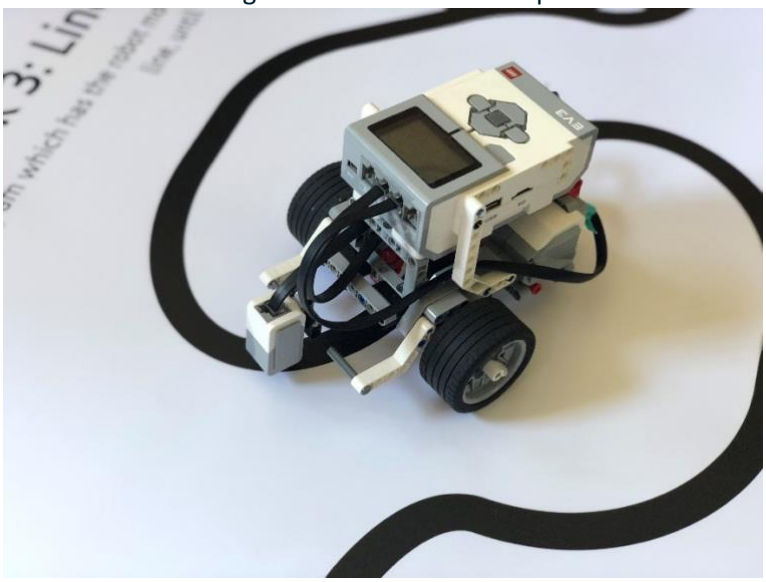


Lego Mindstorms Colour Sensor Workshop

Teacher Pack

Produced by Thomas Dack (2018) Updated by Daniel Westwood (2024)

thomas.dack@stfc.ac.uk daniel.westwood@stfc.ac.uk

This workshop is designed to introduce the concepts of robotics and computer programming through the use of Lego Mindstorms robots. The activity consists of an introductory presentation and three programming tasks.

Each task worksheet has student instructions as well as the pseudocode for the logic needed to write the model solutions. Model solutions for each task can be found within this pack.

Provisional timings for a typical 50 minute session are:

Introductory Talk & Presentation: 15 minutes

Task 1: Stop at Line: 5 minutes

Task 2: Line Counting: 15 minutes

Task 3: Line Following: 15 minutes

Debrief & Wrap Up: 5 minutes

Last updated: April 2024

The curiculum links for this activity are as follows:

KS2:

- design, write and debug programs that accomplish specific goals, including controlling or simulating physical systems; solve problems by decomposing them into smaller parts
- use sequence, selection, and repetition in programs; work with variables and various forms of input and output
- use logical reasoning to explain how some simple algorithms work and to detect and correct errors in algorithms and programs

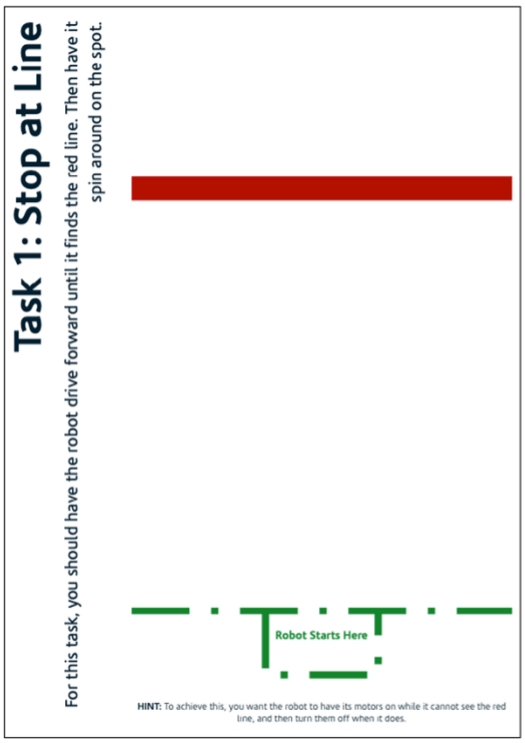
KS3:

- design, use and evaluate computational abstractions that model the state and behaviour of real-world problems and physical systems
- use two or more programming languages, at least one of which is textual, to solve a variety of computational problems; make appropriate use of data structures [for example, lists, tables or arrays]; design and develop modular programs that use procedures or functions
- understand the hardware and software components that make up computer systems, and how they communicate with one another and with other systems

Task 1: Stop at Line

This task introduces students to the Mindstorms colour sensor by having them drive forward until the robot sees a red line.

The worksheet for this task is:



Above is the task map for this activity.

Task 1: Stop at Line

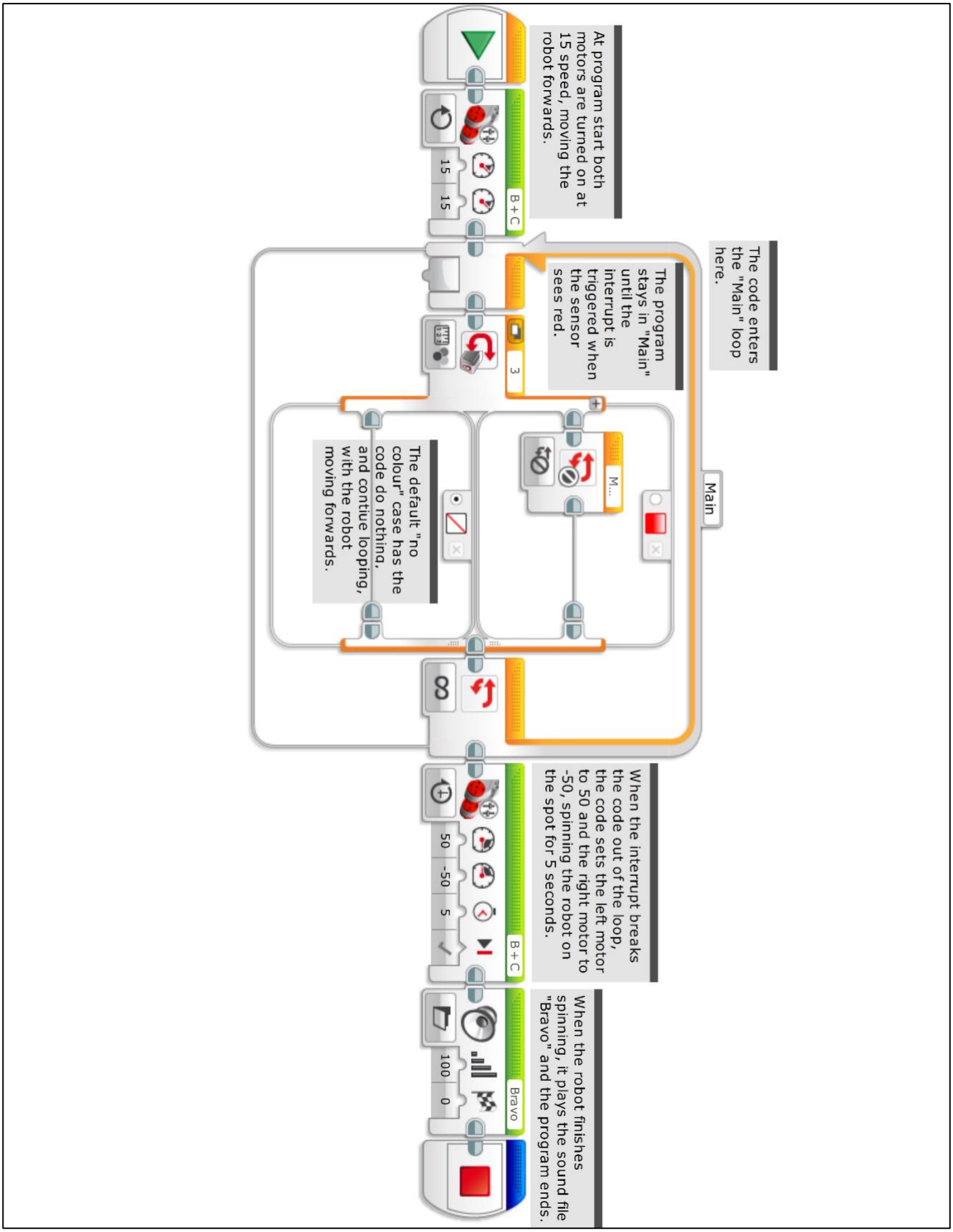
This task involves programming the Mindstorms robot to drive forwards from the start zone (green) until its colour sensor detects the red line. When the red line is found the robot should spin on the spot and then the program should end.
Here is the process your robot should do:

Program starts
Start driving forward
Star looping forever
    If robot sees a red line stop looping
    If robot doesn't see a red line keep looping.
Once stopped looping, spin on the spot for 5 seconds.
Program ends

Science and Technology Facilities Council

Hint text: To achieve this, you want the robot to have its motors on while it cannot see the red line, and then turn them off when it does.

A model solution for Task 1 would be:

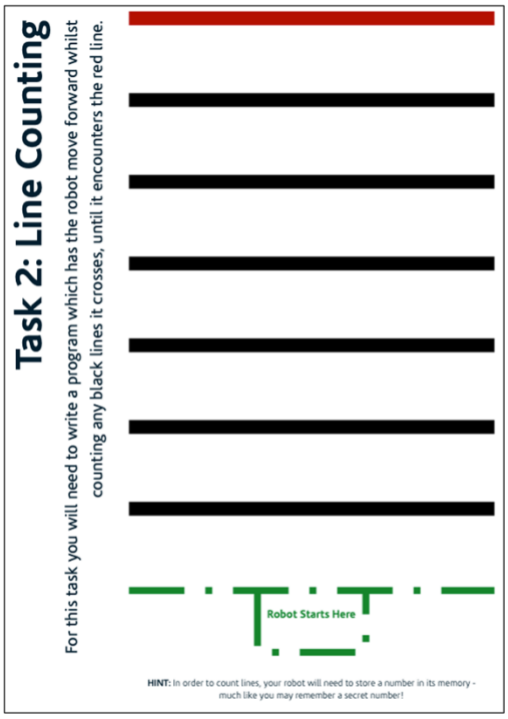


Task 2: Line Counting

This task introduces the concept of variables, as the students must store the number of lines the robot passes over.

With this exercise one potential error is that the robot counts each line more than once. This is due to the program progressing through the loop body multiple times before the sensor moves away from the black line. One "brute-force" solution is to have the program sleep after it detects black, so that it has time to move away from the line. A better solution is depicted in the model solution for this task, where the code waits until it detects white again.

The worksheet for this task is:



Above is the task map for this activity.

Task 2: Line counting

This task involves programming the Mindstorms robot to drive forwards from the start zone (green) and count how many black lines it crosses until it sees the red line. When it sees the red line it should stop moving and display how many lines it counted. With this activity you need to be careful to make sure the robot only counts each line only once as it passes over them!

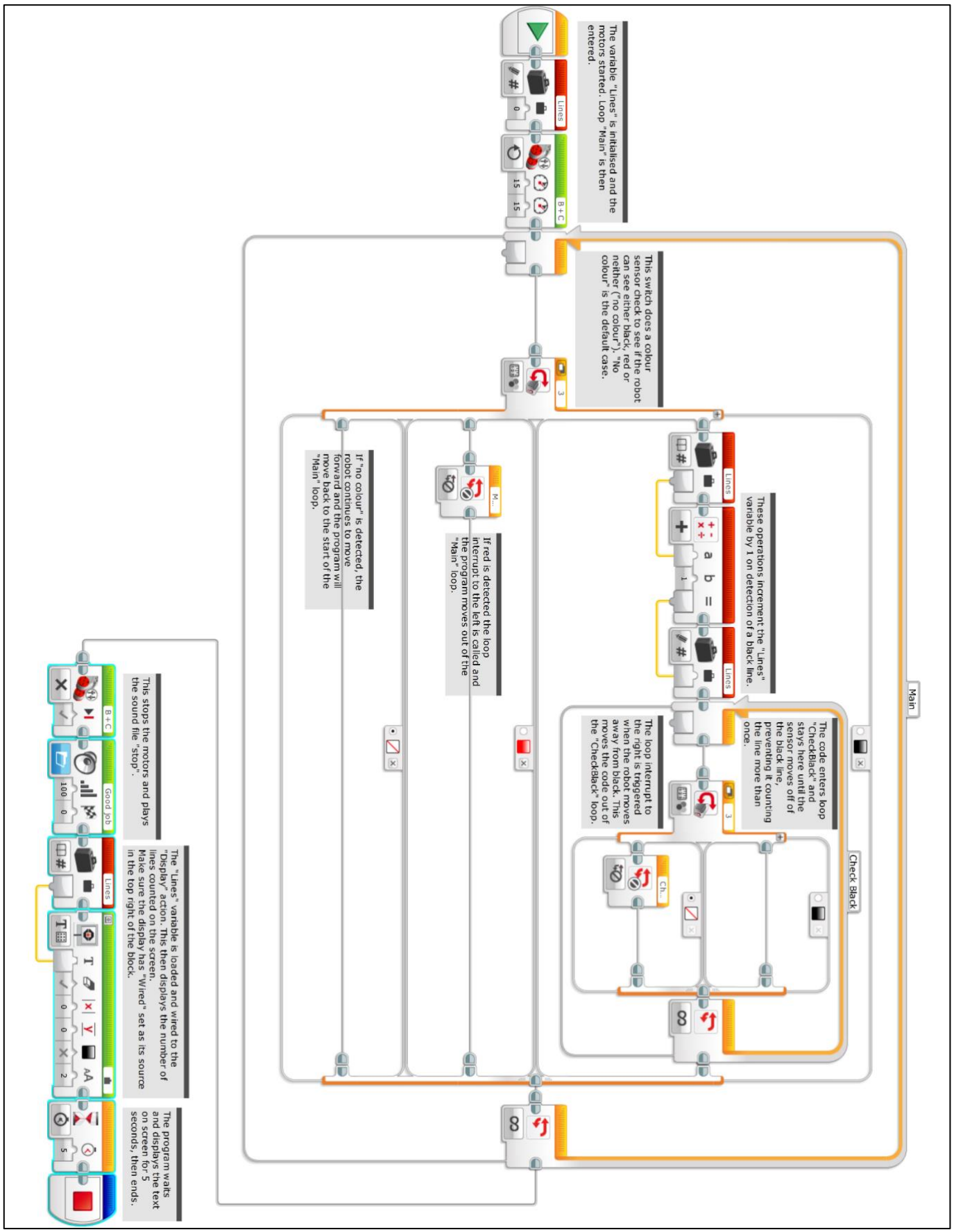
Here is the process your robot should do:

Program starts
Create a variable to remember as 0
Start driving forever
Star looping forever
    If robot sees a black line increase the variable by 1
    If robot sees a red line stop looping
    If robot doesn't see a red line keep looping.
Once stopped looping, display the variable.
Program ends

Science and Technology Facilities Council

Hint text: In order to count lines, your robot will need to store a number in its memory - much like you may remember a secret number!

A model solution for this would be:



Task 3: Line Following

This task has students program a robot to follow a line. The coding solution for this task will likely be easier than Task 2, however the logic required for the solution will require more thought. A line following robot will follow the edge of a line, and so the model solution has the robot turn left if it sees black (towards the left edge) and turn right if it does not (returning it towards the left edge).

The worksheet for this task is:

Above is the task map for this activity.

Task 3: Follow the Line

This task involves programming the Mindstorms robot to drive forwards from the start zone (green) and follow a black line until it reaches a red line. When the robot finds the red line it should stop moving.
Here is the process your robot should do:

Program starts
Star looping forever
    If robot sees the black line it should move to stay on it
    If robot sees a red line stop looping
    Otherwise move back towards the black line.
Once stopped looping, spin on the spot for 5 seconds.
Program ends

Science and Technology Facilities Council

Hint text: It may be easier to think of the robot following the left edge of the line: if it sees black, how will it find the edge? What if it sees white?

A model solution for this would be: