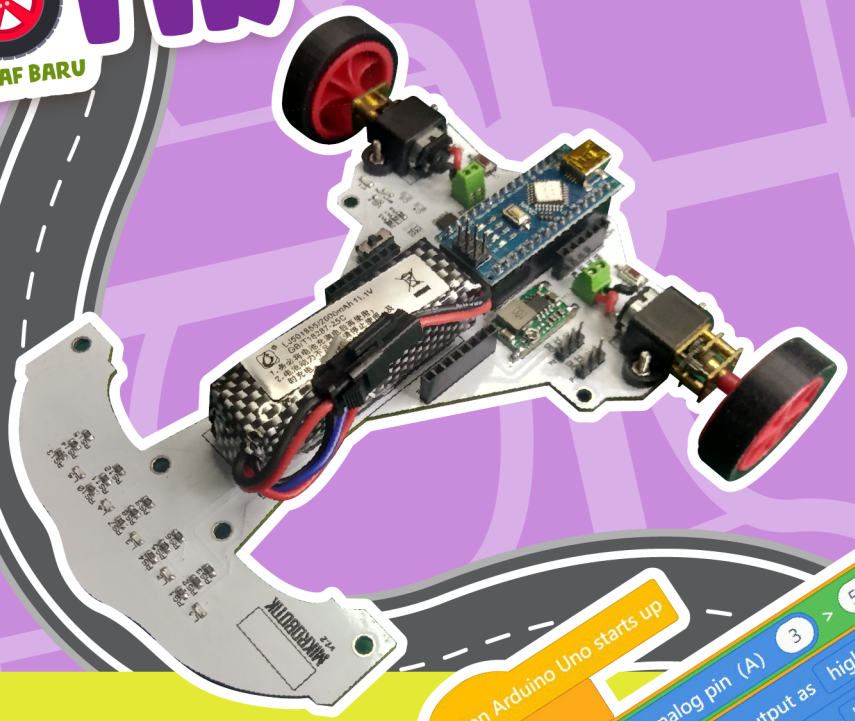
MCT
MICRO CONCEPT TECH

EKSPLORASI ILMU

MIKROBOTIK

KIT PEMBELAJARAN ROBOTIK ALAF BARU



- Robot pembelajaran dengan spesifikasi pertandingan.
- Pergerakkan berautonomi mengikut garisan.
- Pergerakkan bebas dengan kawalan 'Bluetooth'.
- Pengekodan grafik yang mudah dan seronok.

UMUR 7+

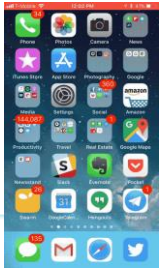
Contents

Mechanical Structure

Mechanical Movement

Element of Robotics

Electronic Hardware

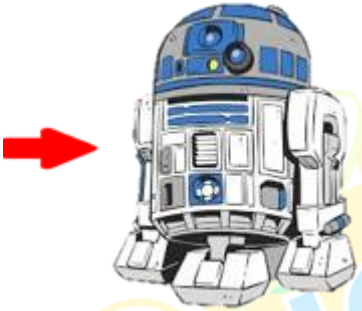


Software Coding

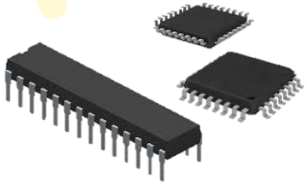
MICRO CONCEPT TECH

What is Electronic Hardware?

Detects and senses the surrounding



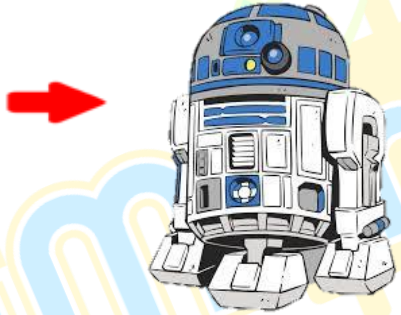
Controls or reacts to surrounding

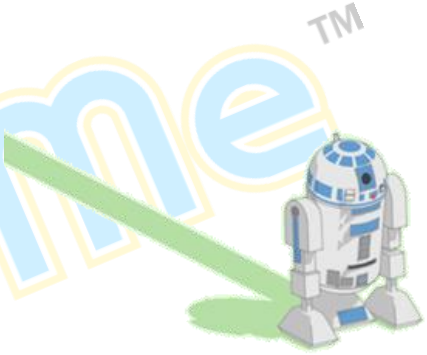


What is Software Coding?

Electronic Hardware Controller

Sets of instruction written in specific language





Operate

Path Finding Robot

Robot designed and built specifically to detect and autonomously follow white and black line. Besides, robot also designed for other functions such as obstacle detector and moving small objects.

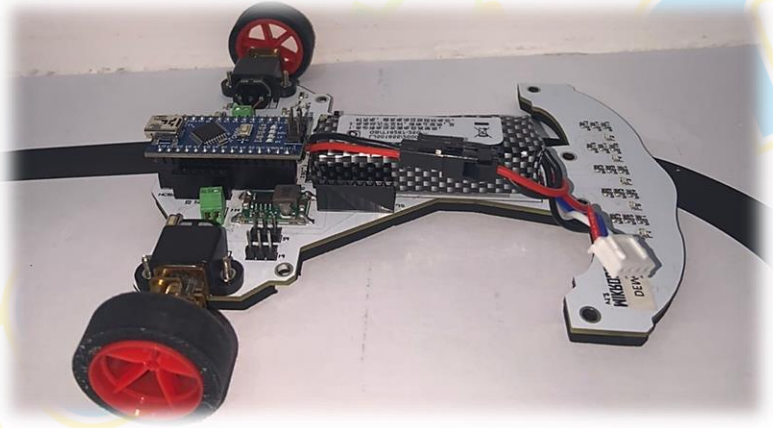
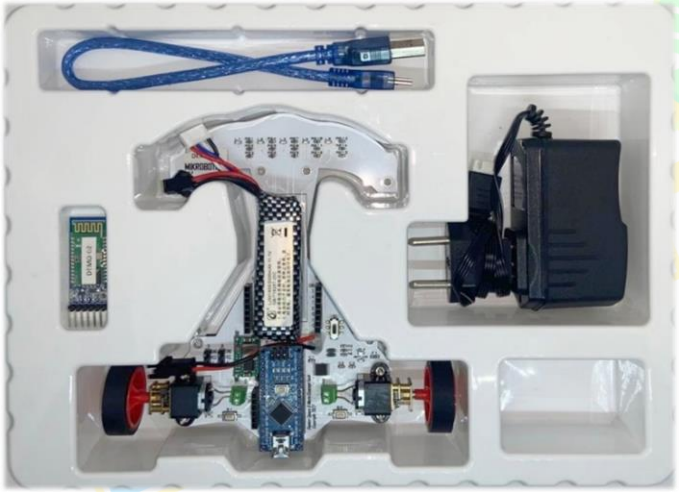


Figure 1: "Mikrobotik" Pathfinding Robot

Contents in the box

1x Usb Cable
1x Charger
1x Mikrobotik
1x Bluetooth Module
1x Mikrobotik Track



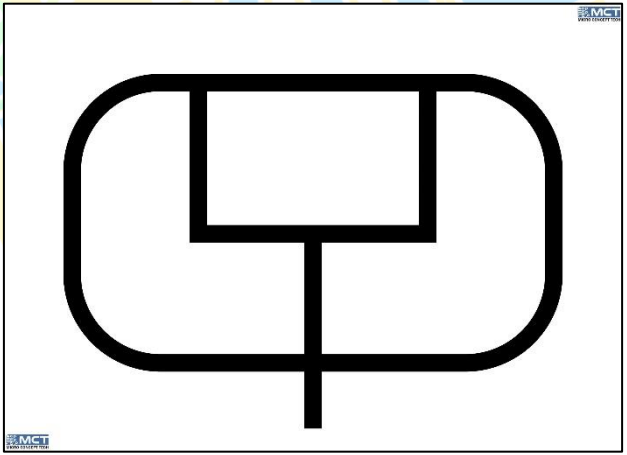
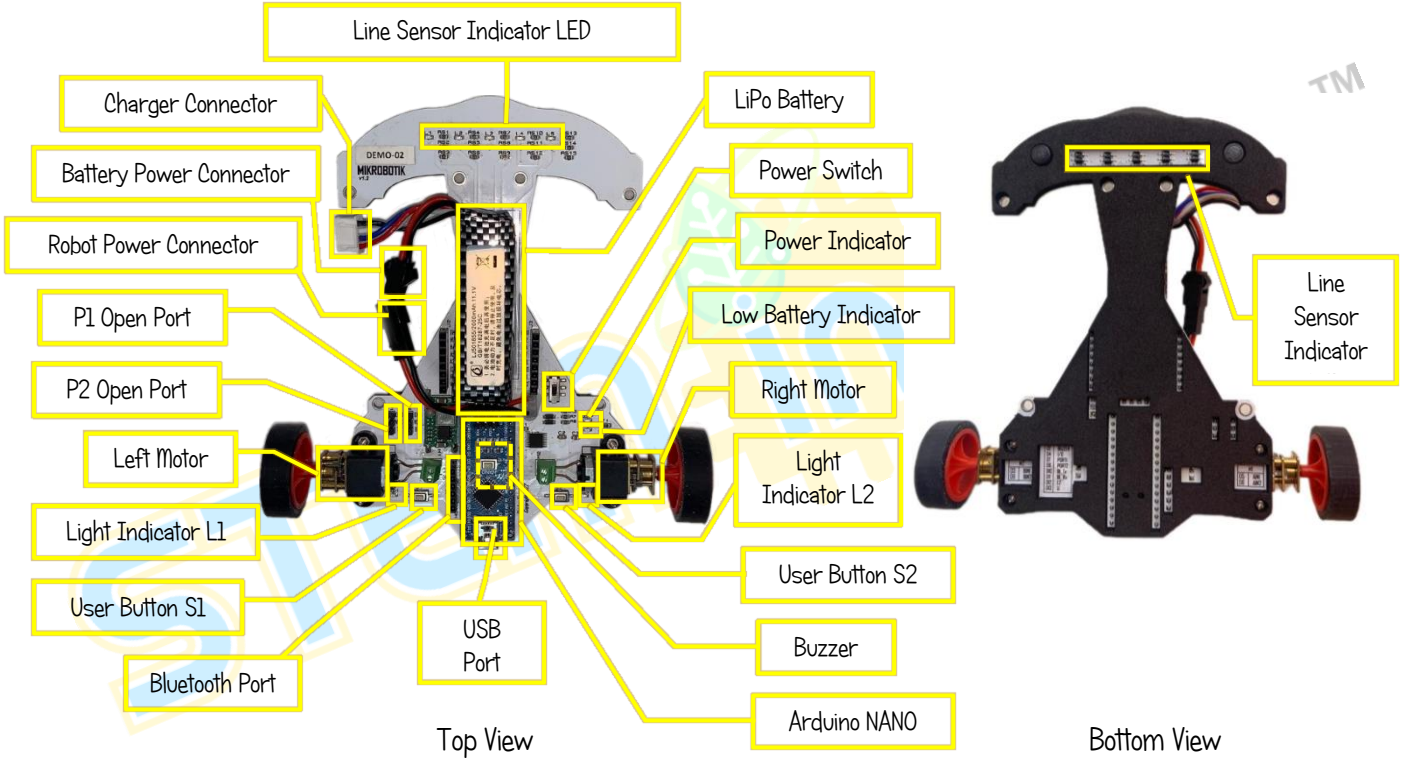


Figure 2: Mikrobotik Track

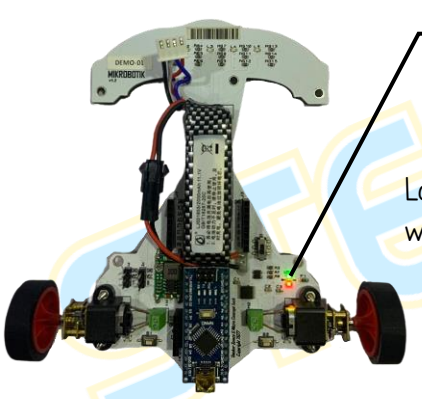
"Mikrobotik" Pathfinding Robot



Low Battery Indicator

Low battery indicator will light up red colour.
The lower the voltage value in battery, the brighter the indicator.
Minimum voltage operated: 11.0 V (Low battery indicator light at maximum bright)

User need to stop using Mikrobotik and start charging when low battery indicator lights at maximum.



Low battery indicator when battery is low.

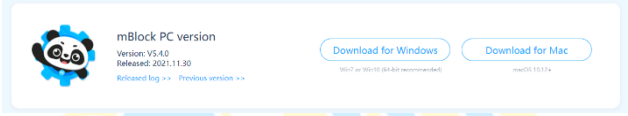
Low battery indicator when battery is full.

Installation of mBlock v5

Step 1 mBlock v5 software can be obtained from:

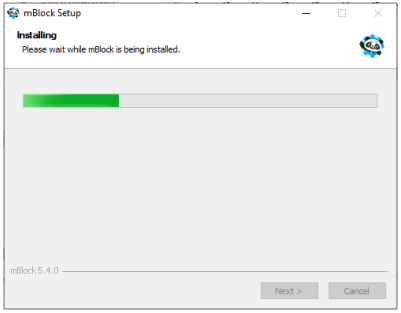
Link: https://mblock.makeblock.com/en-us/download/ @ QR:

Step 2 Download the latest version of mBlock v5 based on the computer operating system.

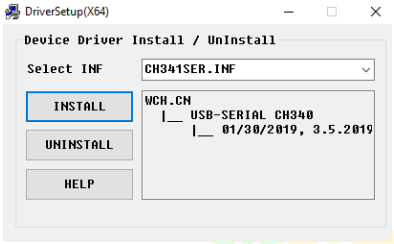


Step 3 Click mBlock v5 on your download location.

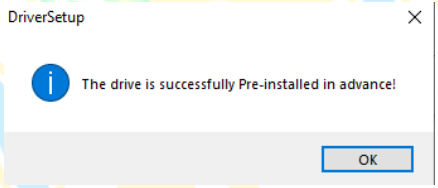
Step 4 Wait until mBlock v5 installation is complete.



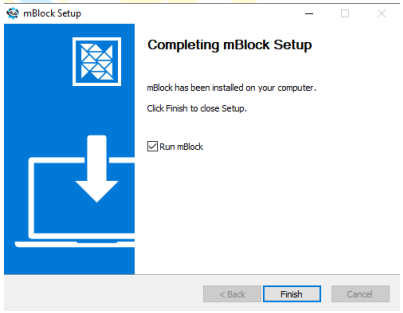
Step 5 Click '*INSTALL*'.



Step 6 Click OK and exit

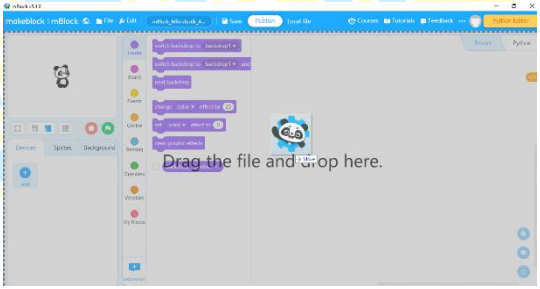


Step 7 Tick "*Run* mBlock".
Click "*Finish*".



Steps for Adding Mikrobotik

Step 1	Mikrobotik software can be obtained from: Link: https://www.microconcept.com.my/stem-robotic/download/
Step 2	Open mBlock v5
Step 3	Go to mikrobotik.mext file and drag into mBlock v5.
Step 4	Now, you can enjoy using mBlock v5!



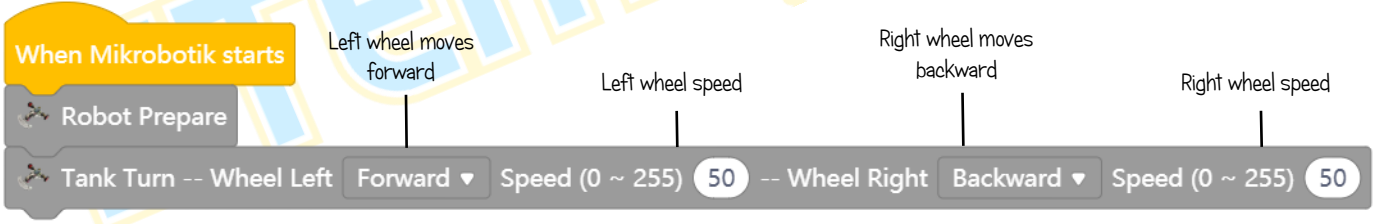
Calibration Process

The calibration process is an important process for the robot to identify between the white line and the black line. The calibration process for this Mikrobotik robot can be done both manually and automatically. This process is done before the robot can follow the line and complete the circuit.

Block Arrangement (Automatic Calibration):

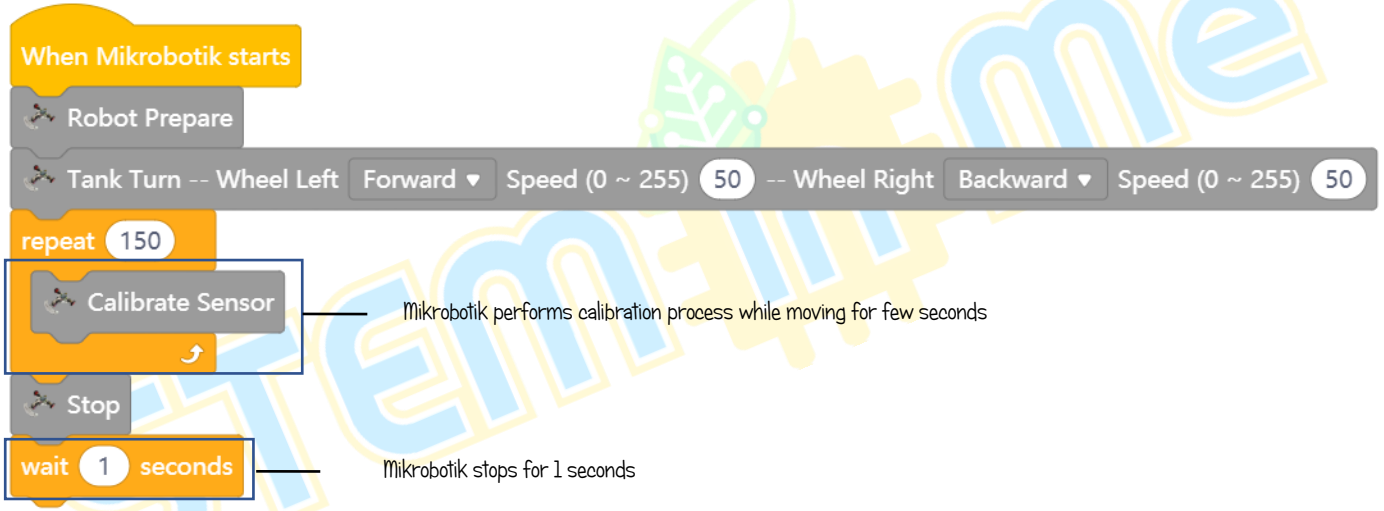
Step 1	Insert block *When Mikrobotik Starts* and combine with block *Prepare*.

Step 2	Next, combine block *Tank Turn (Wheel Left-Forward, Speed-50, Wheel Right-Backward, Speed-50)* under *Robot Prepare*.

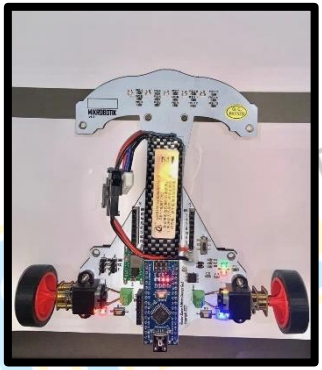


Step 3 Furthermore, combine block *Repeat* with block *Calibrate Sensor*. Combine these blocks with blocks in Step 2.

Step 4 Lastly, drag *Stop* and block *Wait (1 second)* and put under block *Repeat*.

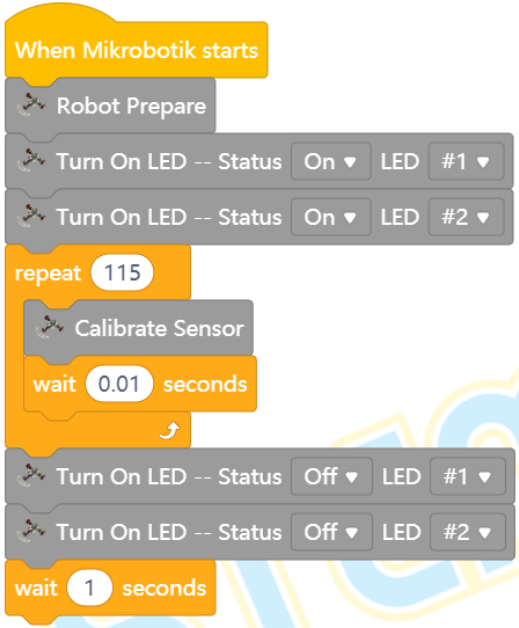


Steps for Automatic Calibration Process



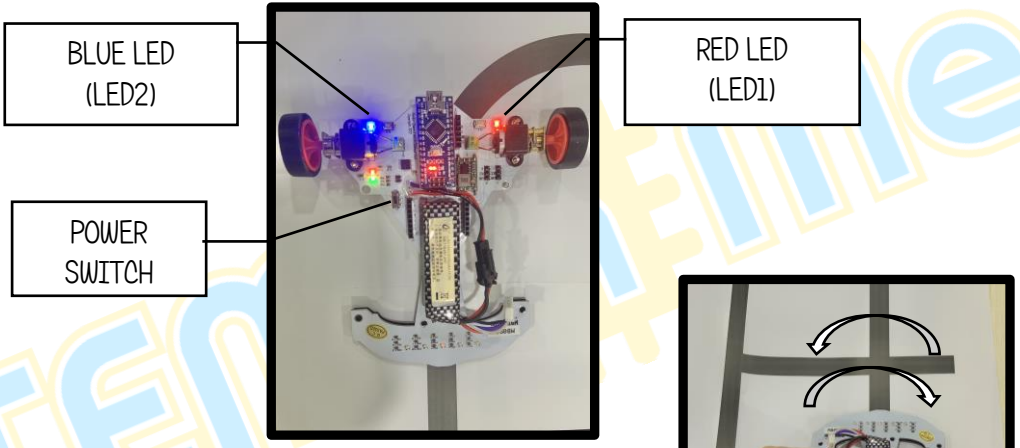
Step 1	Put Mikrobotik on the track. Make sure all sensors starting from TR1 (LED L1) until TR5 (LED L5) were put on black line.
Step 2	Switch on the power switch for Mikrobotik. Red LED1 light and blue LED2 will light up. The robot will automatically rotate to carry out the calibration process.

Block Arrangement (Manual Calibration):

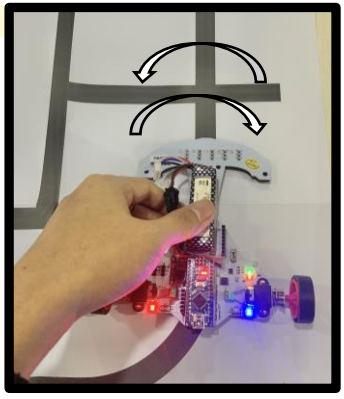


Steps for Manual Calibration Process

Step 1 Switch on Mikrobotik.
LED1 with red colour and LED2 with blue colour will light up.

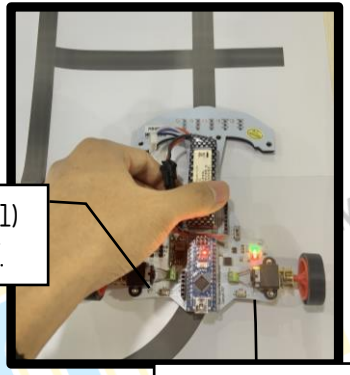


Step 2 Move all sensors starting from sensor labelled with TR1 (LED L1) until TR5 (LED L5) and bring back to TR1 (LED L1).



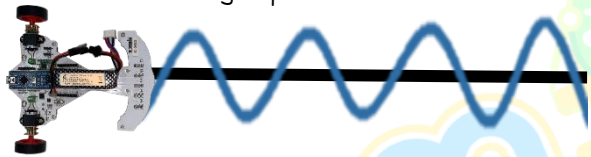
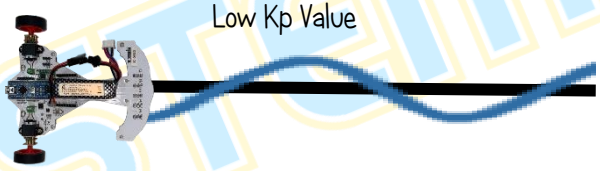
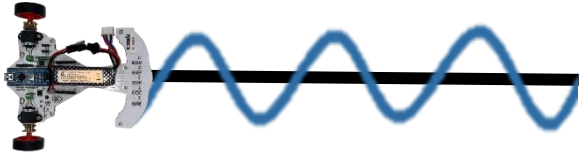
Step 3	Repeat movement in Step 2 until both LED (LED1 and LED2) light off.

Make sure all the sensors can detect the black line. The LED on the sensors will light up if the sensor detects a black line. For example LED L1 will light up if sensor TR1 detects a black line on the circuit.

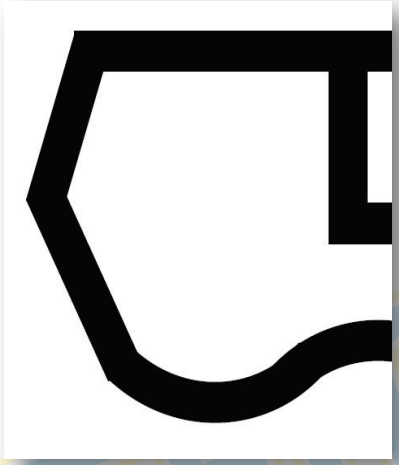


Autonomous Robot PID Algorithm

PID algorithm is a control strategy suitable to assist determine the direction and speed of the robot such that it autonomously drive and follows the line as close and fast possible. PID algorithm will ensure the robot does not overshoot from the line track when turning and moves straight along the line.

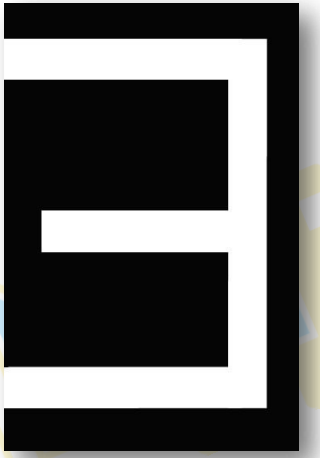
Kp	Kd
High Kp Value  Steers vigorously	High Kd Value  Counter steers early
Low Kp Value  Steers gently	Low Kd Value  Counter steers late

Type of Tracks



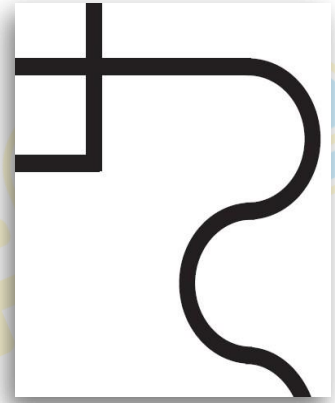
Black Line

(aprox 20mm)



White Line

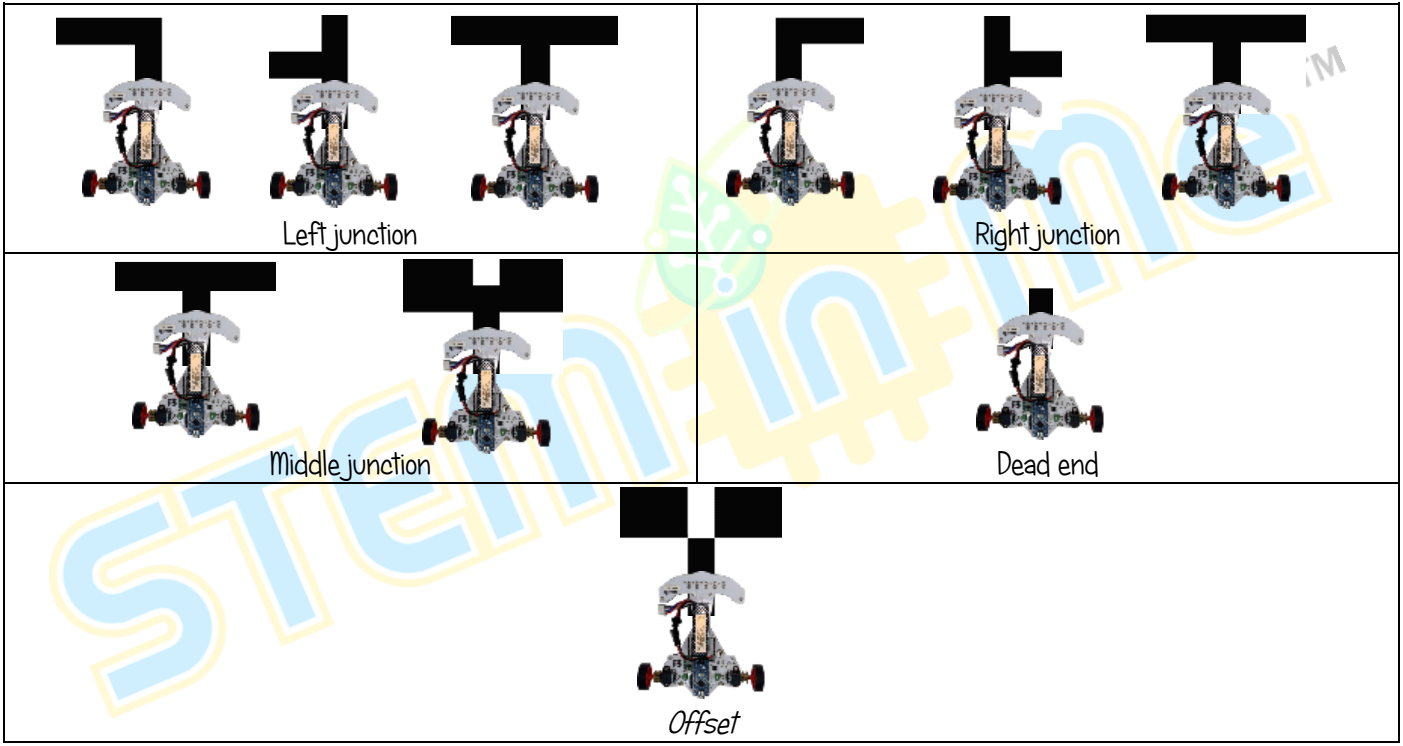
(aprox 20mm)



Black Thin Line

(aprox 10mm)

Type of Junctions



Objective 1: Vroom Vroom

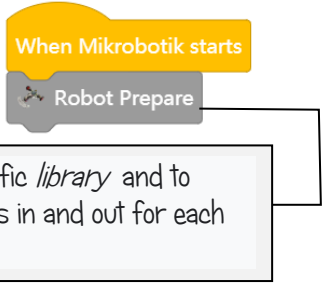
The robot will use a buzzer to produce a simple sound. It can only produce one tone at a time. This code block can be used to produce different tones to create an interesting sound pattern.

Introduction to Buzzer:

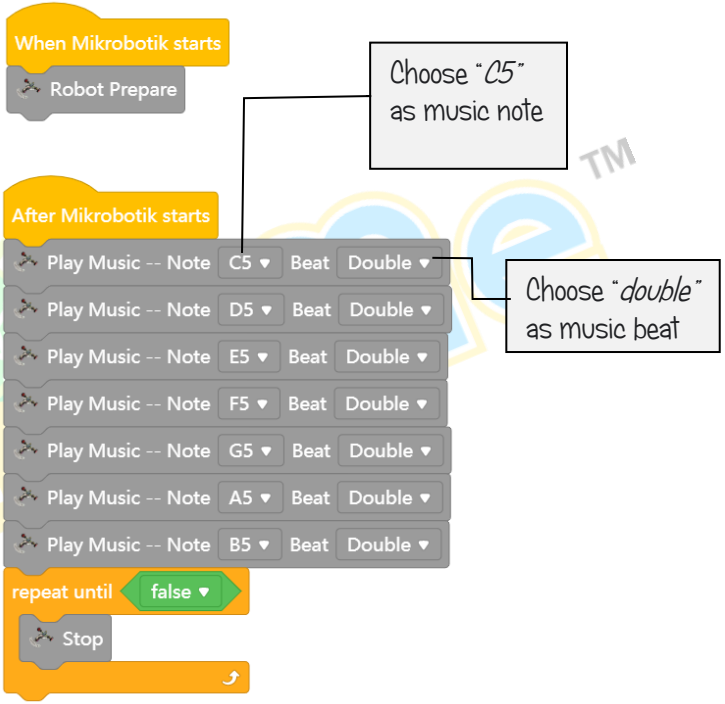
Buzzer is a type of sound device that converts audio models into sound signals. It is usually used for alarms.

Step by Step block arrangement:

Step 1 Combine block *When Mikrobotik starts* with block *Robot Prepare*.



Step 2	Next. combine block *After Mikrobotik starts* with block *Play Music (Note-C5. Beat-Double). (Note-D5. Beat-Double). (Note-E5. Beat-Double). (Note-F5. Beat-Double). (Note-G5. Beat-Double). (Note-A5. Beat-Double). (Note-B5. Beat-Double)*

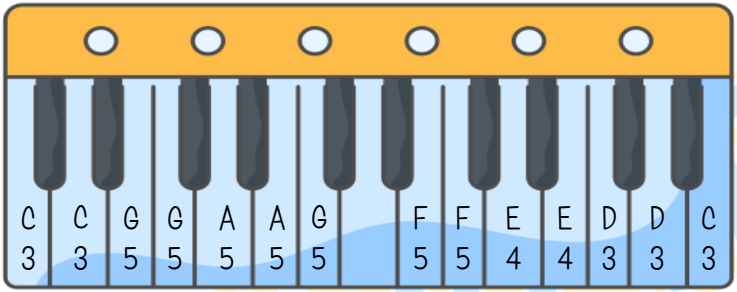


Step 3	Then. combine block *repeat until (false)* with block stop. Combine these blocks with blocks in Step 2.

Step 4	Once the program is uploaded. the robot will produce the sound or tone you have entered.

Challenge!!

In this challenge. you have to enter the provided musical notes and try to guess the name of the music.



Objective 2: Please Switch On The Lights!

Light-Emitting Diode (LED) on robot used as indicators. LEDs on the robot can be seen at power indicator, low battery indicator, LED1, LED2, Arduino NANO and line sensors.

Introduction to Light-Emitting Diode (LED)

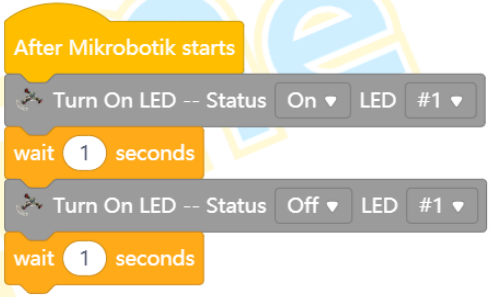
Light-Emitting Diode or LED functions to emit light and convert electric current into light. Used as an application for indicator and light source.

Step by Step block arrangement:

Step 1 Combine block *When Mikrobotik starts* with block *Robot Prepare*.

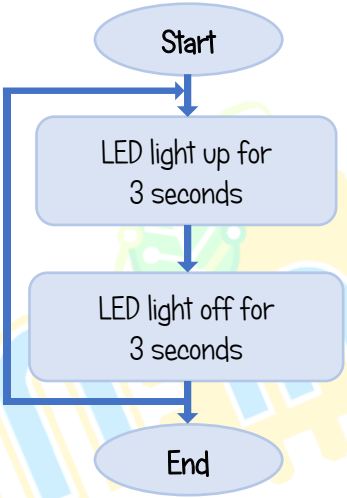
Step 2	Combine block *After Mikrobotik starts* with block *Turn On LED* with choice of *Status On* and LED #1 and block *wait 1 second*. Put the blocks under the block in Step 1. This program will light up the LED.

Step 3	Add another block *Turn On LED* with choice of *Status Off* and LED#1 with block *wait 1 second* and combine with block in Step 2 to switch off the LED.



Step 4	Lastly, upload the program. After the program uploaded, LED 1 will light up in one second and light off in one second. The program will continue to run until the robot is turned off by the user.

Challenge!!



Program above show LED will light up in three second and light off in three second. The program will continue to run until the Mikrobotik is turned off by the user.

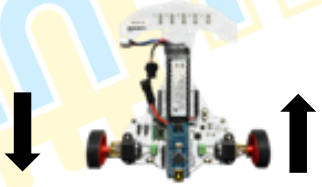
Objective 3: Our Adventure Begin (Free Movement)

The robot is moved using the code block "tank turn" to move without following the line. This code block is suitable for solving maze circuits (labyrinth). The robot will move depending on the speed and direction of the left and right motors set by user.

Introduction to Basic of Free Movement:

Moves Forward	Moves Backward
	
Left motor forward, right motor forward left motor speed = right motor speed	Left motor backward, right motor backward left motor speed = right motor speed

Steer to right	Steer to left
	
Left motor forwards, right motor stops	Left motor stop, right motor stops

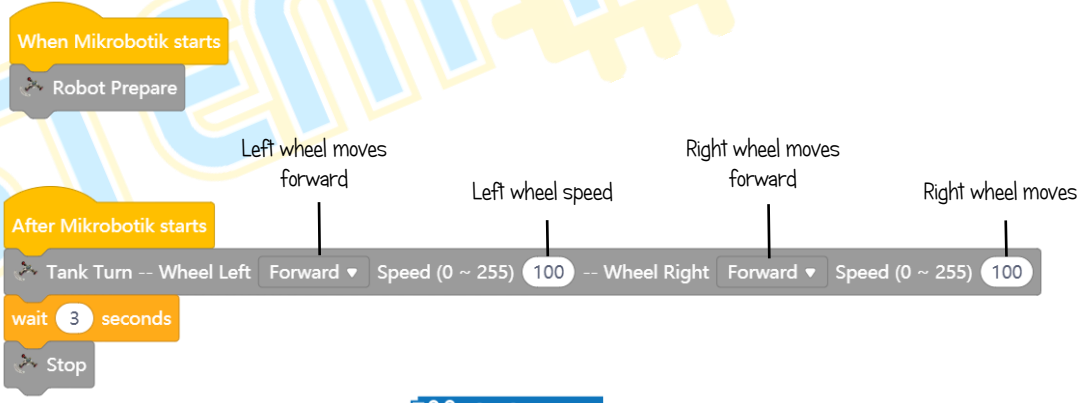
Turn to right	Turn to left
	
Left motor forward, right motor backward left motor speed = right motor speed	Left motor backward, right motor forward left motor speed = right motor speed

Step by Step block arrangement

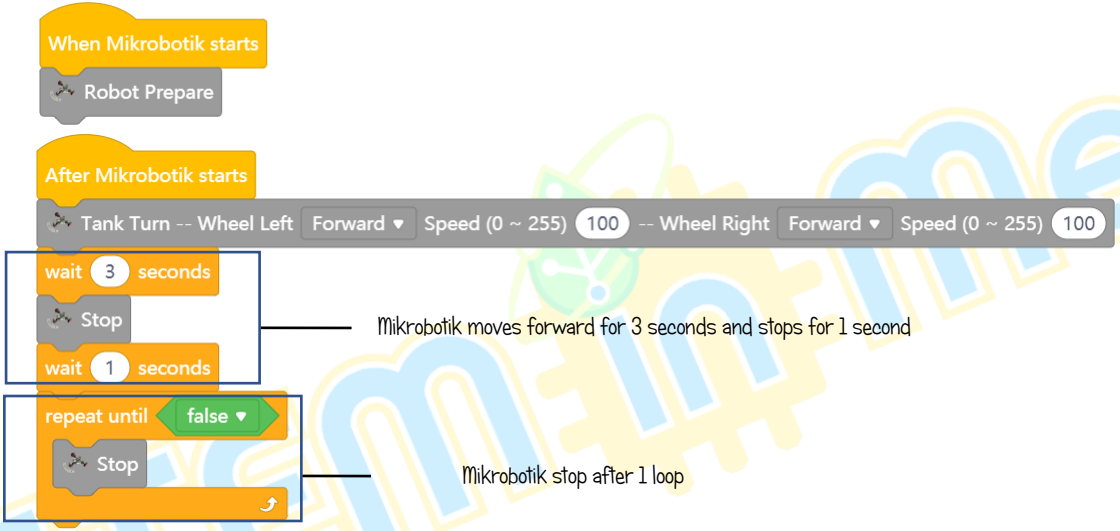
i) Move forwards

Step 1	Combine block *When Mikrobotik starts* with block *Robot Prepare*.

Step 2	Combine block *After Mikrobotik starts* with block *Tank Turn* (*Wheel Left -Forward*. *Speed*-100. *Wheel Right-Forward*. *Speed*-100). block *wait* (*3 seconds*) and block *stop*. Drag those blocks and put it under the blocks in Step 1. Left motor and right motor will move with the same speed.



Step 3 Lastly, combine block *wait* block *repeat until (false)* and block *stop* and combine those blocks with blocks in Step 2.

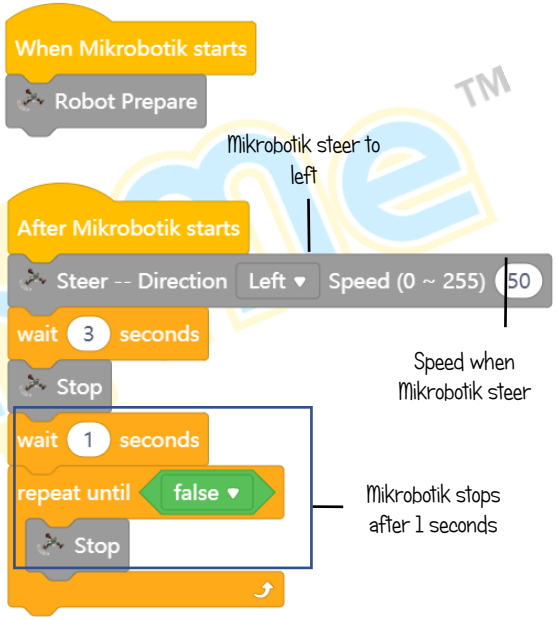


Step 4 After the program uploaded. Mikrobotik will move forwards for duration of 3 seconds.

Step to make the robot move backwards is the same as step to move forwards. You only need to change the direction of the left wheel to backward and the direction of the right wheel to backward.

ii) Steer to left

Step 1	Combine block *When Mikrobotik starts* with block *Robot Prepare*.
Step 2	Combine block *After Mikrobotik starts* with block *Steer (Direction -Left, Speed-50)*. Then, add block *wait (3 seconds)* and block *stop*. Left motor will stop and right motor will move forwards with the set up speed.
Step 3	Lastly, combine block *wait (1 second), repeat until (false)* and block *stop* and combine those blocks with blocks in Step 2.



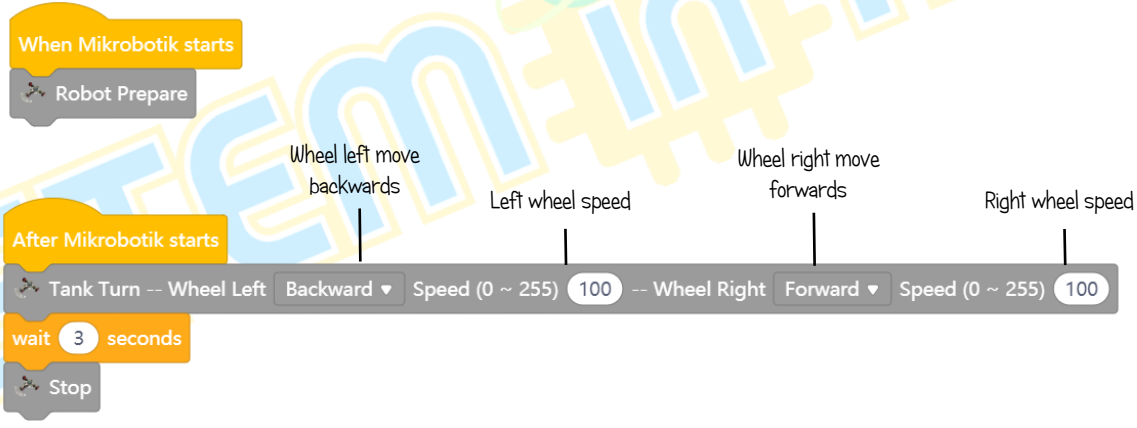
Step 4 After the program uploaded, Mikrobotik will steer to left for duration of 3 seconds.

> Step for steer to the right is similar to the step for steer to the left. You just need to change the direction to right.

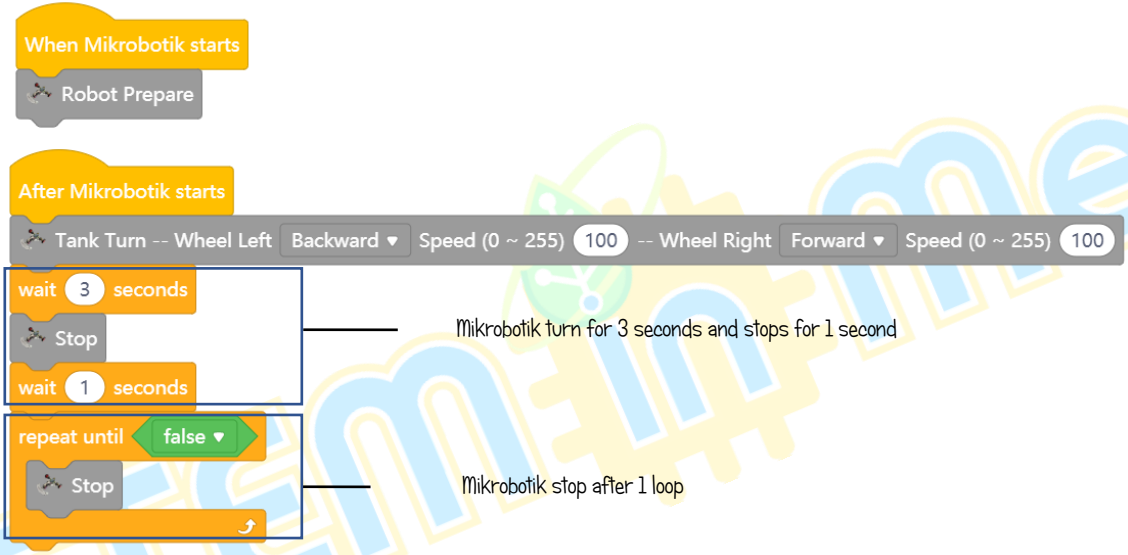
iii) Turns left

Step 1 Combine block *When Mikrobotik starts* with block *Robot Prepare*.

Step 2 Combine block *After Mikrobotik starts* with block *Tank Turn (Wheel Left -Backward. Speed*-100. *Wheel Right-Forward. Speed*-100). block *wait (3 seconds)* and block *stop*. Drag those blocks to under block in Step 1. The left motor will move backward and the right motor will move forward with the same speed.



Step 3 Lastly, combine block *wait (1 second)*, *repeat until (false)* and block *stop*. Then, combine those blocks with blocks in Step 2.

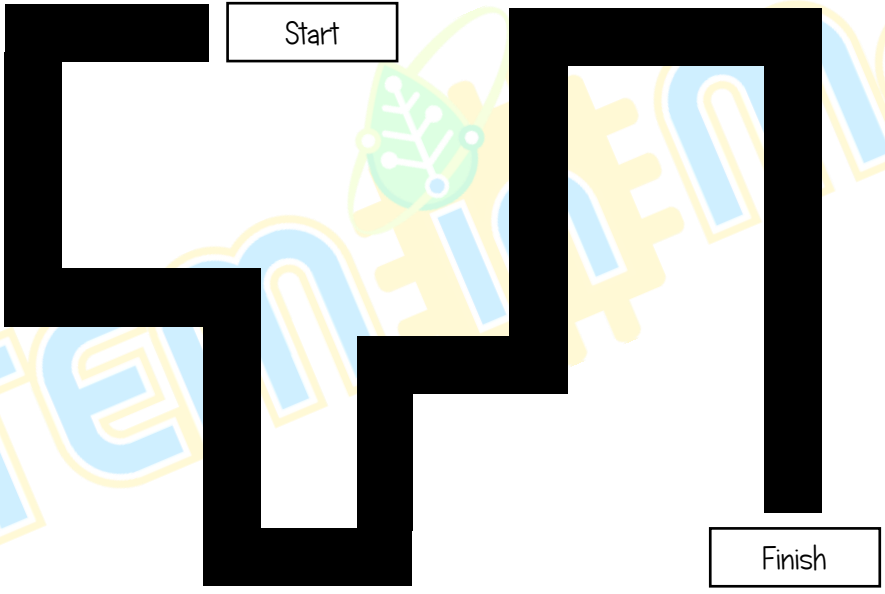


Step 4 After program uploaded. Mikrobotik turns left for 3 seconds and stops.

Step for turn to the right is similar to the step for turn to the left. You just need to change the wheel left direction to forward and wheel left direction to backward.

Challenge!!

In this Challenge. you need to ensure that Mikrobotik moves along the path that has been prepared by applying the knowledge that has been learned.



Objective 4: Let's Follow The Line!

The robot will follow the line (Black or White) continuously. The robot will always move even if it meets a left or right junction.

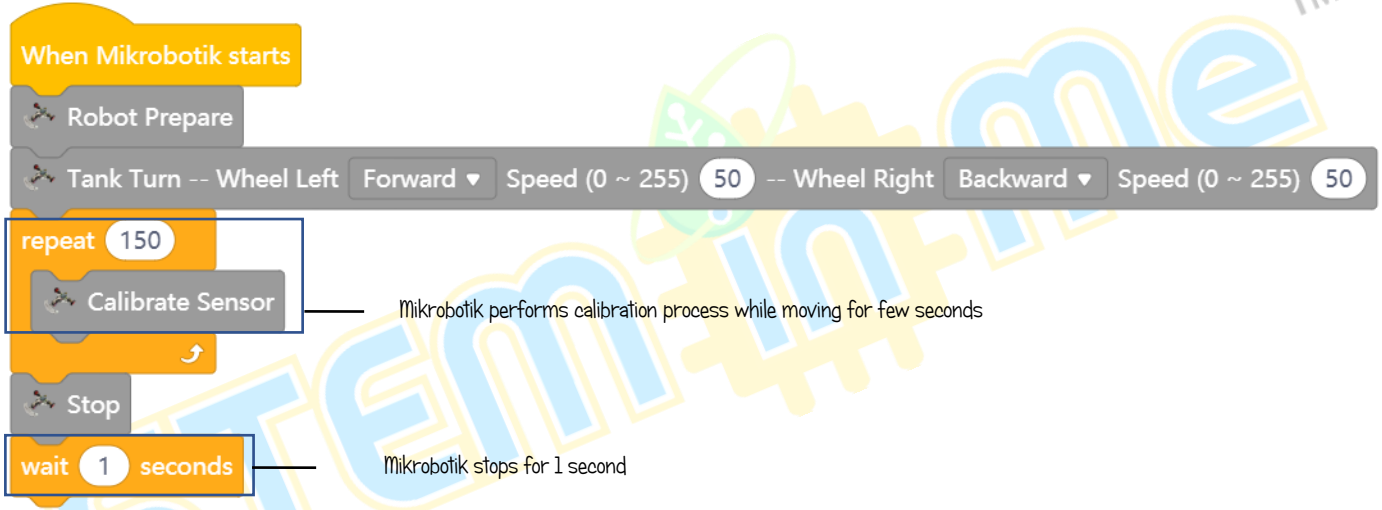
Introduction to Line Tracer Time and its Mechanism

Line Tracer Time used to make Mikrobotik follow along line with Black or White colour until it reach maximum time period (in ms).

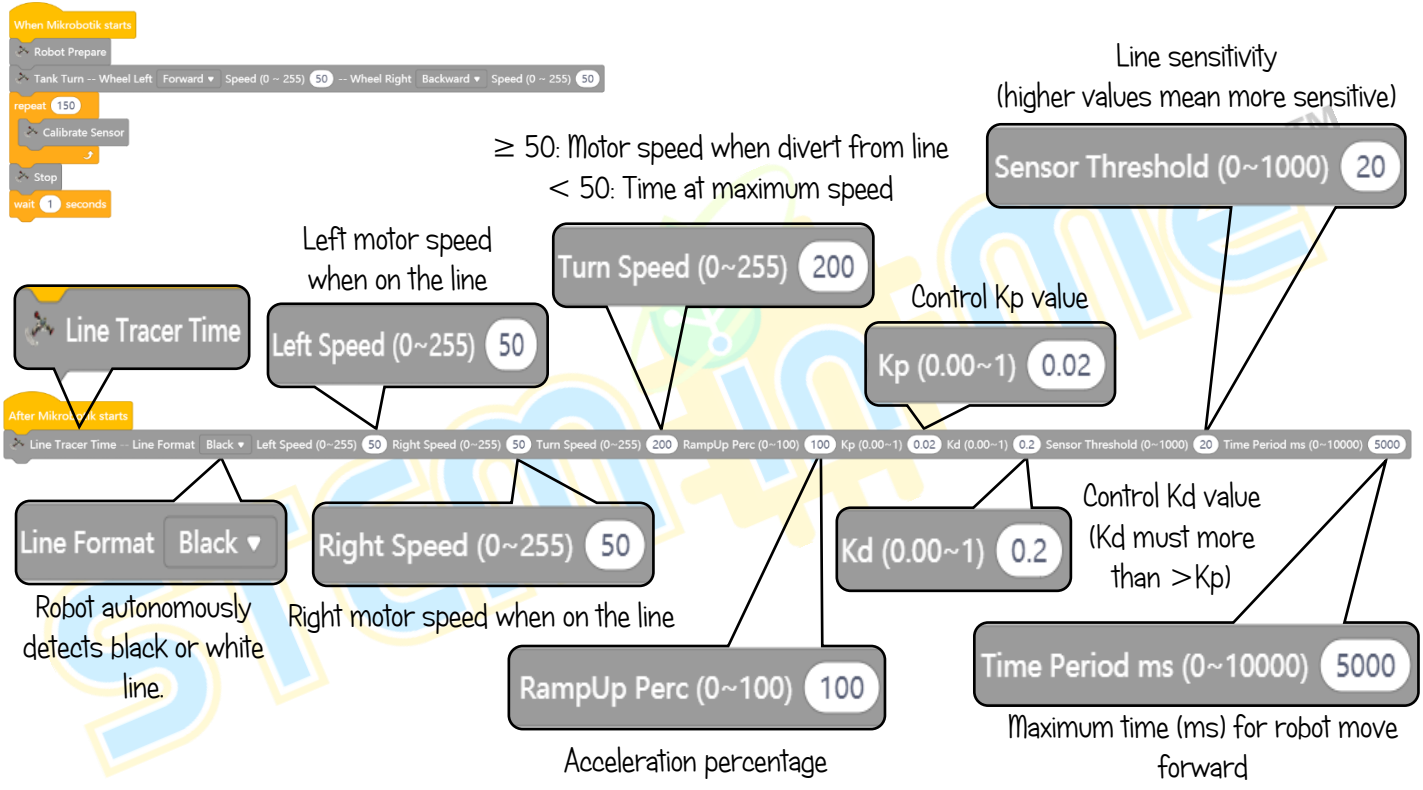
When Mikrobotik reaches the maximum time period. Mikrobotik will stop. Mikrobotik will move continuously without making turn when meeting left junction. right junction and middle junction.

Step by Step block arrangement:

Step 1 Prepare the arrangement blocks for automatic calibration.



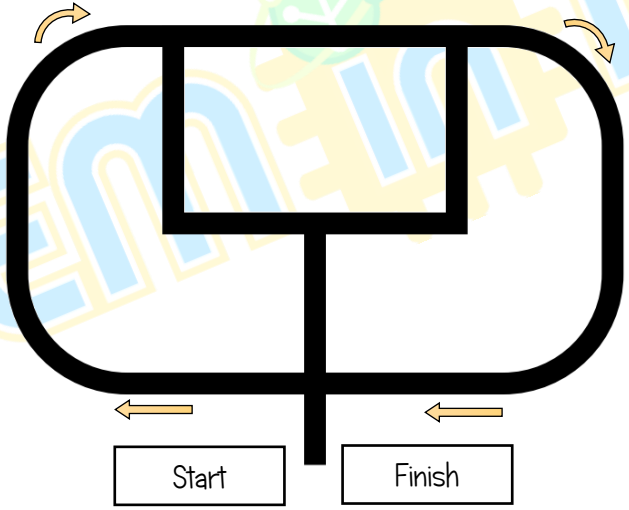
Step 2 Then, insert block *After Mikrobotik Starts* and combine with block *Line Tracer Time*.



Step 3	After uploading the code. Mikrobotik will start moving forward temporarily. Perform the calibration process on the line sensor. After that. Mikrobotik will follow the line of either Black or White until it reaches the maximum time period (in ms).

Challenge!!

Apply *Line Tracer Time* to solve the track below.



Objective 5: What To Do When Meeting Junction?

Robot will move autonomously and can decide whether to turn left. turn right or stop at the junction. The technique used to know as *Steer Turn Method*.

Introduction to *Path Finder* and its Mechanism

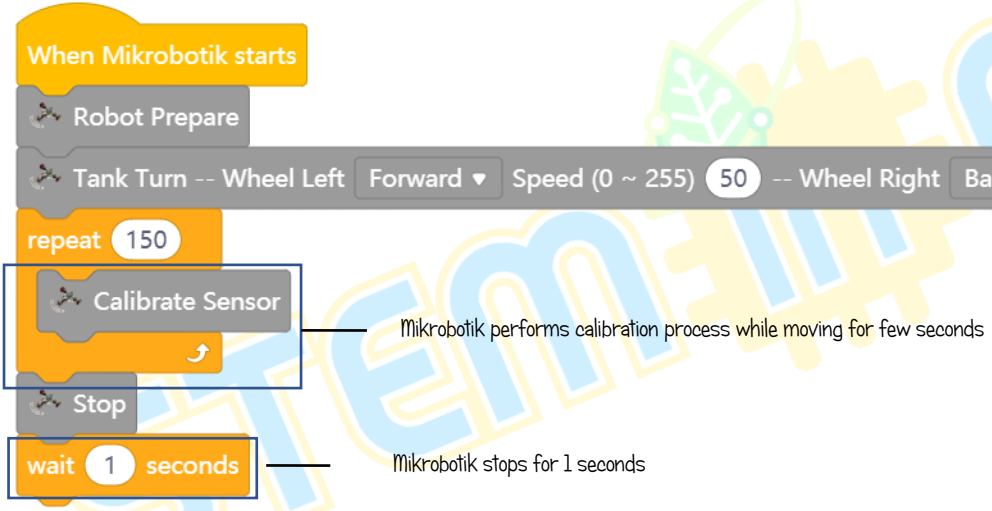
Path Finder is used to move the Mikrobotik autonomously follow over a white or black line until the Mikrobotik finds a junction (right or left or middle or dead end or offset).

At the junction. Mikrobotik will act to turn (left or right or stop) for a set period of time or until the robot finds the next line and will stop.

Robot will turn by using *Steer Turn Method*.

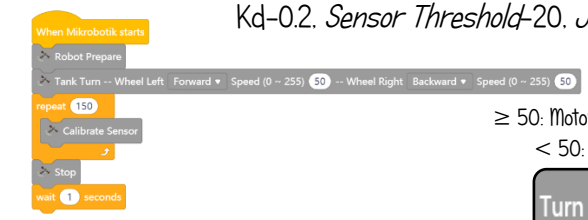
Step by Step Block Arrangement:

Step 1 Prepare the blocks arrangement for automatic calibration.



Step 2

Lastly, combine block *After Mikrobotik Starts* and block *Path Finder (Line Format-Black. Junction-Left. Action-Turn Left. Left Speed*-50. *Right Speed*-50. *Turn Speed*-200. *RampUp Perc*-100. Kp-0.02. Kd-0.2. *Sensor Threshold*-20. *Junction Speed*-50. *Forward Delay*-50. *Turn Period ms*-300).



Robot tracking and move forward until meeting junction

Junction Left

Left motor speed when on the line

Left Speed (0~255) 50

≥ 50: Motor speed when divert from line
< 50: Time at maximum speed

Turn Speed (0~255) 200

Control Kp value

Kp (0.00~1) 0.02

Line sensitivity (higher values mean more sensitive)

Sensor Threshold (0~1000) 20

Time for robot move forward after meeting junction and make action

Forward Delay 50

Path Finder

Action Turn Left Left

Robot action when meeting junction

RampUp Perc (0~100) 100

Acceleration percentage

Junction Speed 50

Turning speed at junction (higher value. sharper turns)

Line Format Black

Right Speed (0~255) 50

Kd (0.00~1) 0.2

Turn Period ms (0~10000) 300

Robot autonomously detects black or white line.

Right motor speed when on the line

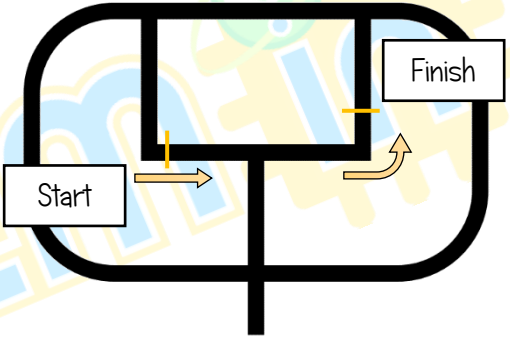
Control Kd value (Kd must more than >Kp)

On time (ms) for robot to turn or till find line (0)

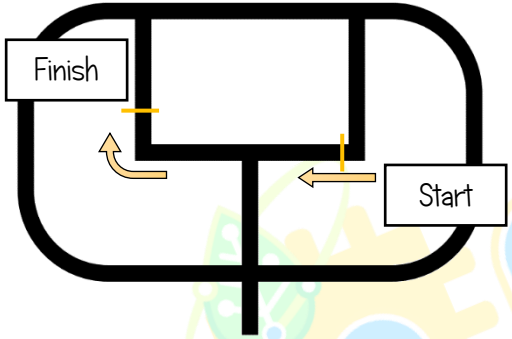
Step 3	After uploading the code. turn on the Mikrobotik switch and perform the calibration process. After that. Mikrobotik will follow the black line and if the robot meets a left junction. Mikrobotik will move forward and then turn to enter the left junction until Mikrobotik meets another line.

Challenge!!

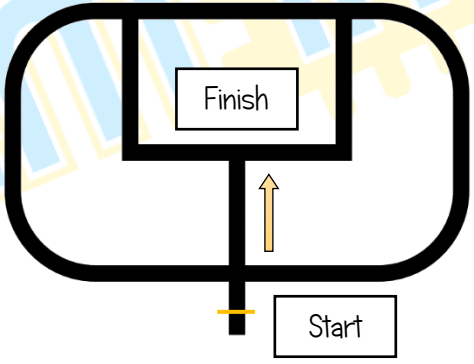
i) *Path Finder* Left Junction. steer left at left junction.



ii) *Path Finder* Right Junction. steer right at right junction.



iii) *Path Finder* Middle Junction. stop.



Objective 6: What Else Can Be Done When Meeting Junction?

Robot will move autonomously and can decide whether to turn left. turn right or stop at the intersection. The technique used known as *Tank Turn Method.*

Introduction to *Path Finder Tank* and its Mechanism

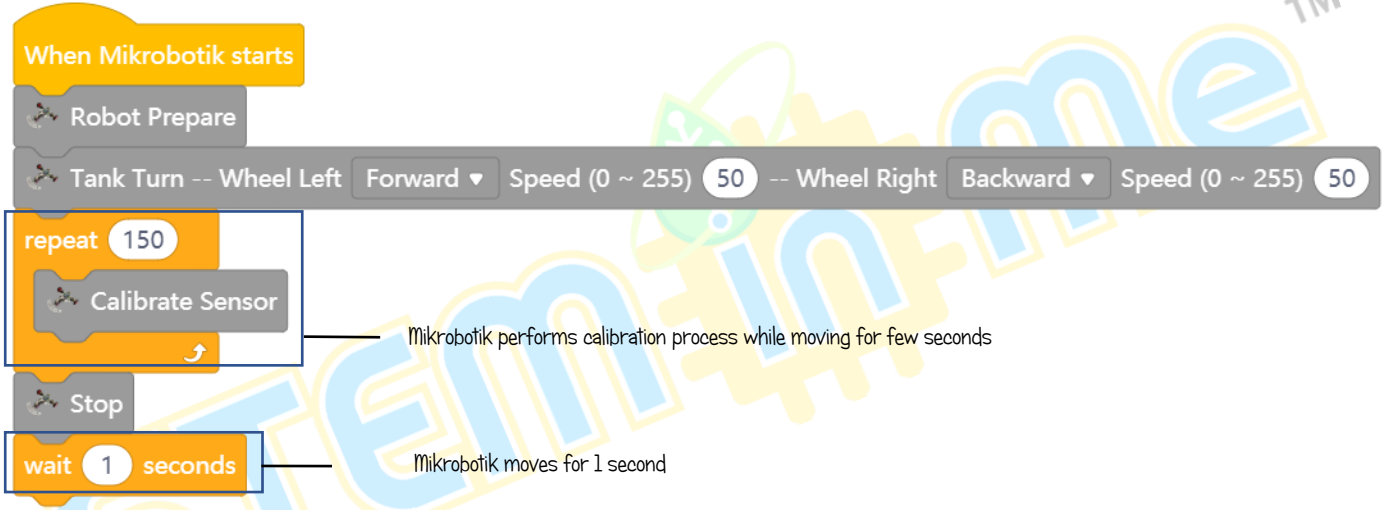
Mikrobotik travel autonomously on lines (Black or White or Thin Black or Thin White) until meet an intersection (Left or Right or Middle or Dead End or Offset).

At the junction. Mikrobotik will act (Turn left or Turn right or Stop) for at least the Minimum Turn Period *(Min Turn Period)* and continue turning until it detects the line and stops.

Mikrobotik will turn by using *Tank Turn Method.*

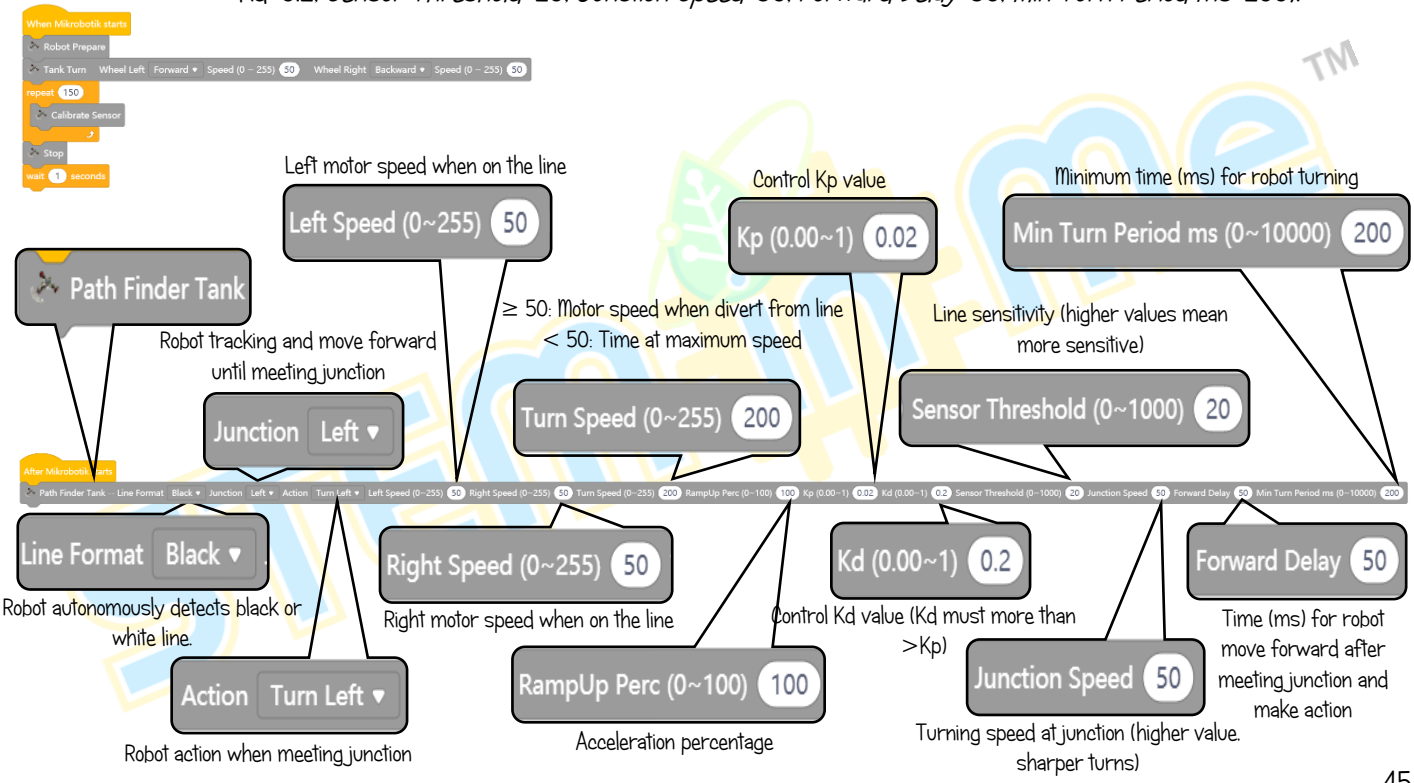
Step by Step block arrangement:

Step 1 Prepare the blocks arrangement for automatic calibration.



Step 2

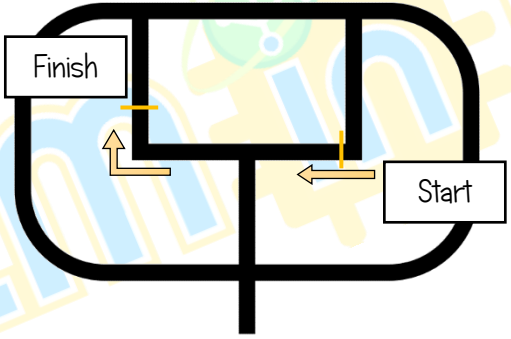
Combine block *After Mikrobotik starts* with block *Path Finder Tank (Line Format- Black. Junction- Right. Action-Turn Left. Left Speed*-50. *Right Speed*-50. *Turn Speed*-200. *RampUp Perc*-100. Kp-0.02. Kd-0.2. *Sensor Threshold*-20. *Junction Speed*-50. *Forward Delay*-50. *Min Turn Period ms*-200).



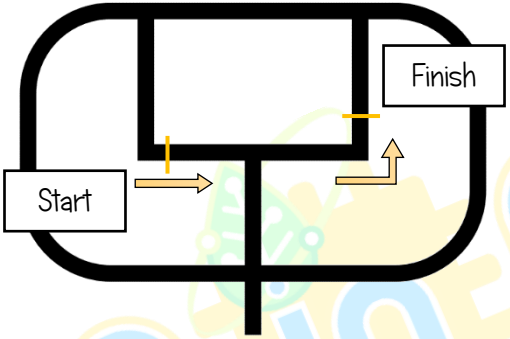
Step 3	After uploading the code. turn on the Mikrobotik switch and perform the calibration process. After that. Mikrobotik will follow the black line and if the robot finds a left junction. Mikrobotik will move forward and then turn for at least the Minimum Turn Period *(Min Turn Period)* and continue turning until it detects the line and stops.

Challenge!!

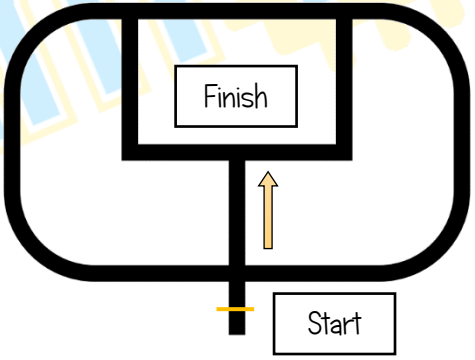
i) *Path Finder Tank* Right Junction. turn at right junction



ii) *Path Finder Tank* Left Junction, turn at left junction



iii) *Path Finder Tank* Middle Junction, stop



Objective 7: Wrong way? Make U-turn

Mikrobotik can make a U-turn on the line it passes through 180 degrees on its axis and turn left or right during the Minimum Turn Duration (Min Turn Period) and continue until it meets the line (Black or White).

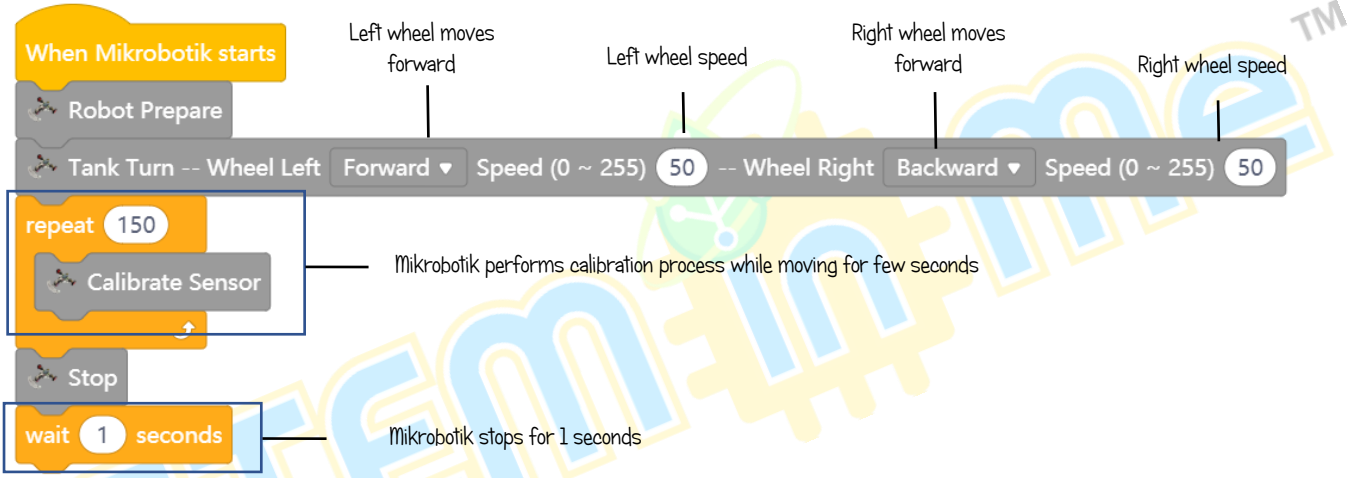
Introduction to *Turn At Centre* and its Mechanism

Mikrobotik will make a tank turn in the direction (left or right) for the Minimum Turn Duration (Min Turn Period) until the robot finds the line and finally stops.

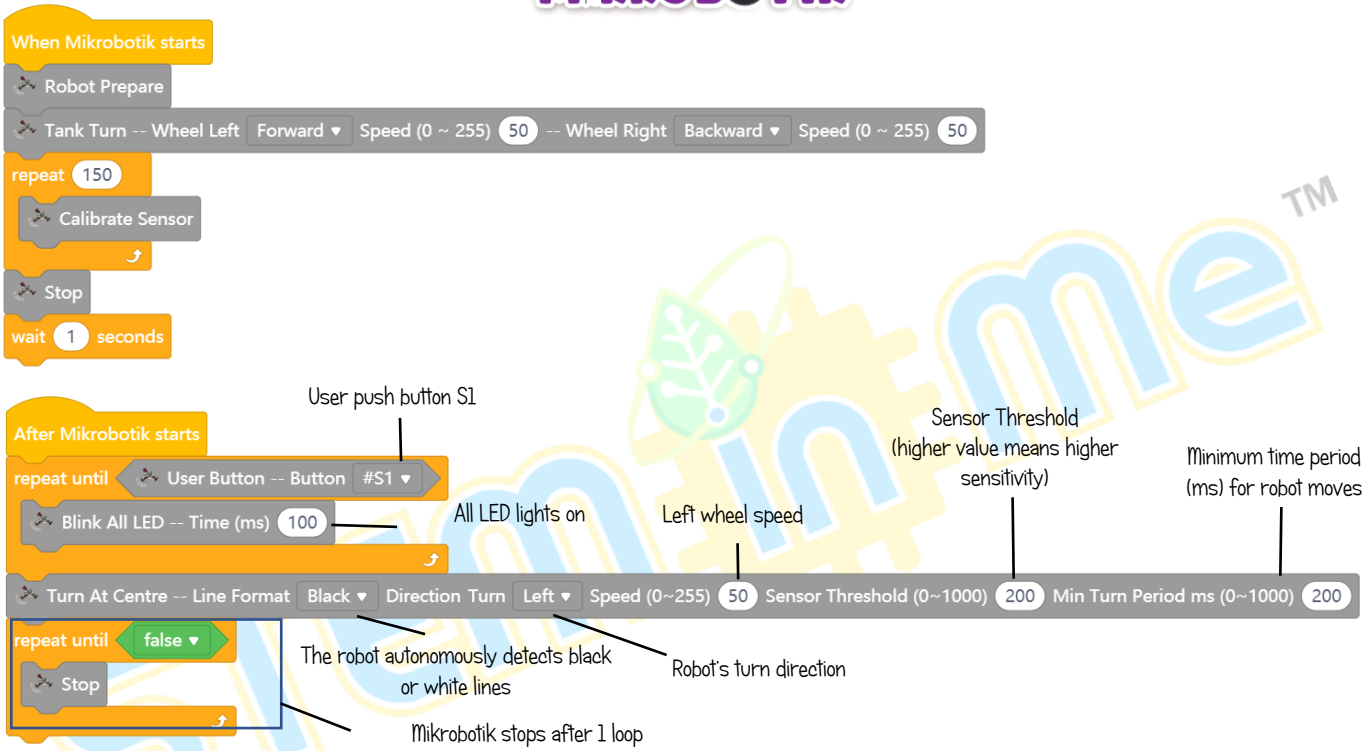
This technique is useful for making U-turns.

Step by Step blocks arrangement:

Step 1 Prepare the arrangement blocks for automatic calibration.



Step 2 Combine block *Find Lind (Line Format- Black. Direction-Forward. Left Speed*-100. *Right Speed*-100. *RampUp Perc*-100. *Sensor Threshold*-20. *Forward Delay*-0) with block *wait (1 second).* Combine those blocks with blocks in Step 4.

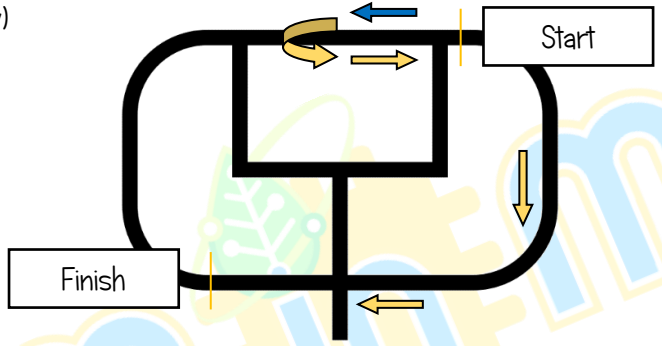


Step 3

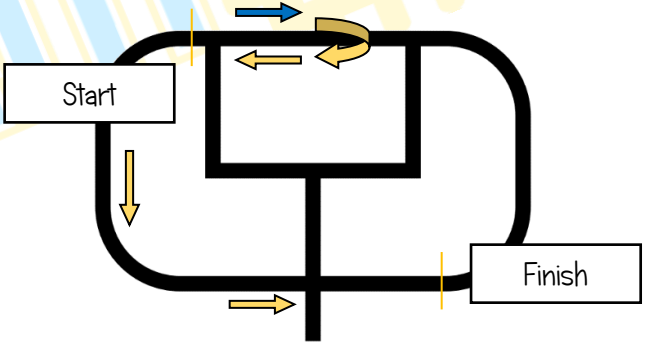
After uploading the code. turn on the Mikrobotik switch and perform the calibration process. After that. Mikrobotik will make a U-turn according to the set direction and will stop after detect the black line.

Challenge!!

i) U-turn at the middle. left direction and move forward.
(blue arrow to yellow arrow)



ii) U-turn at the middle. left direction and move forward.
(blue arrow to yellow arrow)



Objective 8: Let's Control Mikrobotik

Bluetooth is a short distance wireless technology used or data exchange between fixed and mobile devices in close range. Besides, it can build private network area. Bluetooth allows Mikrobotik to exchange the desired data with other devices directly.

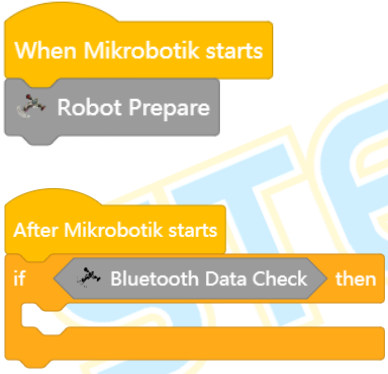
Introduction to Bluetooth and its Mechanism

Mikrobotik can be control in close range using the Bluetooth approach as it can be find and control easily. The Bluetooth module inserted into the port provided. This Bluetooth module contains 4 pins which are RXD, TXD, GND, and VCC.

Step by Step block arrangement:

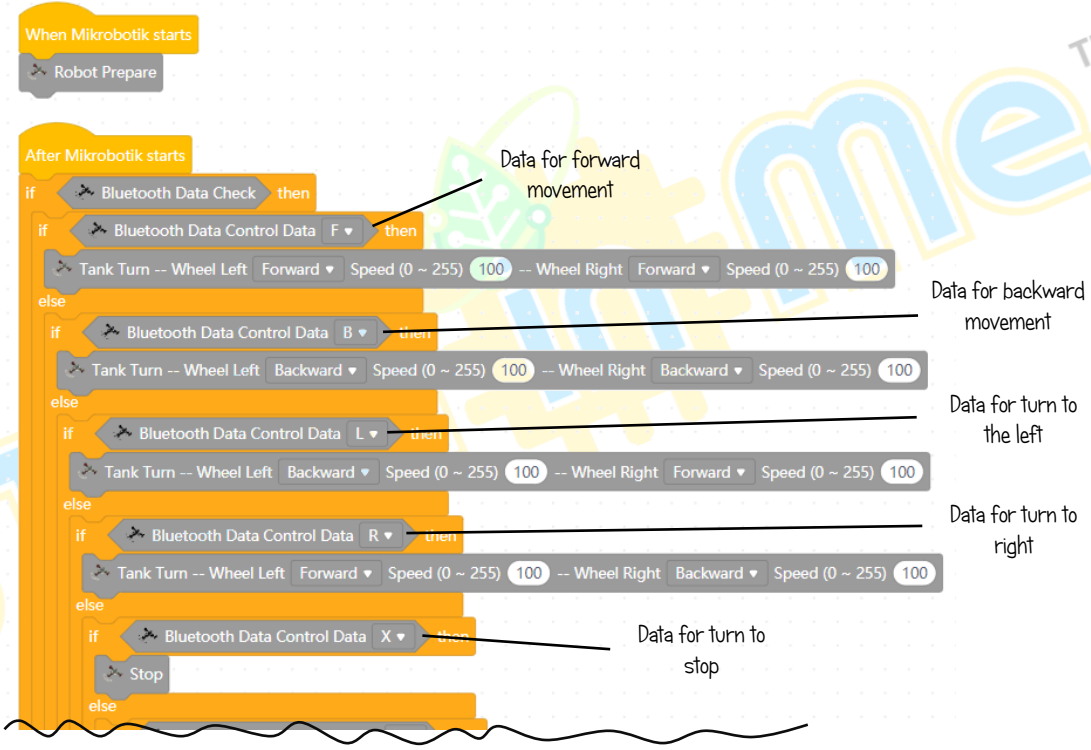
Step 1	Insert block *When Mikrobotik Starts* and combine with block *Robot Prepare*

Step 2	Combine block *After Mikrobotik starts* with block *if* after combined with block *Bluetooth Data Check*. Put those blocks under the block in Step 1.



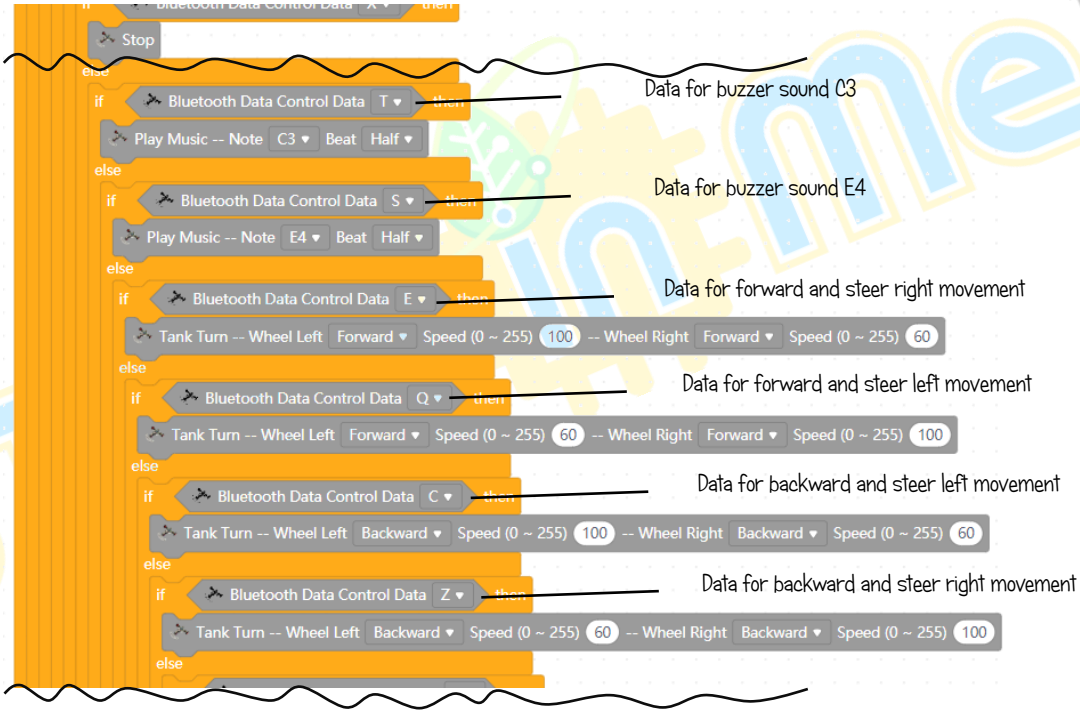
Step 3

Under the block *After Mikrobotik starts*, combine 5 blocks *Bluetooth Data Control Data (F, B, L, R, X)* with 5 blocks *if* and under the block *then* combine with 5 blocks *Tank Turn* to get forward, backward, turn to left, turn to right and stop movement.



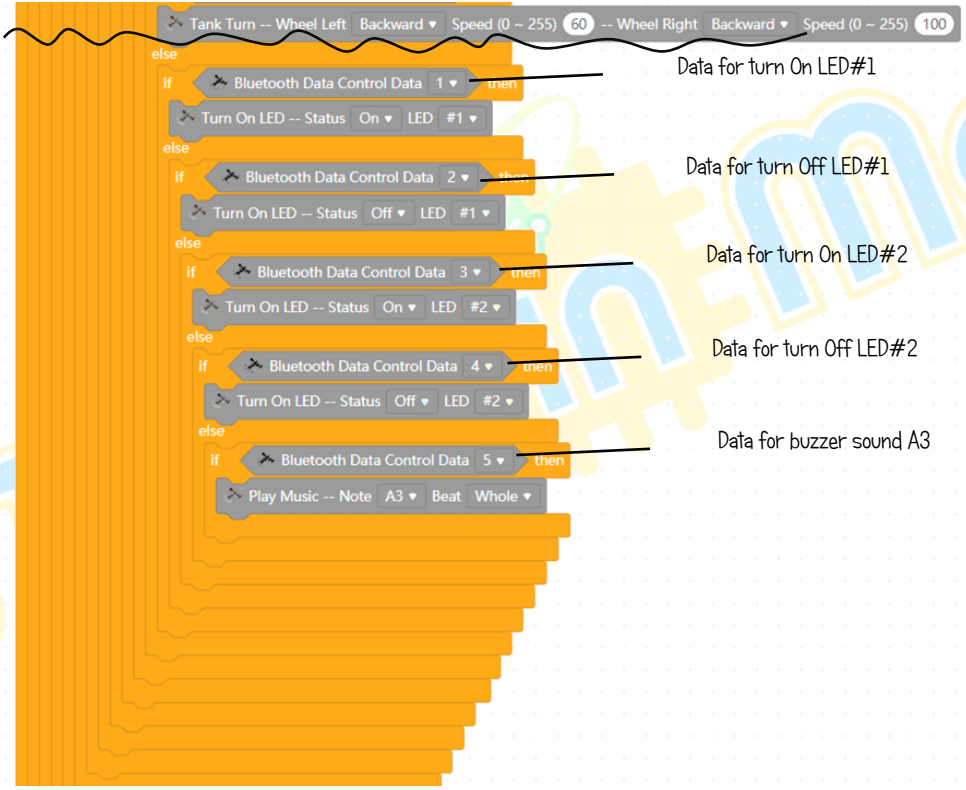
Step 4

For the next block. combine 6 blocks *Bluetooth Data Control Data (T. S. E. Q. C. Z)* with 6 blocks *if* and under the block *then* combine with 2 blocks *Play Music (Note-C3. Beat Half)*5 and 4 blocks *Tank Turn* to get forward and steer right. forward and steer left. backward and steer left. backward and steer right movement.



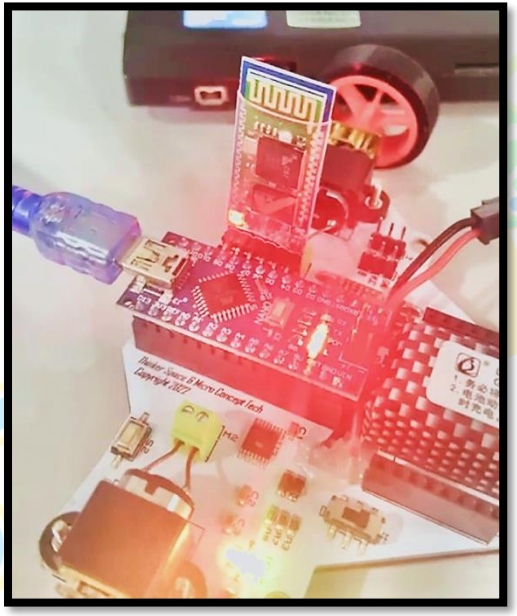
Step 5

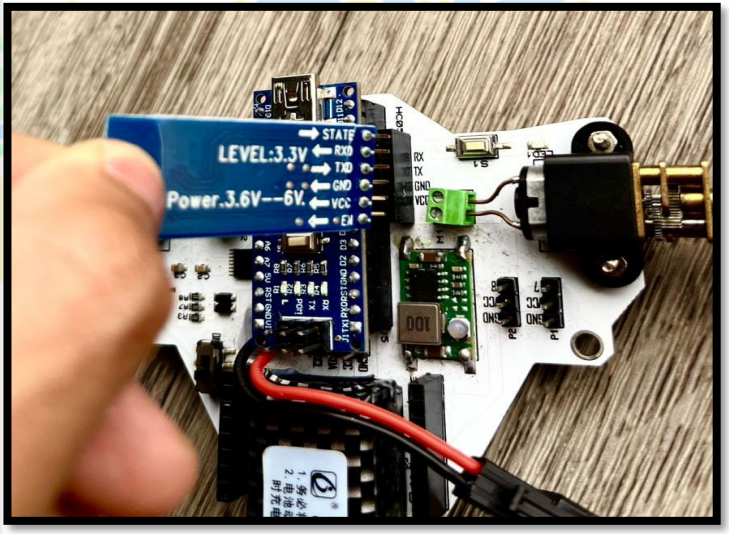
For the next block, combine 5 blocks *Bluetooth Data Control Data (1. 2. 3. 4. 5)* with 5 blocks *if* and under the block *then* combine with 2 blocks *Turn On LED (#1 On. #1 Off. #2 On. #2 Off)* and 1 block *Play Music (Note-A3. Beat Whole)*.



Step 6

After uploading the code. pair the Bluetooth module on the Mikrobotik and match it with the device. Mikrobotik is ready to be controlled by the device. Make sure all Bluetooth pins are connected to the Bluetooth port (RXD-RX. TXD-TX. GND-GND. VCC-VCC)

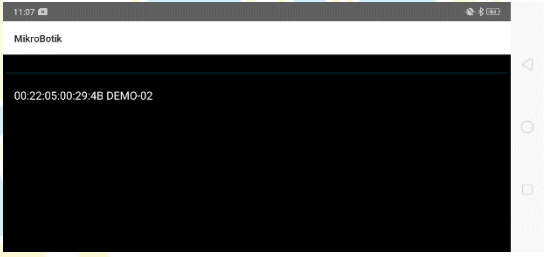




Mikrobotik Mobile Apps

Langkah 1	Download application from: https://www.microconcept.com.my/stem-robotic/download/

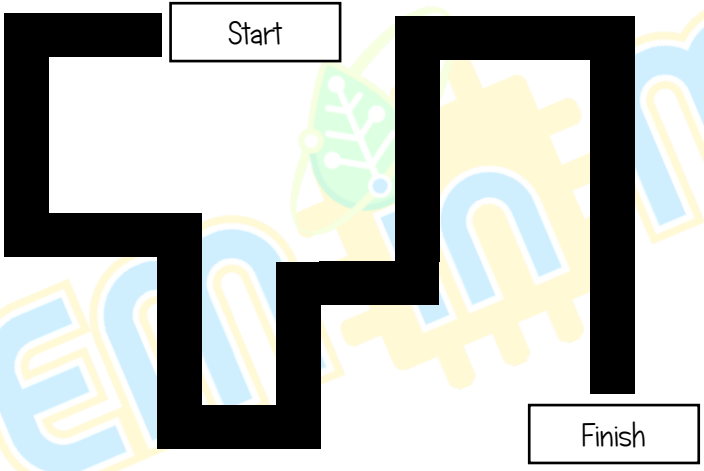
Langkah 2	Open application and click on "Bluetooth connection". Choose based on the number of Bluetooth.



Langkah 3	Make sure "Connected" displayed. Now Mikrobotik can be controlled in free movement.

Challenge!!

In this challenge. you need to make sure that Mikrobotik moves along the path provided by using a device that has been paired with a Bluetooth module on Mikrobotik.



Objective 9: Try Upgrade and Self Program

DEVICE PORT	ARDUINO NANO PIN	PERIPHERALS	ADDITIONAL INFO
ITR1	A6	Sensor Pengesan Garisan – Kiri Luar	ITR8307
ITR2	A3	Sensor Pengesan Garisan – Kiri Dalam	ITR8307
ITR3	A2	Sensor Pengesan Garisan – Tengah	ITR8307
ITR4	A1	Sensor Pengesan Garisan – Kanan Dalam	ITR8307
ITR5	A0	Sensor Pengesan Garisan – Kanan Luar	ITR8307
S1	A7	Suis Pengguna S1	Nilai bacaan < 100
S2	A7	Suis Pengguna S2	Nilai bacaan ≥ 100 & < 400
BUZZER	D2	Pembaz	
LED1	D13	Lampu Indikator L1	
LED2	D12	Lampu Indikator L2	
M1 – AIN1	D5	Motor Kiri – Bridge A Input 1	DRV8833 Dual H-Bridge Motor Driver
M1 – AIN2	D6	Motor Kiri – Bridge A Input 2	DRV8833 Dual H-Bridge Motor Driver
M2 – BIN1	D3	Motor Kanan – Bridge B Input 1	DRV8833 Dual H-Bridge Motor Driver
M2 – BIN2	D9	Motor Kanan – Bridge B Input 2	DRV8833 Dual H-Bridge Motor Driver
P1	D7	Pot Terbuka P1	
P2	D8	Pot Terbuka P2	
BT – TX	D10	Pot Bluetooth TX	
BT – RX	D11	Pot Bluetooth RX	

MCT
MICRO CONCEPT TECH

MERAKYATKAN TEKNOLOGI

- Industry 4WRD
- Pemikiran Kreatif
- Pembudayaan Inovasi
- Kesejahteraan Hidup
- Kelestarian Alam
- Pembelajaran Menyeronokkan

PENGELUAR:

MICRO CONCEPT TECH SDN BHD
1230153-W

No. 5-5, Pusat Dagangan Shah Alam,
Persiaran Damai, Seksyen 11,
40100 Shah Alam, Selangor, Malaysia

@steminme

http://www.microconcept.com.my
steminme@microconcept.com.my