Your EdVenture into Robotics
You're a Programmer

Contents

Introduction

Edison is your new robot buddy who will teach you about electronics, programing and robotics in a fun and engaging way.

He's equipped with all the sensors, outputs and motors needed to introduce you to the amazing world of robotics.

That's great, but what is robotics? Well that's not an easy question to answer. Edison's creator, Brenton O'Brien says *"a robot is a machine that can behave autonomously"* . This means that a robot can think or make decisions on its own and act on those decisions. Many other people have different definitions, but we like this one as it's nice, simple and applies to what you're about to learn.

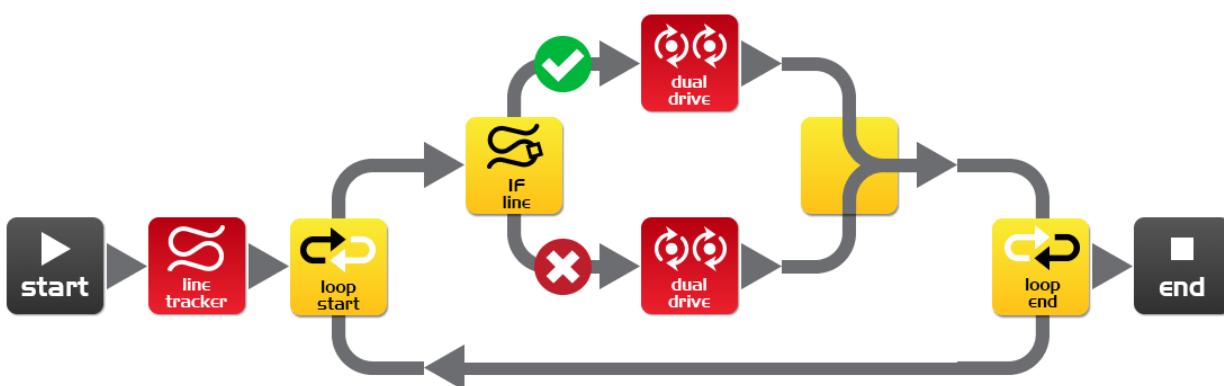Edison the LEGO compatible robot

Robotics wouldn't be possible without electronics, so Edison has his own electronics and you can see it all through his transparent top. There are resistors, capacitors, transistors, motors and more, but the most important electronic part is Edison's microcontroller.

Edison's microcontroller

The microcontroller is like Edison's brain. It's where all his '*thinking*' happens. Edison's microcontroller is very similar to the processor chip inside a computer, only much smaller. And just like a processor chip in a computer, Edison's microcontroller has programs. The programs allow Edison to make decisions and '*think*' for himself.

The most amazing thing about Edison, and what you're about to learn, is that **YOU** get to write Edison's programs! You can tell Edison how to think, behave and respond to his environment. Programming Edison is surprisingly easy. Here's an example program that tells Edison to follow a line.

A simple program that tells Edison to follow a line

Getting started

Before we can program Edison we need to get a few things ready. Here's what we're going to do:

1. Get Edison ready

2. Get to know Edison

3. Install Edison's software 'EdWare'

4. Check that everything is working by downloading a test program

Get Edison ready

Ensure the batteries are the right way

Open the battery compartment and remove the programming cable. Now insert 4 'AAA' batteries. Refer to the image to ensure that the batteries are the correct way around and clip the battery cover on.

If you haven't already done so, stretch the tyres over the wheels.

Turn Edison on by sliding the power switch to the on position. Edison's red LED lights will now start flashing.
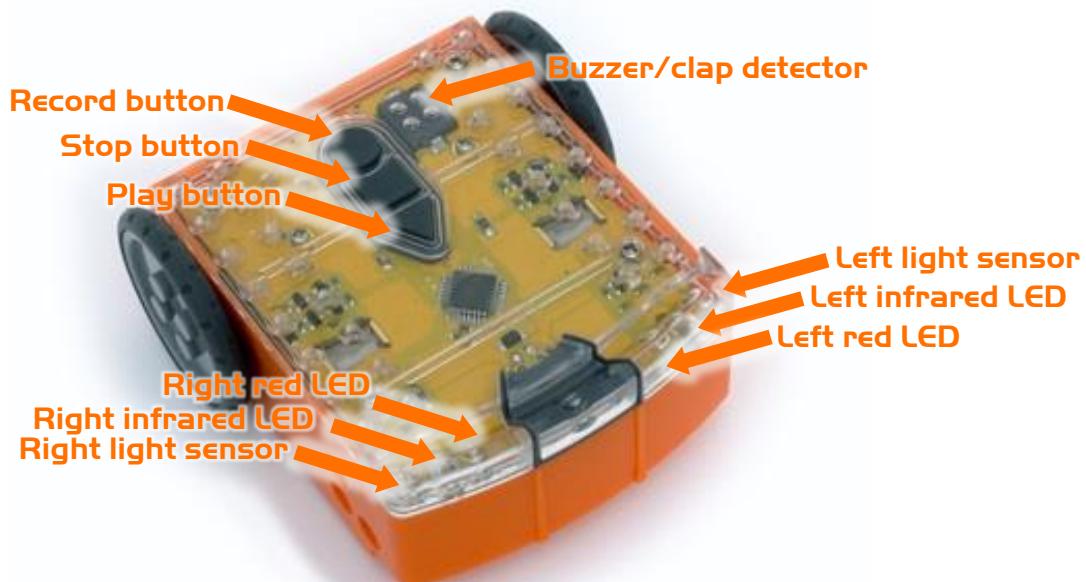
Edison is ready to go!

Push the switch towards the on symbol

Get to know Edison

To use Edison you're going to need to know where all his sensors are and what his three buttons do. Have a look at the images below. You may need to come back and have another look as we go through the EdVentures.

Get to know Edison's sensors and buttons

Play button – Start program
Stop button – Press to stop a program
Record button – 1 press = download program, 3 presses = read barcode

Edison's power switch and line tracking sensor

Edison's line tracking sensor is made up of two parts a red LED light and a light sensor. The red LED shines light on the ground, if the ground is white and therefore, reflects light then the light sensor will get a high light reading. If the ground is black and therefore, does not reflect light then the light sensor gets a low light reading.

The EdComm cable is used to download your programs to Edison. It connects into the headphone socket on your computer or tablet.
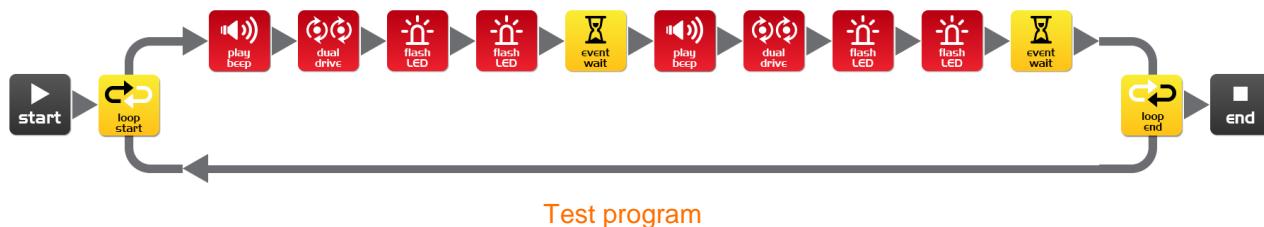
EdComm programming cable
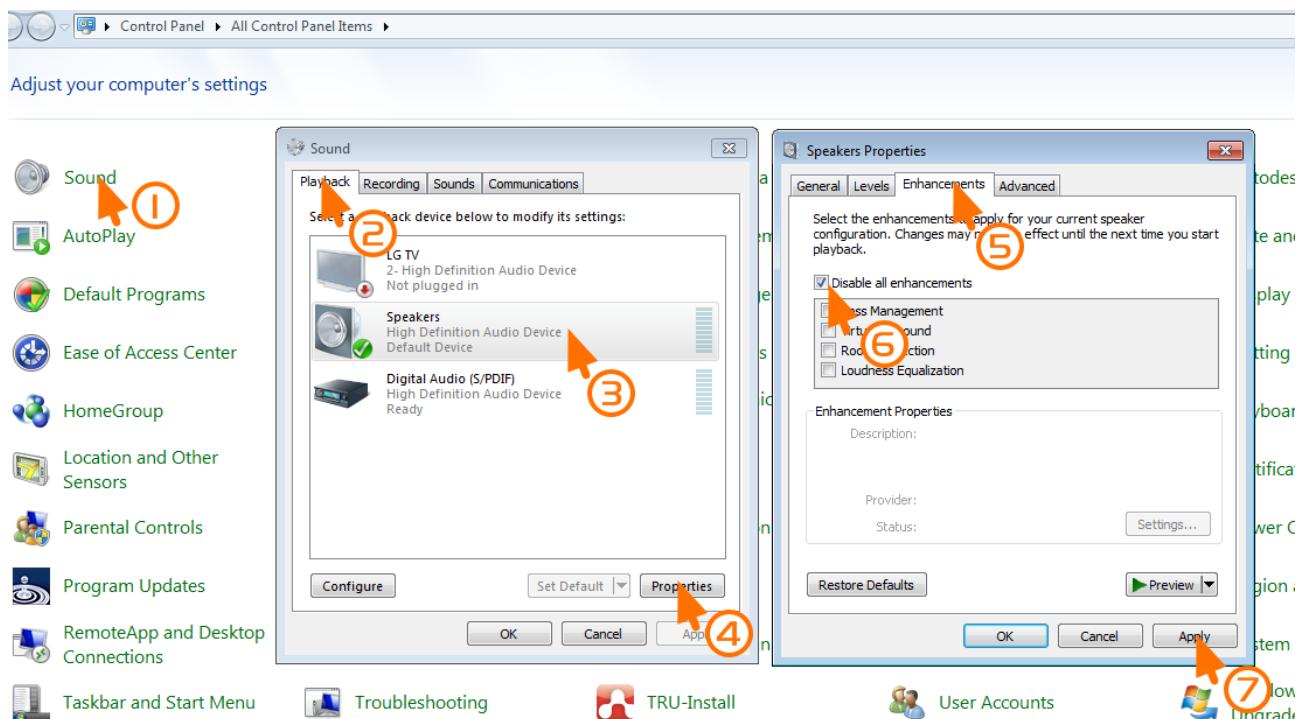
Installing EdWare

The EdWare programming software is available for Windows, Mac, Linux, iOS, Android and Raspberry Pi. Point your internet browser to meetedison.com/downloads to obtain the installation files and instructions on how to install EdWare for your operating system.

Downloading a program

Once you have installed EdWare on your computer or tablet open the 'TestProgram.edw' file (File>open: EdWare/My Programs). A program that looks like the one below should appear.

Test program

Connect the EdComm cable to the headphone socket on your device and turn up the volume to full. If you are using Windows check that audio enhancements are disabled. To do this go into Control Panel and follow the illustrated steps below.

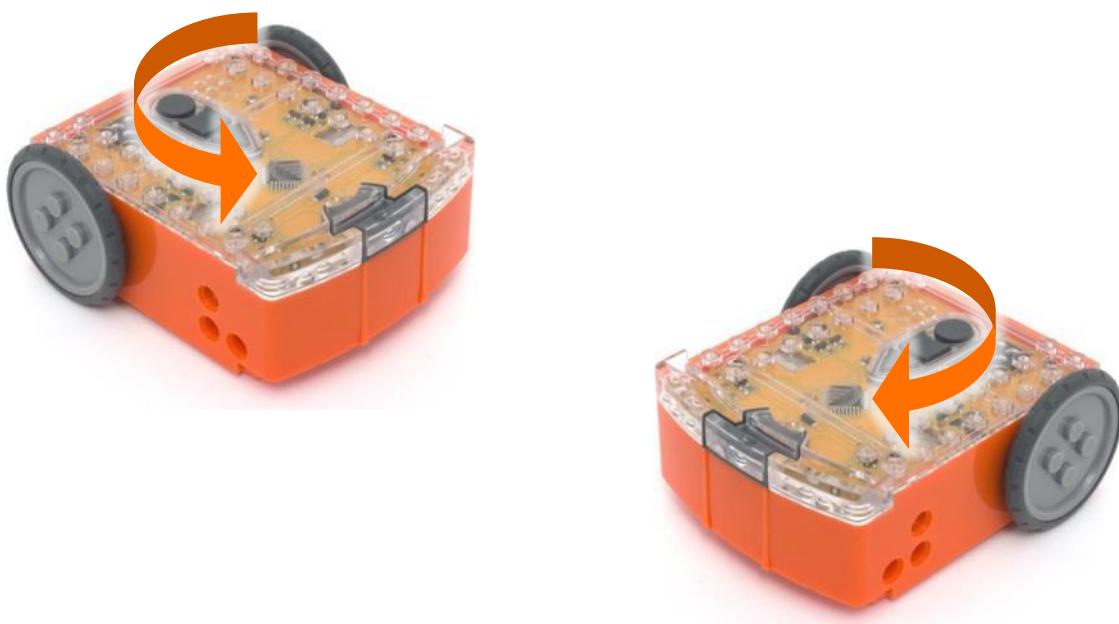How to disable all enhancements in Windows 7

Plug the other end of the EdComm cable into Edison as shown.

To download the test program follow these steps:

1. Press Edison's record (round) button once

2. In EdWare press the 'Program Edison' button and then 'Start Download'

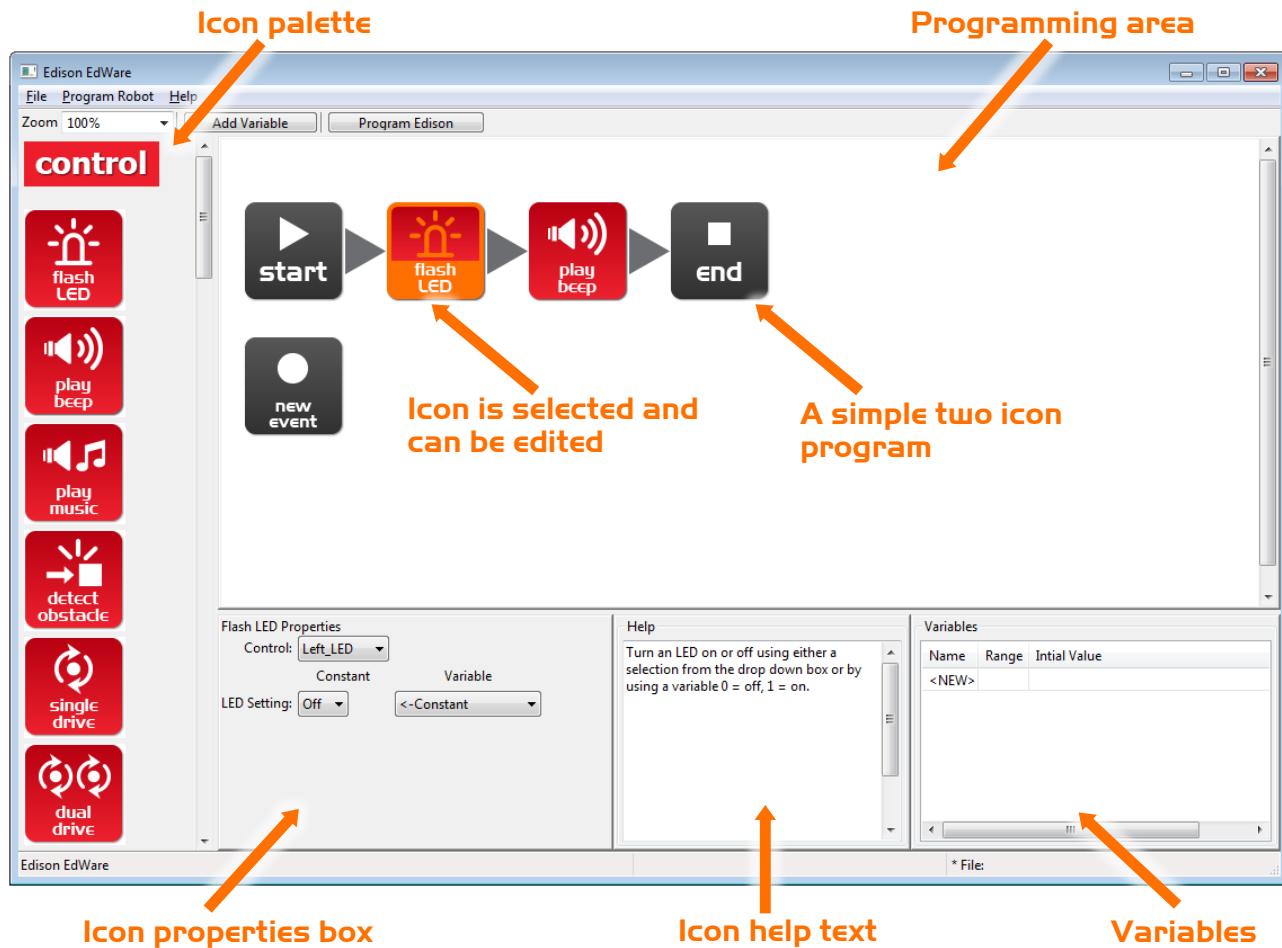3. Press Edison's play (triangle) button to start the program

Edison will now run the test program and spin left and right, flashing his lights and beeping.

Meet EdWare

Before we get into the EdVentures, let's take a quick tour of EdWare.

Here is what EdWare for Windows looks like. EdWare for the other operating systems looks very similar.

Icon palette

Programming area

Icon is selected and can be edited

A simple two icon program

Icon properties box

Icon help text

Variables

To start programming, grab icons from the palette on the left and drag them onto the programming area. Place the icons between the 'start' and 'end' icons.

Select an icon and adjust the settings in the icon property box to control how Edison responds to that icon.
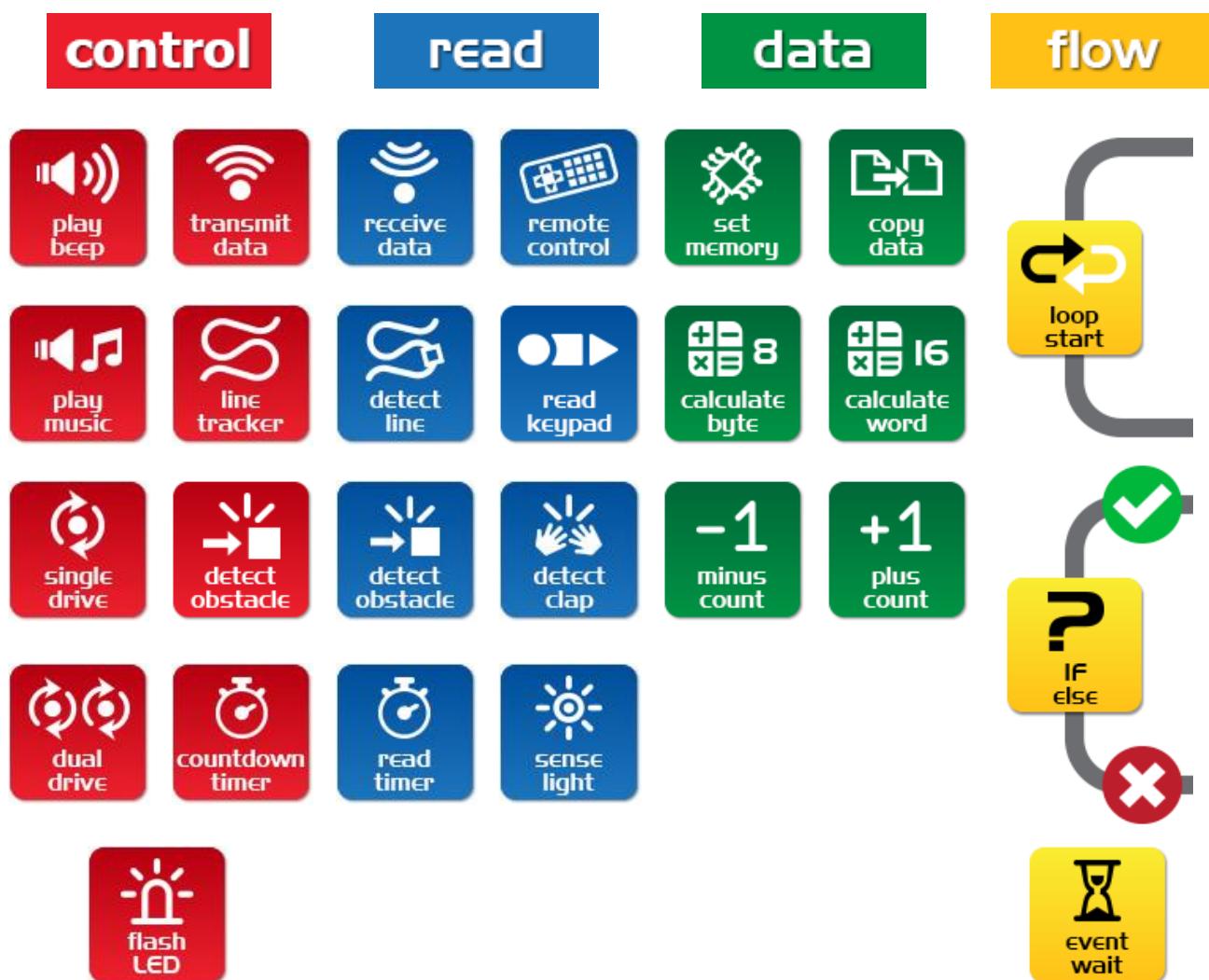
Use the help text as a guide while programming. Everything you need to know about an icon can be found here.

The variables section is where you can create and view little pieces of Edison's memory. More on this later!

EdWare icons

There are four main types of icons used in EdWare. These are control icons (red), read icons (blue), data icons (green) and flow icons (yellow).

Throughout this EdBook we will use almost every icon at least once. Some of the more advanced icons, such as the read icons will be covered in future EdBooks. For more information about all of the icons see the EdWare programming manual [due February 2015].
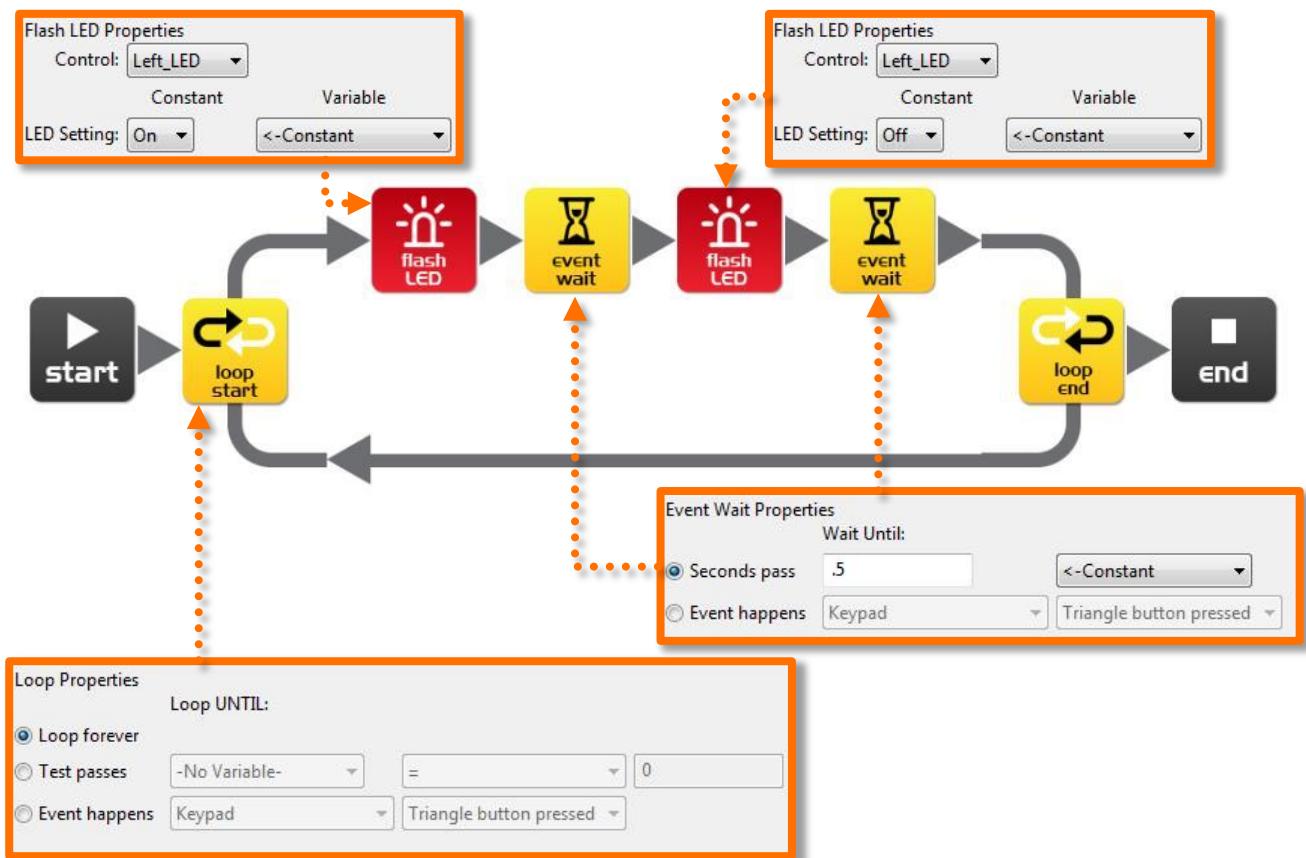
EdVenture 1 - Flash a LED

Simple loop program to make the left LED flash

The traditional first program to write is to make a LED light flash. This is a very simple program and as the name suggests turns on and off the left LED light.

Drag across the icons to form the program below, then click on each icon and set the properties box as shown.

How the program works

Edison's microcontroller (brain) follows each icon as a step and the arrows show the direction of the steps. Let's follow the steps in the program above.

Step 1: the program begins at the start icon

Step 2: the program follows the arrow out of the top of the loop icon

Step 3: the LED is set to on by the flash LED icon

Step 4: the program waits for 0.5 seconds by the event wait icon (keep the LED on)

Step 5: the LED is set to off by the flash LED icon

Step 6: the program waits for 0.5 seconds by the event wait icon (keep the LED off)

Step 7: what happens here is very important! Rather than following the arrow to the right of the loop end icon, the program moves out of the bottom of the icon and goes back to the loop start icon. This happens because the loop icon is set to 'Loop forever'. The program therefore once again goes to the first flash LED icon and turns on the LED and then follows the same sequence outlined above. This will continue *forever* or at least until the batteries go flat!

Download and play

Connect the EdComm cable between Edison and the headphone jack on your computer/tablet. Click the Program Edison button, then the record (round) button once on Edison. Now click Start Download.

Press the play (triangle) button and the left LED will flash on and off.

Congratulations! You have written and downloaded your first Edison program.

Experiment

Try adjusting the event wait times and adding more flash LED icons to control the right LED. Can you make a cool flashing light display?

EdFact

L.E.D. stands for Light Emitting Diode.

Unlike the original light bulb invented by Thomas Edison (no direct relation to your Edison robot) a LED has no filament or special wire that produces light when electricity passes through it. Instead, LEDs use advanced semiconductor material, just like that found inside computer chips.
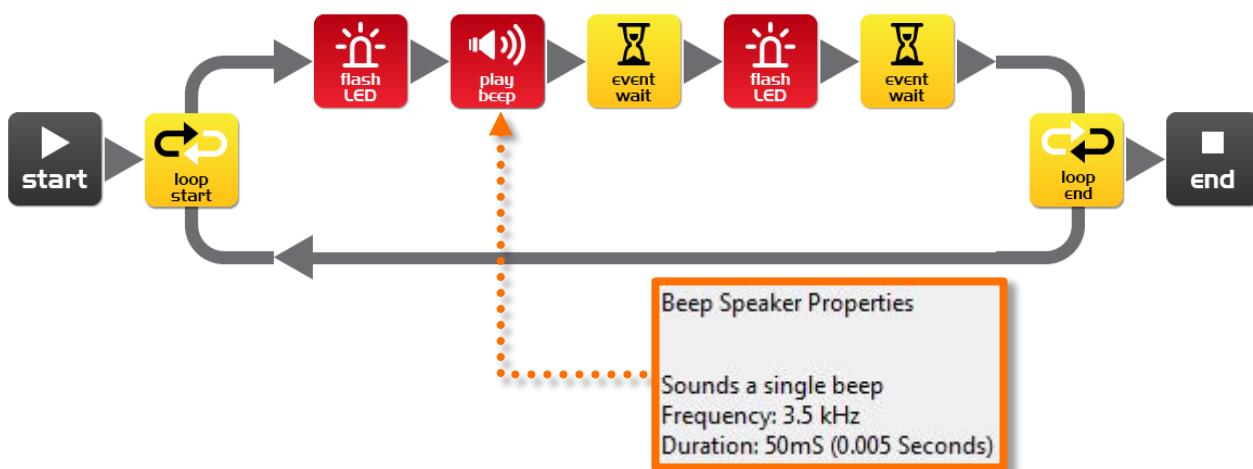
LEDs are superior to traditional light bulbs as they last longer, are more robust and use much less power.

So next time you have a great idea don't think of a light bulb coming on. Think of an LED coming on!

EdVenture 2 - Beep!!... Beep!!...

Add sound to the above program

Drag across the play beep icon and place it as shown below. Play beep sounds a short 50 millisecond (0.05 second) beep. There are no properties setting for play beep.

Beep Speaker Properties

Sounds a single beep
Frequency: 3.5 kHz
Duration: 50mS (0.005 Seconds)

How the program works

Just as the previous program goes around and around in a loop repeating every icon, this program is no different, except that when the LED comes on a beep is played.

Experiment

Try adding more play beep icons, changing the event wait periods and adding more flash LED icons. You can now add sound to your cool flashing light display!

EdFact

Speakers make sound by converting electrical signals into tiny backwards and forwards movements. These tiny movements cause small rapid changes in air pressure, which we call sound.

Edison has a slightly different type of speaker than you will find in a typical radio and is called a piezo transducer. The advantage of piezo transducers are that they use very little power, are inexpensive and can easily work in reverse by converting sound into an electrical signal. We'll look at that last point again when we have Edison respond to sound commands.
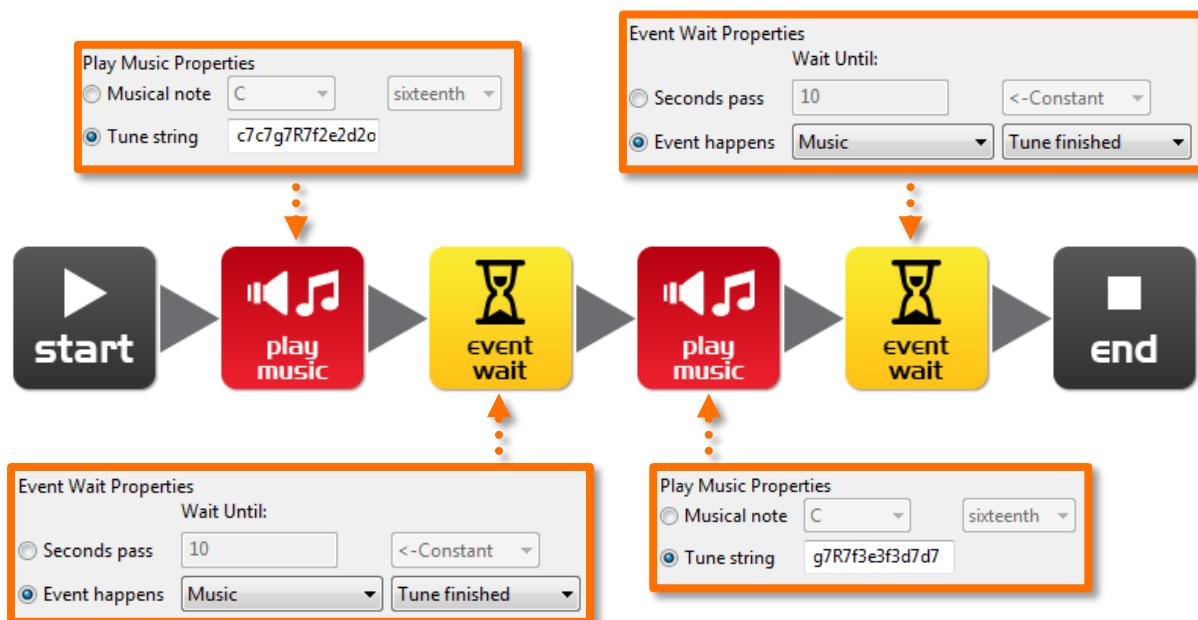
EdVenture 3 – Robots make music too!

Let's play a tune

Edison's little piezo transducer (speaker) can produce a range of musical notes. These are easily programmed from the play music icon. Get started by creating the program below.

The tune string in the first play music icon is: c7c7g7R7f2e2d2o7o7g7R7f3e3d3o7o7

The tune string in the second play music icon is: g7R7f3e3f3d7d7

How the program works

The play music icon contains a tune string (more on that below) and the event wait icon waits until the tune is finished. The next play music icon and event wait do the same. There are two sets of icons because a single play music icon will not fit the entire tune.
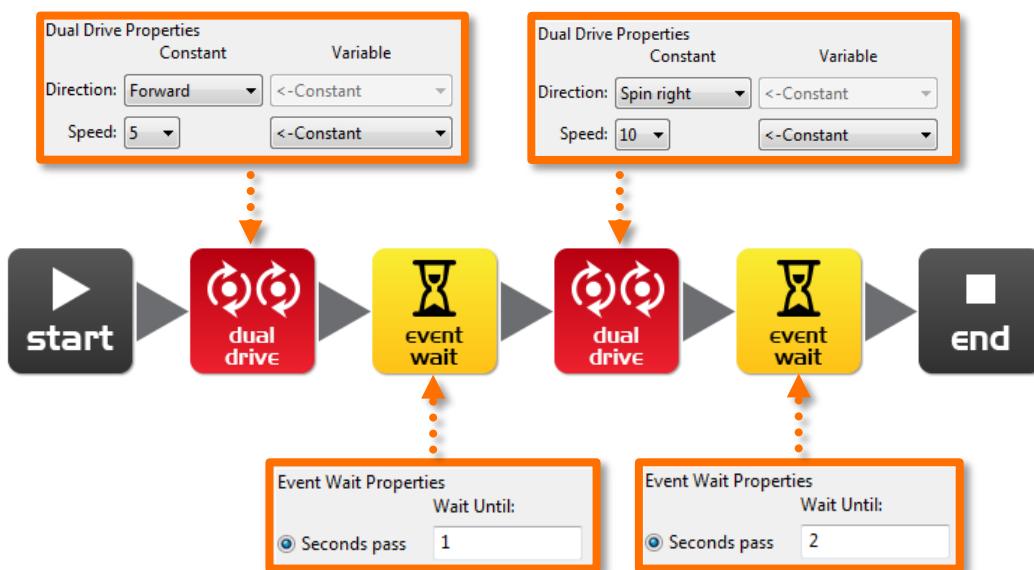
Can you guess what the tune is?

A tune string looks like this: "ndndndndndnd..." where 'n' is a note from the following table, and 'd' is the note time from 0 to 7 in 20th of a second increments. There is a limit of 16 pairs per play music icon.

character	note	character	note	character	note
m	A, 6th octave	d	D	g	G
M	A#	D	D#	G	G#
n	B	e	E	a	A
c	C, 7th octave	f	F	A	A#
C	C#	F	F#	b	B

EdVenture 4 – Let's get moving

Drive Edison in a pattern

Edison has wheels, so let's us them! Drag across two dual drive icons and two event wait icons and arrange them as shown. Now, set the four properties boxes with the shown settings.

How the program works

The dual drive icons control both motors that turn the wheels. The first dual drive icon is set to 'Forward' and at speed 5 (half speed). The next event wait icon is set to 1 second, so the program waits there for 1 second. While the program is waiting Edison is driving forward. The next dual drive icon changes Edison's direction to spin right at speed 10 (full speed). The final event wait icon waits for 2 seconds while Edison goes around and around. Once the 2 seconds is up the program goes to the end icon and stops.

Experiment

This is a very simple program to get Edison to drive. You are encouraged to add more drive icons and event wait icons and see what he can do! *Can you make him dance?*

Mission

Using LEGO bricks create a short maze for Edison to travel through. Then write a program that drives him through the maze without hitting the sides.
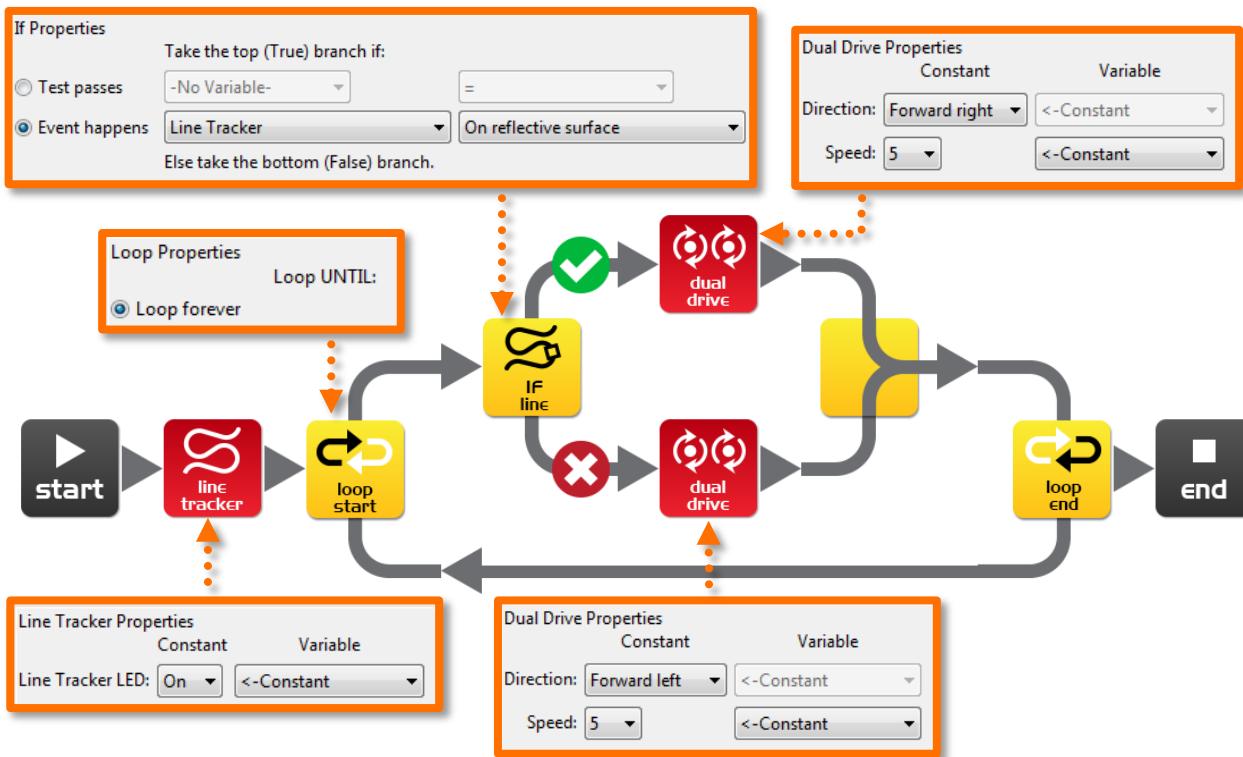
EdFact

There are many types of drive systems that robots can use, some even walk. Edison's drive system is called a '*differential drive*' and it's one of the most common systems that robots use. This is because it allows the robot to move in any direction and requires very few mechanical parts to work.

EdVenture 5 – What IF?

Give Edison the power to make decisions

The introduction said that a robot can think or make decisions on its own and act on those decisions. Well, now were going to see how that happens using the IF icon.

Create the program and download it to Edison. Place him next to a thick black line on a white surface and press the play button. He will follow the line.

How the program works

The first thing the program does is turn on the line tracking LED. Next, the program enters an endless loop. In the loop is the, all important, IF icon. The IF icon asks: *Is the line tracker on a reflective surface (white)?* If the answer is *YES,* then the program follows the tick path out of the top of the IF icon. The dual drive icon turns Edison right. This drives him onto the black line. But, if the answer is *NO*, then the program follows the cross path out of the bottom of the IF icon. This dual drive icon turns Edison left and drives Edison off the line. The program then loops around again, and again, and…

Edison is in a constant battle with himself. When he's on the line, he wants to get off it. When he's off the line he wants to get on it. Frustrated movement by movement he moves forward.
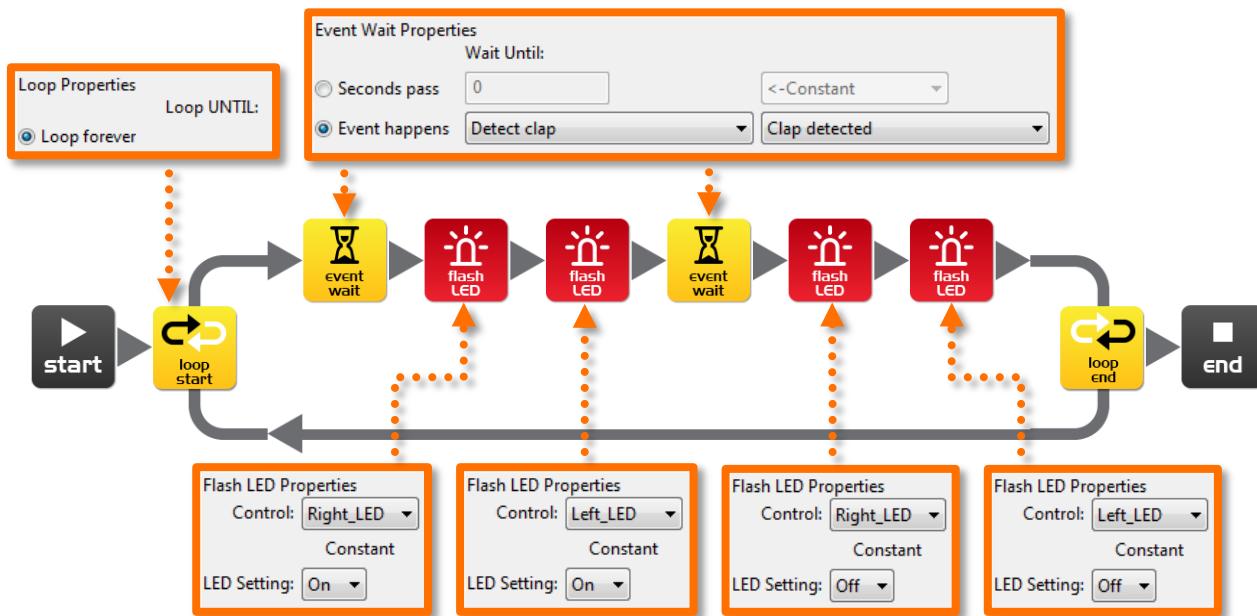
EdFact

This program is very simple, but demonstrates the principal of 'artificial' decision making (intelligence). Scientist still don't fully understand how the human brain works and are now using computer scientists to help make sense of their research. *Do you think your brain is a giant computer?*

EdVenture 6 – What's the hurry? Let's wait!

Meet the *event* in the event wait icon

Edison can wait for more than just time. He can wait for a specific event to occur before progressing through your program. This program takes advantage of Edison's clap sensor.

How the program works

Once again we have a loop, which you should be familiar with now. The first icon in the loop is the event wait icon and is set to wait until a clap is detected, so the program will wait here until a clap is detected. Once this happens, the program progresses to turn on the right and left LEDs and encounters another event wait icon with the same settings as the first one. When another clap is detected the program progresses to turn off the right and left LEDs, and loops around again.

Important!

The clap detection sensor becomes overwhelmed with noise when the motors are running, so you can't detect for claps while Edison is driving.
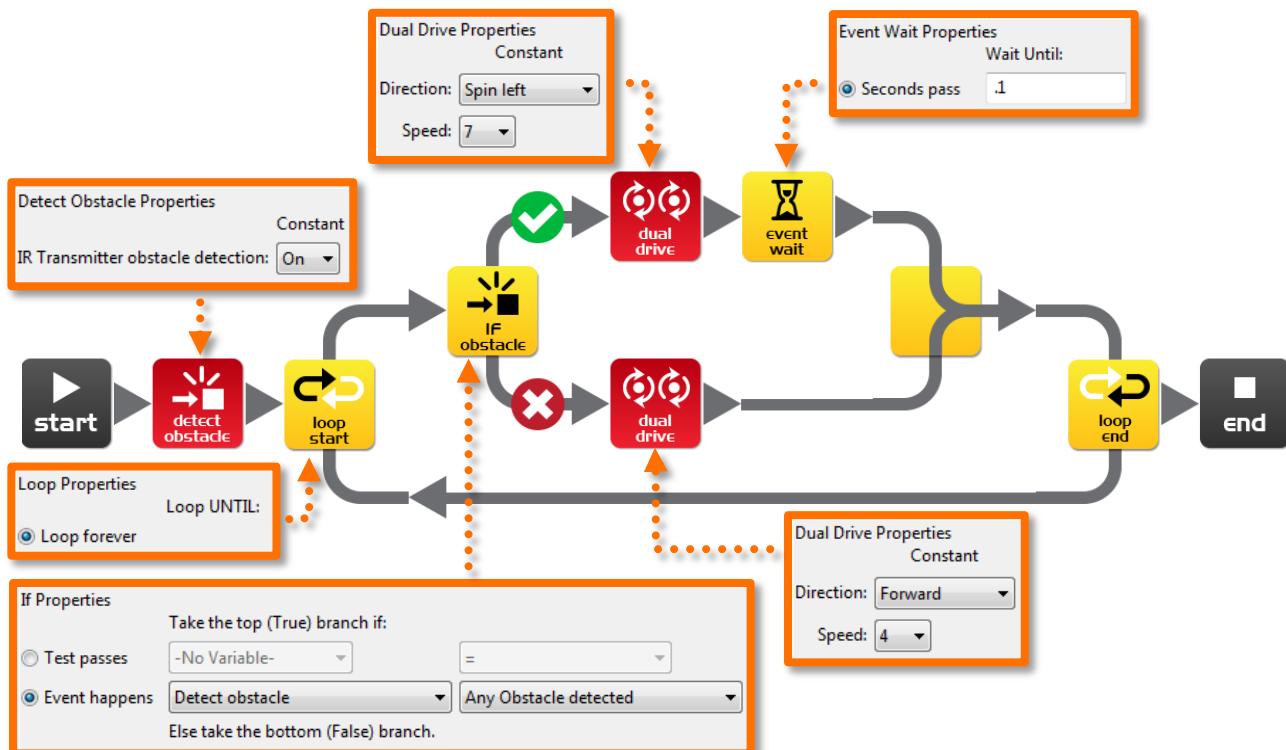
EdFact

As mentioned earlier Edison uses an electronic component called a piezo transducer to both make sound and detect claps. Piezo transducers have two plates that are separated by a thin layer of ceramic. When an electrical signal is applied to the two plates they move, which produces sound. They also work in reverse, when sound or vibration is applied to the plates they produce an electrical signal.

The name piezo comes from the Greek language meaning press or squeeze and a transducer is anything that converts one form of energy into another (i.e. sound energy into electrical energy and electrical energy into sound energy).

EdVenture 7 – Watch out! There's an obstacle!

Let's do some autonomous driving

Edison is equipped with an obstacle detector, so he can *see* obstacles in his path and avoid them. Here's a simple program that does this.

How the program works

Before entering an endless loop, the program turns on Edison's obstacle detection system. Edison now emits infrared (IR) light from two light emitting diodes (LEDs), one on the left and one on the right. In between the two LEDs is an IR sensor. The sensor detects when IR is reflected from an obstacle. If the IR is reflected from the left LED then the obstacle is on the left. If the IR is reflected from the right LED, then the obstacle is on the right.

The IF icon asks; *'have any obstacles been detected'?* If the answer is no, then the cross path is taken and Edison drives forward. If the answer is yes, then the tick path is taken and Edison spins left for 0.1 of a second (100 milliseconds). See next page for calibration.

Experiment

Try detecting for obstacles on the left and right. See if you can add more IF icons and have Edison spin left to avoid obstacles on the right and spin right to avoid obstacles on the left.

EdFact

I'm sure you've heard of, or even seen, robot vacuum cleaners; well they use the same IR system as Edison to detect obstacles. The 'Roomba' has two of these sensors. One is used just like Edison to detect obstacles, and the other is used as a 'cliff detector' and looks at the ground in front of the robot to ensure it isn't about to dive down a staircase.

Calibrate obstacle detection

You can control the sensitivity of Edison's obstacle detection system. By making it more sensitive he can detect obstacles further away, and by making it less sensitive he will only detect very close obstacles.

Reading the barcode

1. Place Edison facing the barcode on the right side
2. Press the record (round) button 3 times
3. Edison will drive forward and scan the barcode

Barcode – Calibrate obstacle detection

Set maximum sensitivity

First scan the above barcode, then press the play (triangle) button. Edison is now in calibration mode. Remove any obstacles in front of Edison.

The left sensitivity is calibrated first.

1. Repeatedly press the play button (this increses sensitivity) until the left red LED is flickering.

2. Repeatedly press the record (round) button (this decreases the sensitivity) until the LED completely stops flickering.

3. Press the stop (square) button to calibrate the right side.

4. Repeatedly press the play button until the right red LED is flickering. Now repeatedly press the record button until the LED completely stops flickering.

5. Press the stop button and the calibration is complete.
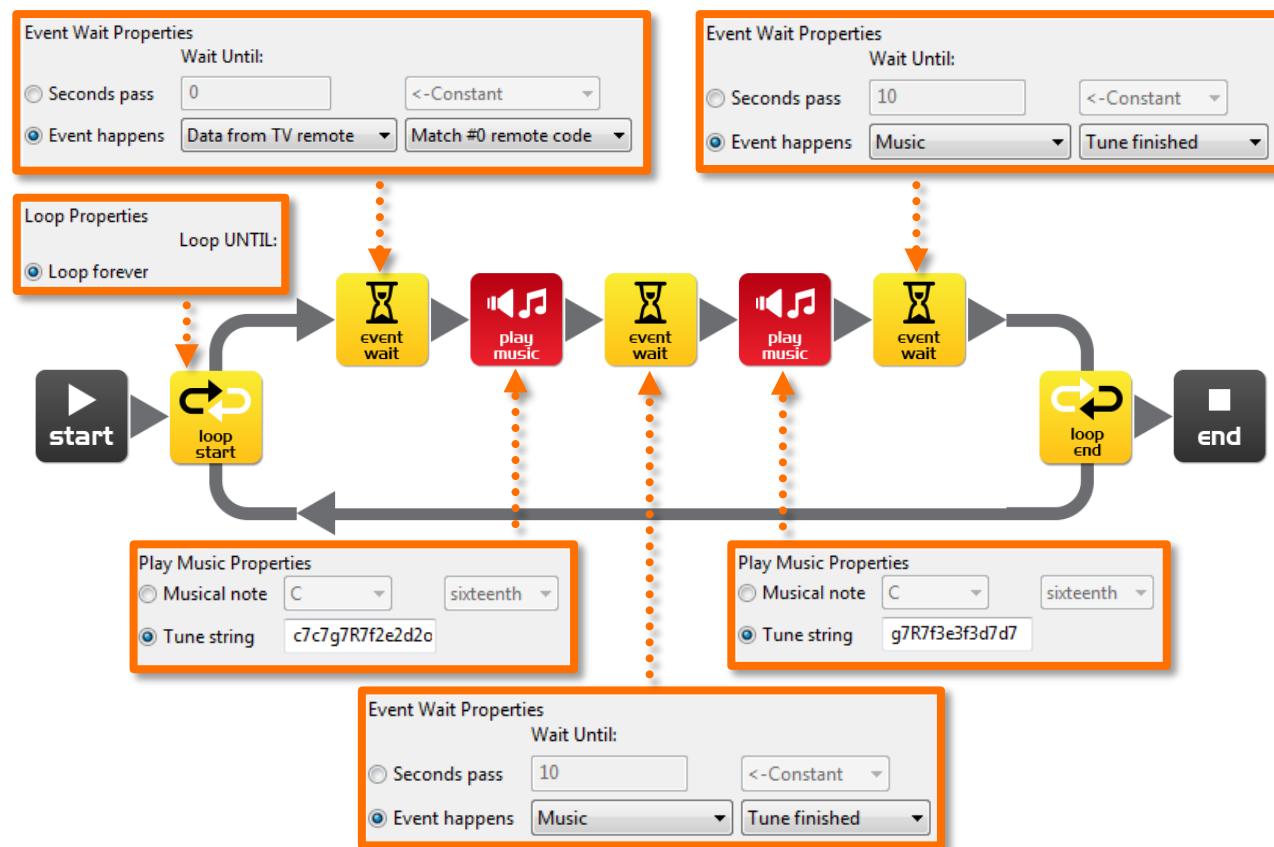
Custom sensitivity

You can set the distance that obstacles are detected by placing an obstacle in front of Edison and repeating steps 1 through 5.

EdVenture 8 – Take control!

Don't just take control, *take remote control!*

Use a TV or DVD remote control to control your program and play Star Wars at your command.

Write the following program. Note that it uses the play music and event waits from EdVenture3.

Getting ready

Before you program Edison turn to the next page and scan in barcode #0. Now download your program to Edison.

How the program works

Right away the program enters a loop, but it doesn't get very far. The event wait icon won't allow the program to progress until the infrared code #0 is received. Once this is received the program can continue and play the Star Wars theme.

EdFact

Edison learns the infrared command signals from your TV/DVD remote control. He stores them in his memory and when he detects and receives a new command he compares it to what he has stored. If there is a match he activates that function.

TV remote control barcodes

Edison can respond to a TV/DVD remote control within a program. Here are the barcodes with their ID numbers for use in EdWare. Note these are the same remote control barcodes that are used to remote control drive Edison.=
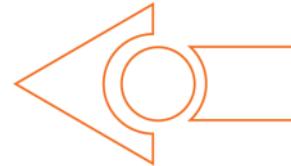
Reading the barcode

1. Place Edison facing the barcode on the right side
2. Press the record (round) button 3 times
3. Edison will drive forward and scan the barcode
4. Press a button on your TV/DVD remote that you want to activate that function
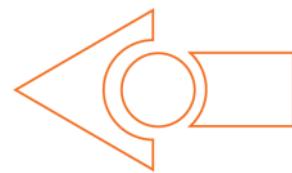
Barcode – TV/DVD remote code #0
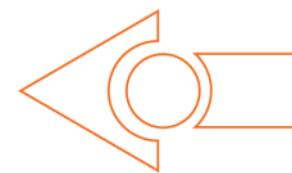
Barcode – TV/DVD remote code #1

Barcode – IR learn turn right #2
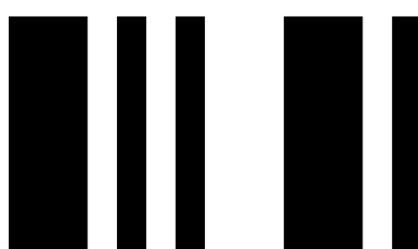
Barcode – IR learn turn left #3

Barcode – IR learn spin right #4

Barcode – IR learn spin left #5

Barcode – IR learn play beep #6

Barcode – IR learn play tune #7

EdVenture 9 – Change is good, let's get variable!

Edison can guard your stuff

We can use Edison's light sensors to trigger an alarm. This could be used as a draw alarm. Place Edison running this program in a draw, then if someone opens the draw and allows light in, Edison sounds the alarm.

Before we look at writing the program you will need to understand what a '*variable*' is and how to use it.

A variable is small piece of computer memory for storing data. What makes variables so useful is that this data can change while the program is running, hence the name variable.
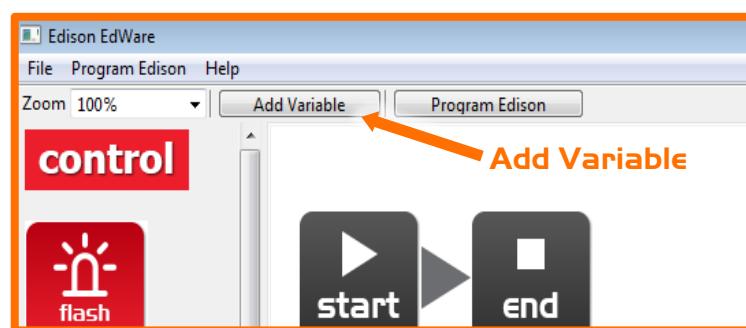
Variables		
Name	Range	Intial Value
Fred	0-255	162
Light_Level	+/- 32767	0
<NEW>		

Variables store numbers like 10, 106, 1,482 etc. and allow a computer program to do maths, this is something that computers are very good at.

Edison has two types of variables that are called '*bytes*' and '*words*'. Byte variables can store numbers from 0 to 255. Word variables can store numbers from -32,767 to +32,767.

To make variables easy to use we give them names. This helps us humans remember what type of information is stored in them. In EdWare, you can name your variables just about anything you like. You could name one '*Fred*', but that might not be a very helpful name for remembering what type of information is stored in Fred. A better name might be '*Light_Level*'. This type of name makes it very easy to remember what the variable is used for and what type of data to find there.
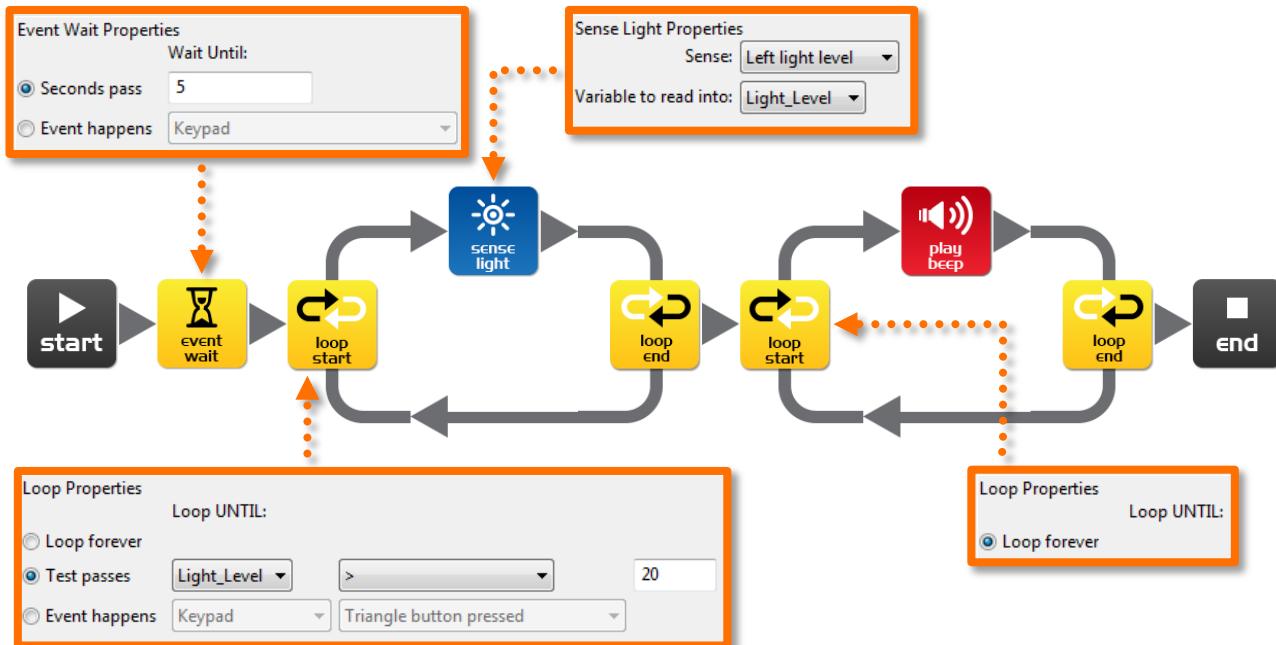
Now that you know about variables let's create one in EdWare for our light alarm program. Click the 'Add Variable' button in the top left corner and a pop up box will appear.

Type in the name of your variable as 'Light_Level', select the variable range as +/-32767 and set the initial value to zero. Now click OK and your variable will be added to the variables table in the bottom right corner.

Now that we have our variable we can use it in our program. Write the program below.

How the program works

The first icon is an event wait icon and pauses the program here for 5 seconds *[enough time to press the play button and place Edison in a draw]*. Next the program enters a loop, but this time it's not forever. Inside the loop, the sense light icon is measuring the light level from the left light sensor and putting it into our 'Light_Level' variable. The loop is set to end when the value in our 'Light_Level' variable is greater than (>) 20 *[when someone opens the draw and lets light in]*. When the loop is exited the program goes to another loop. This one loops forever and has a beep icon inside *[raises the alarm! There has been an intruder!]*.

Just for fun

This program is just for fun and learning about electronic systems and programming. Leaving Edison in a draw switched on for more than 20 hours will significantly reduce the batteries, so unfortunately it's actually not a practical alarm system.

EdFact

Edison has three different types of memory.

1. Flash – Your programs are stored here, and is similar to a hard drive in a computer. This is also called non-volatile memory as data is not lost when the power is turned off.

2. EEPROM – Electrically Erasable Programmable Read-Only Memory is where Edison's settings and IR remote control codes are stored and is also non-volatile memory.

3. RAM – Random Access Memory is where the variables are stored. This is volatile memory and is lost if the power is turned off.
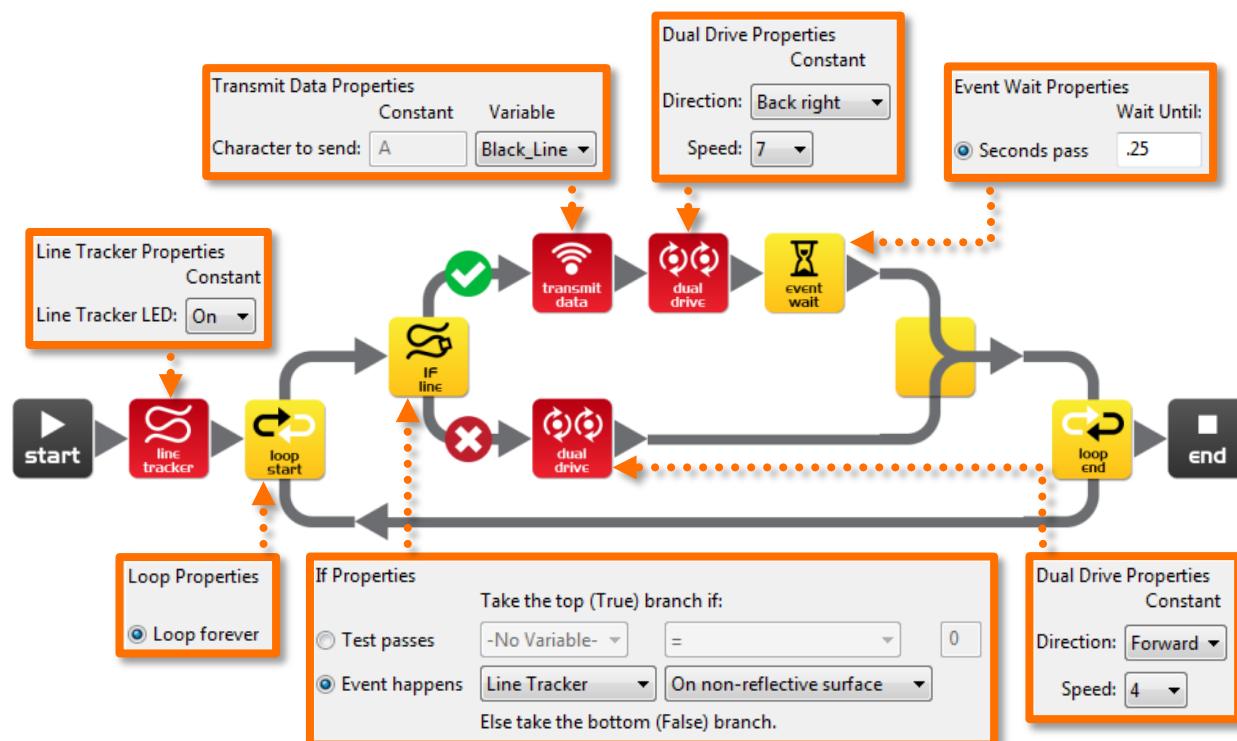
EdVenture 10 – Robot communications

Bounce *without* borders

A robot swarm is when many robots work together to solve a common problem. Generally the robots are small and by themselves aren't very intelligent, but as a swarm they can achieve complex tasks (*think of ants building a complex nest*). The most important part of swarm robotics is communication. Without robot to robot communication a robot swarm is not impossible. Here is an introduction to robot communication.

In this EdVenture you will need at least two Edison robots. The first Edison will bounce within a border and tell the second robot each time a border is encountered. The second robot will use this information to mimic the first robot and will seem to bounce in side an invisible border.

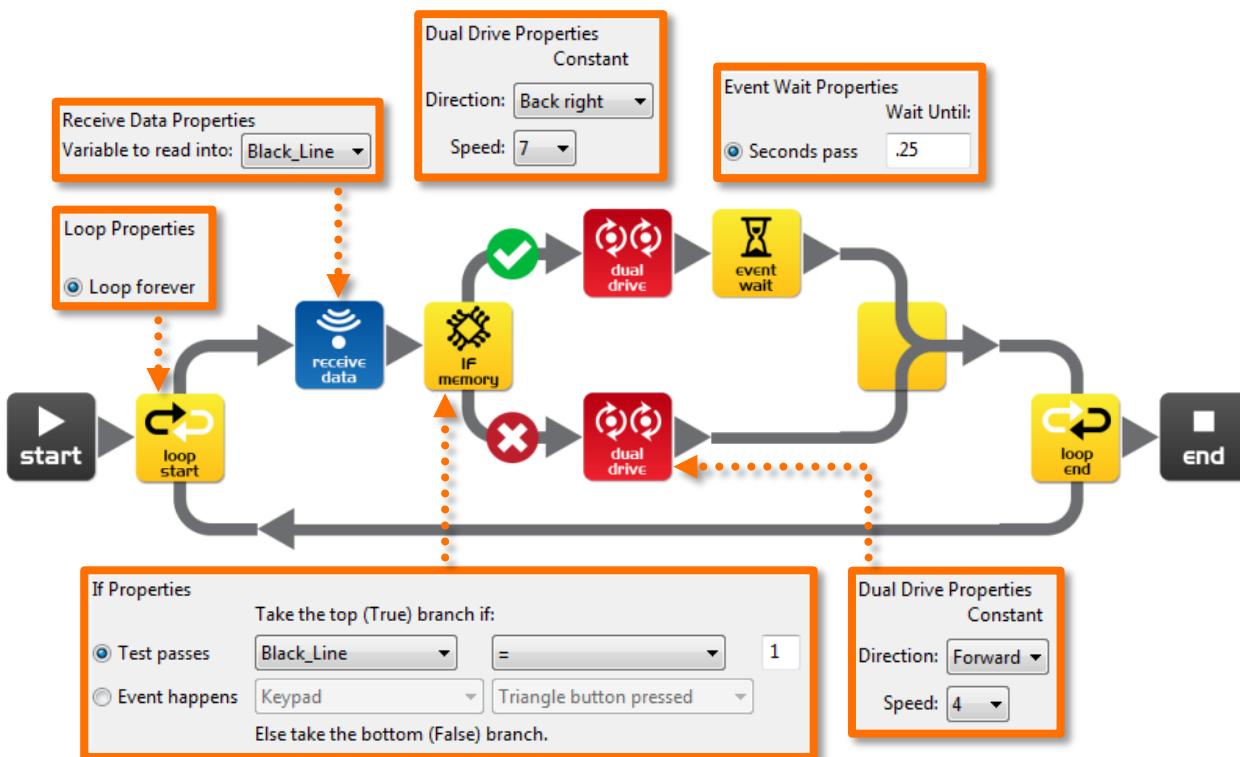Write the following program for the first Edison (Edison with borders).

Create an 8 bit variable called 'Black_Line' and set the initial value to 1.

How the program works

The first icon turns on the line tracking sensor LED. The program then enters an endless loop. The IF icon in the loop checks the state of the line tracking sensor, if the surface is white (reflective) then the dual drive icon drives Edison forward at speed 4. If the surface is black (non-reflective) then the contents of the variable 'Black_Line' (1) is transmitted. Edison then reverse-turns for 0.25 seconds and repeats the loop.

Write the following program for the second Edison (Edison without borders).

Create an 8 bit variable called 'Black_Line' and set the initial value to 0.

You can program a third, fourth or fifth Edison with this program for more fun.

How the program works

The program goes straight into an endless loop and reads incoming data using the receive data icon. The receive data icon places the data into the variable 'Black_Line'. The IF icon then checks if the data in the variable is equal to 1. If it isn't (no data from the other Edison) then the dual drive icon drives Edison forward at speed 4. If the data does equal 1 (the other Edison has encountered a line) then the dual drive icon drives Edison in a reverse-turn for 0.25 seconds. The loop is then repeated.
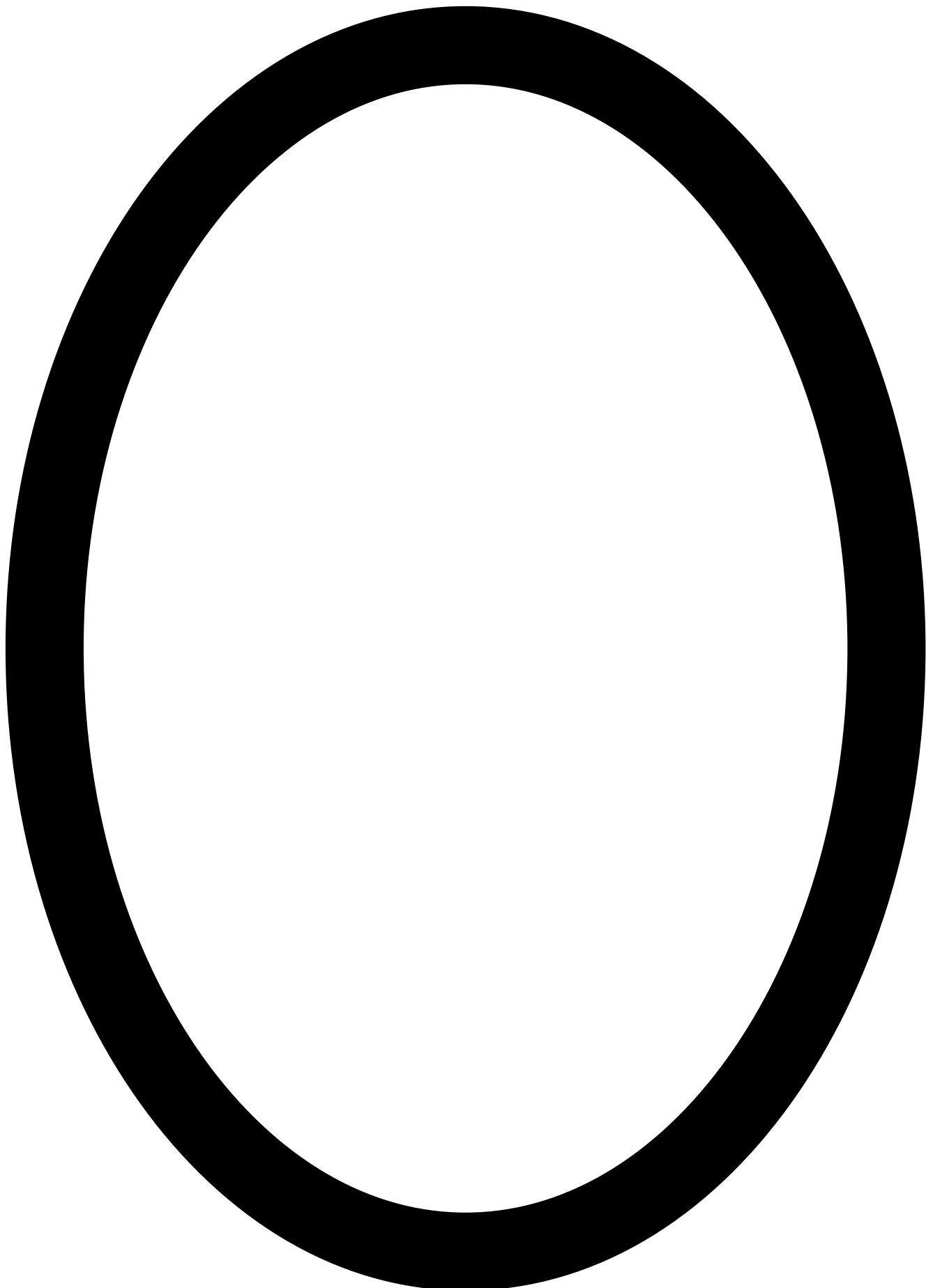
What to do

Print the track on the next page and place the first Edison inside the border. Place the second Edison outside the border and press the play button on both Edisons.

The Edison inside the border will not leave the oval and the second Edison will mimic the firsts every move.

Experiment

This is just an introduction to robot communications. Can you improve on this program and can you achieve two way communications? What if the first Edison couldn't turn until the second Edison confirmed that it had received data?

What's next?

You now have a good understanding of how to program Edison. You can now use this knowledge to create your own programs. *What can you dream up?*
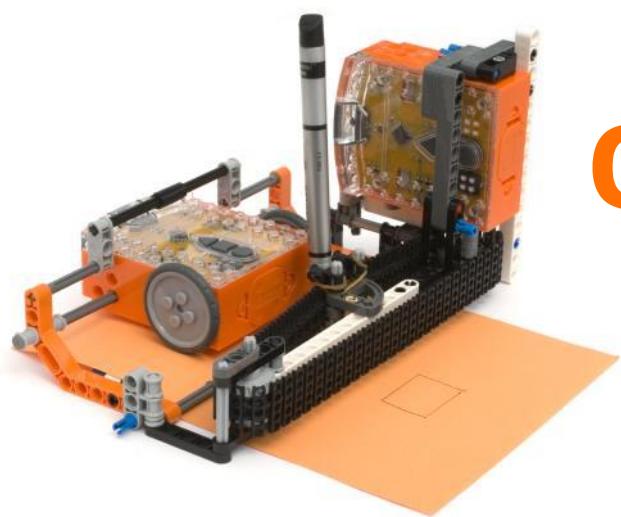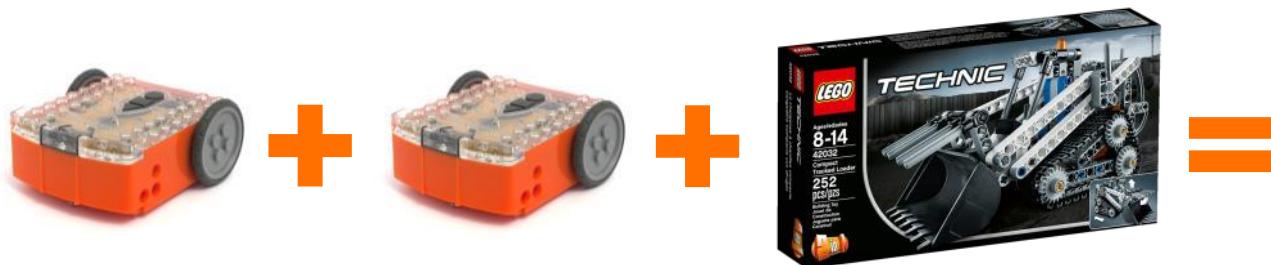
A few challenges

Can you write a faster line tracking program that can beat the barcode activated one around a track?

What tricks can you get Edison to do in response to your claps?

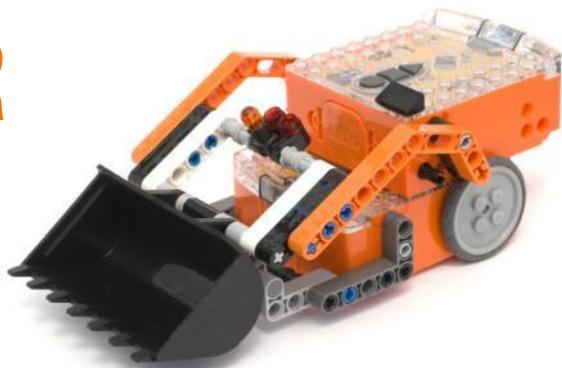Can you write a better Sumo program that can beat the barcode activated one?

EdBook3

In EdBook3 'Your EdVenture into Robotics - *You're a Builder'* you can combine two Edison robots with the LEGO Compact Tracked Loader (set 42032) to build new contraptions.

OR

EdPrinter   EdDigger

LEGO (R) is a registered trademark of The LEGO Group