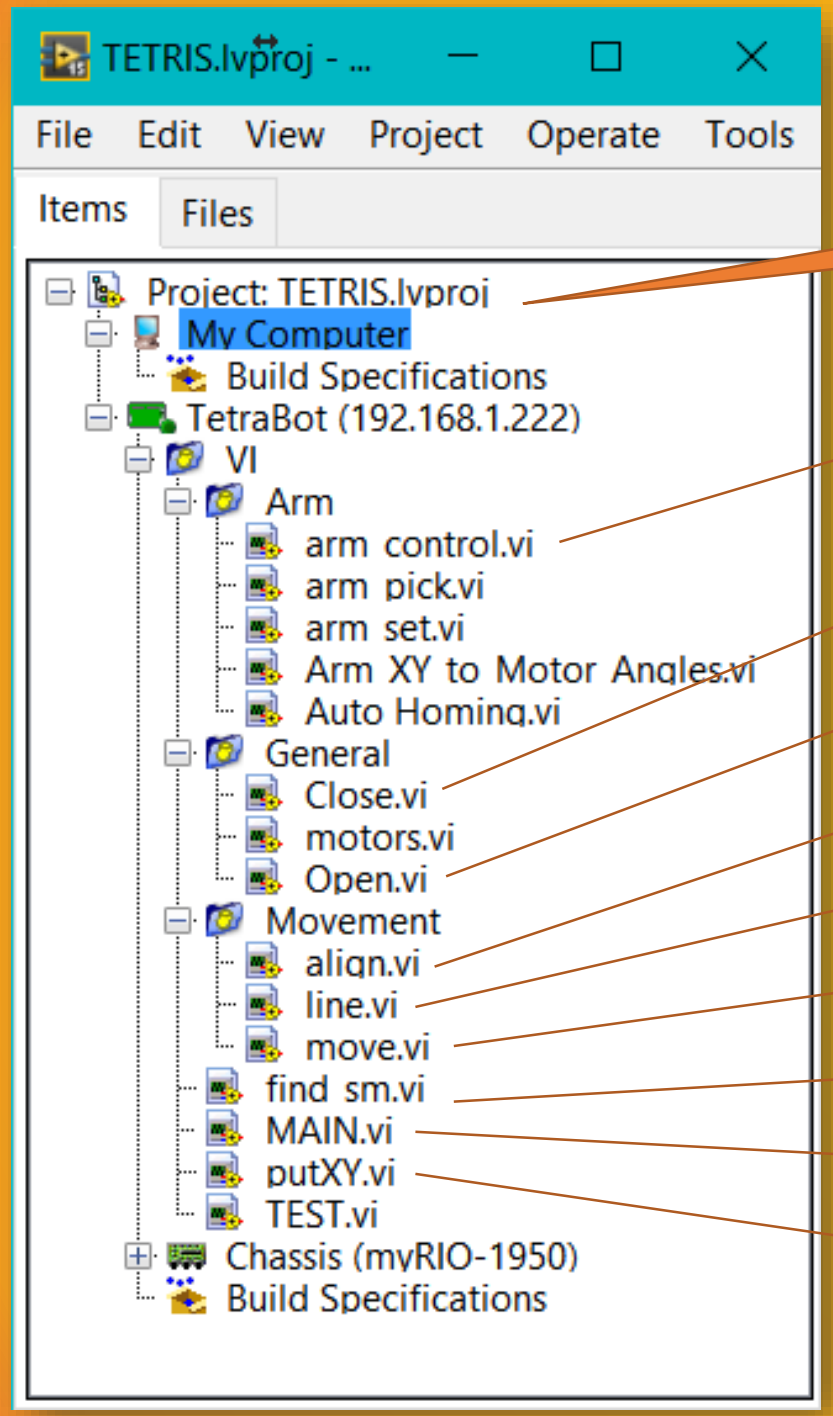
WRO 2017
Advanced Robotic Challenge

Solution Program Introduction

Made by *Alex Tkachenko*

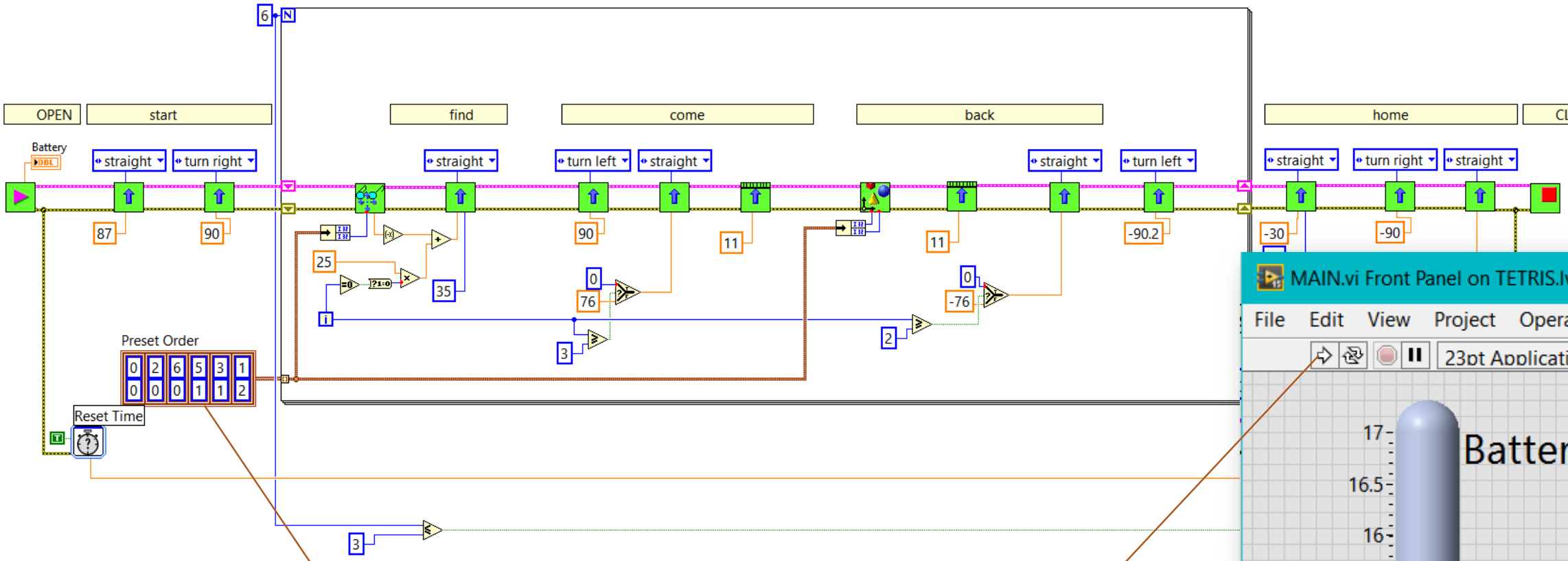


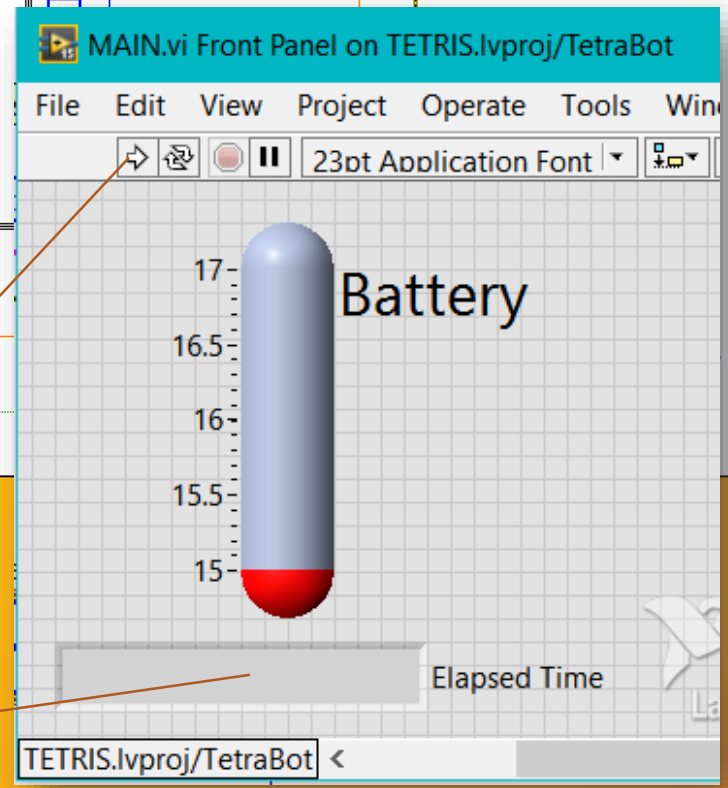
TETRIS LabVIEW Project

VI's you need are listed below

- Move arm
- Close robot connection
- Open robot connection, Initialize
- Push Robot to the border
- Go to the line
- Move robot
- Find next object and pick it up
- MAIN Program, Run it
- Put the object into the stacking form

MAIN program



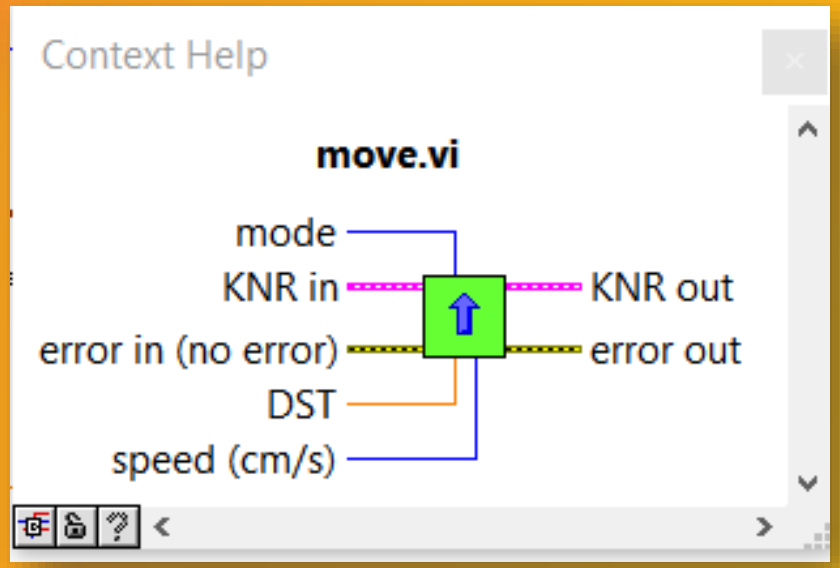


Coordinate order of the objects in the stacking form *(see later)*

Run

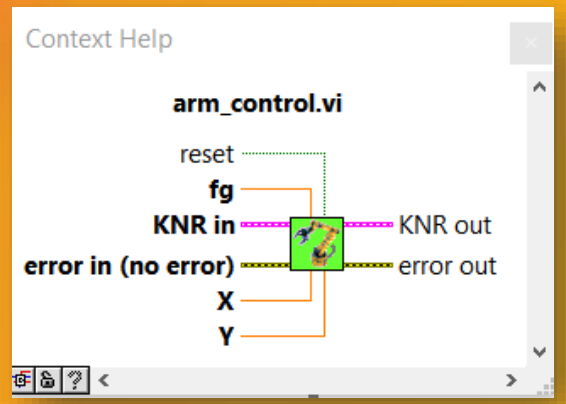
Battery Level

Run time

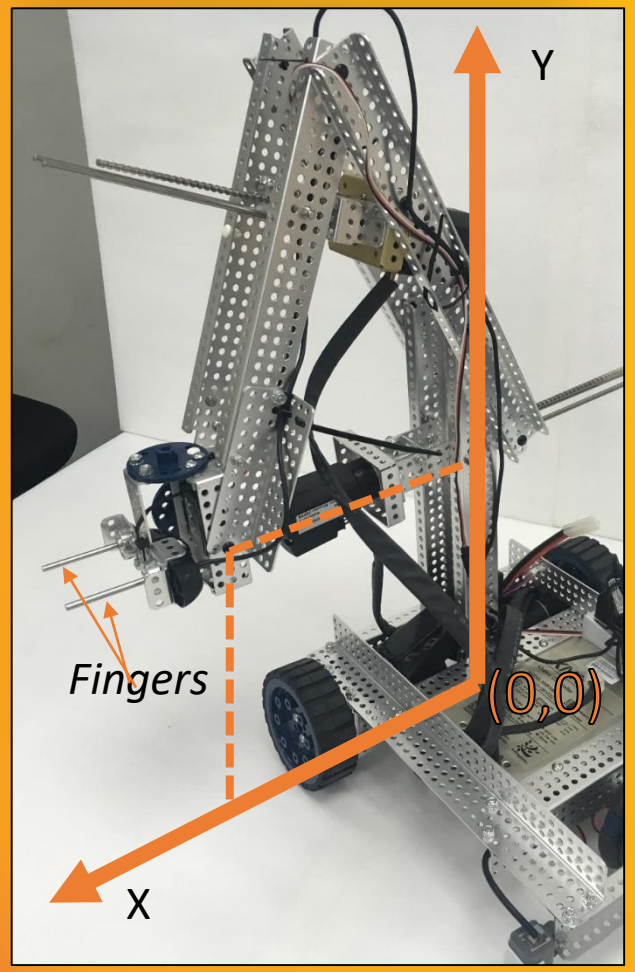


Move the robot

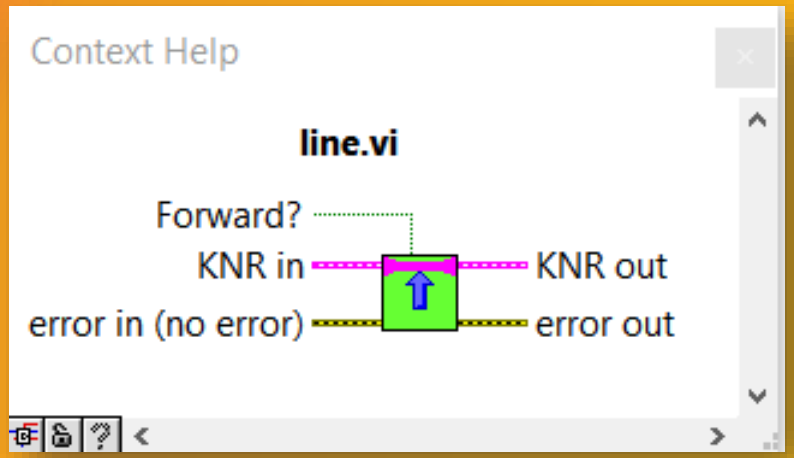
Input / Output	Value
KNR in/out, error in/out	Connect to other blocks.
mode	Movement type. Right click -> Create -> constant/control. Type enum. Values are <straight, turn left, turn right>.
speed	Movement speed in centimeters (cm) per second. Any value in range [0, 50] . Example: 0 – stop, 10 – slow, 50 – fast.
DST	In *straight* mode – distance in centimeters (cm) to travel. In *turn* mode – turn degree. Only one wheel is moving. Example: 180 - turn around Negative (-) value means another direction.



Move arm

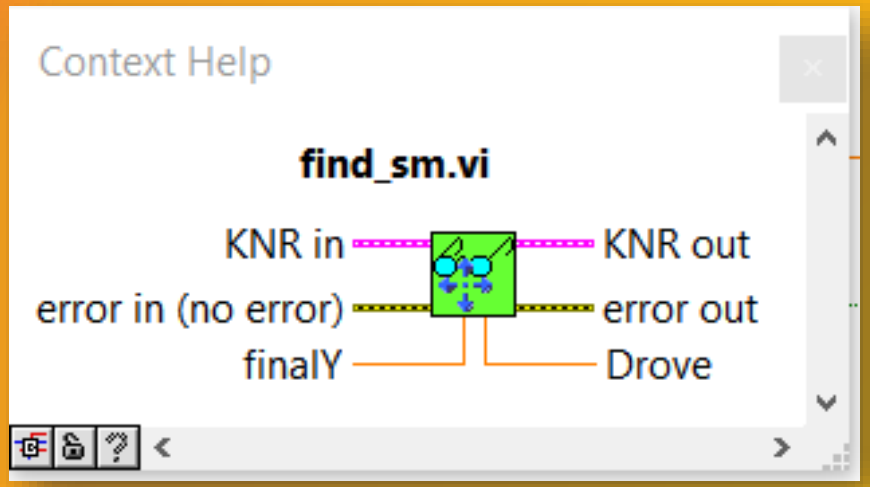


Input / Output	Value
KNR in/out, error in/out	Connect to other blocks.
X,Y	X,Y coordinate in the robot coordinate system in (cm). Arm would be moved there.
fg	Final absolute position of the *fingers.* *Ex: 0 – horizontal (as on picture).* *Ex: -90 – look down*



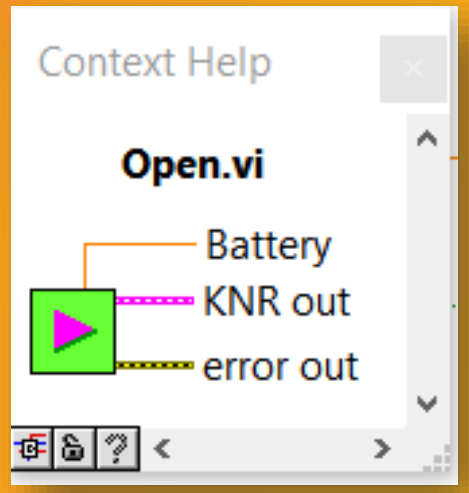
Move till light sensor detects a **black** line

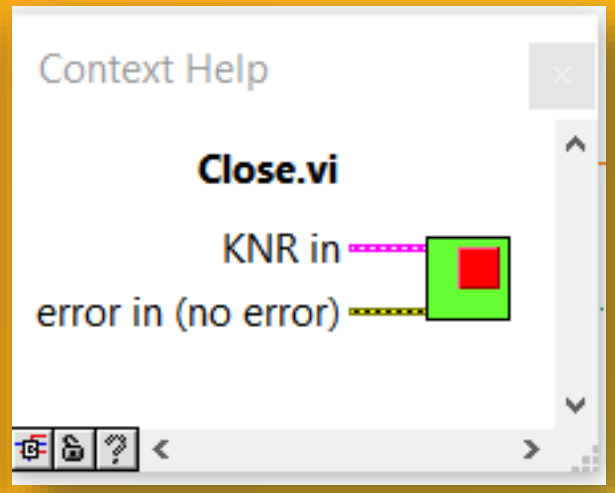
Input / Output	Value
KNR in/out, error in/out	Connect to other blocks.
Forward?	TRUE – move forward FALSE – move backward



Moves, detects the object and picks it up

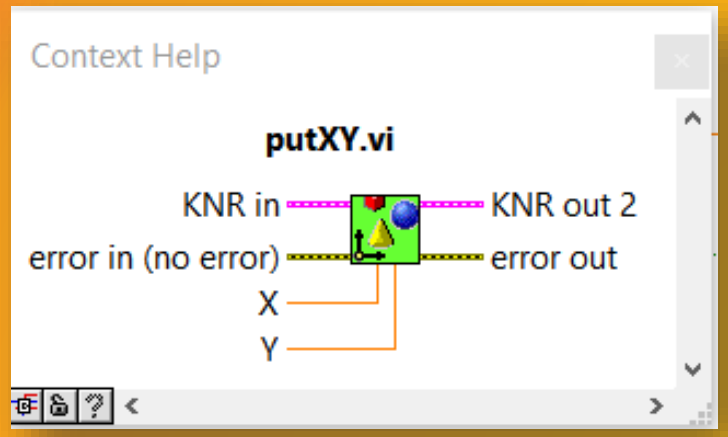
Input / Output	Value
KNR in/out, error in/out	Connect to other blocks.
FinalY	In the end raises the object to the desired level in the *stacking from coordinate system (Discussed further)*
Drove	The distance the robot travelled to find the objects. Is used to calculate the way back.





Opens/ closes connection

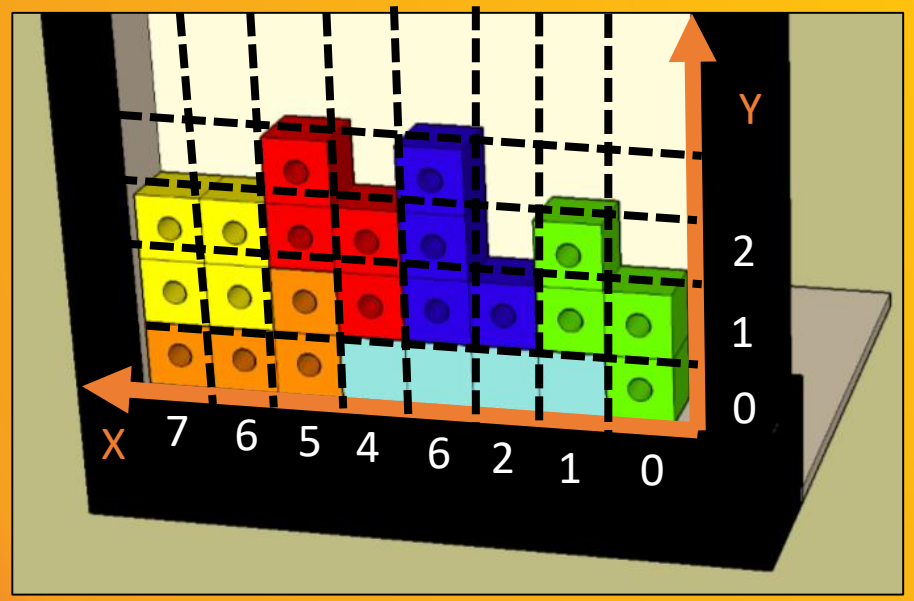
Input / Output	Value
KNR in/out, error in/out	Connect to other blocks.
Battery	Battery charge level

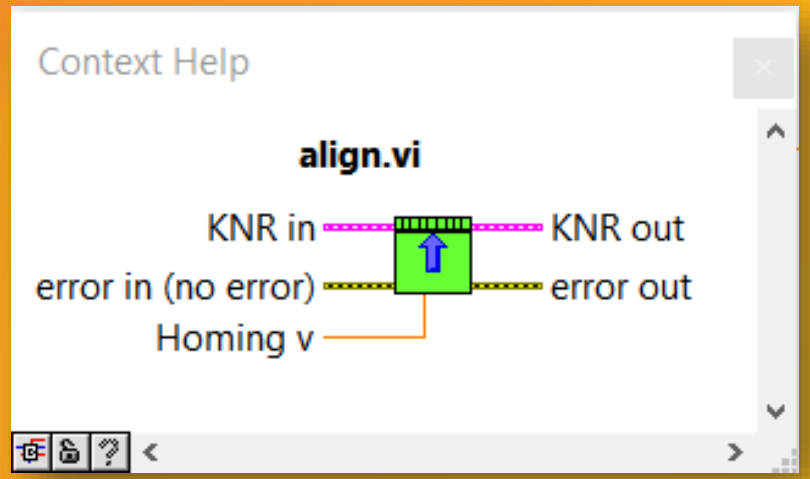


Puts the object into the stacking form

Input / Output	Value
KNR in/out, error in/out	Connect to other blocks.
X,Y	X and Y coordinate in the *stacking form coordinate system*

Stacking form coordinate system





Pushes the robot to the border

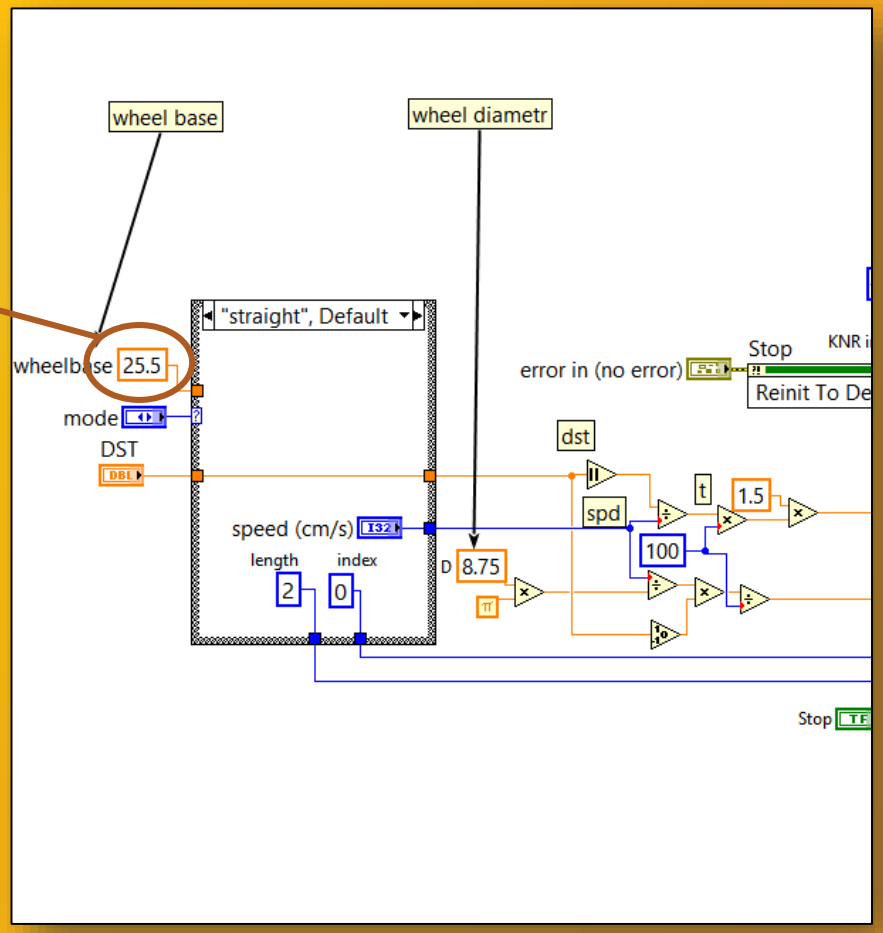
Input / Output	Value
KNR in/out, error in/out	Connect to other blocks.
Homing v	Power on the motors (out of 100%).

Troubleshooting

1. Robot **turns** not exactly 90°

Your wheel base might be slightly different.
Wheelbase – distance between wheels.
Turns too much? Decrease value
Turns not enough? Increase value

Move.vi



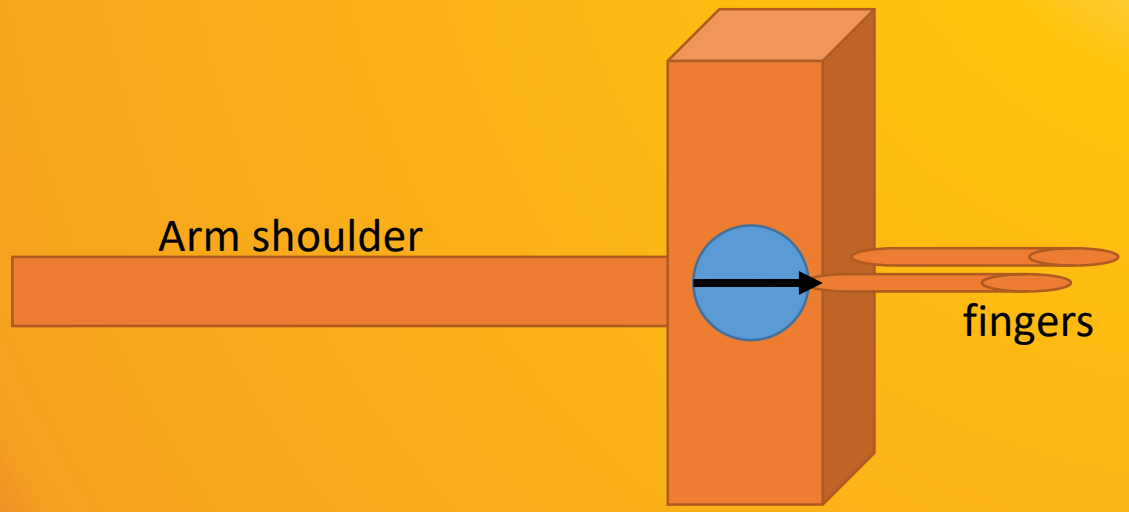
2. Robot **finger** position is not calibrated

Calibrated fingers work as follows:
Middle RC position (1500) means fingers are aligned parallel with the arm shoulder
However, this value may not be exactly 1500, set your own value there.

Arm XY to Motor Angles.vi

```
p3 = max(min(p3, 16.5), 3.4) -3.3;
p4 = max(min(p4, 16), 6) - 5.9;
float64 rev3 = p3 * 2.5;
float64 rev4 = p4 * 2.5;

uInt16 fgout = 1476 + (1600/180)* (-270-fg+(alpha+beta)*180/pi);
fgout = max(min(fgout, 2300), 700);
```

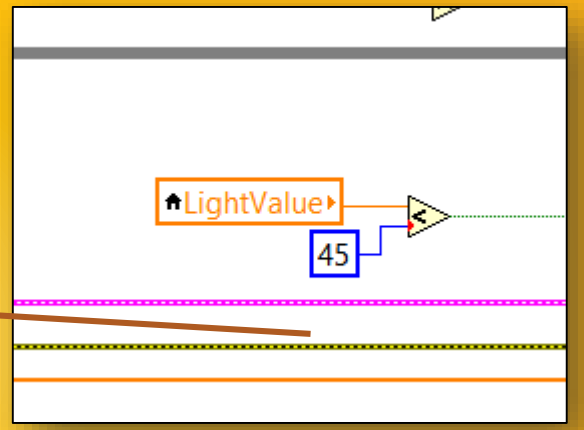


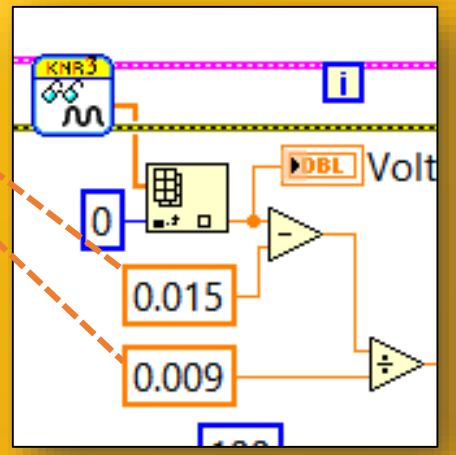
3. **Light sensor** has problems detecting the line

Different light sensors may have different sensitivity.
At first, try to change black threshold.
If this doesn't help, try to check setup values. First number is the voltage on white, second – range from white to black.

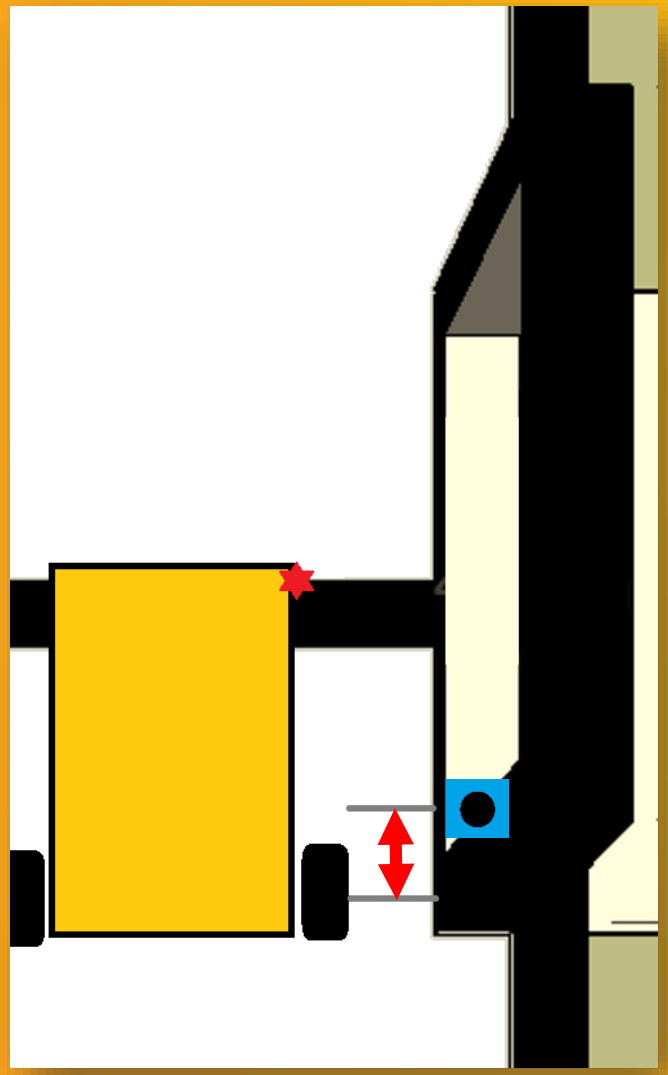
Also check the connection at the analog port.

Line.vi



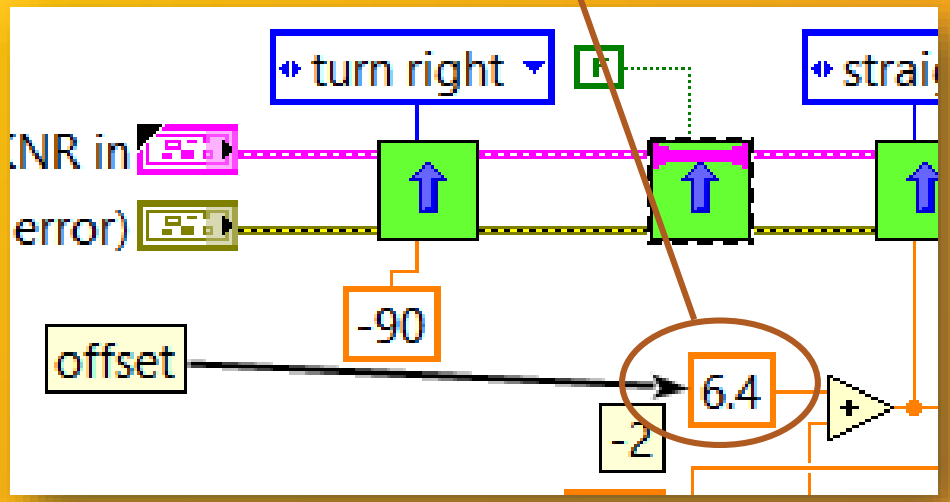


4. Robot doesn’t put the object in the right position at the stacking form



After we found the line, there is a starting offset (in cm) between the **current position** and the center of the **first cell** in the *stacking form coordinate system.* This value may differ, if your stacking form is slightly misaligned with the central black line.

putXY.vi



More at
http://www.knrrobotics.com

Made by *Alex Tkachenko*