techcamp.org.uk

D - Transport the Nuclear Waste

We need to transport some very unstable nuclear waste across the planet, so we must program the robot to move as smoothly as we can.

This document was generated on 2022-01-04 12:26:32 AM (MST).

INTRODUCTION

We need to transport some very unstable nuclear waste across the planet, so we must program the robot to move as smoothly as we can.

Step 1

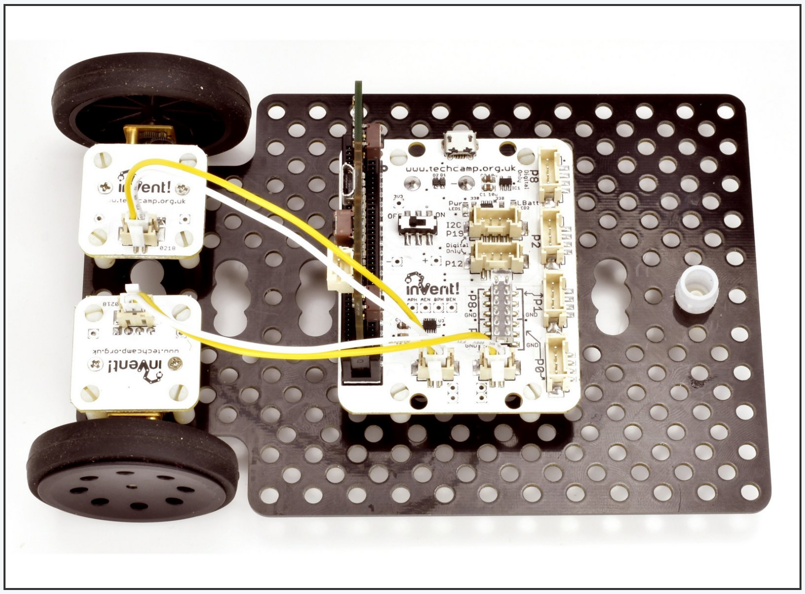
Nuclear Waste

- Some **nuclear waste** has been found near the base, and we need to move it to the other side of the planet as it is **very dangerous.**
- The nuclear waste is **extremely unstable**, so we need to make our robot **accelerate** and **decelerate smoothly** so it doesn't explode!
- To do this, we are going to learn about **variables.**

Step 2

Assemble the Robot

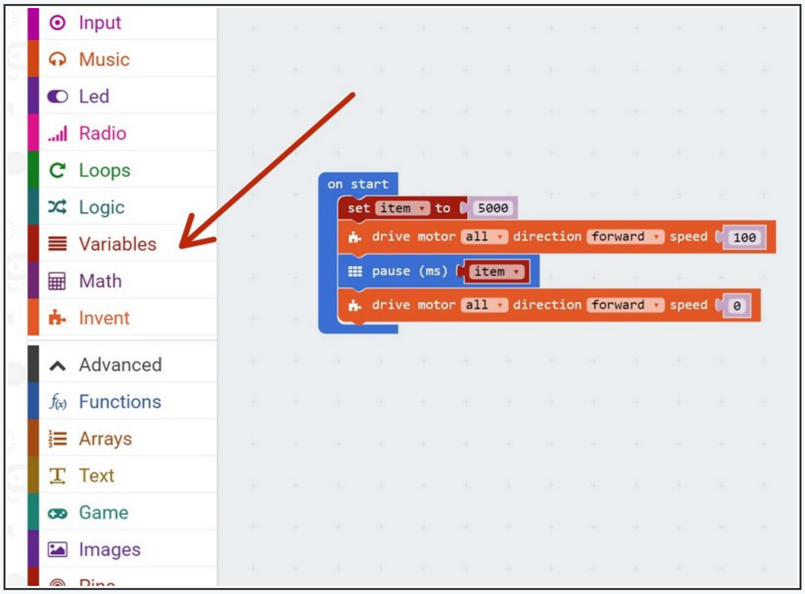
- Put your robot together just like the picture! The connections should be:
  - Left Motor > **M1**
  - Right Motor > **M2**



Step 3

Test Program

- **Build** the program in the picture!
- You can find all the blocks you need in the **variables menu.**
- **Before** you try programming your robot, **what do you think this program will do?**



```
on start
    set item to 5000
    drive motor all direction forward speed 100
    pause (ms) item
    drive motor all direction forward speed 0
```

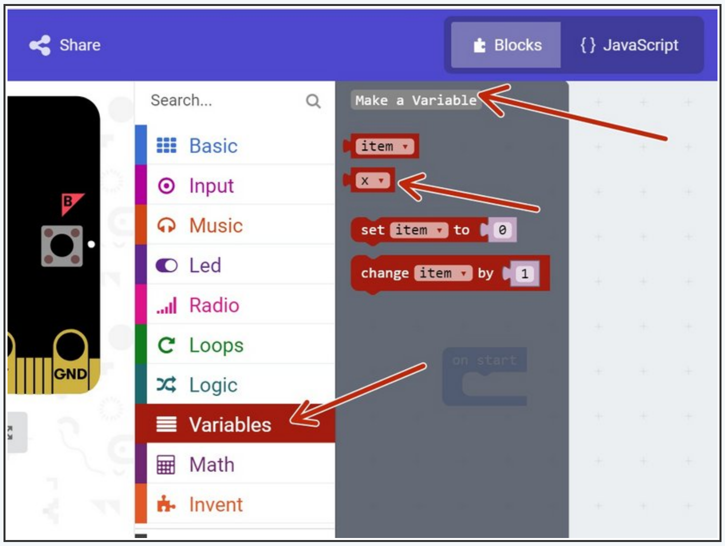
Step 4

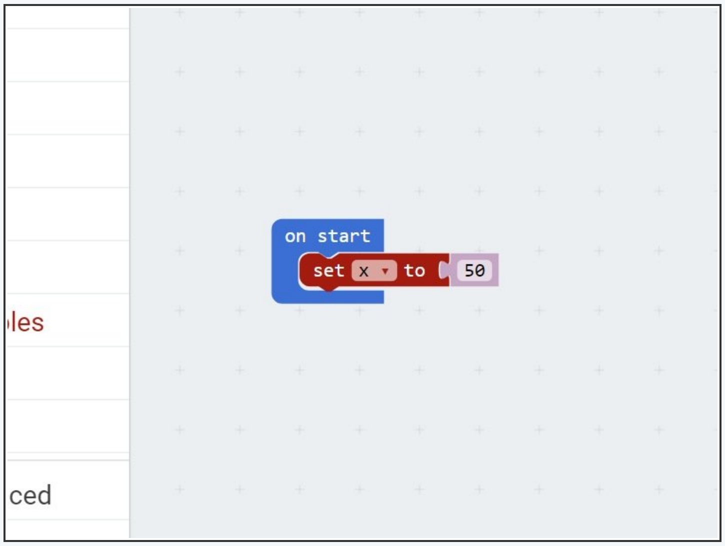
What are Variables?

- A good way to understand variables is to think of them like your **lockers** at school.
- To use a locker, you need to put your **name** on it - we do the same with a variable, and you can call it **anything you want!**
- We can then put **whatever we like** inside the locker - books, bags, clothes, anything! We can do the same with variables, but for now we'll just put **numbers** inside them.
- We can **add, remove and change things** in the locker whenever we like - its the same for the number in the variable!
- Most usefully, we can go back to the locker or variable at any time and **see what's in it** (so long as we know the **name** of the locker or variable!).

Step 5

Using Variables





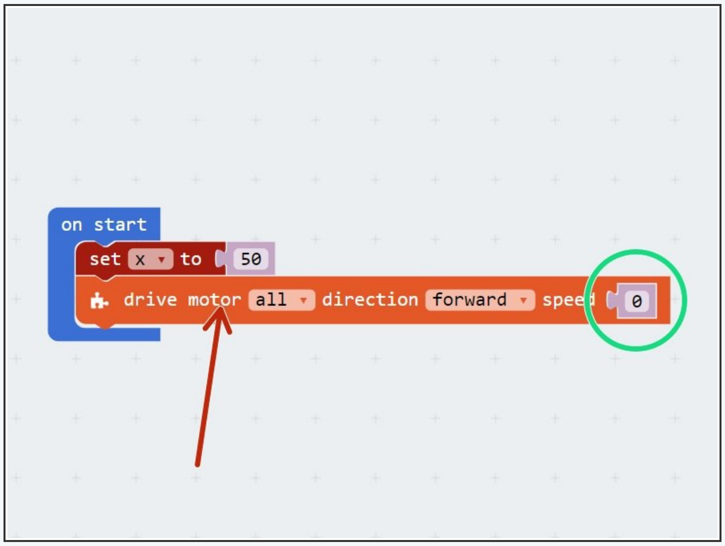
```
on start
  set x to 50
```

- If you don't quite understand, don't worry - for now, just remember we can do these things with variables:
  - **Call** them anything we like (variable **name**)
  - **Store** any number we like inside them (variable **contents**)
  - **Change** the contents at any time (add, subtract, multiply, divide and so on)
  - **Access** the contents at any time, so long as we know the **name** of the variable.
- Let's start a new program by **calling** a new variable **x** (click on **Make a Variable**), and **setting** it to **50**.

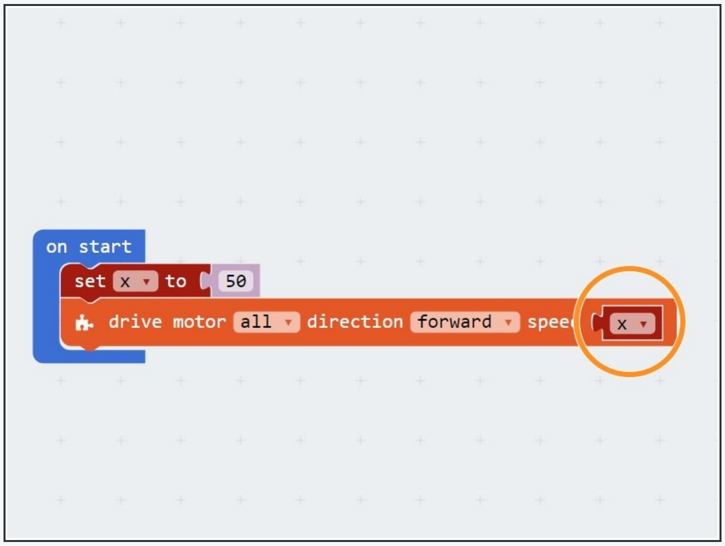
This document was generated on 2022-01-04 12:26:32 AM (MST).

Step 6

Accessing Variables



```
on start
  set x to 50
  drive motor all direction forward speed 0
```

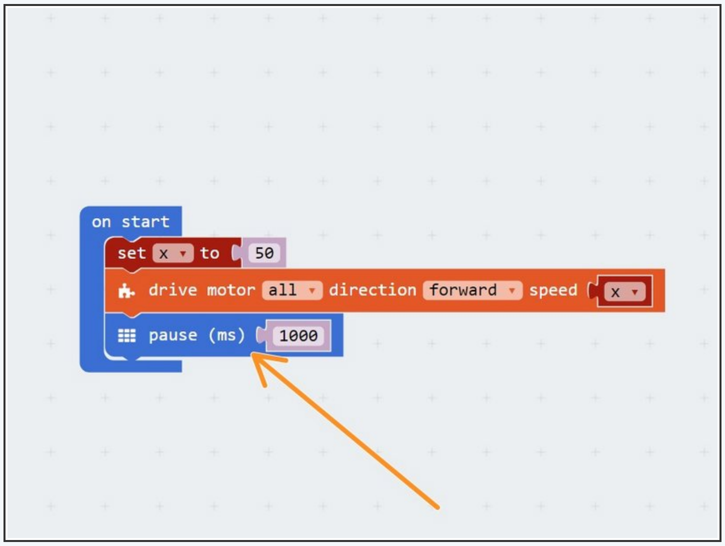


```
on start
  set x to 50
  drive motor all direction forward speed x
```

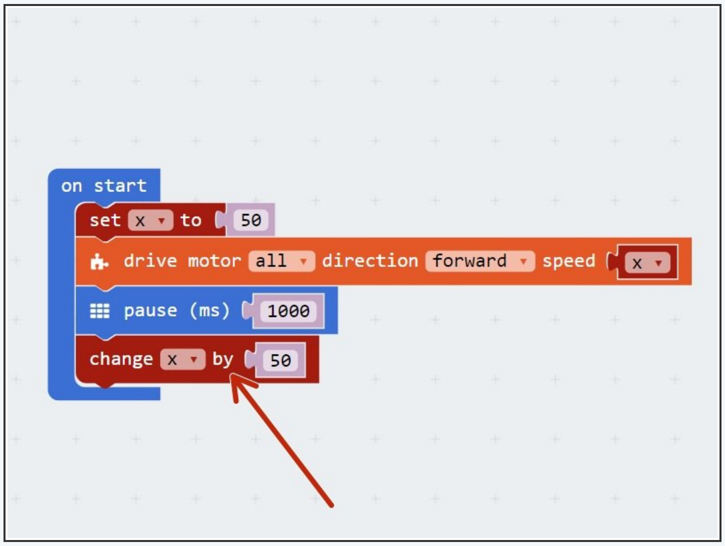
- Let's **access** the number inside our variable, **x,** and use it to **turn on the motors.**
- Drag in a **motor block** and make it set both motors going **forwards**.
- Wherever there is a **number** in a block that we can change, we can use a **variable** in its place if we want to.
- Replace the motor speed with an **x block**, so the motor speed is set by the **number** in the **variable x**

Step 7

Changing Variables

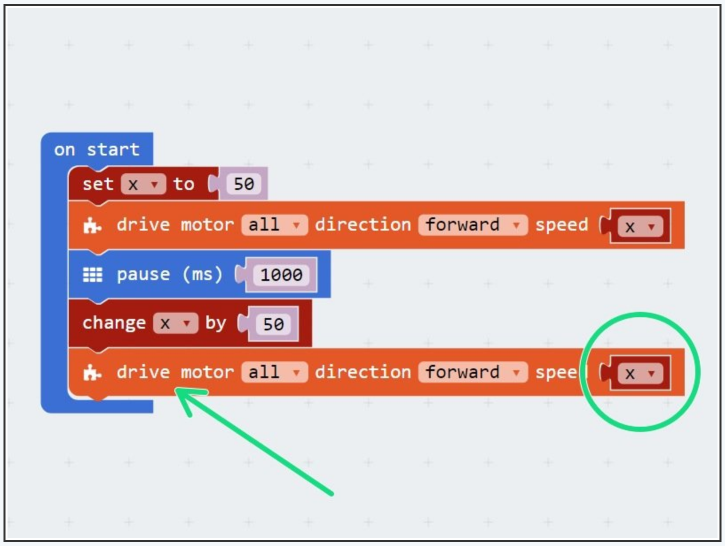


```
on start
    set x to 50
    drive motor all direction forward speed x
    pause (ms) 1000
```



```
on start
    set x to 50
    drive motor all direction forward speed x
    pause (ms) 1000
    change x by 50
```

- Now let's try **changing** the motor speeds by **changing the variable!**
- First, **add a pause block** so the robot moves forward at the first speed **(50)** for **1 second.**
- Now change the variable by using a **change block** to increase x by **50.**
- Finally, add **another motor block** to set each motor to speed **x** again - try it out, the robot should **change speed** this time!

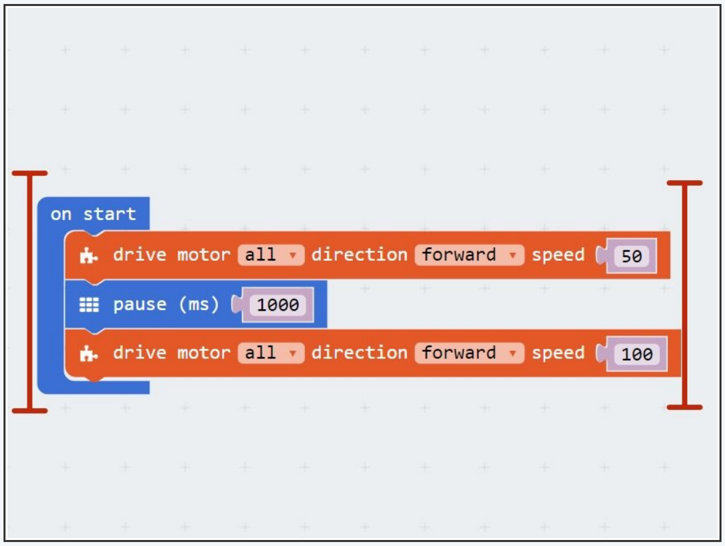


```
on start
    set x to 50
    drive motor all direction forward speed x
    pause (ms) 1000
    change x by 50
    drive motor all direction forward speed x
```

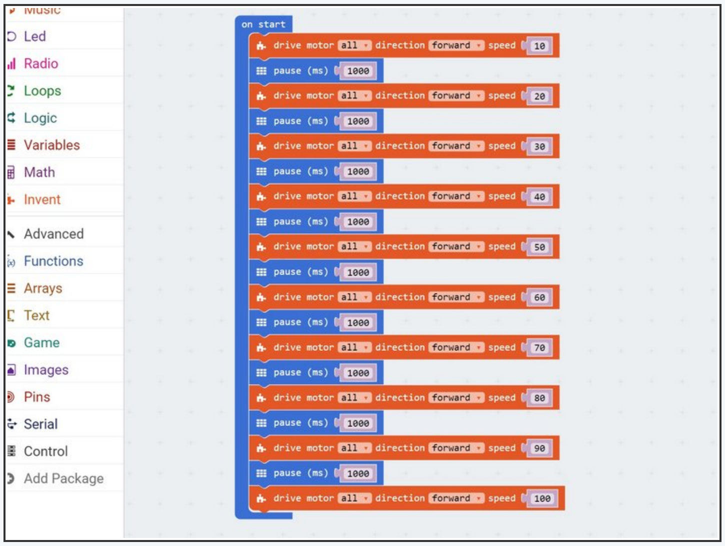
This document was generated on 2022-01-04 12:26:32 AM (MST).

Step 8

Why do we need variables anyway?



```
on start
    drive motor all direction forward speed 50
    pause (ms) 1000
    drive motor all direction forward speed 100
```



```
on start
    drive motor all direction forward speed 10
    pause (ms) 1000
    drive motor all direction forward speed 20
    pause (ms) 1000
    drive motor all direction forward speed 30
    pause (ms) 1000
    drive motor all direction forward speed 40
    pause (ms) 1000
    drive motor all direction forward speed 50
    pause (ms) 1000
    drive motor all direction forward speed 60
    pause (ms) 1000
    drive motor all direction forward speed 70
    pause (ms) 1000
    drive motor all direction forward speed 80
    pause (ms) 1000
    drive motor all direction forward speed 90
    pause (ms) 1000
    drive motor all direction forward speed 100
```

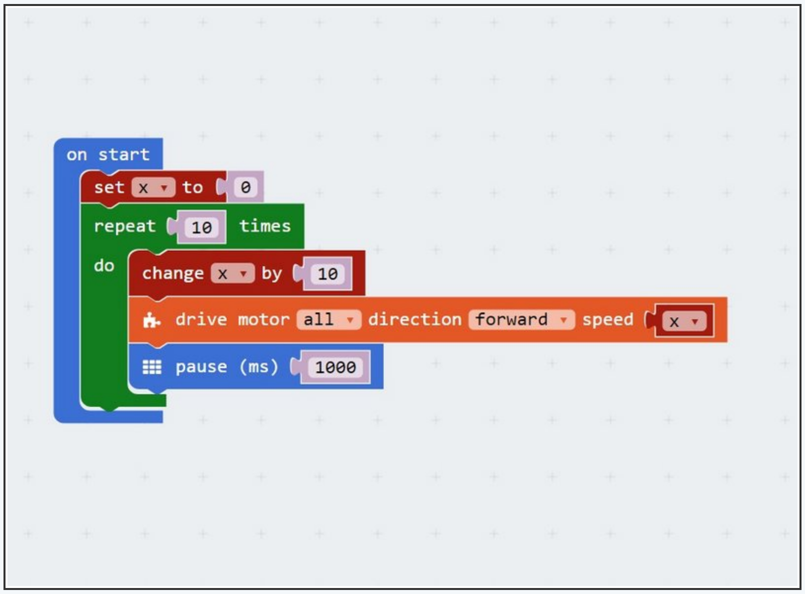
- You might be thinking - **why bother** using a variable to do this? We could have just used the **simple** program in the picture!
- Well, what if we wanted to increase the speed of our robot **(accelerate)** slowly?
- Even if we started the speed at **0** and increased the speed by just **10 every second** (0,10,20,30,40......), the program would require **10 motor blocks!**
- Have a look at the second picture to see this program - it is **far too long.**

This document was generated on 2022-01-04 12:26:32 AM (MST).

Step 9

Variables and Loops

- By combining **variables** and **loops**, we can program things like acceleration **very easily.**
- **Have a look** at the program in the picture - can you **work out** what is going on?
- The loop runs **10 times** - each time it runs, **x is increased by 10**, and the speed of the motors is **set to x!**
- See how much **shorter** this program is?

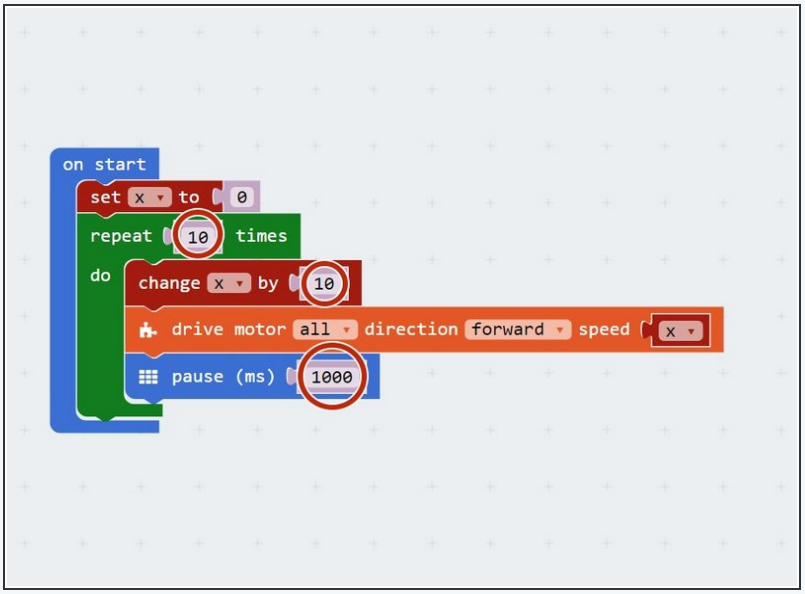


```
on start
  set x to 0
  repeat 10 times
  do
    change x by 10
    drive motor all direction forward speed x
    pause (ms) 1000
```

Step 10

Smooth Acceleration

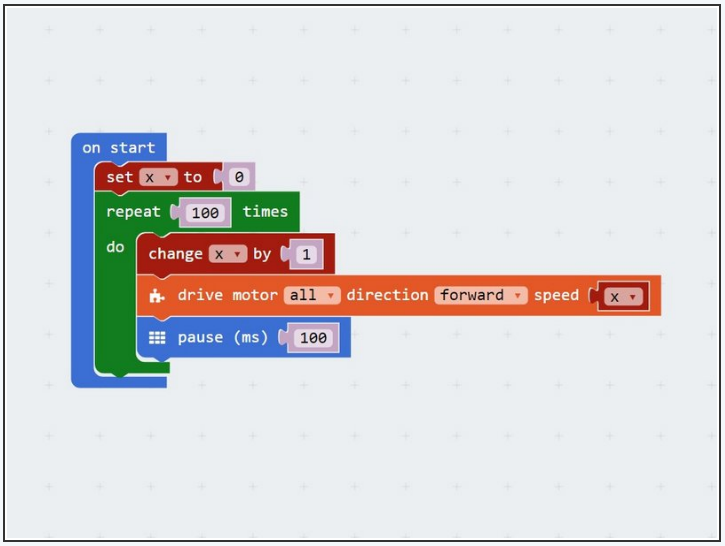
- We're nearly there! Let's **change some things** in our program to make the robot accelerate **really smoothly,** so we don't set off the nuclear waste.
- Change the program so that:
  - The loop repeats every **0.1 seconds** (100 milliseconds) instead of every second
  - Each time the loop repeats, x is **increased by 1**
  - The loop runs enough times for the motors to change speed from **0** all the way to **100.**
- If you need some **hints**, we've marked the parts of the program you will need to **change!**
- **Test your program** - your robot should now speed up really smoothly.



```
on start
  set x to 0
  repeat 10 times
  do
    change x by 10
    drive motor all direction forward speed x
    pause (ms) 1000
```

Step 11

Challenge - Smooth Deceleration



```
on start
    set x to 0
    repeat 100 times
    do
        change x by 1
        drive motor all direction forward speed x
        pause (ms) 100
```

- We need to be able to **decelerate** (slow down) smoothly as well to stop on the other side of the planet.
- **Change your code** so that the motors start at 100%, and **decelerate smoothly to 0.**
- If you need it, the second picture has the correct program for smooth **acceleration.**

Step 12

Transport the Waste

- Now you have learnt everything you need to **safely** move the waste!
- Write a program that:
  - **Starts** at your base
  - **Accelerates smoothly** towards the other side of the planet
  - **Decelerates smoothly** and **stops** at the other side
  - **Waits** for **5 seconds** so the waste can be unloaded
  - **Spins** on the spot **180** degrees
  - Drives back to base at **full speed**, and stops in the right place.

This document was generated on 2022-01-04 12:26:32 AM (MST).