Preamble

The educational content for the "Smart-World add-on for mBot2" is provided in form of "challenges" –different from previous lesson plans and activities. While a first approach to a new digital learning tool, like mBot2, should provide detailed introduction to soft- and hardware used and incorporate a shallow learning curve with staged activities becoming more and more sophisticated, this content is different.

The "Smart-World add-on for mBot2" presents three robotic models that can be used in a smart future in different environments and with different contextual framing: dense urban environments – high tech industrial production plants – smart farming. For sustainable living, we need to address more than one challenge – actually, the Unites Nations (UN) present 17 goals – and make an effort to provide the best solutions to them!

With this set, teachers and students can explore a smart, modern world with support from robots in the areas of food production (farming), industry and dense urban environments. The first row of goals can easily be addressed with the challenges presented here, if the scope is extended, e.g. to medical care & delivery, the second row is achievable as well:

The robots can serve different purposes within this set of sustainability goals, and the challenges presented here will give a short introduction to each topic and give an overview of the tasks that need solving. However, this will be a more abstract discussion, not explaining every little step or giving a detailed lesson plan. The basic structure of the 5E (Engage, Explore, Explain, Elaborate, Evaluate) with an extension to a 6th one, Exchange, is kept. Critical tasks are explained, but the overall structure is much more open-ended and project oriented.

The mBot2 Getting Started Activities (https://education.makeblock.com/resource/) should have been completed and are considered as a prerequisite for these challenges.

makeblock education

Challenge: Monitoring vertical crops

Theme: Smart Farming

Duration: 90 minutes (2x45 min sessions)

Grade(s): 7th and up

Difficulty: medium

Objectives

The purpose of this challenge is for students to:

- Precisely steer the robot to keep a constant distance to a barrier/guide on one side ("maintenance work" or "smart farming")
- Identify coloured objects in different hights on or above the barrier/guide
- Advanced: transmit the position of the objects to another robot.

Overview

This exercise introduces students into the Smart Farming theme, part of the Smart World concept, and presents a challenge for them to solve by using the mBot2 in the Surveying Robot form. First, students should develop an algorithm that allows a precise distance control to one side, examine the optimal distance for detecting colours, identify and report the coloured objects, and lastly develop a solution for estimating the position once the colour is identified.

We suggest the Smart Farming theme, but since this challenge can be used in a different thematic framing, it should be up to the teacher to select the adequate one depending on local curriculum requirements and topics already discussed. Students should be encouraged to discuss their results and extend their ideas and developments further, ideally networking the different robots to cooperatively work together.

Recommended tools

For the teacher:

- Laptop with mBlock 5 installed or browser version [incognito-ide.mblock.cc]
- Example of core program tasks (distance to the wall, report position)

For the students:

- Laptop with mBlock 5 installed or browser version [incognito-ide.mblock.cc]
- mBot2 kit.
- Smart World add-on for mBot2, with the Surveying Robot form assembled and installed on mBot2.
- Tape in the following colours: green, yellow, red, black.
- A straight wall with a smooth texture, where mBot2 can drive close to, and where the colour tape can be stick to, or
- Materials to build a provisional wall, for example cardboard.
- The red and green balls included in the Smart World add-on.

Activities

Introduction [Engage and Explore]

A Smart World

In a smart world, many areas of production, logistics and care are networked and automated to enable high quality living and working standards while taking sustainability at every level into account.

Smart Farming

Good and healthy food is a key component for humanity, but it should also be affordable and grown sustainable – and one "easy" approach is automated local farming (instead of transporting food from remote locations). The students should research and discuss the following questions:

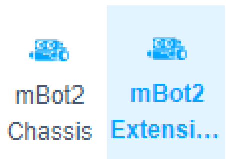
- What do plants need for growing – and how could these processes be monitored?
- Which crops are needed as essential food?
- Can you think of ways to increase the area for growing plants without simply using more land?

- Perform an internet research on these questions and look for automated or robotic farming as well!
- Share and discuss your ideas – which ideas could be simulated with the Surveying Robot?

Since land-use is another important problem in many countries, growing plants has extended into vertical farming. The purpose of this challenge is to use the provided instructions for building a Surveying Robot as a common basis for all student groups to improve the algorithms steering it. The colour sensor provided allows to check colour codes at 4 different hight levels, simulation different degrees of ripeness for 4 layers of vertical farming. Since the sensor needs to be within optimal distance to the object for colour detection, the steering algorithm of the robot must try to maintain that distance...

Servo motors

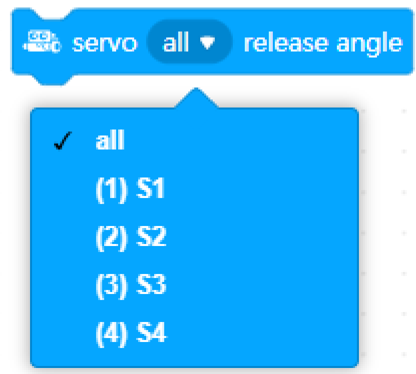
Servo motors are different to motors powering the mBot2 – their purpose is to turn the axis to a given angle and keep that position. They require a 3-pin connector (plus and minus from a power source and a wire that is used to transmit the target value in a coded form). Internally, there is an electronic and mechanical system that checks the current angle to the target angle and corrects any deviations automatically. This is called a closed loop system, because the measurements (of deviation in rotation) are directly used to correct them. Open loop systems take for granted that the target value can be realized, but don't include any internal or external feedback. With these servo motors, the control system is only inside the servo, the current angle is not reported back to the computer. To use a servo motor with the mBot2, you connect it to the mBot2 Shield and use the second category for code blocks:



Use these code blocks to set one target rotation values for one or more servomotors at a time, or even specify them for each motor individually:

Servo Motors rotate between 0 and 180 degrees; however, this can be limited by the mechanical construction they are used with. So, Students should test the values they will use with care!

You can use this command to stop the servo motors from holding (or trying to reach) the target position and switch off:



Instructions [Explain + Elaborate]

The challenge: Monitoring vertical crops

In this challenge, students will work on a solution to complete two mandatory tasks:

1. The robot must maintain a certain distance from the vertical crop while driving.
2. The robot must detect the status of the fruits/vegetables based on their colour.

As an addition, they can then estimate the robot's position when finding a plant ready for harvesting and develop an approach how the different robot types in this add-on can work cooperatively.

The setup

Before students can work on solving the challenge, they need to setup the model of the vertical crop:

1. First, they need to create the vertical wall that will represent the crop. They can do this either by using one of the straight walls from the classroom, which has no obstacles that could block the movement of mBot2, or they can create one using available materials like boxes or cardboard. The height of the wall should be at least 8 cm for the ultrasonic sensor on mBot2 to be able to detect it.
   The test "track" should not be longer than 1 meter.
2. On the vertical crop, students need to setup the fruits/vegetables that mBot2 will scan using the Quad RGB sensor. This can be done in multiple ways. For example, they can use different colour tape on the wall at the height of the Quad RGB sensor (make sure it can be removed without any trace!). Or, if the dimensions of the crop/wall allow it, they can place the colour balls on top it at a height that the Quad RGB sensor can detect them.

Support indications

To develop a solution for the tasks given, Students should follow the Engineering Design Process:

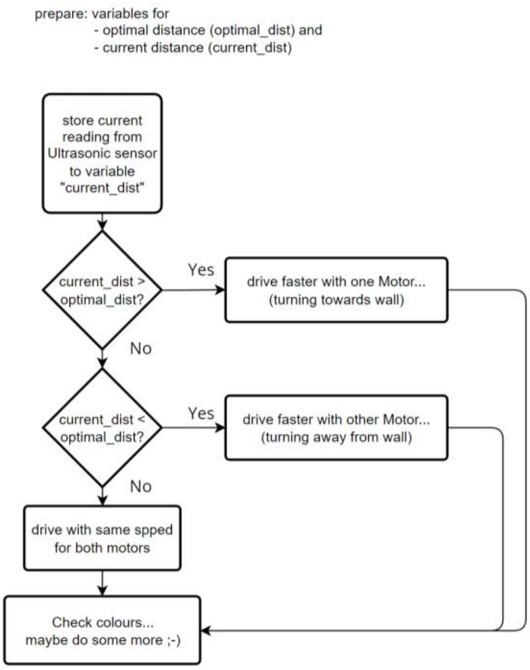
Ask – Imagine – Plan – Create – Experiment – Improve…

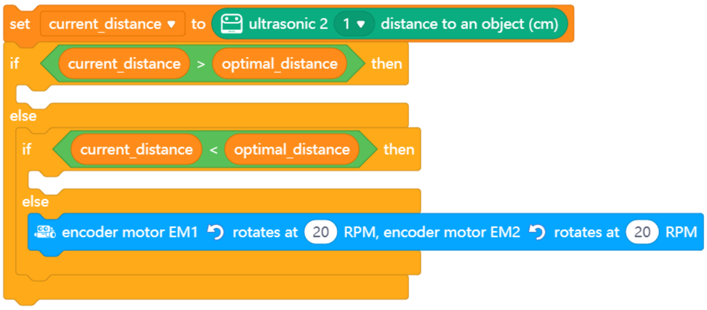
https://www.nasa.gov/audience/foreducators/best/edp.html (follow the suggestions on the website and watch the movie about this design process, then elaborate this for the tasks given).

Suggestions for the students to consider:

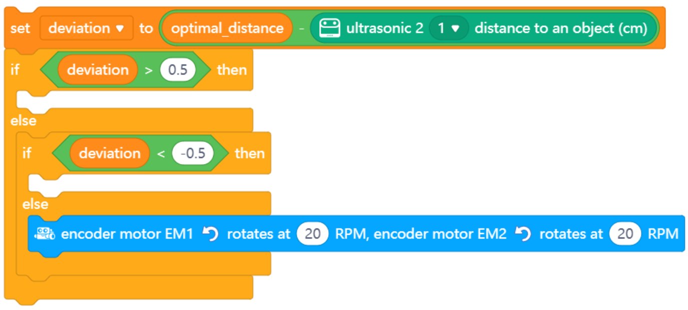
- ✓ Set the servomotor to a position that allows the detection on one side, e.g. right.
- ✓ Find the optimal distance for reliable colour detection – we need this as a variable in the program code.
- ✓ Keep the optimal distance either by conditional statements (equal to distance, grater or smaller) or control the steering proportional.

Example for conditional statement (only part of the code):





In these statements, the robot would only drive straight on, if optimal and measured distance are identical… and the ultrasonic sensor measures with an accuracy of 0.1 cm. How likely is an identical value of optimal and measured distance? This could be improved by using the deviation from the optimal position and including this into the conditional statements… allowing a range of values for driving straight on:



The range should be determined experimentally as well as the driving speeds.

An even more accurate solution would be to calculate the motor speeds proportionally depending on the deviation of the distance from the optimal one:
At optimal distance the robot cruises with a constant speed for both motors. The greater the difference from the optimal distance, the greater the difference of speed of both motors must be set to compensate this.

Ask Students to formulate this in more detail for left or right following…

- Set a basic speed for driving at optimal distance, e.-g. 20 RPM, save it in a variable

- Calculate the deviation from the ideal distance, save it to a variable
  Multiply the deviation with a factor (stored in a variable) and add it to the basic speed for one motor, subtract it from the basic speed for the other motor.
  Tune the parameter (factor): if robot is oscillating, reduce it – if robot is reacting too slow, increase it.

These three different approaches are suggestions in case the students are unsure how to proceed. In the run of the project, the different groups may develop solutions with different precision and complexity of coding. Evaluating these experiments and developments will take place in the next step.

Evaluate [+ Exchange]

It is now time for a brief reflection. The students should think on their own and discuss with the group the following questions:

- What do you think turned out well?
- What could be better?
- Which parts of the challenge did you find easy, and which did you find more difficult?
- What would you like more explanation about?
- Who could help you with that?
- Could you think of improvements regarding the program code or the hardware?

Extending the project

In a smart world, it is important that the different systems are interconnected and interrelated... Monitoring crops is one aspect of a bigger scenario. The students should try to extend this challenge and involve the other robots that can be build from the Smart World add-on:

- How can you collaborate with the robotic arm to harvest the fruits/vegetables?
- Can you transport them to a storage location?
- How can the surveying robot find out its position to inform one mBot2 with the robotic arm?
  (Tip: Reset the counter for each encoder motor and use the mean of both motor rotations to estimate the driven distance to a fruit ready for harvesting...)
- Are there any other modification you could develop – maybe using 3D-printing or Laser cutting?

Challenge: Transporting goods around the farm

Theme: Smart Farming

Grade(s): 7th and up

Duration: 90 minutes (2x45 min sessions)

Difficulty: medium

Objectives

The purpose of this challenges is for students to:

- Using the Quad RGB sensor, the robot should identify different locations in a work area (a farm).
- Program the robot to transport objects between different locations in the work area.

Overview

This exercise introduces students to the Smart Farming theme, part of the Smart World concept, and presents a challenge for them to solve by using the mBot2 in the Robotic Carrier form. Students should develop an algorithm for the mBot2 to follow a self-designed network of tracks, identify different colour markings on the ground that correspond to different positions the map, and transport items to these different locations.

We suggest the Smart Farming theme, but since this challenge can be used in a different thematic framing, it should be up to the teacher to select the adequate one depending on local curriculum requirements and topics already discussed. Students should be encouraged to discuss their results and extend their ideas and developments further, ideally networking the different robots to cooperatively work together.

Recommended tools

For the teacher:

- Laptop with mBlock 5 installed or browser version [https://incognito-ide.mblock.cc/]

For the students:

- Laptop with mBlock 5 installed or browser version [incognito-ide.mblock.cc]
- mBot2 kit.
- Smart World add-on for mBot2, with the Robotic Carrier form assembled and installed on mBot2.
- Materials to create a line-following map (for example, black insulation tape, A0 paper sheets, color tape or stickers, etc)
- The red and green balls included in the Smart World add-on.
- Other simple materials to recreate a farm setting (cardboard, farm images, colors, sticks, scissors, glue, etc)

Activities

Introduction [Engage and Explore]

A Smart World

In a smart world, many areas of production, logistics and care are networked and automated to enable high quality living and working standards while taking sustainability at every level into account.

Smart Farming

Good and healthy food is a key component for humanity, but it should also be affordable and grown sustainable – and one easy approach is automated farming. The students should research and discuss the following questions:

- Think about the different farms for rearing animals (or for growing crops if you prefer) how are they designed?
- What are the different areas in the farm?
- How are the spaces organized and what is the workflow between the different areas?
- Are there any parts where the workflow becomes more difficult?
- Do some internet research on these questions and look for automated or robotic farming as well!
- Share and discuss your ideas – which ideas could be simulated with the Robotic Carrier?

In order to alleviate labour shortage and improve production efficiency in farms around the world, support has come in the shape of automation. The purpose of this challenge is to use the provided instructions for building a Robotic Carrier as a common basis for all student groups to improve the algorithms driving it. The Quad RGB sensor included allows to identify different color marks on the floor while following a path. The platform attached to mBot2 allows to bear a load and can be tilted with the help of the servo to release the payload. Students must then design their own layout of a farm based on their research and create a map where the robot can navigate between different locations to transport the necessary goods for the operation of the farm.

Instructions [Explain + Elaborate]

The challenge: transporting goods around the farm

In this challenge, students will work on a solution to complete two mandatory tasks:

1. Using the Quad RGB sensor, the mBot2 should identify different color markings on the ground that correspond to different locations on the layout of a farm.
2. The robot needs to transport and deliver certain goods to specific locations in the farm by following tracks on the ground.

As an addition, students should determine and develop the method to provide the robot with the instructions on where to go. Perhaps with voice commands (internet connection is required), using the onboard buttons, via messages sent through an interactive control panel on the stage in the mBlock coding platform, or even via wireless communication from other robots (for example one delivering goods).

The setup

Before students can work on solving the challenge, they need to setup the model of a farm:

1. Based on their research, students should choose a type of farm to recreate, for example a dairy farm. Students should identify the basic layout and processes of the farm.
2. With this information, students design a **simple model** and create a layout where the different locations relate to paths that the mBot2 can follow and marked with different colours that the robot can identify.

   Students should **begin with a simple design of the farm**, so they are able to test their solutions quickly. As they gain expertise and confidence, they can move to more complex layouts.

makeblock education

The dimensions of the map depend on the space available, and the design of the driving paths and color marks should be developed and tested.

Support indications

To develop a solution for the tasks given, Students should follow the Engineering Design Process:

Ask – Imagine – Plan – Create – Experiment – Improve…

https://www.nasa.gov/audience/foreducators/best/edp.html (follow the suggestions on the website and watch the movie about this design process, then elaborate this for the tasks given).

Suggestions for the students to consider:

- ✓ Students can consult the mBot2 Getting Started Activities to understand the principles of line following and color identification with mBot2.
- ✓ For the design of the farm, students can make use of the Map Design Guidelines (see appendix).
- ✓ **Before getting into complex designs of layouts for the farm, it is recommended to start simple.** For example, students can start with a straight line that the mBot2 can follow, with 3 or 4 colour marks at certain distances representing a few locations in the farm, where the robot can turn and either retrieve or deliver the load it is meant to transport. Once the principles have been tested and evaluated, the design and process of developing solutions can re iterated to more complex forms.

Evaluate [+ Exchange]

It is now time for a brief reflection. The students should think on their own and discuss with the group the following questions:

- What do you think turned out well?
- What could be better?
- Which parts of the challenge did you find easy, and which did you find more difficult?
- What would you like more explanation about?
- Who could help you with that?
- Could you think of improvements regarding the program code or the hardware?

Extending the project

In a smart world, it is important that the different systems are interconnected and interrelated… Transporting goods around the farm is one aspect of a bigger scenario. The students should try to extend this challenge and involve the other robots that can be built from the Smart World add-on:

- How can you collaborate with the robotic arm to receive and deliver the goods?
- How can the robotic carrier inform its position to the robotic arm or vice versa?
- How can the robotic carrier be “called” to go to a location where it is needed?
- Are there any other modification/improvement you could develop – maybe using 3D-printing or Laser cutting?

Challenge:
Processing waste

Theme: Sustainability in the City **Grade(s):** 7th and up

Duration: 90 minutes (2x45 min sessions) **Difficulty: medium**

Objectives

The purpose of this challenge is for students to:

- Create a program that distinguishes different kinds of objects based on image recognition using the Teachable Machine extension in mBlock and a webcam.
- To program the Robotic Arm to manipulate and transport the identified objects to specific locations within a work area (a waste processing plant).

Overview

This exercise introduces students into the Sustainability in the City theme, part of the Smart World concept, and presents a challenge for them to solve by using the mBot2 in the Robotic Arm form. First, students should create an image recognition program using the mBlock software, which they can train (simply by using webcam images) and which allows them to identify different types of objects (that the robot is capable of gripping and lifting). Then, the Robotic arm should be programmed to transport and deliver them to specific locations after they have been identified by the image recognition program.
For an introduction to simple, block-based image processing with mBlock5 software, please refer to Lesson 9 of the getting started activities.

We suggest the Sustainability in the City theme, but since this challenge can be used in a different thematic framing, it should be up to the teacher to select the adequate one depending on local curriculum requirements and topics already discussed. Students should be encouraged to discuss their results and extend their ideas and developments further, ideally networking the different robots to cooperatively work together.

Recommended tools

For the teacher:

- Laptop with mBlock 5 installed or browser version [incognito-ide.mblock.cc]

For the students:

- Laptop with mBlock 5 installed or browser version [incognito-ide.mblock.cc]
- mBot2 kit.
- Smart World add-on for mBot2, with the Robotic Arm form assembled and installed on mBot2.
- Materials to create a line-following map (for example, black insulation tape, A0 paper sheets, color tape or stickers, etc)
- The red and green balls included in the Smart World add-on.
- Other simple materials to recreate a waste processing plant setting (cardboard, images, colors, sticks, scissors, glue, etc)
- Webcam with USB-connection

Activities

Introduction [Engage and Explore]

A Smart World

In a smart world, many areas of production, logistics and care are networked and automated to enable high quality living and working standards while taking sustainability at every level into account.

Sustainability in the City

One of the outcomes of human activities is the generation of waste. Cities in particular see a high concentration of waste coming from households and from the tertiary sector or service sector. Because of this, cities now and in the future need to figure out effective ways to dispose of, and even take advantage from, these materials. The students should research and discuss the following questions:

- How is waste processed in your area?
- What are the most common materials that constitute waste in your area?
- Can you think of ways to improve the waste management process?
- Perform an internet research on these questions and look for automated or robotic waste management as well!
- Share and discuss your ideas – which ideas could be simulated with the Robotic Arm?

Because of resource scarcity and simply dumping waste in landfills can result in environmental hazards, recycling some materials has become one of the best solutions. The purpose of this challenge is to use the provided instructions for building a Robotic Arm as a common basis for all student groups to create an algorithm to identify and collect residues in a simulated waste processing plant. By using the Teachable Machine extension in the mBlock coding software in combination with the webcam, students can create a visual recognition system to separate different types of waste. The robotic arm can be used to manipulate the materials and place them on specific locations or to collaborate with other robots for transportation. Lastly, the Quad RGB sensor can help the robot to identify the right locations where it needs to transport each type of waste for further processing, as well as to find a path to these locations.

Instructions [Explain + Elaborate]

The challenge: Processing waste

In this challenge, students will work on a solution to complete two mandatory tasks:

3. Identify different types of waste materials by using a webcam and an image recognition program created in the mBlock platform.
4. Use the robot to bring the materials to the appropriate location for further processing. In this task, the computer should communicate the information about the waste with the mBot2 for the correct disposal.

The setup

Before students can work on solving the challenge, they need to setup the model of a waste processing plant:

1. The Teachable Machine extension in mBlock will be used to identify the waste, and it requires a webcam or camera to be connected to the computer while the extension is being used. For this reason, students need to set a "waste separation station", where the webcam is aiming at a fixed background and where the disposed materials will be placed for analysis. This same setting will be used to train the image recognition model. More details on this in the following sub-section.
2. A layout of the plant is needed to bring the materials from the waste separation station to new locations where each particular material will be processed. A simple way to do this is with a straight line which has the waste separation station in one end and 3 or 4 colour marks towards the opposite end, which serve to indicate the stops where the robot should drive to with the waste after separation.

Support indications

To develop a solution for the tasks given, students should follow the Engineering Design Process:

Ask – Imagine – Plan – Create – Experiment – Improve…

https://www.nasa.gov/audience/foreducators/best/edp.html (follow the suggestions on the website and watch the movie about this design process, then elaborate this for the tasks given).

Suggestions for the students to consider:

- ✓ Students can consult the mBot2 Getting Started Activities to understand the principles of the Teachable Machine extension, line following, and color identification with mBot2.
- ✓ Professional image recognition systems require a great amount of data (images) and processing power to deliver very advanced and accurate results. Since this challenge involves a **simplified version of such system, the data base** (images used to train the model) **and the materials to be recognized must be simple. Create only 3 or 4 different categories of waste materials and make sure that they are easy to differentiate from one another.**

  Each category should contain 15-35 webcam images from different perspectives, until the material is reliably recognized.
- ✓ For the design of the waste processing plant, students can make use of the Map Design Guidelines (see appendix).

makeblock education

- ✓ **Before getting into complex designs of layouts for the waste processing plant, it is recommended to start simple.** For example, students can start with a straight line that the mBot2 can follow, with a start position and 3 or 4 colour marks at certain distances representing some locations in the plant, where the identified waste categories will be processed.

Evaluate [+ Exchange]

It is now time for a brief reflection. The students should think on their own and discuss with the group the following questions:

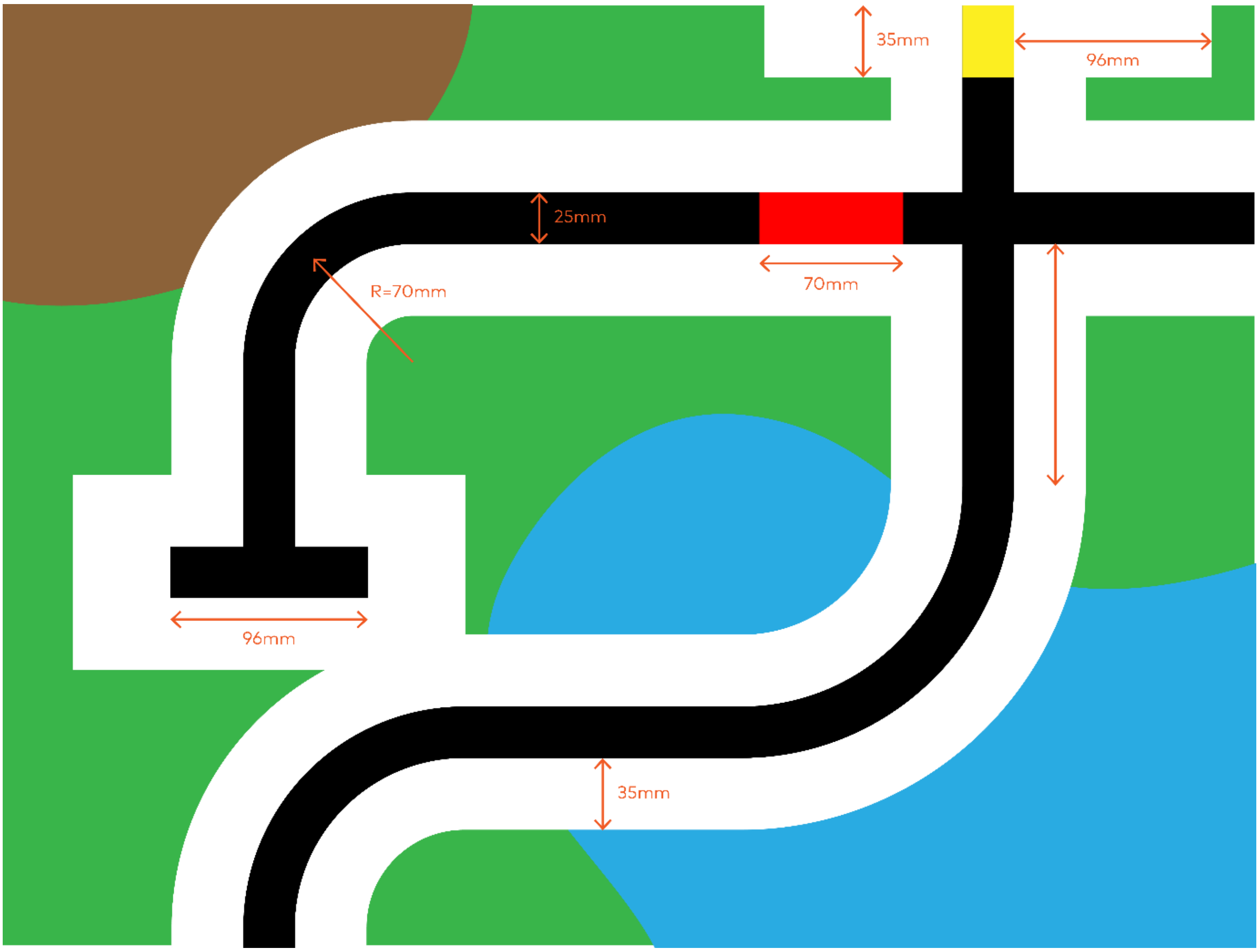
- What do you think turned out well?
- What could be better?
- Which parts of the challenge did you find easy, and which did you find more difficult?
- What would you like more explanation about?
- Who could help you with that?
- Could you think of improvements regarding the program code or the hardware?

Extending the project

In a smart world, it is important that the different systems are interconnected and interrelated... Processing waste is one aspect of a bigger scenario. The students should try to extend this challenge and involve the other robots that can be built from the Smart World add-on:

- How can you collaborate with the Robotic Carrier to transport the waste?
- How can the Robotic Carrier communicate with the Robotic Arm after it has successfully delivered a material to a destination?
- Are there any other modifications students could develop – maybe using 3D-printing or Laser cutting?

Map design guidelines



Overview of basic guidelines.

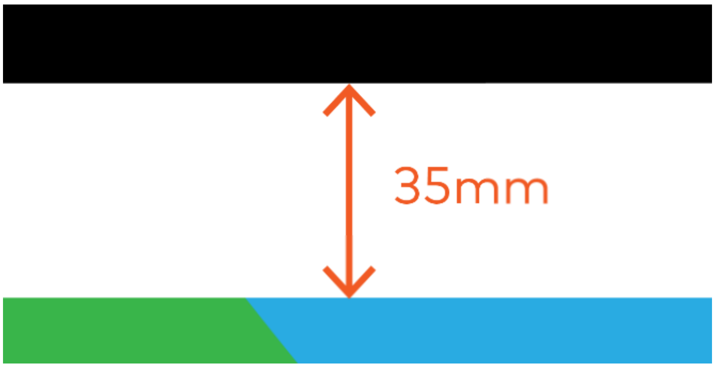
★ Black/white track on white/black background

1. Track width

The line is recommended to be 25mm wide, so it can better cover the inner sensor probes L1 and R1 of the Quad RGB sensor.

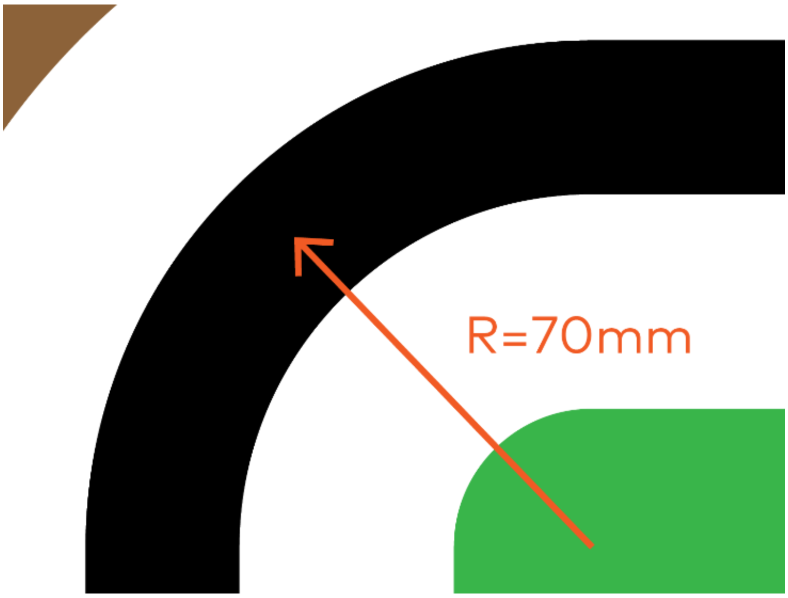
2. Decorative backgrounds

It is best to keep a solid colour background area of no less than 35mm on both sides of the line, ideally more than 45 mm, to ensure that the Quad RGB sensor doesn't recognize the decorative background of the map.



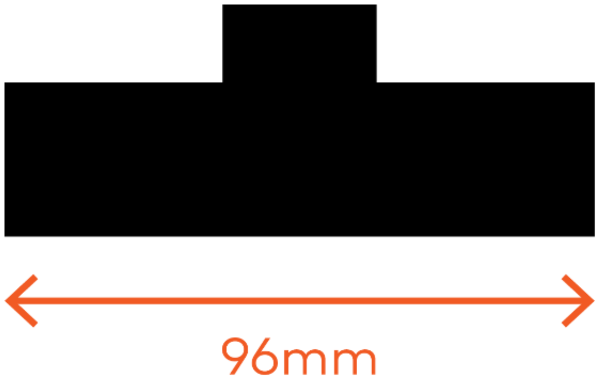
3. Curve on the track

If there is a curve on the line, the corner radius of the line is recommended to be no less than 70mm from the centre of the line, so the bend is not too sharp and the line patrolling operation is stable.



4. Crossing, T-intersections

If there is a crossing or T-intersection on the line, the width of the intersecting line should be no less than 96mm, to ensure that all 4 sensor probes can detect the intersection.



5. Distance after/before curves

Before any intersection or crossing there must be at least 80mm of straight line (from all sides) to ensure the robot is aligned with the track.

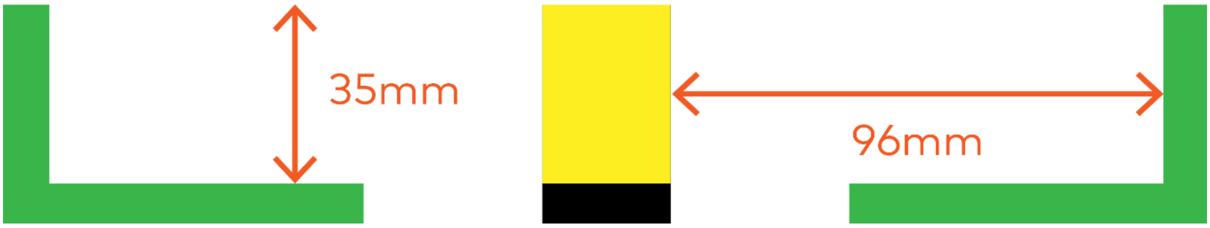
★ Black/white with different colour segments on white/black background

6. Colour segment length

If the line incorporates colour segments, these should be no less than 70mm in length, to ensure that the sensor can detect the colour segment even when moving at high speed.

7. Calibration area

The map should include a calibration segment for line detection. If the line includes colour segments, the one with the brightest colour should be included. Otherwise, the track colour (usually black) should be used. The solid colour background area (usually white) should be at least 96mm wide on both sides of the line. The length of the segment should be at least 35mm. These measurements will ensure that there is enough space for swiping the sensor over this area during the calibration.



Black/white with different colour markers next to the track

8. Colour marker width

A map can also incorporate colour markers next to the track on straight sections. The minimum width of these markers is the same as the track: 25mm. This width will allow the external sensors of the Quad RGB sensor (L2, R2) to detect the marker. The markers can be used on one or both sides of the track, depending on its purpose. For instance, they can help to indicate the robot's position on a map.

9. Turning on a colour marker

In order to turn left or right on a colour marker, the rear wheels of mBot2 need to be centred on the marker. The travel distance from the point where the colour marker is detected until the rear wheels are centred on them is approximately 90mm.

10. Mixing colour markers on both sides of the track

If the mBot2 always drives the same track on a single direction, different colour combinations of pairs of markers can allow you to color-code a bigger number of positions than if using single colours. If the robot drives a track on both directions, mixed colours of pairs of markers can be used to inform the robot about its direction apart from its position:

Position	Direction	L2	R2
Position 1	Driving from left to right	Green	Red
Position 2	Driving from right to left	Red	Green