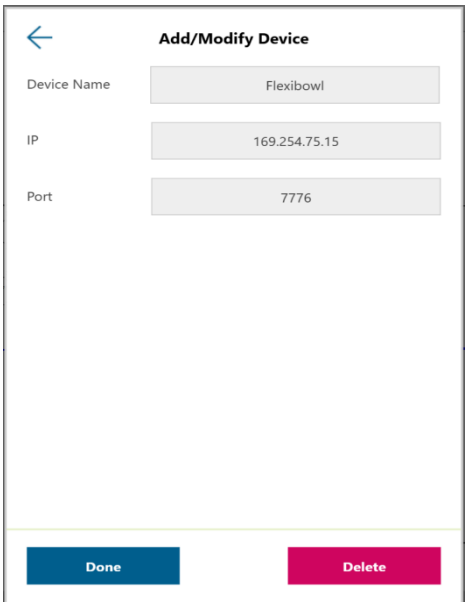


TM
FLOW


Set
Node Name
Prepare_Array
Digital I/O
IO(0)
Variables
Variables(4)
Analog I/O
AIO(0)
OK
Delete this node


Expression Editor Setting
=
Add
byte[] var_Byte_To_Send[2] = 81
byte[] var_Byte_To_Send[3] = 88
byte[] var_Byte_To_Send[4] = 50
byte[] var_Byte_To_Send[5] = 13
OK

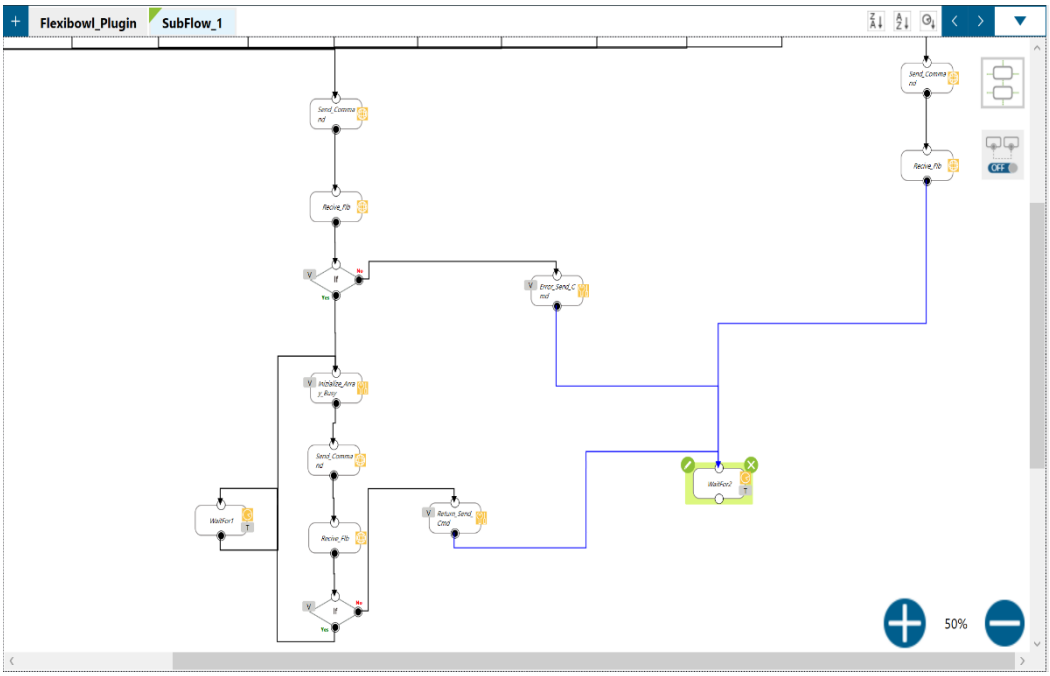

Network
Node Name
Send_Command
Choose Device
Flexibowl
Add Device
Edit Device
Receive to Variable
Send
Typing
Variable
var_Byte_To_Send
Wait Time
ms
Text
Send Status
OK
Delete this node

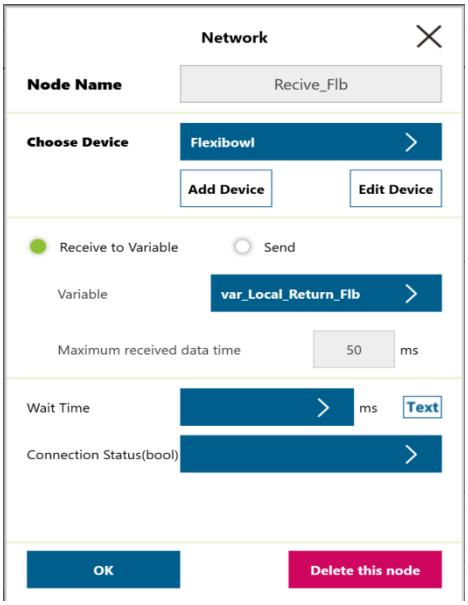

Add/Modify Device
Device Name
Flexibowl
IP
169.254.75.15
Port
7776
Done
Delete

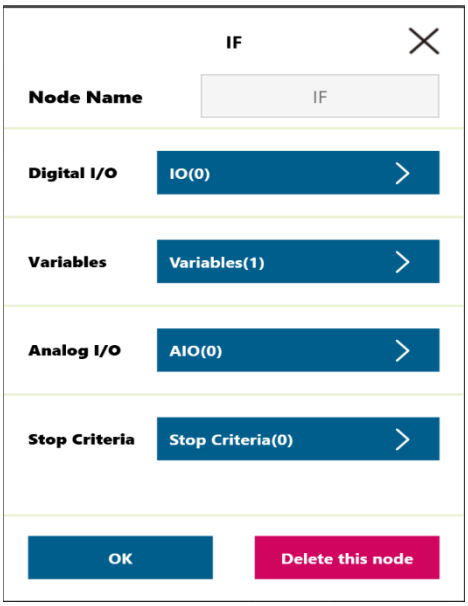
TM
FLOW

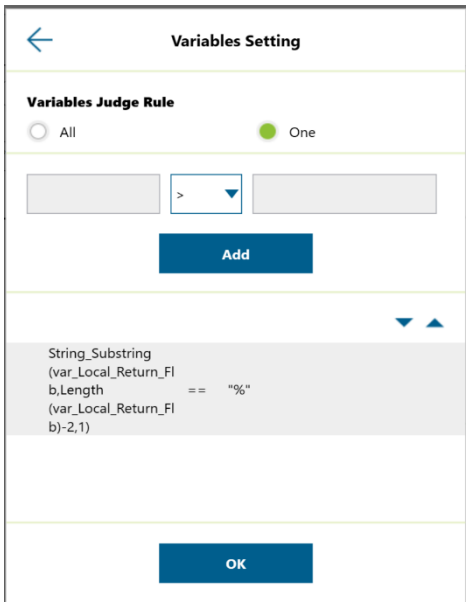
STEP 4:

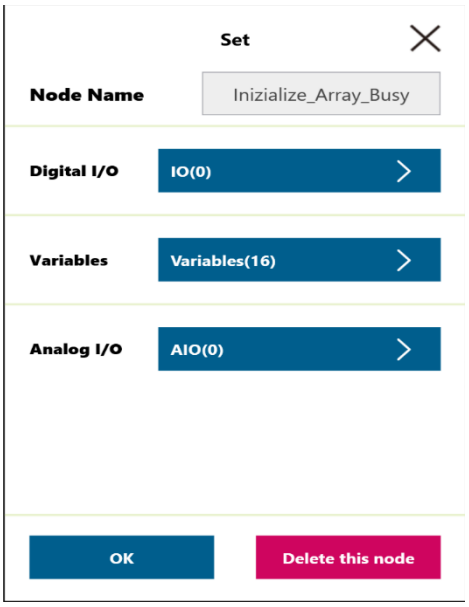
Wait End Movement:
Once the Flb movement command is sent, we will wait for the movement to stop in this small sub-program. To do so the “IO” request must be sent and the Loop response must be analysed





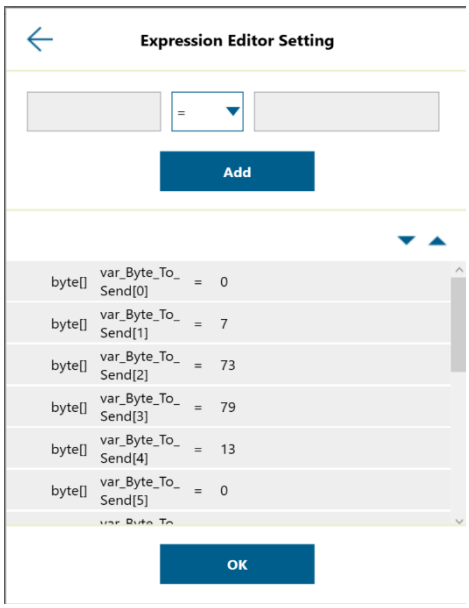


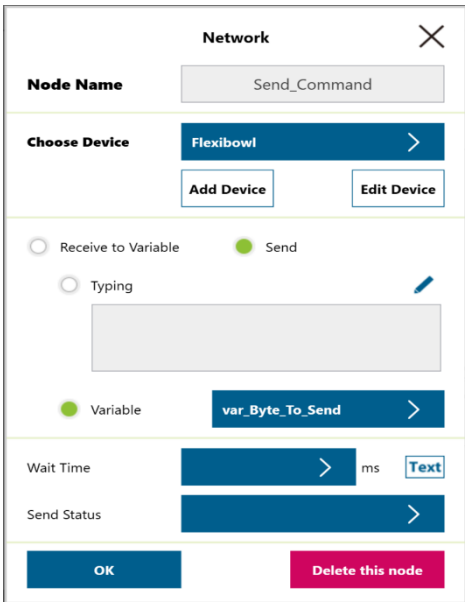




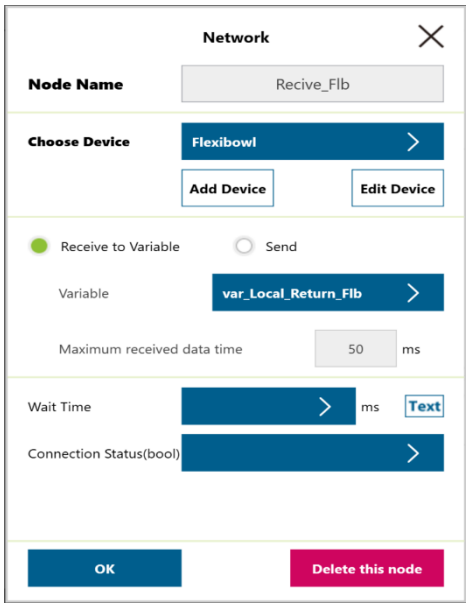
If the % symbol is not returned in the previous If, the FLB has not interpreted the correct command, this means we have encountered an error and will write Return_Flb = “Error Send Command” on the global variable.

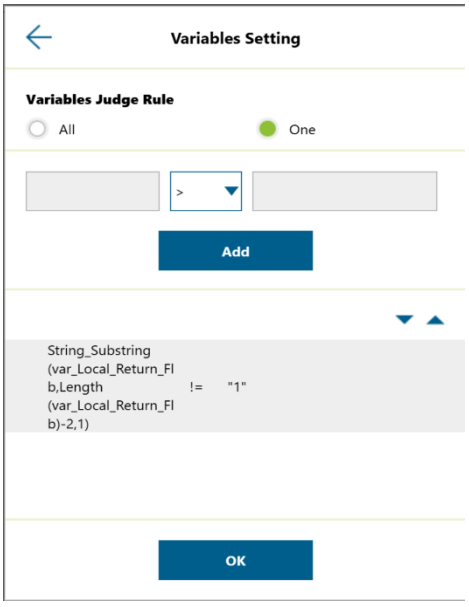
Send the IO command to understand if the Flb is moving.

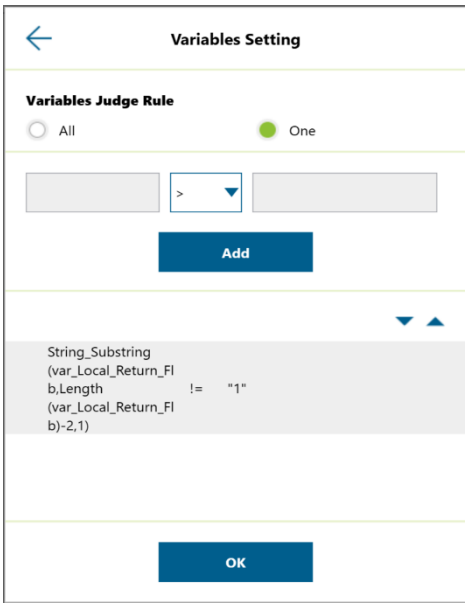




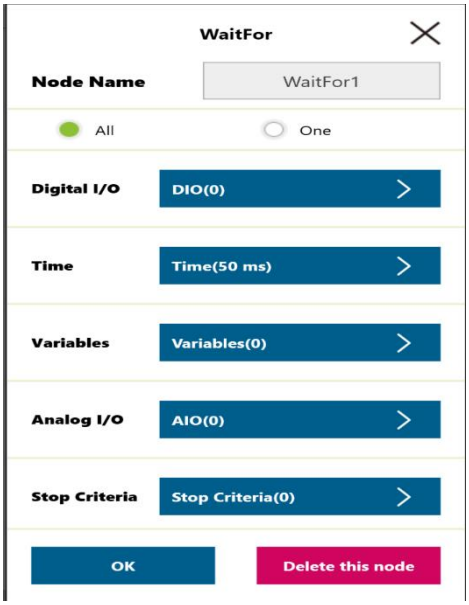
TM
FLOW



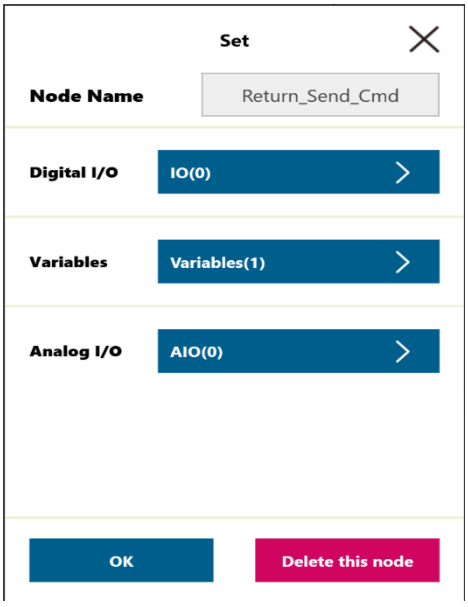


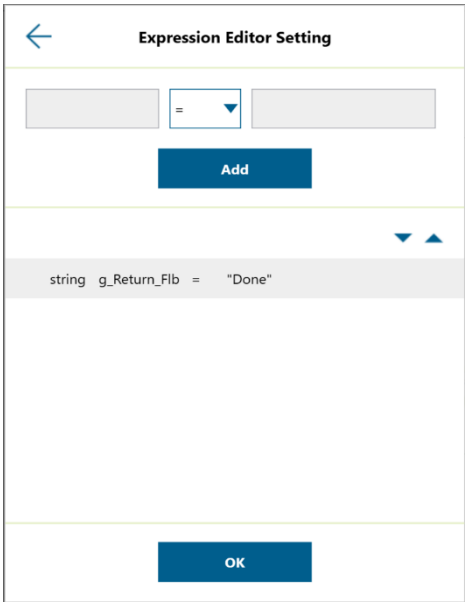


If SubString == 1 the movement is finished, if <> gives 1 the flexibowl is still moving, therefore we will take a small pause (50ms) and interrogate the Flb once again.



Once the movement is finished we will write “Done” on the Return_Flb global variable



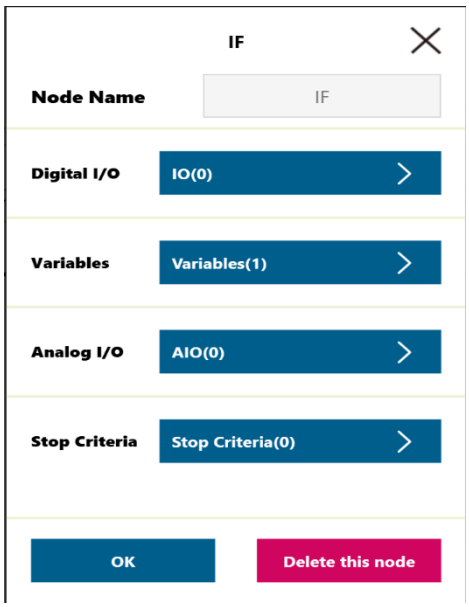


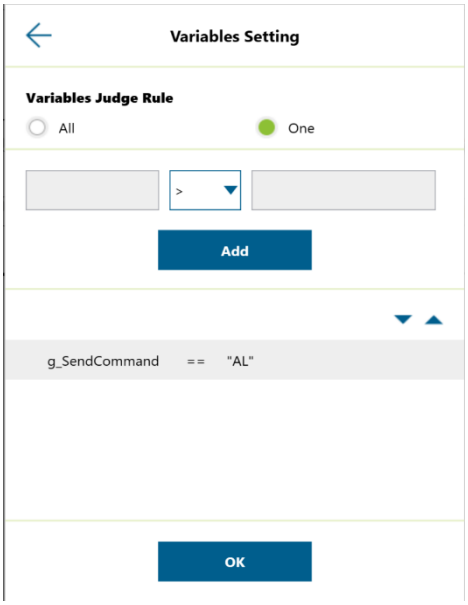
TM
FLOW

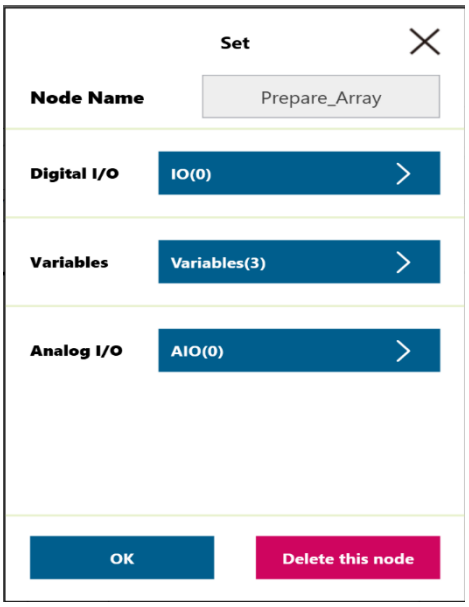
STEP 5:

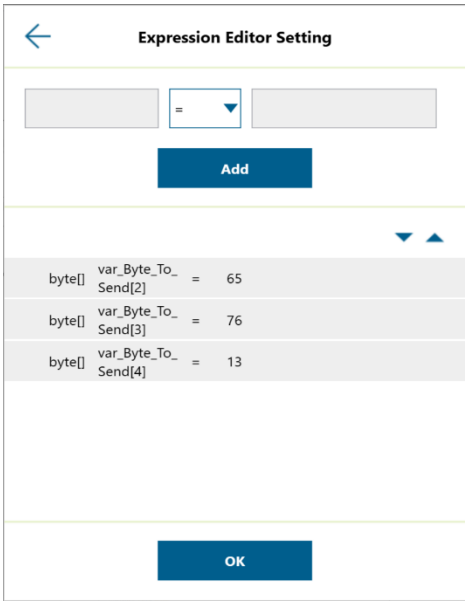
Alarm Request:

If we send the “AL” command to the Flb it will tell us if there are active alarms, refer to the Flb manual for details.

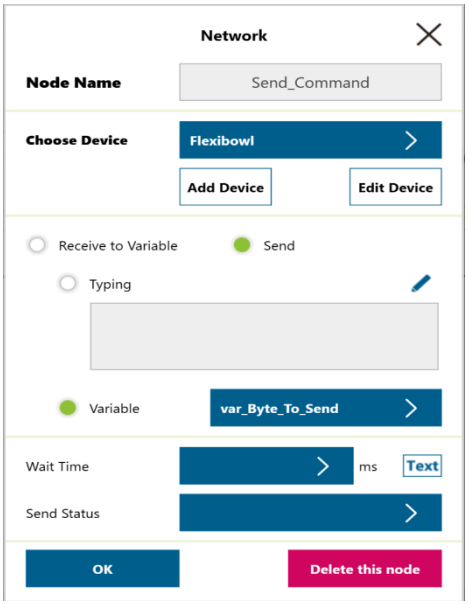


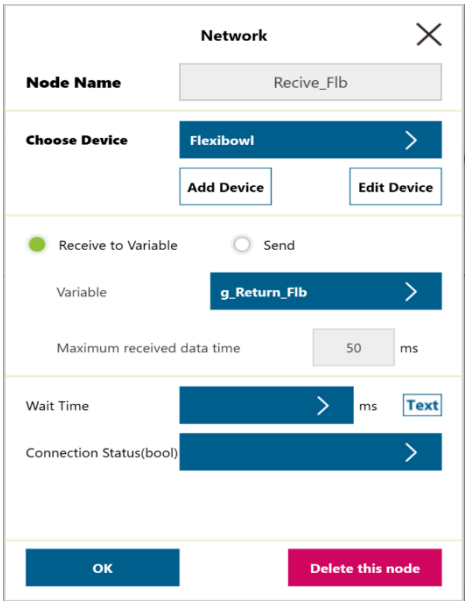






TM
FLOW


Network
Node Name
Send_Command
Choose Device
Flexibowl
Add Device
Edit Device
Receive to Variable
Send
Typing
Variable
var_Byte_To_Send
Wait Time
ms
Text
Send Status
OK
Delete this node


Network
Node Name
Recive_Flb
Choose Device
Flexibowl
Add Device
Edit Device
Receive to Variable
Send
Variable
g_Return_Flb
Maximum received data time
50
ms
Wait Time
ms
Text
Connection Status(bool)
OK
Delete this node