

ROLL'S ROYCE

Members

Hwi-jun Koo
Sung-min Park
Hyun-bin Kim
Hyeon-seong Kim
Jae-ho Lee
Jeong-ha Lee
Je-gyeong Cho

Professor

Hae-won Park



Contents

Final Design and Analysis Results

System Design
Motor Control
Vision Processing
System Integration

Prototype Demo Video

Mission Completion



How Can We Design a Good Robot?

Faster!

Never Stop

Move Faster

No Avoiding Motion

Safer!

Safe Path Algorithm

TCP/IP Error Handling

Fault Pickup Handling

Fast Calculation



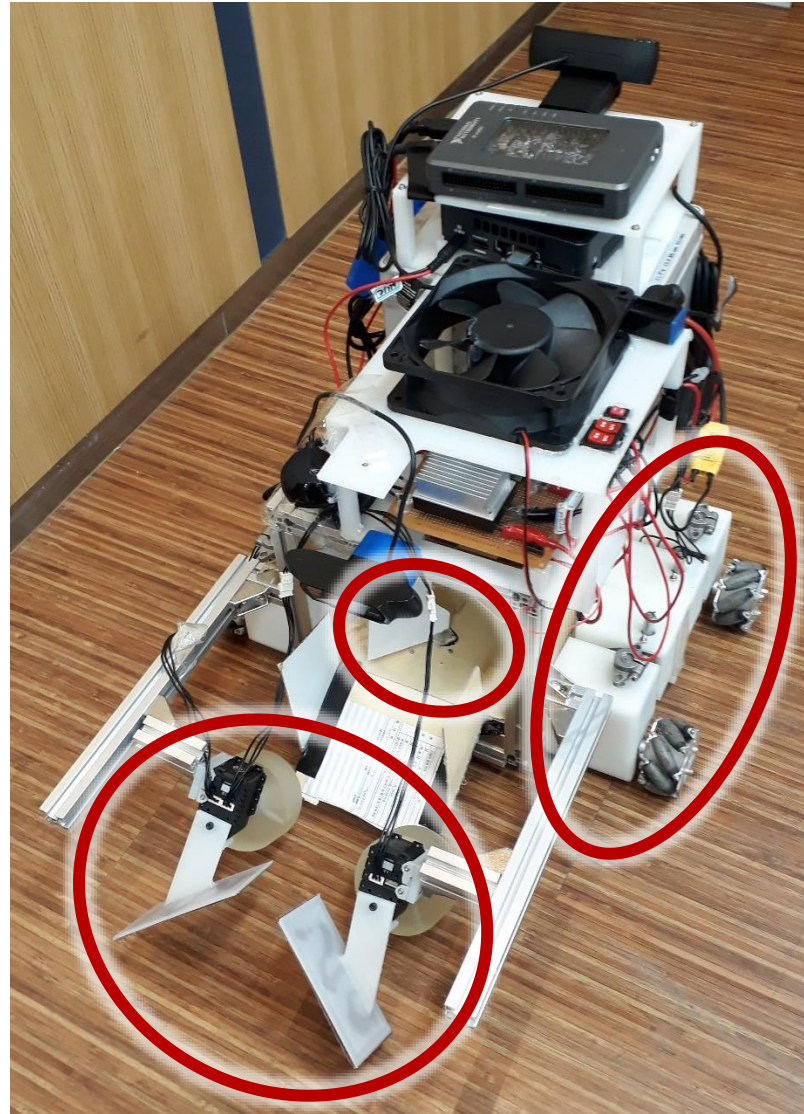
Overall Design

Faster!

Rollers

Gears

Sorting Board



Safer!

Safe Path Algorithm

TCP/IP Error Handling

Fault Pickup Handling

Fast Calculation



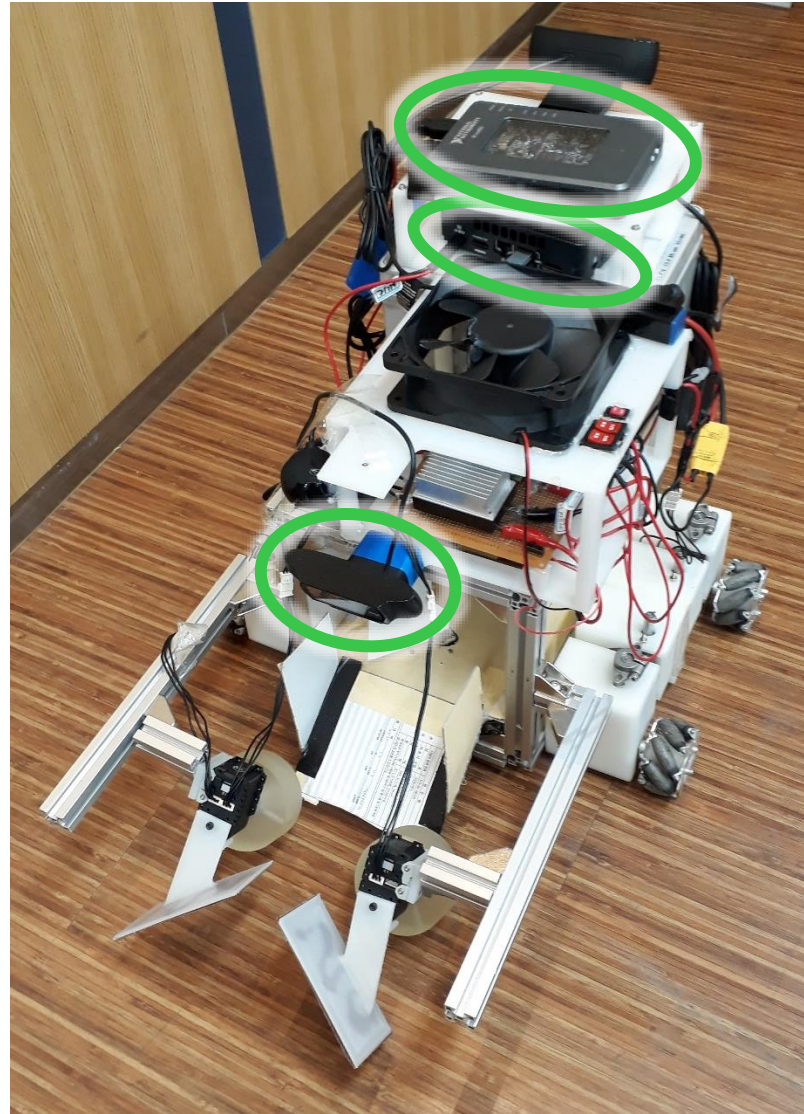
Overall Design

Faster!

Rollers

Gears

Sorting Board



Safer!

Path Decision

Multiple TCP/IP

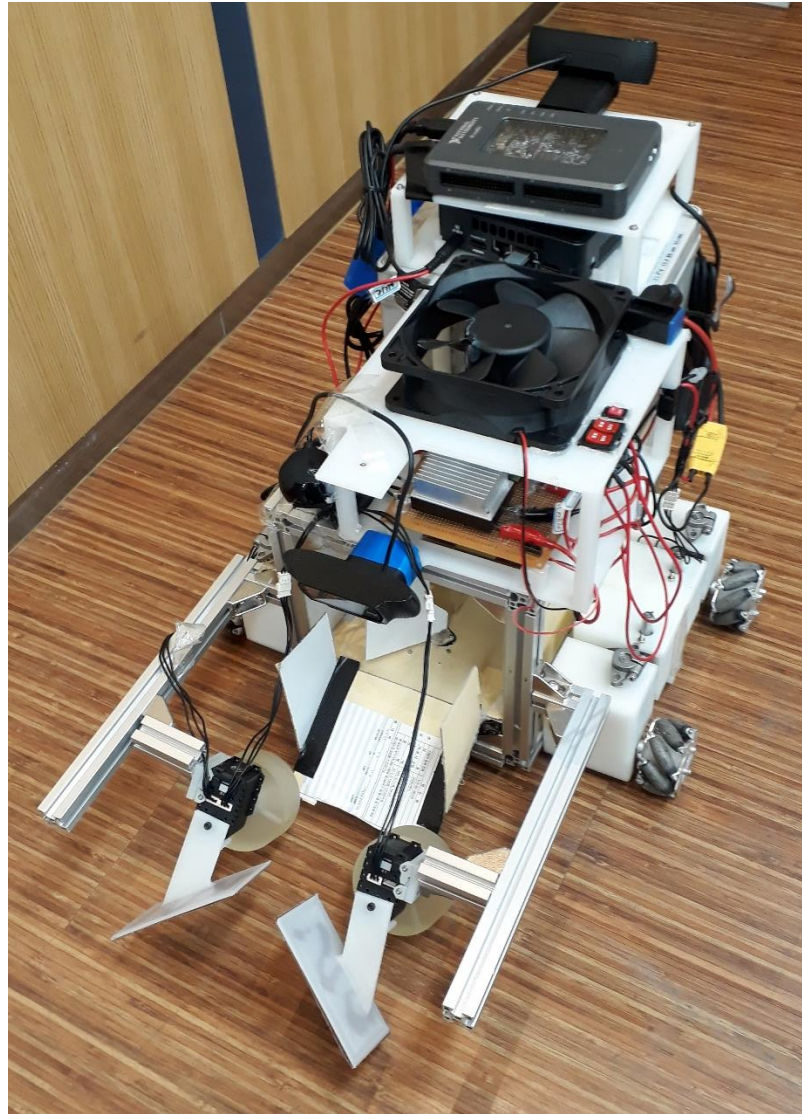
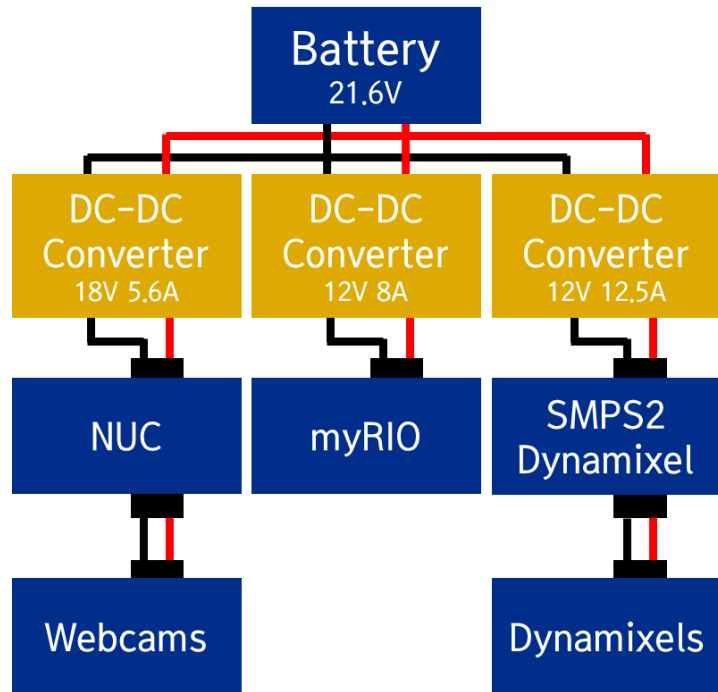
Emergency Pickup

Resolution of Webcam

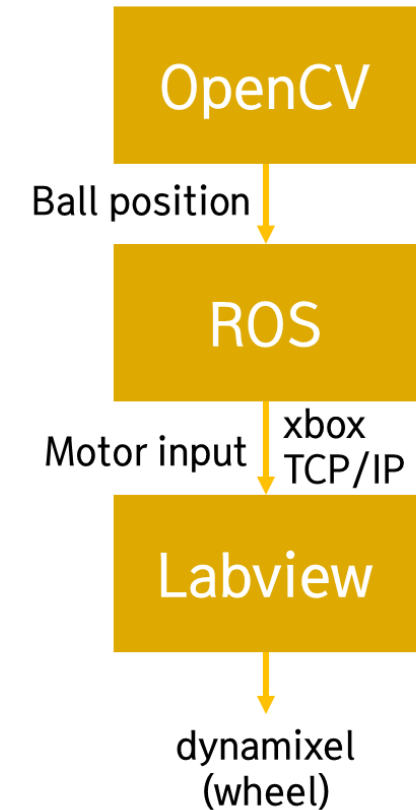


Overall Design

PMS Circuit



Data Line





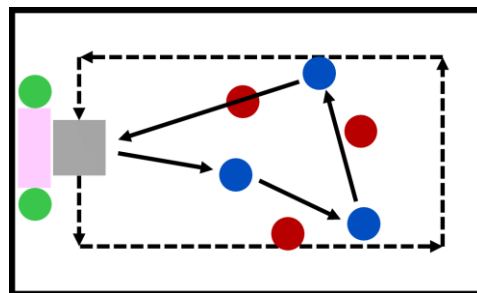
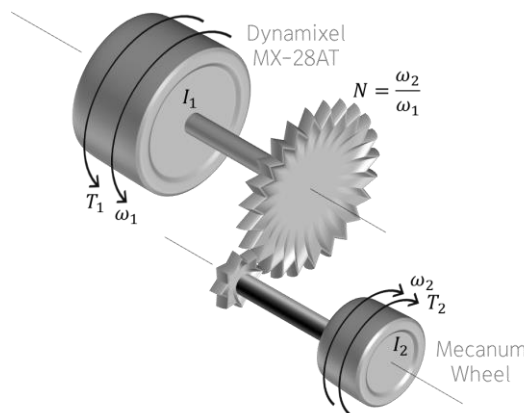
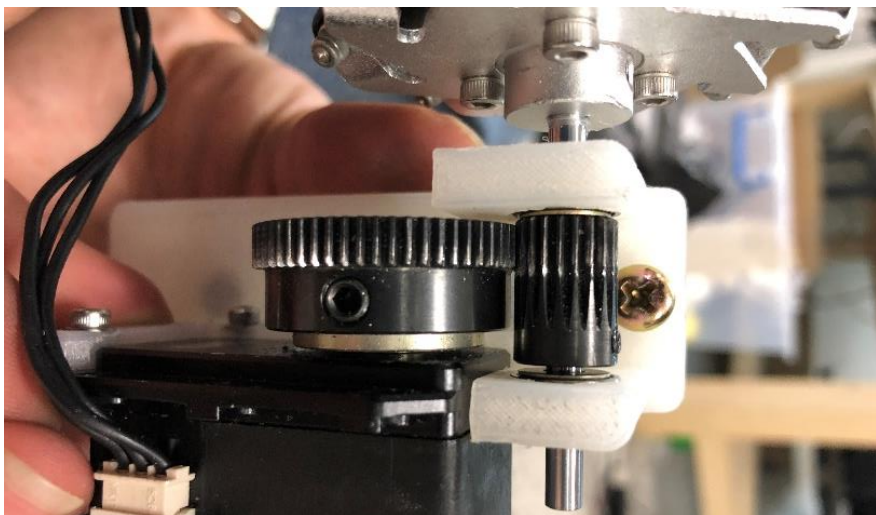
System Design





Gears

Problem Robot's translation motion is too slow!



From Dynamixel MX-28AT data spec,
Maximum no load speed $\omega_{\text{no load}} = 55\text{rpm}$
Let safety factor $S.F. = 3$
Radius of mecanum wheel $r_w = 45\text{mm}$

Possible maximum length of path $\sim 12\text{m}$
Target mission completion time $t \leq 50\text{sec}$
Target maximum speed $v_{\text{max}} = 0.24\text{m/s}$

Apply gear ratio = 3:1
Then maximum wheel speed $v_{w,\text{max}} = 0.26\text{m/s}$

Gear Ratio = 3:1

Max. Wheel speed = 26 cm/s

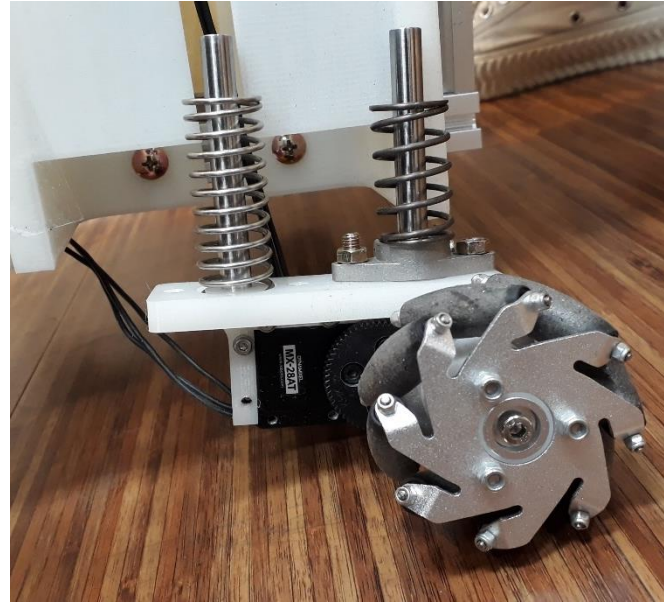
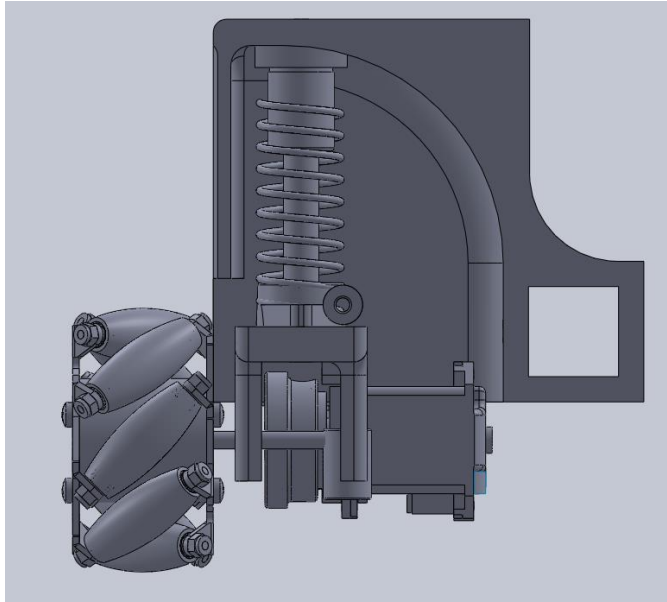
Using gears will make robot move faster!



Suspension

Problem

- Vibration & pitching motion occurs!
- 3 point support is not desired!



2 springs for each wheel:
 $k = 0.98\text{kN/m}$

Springs are connected in parallel
for 4 wheels:

$$k_{\text{net}} = 4 \times (2k) \\ = 7.84\text{kN/m}$$

Total mass of the robot:
 $M = 10.1\text{kg}$

Pressed height:

$$h = \frac{Mg}{k_{\text{net}}} = 12.6\text{cm}$$

Using springs will improve traction!

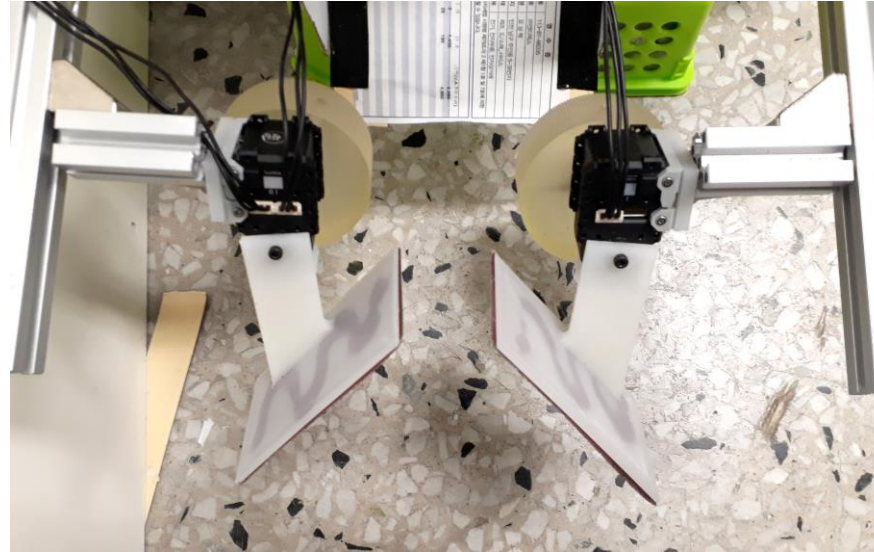
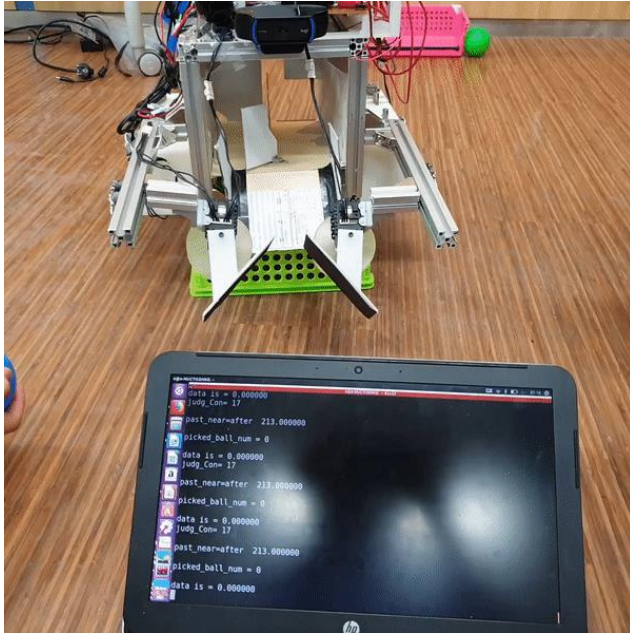


System Design

Function Design

Rollers

Problem Robot must not stop to pickup the balls!



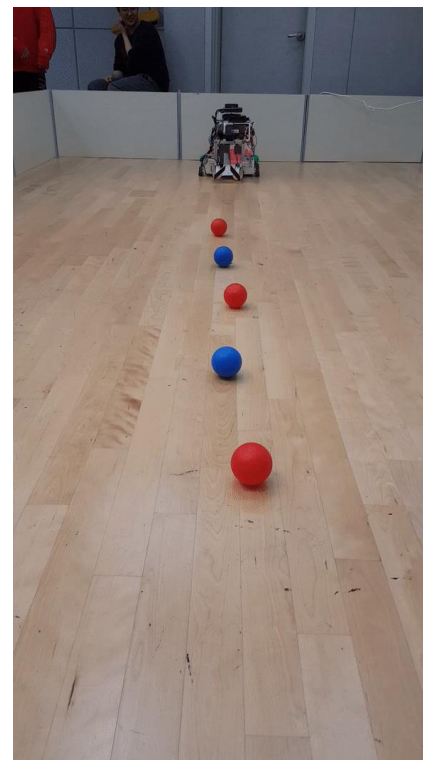
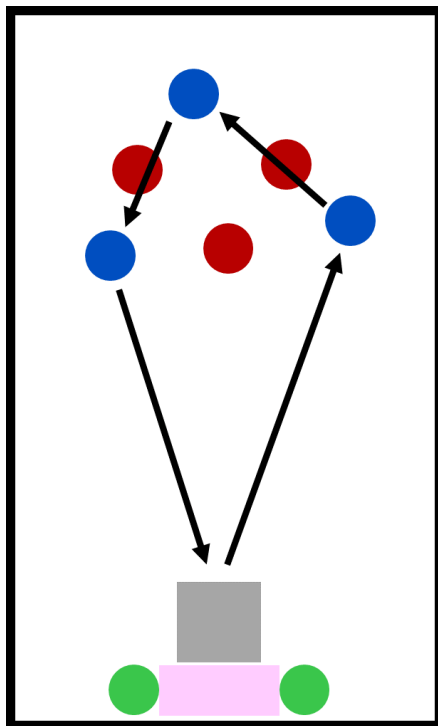
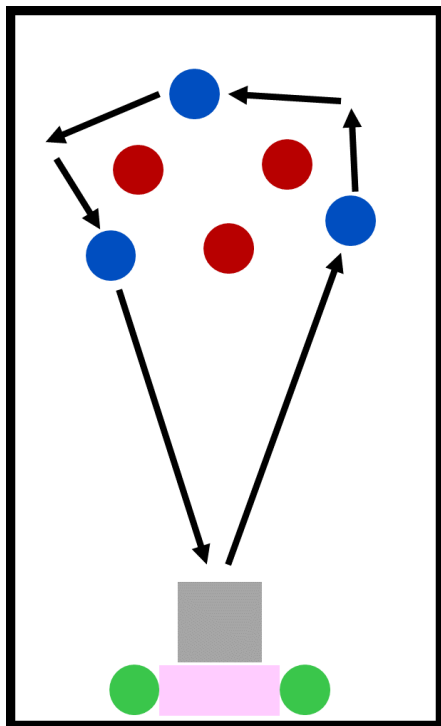
Robot doesn't have to stop to pickup balls
if we use 2 rollers!



Sorting Board

Problem

If robot avoids red balls, path gets longer!



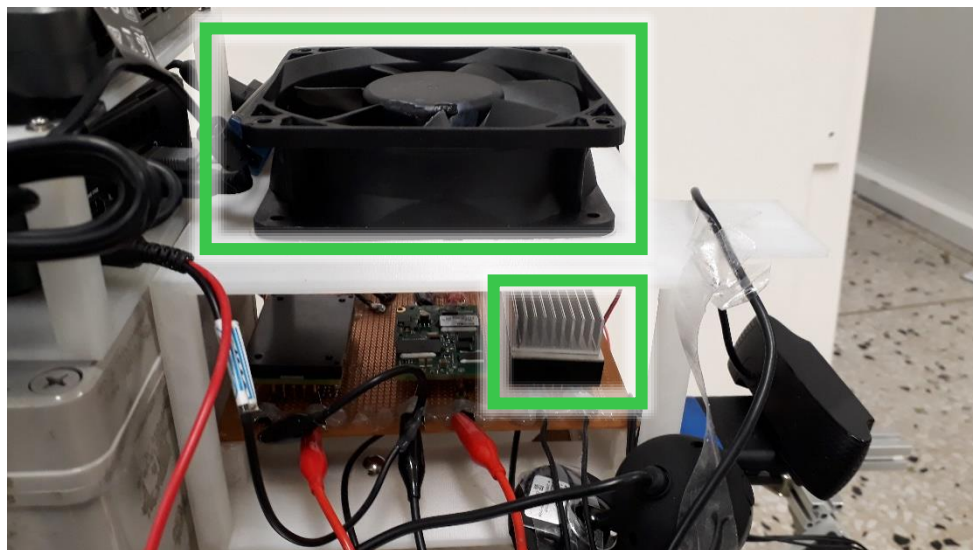
Picking both colored balls will make path shorter!



Fin & Fan

Problem

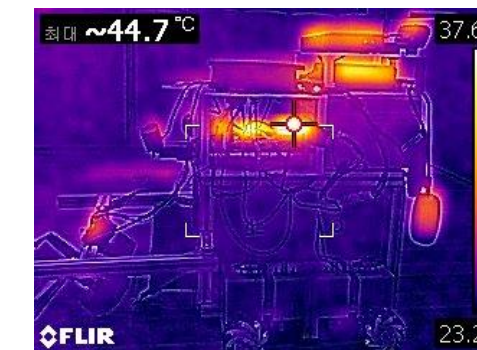
Components of PMS circuit gets too hot (especially DC-DC converters)



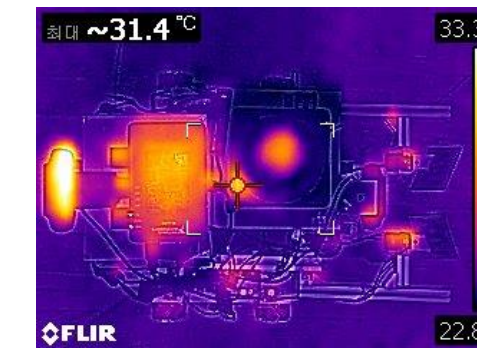
No Fan
No Fin



Only Fan
(No Fin)



Only Fin
(No Fan)



Both
Fin & Fan



Using both fin & fan cools down the system enough!



Motor Control



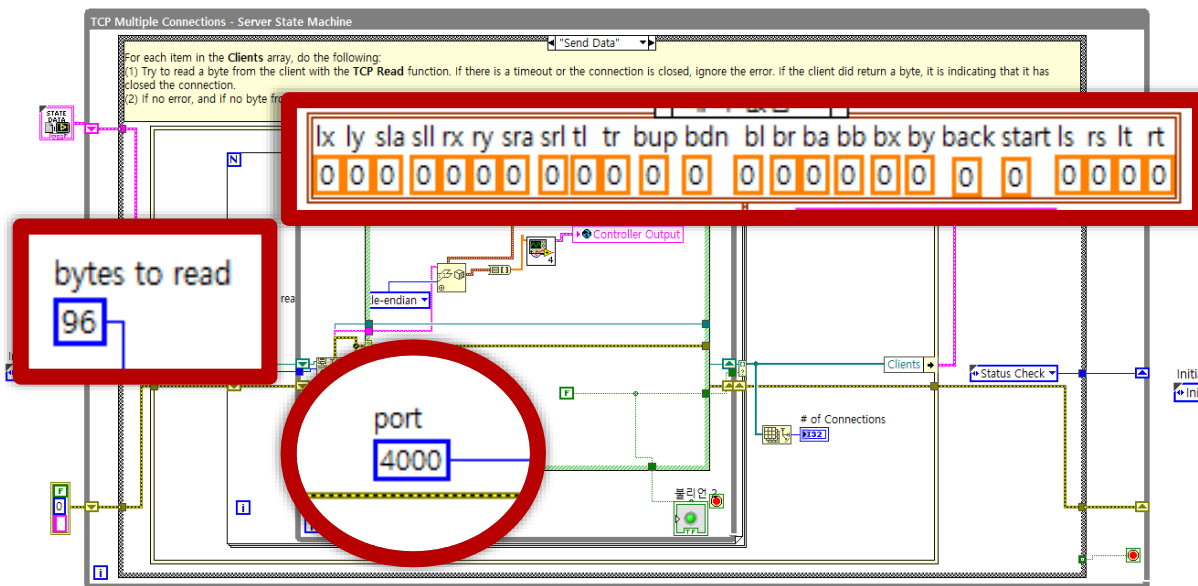


2 TCP/IP Loops

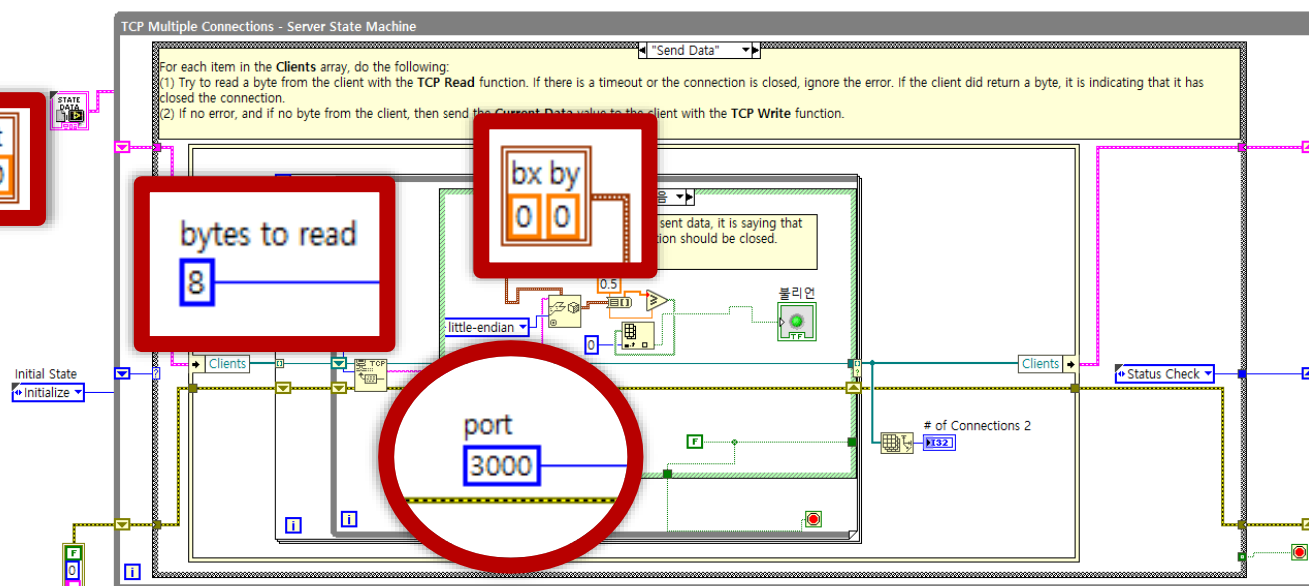
Problem

Sending data for sorting might be delayed!

Loop sending less data will iterate faster

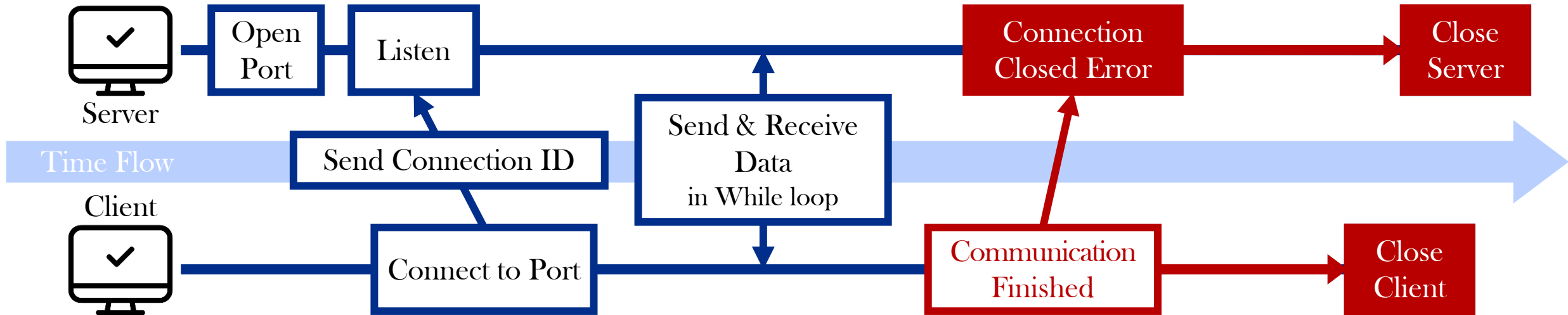


▲ For Overall Control



▲ For Sorting Board

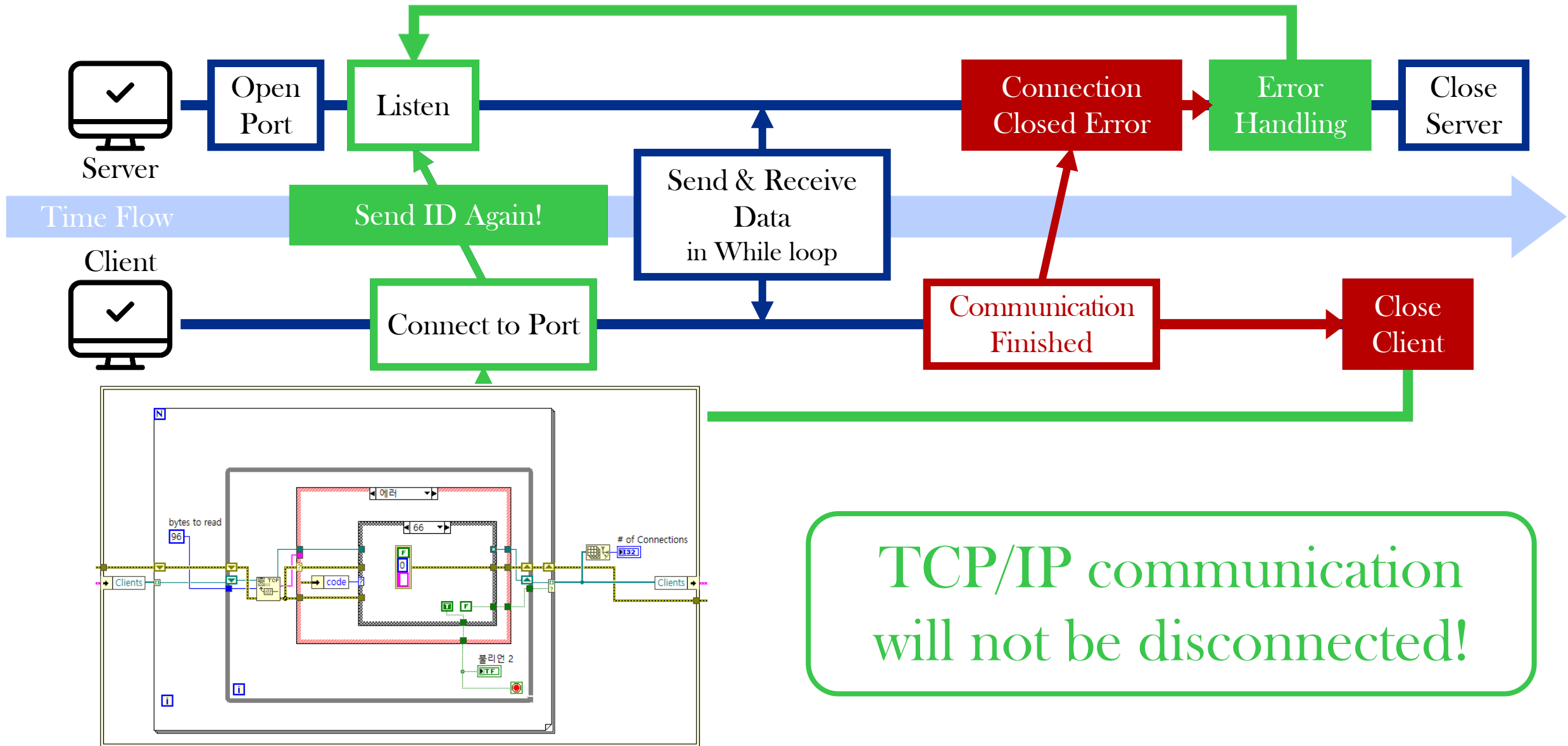
2 loops will reduce chance of sorting error!





Motor Control

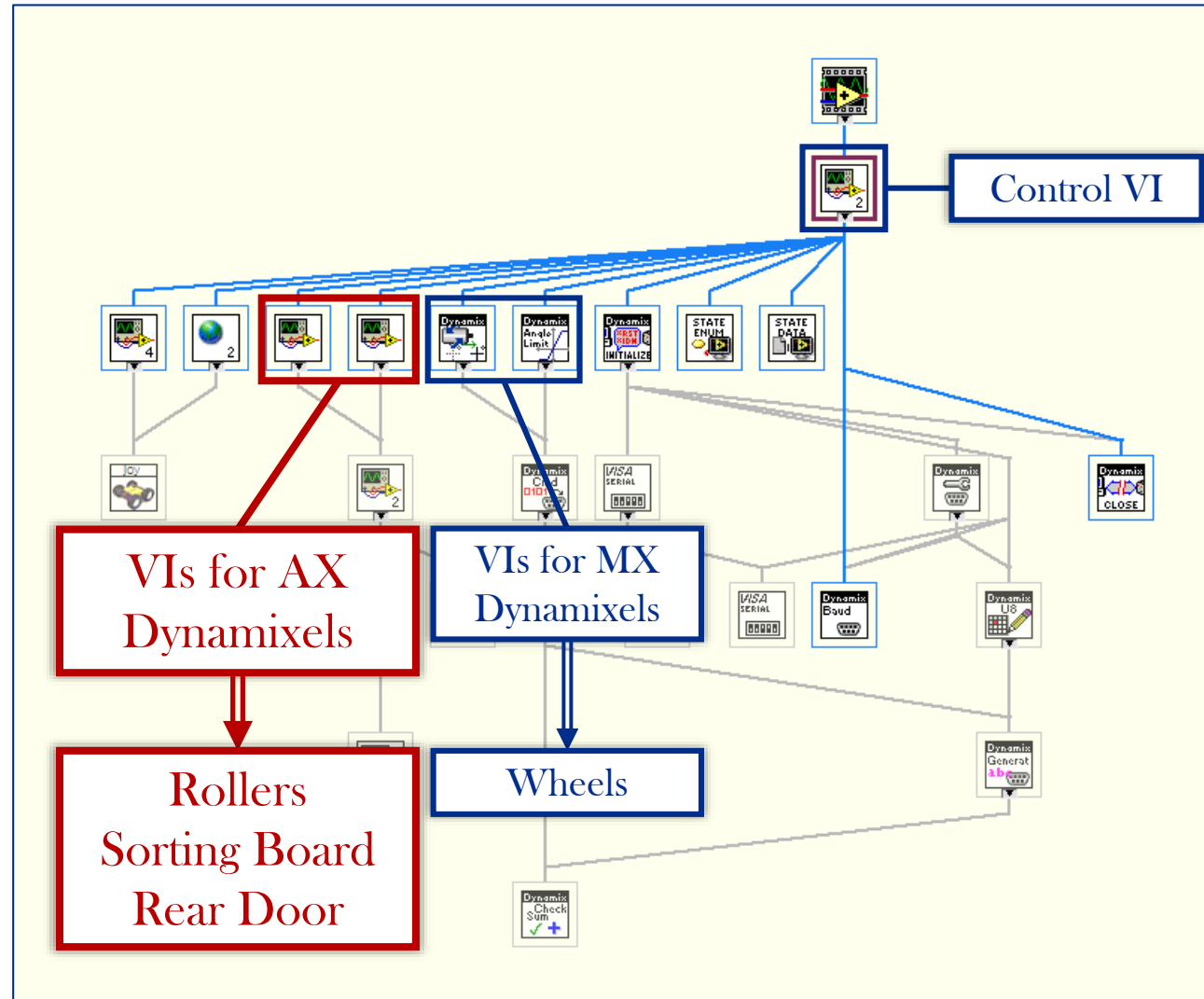
TCP/IP Communication



TCP/IP communication
will not be disconnected!



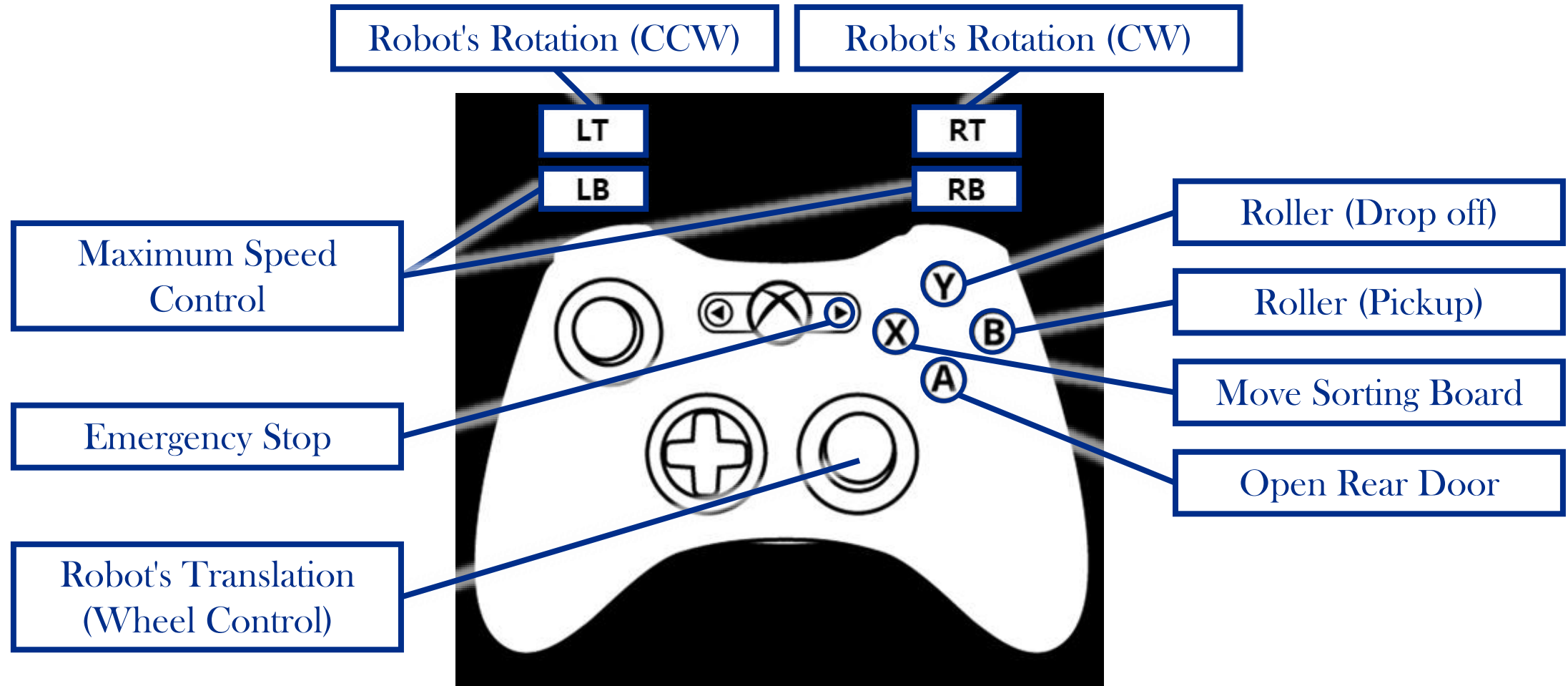
Hierarchy of Control VI





Motor Control

Xbox Controller





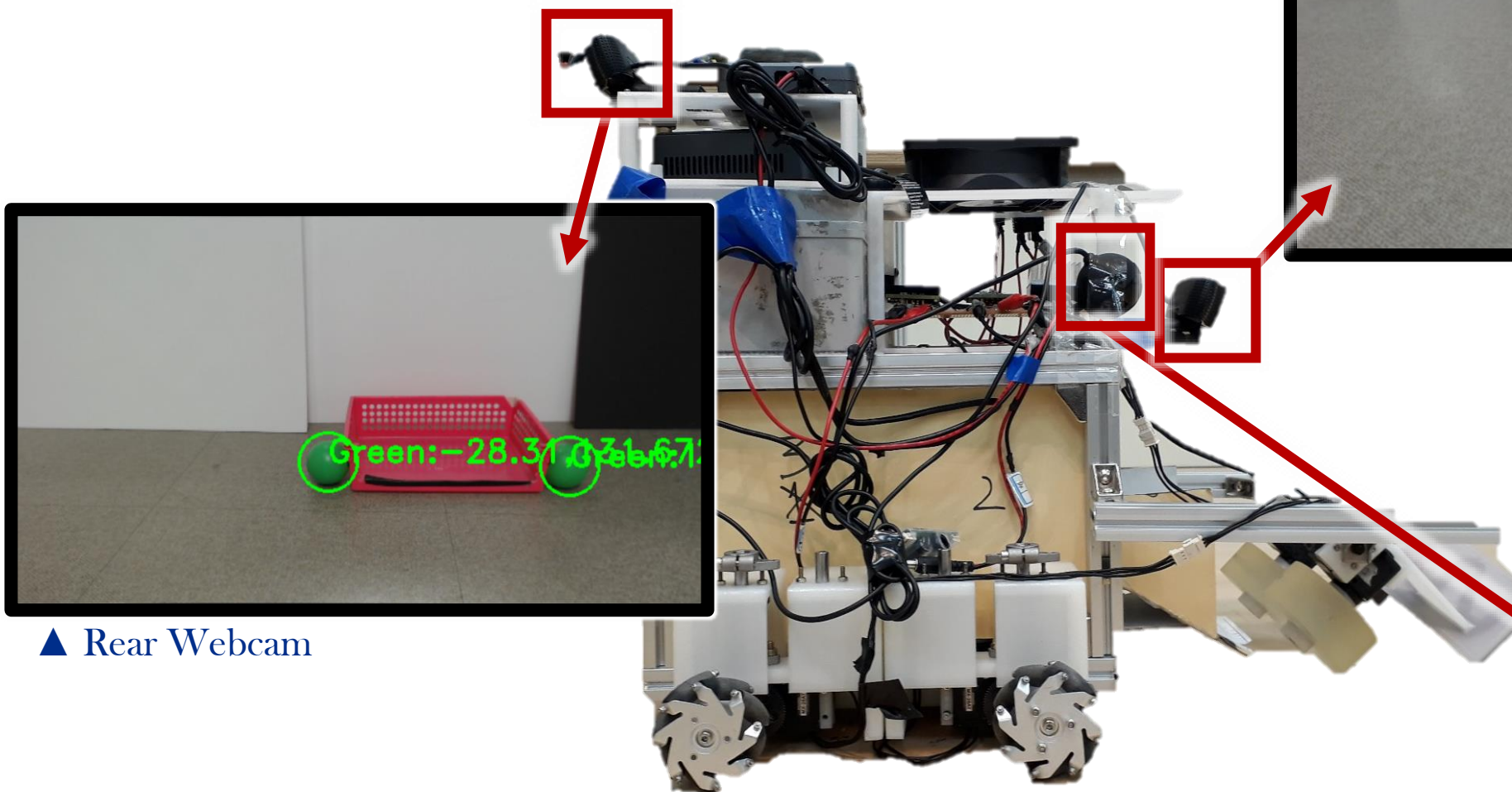
Vision Processing



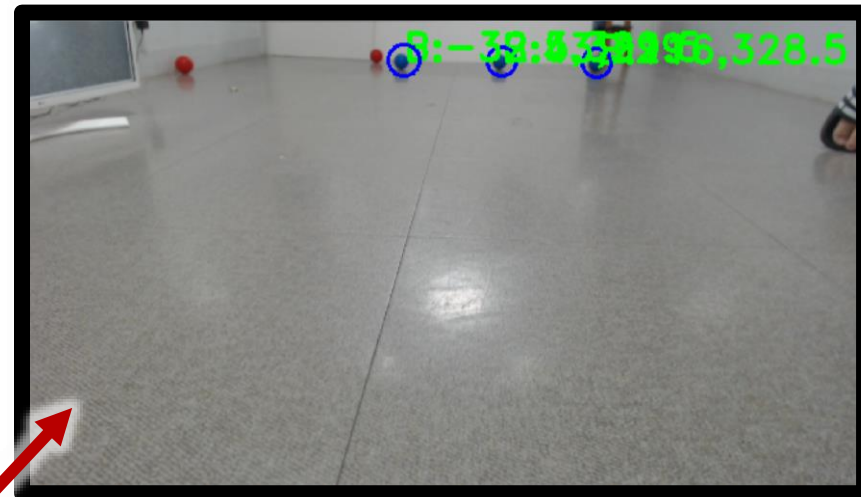


Vision Processing

More Webcams



▲ Rear Webcam



Front Webcam ▲

Sorting Webcam ▼





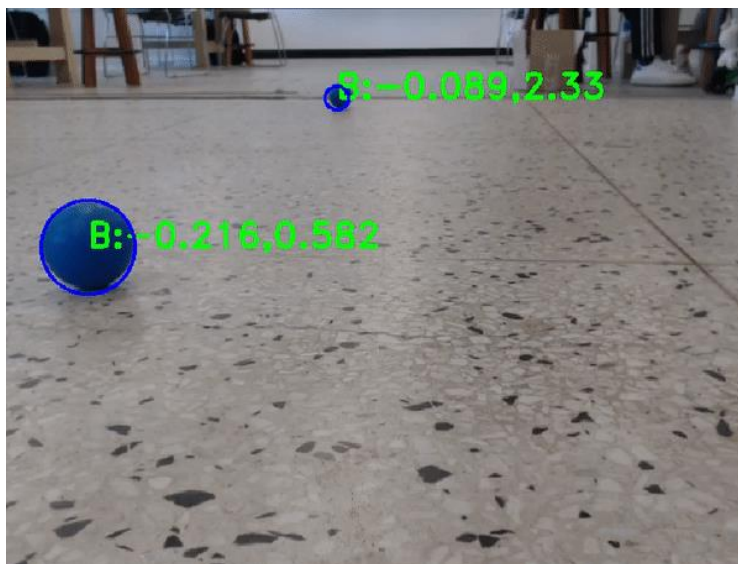
Goal To determine near ball, we have to compare appropriate value!

Calculated Distance

Calculate distance using
radius of the ball

Values are discrete,
farther distance leads to
more discreteness & error
(1.792m ~ 2.589m)

Unstable, Large
Error



Pixel Value



Relatively more continuous
values,
error is not affected by
distance (± 1 pixel)

Doesn't requires additional
algorithm

Stable,
Small Error

Using pixel value is better to compare distance!



Goal Our path algorithm requires to detect all balls at the start of the mission!

For wider angle, aspect ratio of 16:9 is better than 4:3

640×480



Too Narrow Angle

640×360



Wide Angle

Wider angle can detect all balls safely!



Goal Find appropriate resolution of aspect ratio 16:9.

1920×1080



Too Slow

640×360



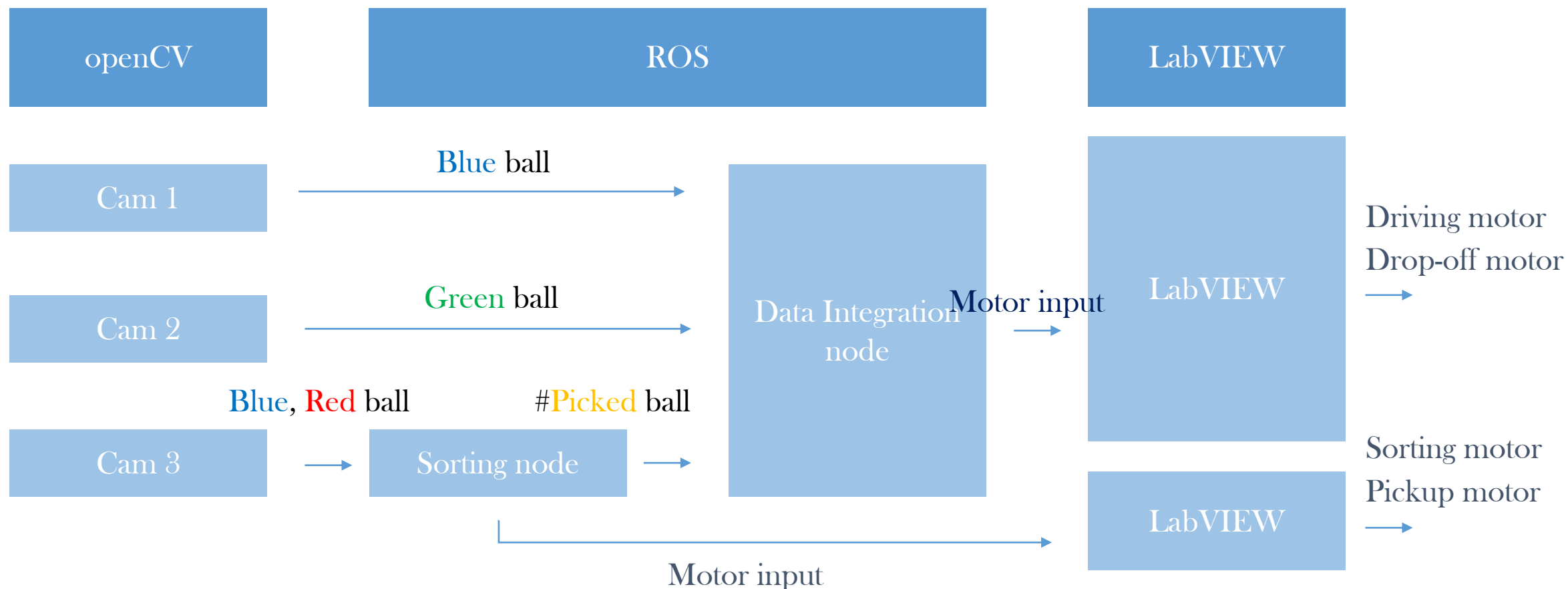
Fast Calculation

Lower resolution's calculation speed is faster!



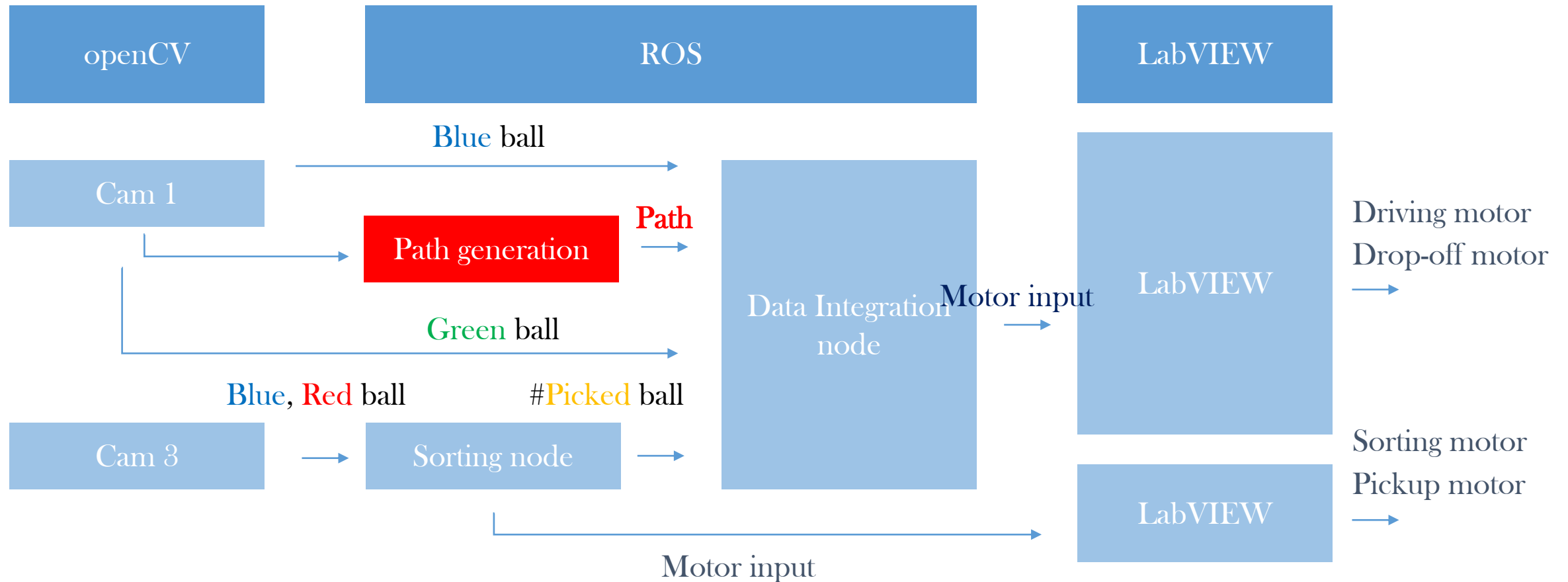
System Integration







Version 1





Version 2

Goal

Catch the target ball certainly(exactly)

Remove the path generation process

Determine 3 path cases based on pixel y-coordinates of the ball

Update the target ball with the smallest y-coordinate of pixels per loop



Version 3

Goal

Safe and predictable behavior algorithm

Reduce rotation in opposite direction

Pickup the ball with a large x-coordinate order

Update the target ball with a large x-coordinate of pixels per loop



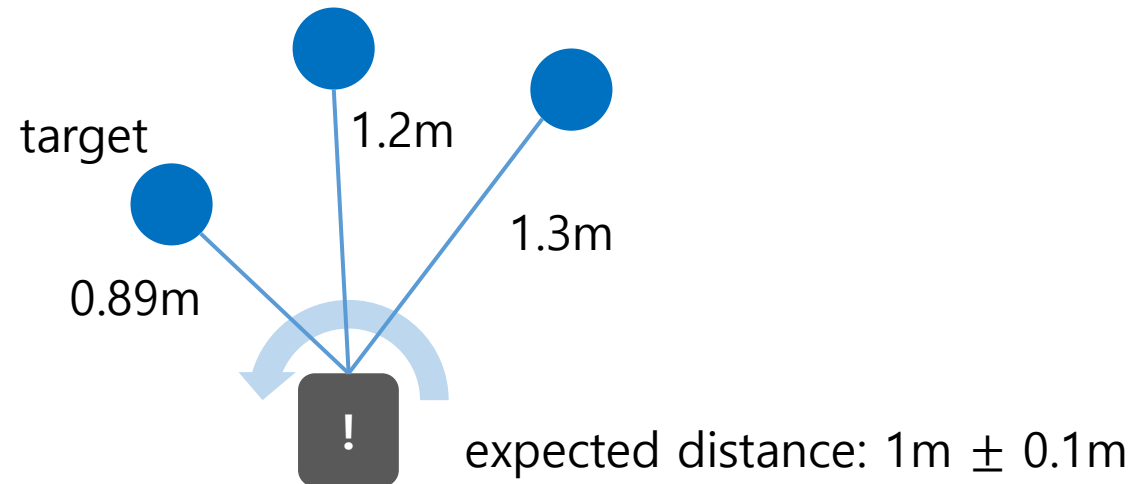
Version 1

Problem

Can't detect the target ball

Reason

Comparing incorrect absolute distance

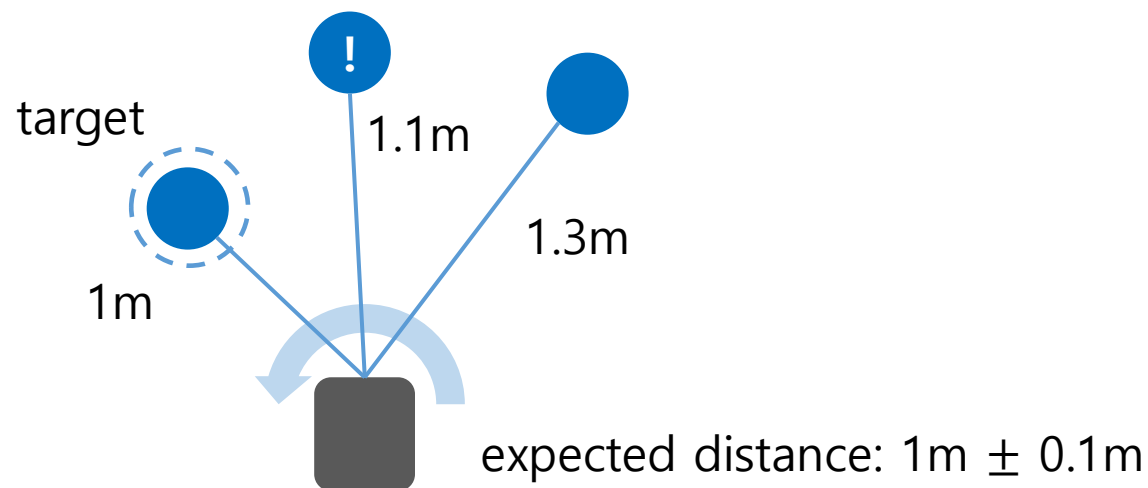


Remove the path generation process



Version 1

Problem	Detect the wrong target ball
Reason	Comparing incorrect absolute distance

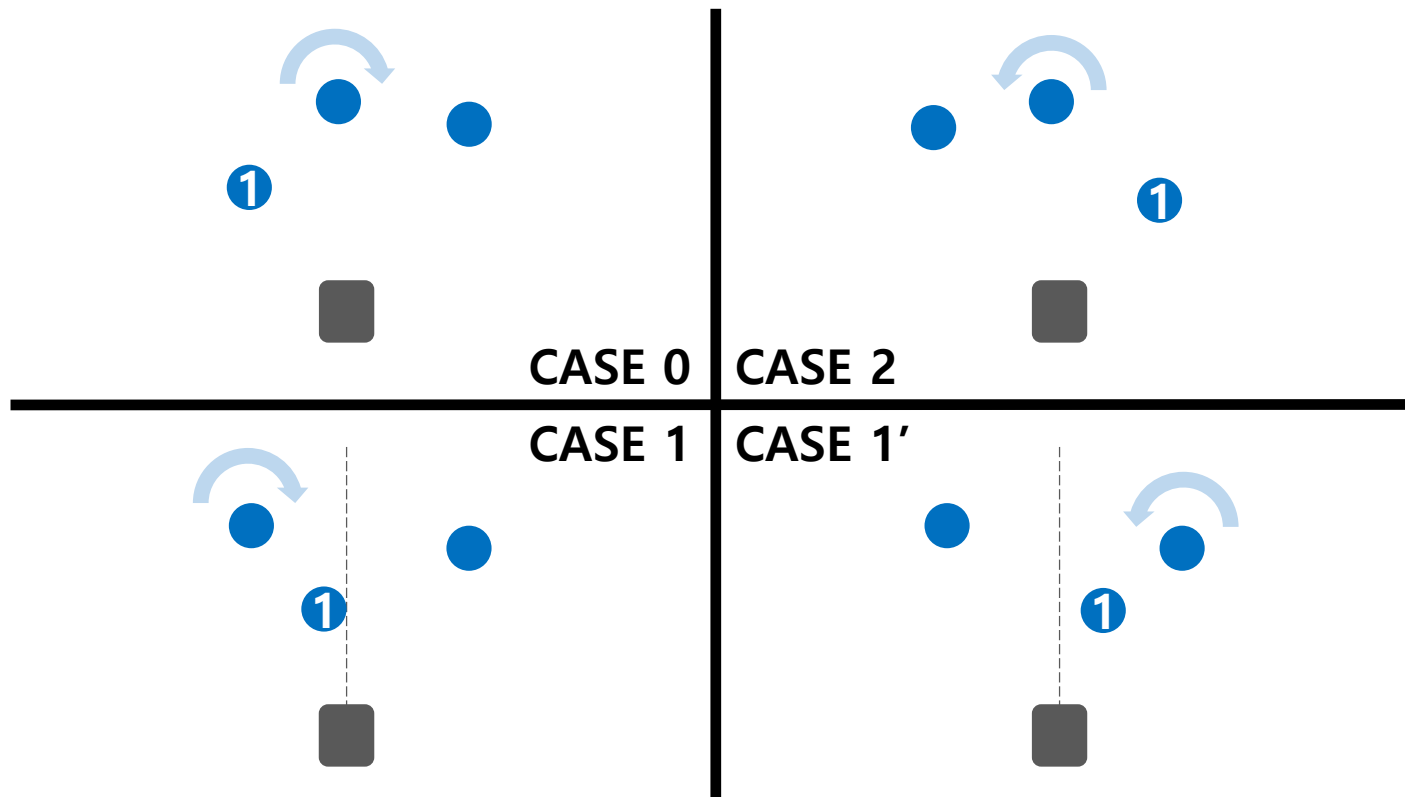


Remove the path generation process



Version 2

Determine 3 path cases based on pixel y-coordinates of the ball



Target ball
=> smallest pixel y-coordinate

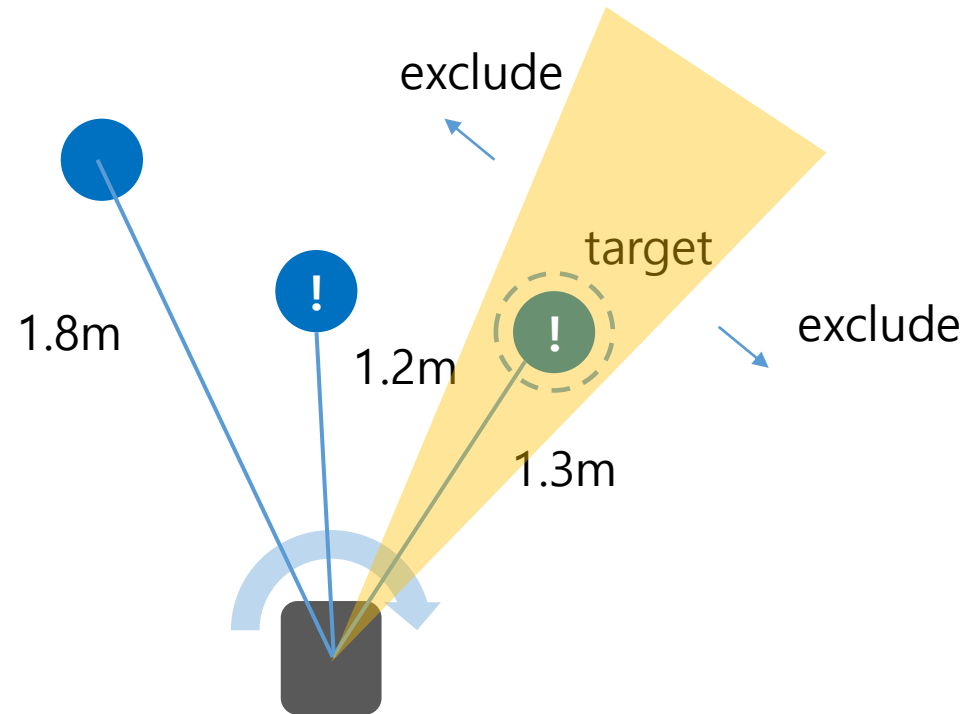
Certainly detect the target



Version 2

Problem 1

The target changes after rotation



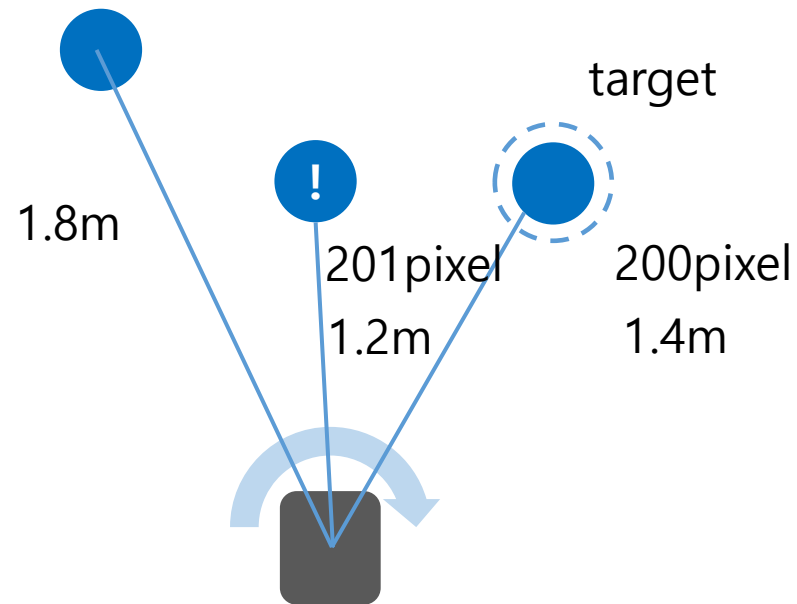
Exclude other balls using x-coordinate when straight



Version 2

Problem 2

Target is constantly changing when the y coordinates of the balls are similar



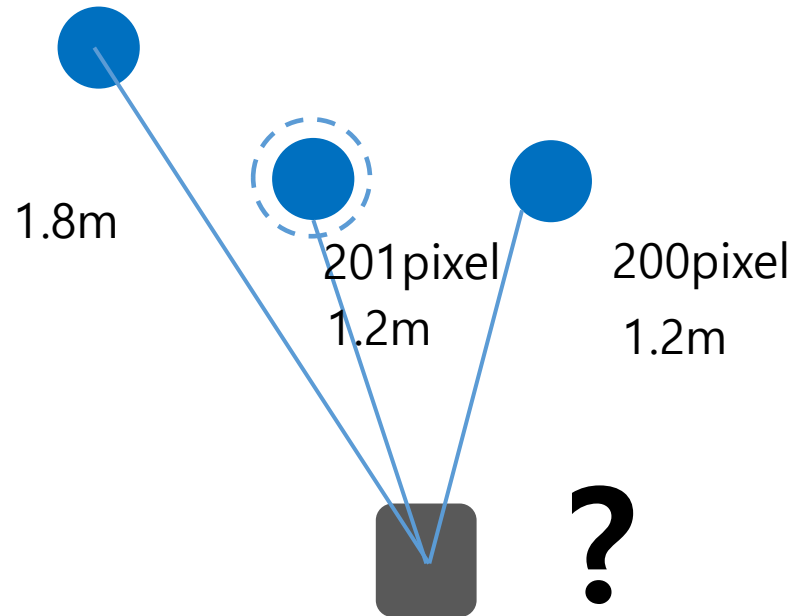
Determine target by distance if y coordinates are similar



Version 2

Problem 3

Unpredictable behavior
Rotation in opposite direction

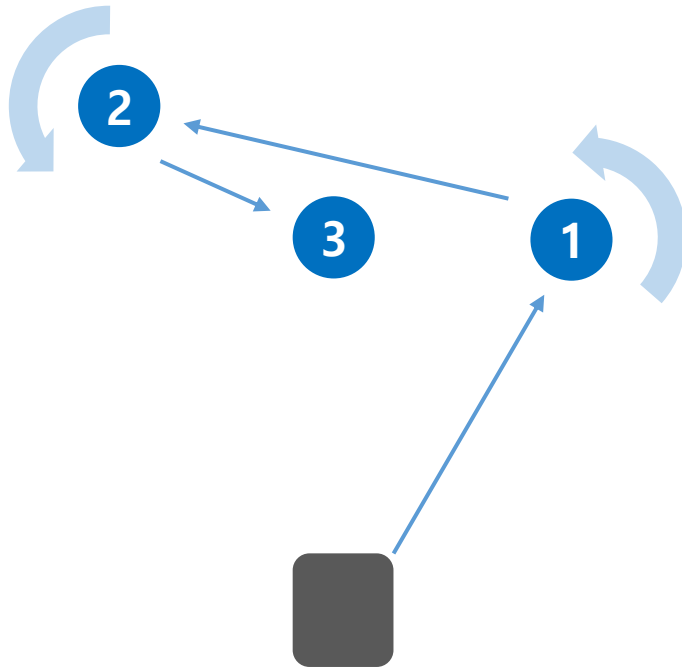


Safe and simple algorithm



Version 3

Pickup the ball with a large x-coordinate order



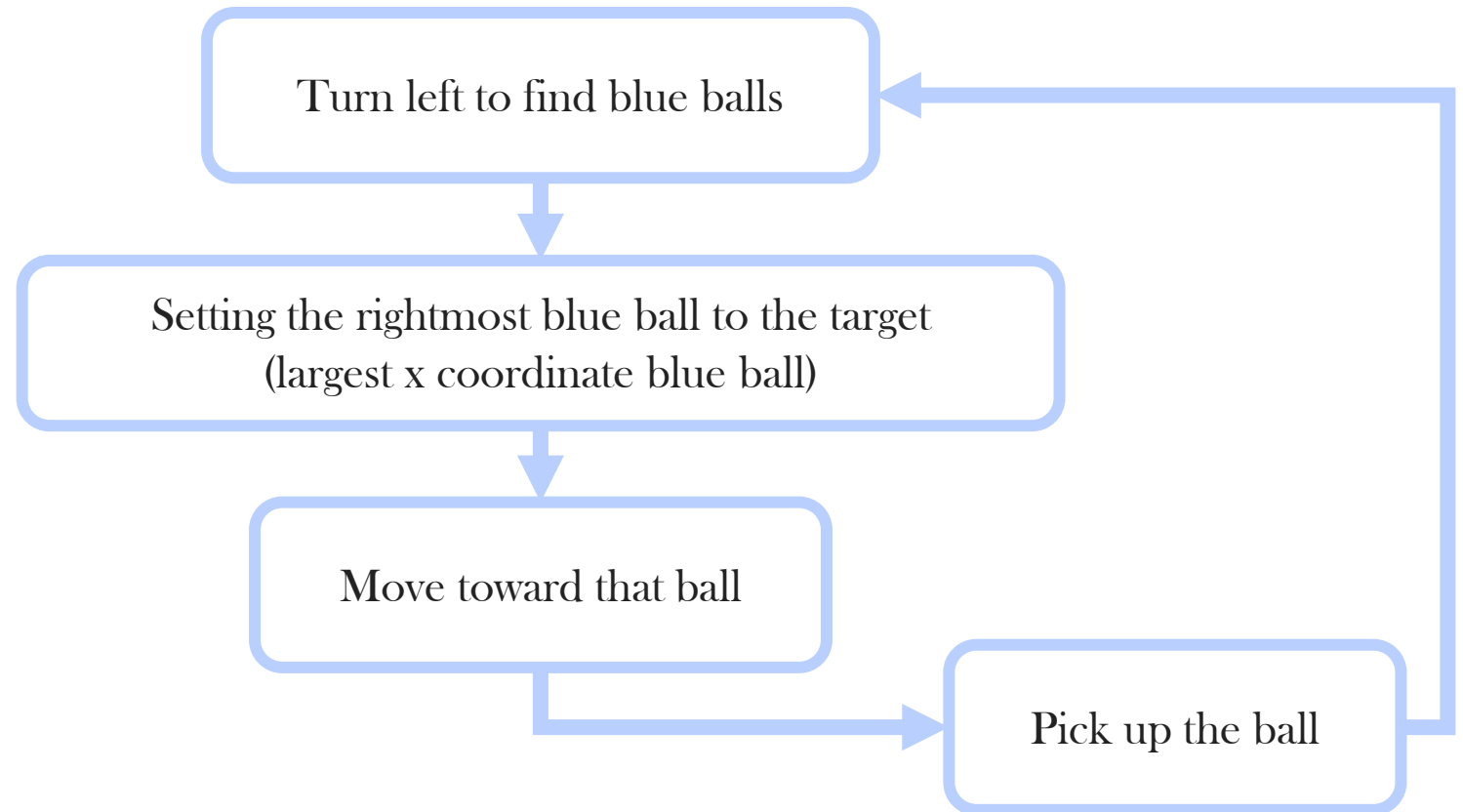
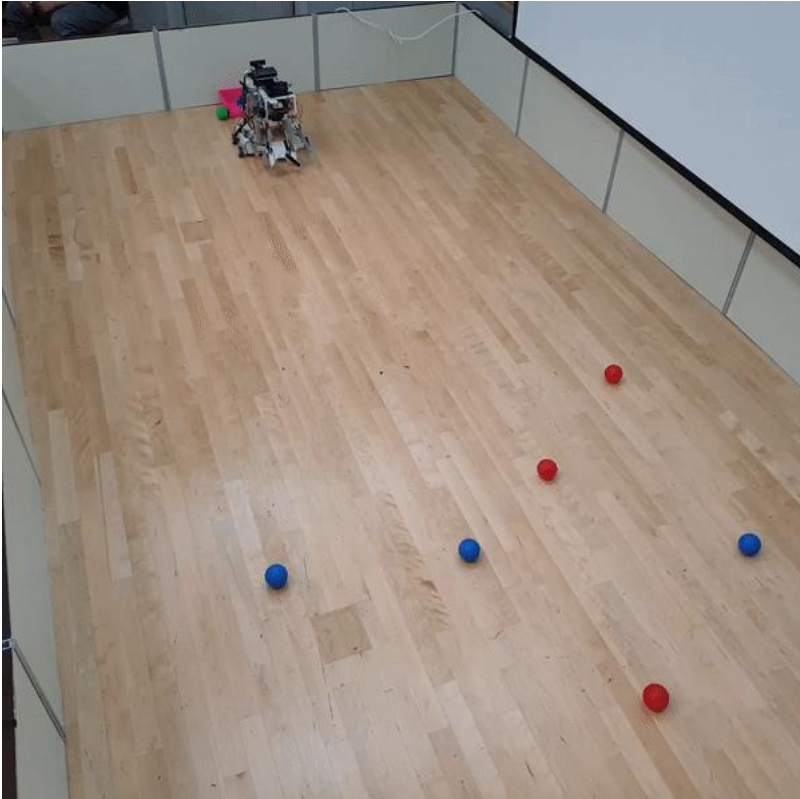
Target ball
=> largest pixel x-coordinate

No rotation in opposite direction

Safe and predictable behavior



Process 1. Search blue ball & Pick up balls

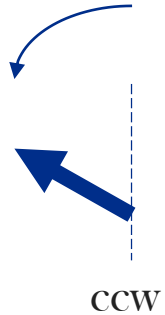


Repeat this process until number of picked blue ball = 3

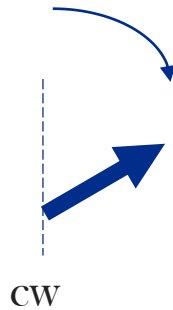


Process 1. Search blue ball & Pick up balls

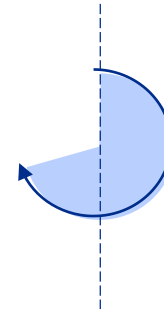
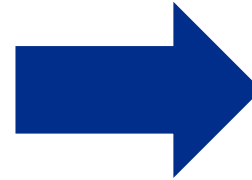
Record degree and direction of rotation during process



Add rotation angle



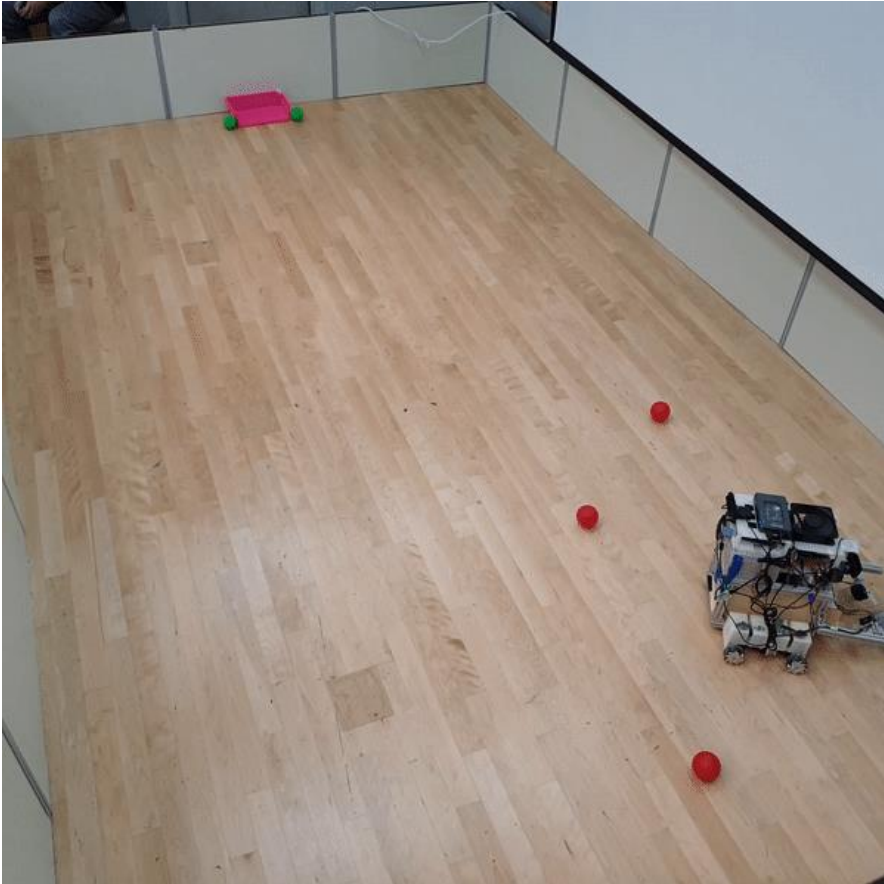
Subtract rotation angle



Record the total rotated angle

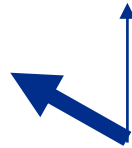


Process 2. Return to basket



- ✓ Determine direction to turn
at the position where the 3rd blue ball is picked up.

Total angle < 0



Turn cw

Total angle > 0

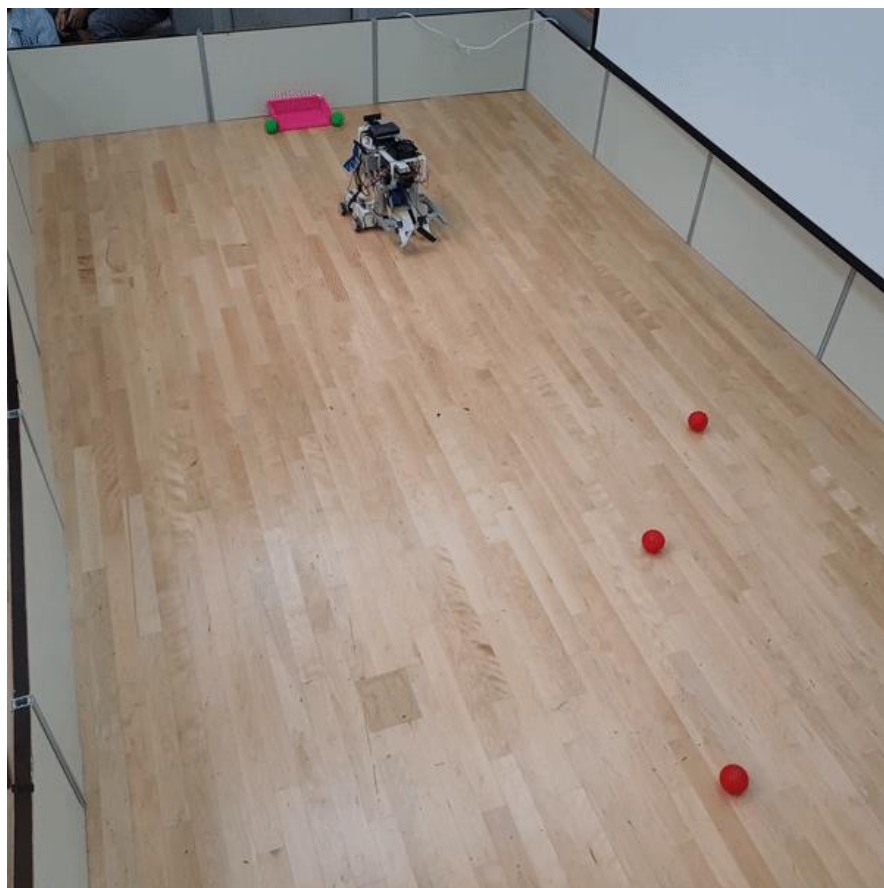


Turn ccw

- ✓ Go back to basket using backside camera

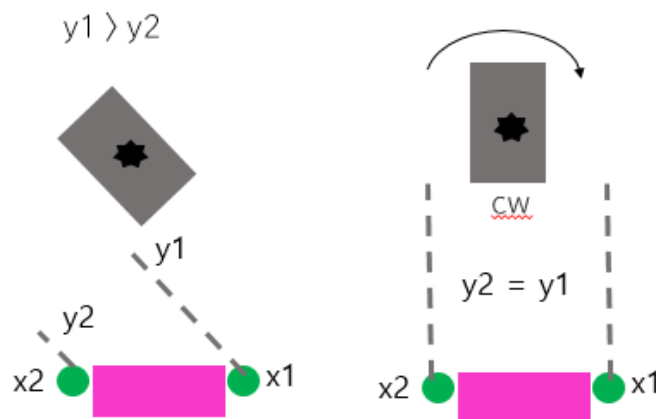


Process 3. Aligned parallel to the basket & Drop-off

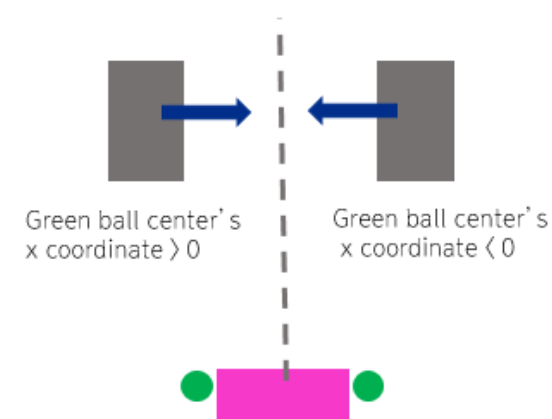


✓ Align robot to basket in parallel

Y align process



X align process





Efforts for Accuracy and Safety

- ✓ Go straight to the center of the ball
- ✓ Using the green balls to return the center of the basket
- ✓ Process consist of several stages
- ✓ Predictable algorithm

Efforts to Reduce Time

- ✓ Make robot rotate less
- ✓ Return to the basket with a backward run
- ✓ Pick up all balls without avoiding motion

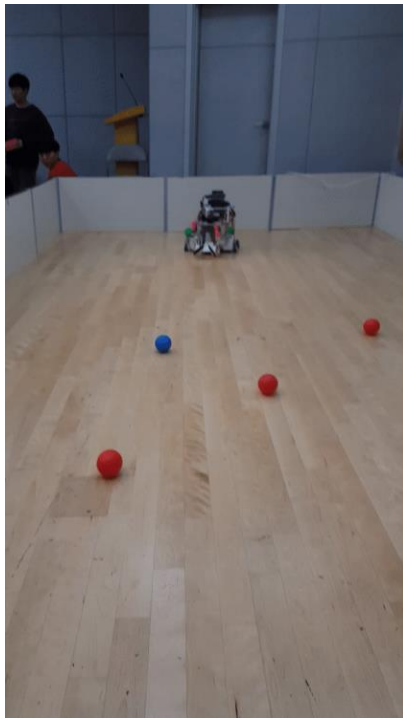
Efforts to Cope with Emergency Situation

- ✓ check blue balls before releasing motion



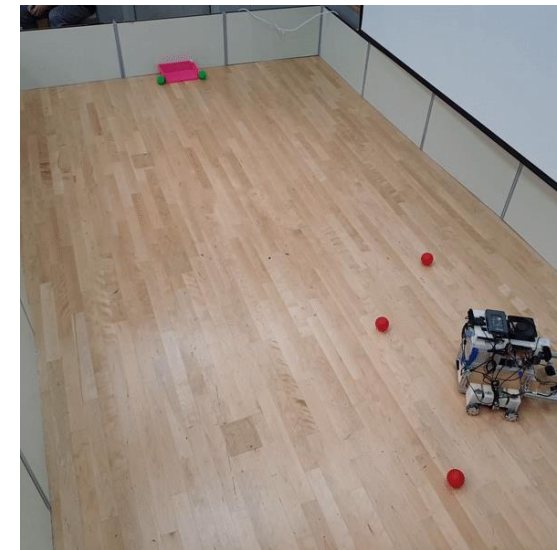
Efforts for Accuracy and safety

Go straight to the center of the ball



Improve pick-up ability

Using the green balls on both sides,
return the center of the basket

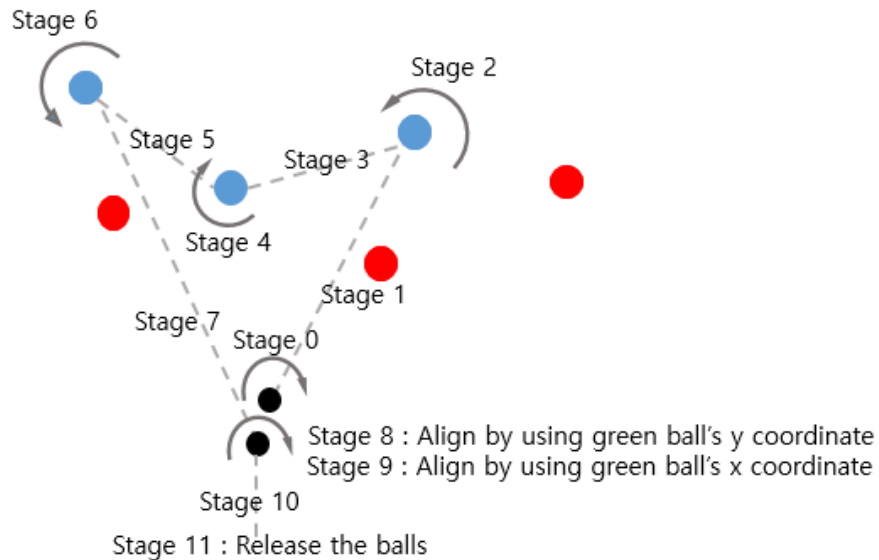


Improve release accuracy :
prevent the ball from escaping of the basket
when releasing the ball.



Efforts for Accuracy and safety

Process divided into stages



Predictable algorithm

distance is not correct

y coordinate can be changed while rotating

use x coordinate relatively correct

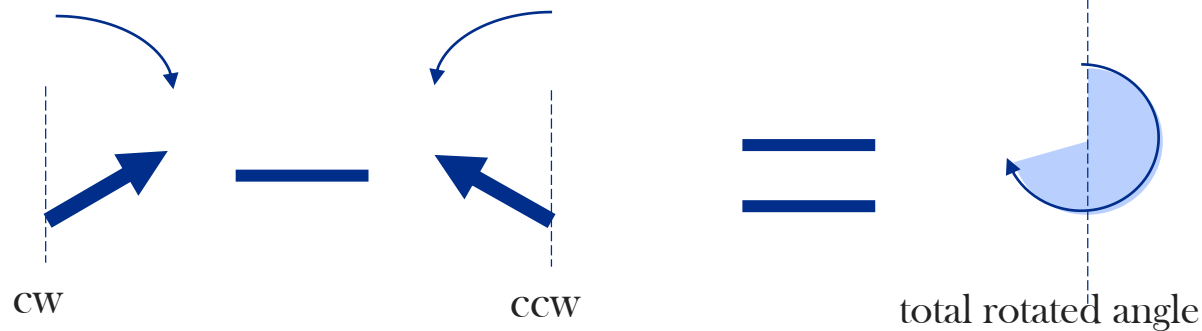
- ✓ Easily move desired process using stage number
- ✓ Easily find the source of error

- ✓ Robot behavior can be accurately predicted



Efforts to reduce time

Record degree and direction of rotation during process

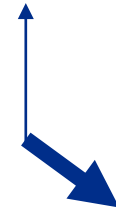


Total angle < 0



Turn cw

Total angle > 0



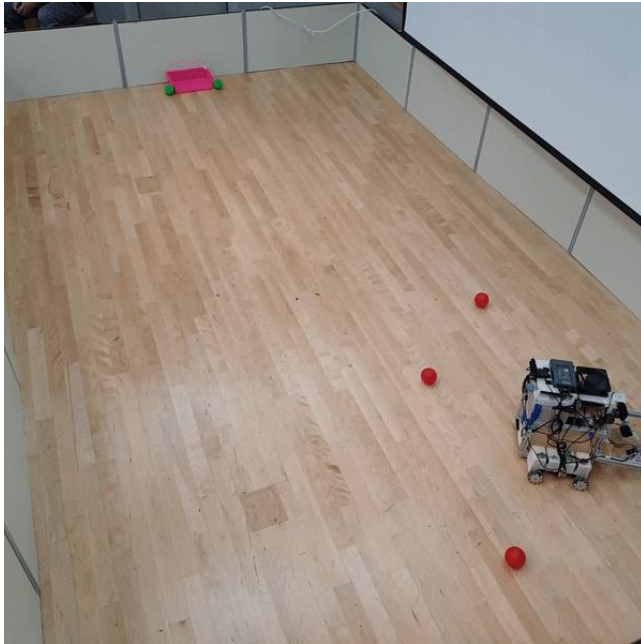
Turn ccw

- ✓ Determine the time efficient direction of rotation at the 3th blue ball position
- ✓ Save time by reducing unnecessary rotations



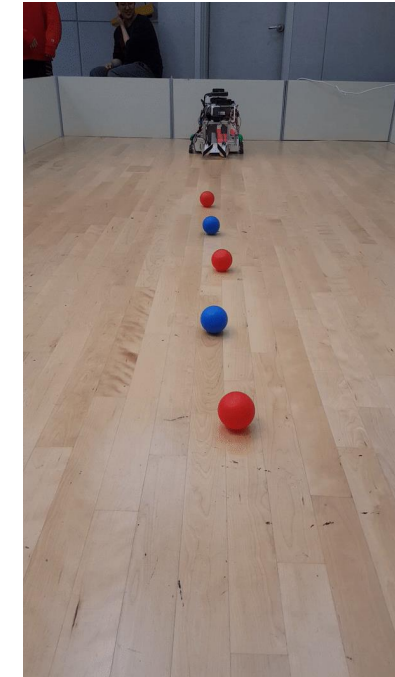
Efforts to reduce time

Use backward movement
when coming back to the basket



✓ Save time by reducing unnecessary rotations

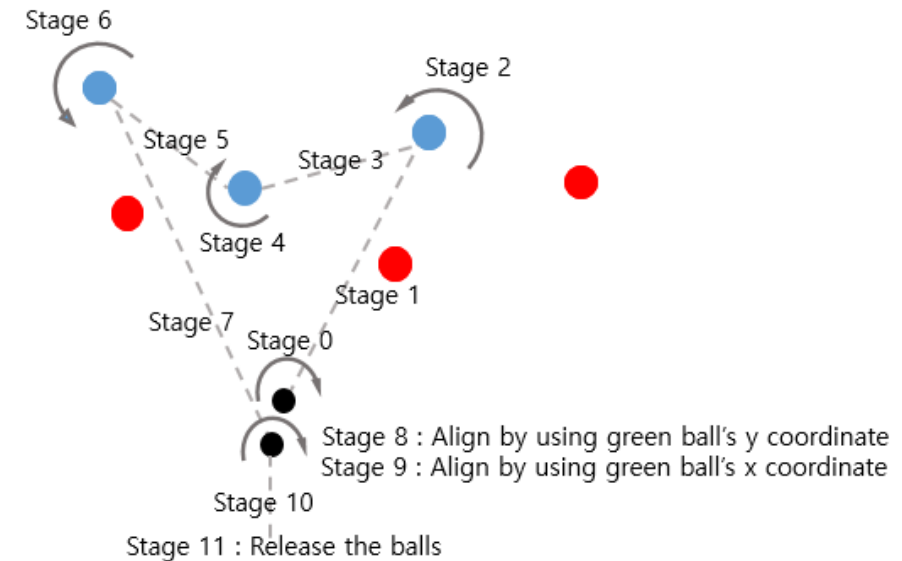
Pick up all balls
without avoiding motion



✓ Save time by reducing unnecessary movements



Efforts to cope with emergency situation



If robot do not pick up all the blue balls completely,
go back to stage 2 or 4 depending on the remaining blue balls
and pick them up again before drop-off



Sorting node

Goal 1

Sorting balls regarding their color



Find the nearest ball



Judge the color of the nearest ball



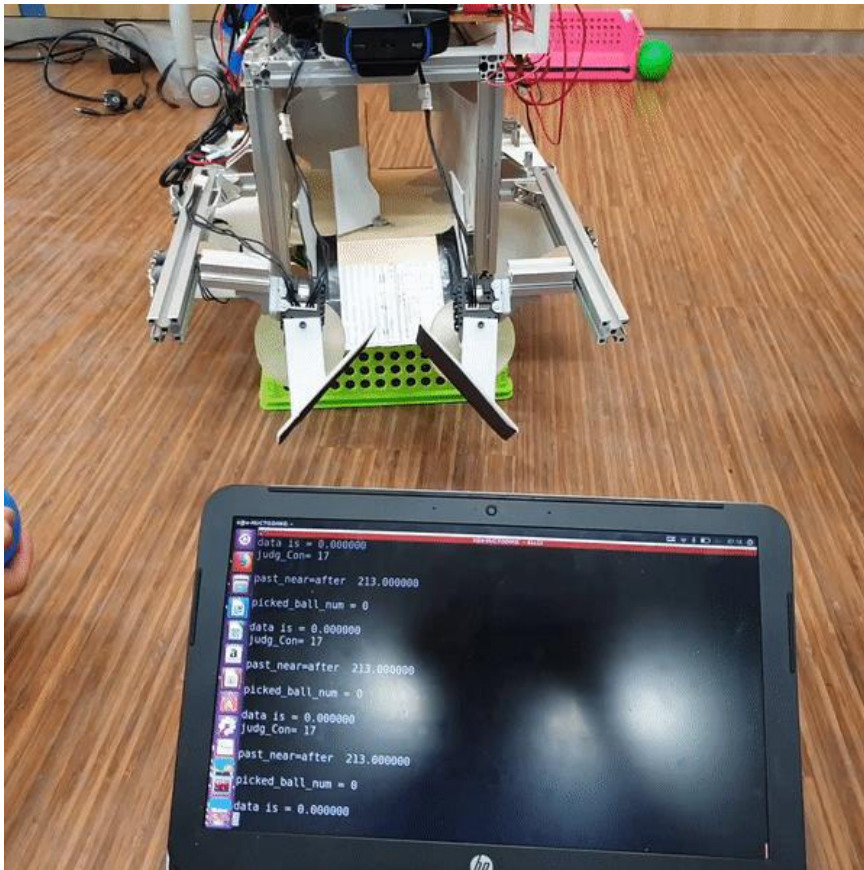
Move the sorting bar



Sorting node

Goal 2

Calculate picked red/blue ball number



Ball's y coordinate decrease

Ball's y coordinate
< OFFSET

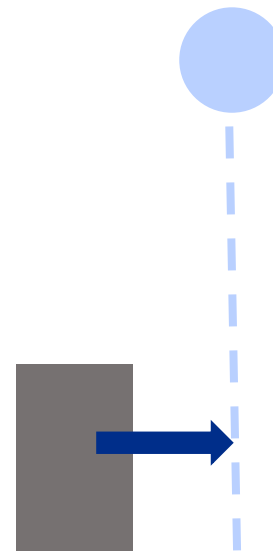
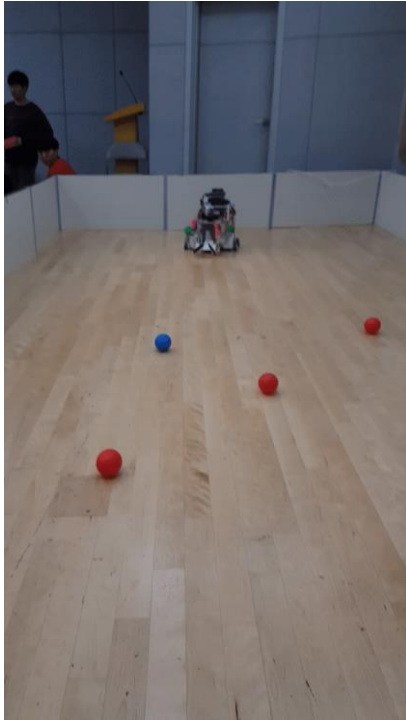
Judge the ball was picked up



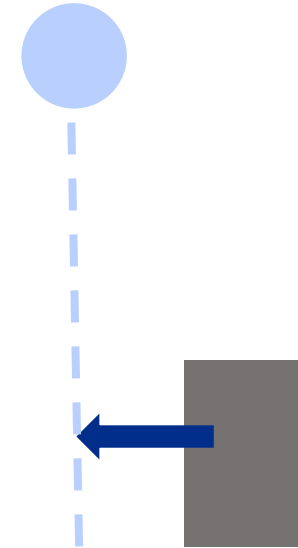
Data integration node

Goal 1

Go straight to the center of the blue ball



Target ball's x coordinate > 0



Target ball's x coordinate < 0

Set the ball with the largest x coordinate as the target

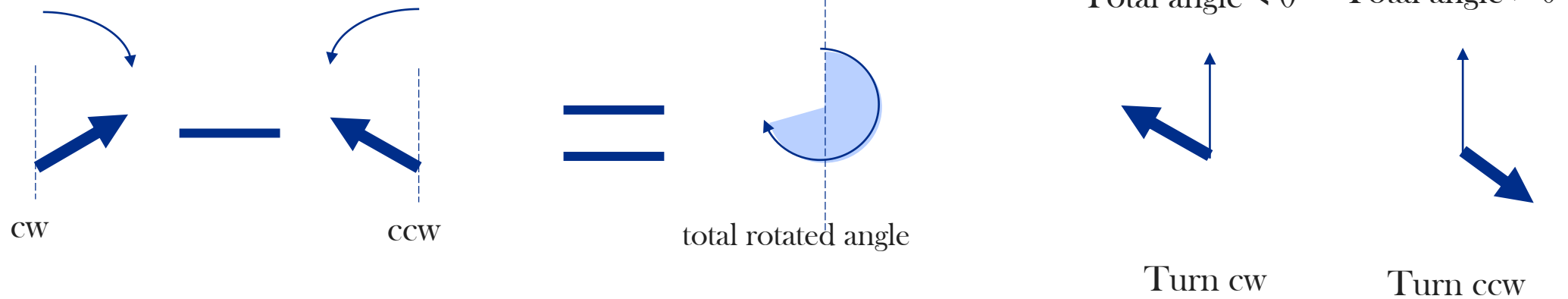
go straight toward center of the blue ball



Data integration node

Goal 2

Record the rotation & determine the rotation direction



Record degree and direction of rotation during whole process

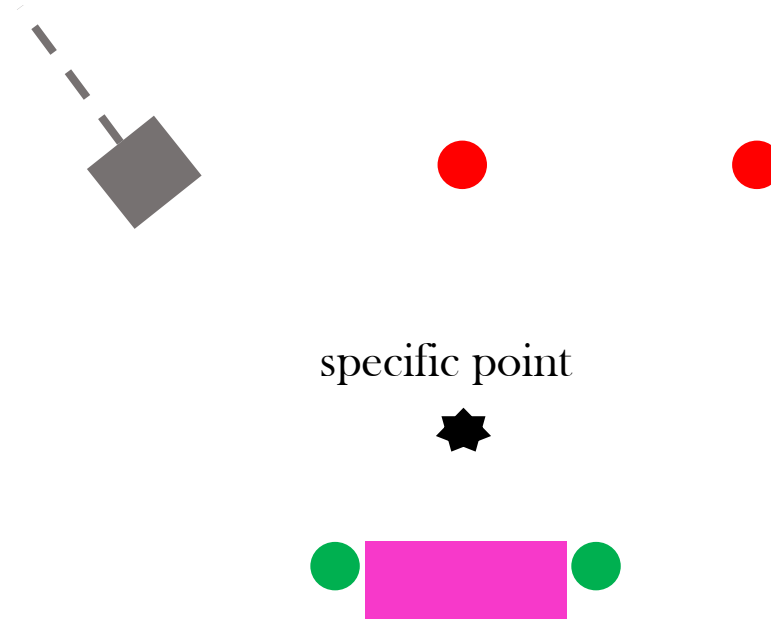
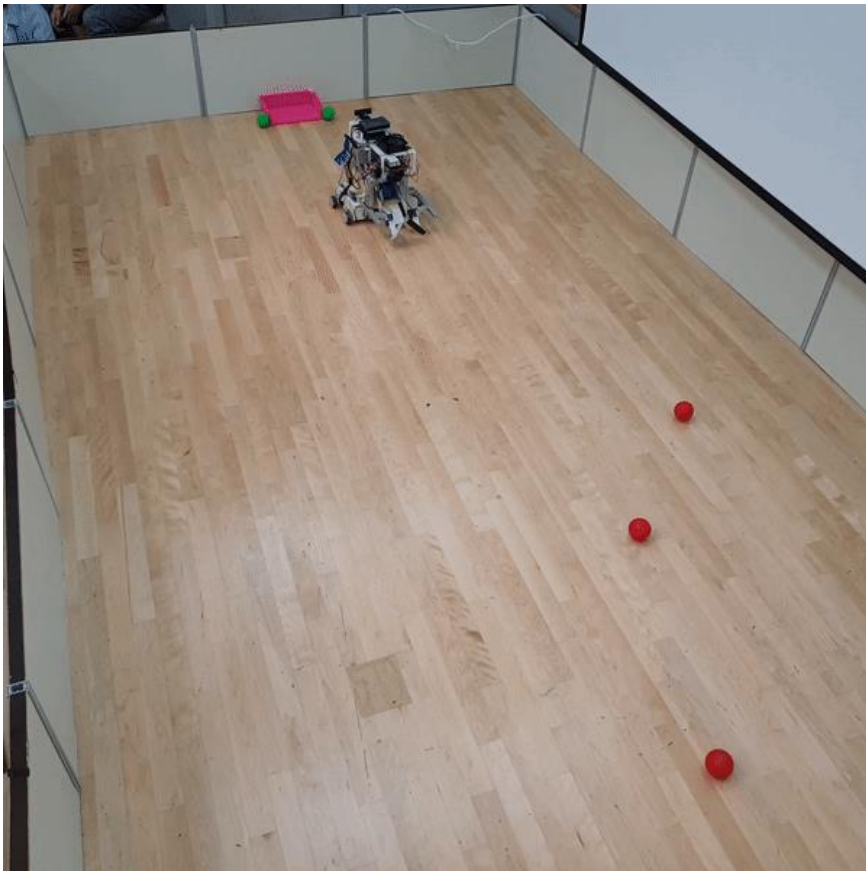
Determine the time efficient direction of rotation at the 3th blue ball position



Data integration node

Goal 3

Align the robot parallel to the basket



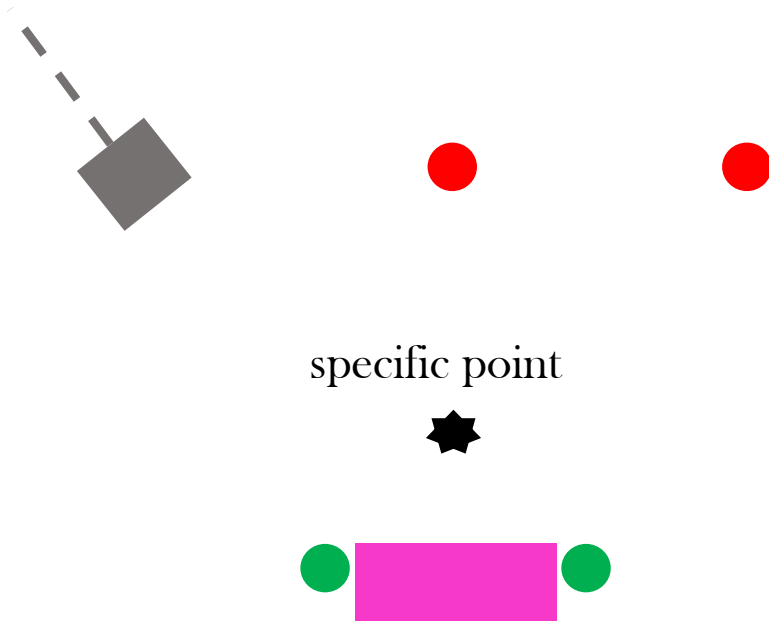


Data integration node

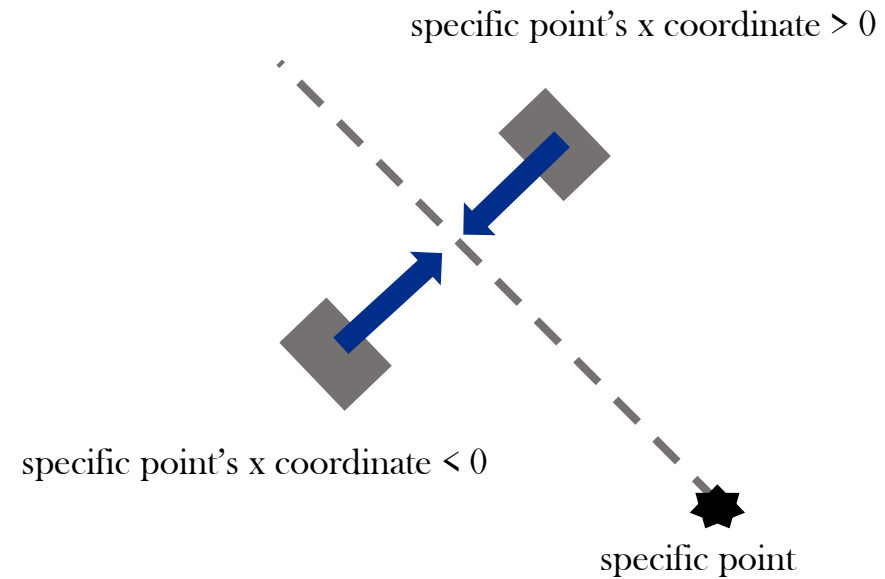
Goal 3

Align the robot parallel to the basket

1. Compute the coordinates of specific point using the coordinates of two green balls



2. X align process



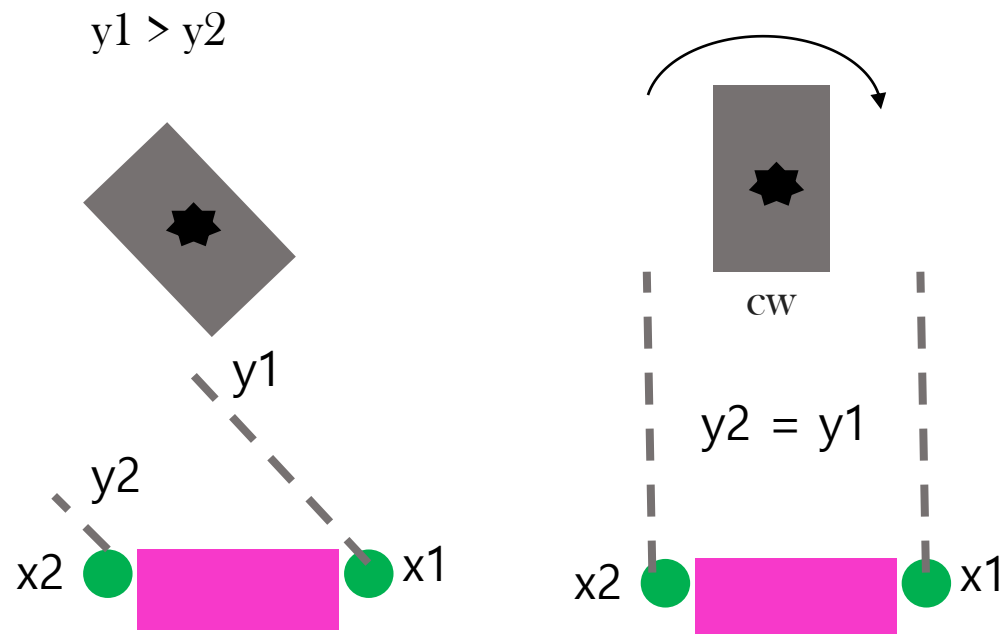


Data integration node

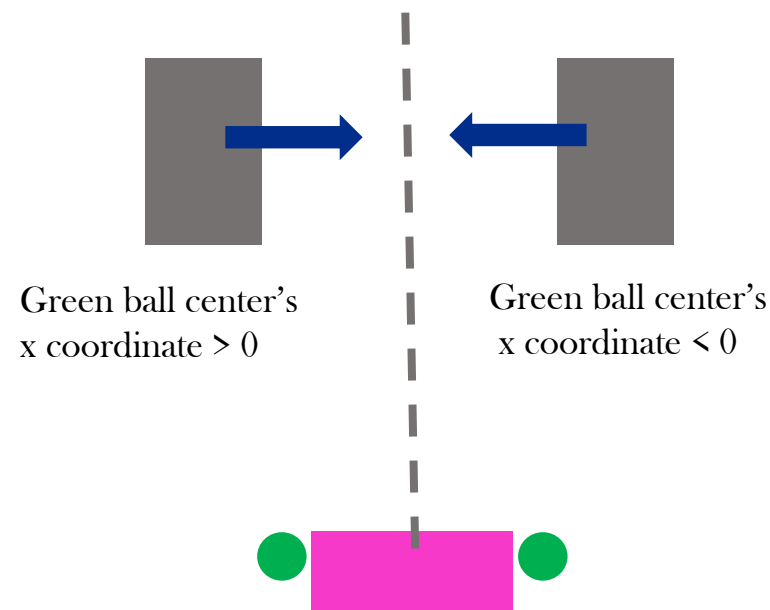
Goal 3

Align the robot parallel to the basket

3. Y align process



4. X align process

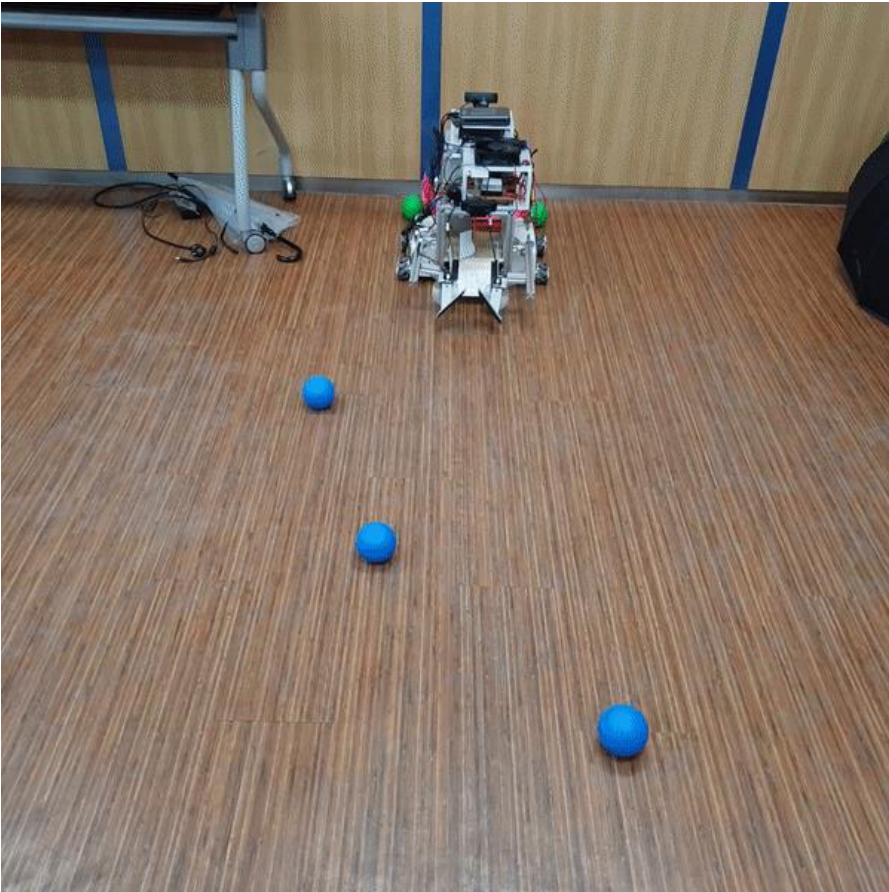




Data integration node

Goal 4

Go back to specific stage and pick up blue balls again



Webcam sees blue ball



do not pick up all the blue ball



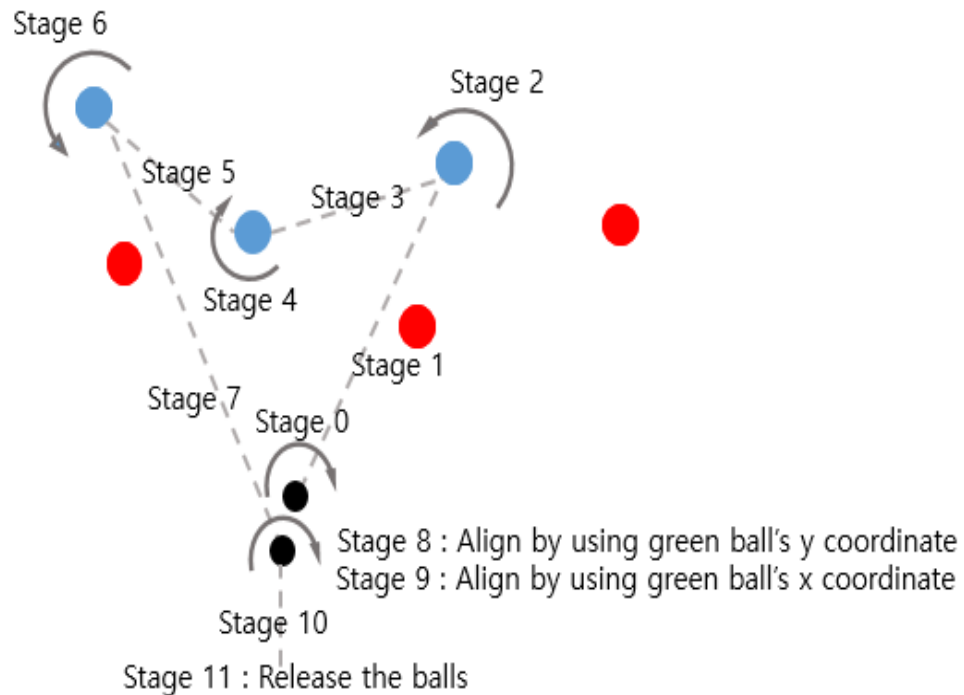
Go back to previous stage



Data integration node

Goal 4

Go back to specific stage and pick up blue balls again



Two blue balls remain

Go back to stage 2

One blue balls remain

Go back to stage 4

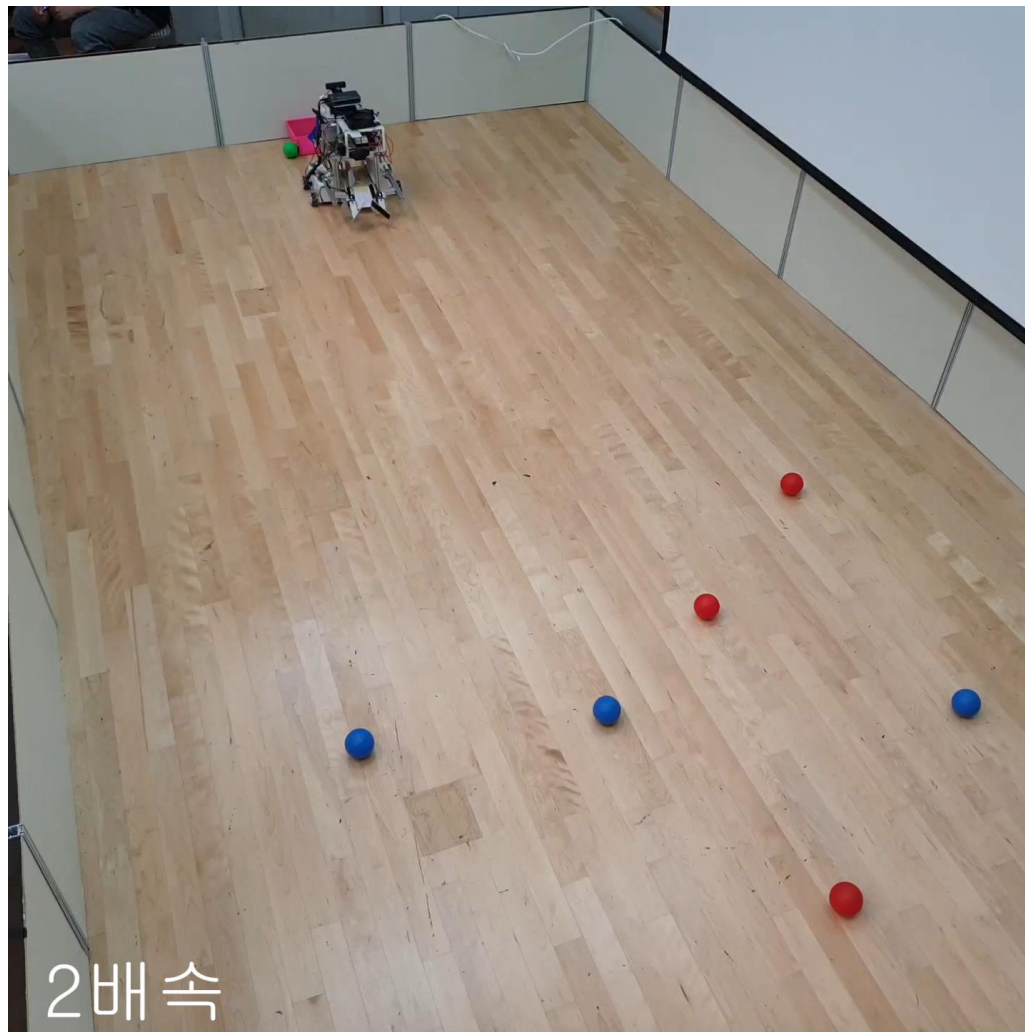


Demo Video





Demo Video



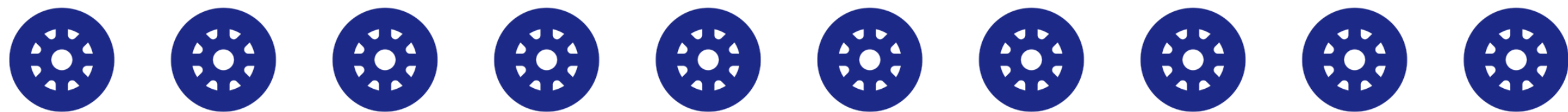
2배속



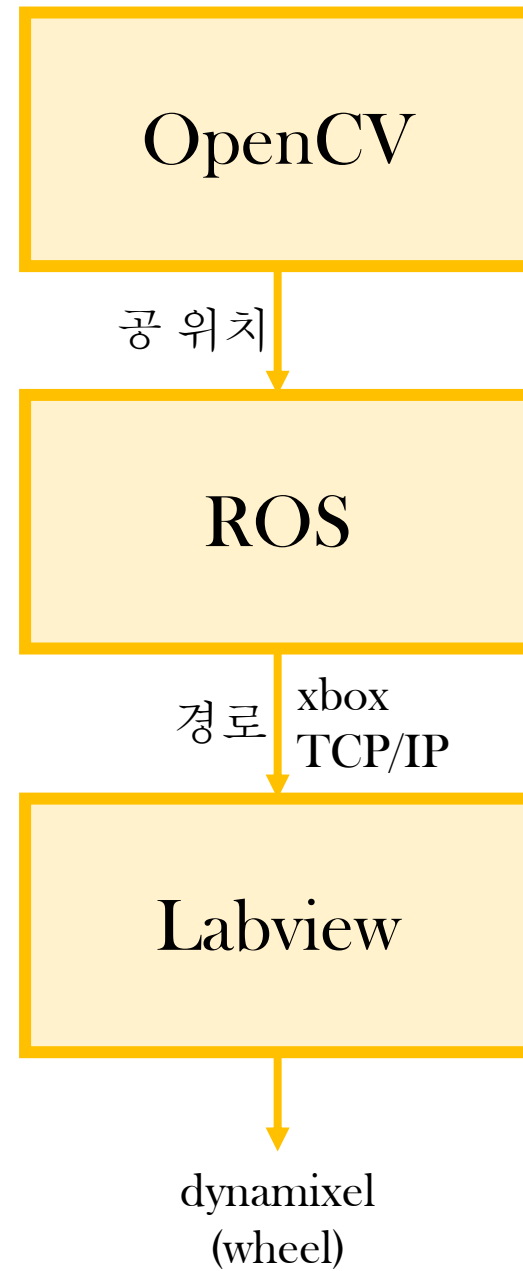
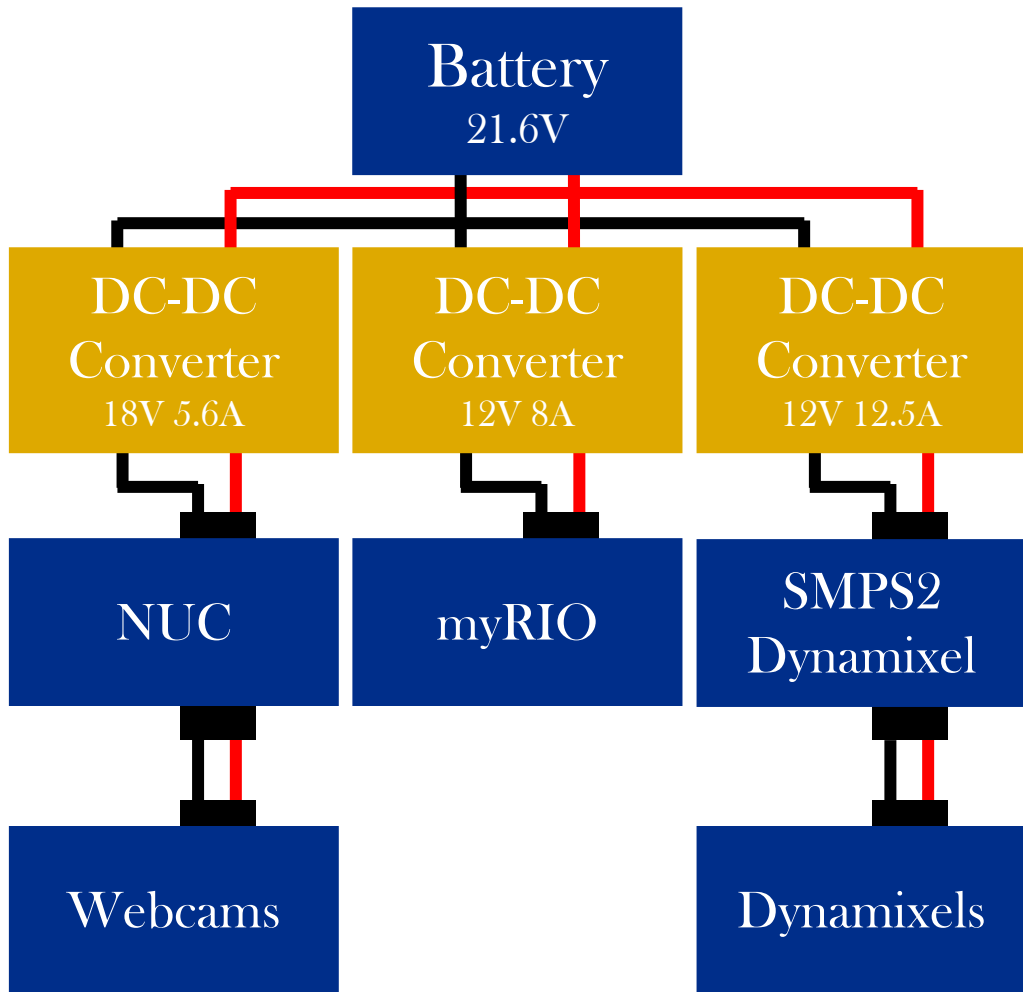
Demo Video

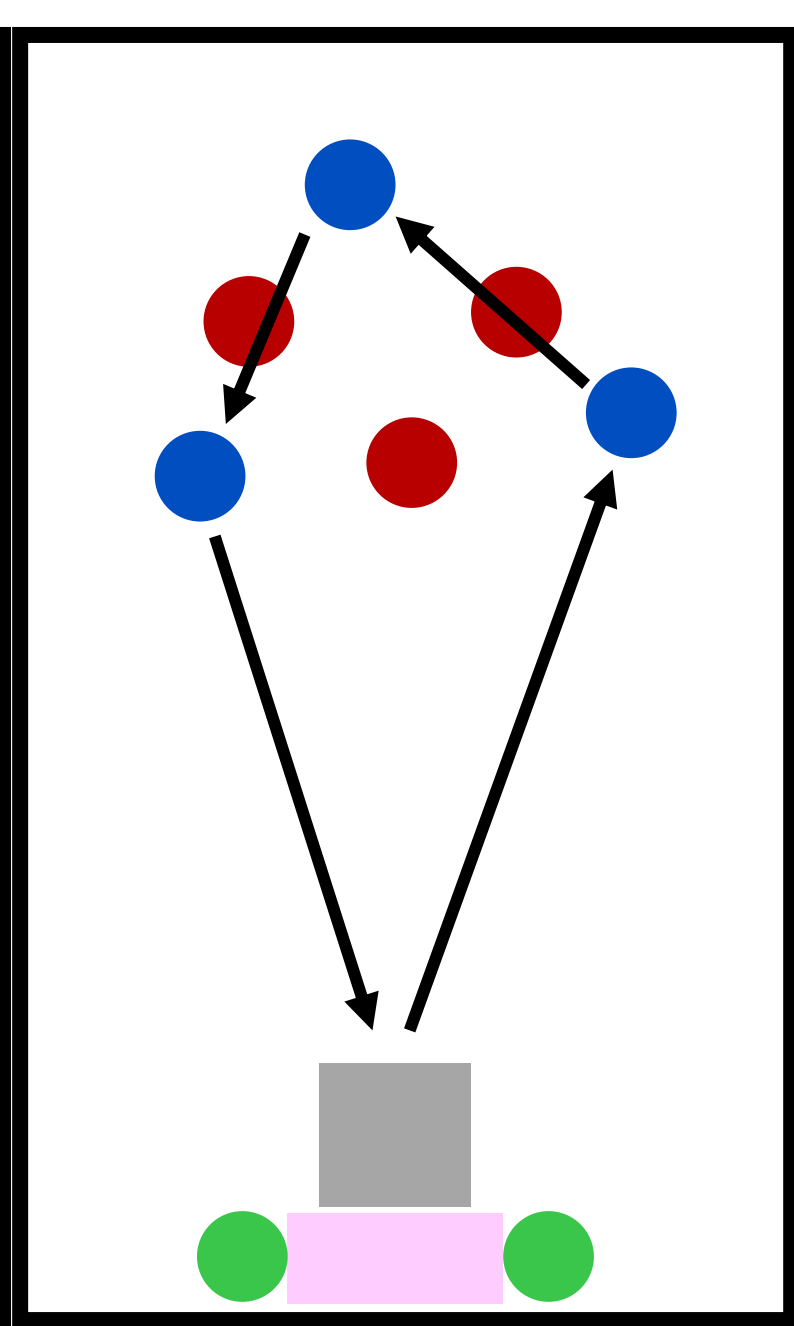
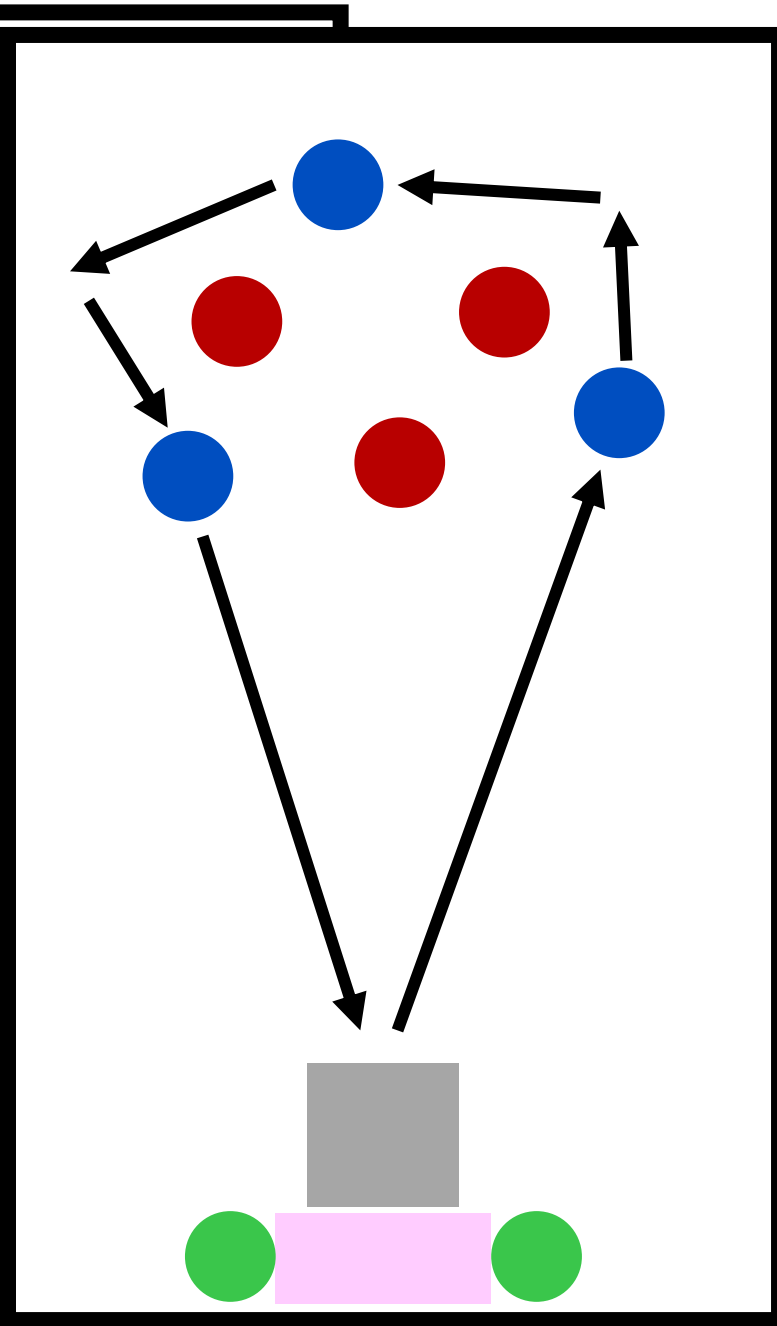
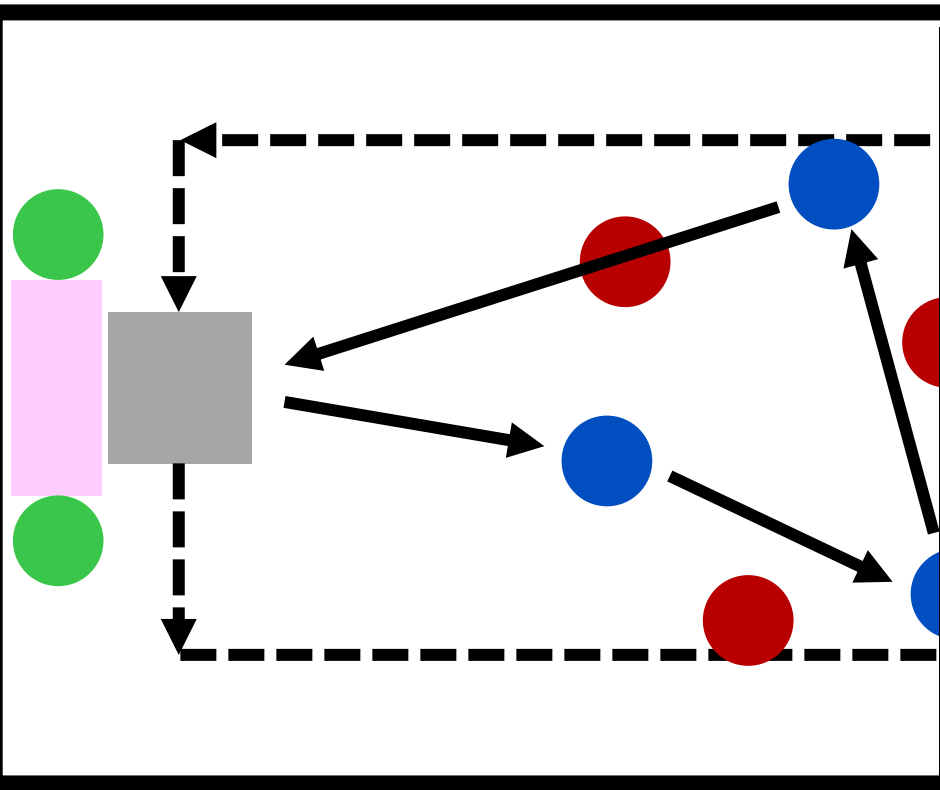


THANK YOU
FOR LISTENING



ROLL's ROYCE

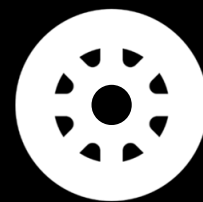




KEEP
CALM



and



ROLL'S
ROYCE