GENIBOT Technical Guide

Unplugged and Physical Computing Reference

GB1-TRG0805P2 August 5, 2024

GENIBOT Technical Guide

Unplugged and Physical Computing Reference

Technical Support and Product Information

Address: G749, 815, Pangyo 2nd Techno Valley Smart Industrial Complex, Daewangpangyo-ro, Sujeong-gu, Seongnam-si, Gyeonggi-do, 13449, Korea
www.genirobot.com

GENIBOT is product and trade name of Dabida (GeniRobot) Co., Ltd. Other trademarks or trade names of products and company names mentioned herein are: Microsoft Windows and Make Code, Google Android, Apple iOS, BBC micro:bit Educational Foundation, Scratch Foundation and Python Software Foundation.

© 2024 Dabida (GENIROBOT) Co., Ltd. All rights reserved.

IMPORTANT

Warnings and Cautions

Statements on symbols, terms and conventions you may see in this guide give the information that you should consider installing and operating the robot GENIBOT.

The robot GENIBOT is designed only for a level of reliability suitable for educational IoT smart devices. It may not be used on or in connection with any equipment, material or system that could reasonably be expected to cause serious personal injury.

In any application this hardware can be impaired by adverse factors, electrical power supply, computer hardware and software malfunctions, unexpected uses or errors on users. If the robot GENIBOT is used and operated with non-certified devices or in any manner not specified by the manufacturer, the protective features of the robot GENIBOT may be damaged and it is not certified for use in hazardous locations.

The materials of the robot GENIBOT are provided “as is,” and is subject to being changed, without prior notice to users.

Dabida (GeniRobot) Co. Ltd. shall not be liable for events, errors, incidental or consequential damages in connection with improper use, performance of this document or of any information described herein.

Precautions for use
Pay special attention to the requirements and operate the robot GENIBOT only as described in the “Warning” and “Caution” conditions.
Using and operating the robot GENIBOT under conditions other than those specified above may result in danger and injury, as well as damage to the robot GENIBOT and external devices connected to the robot.

AT A GLANCE

Technical Features

Specifications

Processor (CPU):	ARM Cortex M4 32bit processor, 64MHz, RAM 64KiB, Flash 512KiB
Power Source	DC Power limit (Rated 5V DC ⎓ 300mA) DC Power adaptor (5V DC ⎓ 1A)
Power Terminal	VDD 5V USB C-type connector
Battery	Li-Polymer 3.7V 1000mAh, overcurrent and overvoltage protection
Connectivity	Bluetooth LE 5 (5.0/5.1) PHY 2Mbps multiprotocol and ISM band 2.4GHz radio protocol Distance < 30m, Signal strength RSSI > -90dBm
Light Sensor and Green LED	Four ambient light sensors and four green LEDs for line following and Grid detection
Acceleration Sensor	Acceleration (State of the-Art 3-axis, range -2g to +2g) Not applicable to GENIBOT Plus version (Model D)
RGB Color LED	RGB color LED (4) for making and changing Color in HSV or RGB color space
Red LED	Red LED on external terminal port for charging battery
OID Module	Optical image detection sensor and decoder for recognizing unplugged coding card or reading position in Cartesian coordinate with 16bit dot code pattern detection.
Speaker	Impedance 8Ω ± 15% 1.0V 1kHz Maximum power 1.0W Effective frequency band 200 to 4kHz and flat response center frequency 2kHz

Audio Amplifier	PCM Class D audio amplifier
Memory	SLC NAND flash memory 128MiB (134MB)
Motor	Stepper (50:1 Geared, 360DPS/1000SPS)
External AI and DIO Port	Standard drive output voltage will be from minimum (VDD - 0.3V) to maximum (VDD + 0.3V) where VDD is +3.3V. When using VDD from +3.0V to 3.3V, the absolute maximum input voltage must be less than (VDD + 0.3V). If VDD is greater than 3.3V (VDD + 0.3V), ESD protection will operate, but nothing is guaranteed about its function and operation. Maximum sink current is 6mA with standard drive at (VDD - 0.4V), and 15mA with high drive at (VDD - 0.4V) where VDD is +3.3V.
External AI and DIO Port Power Source	Standard drive output voltage will be from minimum (VDD - 0.3V) to maximum (VDD + 0.3V) where VDD is +3.3V. Maximum output current is 300mA, which can provide sliding current to drive Geek Servo 270 (Rated DC current = 200mA at +3.0V) or Arduino BBC micro:servo 180 motor (Rated DC current is 75mA at +3.0V)

⚠

Battery Consumptions
Battery life is approximately 1.5 hours when the motor is running continuously. If the robot is used intermittently during class activities, it can be used for more than a few days. If you rarely use the robot, the battery may last more than a month after a full charge.
When charging the robot GENIBOT, you must use a certified power adapter such as KC, CE, or FCC that has been verified for safety.

Button and Color LED

Power Button	To turn on the robot, press the power button. When the robot starts, it says “Hello, I’m GENIBOT”. To turn off the robot, press and hold the power button for more than two seconds. When the robot shuts down, it says “Bye, bye see you later”. However, in certain activity situations, the robot may restart without a power-off greeting. To start or stop the unplugged coding when coding data set is stored in memory, press the power button. To advertise the robot’s Bluetooth discovery name while the Star robot or smart device is scanning GENIBOT, press the power button. To connect a star robot or peer robot, just press the power button once. To set up a BBC micro:bit radio group, press the power button and then tap two number cards.
Color LED	Color will be changed when GENIBOT is connected to a star robot or smart device but also color will be indicated to matching card color when it is detected by GENIBOT OID image detection sensor Color can be changed in HSV color coordinate as: Red, Green, Blue, Cyan, Magenta, Yellow, Violet, Orange, Spring Green, Light Pink and White by using GENIBOT Android and iOS application, Scratch or Python. Moreover, with Scratch and Python programming, you can change the left, right, front, and back colors independently. Hue in HSV (Hue, Saturation and Value) color space can be changed from 0 to 360 in degrees.

GETTING STARTED

Before Start

The GENIBOT application can be used to perform a variety of coding activities and offers expanded functionality for those learning and teaching STEAM (Science, Technology, Engineering, Arts, and Mathematics).

The GENIBOT app for Google Android or Apple iOS is easy to use and offers a variety of coding activities including Action Bar, Controller, Card Coding, Line Tracing, Button Coding, Music Coding, Drawing, Math Coding, and Voice Coding using AI voice recognition. and operational functions.

To start coding with the GENIBOT application for Google Android or Apple iOS, first press a button on the robot to connect it to your smart device, then select an activity such as Remote Control, Card Coding, Music Coding or Voice Coding.

For Unplugged coding without connecting the robot to a smart device, Physical Computing or EPL (Educational Programming Language) activities, refer to this document or others provided separately such as GENIBOT Arduino Reference, BBC micro:bit-GENIBOT Radio Programming Guide and etc.

In the classroom
Turn off the robot when not in use. Fully charge the robot to carry out a variety of coding activities over long periods of time in the classroom.

CARD CODING

Unplugged Coding Activity

Unplugged coding

Robot programming is possible without a computer, and you can create motion or music using unplugged coding cards, including sounds, lights, motions (move forward, move backward, turn left, turn right, pause), functions, and music notes

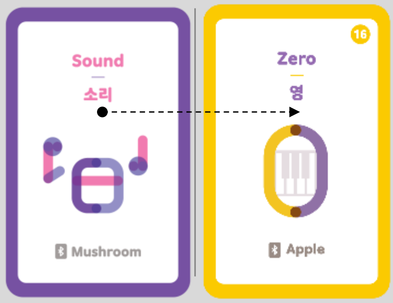
Change Language

To change the preferred language, tap the Speaking card and then the number 2 card for another different language or number 1 card for the default language in English. When you tap the number card to configure a language, the robot will store it to non-volatile flash memory and then reboots.

When you tap the speaking card and number card 1 or 2 to set the on-start language, the robot will write it to non-volatile flash memory and then reboots. The robot will always speak in the set language unless you need to change it again.

To configure language
Tap the Speaking card, then tap the number 1 or 2 card.

To mute speaker
To mute speaker, tap the Speaking and number 0 card.
After turn off it, the robot will not speak a card name when you tap it.

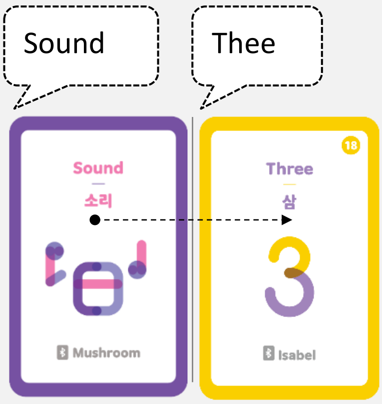


Change Speaker Volume

The speaker volume can be changed from 1 to 9. When you turn on the robot, the speaker produces a tone at the default maximum volume. Use the Plus card to reset the level to the default volume. Use the Rest card to allow the changed volume even after the reboot is rebooted.

To change volume

To set a volume, tap the Sound card and number card from 0 (mute) to 9, or Plus card (reset to default maximum volume)

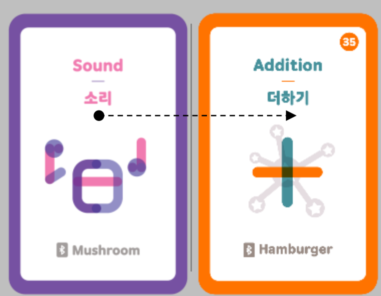


To write volume to flash

To write a volume to non-volatile flash memory, tap the number card, then tap the Rest card.

To set maximum volume

Tap the Sound card, then tap the Plus card.

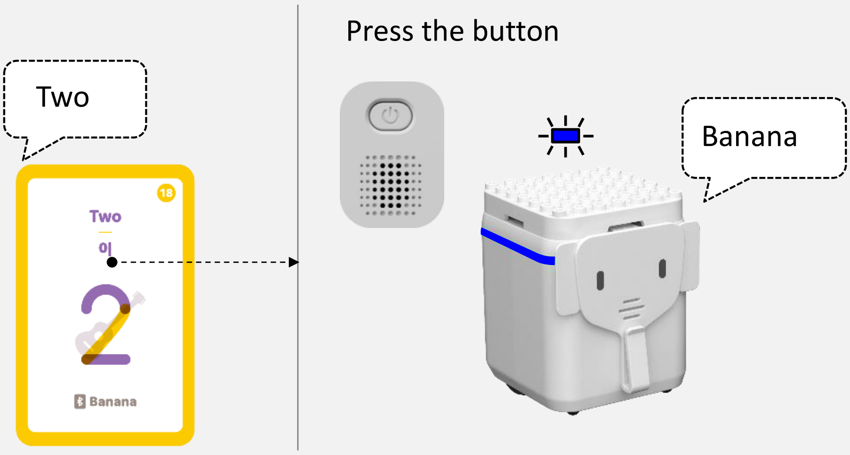


Change Bluetooth Name

When you press the power button, the robot will speak its Bluetooth name, blink blue every 0.5s, and wait until 30 seconds to connect to the Bluetooth host device. To change the Bluetooth name, select and tap the card with the Bluetooth name written on it, then press the button to start Bluetooth. To store Bluetooth name to the robot, tap the rest card while it is blinking Blue.

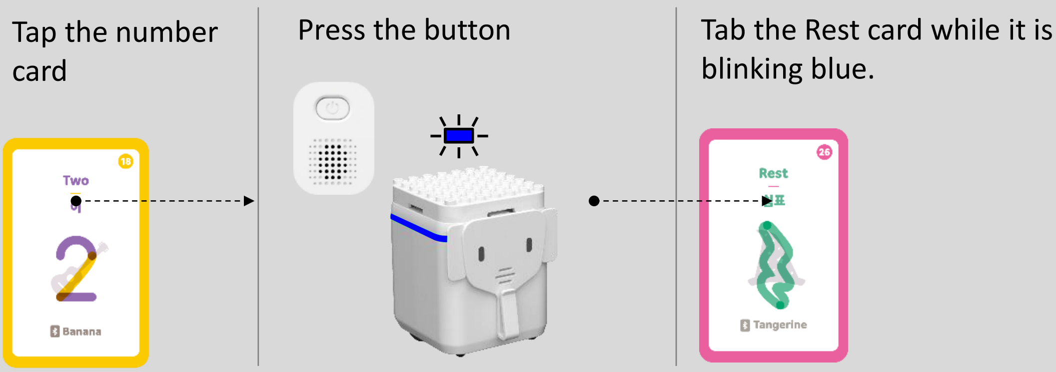
To change Bluetooth name

To change Bluetooth name, choose a card that you can view Bluetooth logo and name at the bottom of the card, for example "Banana", and then press the power button of the robot to start Bluetooth.



To write Bluetooth name to the robot

Tap the rest card while it is blinking blue and the robot will reboot.

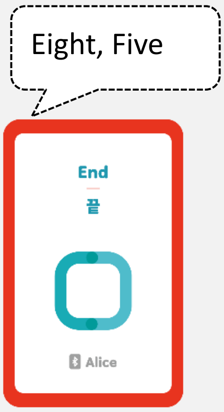


Say Battery Percentage

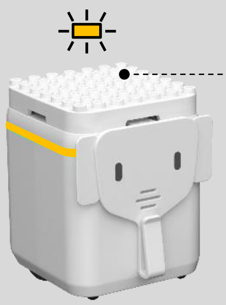
When you are not programming the robot, you can read the battery percentage by tapping the end card. The robot speaks its battery percentage, so if the percentage value is very low, the battery needs to be charged to run the robot.

To read battery percentage

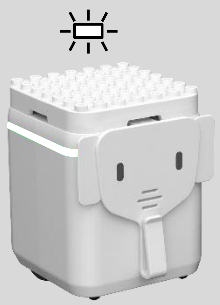
Tap the End card while you're not coding. The robot speaks the battery percentage value. For example, if the robot says the two numbers "8" and "5", it means 85%.



Yellow warning when battery level is low.



A white warning appears after 10 minutes of inactivity.



Sleep and low battery warnings

If the battery is low, an error will occur when the motor operates.

If the battery percentage value is very low, it will flash yellow 4 times and the robot will turn off.

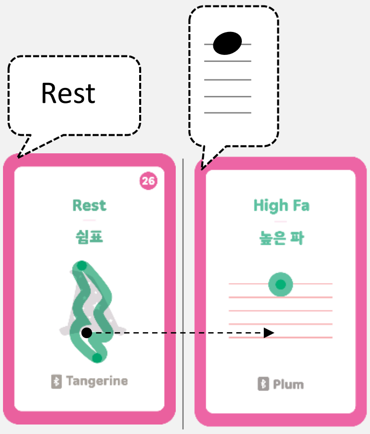
If the robot has no activity for 10 minutes, it will flash white to save power and then it will turn off.

Say OID Code

Tap the Rest card, and then High Fa card in that order. Then, the next time you tap any card, the robot will say the OID code of the card you tapped. If you send the OID code to the robot using a programming language such as Scratch or Python, the robot can say the card name corresponding to the OID code, moreover, functions and operations defined in coding cards with OID code values can be executed.

To read OID code

Tap the High Fa, Rest, Hagh Fa cards in that order. When you tap any card, the robot says a four-digit number.

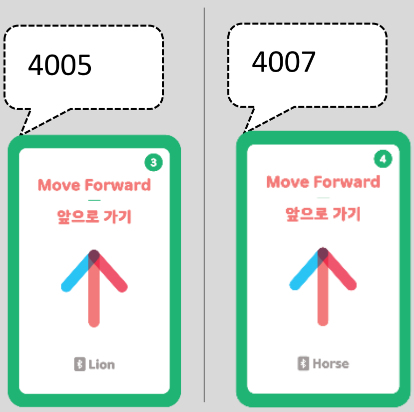


When you tap the High Fa card, the robot plays the piano note Fa (F5).

OID code according to Bluetooth name

Even with the same coding card, the Bluetooth name is different, and the OID code is also different accordingly.

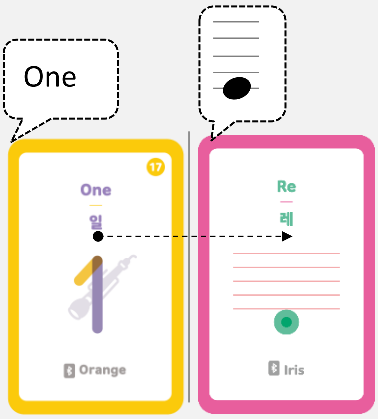
Refer to the appendix Audio Code Reference.



Change Music Instrument

To change the instrument from piano to flute or strings, tap number 1 or 2, then tap the music card. Afterwards, tap any music card again to hear the sound of the instrument you set. Music cards are available from Ti (B3) to Fa (F5). To change the instrument back to piano, tap the number 1 card, then tap any music card. When you turn the power off and on, the piano becomes the default instrument as usual.

Piano, flute or string configuration
If there is a music card in the coding data, it will be played with the instrument set before coding begins. If you want to change the instrument in the coding data, place the number cards 0 (piano), 1 (flute), and 2 (string) in front of the music card.



Half note # or ♭
If you want to add the half notes of the music card to the music note data, program with Scratch or Python.

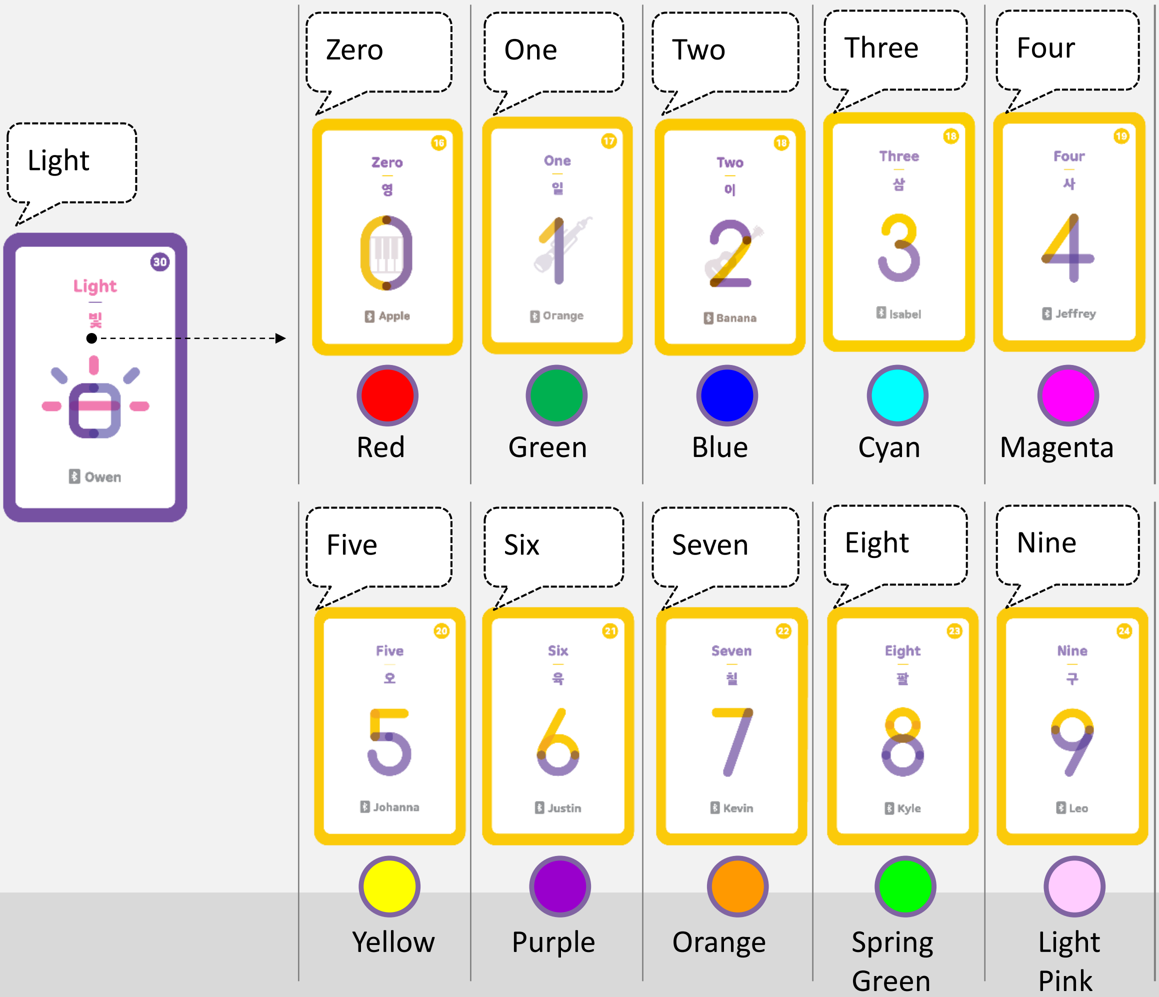
Piano Key	Music Note	
37	A3	La
38	A#3	
39	B3	Ti (Si)
40	C4 Middle C	Do
41	C♯4/D♭4	
42	D4	Re
43	D♯4/E♭4	
44	E4	Mi
45	F4	Fa
46	F♯4/G♭4	
47	G4	Sol
48	G♯4/A♭4	
49	A4	La
50	A♯4/B♭4	
51	B4	Ti (Si)
52	C5	Do
53	C♯5/D♭5	
54	D5	Re
55	D♯5/E♭5	
56	E5	Mi
57	F5	Fa

Change LED Color

To change the hue of color LEDs on your robot, tap the Light card, then tap the number card. The colors are defined on the number card as follows: Red 0, Green 1, Blue 2, Cyan 3, Magenta 4, Yellow 5, Purple 6, Orange 7, Spring Green 8, Light Pink 9.

Set color using number cards or coding stickers

To change the LED color, tap the Light and number cards 0 to 9, or tap the coding sticker. The hue of color in the HSV color space can be varied from 0 to 360 degrees using Scratch 3 or Python.

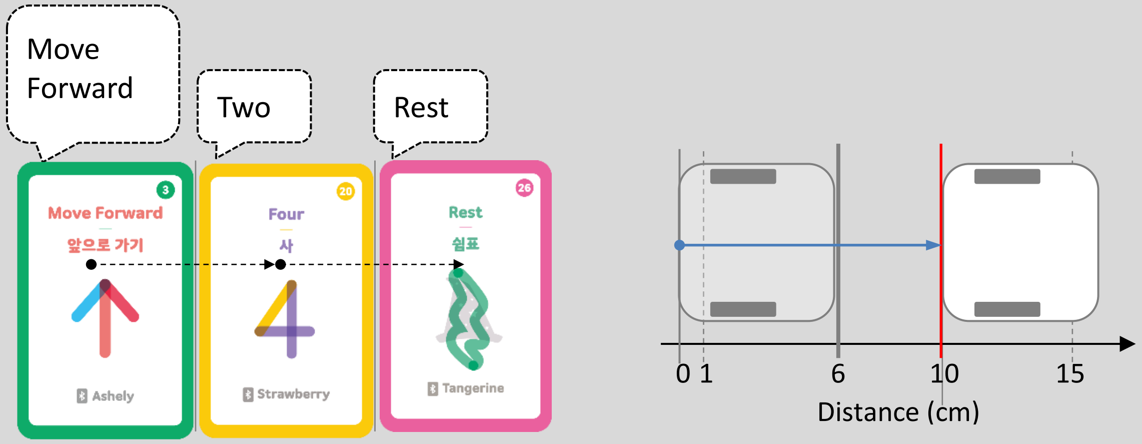


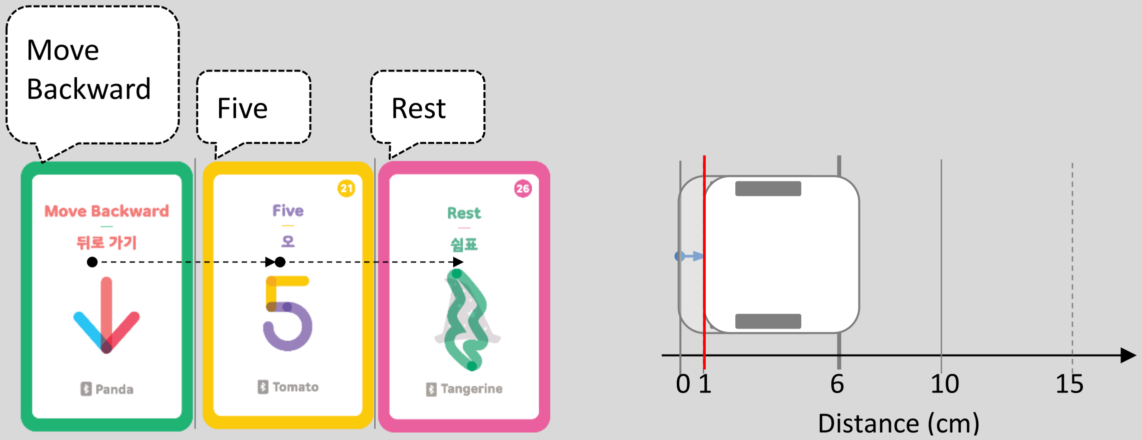
Change Distance

To change the distance the robot moves forward or backward, tap the Move Forward or Backward card, then tap the number card. The default distance for Move Forward or Backward cards is 6cm. When you tap a number card, the distance is added or subtracted from the default value by the number. The distance range by the Move Forward or Backward card is from 1cm to 15cm.

Increase or decrease distance

Tap the Move Forward card and the number cards (1 to 9) to increase the distance, or tap the Move Back and number cards (1 to 5) to decrease the distance. After adjusting the distance, tap the Rest card to write the value to non-volatile memory. Even if you turn the power off and on afterwards, the set value is applied to the distance value.



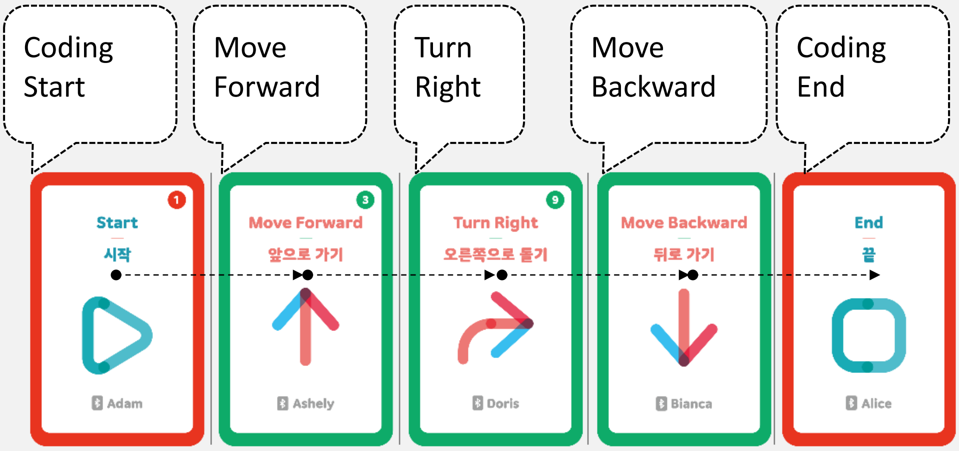


Create Coding Data

Tap the card to start unplugged coding. The robot automatically detects the card and speaks the card name through the speaker. Tap the start card, then other cards to create a set of coding data, and then finally tap the end card to have the robot run the coding data.

To create coding data

Coding cards must be placed between the Start and End cards. Once the coding data set is complete, the robot will run the coding data about 2 seconds later.



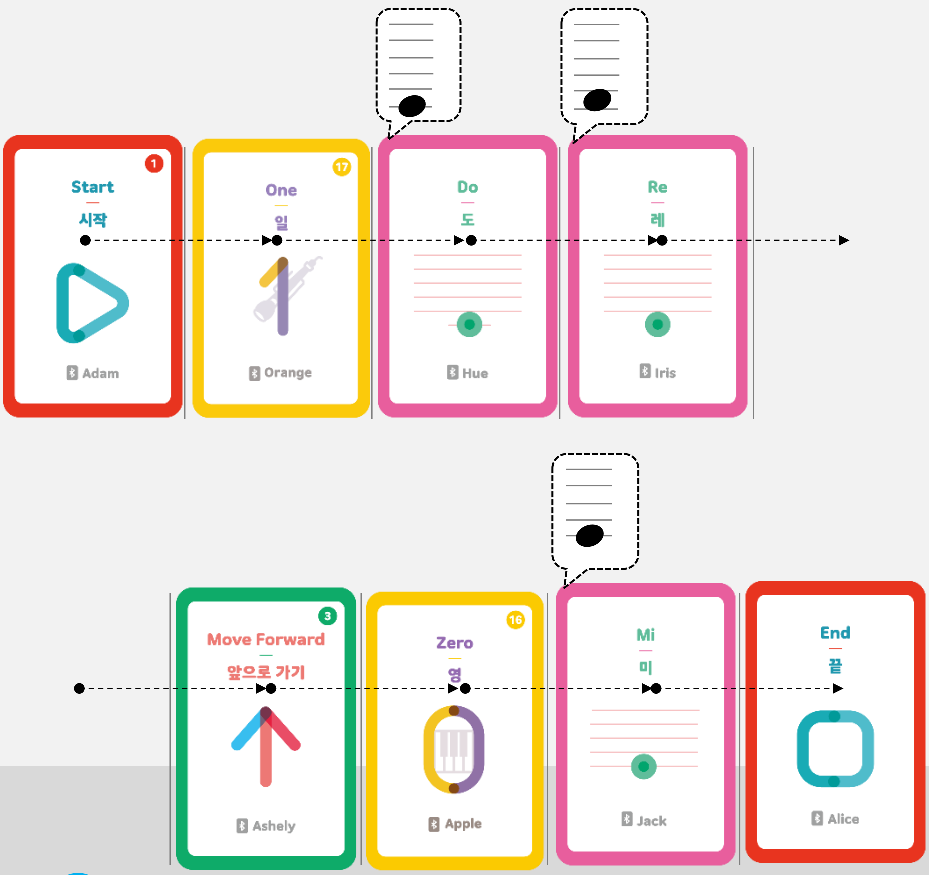
Press the button to start or stop coding data in memory

If an unplugged data set is in volatile flash memory, press the button to run the data or stop it immediately.

Set Instrument to Music Data

To set an instrument to a note data set, place the number card before the music card. If a number card between 0 and 2 precedes a note, the instrument corresponding to that number is set to the note.

To create music data with different instrument settings
Place the instrument card in front of the music card, and add motion cards or other cards to the data set.

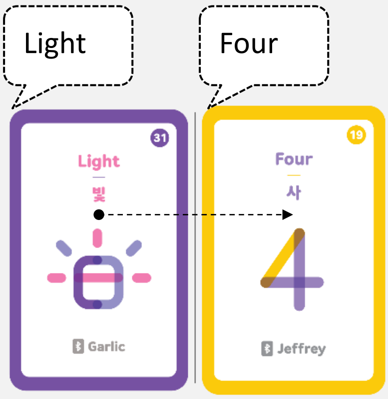


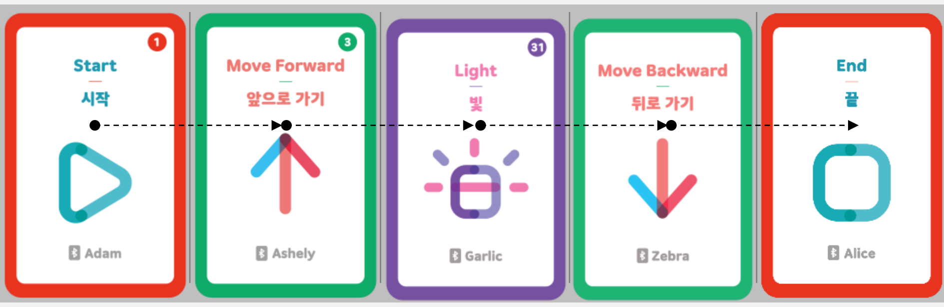
Whole or quarter note
When tapping a music card, it is played as a whole note, but when played as a music recipe within coding data, it is played as a quarter note.

Color Dance

When the Light card is added to the dataset, the zigzag motion is repeated several times while rotating at a small angle left and right as many as the number set on the light card and number card. And as it moves in a zigzag motion, the color changes quickly.

To create zigzag motion and color changes
To set the repeating zigzag number and color, first use the Light and number cards 1 to 9 to set the LED color, then add the light card to the coding data.





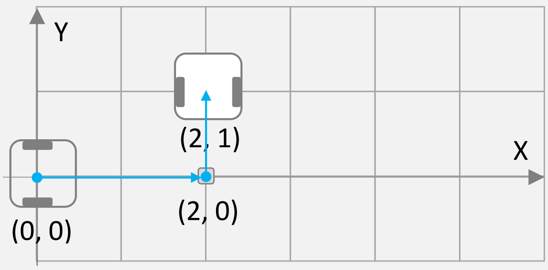
In the classroom
Due to resistance errors, systematic errors between the two wheels, gear ratio and backlash errors, step angle resolution, etc., unwanted position errors will occur after the color dance.

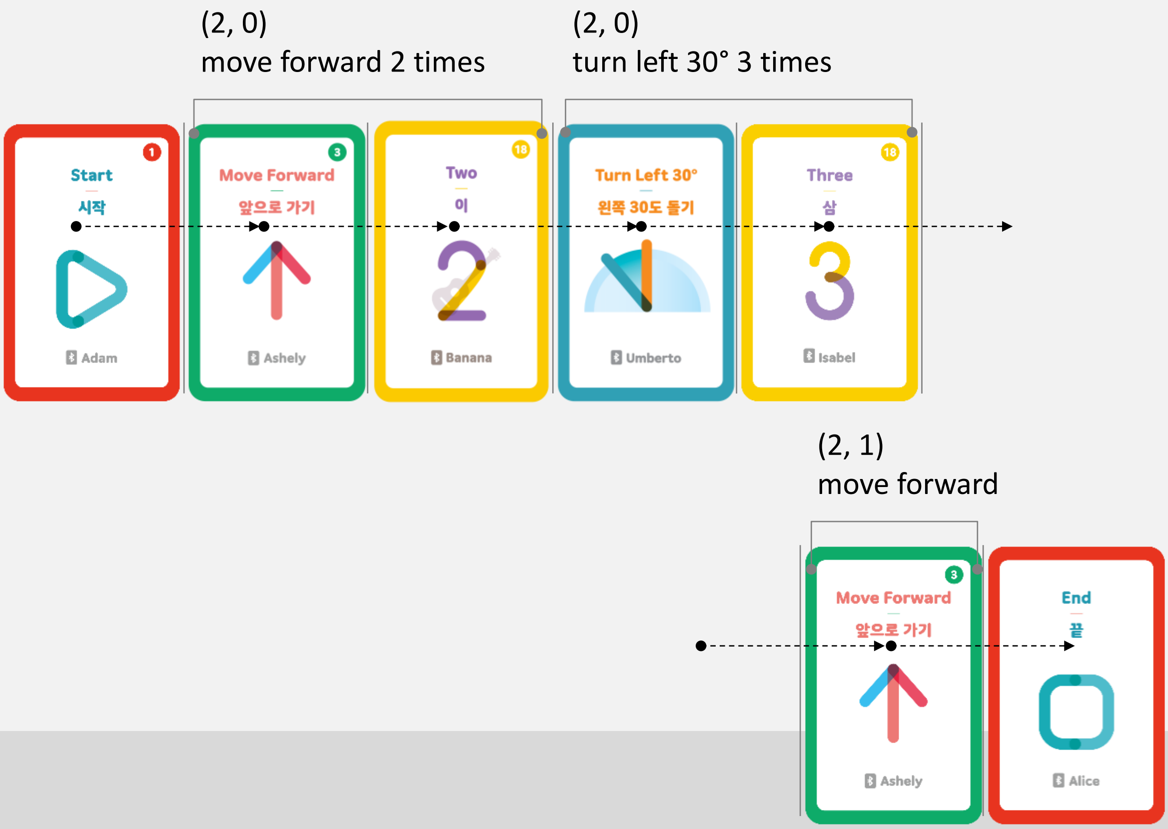
Linear and Rotational Motion

To create a linear motion or rotational motion on a horizontal plane, use the Move Forward, Move Backward cards or the Turn Left, Turn Right cards or, the Turn Left and Right 15°, 30°, 60°, 90° cards.

To add motion and rotation card to coding dataset

To create straight or rotating motions with fewer cards, it is useful to combine motion, rotation and number cards.



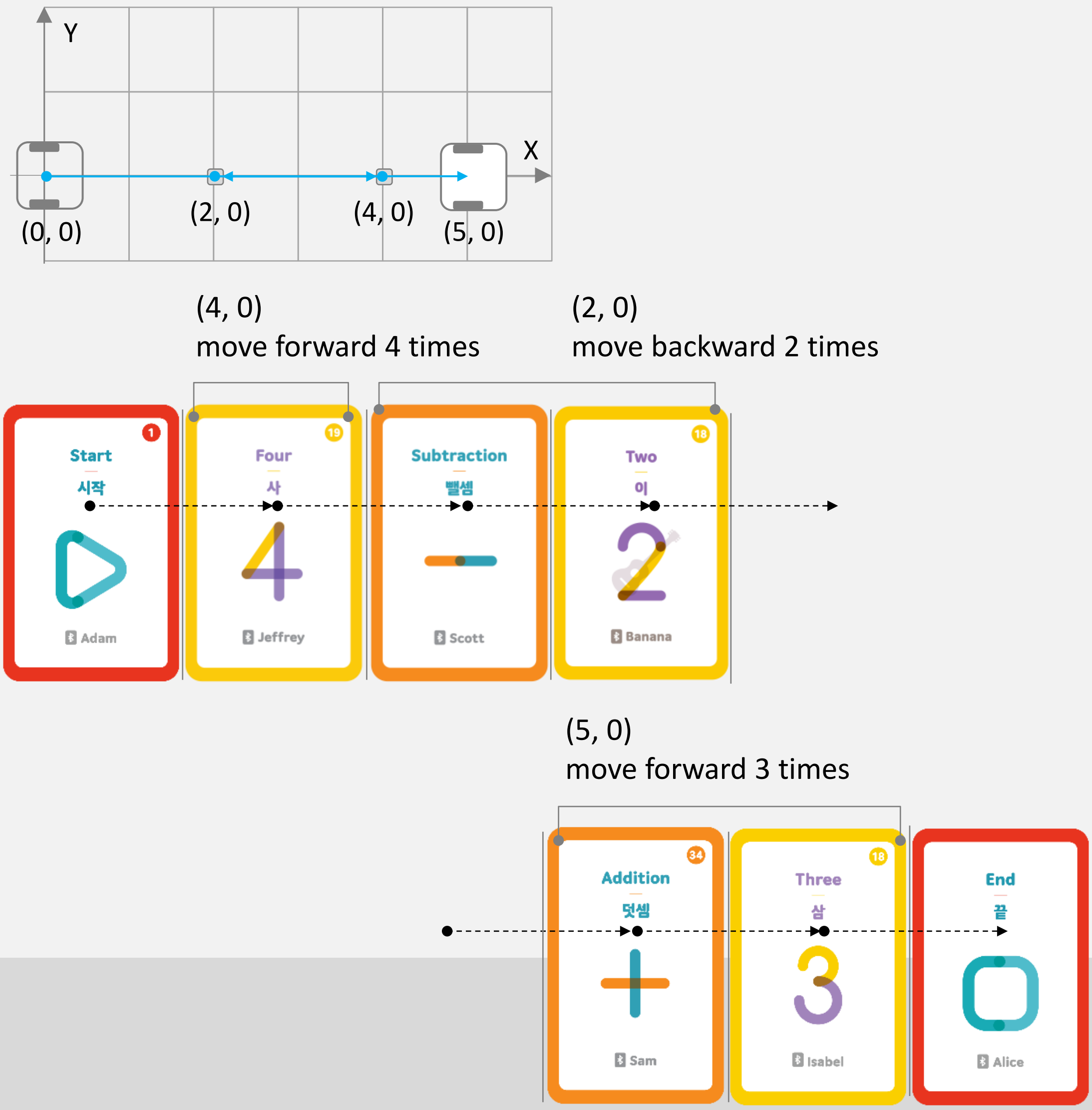


Linear Motion and Numerical Recipes

To move back and forth in a horizontal plane, use a decimal number card that matches the number of times you want to repeat the Move Forward or Back card. The numeric operator the Plus (+) card can be used to move forward and the Minus (-) card can be used to move backward.

To add decimal number card to coding dataset

Use number and operator cards according to numerical recipe data defined as (number, operator, number).

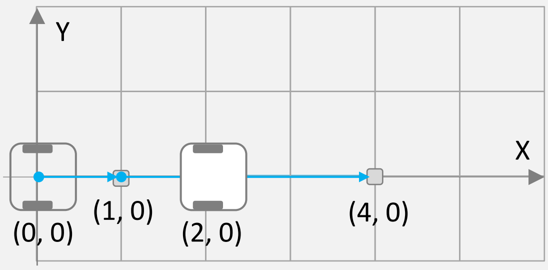


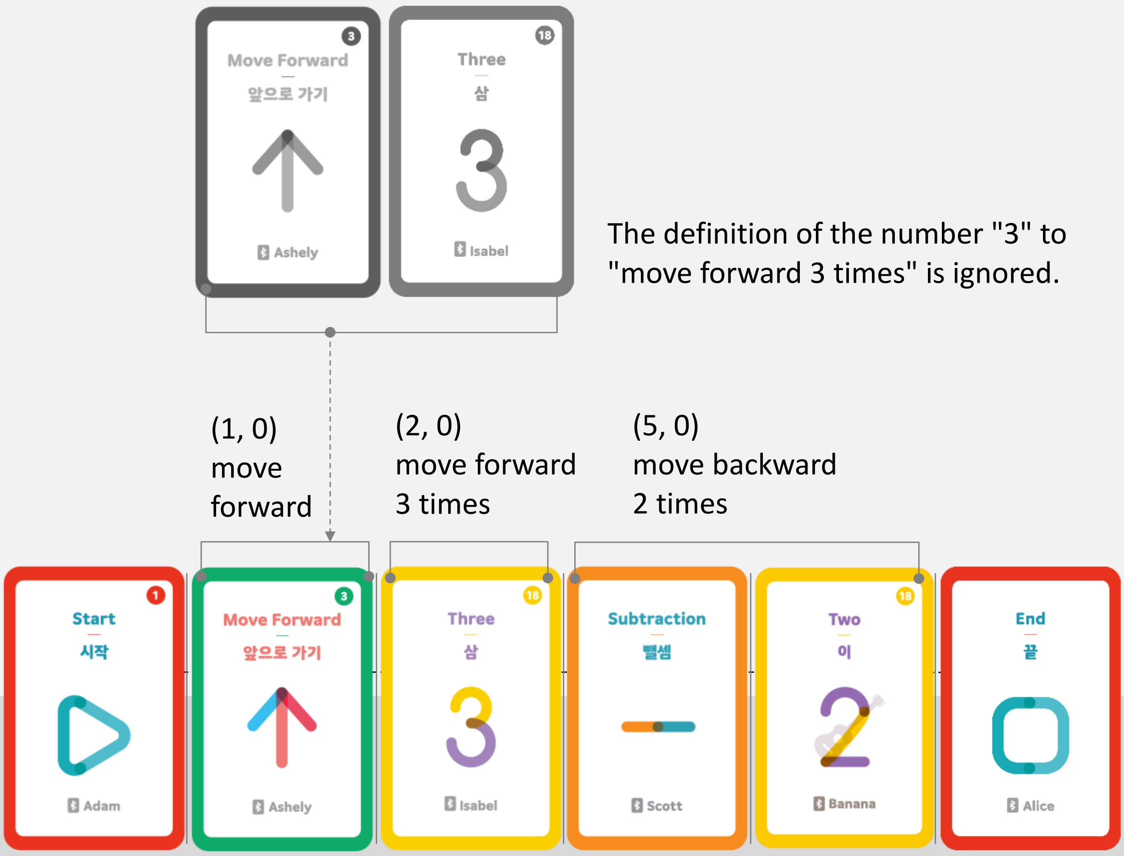
Example 1: Complex Expressions with Mathematics

When action, math, and number cards are in complex representations, priority can affect the resulting action. If you place one motion and one number card each before the mathematical operation, the motion card will be ignored because the math operation has a higher priority over the motion card.

Combining motion and math operations

Motion card preceding math operations is ignored.



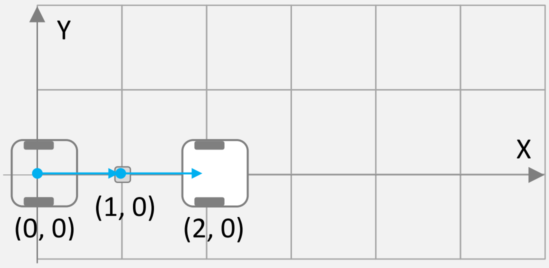


Example 2: Complex Expressions with Mathematics

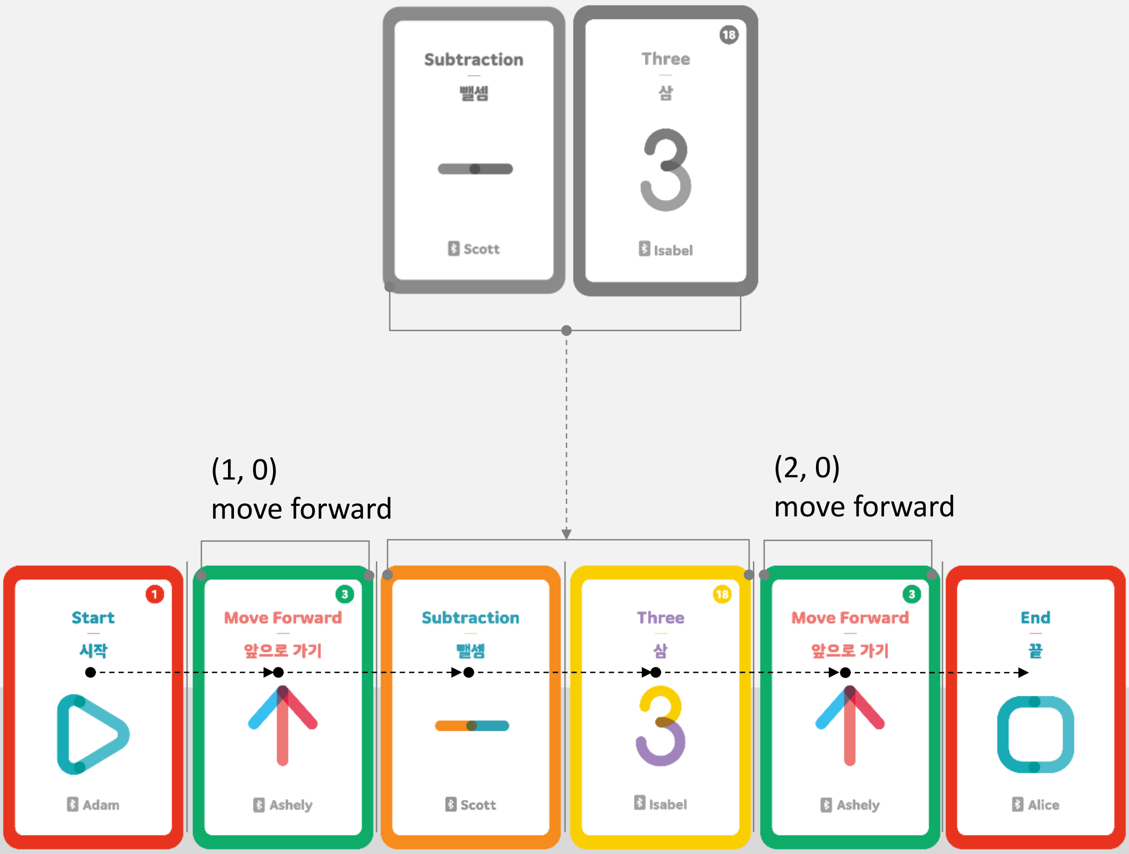
When complex expressions have action, math, and number cards, their priority affects the resulting action. If you place some motion and number cards before the mathematical operation, the math operation has a higher priority, so the math and number cards are ignored.

Combining motion and math operations

Motion card preceding math operations is ignored.



Invalid math operations 'subtract and number 3' are ignored. The number card must be placed in front of the math operation card.

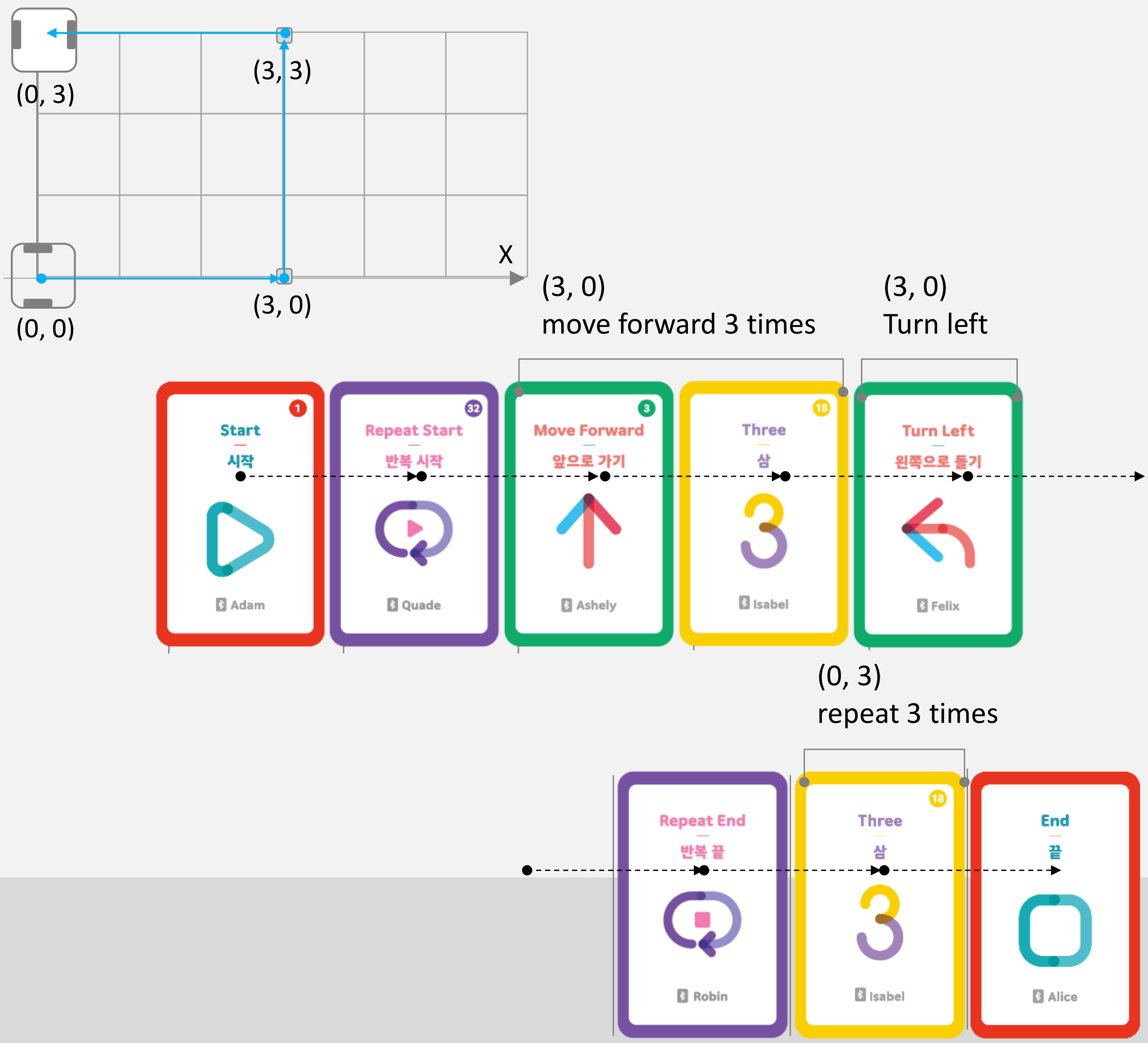


Repeat Coding Data

To define a repeating function with a dataset, add a card between the two repeating cards. You can add coding days to your coding recipe, such as repetitions, functions, numbers, notes, sounds, actions, or rotations between repetitive datasets.

To define repeat number

To repeat a set of coding data, you must add a number card to the end of the Repeat card. If no number card is placed, repeat 2 is assigned. When creating a coding recipe, you can also add number cards to the dataset for other purposes.

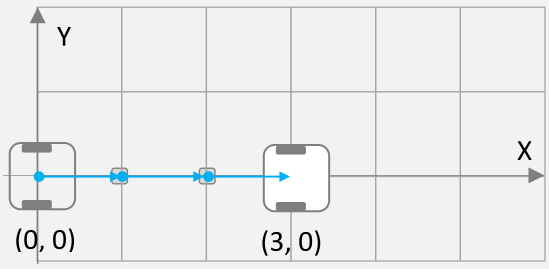


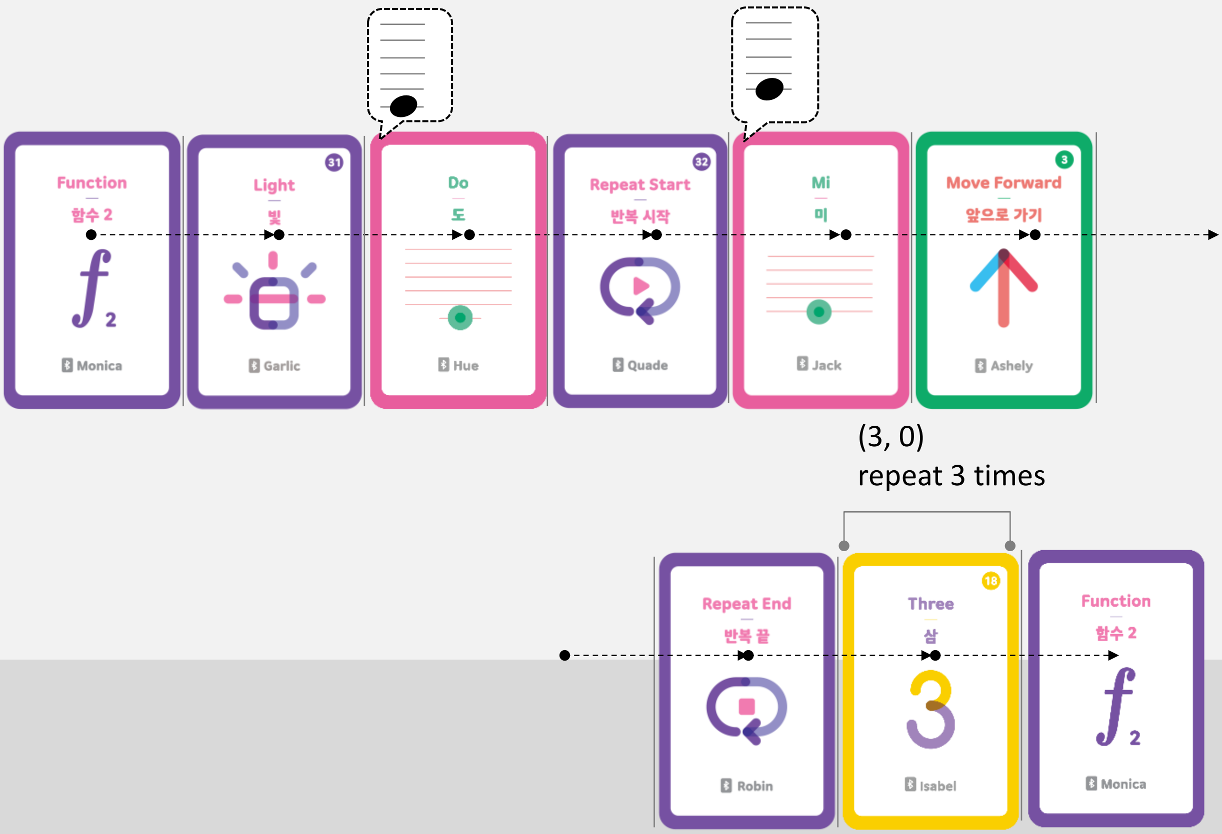
Create Function Data

To define a function with a data set, add a card between the two function cards. Motion, rotation, light, repeat, sound, music note, and number cards can be inserted into the function data set. Tap the Function card to start defining your function data set, or tap it again to stop.

To define function data

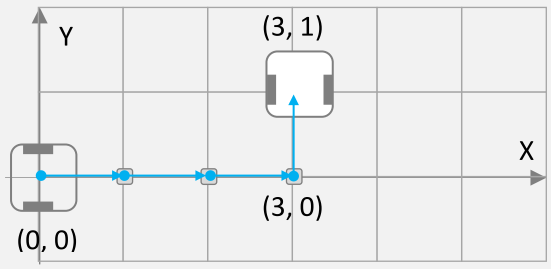
Add cards containing coding recipes between two function cards. Function cards containing coding recipes can be used in coding activities along with other cards.





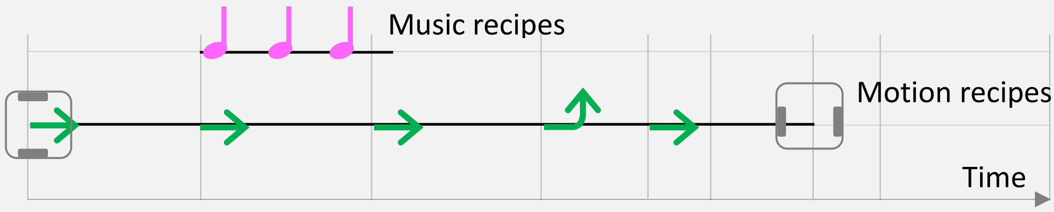
Add Function to Coding Data

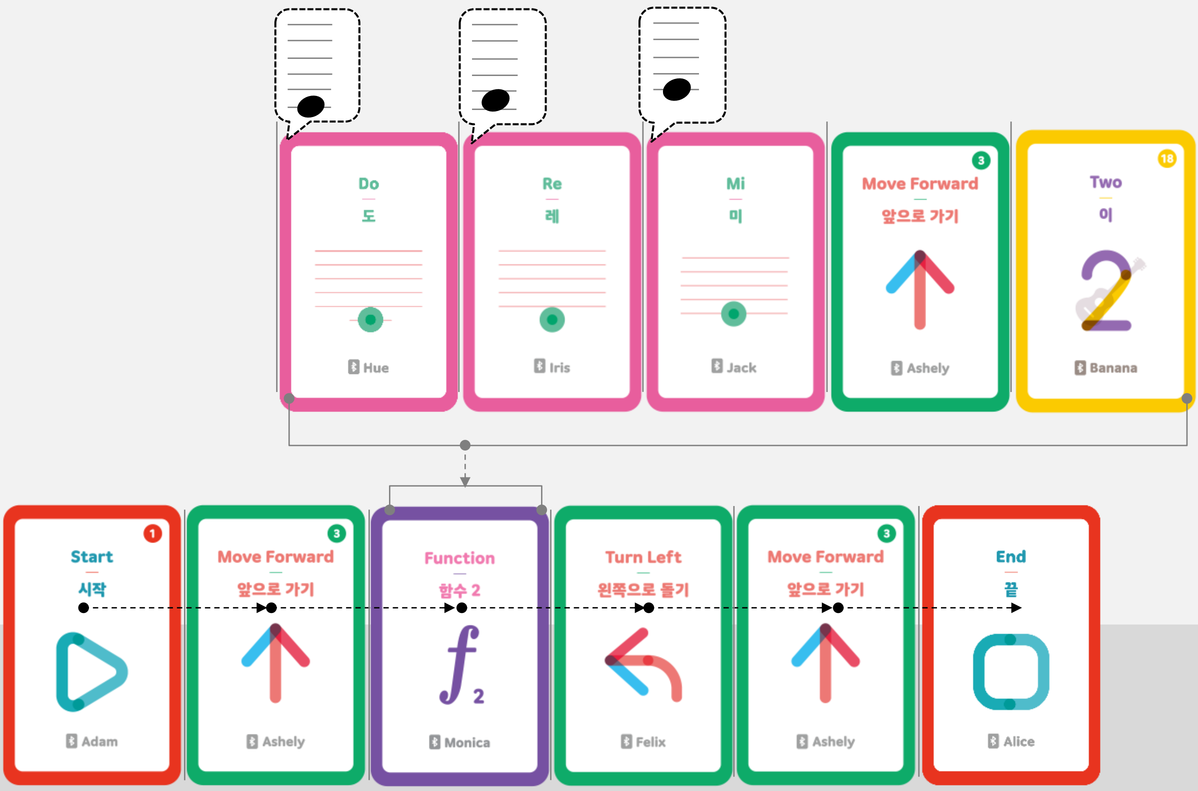
Previously created function data can be added to the coding dataset using the Function card. In a coding dataset, you can add repetitions, functions, numbers, music notes, linear or rotational mostions as your coding recipes.



Concurrent time series data

In the example below, you can see a music recipe series and a motion recipe series running simultaneously.



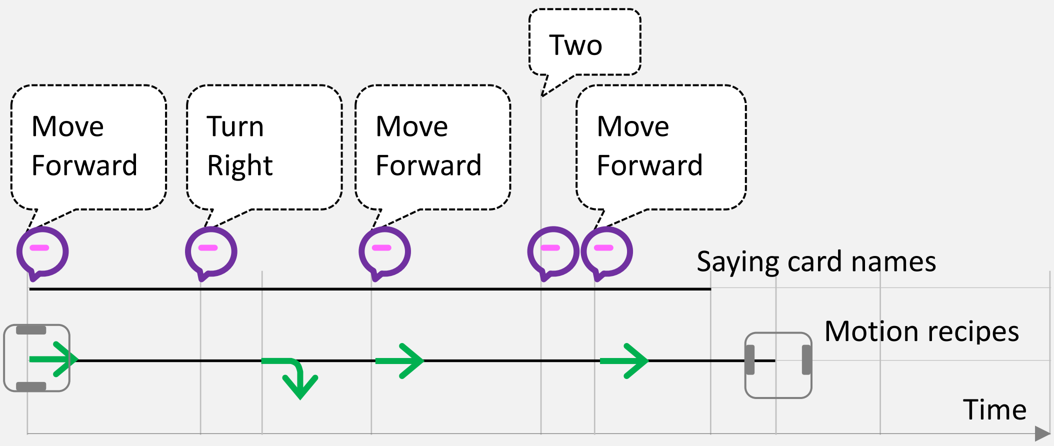


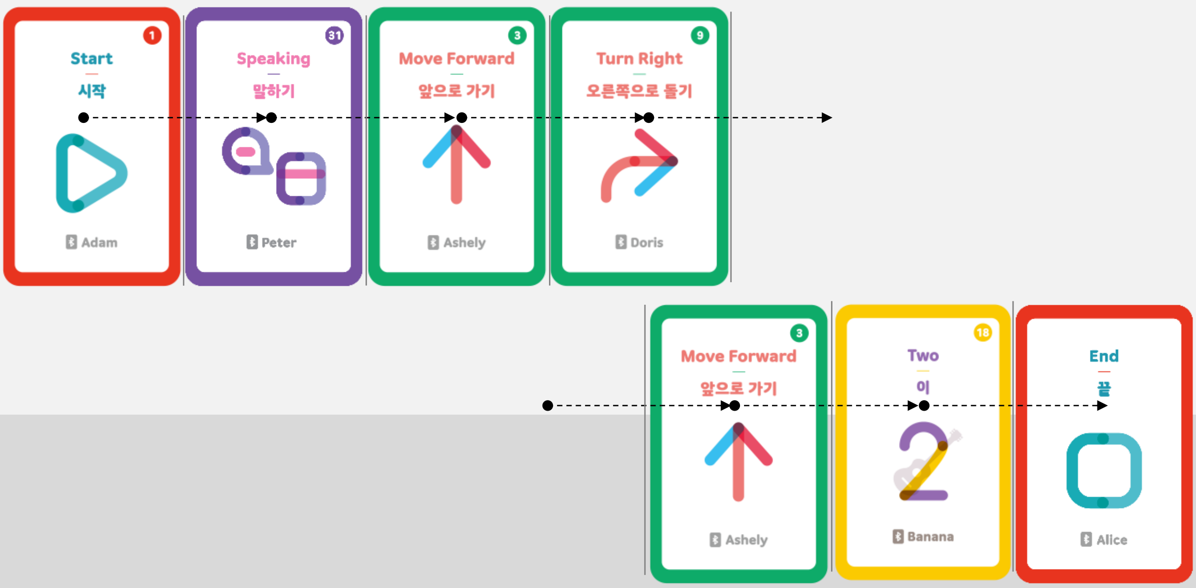
Say Card Name while Robot Runs

When you tap a coding card to begin unplugged programming, the OID optical image sensor automatically detects the card and speaks its name over the speaker. As the coding data progresses sequentially, the robot can say the card name of each coding data.

Say card name before action

Before executing the coding recipe, the robot speaks the card name and immediately takes action while speaking. However, as an exception, in the case of rotation, the action is executed after speaking.





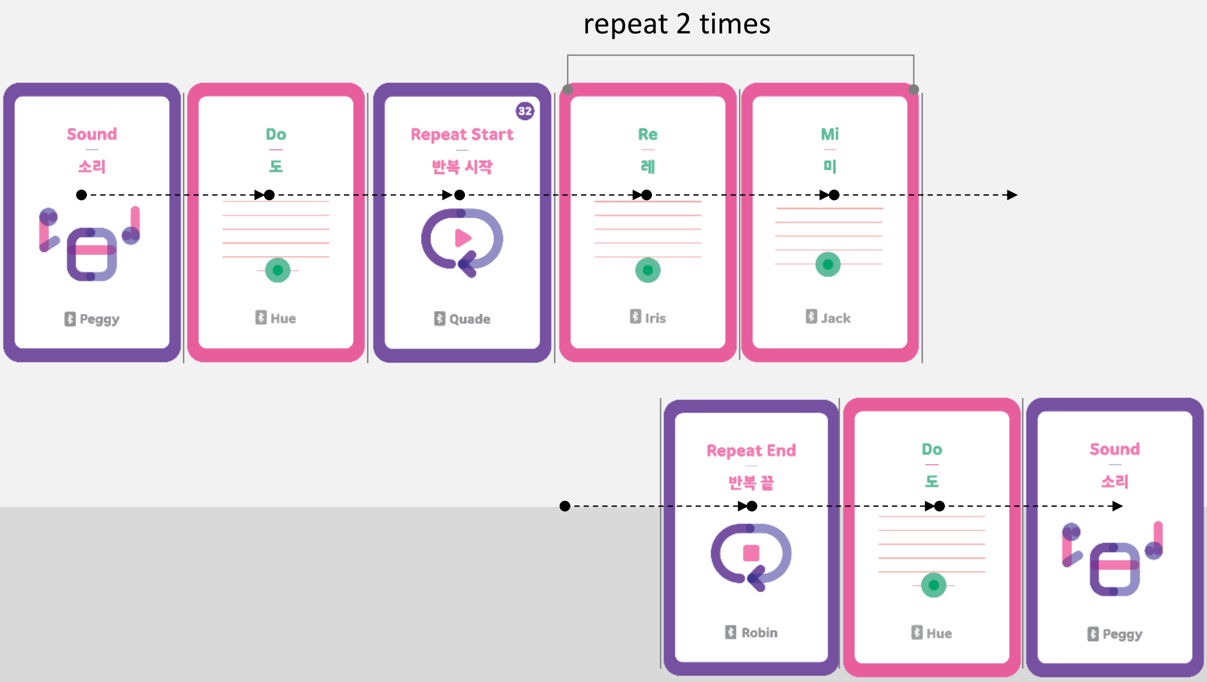
Create Sound Data

To create sound data, add a music card between two sound cards. Only Music Note and the Repeat card can be added to the sound dataset. Tap the Sound card to start defining a sound dataset, then tap again to finish. For sticker coding, place the Sound sticker on a black line and use the Grid sticker to trace the line. When the robot encounters the Sound sticker, it will play the music note previously created using the Sound card.

To create sound dataset

Create a music note recipe by adding music cards between two Sound cards. When you tap a music card, the note length is a whole note, but within a music recipe, the note length is fixed to a quarter note. In music recipes, increasing the number of repetitions with a number card after the Repeat End card is not allowed.

To create music recipes with different note lengths using the GENIBOT app for Google Android or Apple iOS.

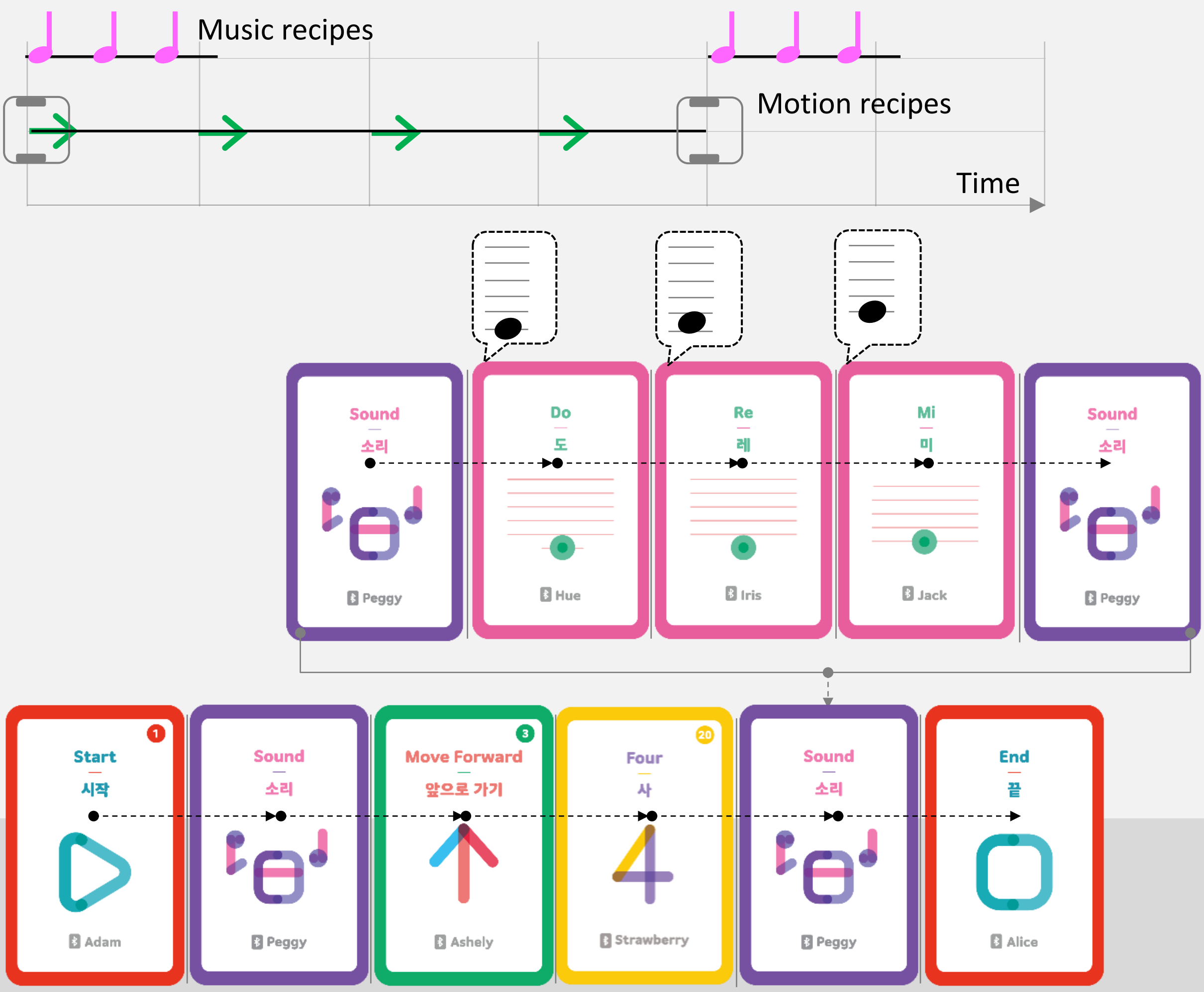


Add Sound to Coding Data

The Sound card allow you to add previously created music recipes to your coding dataset. Repetitions, functions, numbers and mathematics, music notes, linear or rotational motions can be added to the coding dataset along with the Sound card.

Music parallel to motion

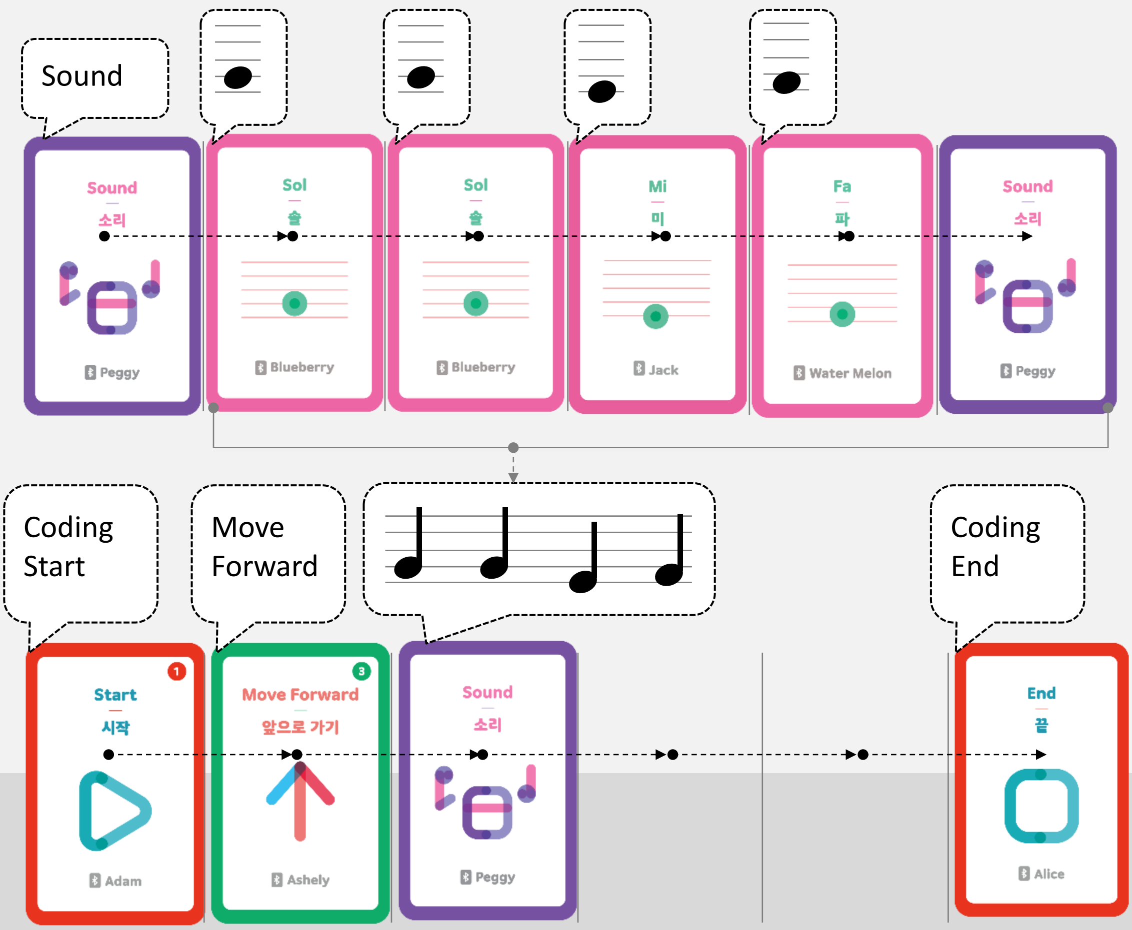
When there is motion behind music cards or a music recipe with the Sound card, this motion runs parallel to the music. The action time of music and motion varies on its recipe, and each action starts simultaneously with multitasking.



Example: Play Music Recipes while Coding

While coding, tapping a sound card that contains a music recipe will play the music recipe instead of saying the card name "Sound", however, even if the music is playing, tapping the next card immediately stops the music.
Note that when not coding, if you tap a sound card that already contains a music recipe, the music recipe will be deleted.

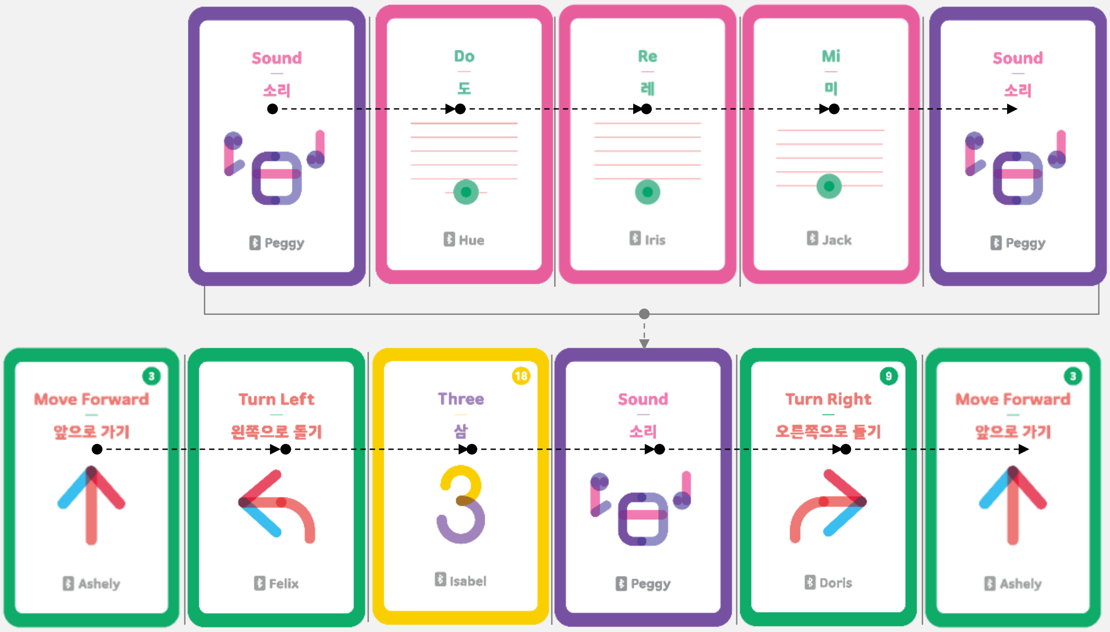
Whole notes and quarter notes
When tapping a music card, it is played as a whole note, but when played as a music recipe within coding data, it is played as a quarter note. However, if you use the GENBOT app for Google Android or Apple iOS, you can add whole notes, half notes, quarter notes, eighth notes, and 16th notes to your music recipe.



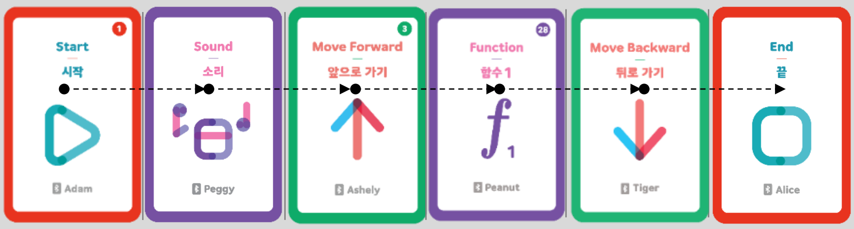
Add Sound to Function Data

Function data created with two function cards can contain various cards such as motion, sound, repetition, and numbers. Sound cards previously defined with music note recipes can be added to the function dataset.

Adding the Sound card with a music recipe in a function recipe
After creating a music recipe with two sound cards, add the Sound card to the function recipe created using two function cards. Finally, not only can you add the Function card to your coding dataset, but you can also add the Sound card to your coding dataset separately.



Coding by mixing sound, motion, and function:

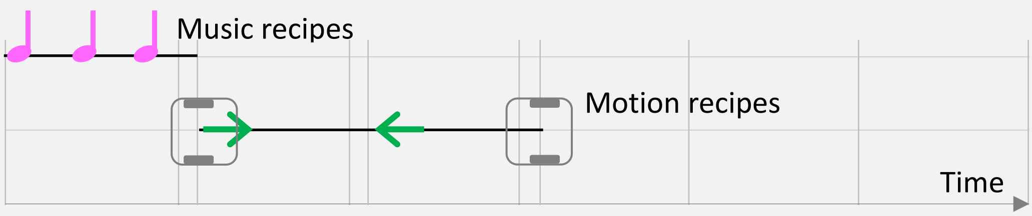


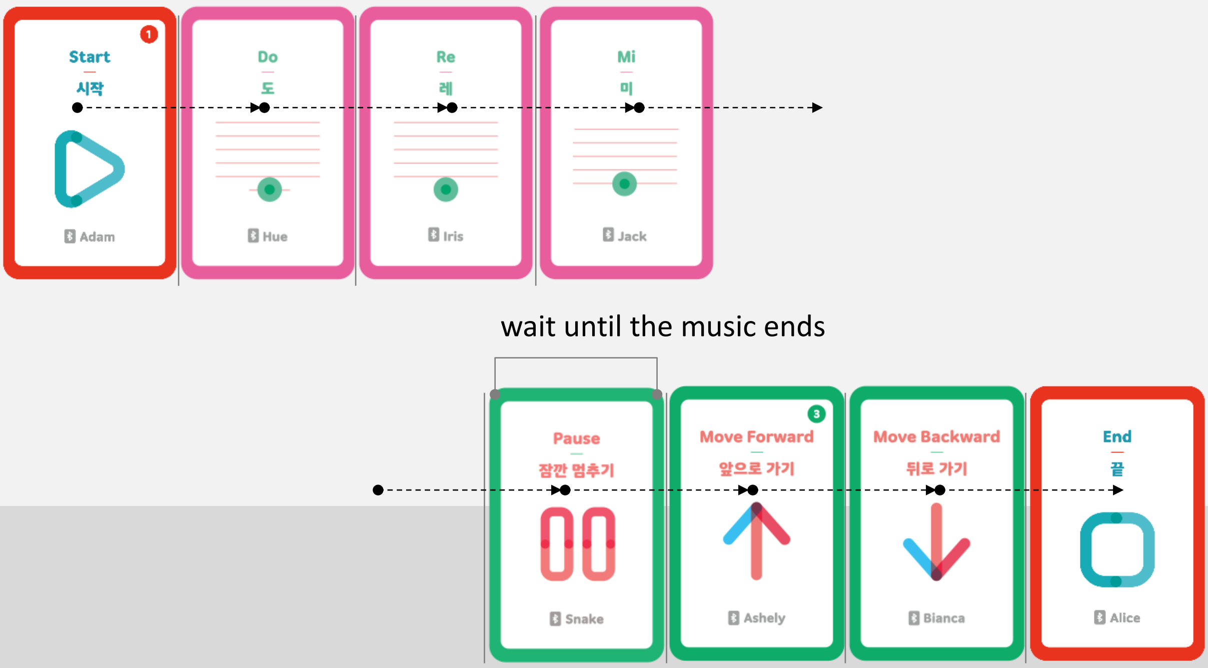
Preempting Music Notes

Motion cards are preempted by default while playing and running music and run in parallel with music note cards. To run a motion card at the end of music playback, add a pause card before the motion card.

Music cards take precedence

When the music card is in front of the motion card, if you place the Pause card in front of the motion, the music and motion will not proceed at the same time, and the motion will be started after completing the music.



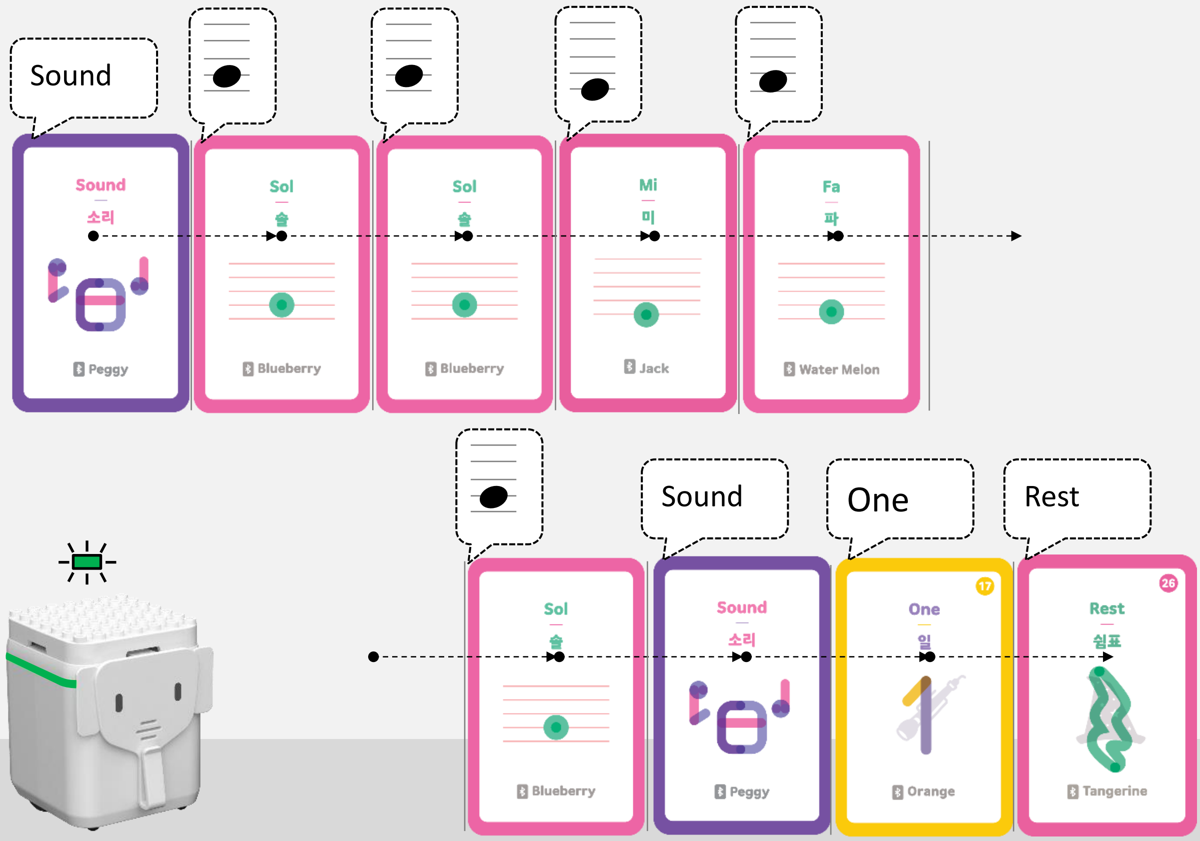


Save Sound Data

Music recipes previously created with the Sound card can be stored in the robot's non-volatile memory. To write music dataset to the robot, first create a music recipe and then tap the number card to temporarily store the music recipe in the corresponding number memory. If you then tap the Rest card, the music recipe will be saved to the robot’s memory so you can use it again even if you turn the power off and on.

Memory numbers for storing music recipes

Up to 10 music recipes can be saved using number cards 0 to 9. Even if you turn the power off and on, you can load the music recipe again using the number card you set it on.



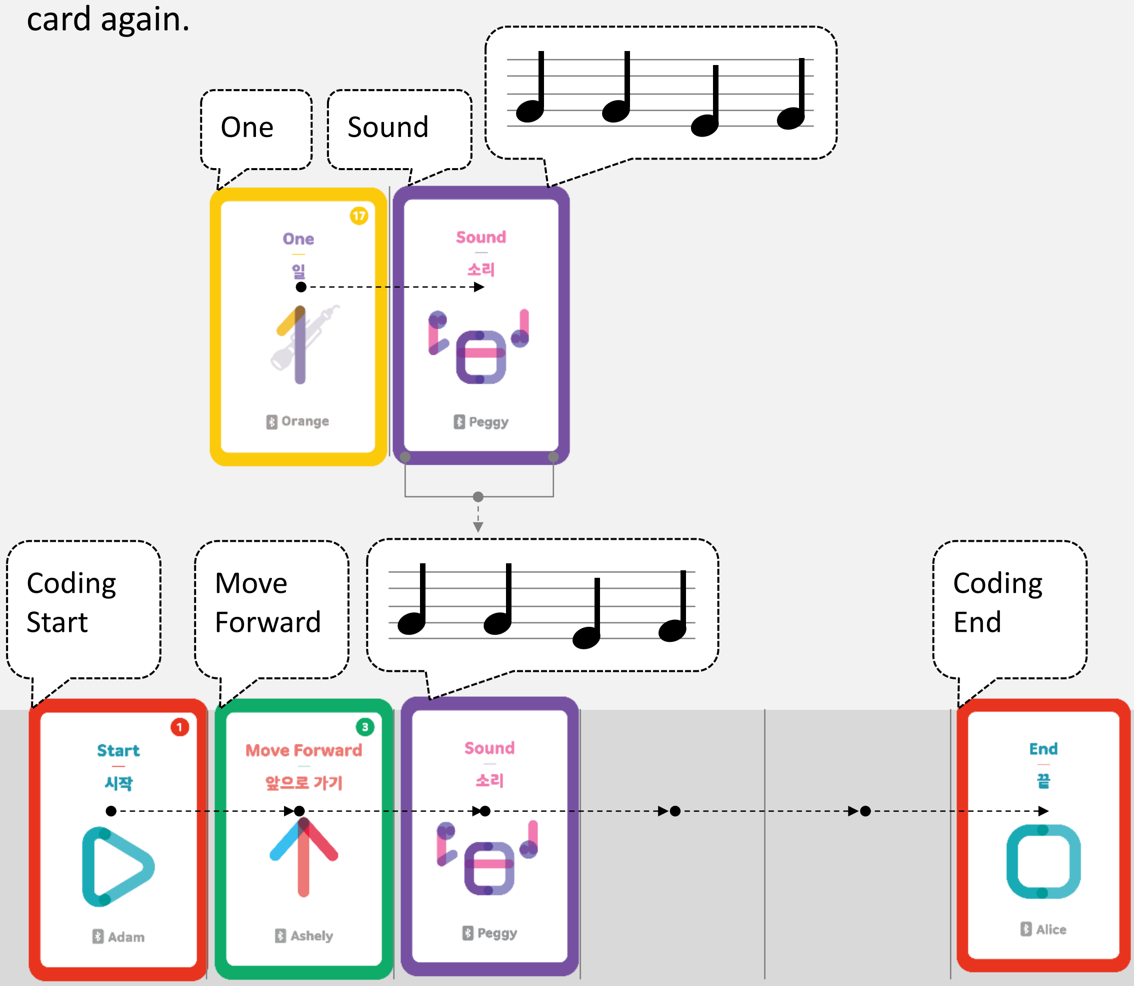
The LED quickly blinks the color corresponding to the memory storage number, indicating that the music recipe is being written to the flash.

Load Sound Data

To read sound data from the robot's memory, tap the number card corresponding to the memory number where the sound data is stored, then tap the Sound card. If the flash memory contains valid sound data, the music recipe can be loaded.

Music cards take precedence

Music recipes loaded from memory are added as sound card data, so music recipes loaded from the sound card can be added to the coding dataset. If you are not coding, tapping the sound card will delete the music recipe added to the sound card, but you can retrieve the music recipe from memory at any time by tapping the memory number card and the sound card again.

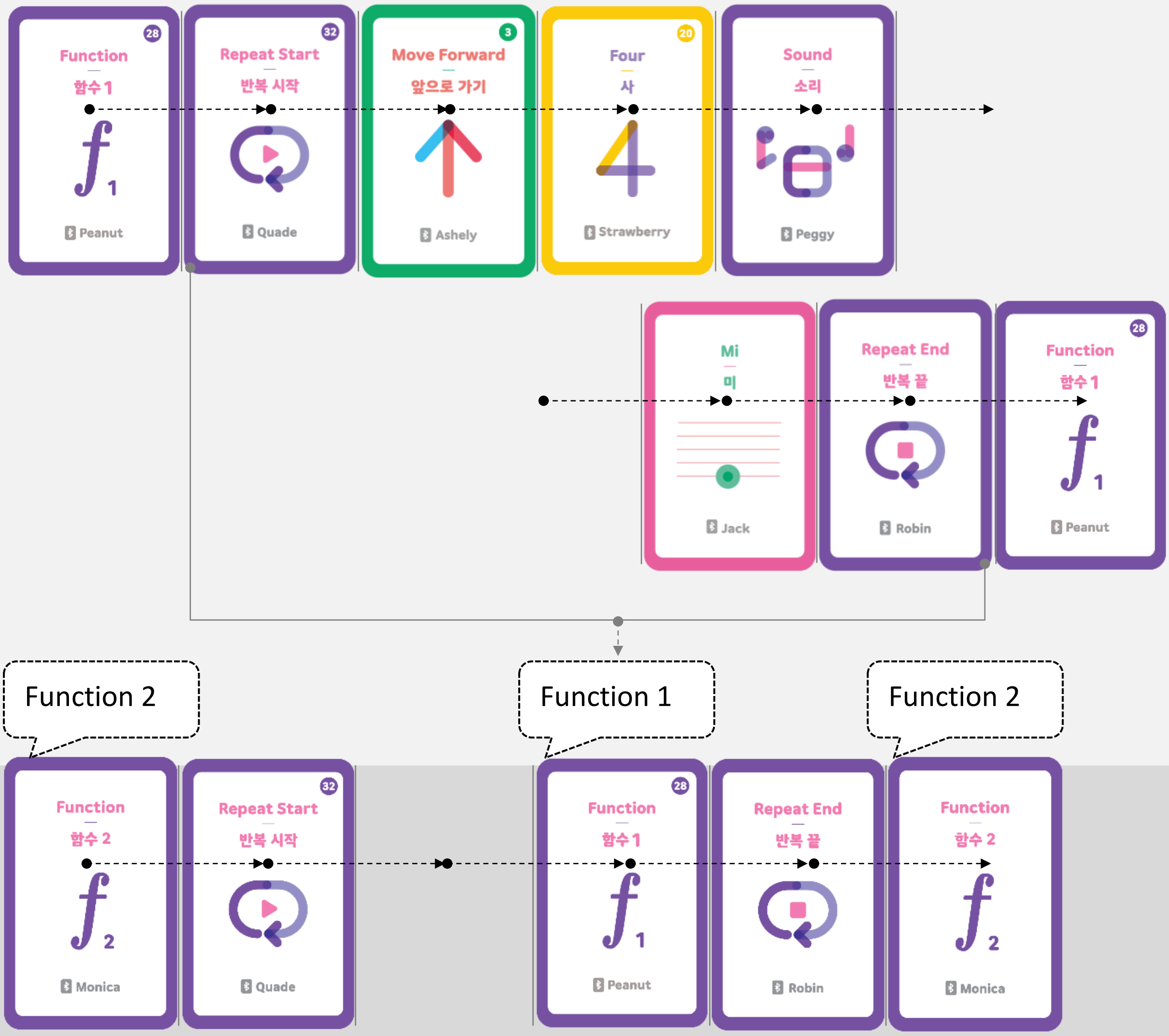


Non-recursive Function Depth

Using multiple cards such as motion, rotation, light, sound, musical note, and number cards, you can add a predefined function card to another function card. Since there are two function cards f1 and f2, so you can add f1 data to f2 data, or vice versa, add f1 data to f2 data.

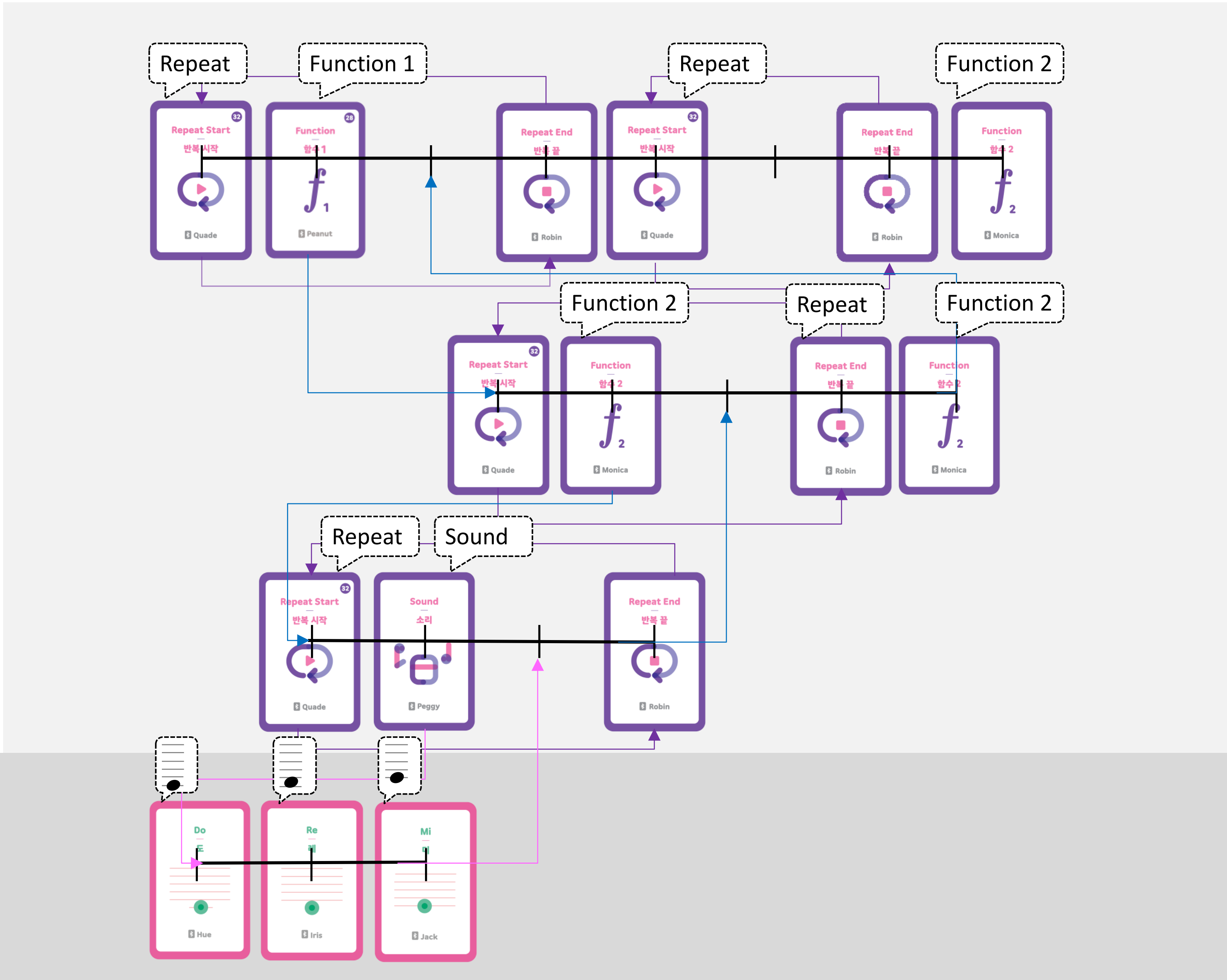
Another function within a function

The Function card f1 can contain Repeat and Sound cards, and another Function card f2 can contain this Function card as well as Repeat and Sound cards, and vice versa.



Advanced: Data Depth and Complexity

When functions and repeat cards are mixed in a complex way, you can try coding to solve data complexity by understanding how to solve them with sequential logic. The example below shows that data inside Function f2 can be used inside Function f1, and can also be used independently outside function f1. In the example below, the music recipe stored on the Sound card is used inside Function f2, but can also be used independently outside of Function cards f1 and f2.



Deleting Card During or After Coding

To delete a card from a coding dataset you are coding or have finished coding, tap the Delete card, then tap the card you want to delete, then tap the Delete card again. If there are multiple cards with the same Bluetooth name as the card you want to delete, the last card in the order of the dataset will be deleted.

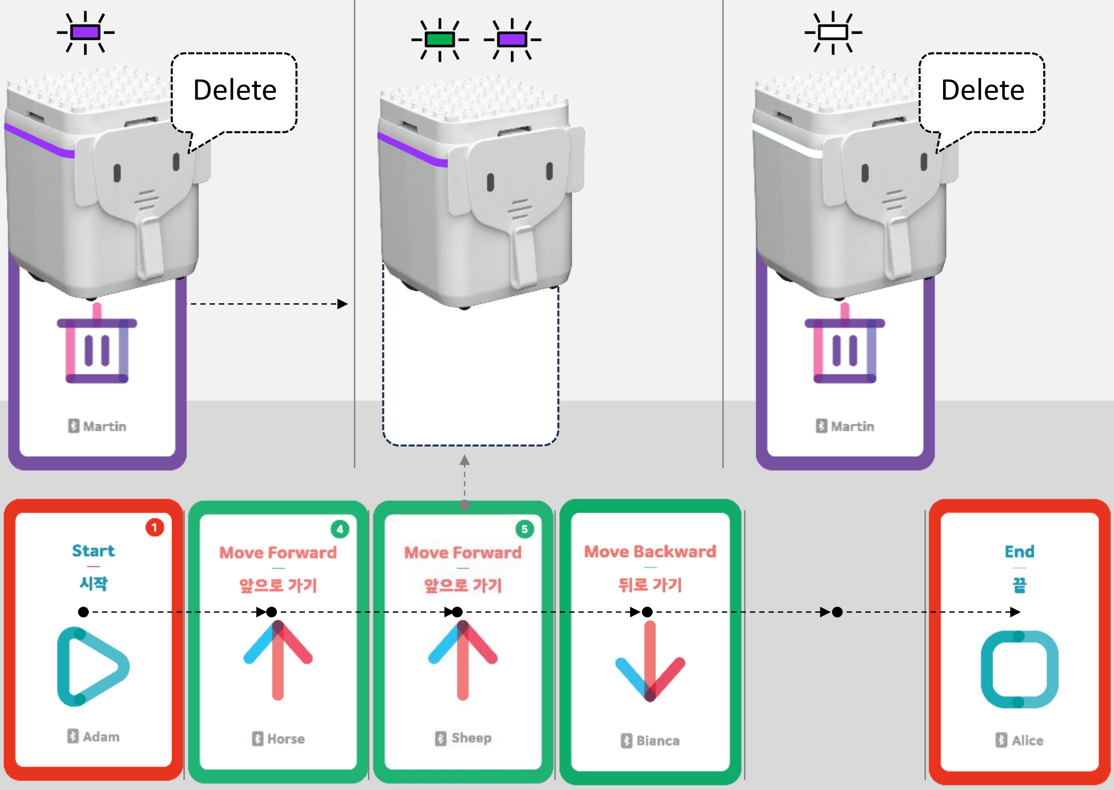
Delete one or more cards from a coding dataset

For example, if your dataset contains Move Forward cards with the Bluetooth name "Sheep" and "Horse", and you tap on the card with the Bluetooth name "Sheep" among the two cards, the tapped "Sheep" card will be deleted.

When you tap the Delete card, it will blink purple and you can delete existing cards.

Delete one or more cards.

Tap the Delete card again to complete the delete action.

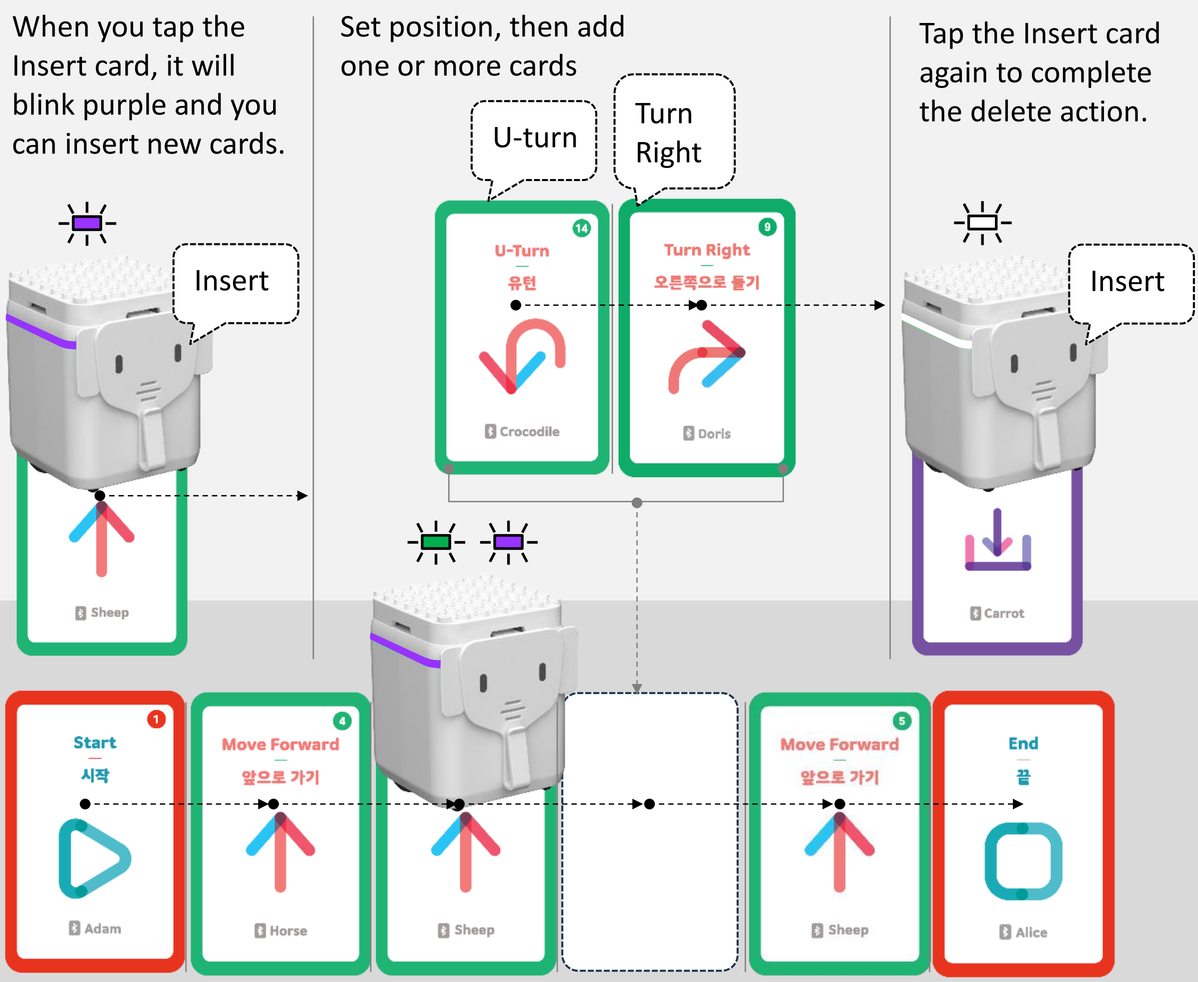


Inserting Card While or After Coding

To insert a new card into a dataset while coding or after coding, tap the Insert card, then tap the card preceding the position in the dataset where you want to add the new card. Once you've finished selecting where you want to insert, tap one or more cards in order to add them, then tap the Insert card again.

Adding one or more cards to a coding data set

For example, if your dataset contains Move Forward cards with the Bluetooth name "Sheep" and "Horse", and you tap on the card with the Bluetooth name "Sheep" among the two cards, the tapped "Sheep" card becomes the starting position for inserting new cards.



Tilt Coding

⚠ Tilt coding is not possible for models without an acceleration sensor.

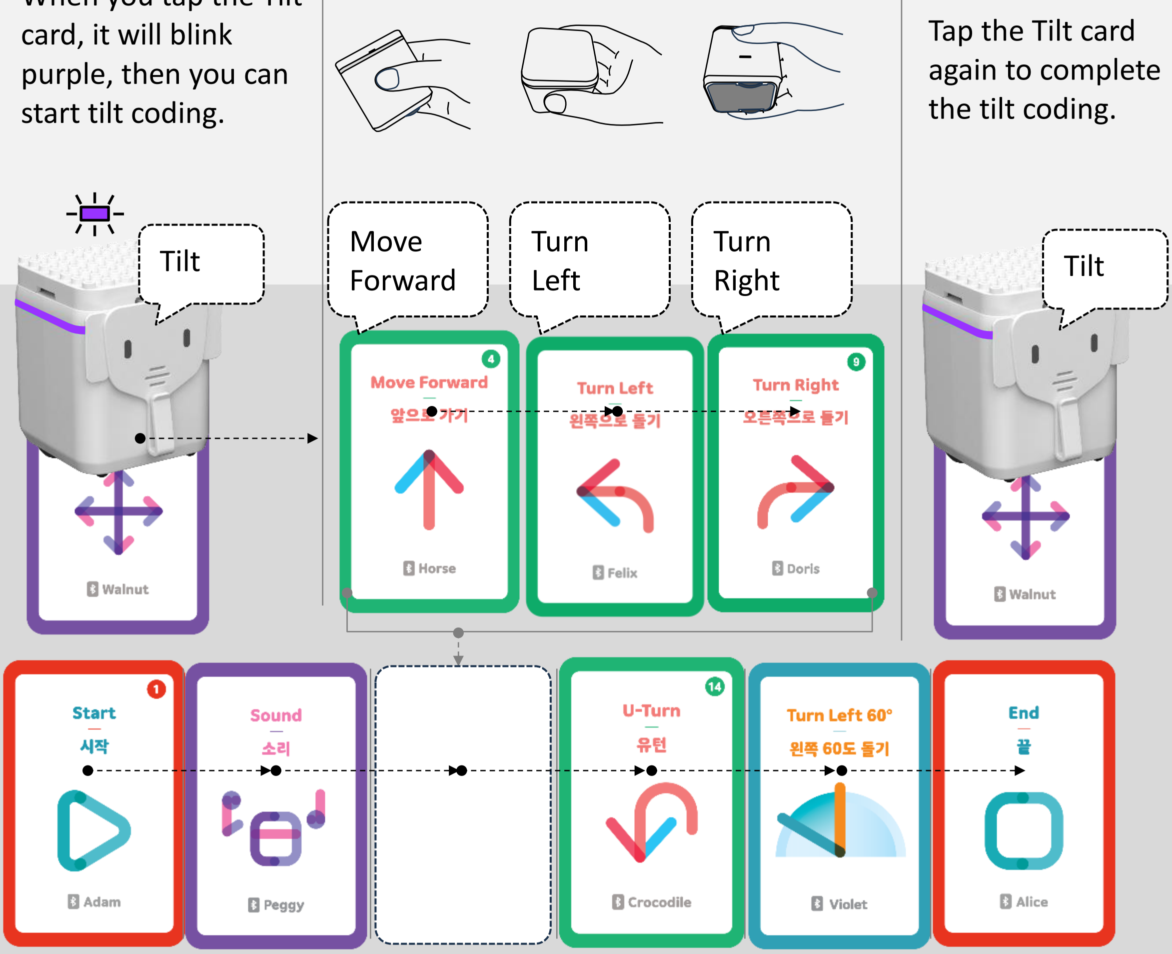
By tilting the robot, you can add the Move Forward, Move Backward, Turn Left, or Turn Right cards to your coding dataset. While coding, tap the tilt card to start tilt coding. You can also add a card by tapping it while the purple color is flashing. If you are finished with tilt coding and want to continue coding, tap the Tilt card again.

Getting started with Tilt coding

When you tap the Tilt card, the robot will blink purple, and you can start tilt coding while card coding. When you tilt the robot forward, backward, left, or right, the card corresponding to the tilt state is automatically added to the coding dataset.

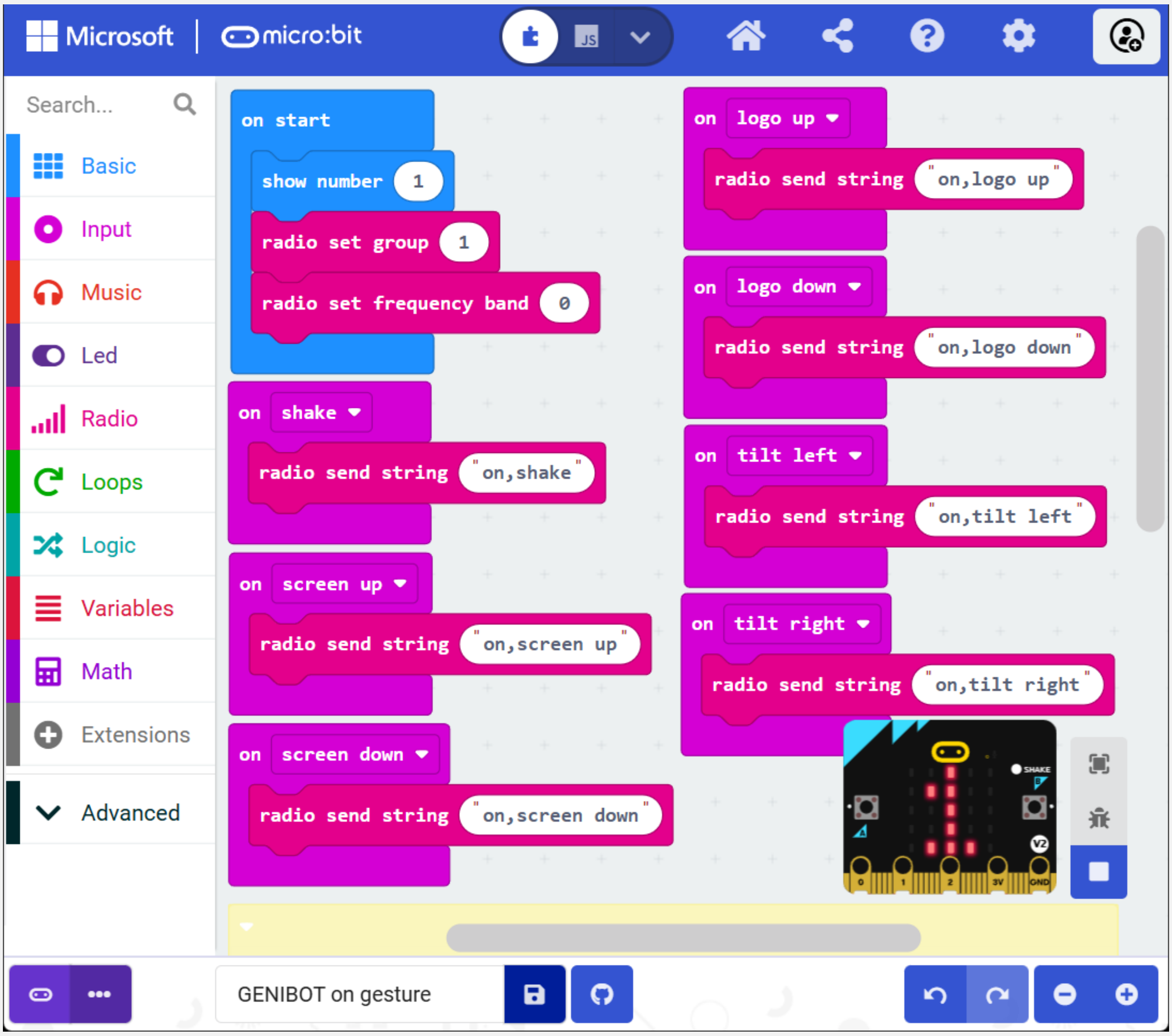
When you tap the Tilt card, it will blink purple, then you can start tilt coding.

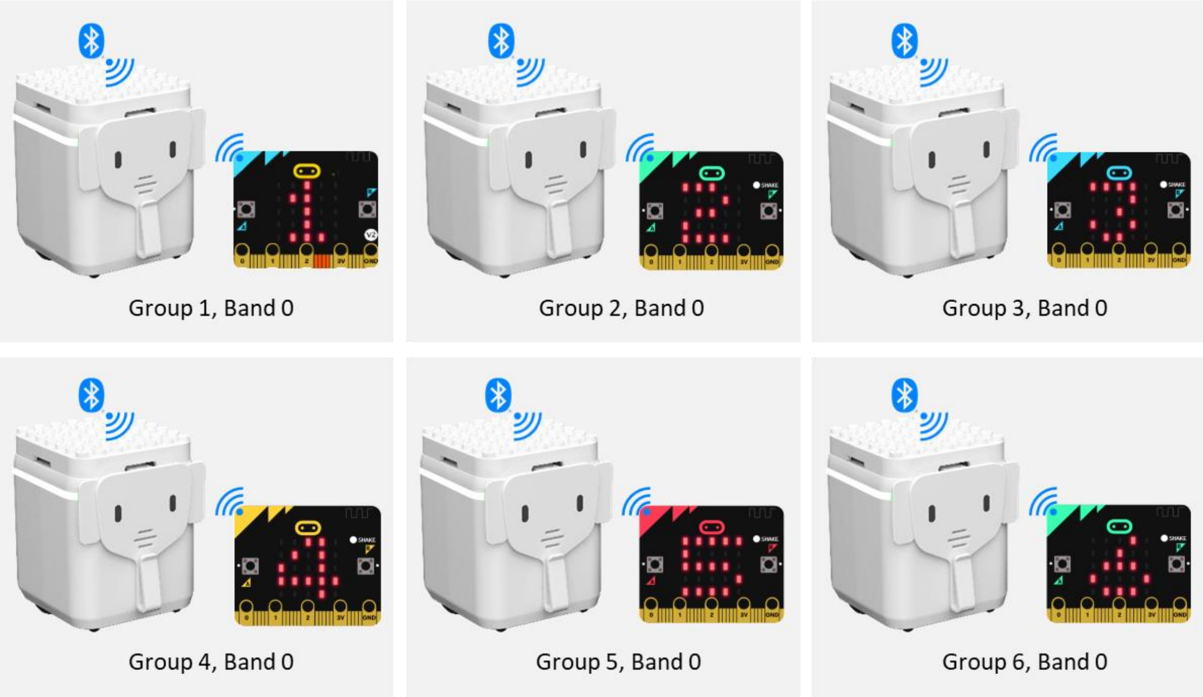
Tap the Tilt card again to complete the tilt coding.



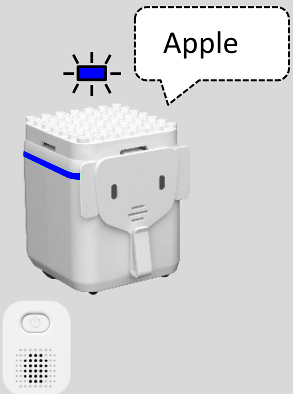
Advanced: Tilt Coding with BBC micro: bit's "On Gesture"

1. Connect the BBC micro:bit to your computer. To program BBC micro:bit, go to MakeCode website https://makecode.microbit.org/#editor.
2. When there are multiple BBC micro:bits in the classroom, set the radio group number independently for each micro:bit. Only GENIBOT and BBC micro:bit with the same group number can broadcast and listen to each other, so you can avoid confusion in the classroom when multiple BBC micro:bits are broadcasting and multiple GENIBOTs are listening at the same time.
3. Create a program using "On Gesture" blocks as follows, download it to the BBC micro:bit and then start the BBC micro:bit.

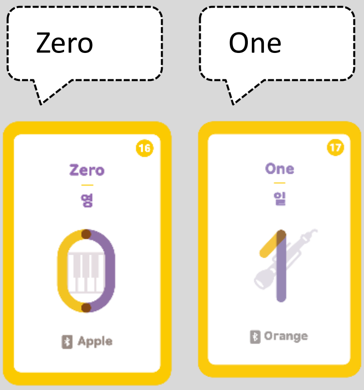




4. Once you have finished programming the BBC micro:bit, set the radio group number of the GENIBOT as shown in the following example. The example below sets the group number to 1. Since the group number is a two-digit number, add the number 0 in front if the number is less than 10. When setting a group number with a card, the band number defaults to the number 0.



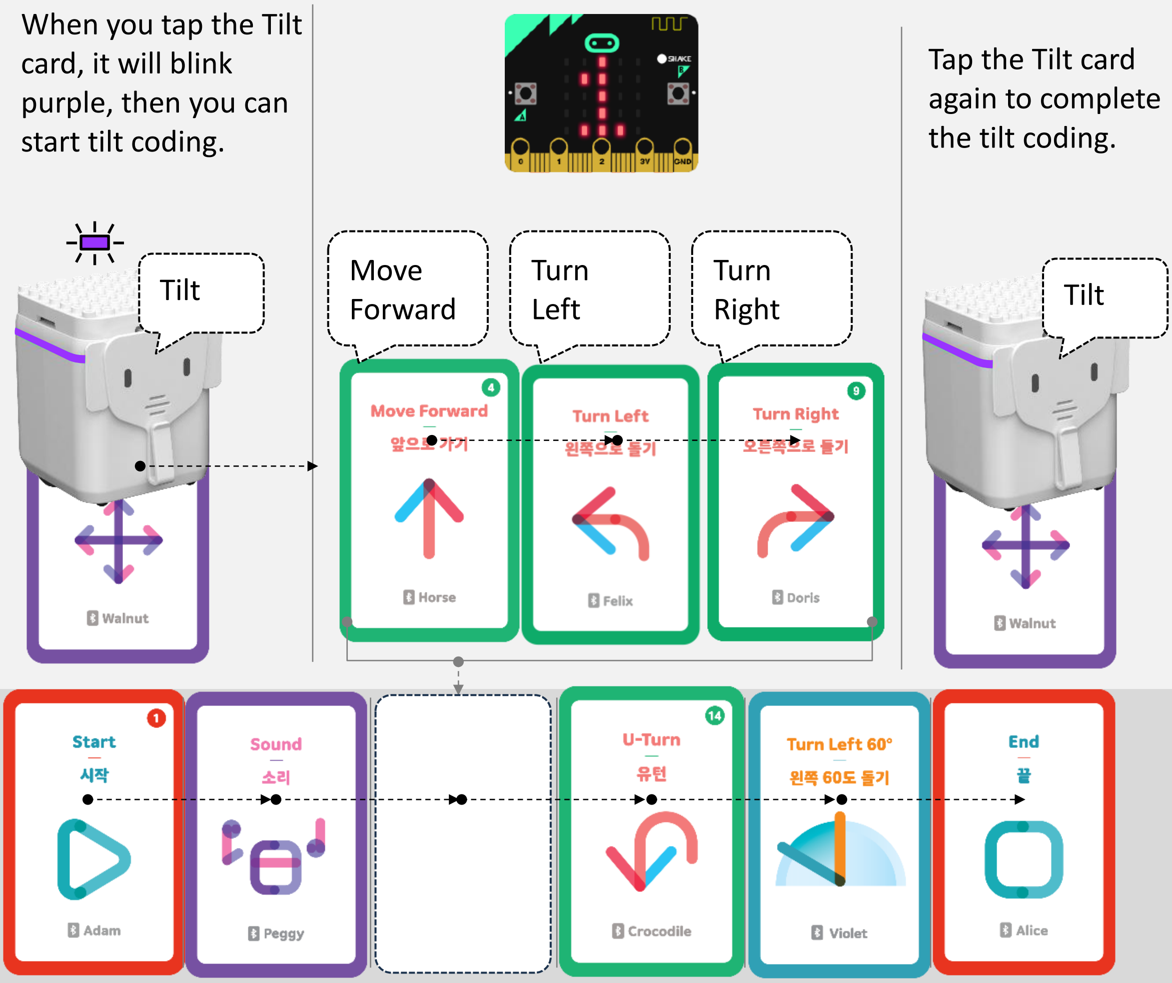
Press the button of the robot.



While the blue light flashes every 0.5 seconds, tap two number cards designated by the radio group number from 01 to 80.

Press the button again to start listening to the BBC micro:bit.

5. Once you have completed the radio settings for the BBC micro:bit and GENIBOT as above, you can do tilt coding using the micro:bit's gestures. The tilt cards that correspond to micro:bit gestures are as follows: Move Forward = “logo up”, Move Backward = “logo down”, Turn Left = “tilt left”, Turn Right = “tilt right”.



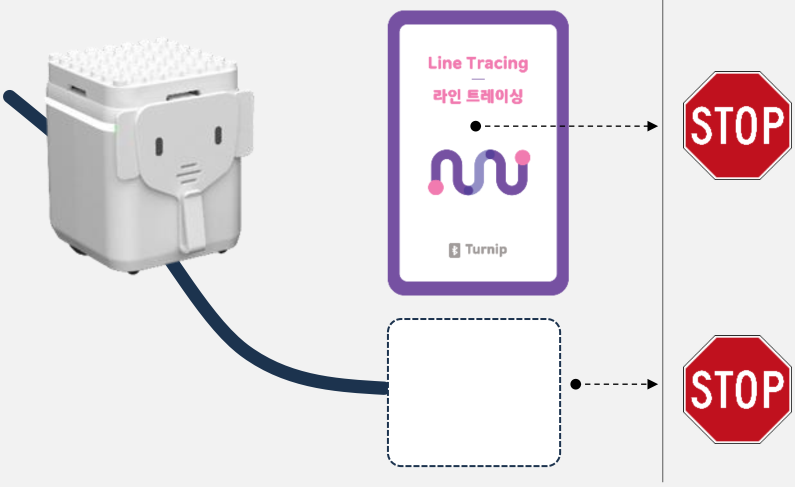
Line Following and Coding

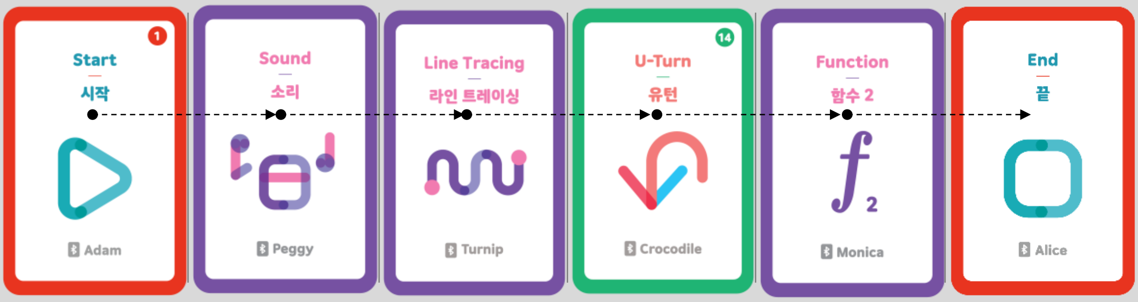
The robot can track straight lines or curves painted in various colors, including black, gray, blue, red, and green. The line width that can be followed varies from 4mm to 6mm. The maximum thickness of the line is 8 mm.

Add the Line Tracing card to coding dataset

When the robot encounters the Line Tracing card in the dataset, it not only traces colored lines or curves, but also stops line following when it detects only a cool white background with no lines and an average color temperature of 5500K to 6500K. Alternatively, while line following, tapping the Line Tracing card will stop line following. After line following stops, the execution order in the dataset increases to the next card.

If there is no line or the Line Tracing card is tapped, the robot stops.





Light Sensor Calibration

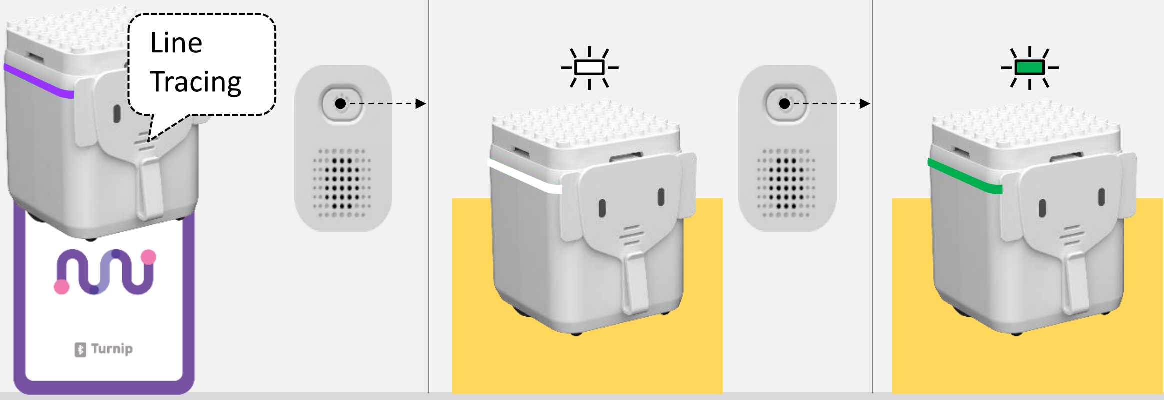
When the robot follows a black line, it can trace lines drawn on various colored papers, and the luminance (brightness) of the color must be above 0.7 (70%) for the robot to stop at the end of the black line. Luminance refers to brightness and is scaled from 0 to 1.0 (100%) in the HSL color space.
To make the robot stop when the lines are broken, or when there are no lines and only a background color, and the background color is not a cool white color, you need to calibrate the brightness of the paper color.

Calibrating the light sensor related to paper color brightness
If you tap the Line Tracing card and press the button within 2 seconds, the robot will blink white. Place the robot on the colored paper and press the button again to start brightness calibration. If the brightness calibration is successful, the robot will blink green and then restart. If the paper color changes, recalibrate the light sensor using the same procedure.

Tap the Line Tracing card, then press the button within 2s.

Place the robot to on the paper, then press the button

The robot will blink green then restart.

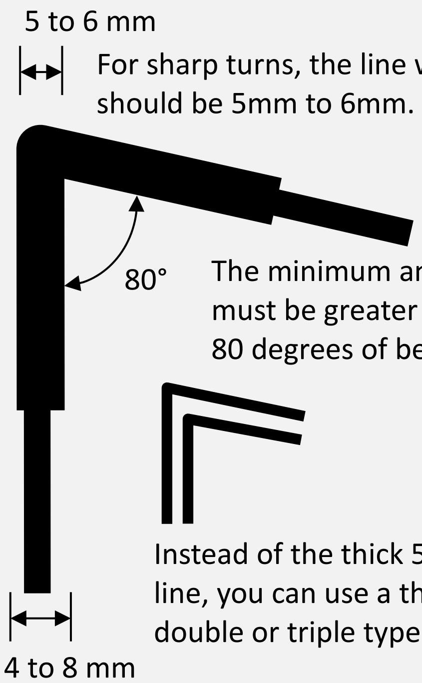


Four light sensors are located on the bottom of the robot. If light sensor calibration does not work, it is because the saturation of the paper color is low. If line tracing does not work well even after light calibration, repeat twice or more.

4 light sensors

Following a Curved or Split Path

If the path is sharply curved or separated like a forked path, the robot can follow curved paths with angles greater than at least 80 degree bends, small or large circles, or hairpin-shaped paths. The line width is 4mm to 8mm when following a straight line, and 5mm to 6mm when following a curved path.

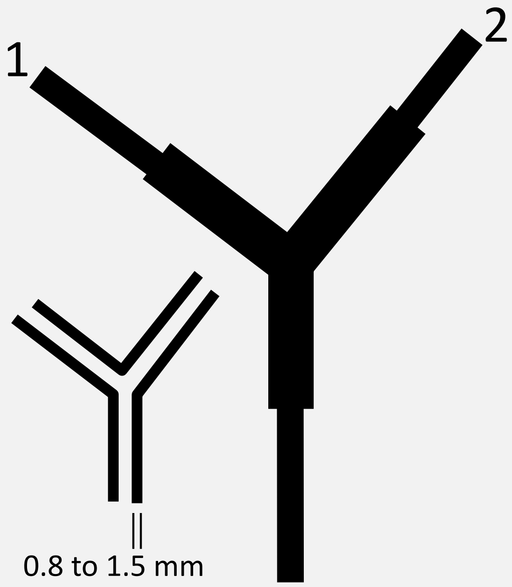


For sharp turns, the line width should be 5mm to 6mm.

The minimum angle must be greater than 80 degrees of bend.

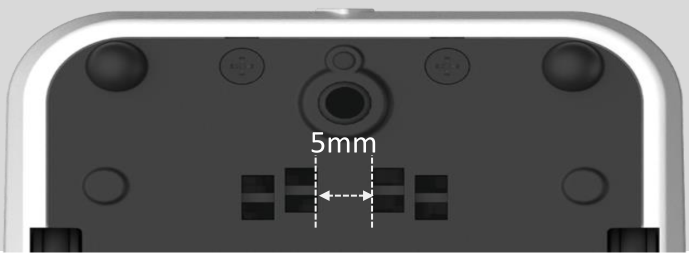
Instead of the thick 5mm line, you can use a thin double or triple type line.

The robot will turn left 1 or right 2 at Y-junction.



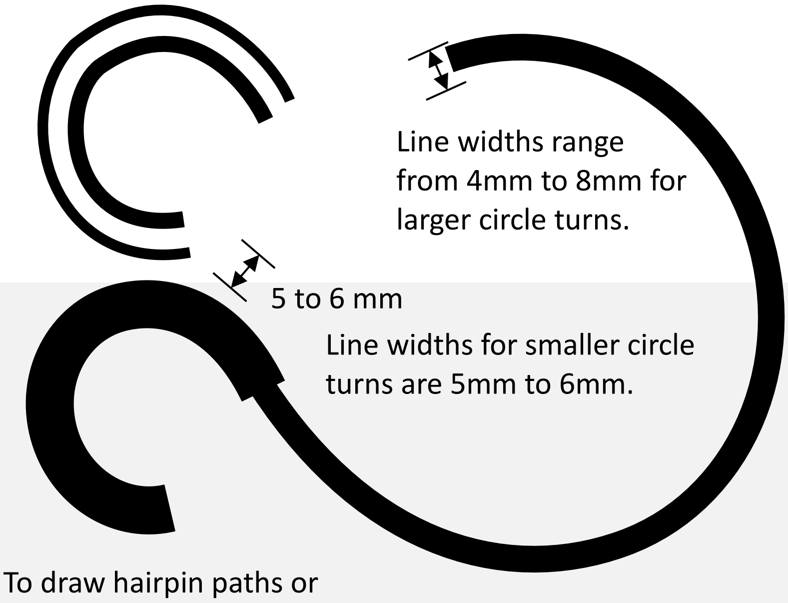
At T-shaped intersections, plus (+) intersections, and three-way intersections, the robot goes straight without turning left or right.

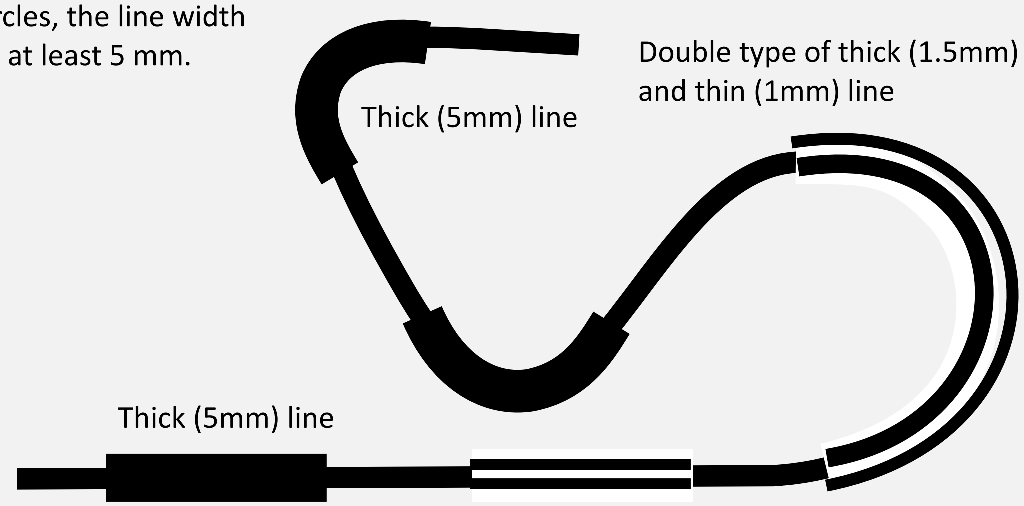
To move in a straight line, the line width should be between 4mm and 8mm. Among the four light sensors, the middle two light sensors are spaced 5mm apart, so the 5mm line width is the average value.

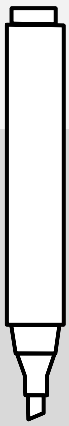


Example 1: Width of the Line

When using a drawing pen, instead of a thick 5 mm line, a thin or thick line of the double or triple type is useful.







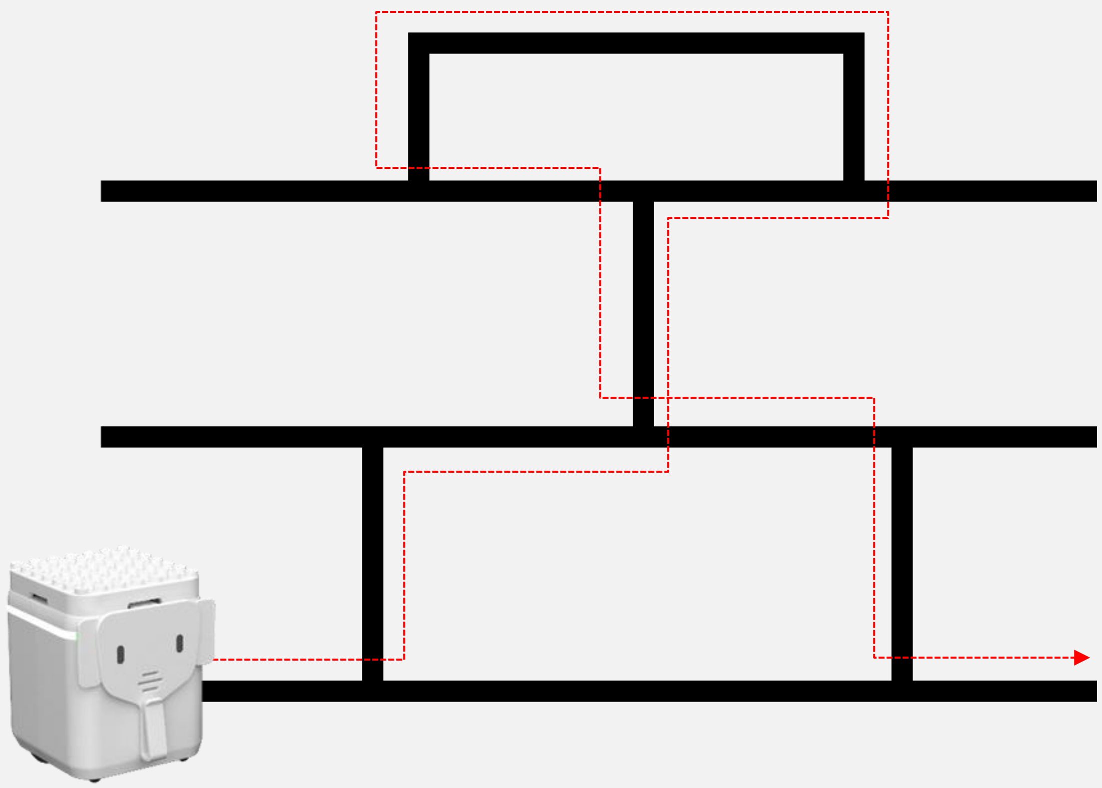
On thin lines the robot will not follow the path, so a 5mm chisel tip marker must be used. However, instead of a thick 5mm line, double or triple thin-thin or thin-thick lines are useful. The minimum width of the thin line is 0.8 to 1.5 mm, allowing you to draw thin-thin lines.

Chisel-flat tip marker

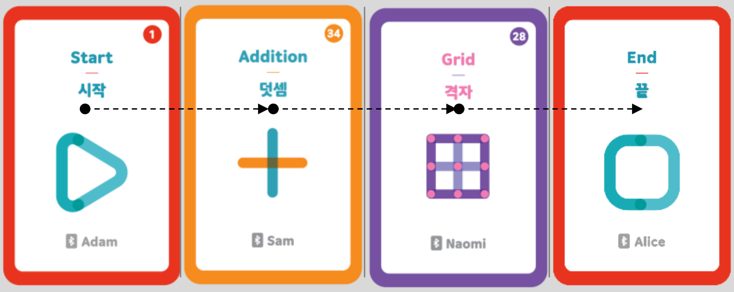
Example 2: Ladder Game

How does the robot know the direction and find the way down? Let's explore the algorithmic principles of direction finding.

There are two conditions: The intersection must be at a orthogonal rectangular angles, and the roads running around the intersection to the sides must be rectangular, not circular. A ladder path can be a straight or curved path excluding any wedge-shaped bends or hairpin turns.



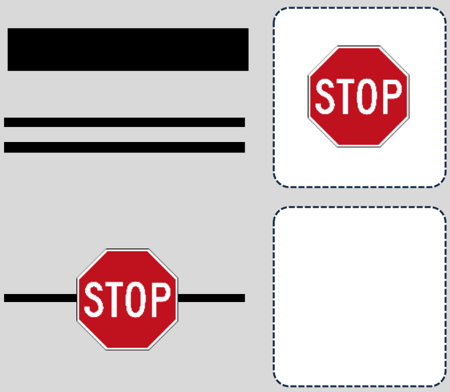
To start a ladder game, tap the Start card, Plus or Minus card, Grid card, and End card in that order.



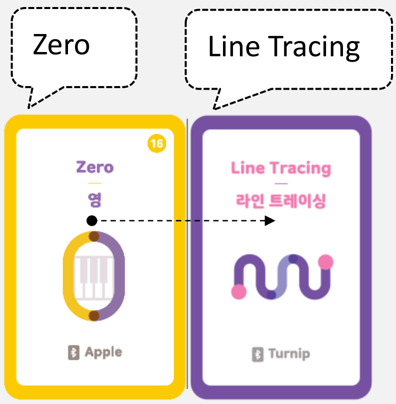
Following a Thin Line

If you want to follow lines thinner than 4mm, you will need to disable stopping at the end of the line. Tap the number 0 card, then tap the Line Tracing card. The robot can follow a thin line and will not stop when the line ends.
The scientific principle of following a thin line is that it is less affected by the thickness of the line because it distinguishes the difference between the line and background color and moves along the largest value of the normal distribution curve calculated from the measurement value of the light sensor. For this purpose, the PID (proportional-integral-derivative) algorithm is used as an engineering method.

The robot follows a line consisting of a thick line or two thin-thin lines, and stops at the end of the line by detecting the brightness of the paper color when the line breaks or ends.



It is less affected by the thickness of the line and the robot follows thin lines with a line width of at least 2mm, but it does not stop when the line is broken or meets the end of the line.

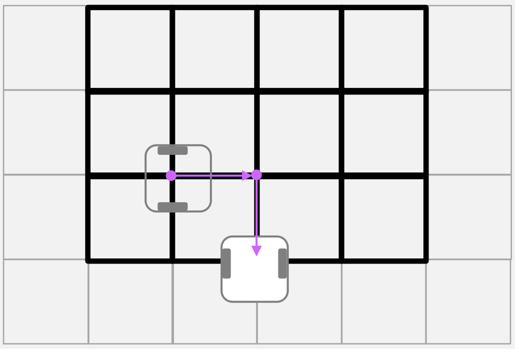


Create Grid Motion

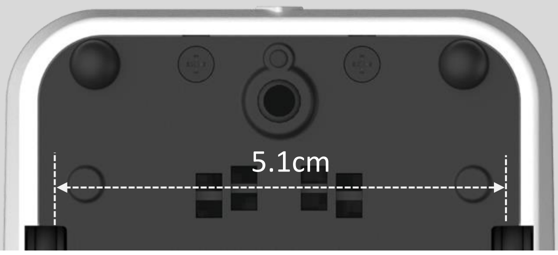
To create motion on orthogonal grid paper, add the Grid card to your coding dataset. The robot moves forward along the grid lines and stops when it detects a three or four way intersection. In grid motion coding, the Move Backward card is not available.

Grid line width and spacing

Grid line width should be 4mm to 6mm. The minimum distance the robot must travel from one intersection to the next must be greater than 5cm. In grid motion, the robot does not stop at the end of a line or at a line break, but only at intersections.



Since the gap between the two wheels of the robot is 5.1cm, grid spacing also requires more than this length.

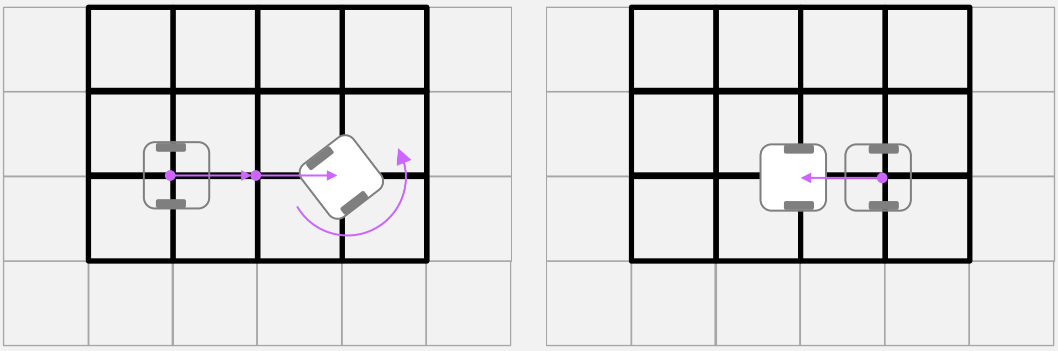


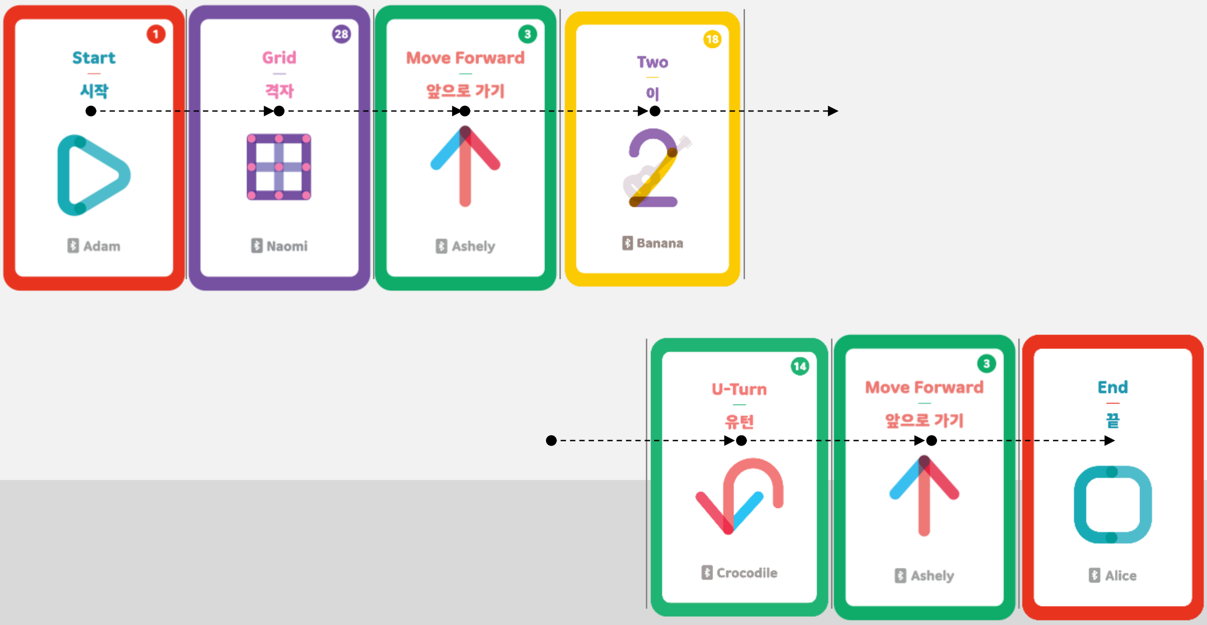
Example 1: Go Back and Borth on the Grid

The grid does not allow the Move Backward card, so instead of having your robot take a step backwards, you can use the U-turn card to change the robot's direction backwards. It always moves in the direction the robot is facing, and never moves in the opposite direction the robot is facing.

Line tracing on grid lines

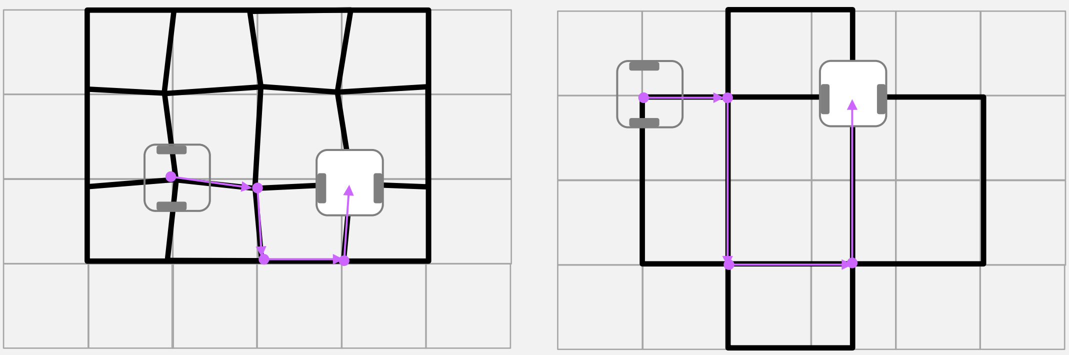
Because the robot performs line tracing on grid lines, the maximum travel distance between intersections is not fixed, and curves as well as straight lines are possible.

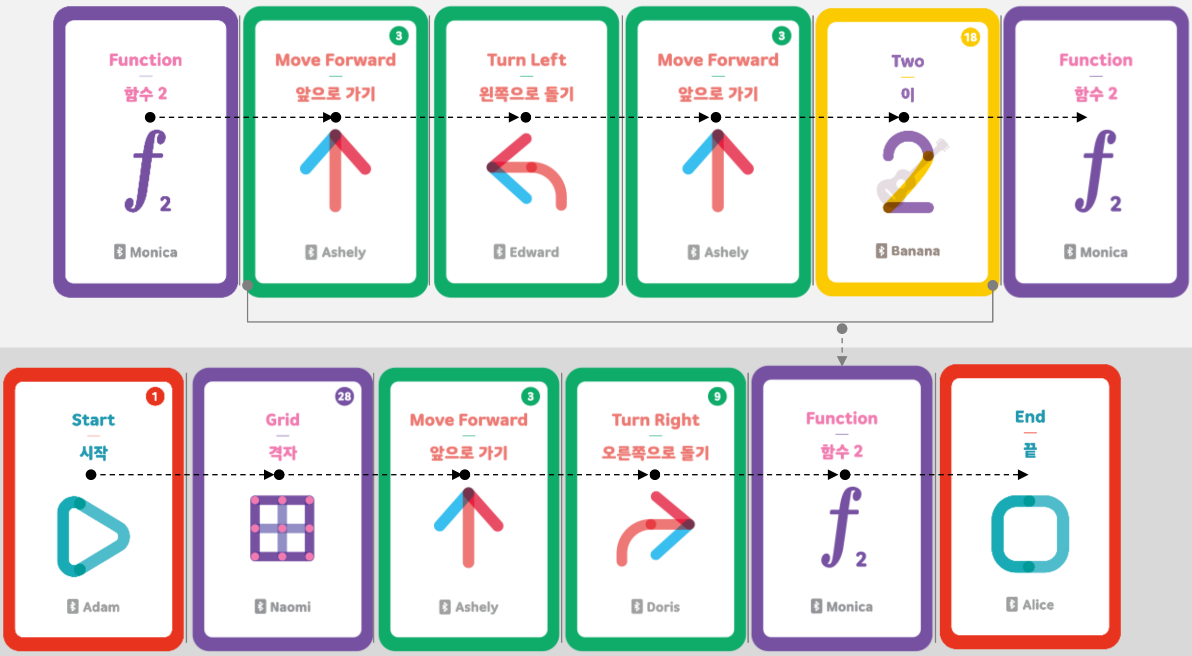




Example 2: Orthogonal and Distorted Grids

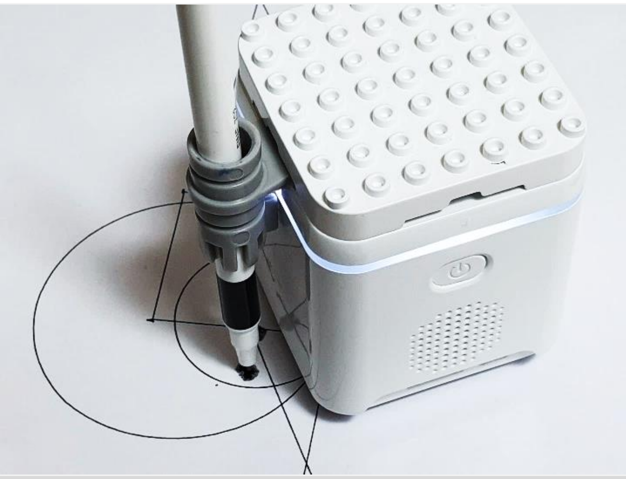
When grid lines meet at orthogonal intersections less than 90 degrees, grid motion is possible as long as the angle is not sharp and wedge-shaped, as shown in the picture. The grid coding method is the same as the previously guided coding method, but since the robot moves by line tracing on the grid line, the distance the robot moves by the Move Forward card is not determined, and the robot continues to move until it meets the next intersection.





DRAWING

Geometric Shape Drawing Activity



?

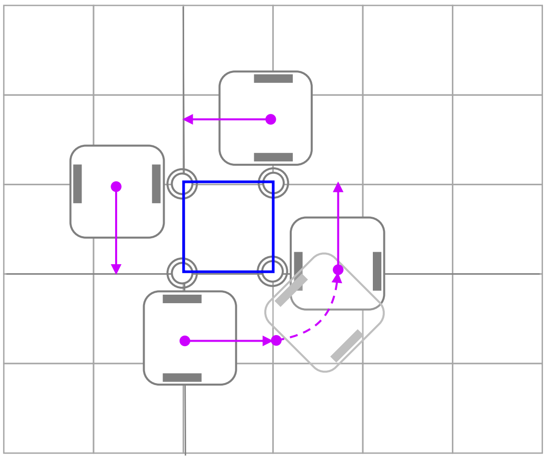
Shape drawing error

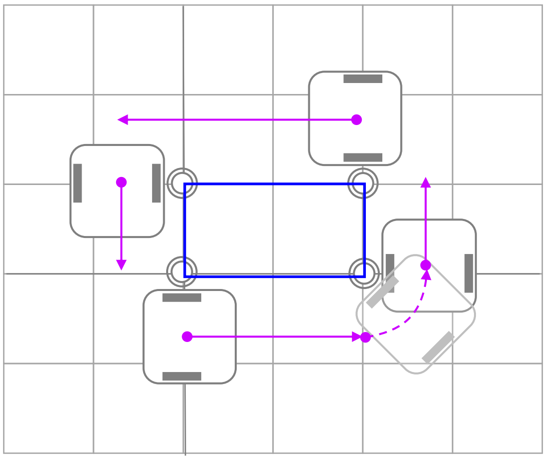
When the robot moves and draws a circle, triangle, or square, the shape may not be drawn accurately due to systematic errors, friction errors between the pen and the paper, etc.

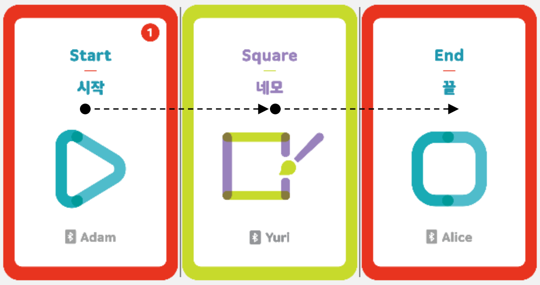
Uncertainty errors arising from the distance between the two wheels, step motor gear ratio, step motor rotation angle resolution, step motor gear backlash error, and pen holder affect the drawing of the shape.

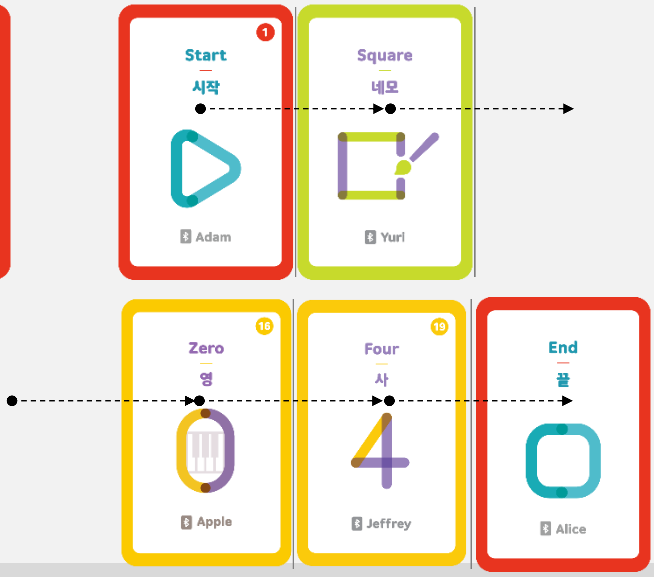
Drawing a Square

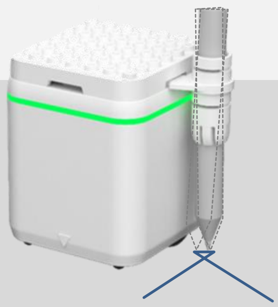
Not only can you draw a square using the Square card, you can also draw a rectangle by adding two number cards after the Square card. The two number cards determine the width and height of the rectangle. The default width and height of shape drawn by the Square card is 4cm.







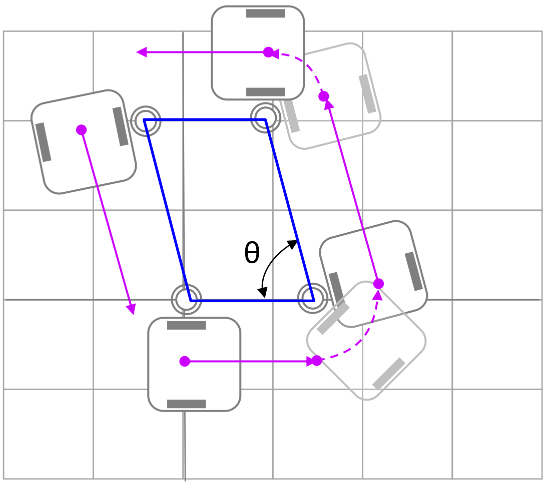




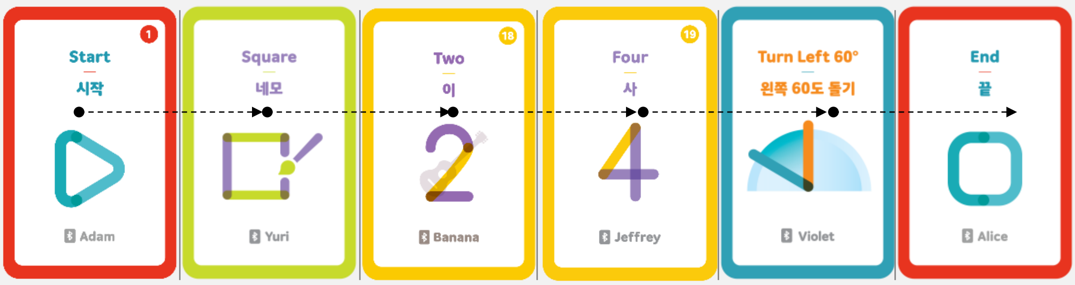
There are unwanted uncertainty errors in drawing a rectangle due to systematic and assembly errors in the robot, pen holder, gear motor, and resistance between the pen and paper.

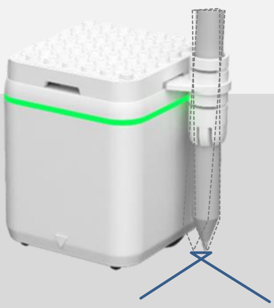
Drawing a Parallelogram

You can draw a parallelogram by adding two number cards to the Square card and a left turn angle card such as Turn Lef 15°, Turn Left 30°, Turn Left or 60°. The two number cards determine the base and leg lengths of the parallelogram, and the left turn angle card determines the acute angle θ. The default width and height of shape drawn by the Square card is 4cm.



Base (Square 4cm + Two 2cm) = 6cm
Leg (Square 4cm + Four 4cm) = 8cm
Acute angle (Turn Lef 60°) = 60°

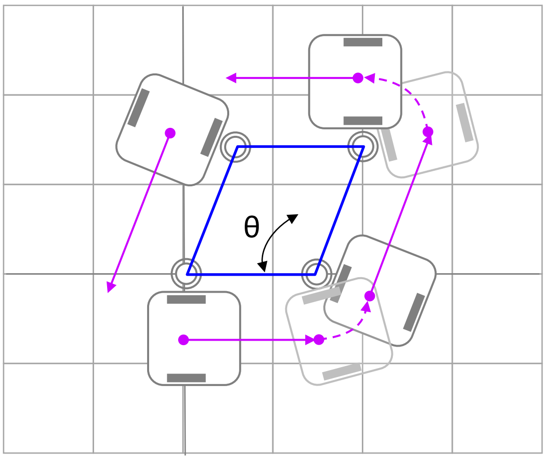




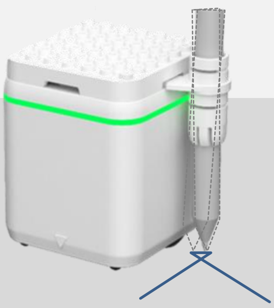
There are unwanted uncertainty errors in drawing parallelogram due to systematic and assembly errors in the robot, pen holder, gear motor, and resistance between the pen and paper.

Drawing a Rhombus

You can draw a Rhombus by adding a left turn angle card such as Turn Left 15°, Turn Left 30°, Turn Left, or 60° to the Square card, and then add two number cards. The two number cards determine the base length and leg length of the parallelogram, and the left angle card determines the acute angle θ. If two numbers are equal, the parallelogram becomes a rhombus. The default width and height of shapes drawn by the Square card is 4 cm.



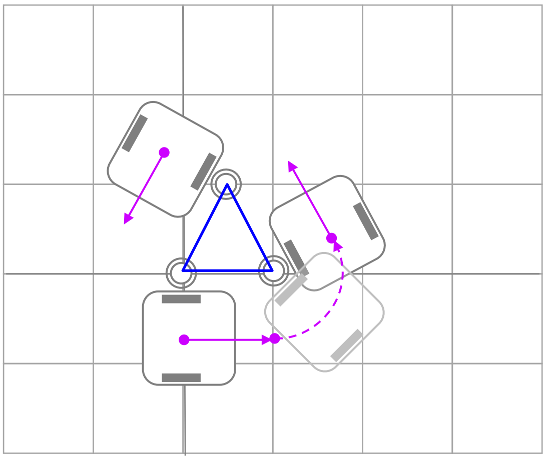
Base (Square 4cm + Two 4cm) = 8cm
Leg (Square 4cm + Four 4cm) = 8cm
Acute angle (Turn Lef 60°) = 60°

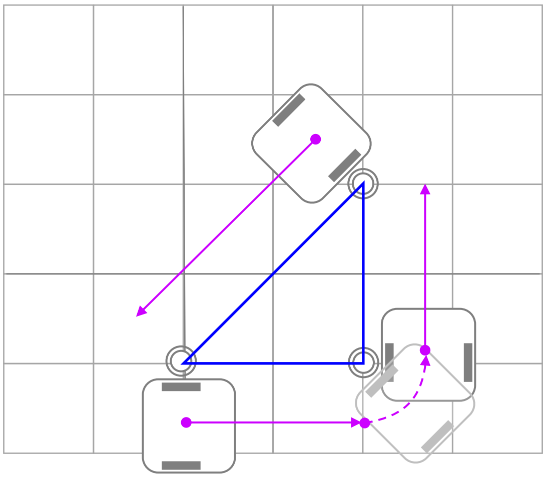


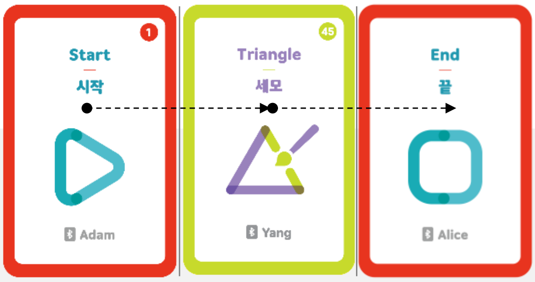
There are unwanted uncertainty errors in drawing parallelogram due to systematic and assembly errors in the robot, pen holder, gear motor, and resistance between the pen and paper.

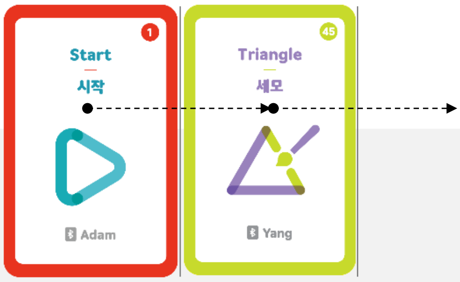
Drawing a Triangle

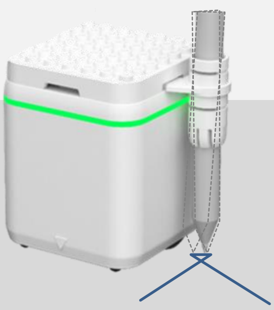
Not only can you draw a triangle using the Triangle card, you can also draw a right triangle by adding two number cards to the Triangle card and Turn Left 90°. The two number cards determine the base and leg lengths of the triangle. The default base and leg of shape drawn by the Triangle card is 4cm.







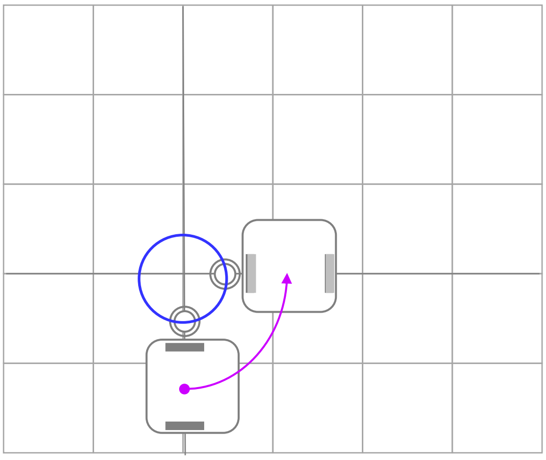


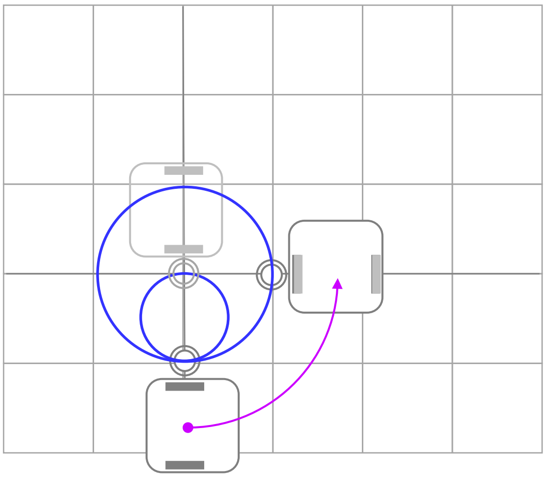


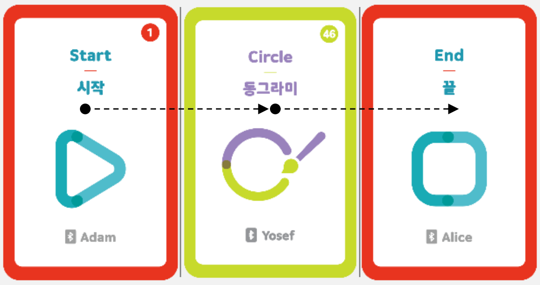
There are unwanted uncertainty errors in drawing a triangle due to systematic and assembly errors in the robot, pen holder, gear motor, and resistance between the pen and paper.

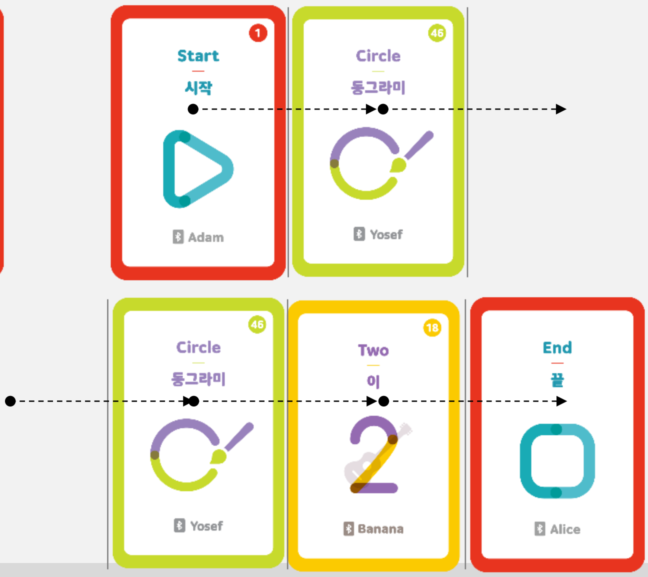
Drawing a Circle

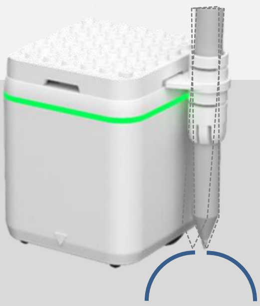
Not only can you draw a circle using the Circle card, you can also increase a radius of the circle by adding one number card after the Circle card. The one number card determine the radius of the circle. The default radius of shape drawn by the Circle card is 2cm.







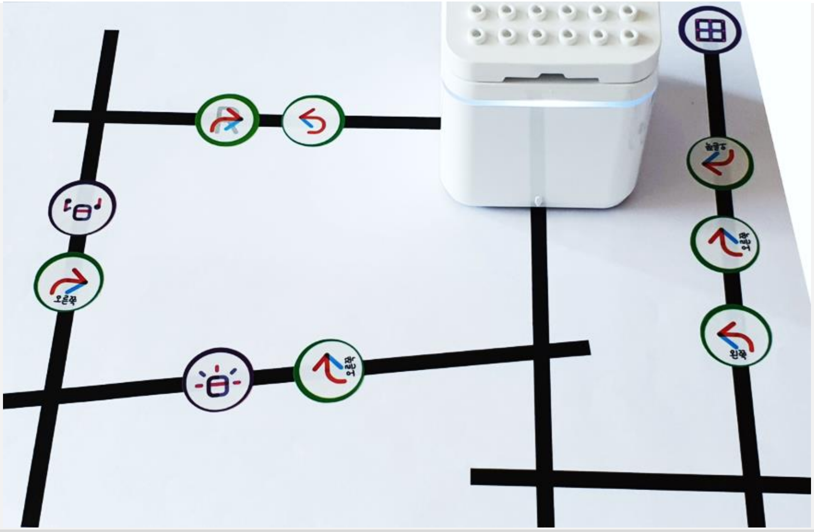




There are unwanted uncertainty errors in drawing a circle due to systematic and assembly errors in the robot, pen holder, gear motor, and resistance between the pen and paper.

STICKER CODING

Advanced Coding Activity



OID sticker coding

OID codes of the stickers are equal to the codes of coding cards such as Sound, Light, Move Forward, Move Backward, Turn Left, Turn Right, Pause, Function and Music Notes.

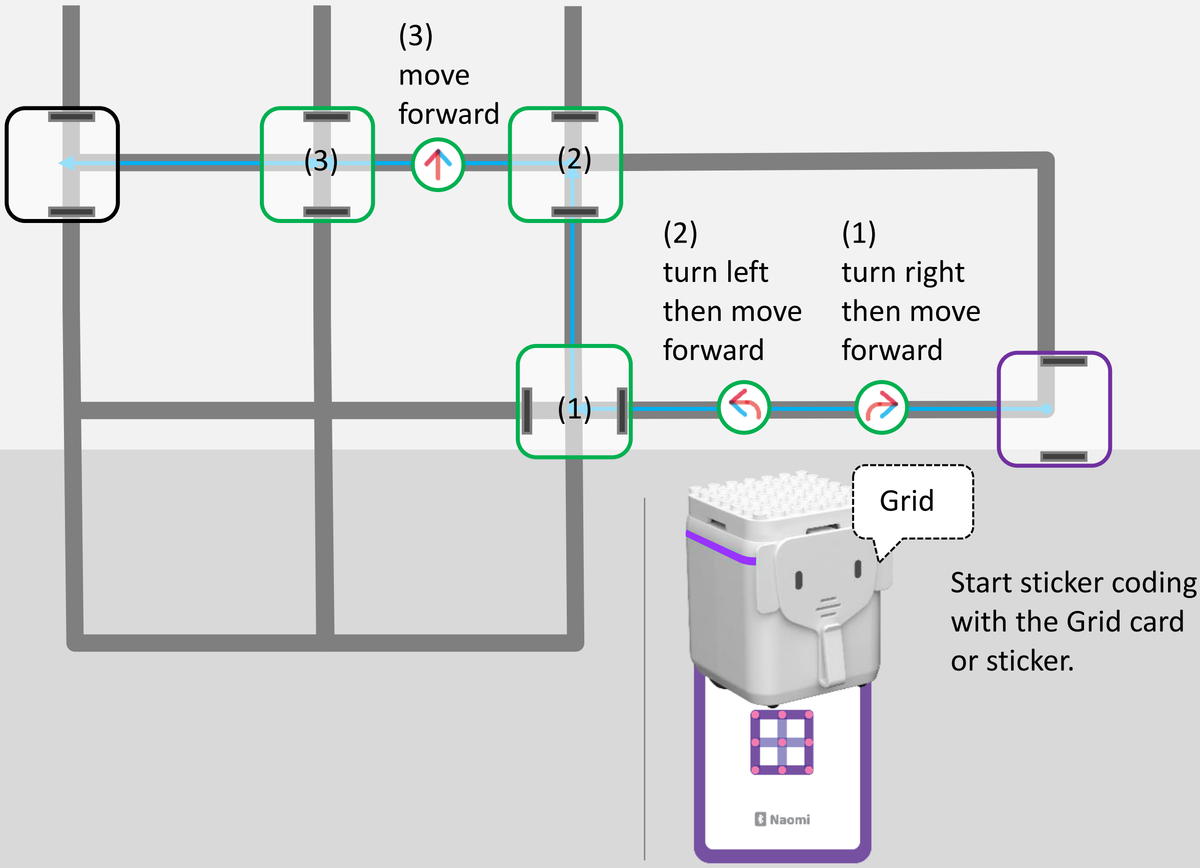
You can create a coding dataset in the same way as card coding. Card coding can run the data set after completing it, but sticker coding is different in that the robot immediately executes the sticker data read during line following.

Start Sticker Coding on the Grid

To create motion on orthogonal grid paper, tap the grid card. Alternatively, you can start by placing the Grid sticker at the starting point. The robot moves forward along the line using line tracing and stops when it detects the Grid sticker again. When the robot encounters stickers while doing line following, they are added to the dataset in order, and becausethe dataset are processed sequentially, the data executed at a grid intersection may be different from the sticker data encountered just before.

Sticker coding on grid lines

To start sticker coding, place the robot on the grid line after tapping the grid card, or on the Grid sticker attached to the grid line. Grid intersection spacing is the same as the card coding on the grid explained previously. Unlike card coding, sticker coding allows the robot to turn and then move forward if there is a left or right turn in the sticker dataset.

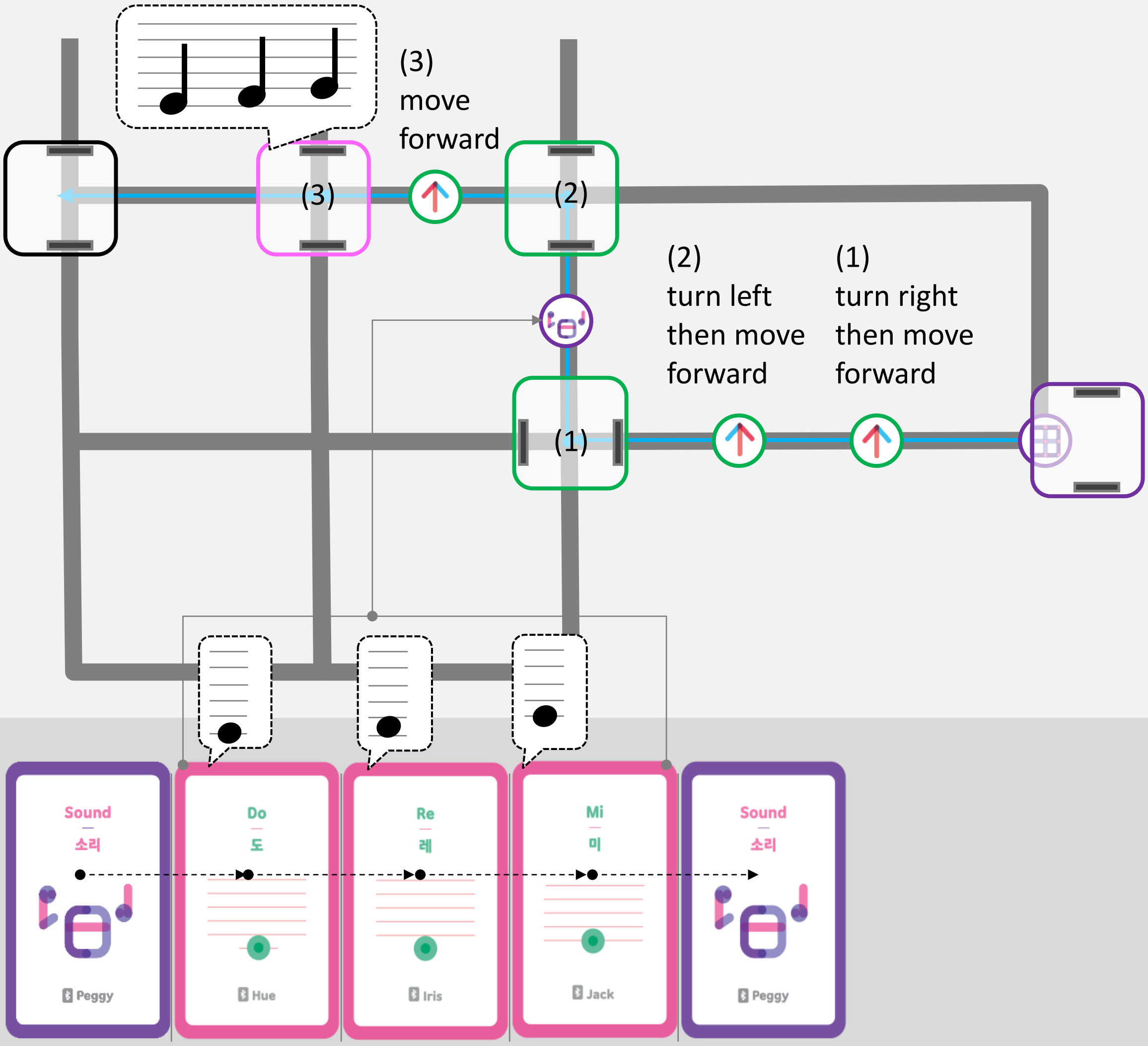


Example 1: Motion and Sound Data

Sticker coding can be performed by complexly mixing various card or sticker data such as motion and sound. For example, when the robot encounters the Sound sticker, the robot will play music previously created with the Sound card. Coding data has a Queue data structure that follows FIFO, so there is no limit to the number of stickers between any two points on the grid.

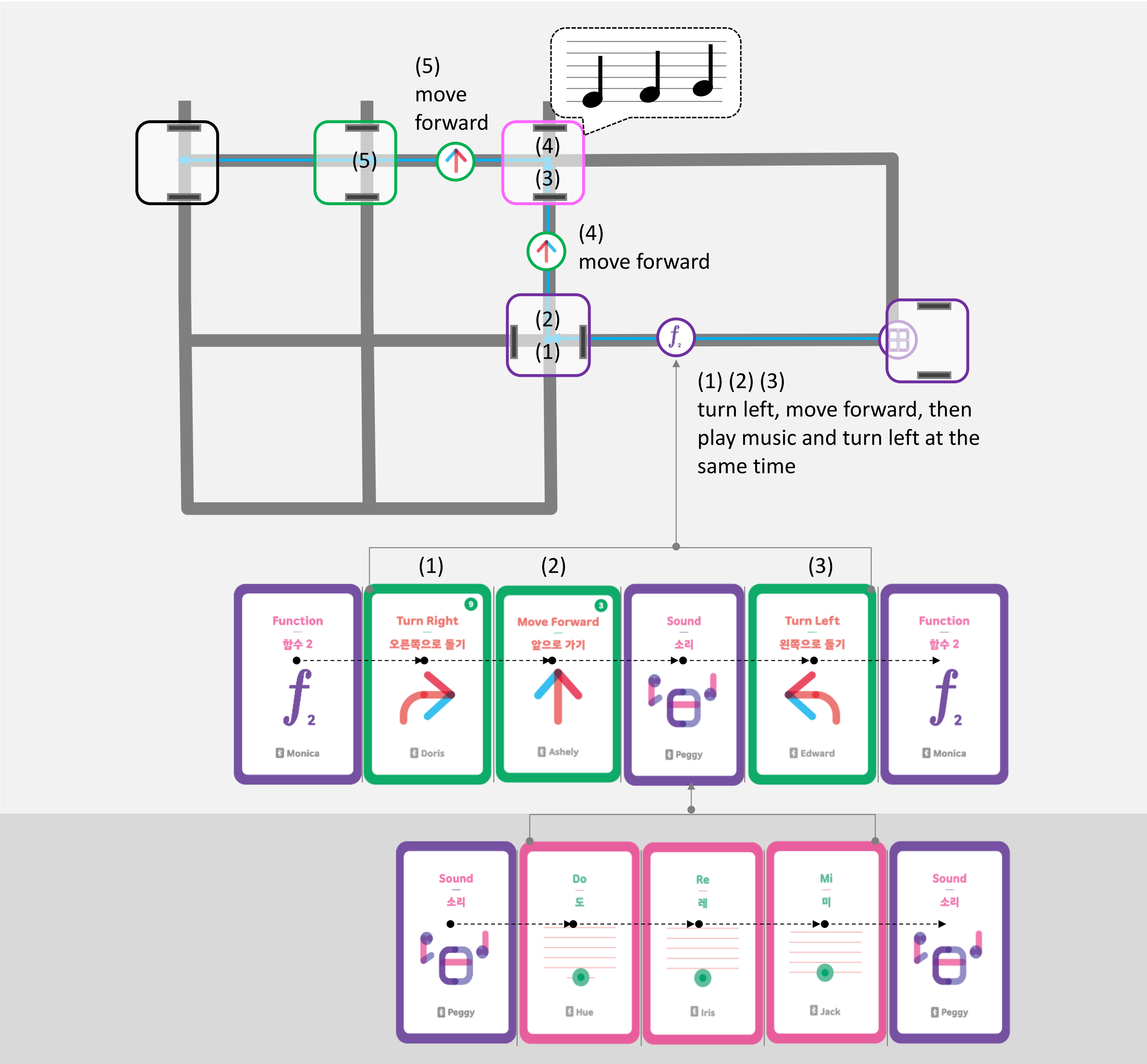
Create sound data

Create a sound dataset by adding a music card between two Sound cards. If you attach the Sound sticker to a grid line, when the robot encounters the Sound sticker during line following, the music previously created using the sound card will be played.



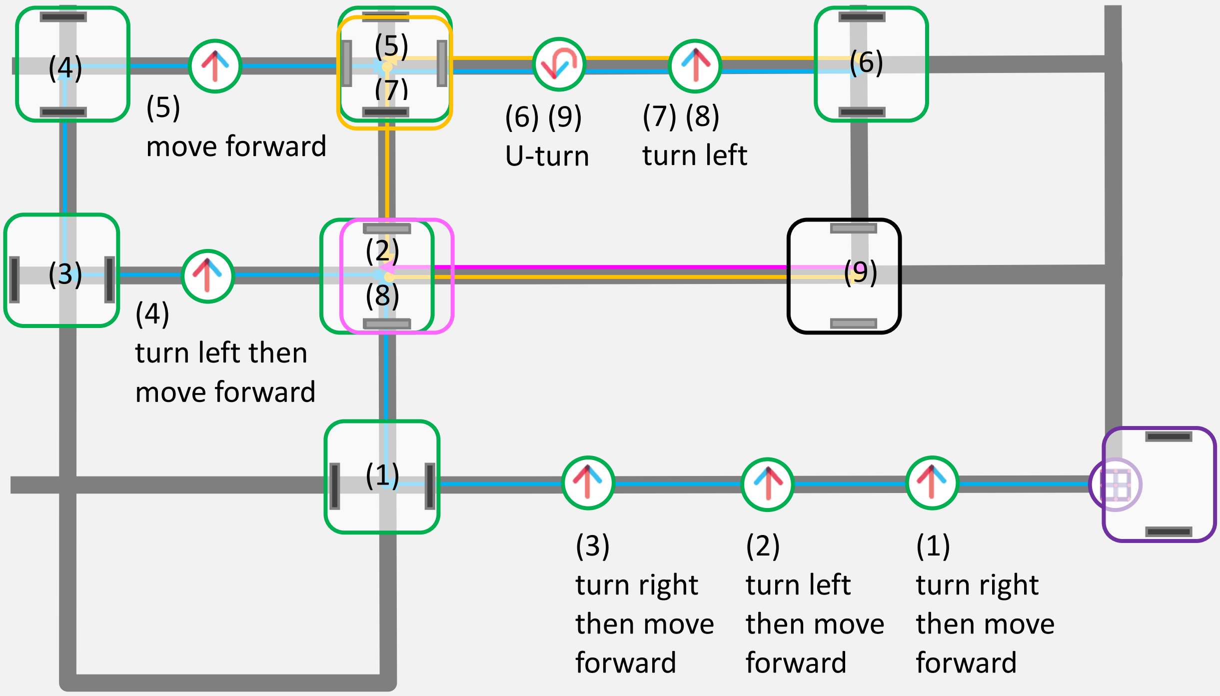
Example 2: Function and Sound Data

Sticker coding complexity using orthogonal grid paper helps improve students' computational thinking skills by mixing coding cards and stickers such as linear or rotatitional motions, functions, sound and music notes, light, etc.

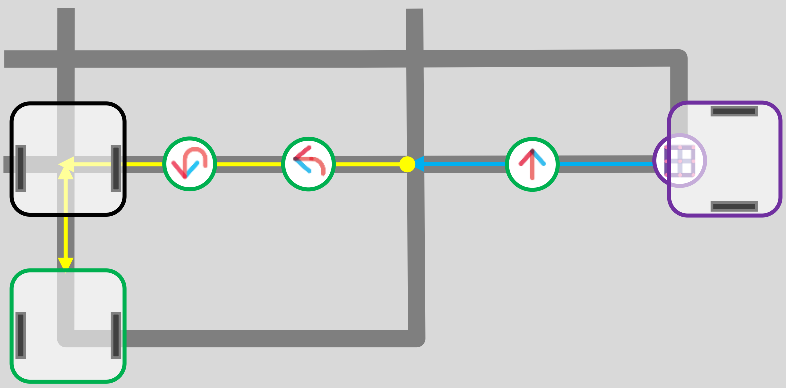


Example 3: Queue Data Structure

This is a sticker coding example of the Queue data structure in which the right order of data input and output is strictly maintained through FIFO (first in, first out) delivery. Note that sticker coding does not cover the standard Queue for scheduling multiple sticker data to be added to the coding dataset.
In the example below, depending on how the U-turn sticker is entered into the Queue dataset, the robot may or may not fall into an infinite loop.



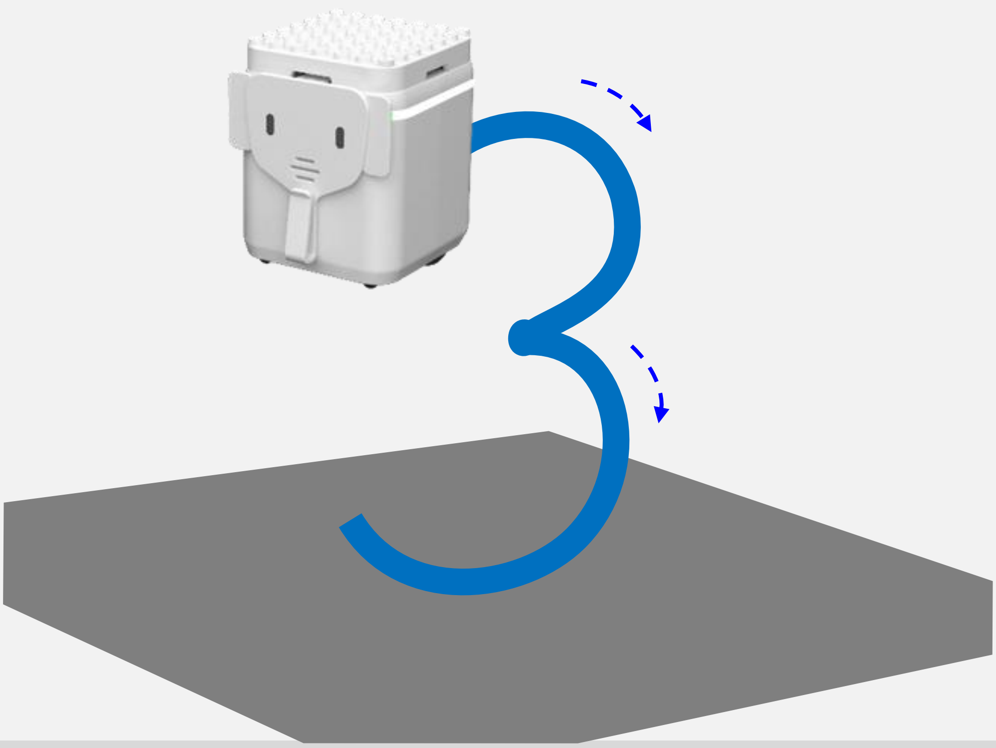
By placing the Left Turn sticker in front of the U-turn sticker, the data input and output Queue allow the robot to break out of the infinite loop created by the U-turn sticker.



AI demonstration is not possible for models without an acceleration sensor.

AI DEMONSTRATION

Air Gesture Classification using Machine Learning

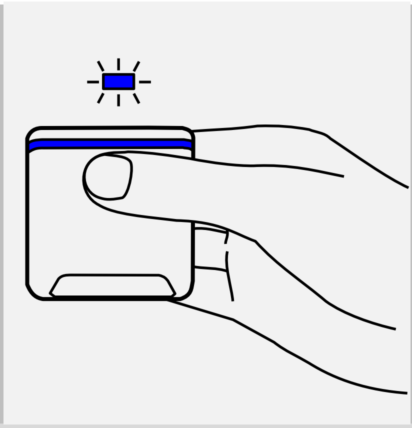


Embedded AI machine learning

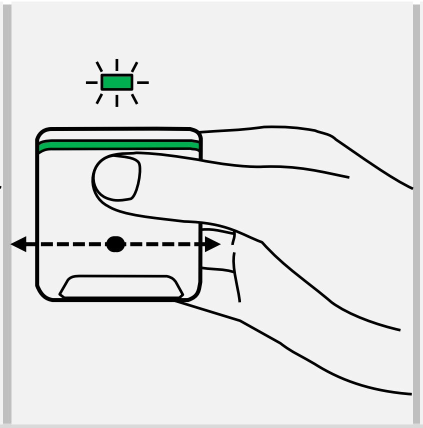
The robot can classify air gestures from 0 to 9. Firmware version 1.0.20 added AI machine learning algorithm and air gesture classification model. This feature is not available in the GENIBOT Plus version (Model D) without an acceleration sensor.

AI Machine Learning Preparation

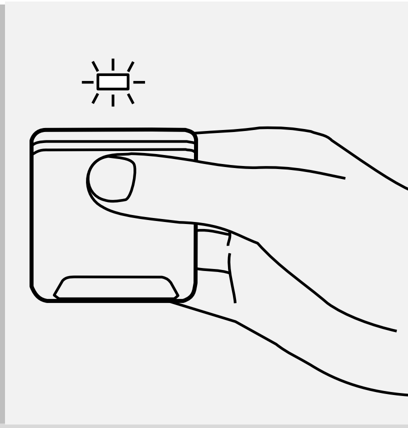
To start air gesture classification, prepare your robot to launch the AI machine learning algorithm and apply the model to the robot before real-time classification. Press the button and shake it left and right quickly at least twice while the blue light is blinking. When you shake the robot, the blinking color changes to green. Keep the robot in a normal steady state while the green is still flashing. Finally, the AI machine learning is ready and white will flash every 0.5 seconds.



Press the button.
Blue flashes every 0.5 seconds.



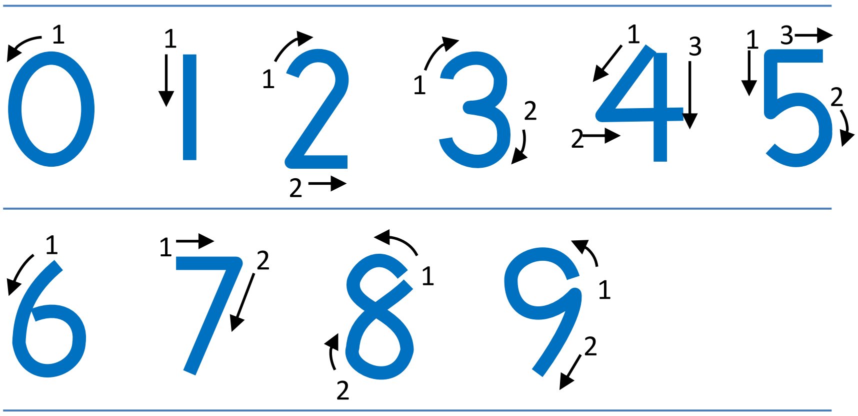
Shake the robot from left to right or right to left as quickly as possible three or more times. Green blinks every 0.5 seconds.

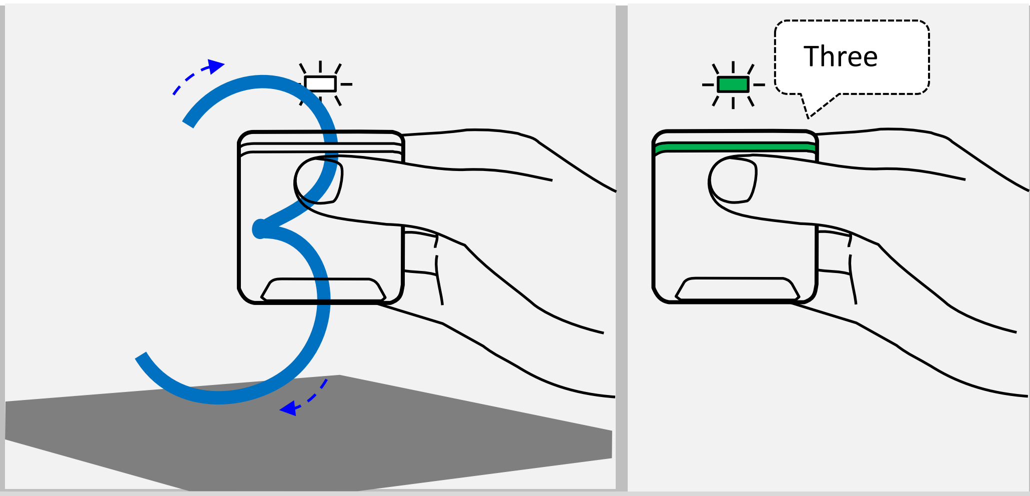


The robot maintains stability by compensating for the 3-axis offset of the acceleration sensor. When calibration is complete, it will blink white and your air gestures are ready to be classified.

AI Gesture Classification

Air gestures are determined by numbers from 0 to 9. To classify air gestures using a pre-trained machine learning model, simply hold the robot in its normal state and draw a number within 2 seconds. After drawing, you can press the button and get the classification results immediately. The robot blinks green twice and speaks the number along with real-time classification.





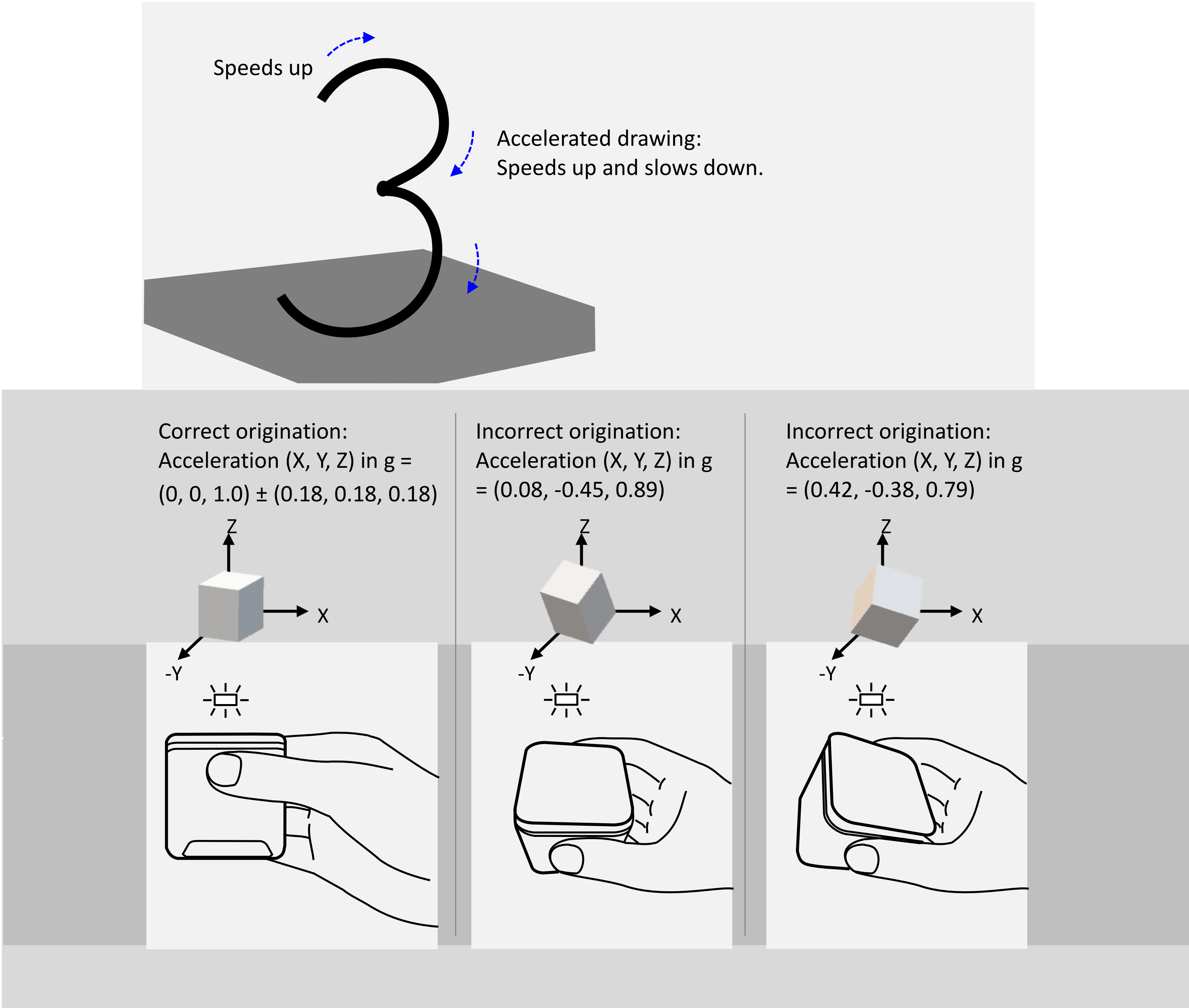
Maintain your stability within 2 seconds and quickly draw the numbers as if you were handwriting.

After drawing a number, you can get immediate classification results by pressing a button. After two short green flashes you will hear the classification number.

While real-time classification is running, 3-axis acceleration data is continuously collected. Keep the robot in a normal steady state before starting to draw numbers.

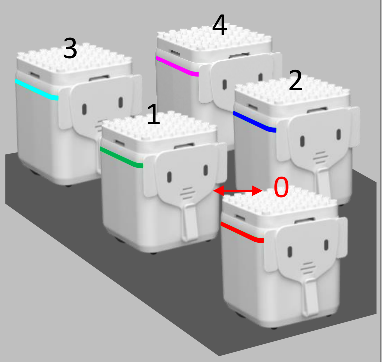
Example: Drawing Numbers in Cartesian Coordinates

Keep the robot at the origin of the Cartesian coordinate system (x = 0, y = 0, z = 1) when performing an “accelerated drawing motion” within 2 seconds. This is because a pre-trained machine learning model was created using a 3-axis acceleration sensor derived from orthogonal coordinates.

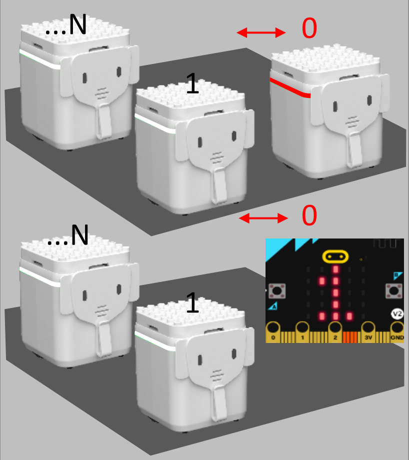


GENIBOT STAR NETWORK

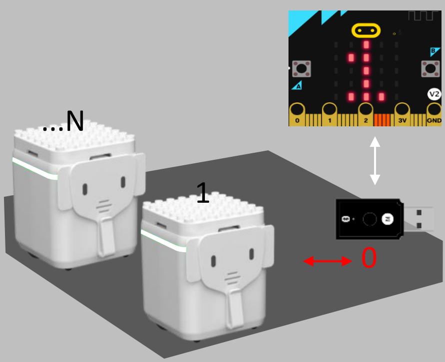
Easy Multilink Demonstration



Unplugged via Bluetooth LE 5 multiprotocol.
Total number of links count is 5



Unplugged via ISM Band Radio Protocol.
The total number of links is theoretically unlimited, but in practice it is affected by ambient noise.

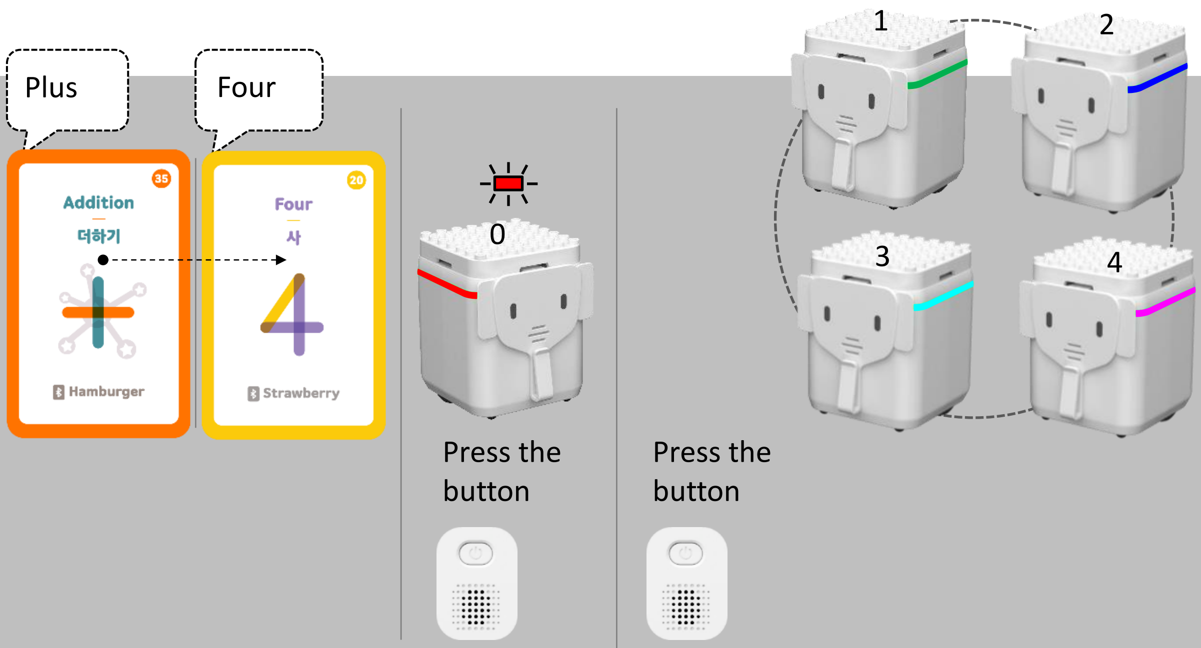


Connected to the computer via ISM Band Radio Protocol and Bluetooth LE 5 multiprotocol.
The total number of links is theoretically unlimited, but in practice it is affected by ambient noise.

Star networking and multilink
Peer robots can automatically connect to the Star robot using the Bluetooth Low Energy 5 multiprotocol or the GENISTICK dongle using the ISM band 2.4GHz ready radio protocol compatible with the BBC micro:bit radio protocol.

Create a Star Network

To create a star robot, tap the star card, which is the Addition (Plus) card, then tap the number card to connect two or more robots. The number card corresponds to the number of connected robots. After creating a star robot, wait for the peer robot to connect. When you press the button on the peer robot, it connects to the star robot. The star robot blinks red at 0.5 second intervals until the number of connecting links equals the number card.
After the connection link is completed, if you press the button on the star robot, the connection link of each robot becomes ready for the star network.

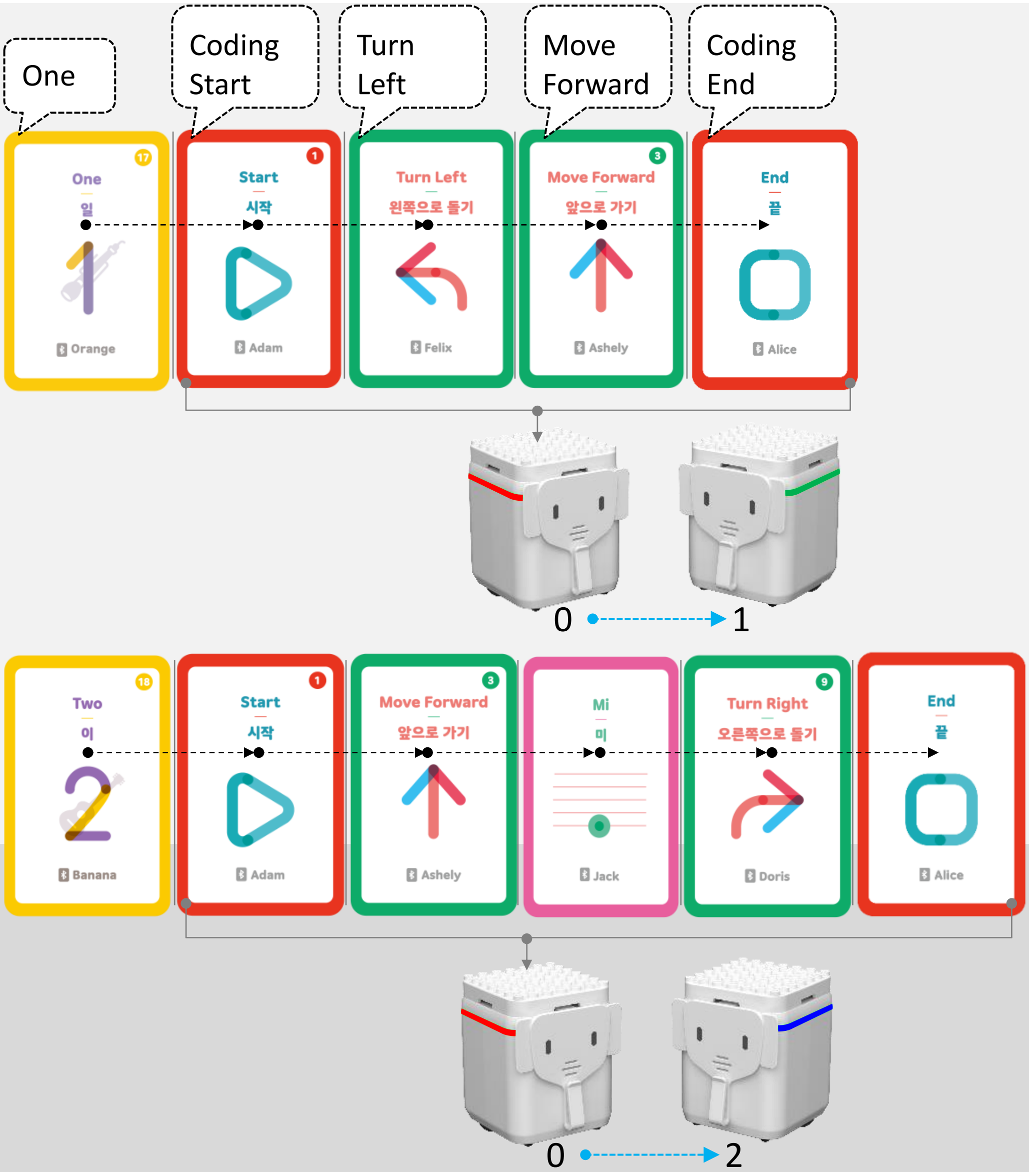


Tap the Addition and number cards to create a star robot. The red light will flash every 0.5 seconds until the connection link is complete.
The number corresponds to the number of connection links, which is the number of peer robots connected to the star robot.

When you press the button on the peer robot, it connects to the star robot. After the connection is complete, when you press the button on the star robot, the star network starts through the links of the connected peer robots.
The link order is indicated by color: Red 0, Green 1, Blue 2, Cyan 3, Magenta 4.

Broadcasting Coding Data

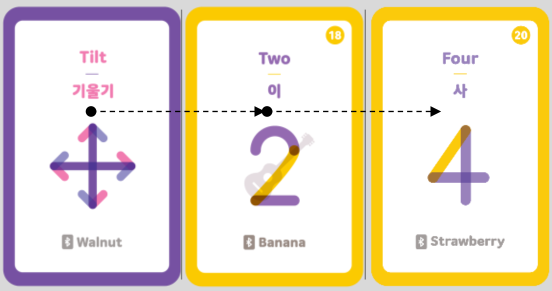
After assigning a robot by tapping a number or star card, you create a coding dataset between the start and stop cards.
To send coding data to each robot, select a number card from 1 to 4 that matches the number of the peer robot, select a star card to broadcast the same data to all robots, or number 0 to create a dataset for star robots only.
When each robot has a coding data set stored, press the button on the star robot to broadcast a message to start the coding dataset for all robots simultaneously.



Advanced: Radio Broadcasting and Listening Demonstration

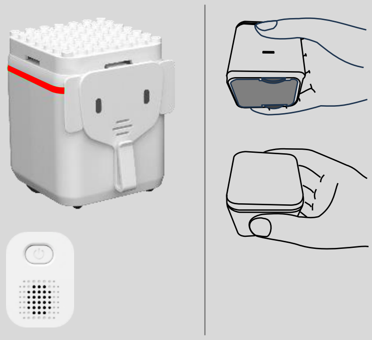
To demonstrate broadcasting and listening to messages using the wireless communication (ISM radio band 2.4GHz, 80 channels) protocol, create a radio group with a star robot and a peer robot. When you tilt the star robot, the star robot broadcasts its tilt status message, and peers immediately listen to it and execute actions according to the tilt status. There is no limit to the number of peers that will listen to broadcast messages.

Press the button of the star robot.

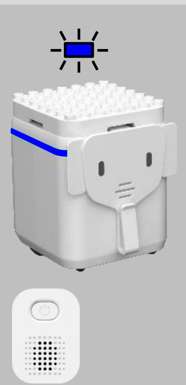


While the blue light flashes every 0.5 seconds, tap the Tilt card and two number cards designated by the radio group number from 01 to 80.

⚠ Tilt is not possible for models without an acceleration sensor.



When you press the button again, the star network group is completed, and the star robot's tilt message continues to be broadcast.



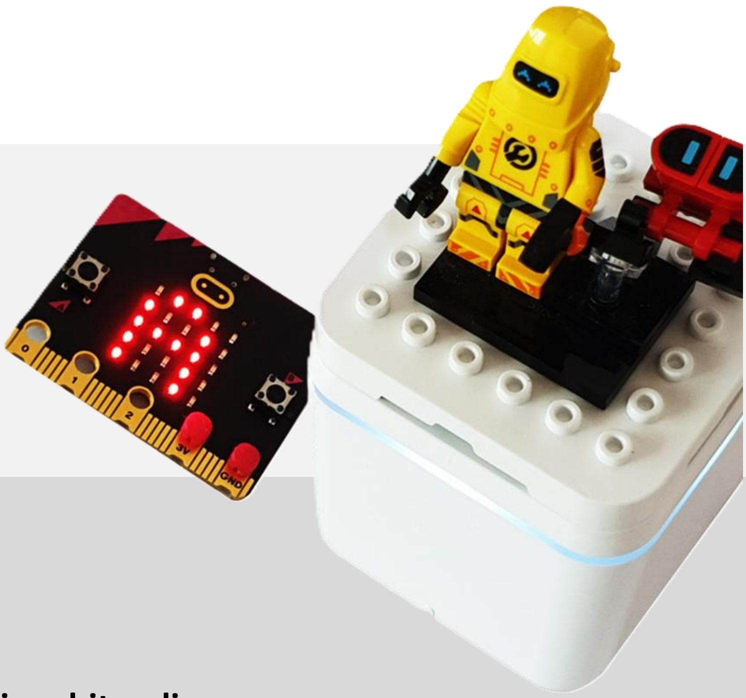
Press the button of the peer robot.

While the blue light flashes every 0.5 seconds, tap the two number cards designated by the radio group number from 01 to 80.

Press the button again to start listening to the star robot’s tilt message.

LISTEN to BBC micro:bit RADIO

BBC micro:bit Radio Programming

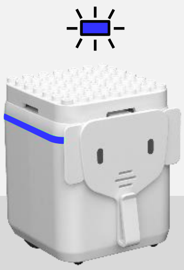


Broadcast or listen to BBC micro:bit radio
The robot can listen directly to the BBC micro:bit via the freely licensed ISM (Industrial, Scientific, Medical) 2.4GHz radio band protocol.
For more information, see the micro:bit-GENIBOT radio programming guide.

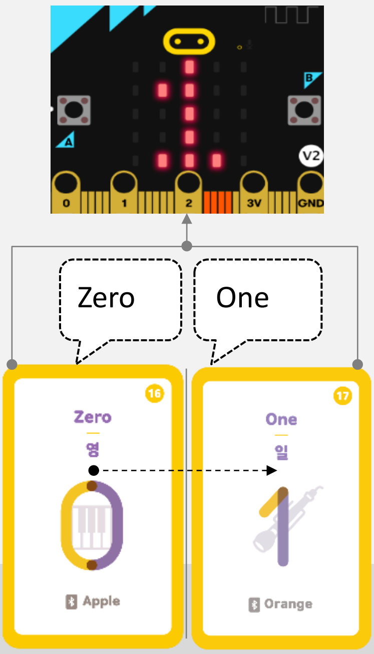
The BBC micro:bit and the robot are not connected to each other with physical wires, and data is exchanged through radio broadcasting and listening.

Setting up Radio Band and Group

There are three ways to set up radio bands and groups. First, as a method of using the coding card, the band number is defaulted to 0 and the group number is set as follows. Press the button to start Bluetooth, then tap the two number cards in numerical order to match the micro:bit radio group numbers 1 to 99. To stop Bluetooth, press the button again. After the blue flashes three times quickly, the robot will start the radio instead of Bluetooth.
The second is to set it with the GENIBOT Android or iOS app, and the third is to set it with Scratch GENIBOT extension.



Press the button to start Bluetooth



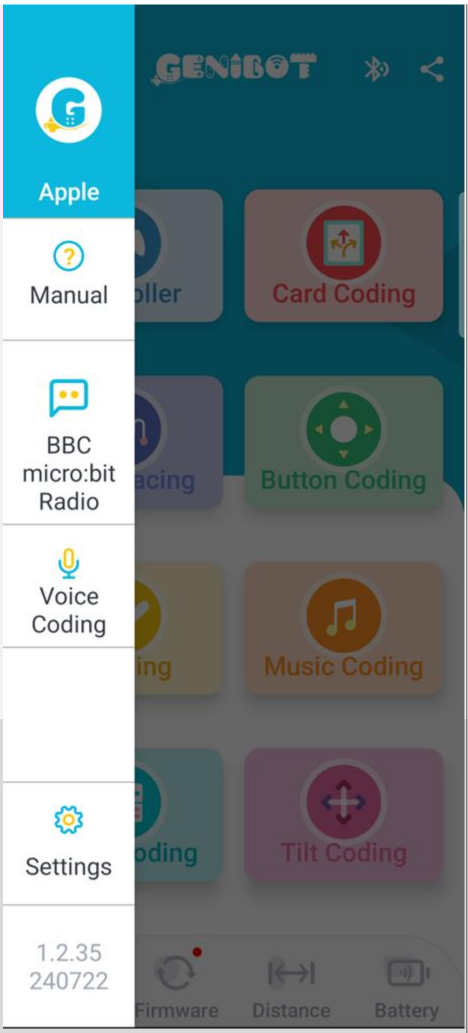
To match BBC micro:bit radio group numbers 1 to 80, tap the two number cards in numerical order. To set the radio group number lower than 10, add a leading zero like two digits. Number (e.g. "01", "04", etc.). If you set up the radio with two number cards, the band number is fixed to 0, and Bluetooth will be disabled and the robot will not be able to connect to a computer or smart device until the robot is rebooted.

Press the button again to stop Bluetooth then start radio.

Example: Set up Radio using the GENIBOT app

Set radio bands and groups using the GENIBOT Android or iOS app as follows. Connect the robot with the app, then click the BBC radio button on the left sliding menu screen, then set the band and group.

You can use the app for robot control, card coding, music coding and other programming while listening to BBC micro:bit. Because the robot can use both Bluetooth and radio simultaneously, this app can send coding data or remote control messages to the robot via Bluetooth while the robot listens to radio messages from the BBC micro:bit.

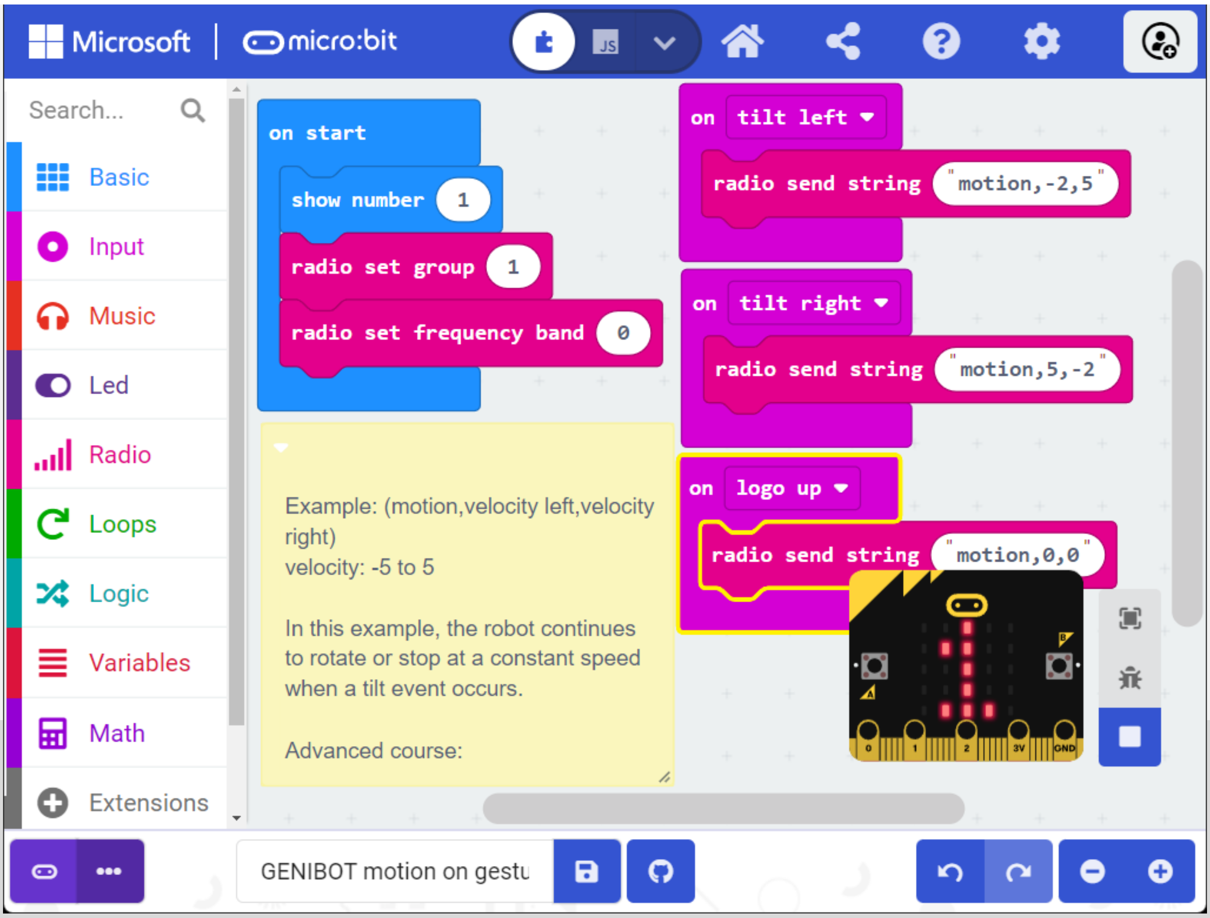


Tap BBC micro:bit Radio in the left sliding menu.

Set radio band and group numbers, then tap the play button.

Programming BBC micro:bit

To enable the robot to listen to the BBC micro:bit radio, program the BBC micro:bit using the Makecode editor. Using the radio set group and radio set frequency band blocks of the Makecode editor, set the band and group, and program the radio message to be sent to the robot with the radio send tring block.
Using the radio string block, you can broadcast command messages such as motion, move, turn, color, sound, tone, remote, and servo to the robot. For more details, see the micro:bit-GENIBOT radio programming guide.



The motion statement in the above example is a syntax defined to control the robot with the BBC micro:bit radio string block. If the syntax is undefined, radio number block can be used to control the robot by broadcasting numeric values.

BBC micro:bit-GENIBOT Radio String Command Syntax

When using the robot's Bluetooth and micro:bit’s radio together at the same time, a value sent by "radio send string" in the "on gesture" event block may be ignored in the robot's Bluetooth event task priority, so it is recommended that the MakeCode program should continue to send event values to the robot using a variable block which receives event values from the "on gesture" block.

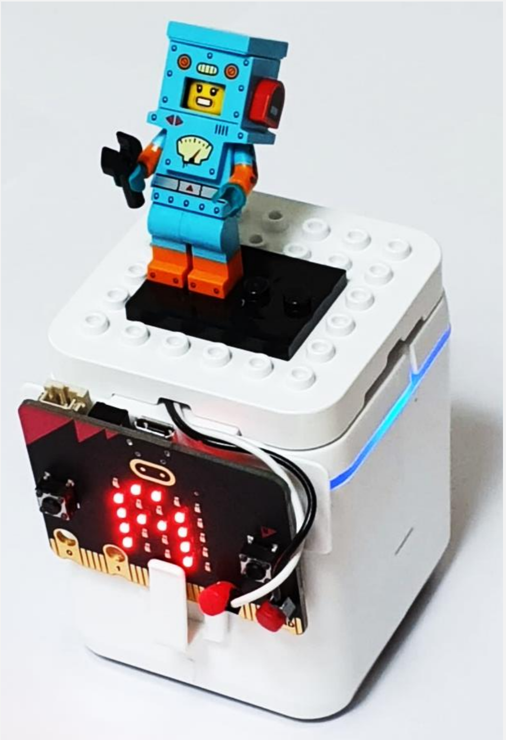
Commands	Descriptions
motion,*velocity left,velocity right*	To make linear or rotational motion: *velocity left and right:* -5 to 5
move,*velocity,distance*	To move forward or backward: *velocity: -5 to 5* *distance (cm): 1 to 15*
turn,*velocity,angle*	To make rotational motion. *velocity:-5 to 5* *angle (degrees): 1 to 180*
color,*LED number,color number,brightness*	To display LED colors: *LED number: left 4, right 2, front 1, back 3, all 0* *color number: red 0, green 1, blue 2, cyan 3, magenta 4, yellow 5, violet 6, orange 7, spring green 8, light pink 9, white 10, black 11* *brightness level: 1 to 8*
sound,*music instrument number,note,volume*	To play music note: *music instrument number: piano 0, flute 1, string 2* *note: 37 (A3 La) to 57 (F5 Fa)* *volume: 1 to 9*

Commands	Descriptions
tone,*frequency,duration, amplitude*	To generate pure tone with sinusoidal waveform: *frequency (Hz): 220 to 1100* *duration (in 0.1seconds): 2 to 20 (0.2s to 2s)* *amplitude: 1 to 10*
remote,*acceleration ax,acceleration ay*	To make any linear or rotational motion, transmit continuous acceleration measurement data set of ax and ay to the robot: *Acceleration ax: -1024 to 1024,* *Acceleration ay: -1024 to 1024*
servo,*geek or micro:servo,angle,wait time*	To make servo motion: *servo type: geek servo 0, micro:servo 1* *angle (degrees): geek servo -45 to 225, micro:servo 0 to 180* *wait time (seconds): 1s to 10s*
on,*gesture*	When you do a gesture which means the way you hold or move the micro:bit, send "on gesture" parameter to the robot. *gesture parameters: shake, screen up, screen down, tilt left, tilt right.*

If there is no string command syntax you want to use, you can send the numeric value directly to the robot using the radio send number block and then use the value received with Scratch or Python program.

EXTERNAL TERMINAL PORT

Physical Computing and Coding



Peripherals programming using Scratch or Python

Peripheral devices such as Arduino sensors, motors, and actuators can be connected to the robot and programmed using Scratch or Python. For more information, see the GENIBOT Arduino reference.

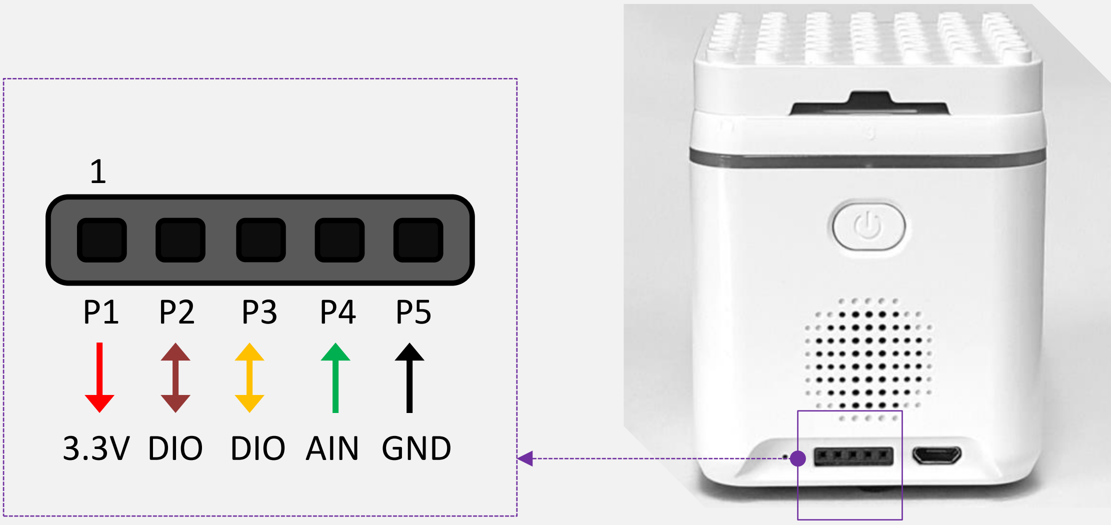
Warning regarding external terminal port use

Pay special attention to the external terminal port information provided here. Operate the robot only as described in “Warning” and “Caution” conditions. Using and operating the robot under conditions not specified herein may result in damage, injury, and may result in damage to external devices connected to the robot.

External Terminal Port Connection Diagram

The following diagram shows how to connect peripheral devices such as Arduino sensors, actuators, motors, and LEDs to the five pins of the external terminal port. For more information, see the GENIBOT Arduino reference and progrmaing guide document. P1 (DC 3.3V) pin of the external terminal port is connected to the robot's power source, so never connect an external power source to any pin of the external terminal port.

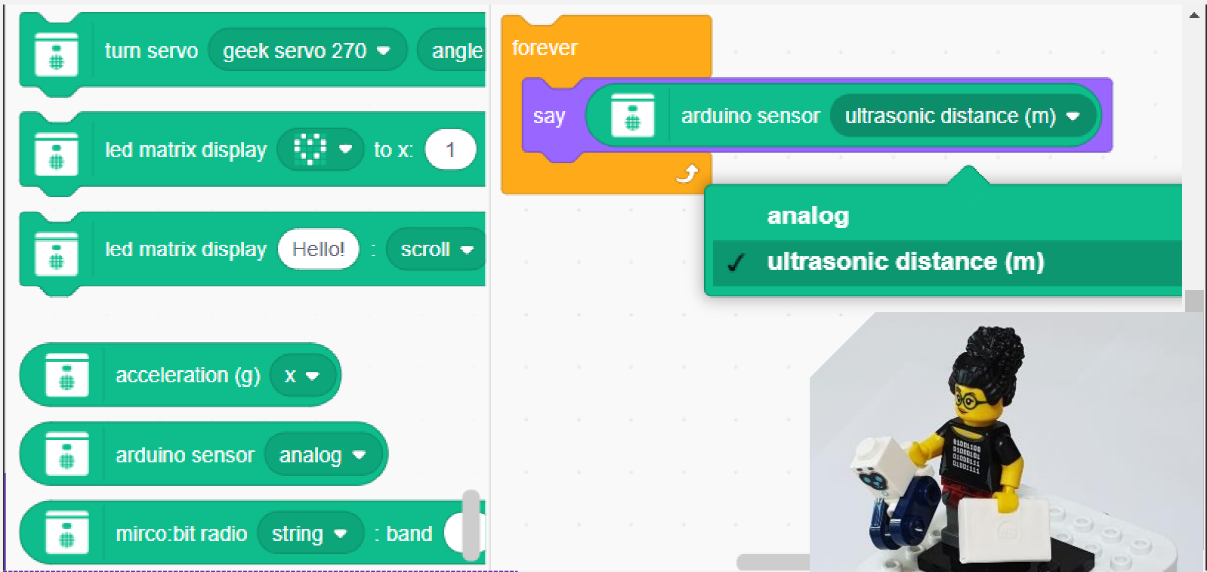
External terminal port
P5 – GND
P4 – AIN (Analog Input)
P3 – DIO (Digital Input / Output) / I2C SCK (Internal pull-up resistor) / Servo
P2 – DIO (Digital Input / Output) / I2C SDA (Internal pull-up resistor)
P1 – DC 3.3V

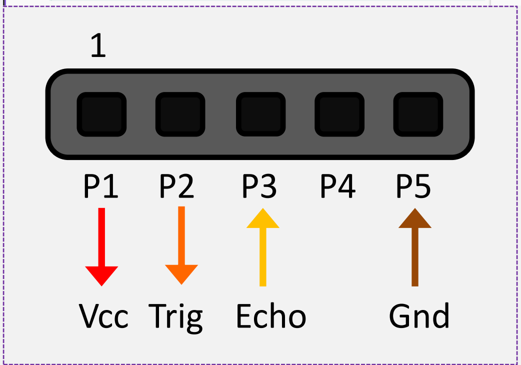


Warning: Connection
Do not short P1 (DC 3.3V) and P5 (GND).
Do not connect an external power supply.
Do not connect positive or negative voltage sources from external devices.
Connect the GND of external device to P5 (GND).

Example 1: Distance Measurement using Scratch

Connect the robot to the computer to measure the distance with scratch programming using the ultrasonic distance sensor HC-SR04P (DC 3.3V). Select the ultrasonic distance (m) item in the Arduino sensor block item on the Scratch editor screen, and then program to show the distance reading.

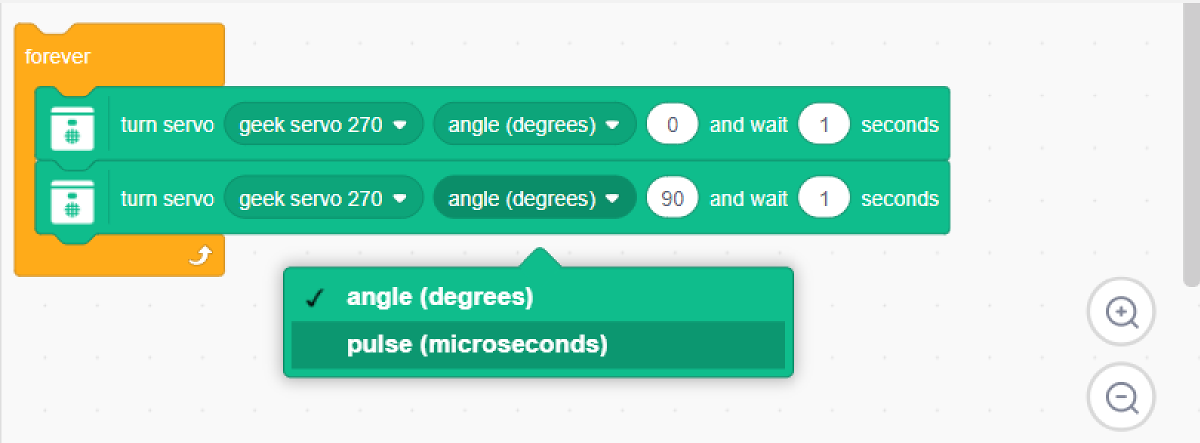


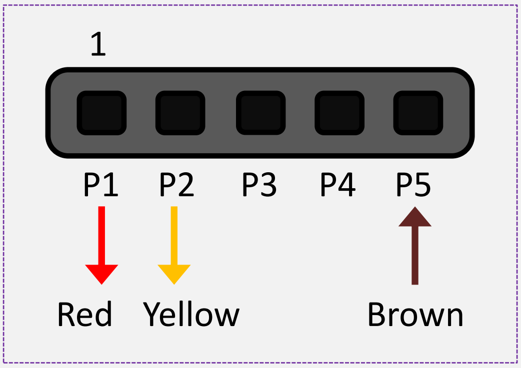


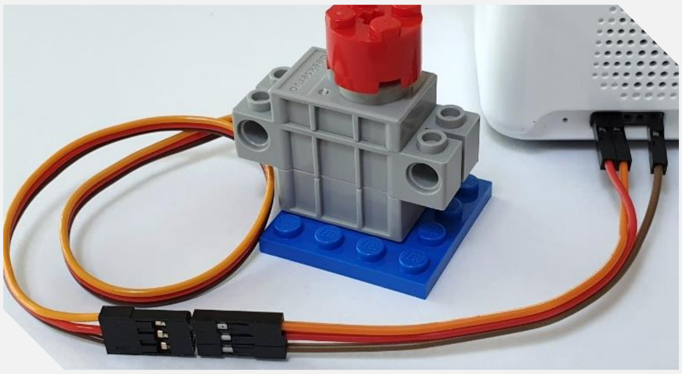
Warning: Connection
Connect the Vcc pin of HC-SR04P to P1 (DC 3.3V).
Connect the Trig pin of HC-SR04P to P2 (DIO).
Connect the Echo pin of HC-SR04P to P3 (DIO).
Connect the Gnd pin of HC-SR04P to P5 (GND).
Caution: Ultrasonic distance sensor operating voltage
HC-SRO4P (DC 3.3V) must be powered by the robot's DC 3.3V power source (P1 and P5).
HC-SR04 (DC 5.0V) cannot be used.

Example 2: Geek Servo 270 or Micro Servo 180

To start a Geek Servo or micro:servo motor, select the turn servo block and enter a decimal number of degrees (-45 to 225 for Geek Servo, 0 to 180 for micro:servo). Enter the wait time in seconds for the block to operate. If you set the wait time to a value lower than 0.5 seconds, when two or more turn servo blocks are connected, the next block will be executed even before the operation of the previous block is completed.





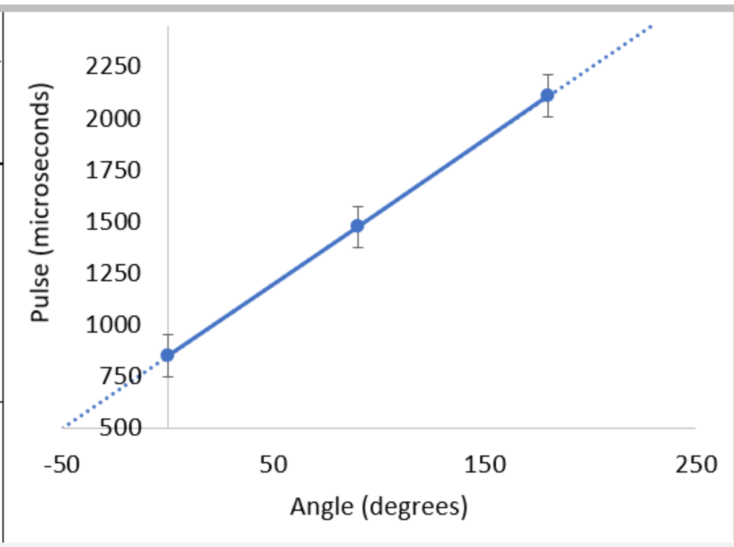


Warning: Connection
Connect the Red cable of Geek Servo 270 to P1 (DC 3.3V).
Connect the Yellow cable of Geek Servo 270 to P2 (DIO).
Connect the Brown cable of Geek Servo 270 to P5 (GND).

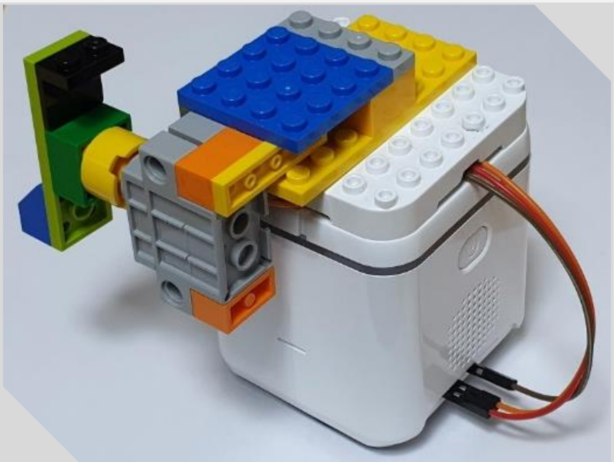
The servo motor calibration values for the rotation angle of Geek Servo 270 and micro:servo have been stored in the robot.

To rotate a servo with pulses, use a calibration equation that converts angles to pulses. These calibration equations were derived from laboratory tests and measurements and may have uncertainties in practice, including systematic errors in the servo motors. If the observed angle is not the same as the angle entered in the turn servo block, user’s calibration is required and error-corrected pulse parameters must be sent to the robot instead of the angle.

Laboratory		Model	
Pulse (µs)	Angle (°)	Pulse (µs)	Error
	-45	533	
850	0	848	3
1475	90	1478	11
2110	180	2108	3
	225	2423	
		Slope	7.00000
		Intercept	848.33333
		Objective	16.66667

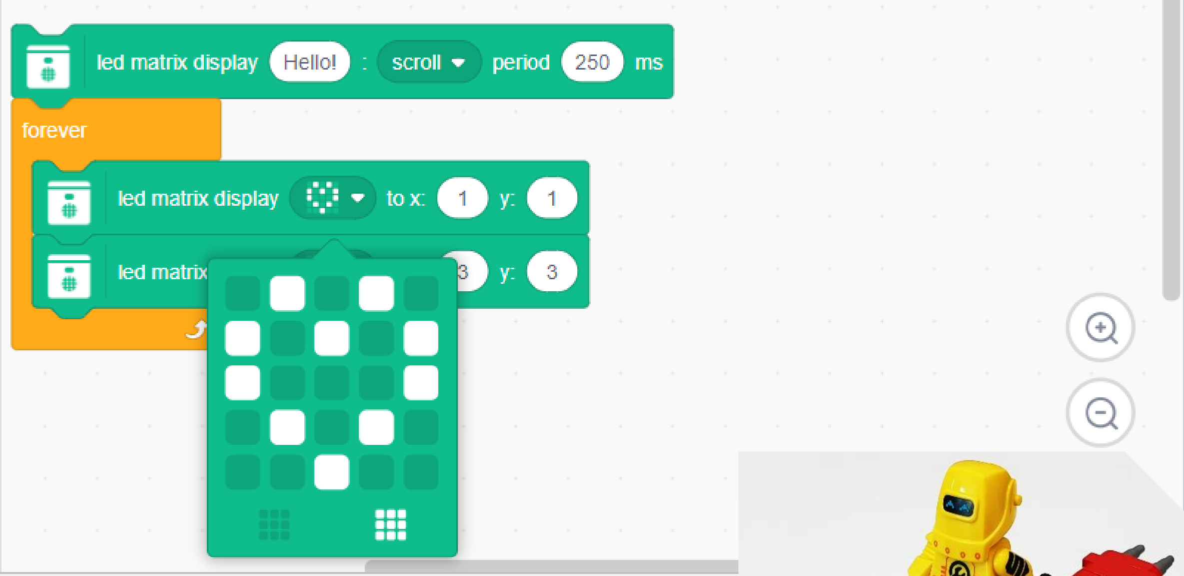


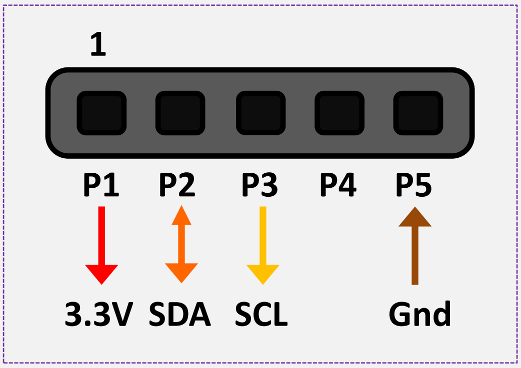
Geek Servo 270 calibration
From the laboratory data table and graph above, the calibration equation for the relationship between angle and pulse is: Pulse = Angle * 7.00000 + 848.33333



Example 3: Monochromatic LED matrix

Connect a single-color LED matrix to the robot's external terminal port, then program it using Scratch or Python to display text or animated pictures.
Available monochromatic LED 8x8 matrices:
Grove HT16K33 8x8 Matrix, Keyestudio HT16K33 8x8 Matrix, Adafruit HT16K33 LED Backpack (Mini 8x8 LED Matrix w/I2C)

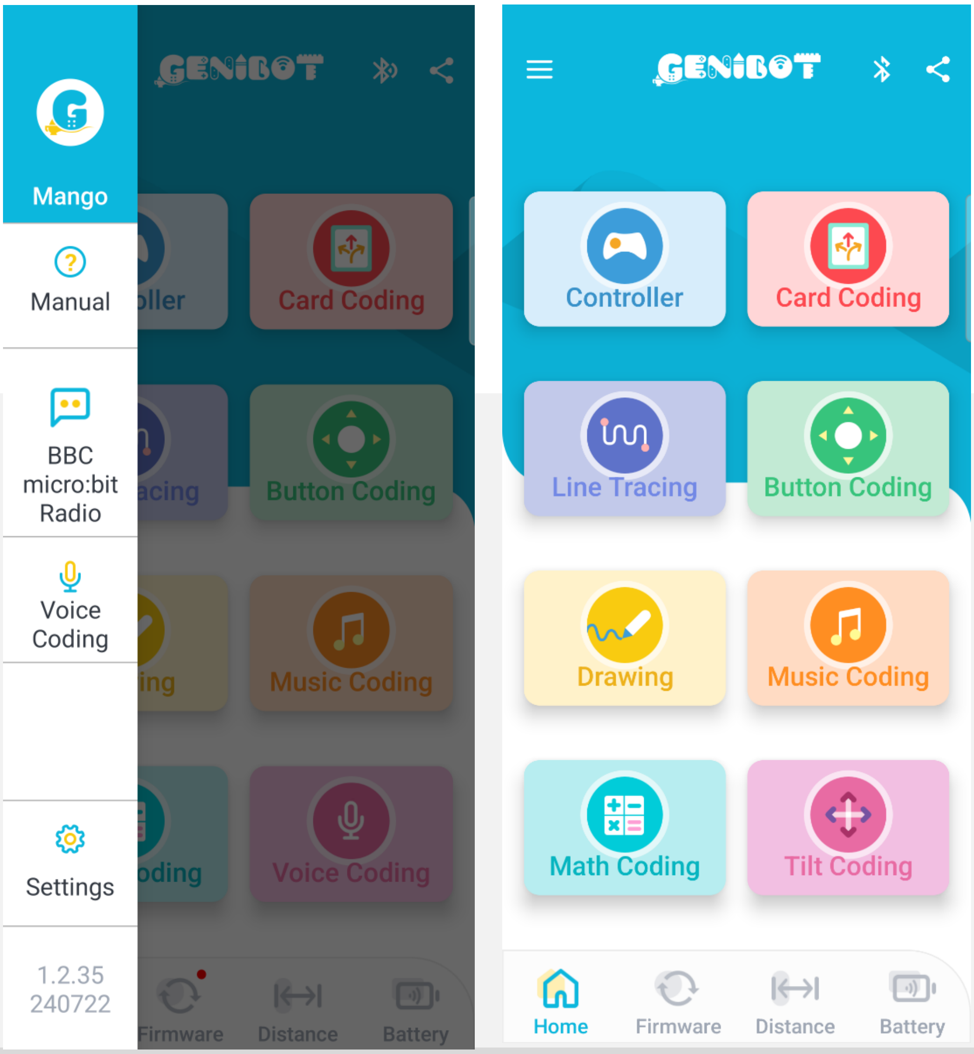




Warning: Connection
Connect LED Matrix DC 3.3V pin to P1 (+3.3V).
Connect LED Matrix SDA pin to P2.
Connect LED Matrix SDL pin to P3.
Connect LED Matrix SCL pin to P5 (GND).

GENIBOT APPLICATION

Google Android or Apple iOS App



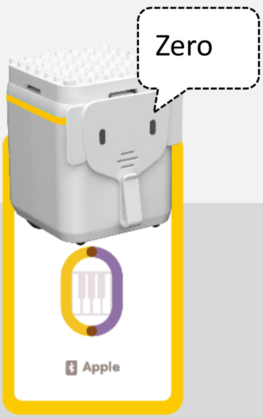
Start coding with the GENIBOT app

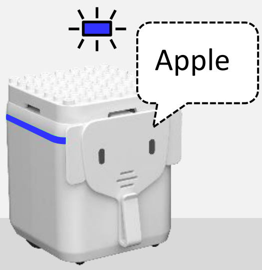
To start the GENIBOT application, your robot must be connected to a smart device for Android or iOS. When you press the button on the robot, the robot's Bluetooth name will appear in the Bluetooth search list on the screen. After connecting, you can select and start Controller, Line Tracing, Drawing, Card Coding, Button Coding, Math Coding, Music Coding, Tilt Coding or Voice Coding on the home screen.

Bluetooth Name

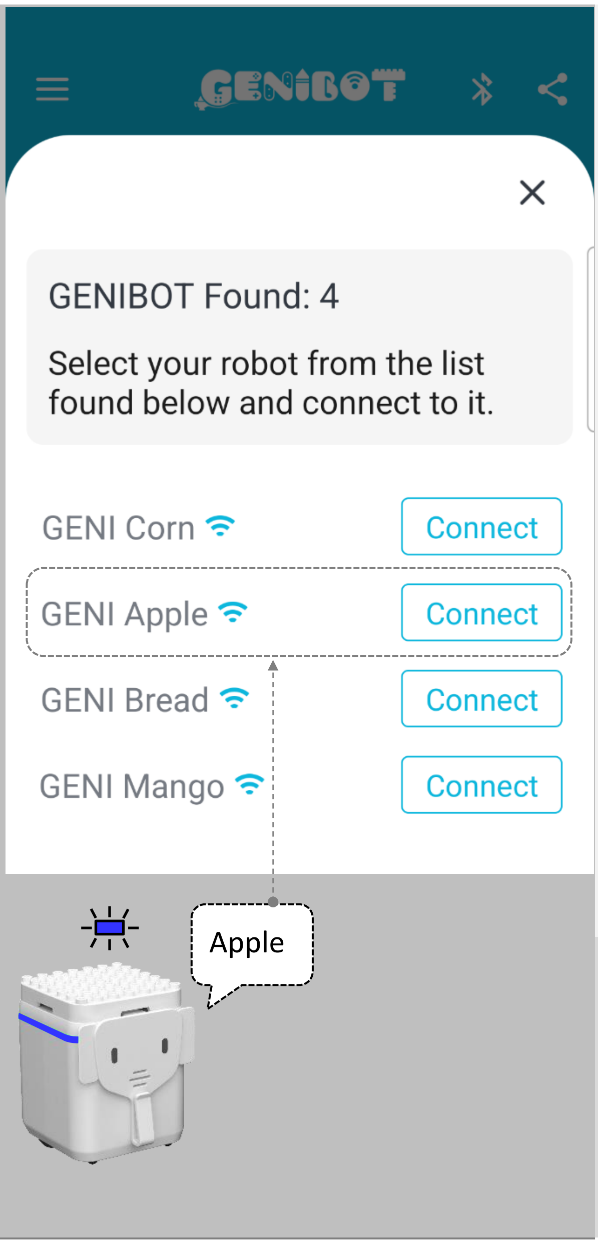
When there are many robots in the classroom, you need to know which robot is yours when you start Bluetooth. When you press the button on the robot, the robot starts Bluetooth and says its Bluetooth name. As explained in the previous chapter, you can also change the Bluetooth name using a coding card.

If the card coding data remains in the robot's memory, the robot will not start Bluetooth, so you must delete the coding data from the robot's memory. The reason is that when you press the button to start Bluetooth, the coding data in memory is executed. This situation may change in the future with firmware updates.
To clear a coding dataset from the robot's memory, simply tap the start card, then tap the end card without tapping any other cards.





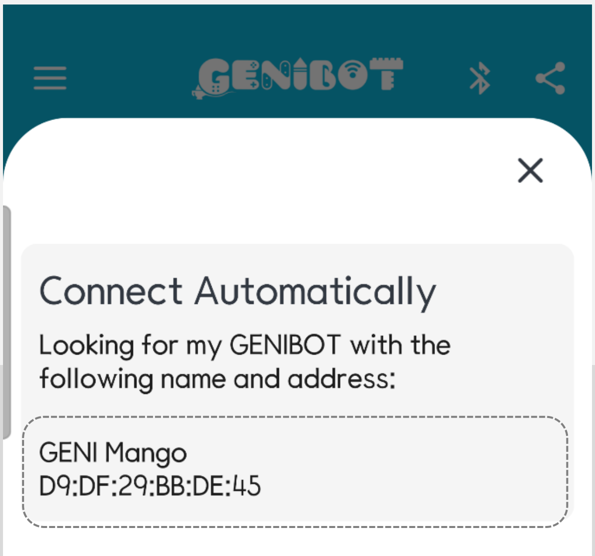
Press the button to start Bluetooth

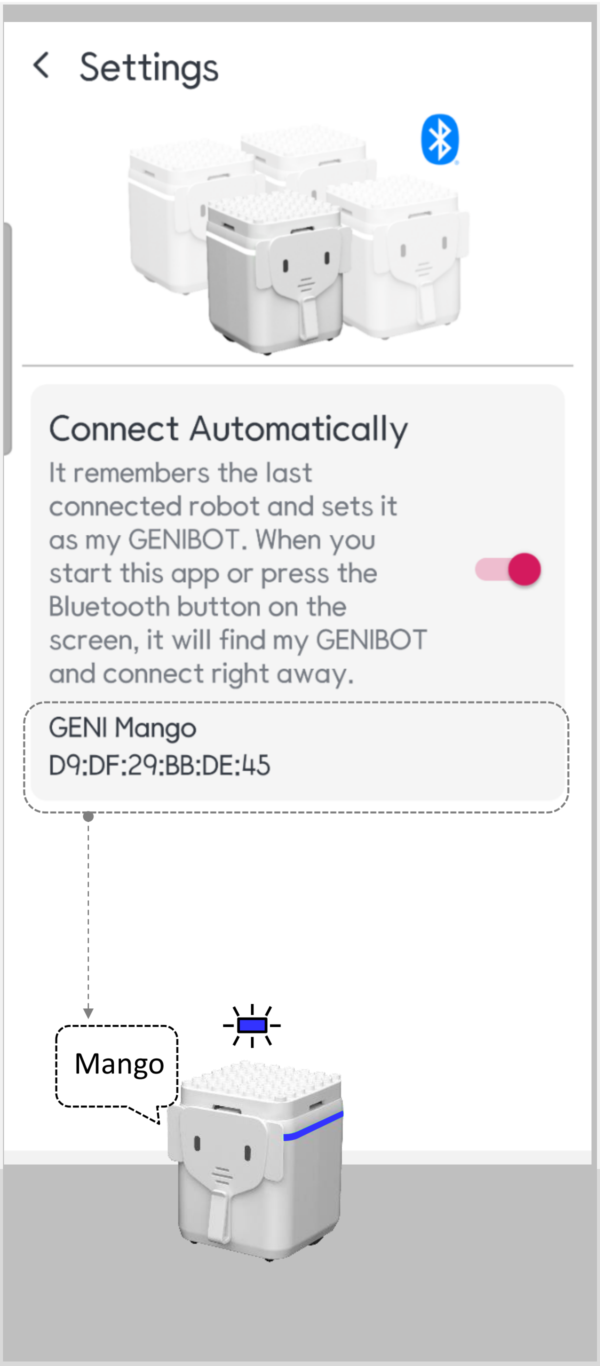


Connect Automatically

It remembers the last connected robot and sets it as “my GENIBOT”. When you start this app or press the Bluetooth button on the screen, it will find “my GENIBOT” and connect right away when there are many robots around in various collaborative or cooperative learning activities.

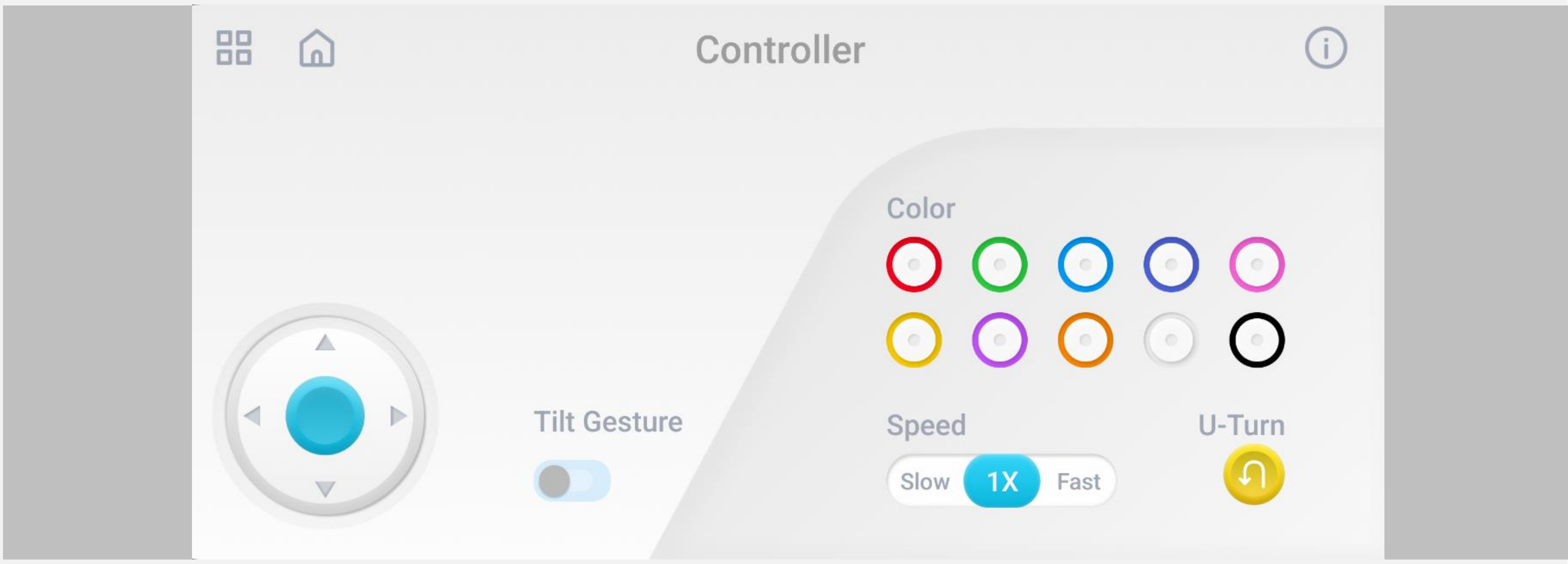
As shown in the picture on the right, after setting up automatic connection and restarting the app, it will find the robot by its configured Bluetooth name and address.



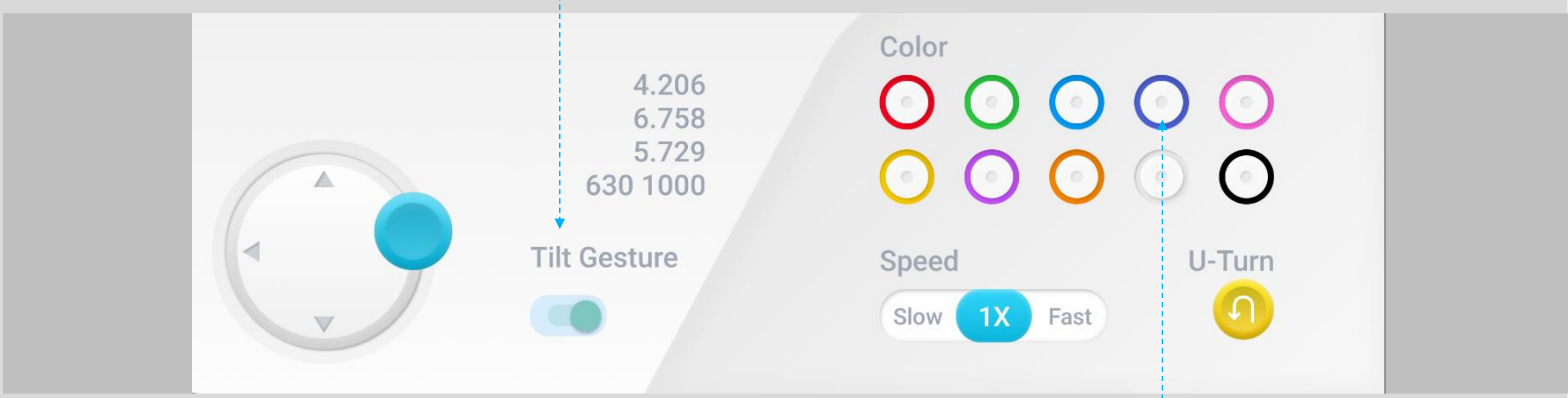


Controller

By tapping or dragging the joystick icon on the display, you can control the robot to move back and forth or rotate. You can change the color of your robot's LED by selecting a color.



By tilting your smart device, you can turn the robot left or right, or move it forward or backward.

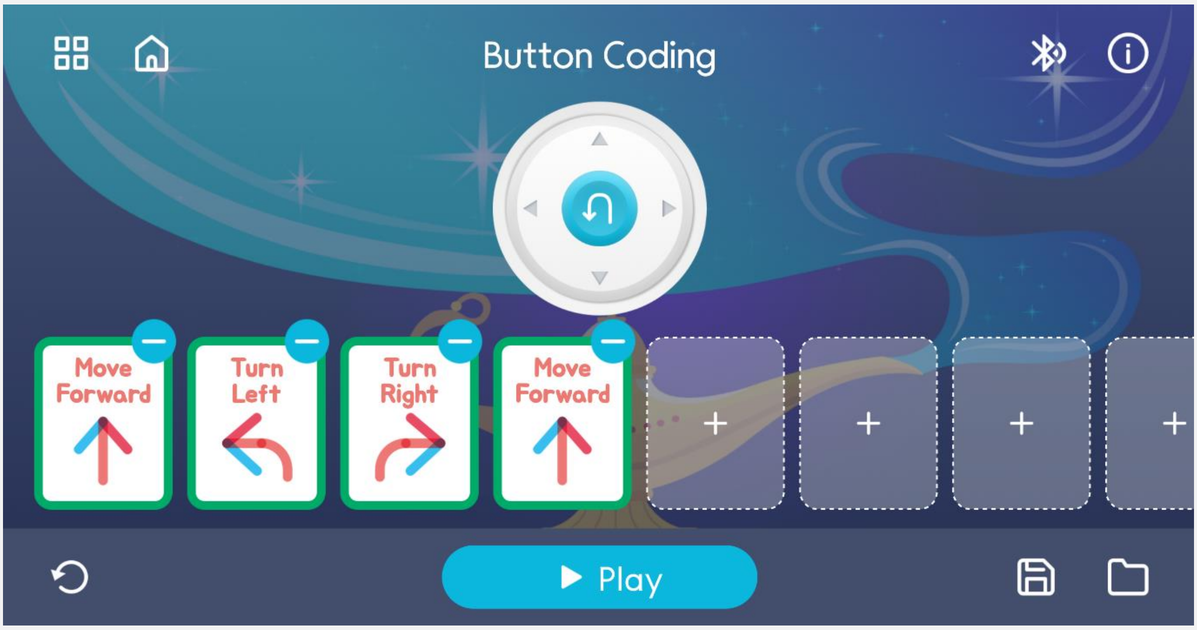


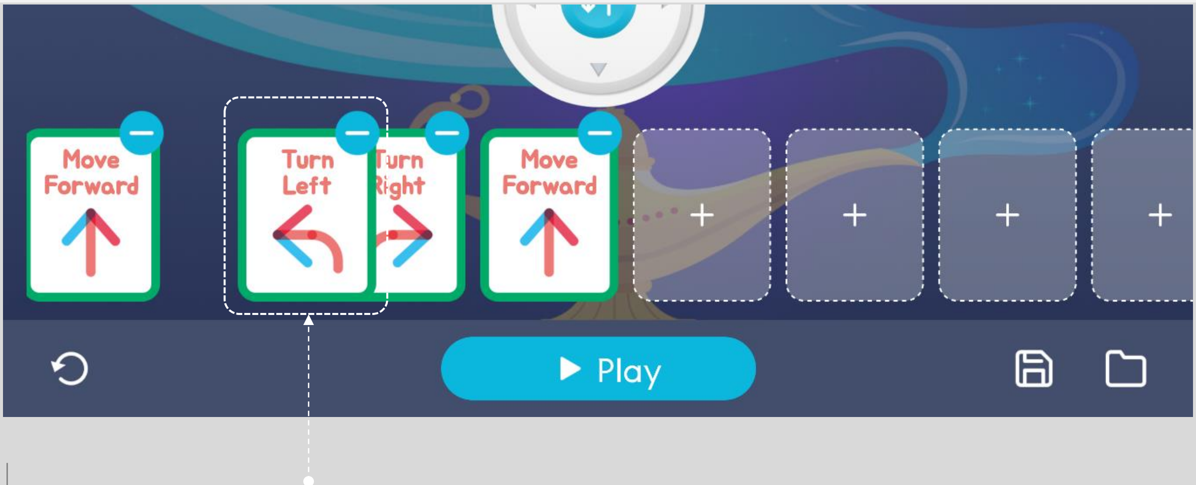
By dragging, you can turn the robot left or right, or move it forward or backward.

By tapping, you can change the robot's LED color.

Button Coding

To create a coding dataset, tap the triangle icon on the joystick to add the Move Forward, Move Backward, Turn Left, and Turn Right cards to the coding data list. After the coding dataset is completed, you can run the coding dataset by tapping the Play button icon.

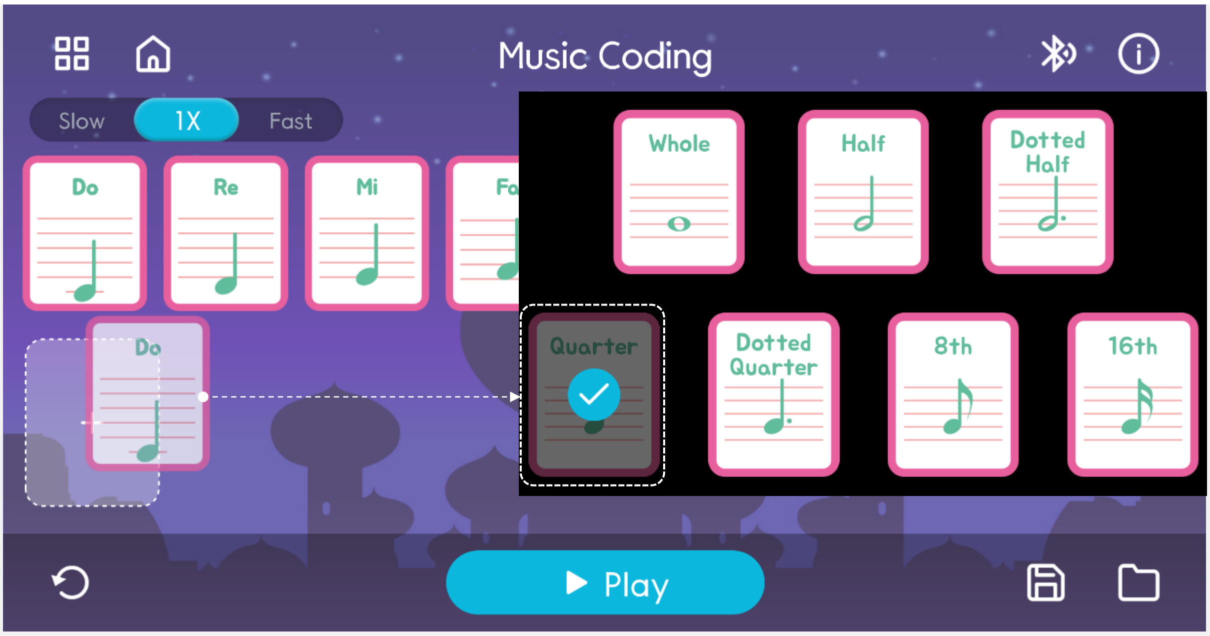


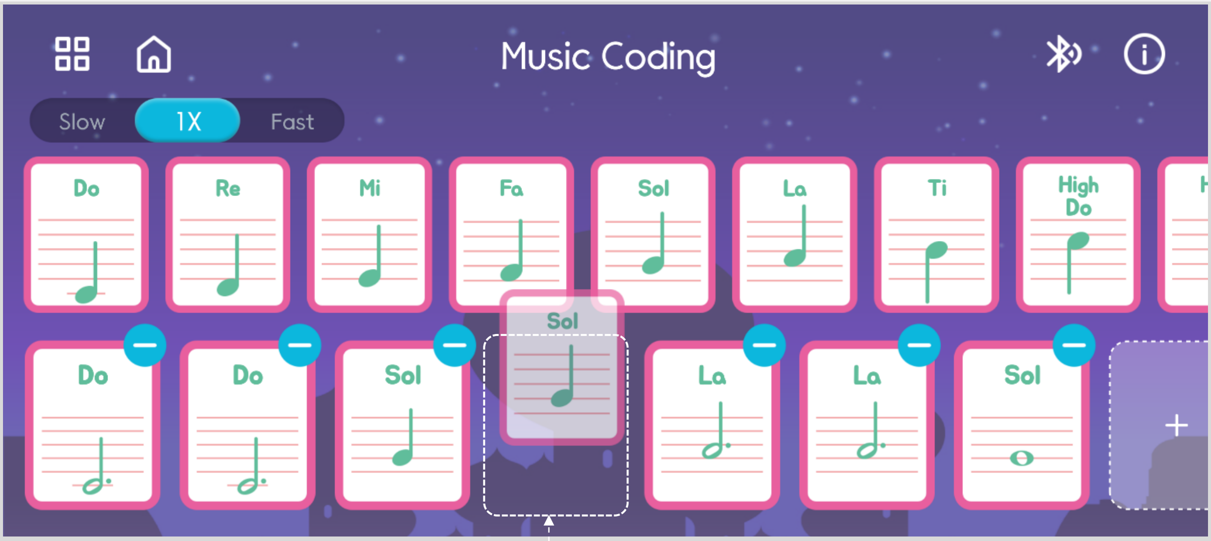


You can change the sequential order of cards in the coding card list by dragging.

Music Coding

To create a song, drag the music card from the music coding screen and dropping it into the music coding list, then a dialog window will appear where you can select a music note. Once the music data is compl ete, tap the Play button icon and the robot can play the song.

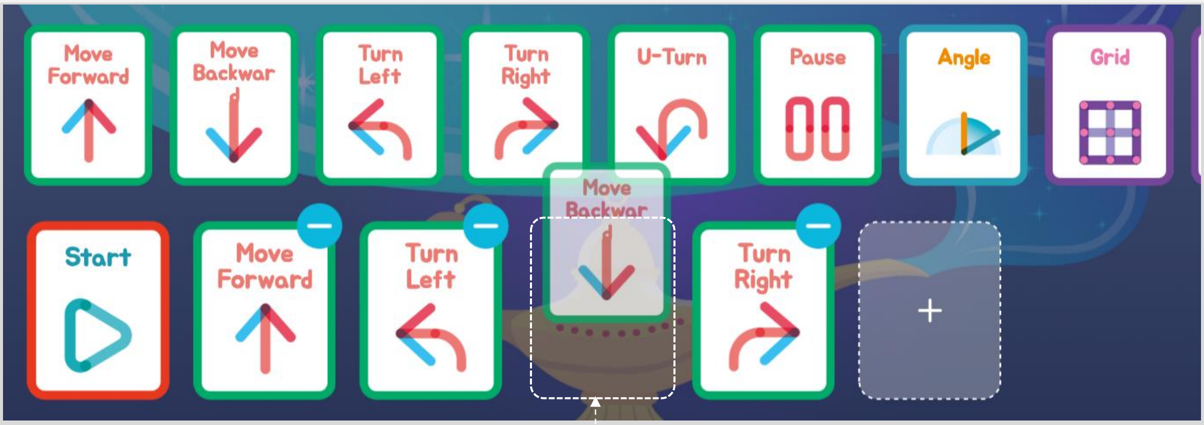




You can drag a music card to insert it anywhere in the music coding card list.

Card Coding

Drag the card icon on the Card Coding screen and place it in the coding card list at the bottom of the screen. Card coding using this app is the same as unplugged card coding. Once the coding dataset is complete, you can tap the Play button icon to have the robot run the coding data.

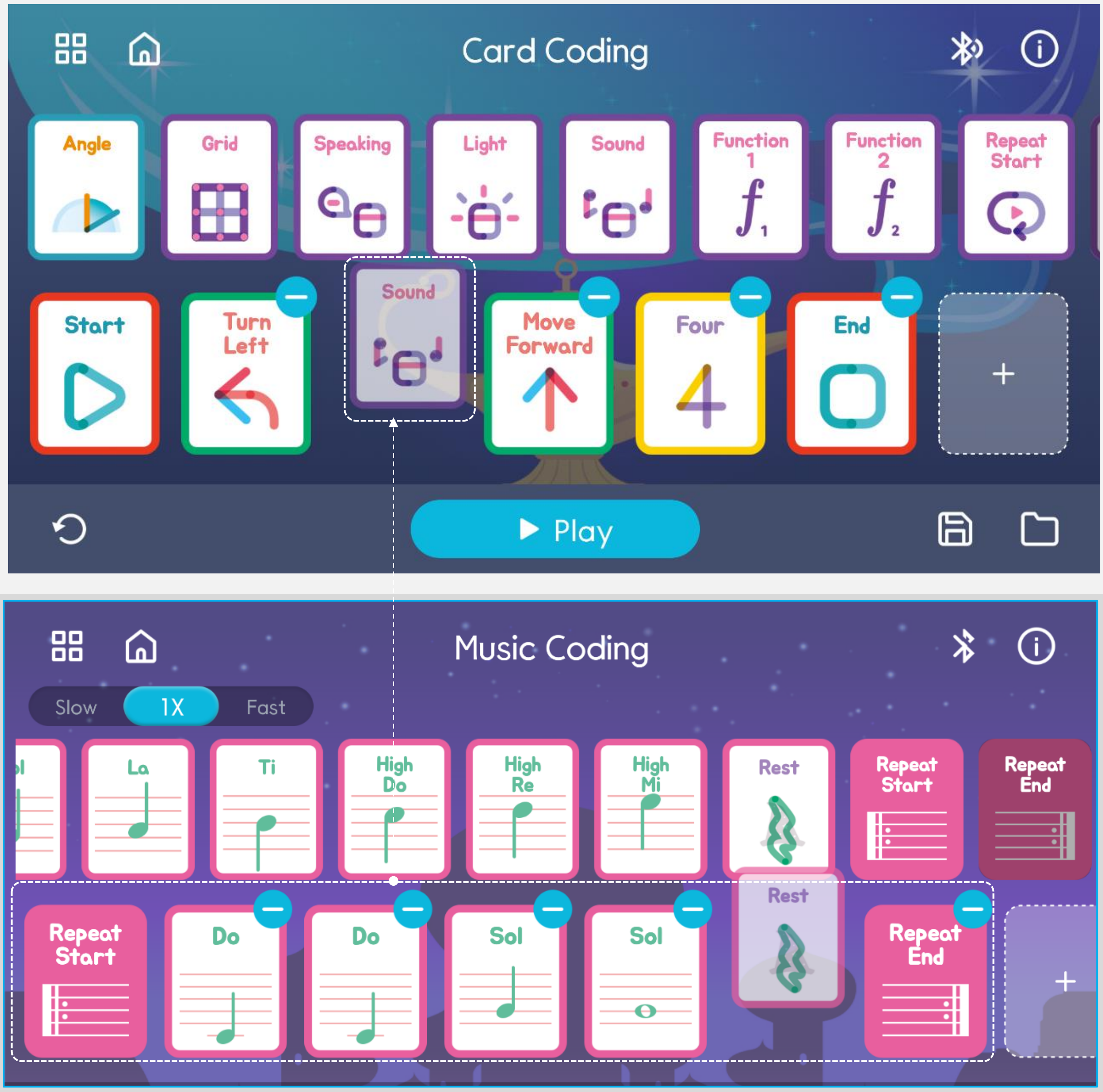


Once coding is complete, you must finally add the End card to the coding card list to enable the Play button.

You can drag a coding card to insert it anywhere in the card coding card list.

Example: Sound Card

If music data is stored on the Sound card, you can use it for Card Coding. To add music data to the Sound card, perform Music Coding first before Card Coding. When you finish Music Coding, the robot automatically adds music data to the Sound card, so you can use it for Card Coding.

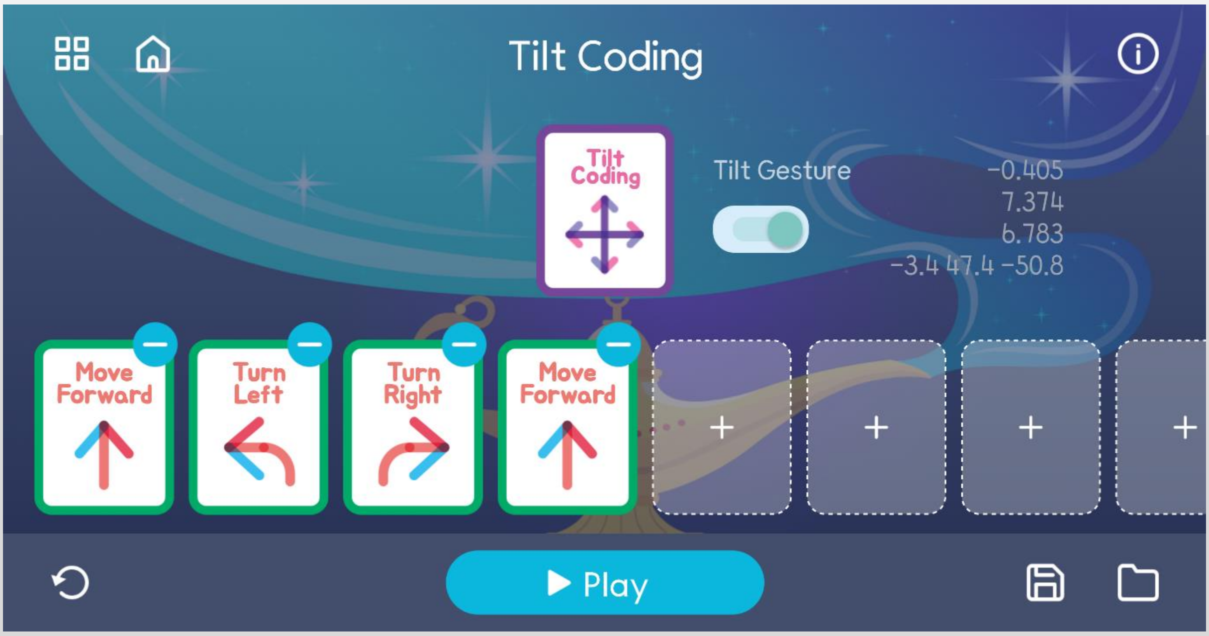


After Music Coding, the Sound Card with music data added can be used in Card Coding.

Tilt Coding

When connecting your robot which has no acceleration sensor, you can use your smart device instead of the robot for tilt gesture recognition.

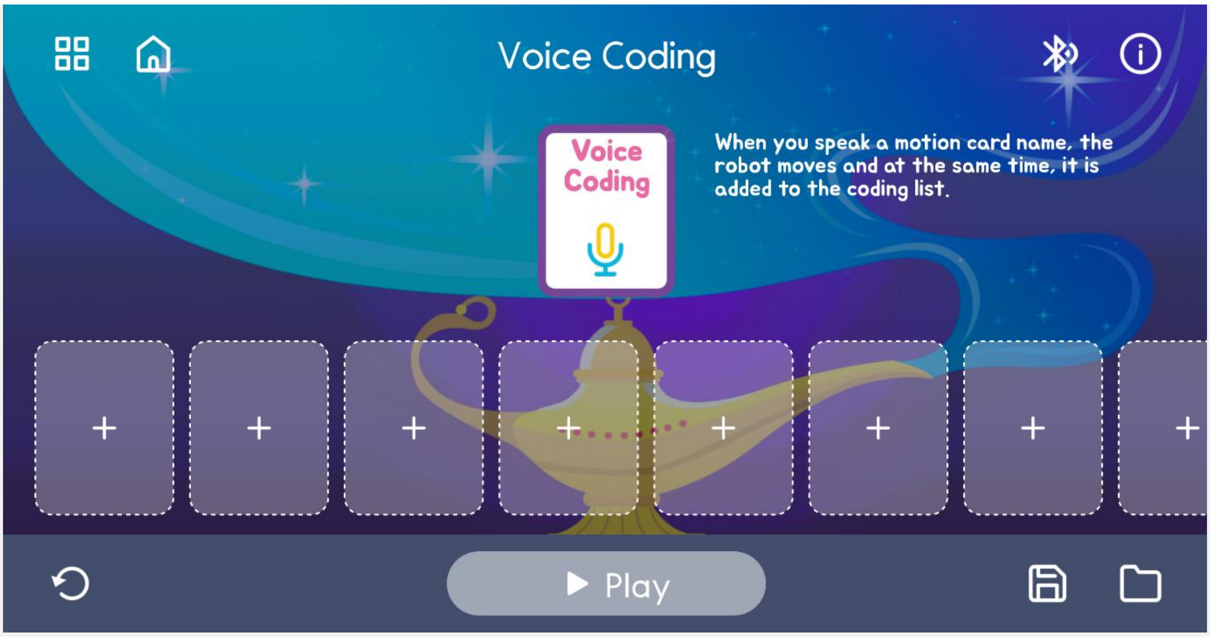
Tilt your smart device left, right, forward, or backward to add Turn Left, Turn Right, Move Forward or Move Backward card to the list of coding cards. At this time, the acceleration sensor and tilt values of the smart device according to the gesture can be read on the screen.

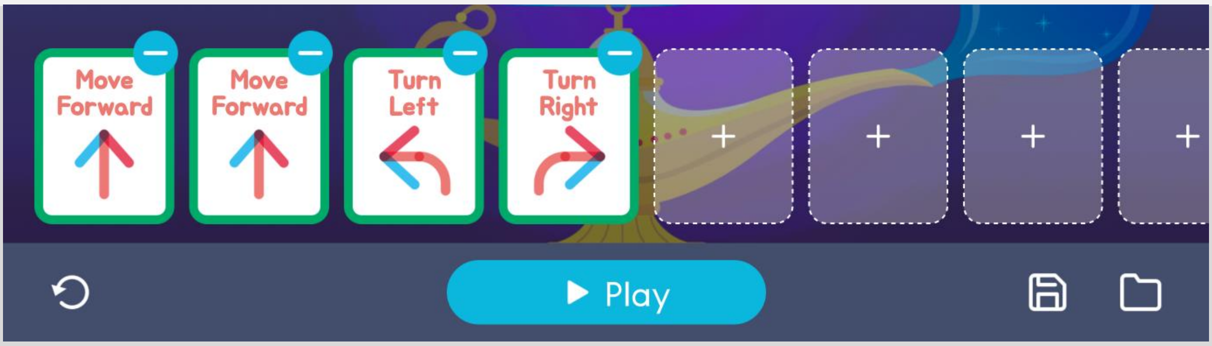


Keep your smart device horizontal, tilt it to add a card to the list, then keep it horizontal again.

Voice Coding

If you select Voice Coding from the left sliding menu, Tilt coding on the home screen changes to Voice Coding. Using Google Android's AI voice recognition function, it automatically detects your voice when you start speaking and displays what you said on the screen. If the voice recognition result is Motion Card, the robot starts executing the card as adding it to the coding list.

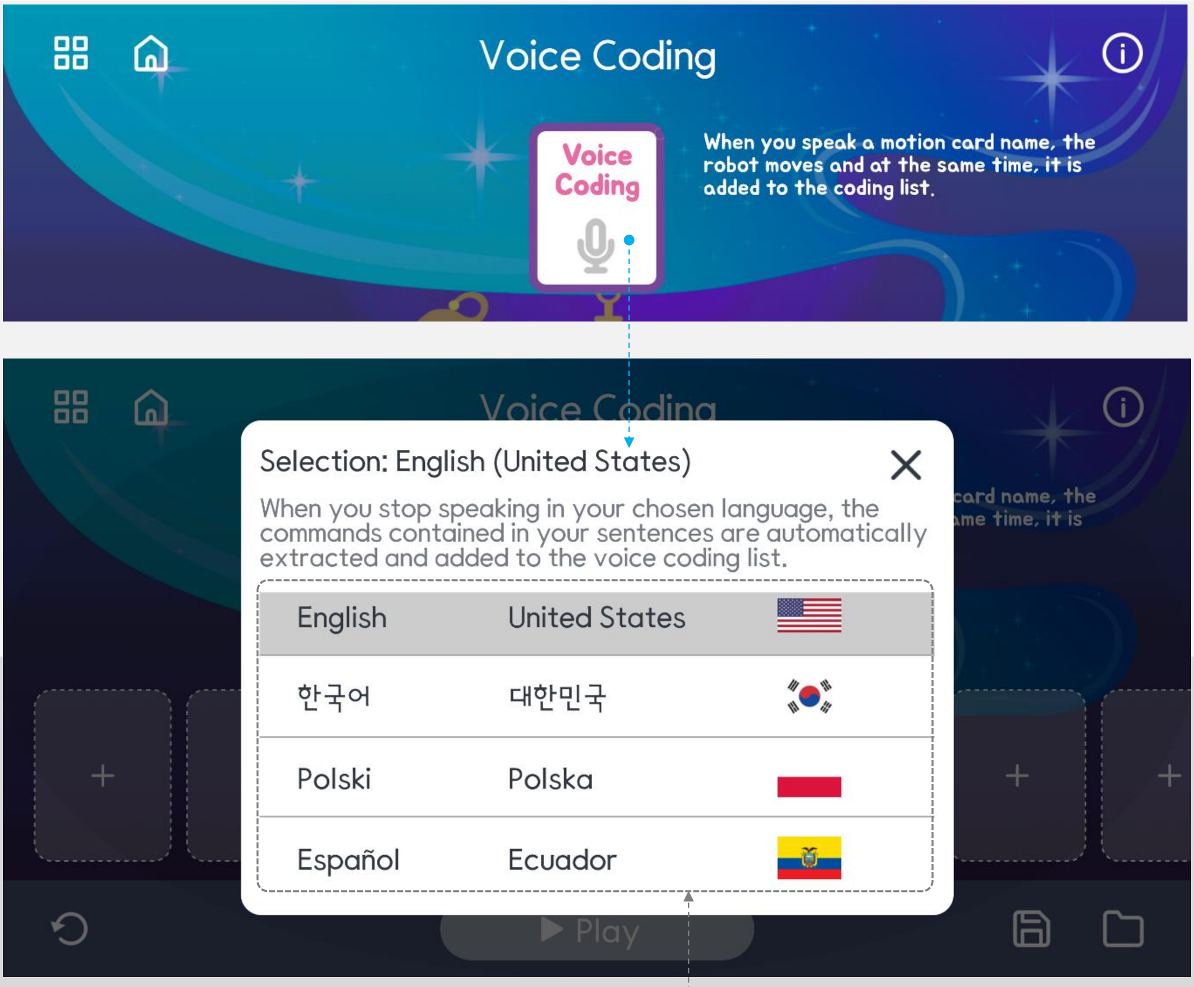




After you finish coding with your voice, when you press the start button, the data in the coding list is sent to the robot, and it executes the coding data.

Example: Language Selection

When you touch the Voice Coding card on the screen, a multilingual language selection dialog will appear, allowing you to select the voice recognition language. When you stop speaking in your chosen language, the commands contained in your sentences are automatically extracted and added to the voice coding list.

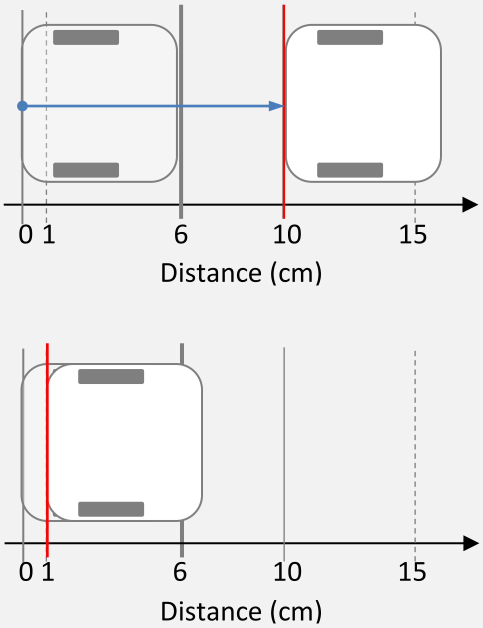


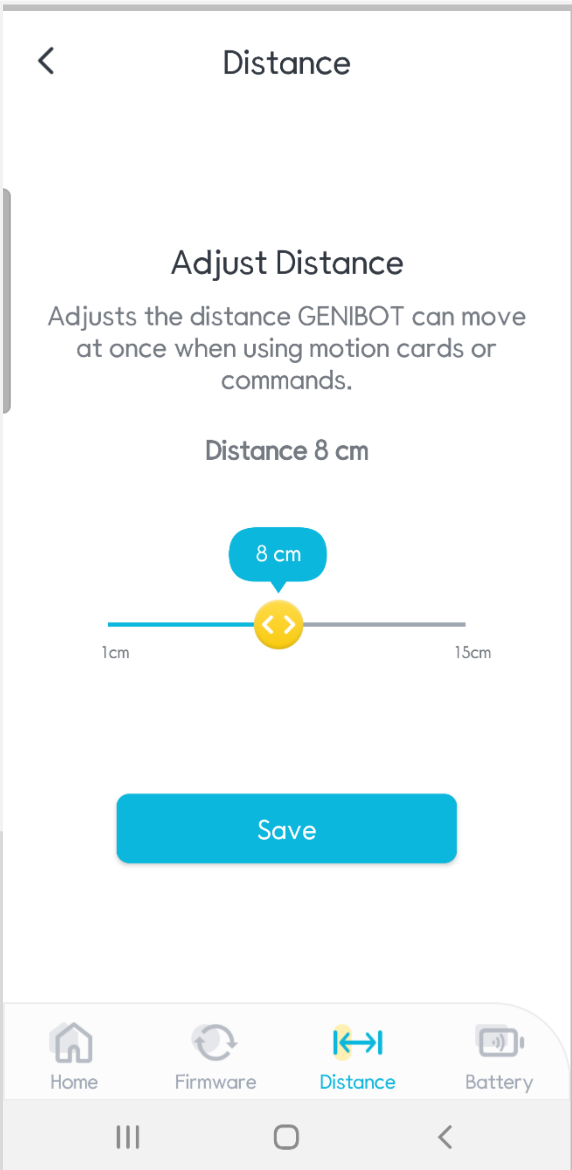
Scroll to select language and then tab the language name.

On the dialog screen, drag up or down to scroll through the list of multilingual languages, then touch the language you want to select. More voice recognition commands and languages may be added in upcoming app updates.

Distance Adjustment

To change the value of the distance the robot can travel by the Move Forward or Move Backward card, tap the Distance button icon in the bottom taskbar. Tapping the Save button icon will send the changed distance value to the robot, but will not save it to the robot. The distance can be changed between 1cm and 15cm.

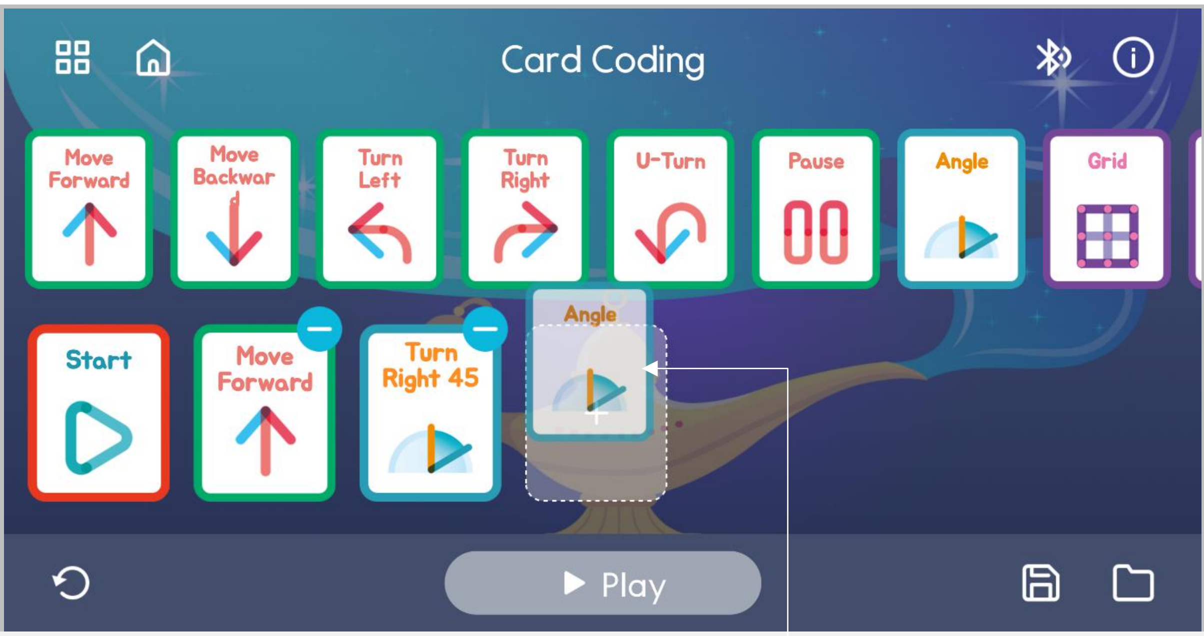




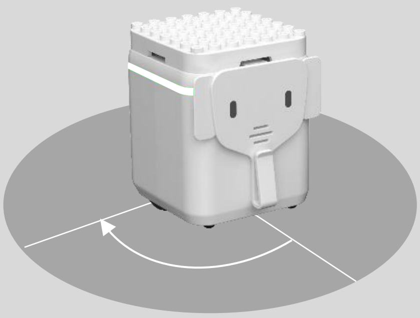
In the future, the firmware can be updated so that the distance value is stored in the robot's memory.

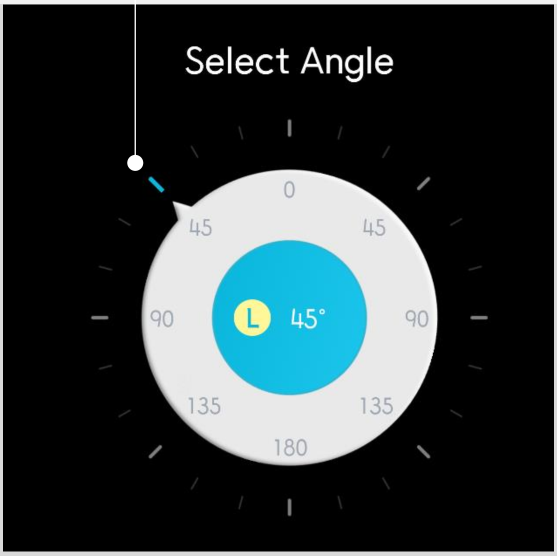
Setting up the Angle Card

To add an angle card to the card coding list during Card Coding, drag and drop it into the card coding list, then drag the rotation angle icon on the Select Angle screen to adjust the angle.



Set the angle of the angle card with the jog shuttle that rotates left and right.





Once coding is complete, you must finally add the End card to the coding card list to enable the Play button.

APPENDIX

Audio Code Reference

	Coding Card	Color Sticker	Card Name	Sticker Name		Bluetooth Name
1	4001		Start			Cat
2	4002		End			Dog
3	4003		Start			Cow
4	4004		End			Pig
5	4005		Move Forward	Move Forward		Lion
6	4006		Move Backward	Move Backward		Tiger
7	4007		Move Forward			Horse
8	4008		Move Backward			Zebra
9	4009		Move Forward			Sheep
10	4010		Move Backward			Goat
11	4011		Move Forward			Bear
12	4012		Move Backward			Panda
13	4013		Move Forward			Wolf
14	4014		Move Backward			Fox
15	4015		Move Forward			Hen
16	4016		Move Backward			Rooster
17	4017		Turn Right	Turn Right		Bird
18	4018		Turn Left	Turn Left		Eagle
19	4019		Turn Right			Jaguar
20	4020		Turn Left			Cheetah
21	4021		Turn Right			Koala
22	4022		Turn Left			Kangaroo
23	4023		Turn Right			Giraffe
24	4024		Turn Left			Rabbit
25	4025		Turn Right			Mouse
26	4026		Turn Left			Hedgehog
27	4027		U-Turn	U-Turn		Crocodile
28	4028		Pause	Pause		Snake
29	4029		U-Turn			Beaver
30	4030		Pause			Squirrel
31	4031	5031	Zero	Zero	Red	Apple
32	4032		Do	Do		Pear
33	4033	5033	One	One	Green	Orange
34	4034		Re	Re		Peach
35	4035	5035	Two	Two	Blue	Banana
36	4036		Mi	Mi		Pineapple
37	4037	5937	Three	Three	Cyan	Melon
38	4038		Fa	Fa		Water Melon
39	4039	5039	Four	Four	Magenta	Strawberry
40	4040		Sol	Sol		Blueberry
41	4041	5041	Five	Five	Yellow	Tomato
42	4042		La	La		Grape
43	4043	5043	Six	Six	Purple	Lemon
44	4044		Ti (Si)	Ti (Si)		Lime
45	4045	5045	Seven	Seven	Orange	Cherry
46	4046		High Do	High Do		Kiwi

	Coding Card	Color Sticker	Card Name	Sticker Name		Bluetooth Name
47	4047	5047	Eight	Eight	Spring Green	Potato
48	4048		High Re	High Re		Sweet Potato
49	4049	5049	Nine	Nine	Light Pink	Mango
50	4050		High Mi	High Mi		Avocado
51	4051		Rest	Rest		Tangerine
52	4052		High Fa	High Fa		Plum
53	4053		Delete			Green Bean
54	4054		Insert			Carrot
55	4055		Function 1	Function 1		Peanut
56	4056		Tilt			Walnut
57	4057		Grid	Grid		Onion
58	4058		Line Tracing			Turnip
59	4059		Light	Light		Pumpkin
60	4060		Sound	Sound		Cucumber
61	4061		Light			Garlic
62	4062		Sound			Mushroom
63	4063		Speaking			Corn
64	4064		Function 2	Function 2		Broccoli
65	4065		Repeat Start			Cabbage
66	4066		Repeat End			Asparagus
67	4067		Repeat Start			Kale
68	4068		Repeat End			Lettuce
69	4069		Addition			Hamburger
70	4070		Subtraction			Pizza
71	4071		Subtraction			Spaghetti
72	4072		Addition			Chicken
73	4073		Turn Right 15			Hotdog
74	4074		Turn Left 15			Cheese
75	4075		Turn Right 15			Sandwich
76	4076		Turn Left 15			Doughnut
77	4077		Turn Right 30			Candy
78	4078		Turn Left 30			Cookie
79	4079		Turn Right 30			Pudding
80	4080		Turn Left 30			Chocolate
81	4081		Turn Right 60			Icecream
82	4082		Turn Left 60			Cake
83	4083		Turn Right 60			Kimbap
84	4084		Turn Left 60			Bibimbap
85	4085		Turn Right 90			Milk
86	4086		Turn Left 90			Yogurt
87	4087		Turn Right 90			Bread
88	4088		Turn Left 90			Jam
89	4089		Triangle			Coffee
90	4090		Square			Cocoa
91	4091		Circle			Coke
92	4092		Triangle			Juice
93	4093		Square			Water
94	4094		Circle			Wine

POWER SAVING

Power off after Sleep Timeout

If there is no operation or coding activity on the robot for approximately 10 minutes, the robot powers off. The auto power off after sleep feature only applies to card coding activities. However, this does not apply to using the GENIBOT app for Google Android and Apple iOS by connecting to a smart device, and to Scratch or Python programming activities by connecting to a computer or tablet.

Low battery notification
When the battery power drops to the lower limit, the robot will blink yellow rapidly four times to indicate low battery, then the robot powers off.
If the battery is low, not only will the robot's motor not operate properly, but its function may also be abnormal. Therefore, before any coding activity, check the battery percentage using the End coding card or the GENIBOT app and fully charge the battery.

Bluetooth SIG: Declaration ID D052026
KC (Korea Certification): Registration No. R-R-Grb-GB1
CE (EU Integrated Standards Compulsory Certification): Declaration No. GB1-EUDOC0424L1
FCC (US Electrical and Communication Equipment Certification): FCC Identifier 2AWFA-GB1
CPC (US Consumer Product Safety): Certificate No. GB1-CPC1006L1
TELEC (Japan Radio Type Approval): Certificate No. 208-200062
ANATEL (Certification for Telecom Products in Brazil): Certificate No. 18043-2316100

GENIBOT is product and trade name of Dabida (GENIROBOT) Co., Ltd.
Other trademarks or trade names of products and company names mentioned herein are: Microsoft Windows and Make Code, Google Android, Apple iOS, BBC micro:bit Educational Foundation, Scratch Foundation and Python Software Foundation.

© 2024 Dabida Co., Ltd. All rights reserved.

GENIBOT Technical Guide
Document Edition
Part Number: GB1-TRG0805P2 First Edition, August 5, 2024

Further Documentation
Visit GENIROBOT website www.genirobot.com for additional and further technical information or up-to-date documentation.