E.a.R.L.2

Coding Robot

A guide for Teachers

Contents

Introduction

E.a.R.L 2 and the National Curriculum for Computing

E.a.R.L 2 can be used to deliver many aspects of the Computing Curriculum. E.a.R.L 2 is particularly well suited to the statements below, however can also be used to support learners in other curriculum areas and throughout the EYFS.

Early Learning Goals

ELG: Numerical Patterns

Children at the expected level of development will:

- *Verbally count beyond 20, recognising the pattern of the counting system;*

ELG: People, Culture and Communities

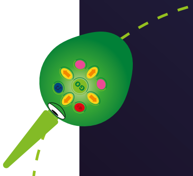
Children at the expected level of development will:

- *Describe their immediate environment using knowledge from observation, discussion, stories, non-fiction texts and maps*

ELG: Creating with Materials

Children at the expected level of development will:

- *Safely use and explore a variety of materials, tools and techniques, experimenting with colour, design, texture, form and function*



Key stage 1

Pupils should be taught to:

- *understand what algorithms are, how they are implemented as programs on digital devices, and that programs execute by following precise and unambiguous instructions*
- *create and debug simple programs*
- *use logical reasoning to predict the behaviour of simple programs*

Key stage 2

Pupils should be taught to:

- *design, write and debug programs that accomplish specific goals, including controlling or simulating physical systems; solve problems by decomposing them into smaller parts*

Building on children's previous experiences

Most children will have had some experience of cause and effect toys, such as remote control cars, prior to meeting E.a.R.L 2. Children can be encouraged to think about these action/reaction toys, i.e. "If I do this… then the toy does that." They can also talk about technology around them which can be described in the same way, e.g. TV remote. Making these real world comparisons helps children embed an understanding of how E.a.R.L 2 works.

Using the activities

The following activities are progressive, with skills being developed and built upon through the activities. Most of the activities will suit Key Stage 1 pupils, but can be easily adapted to suit EYFS and Key Stage 2 learners. The following activities are examples of how to make good use of E.a.R.L 2 but are by no means an exhaustive list. The amount of time spent on each activity can be flexible to suit your class size and children's needs.

Coding

To maximise your use of E.a.R.L 2's capability, please download the E.a.R.L 2 scheme of work that can be found via the E.a.R.L 2 home page at **hope-education.co.uk/earl**.

There you will find lessons using Scratch 2.0 which make use of E.a.R.L 2's advanced features using customisable movements.

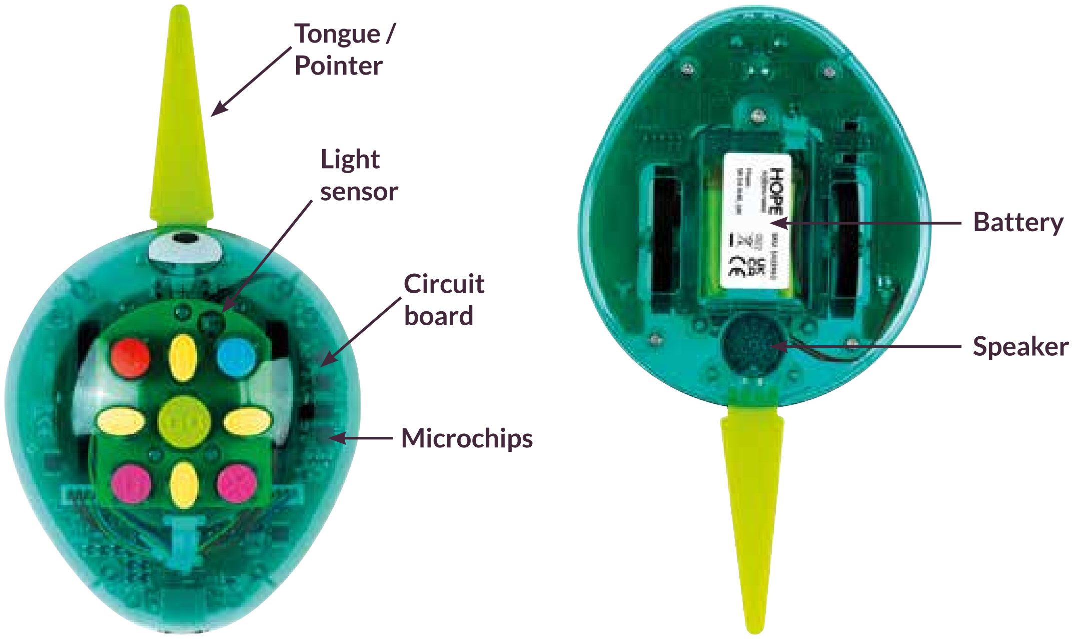
Activity 1 – Who is E.a.R.L 2?

Looking at E.a.R.L 2 inside and outside

E.a.R.L 2 has a transparent body allowing its internal components to be seen. Children should be encouraged to look closely at E.a.R.L 2 and talk about what they can see on the outside and then the inside, e.g. How many buttons can they see? How many switches can they see? How many wheels? etc.

Children should then discuss what they think the different components might do.

Note: When looking at the inside of E.a.R.L 2, the battery, speaker, motors and circuit board can be seen. Microchips can also be seen on the circuit board.



There is one external connector on the back of E.a.R.L 2. This is used to charge E.a.R.L 2. The link between charging the robot (E.a.R.L 2) and the battery holding that charge could be made. Children should develop an understanding that electrical power is needed by computing devices e.g. mobile phones, tablet computers etc. (Also see the 'label E.a.R.L 2' activity sheet at the end of this document).

Investigating E.a.R.L 2's buttons

After discussing the buttons on top of E.a.R.L 2 the children can test them to see what they do. This could be done as a class discussion or as a small group activity. Point out that before anything will work, E.a.R.L 2 must be switched on. The children should be able to discover what most buttons do.

Note: Some buttons might be a little more difficult to work out. Children should learn that the buttons allow E.a.R.L 2 to be programmed and that program is executed (runs) when the 'Go' button is pressed. The 'X' button clears E.a.R.L 2's memory. It can also be used to stop E.a.R.L 2 mid program. All the other buttons are command buttons and each press adds that command to the end of the program stored in E.a.R.L 2's memory.

Investigating E.a.R.L 2's lights and sound

When E.a.R.L 2 is being used it will light up and make sounds (if the sound is switched on). Initially children should spend time experimenting with E.a.R.L 2's lights. They can try and work out how many different light effects there are. A list of different light sequences could be drawn up as a class e.g. when E.a.R.L 2 does this... the lights do this.... The activity can then be repeated focussing on E.a.R.L 2's sounds. (Also see 'E.a.R.L 2's lights' activity sheet at the end of this document).

Activity 2 – Moving with a purpose (Part 1)

Introducing algorithms

A key term in the Computing Curriculum is 'algorithm'. An algorithm is a sequence of instructions to do something. An algorithm can be seen as the steps to solve a problem. By pressing the buttons on E.a.R.L 2, children create an algorithm. During the first few activities the term algorithm can be introduced and reinforced.

Using E.a.R.L 2's forward and back buttons

Having explored what E.a.R.L 2 can do, children can begin programming it. Initially the children should use only the forward and back buttons.

E.a.R.L 2 should have a set start point and a specified destination on a mat. Starting from a given square, the children should program E.a.R.L 2 using only the forward button to get to a specified destination. Before they press 'Go' each time, E.a.R.L 2 must be placed back in its designated starting square. The children can move on to make predictions e.g. if the forward button is pressed three times which square will E.a.R.L 2 stop on?

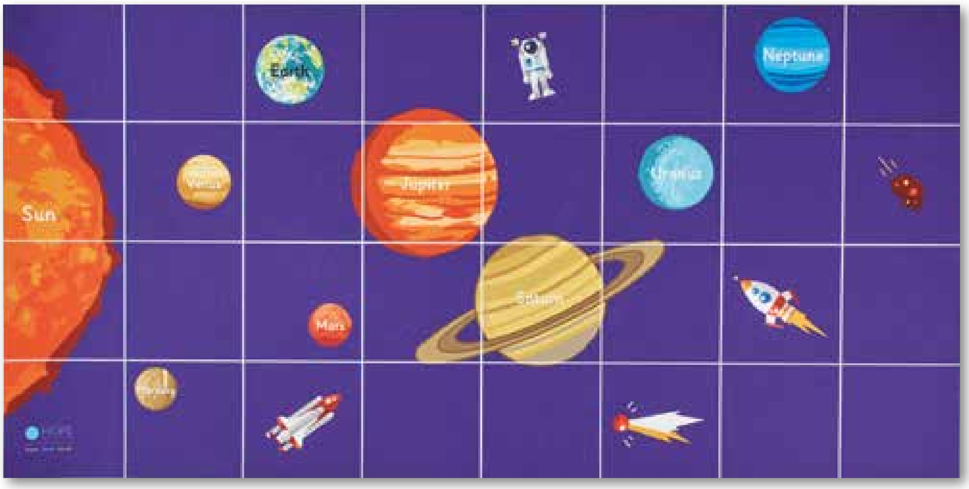
Next, children can be given a start square and two specified destinations. This time children should use the forward button to reach one destination and the back button to reach the other. Again the children should be encouraged to make predictions. If the forward button is pressed 4 times and then the back button is pressed twice, where will E.a.R.L 2 stop?

Activity 3 – Moving with a purpose (Part 2)

Using all E.a.R.L 2's direction buttons

A variety of mats can be used for this activity. Ideally children will work in groups of two or three. Children select a starting square for E.a.R.L 2 and decide which way E.a.R.L 2 should face. This square and direction could be marked with a piece of paper or tape. The children should then challenge each other to program E.a.R.L 2 to get to a specified square. These challenges can be solved by trial and error. Sometimes a command or two may need adding, at other times the entire program may need deleting before trying again.

When the children become confident in getting to the target square, an additional target can be added, e.g. can you get to this square via this square? Children may need reminding to always start from the same starting square, pointing in the same direction.



Activity 4 – Planning algorithms

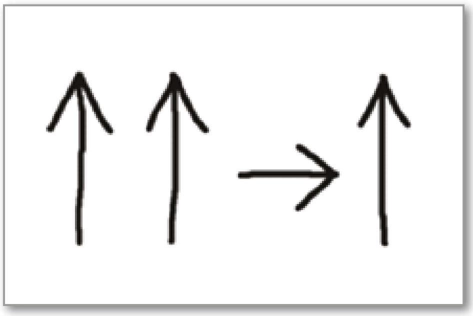
Recording algorithms

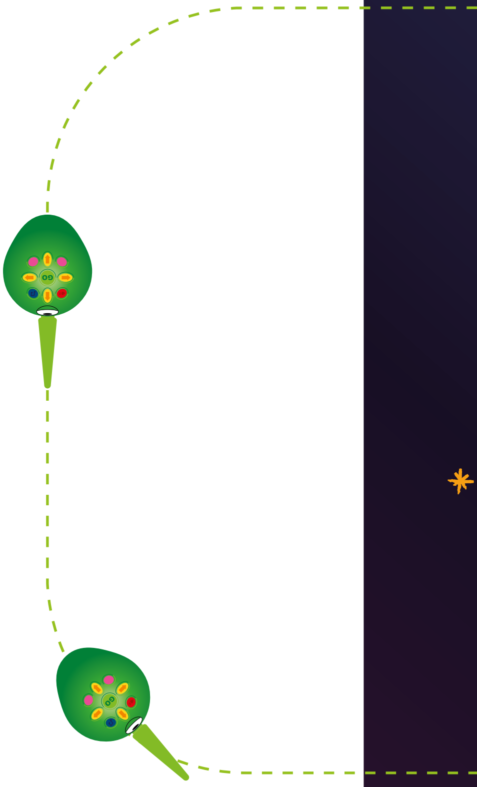
As the children become more confident using E.a.R.L 2 they need to start writing their algorithms down before pressing any buttons on E.a.R.L 2. This gives them the opportunity to plan their steps and later to start debugging.

This activity will work best when the children work in groups of two or three. It can help to give children specific jobs, e.g. challenge setter, program writer and tester (button presser), with each child taking turns in each role.

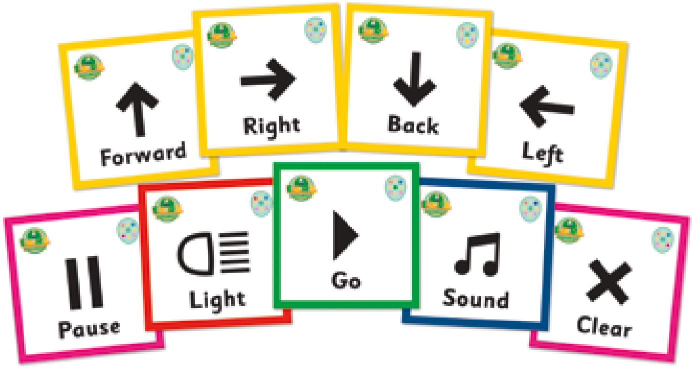
A variety of mats can be used. E.a.R.L 2's starting point, direction and target should all be marked on the mat using tape or a piece of paper. The children will need to plan out their solution. This plan can be recorded on a dry wipe board, on paper or using the E.a.R.L 2 programming cards. The programming cards are particularly useful when debugging algorithms later.

An E.a.R.L 2 program on a dry wipe board:

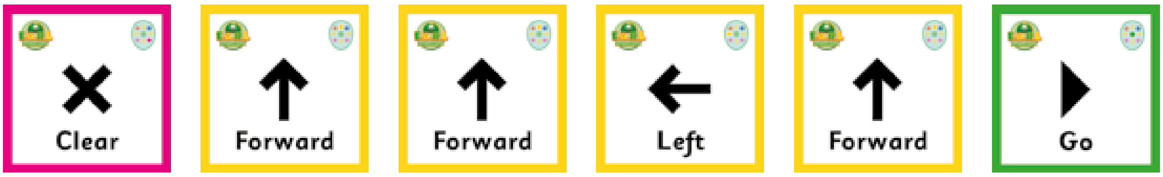




E.a.R.L 2 Sequencing Cards:



An E.a.R.L 2 program using the cards:



The sequence could be recorded horizontally or vertically. Also note the idea of using "Clear" at the start of the sequence and "Go" at the end. Using "Clear" at the beginning introduces the programming concept of initialisation i.e. things that have to happen before the main program is executed.

Activity 5 – Fixing bugs

Debugging a program

As the challenges with E.a.R.L 2 become more complex, children will find their algorithm doesn't always solve the challenge first time. When E.a.R.L 2 doesn't arrive at the given target they will need to fix their program and fix any errors. In computing terms, an error in a program is called a 'bug', hence the term 'debugging'.

To develop debugging skills, activity 4 can be repeated. If E.a.R.L 2 doesn't reach the set target the children need to review the program and fix it (debugging). Once they think they have found the bug, they fix their recorded plan, and then try the modified sequence on E.a.R.L 2. They should continue debugging their programs until they are successful.

Note: Challenges can be made more complex by facing E.a.R.L 2 away from the destination, adding multiple destinations or blocking certain routes. The children should be encouraged to make E.a.R.L 2 light up or play a sound as it reaches each point on its journey.

An algorithm is a sequence of steps to solve a problem. There may be more than one solution which successfully gets E.a.R.L 2 to its destination and any working solution should be considered valid. This is also the case in most programming, i.e. there will be a number of ways to achieve the same result. Some solutions may be considered more efficient than others. This concept can be introduced to children by asking them to write an alternative algorithm for a challenge they have completed.

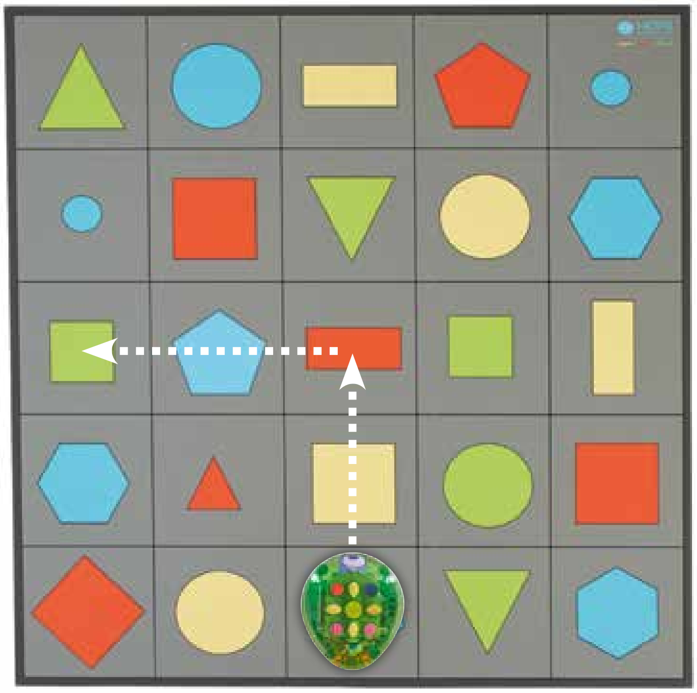
Activity 6 – Breaking down challenges

Introducing decomposition

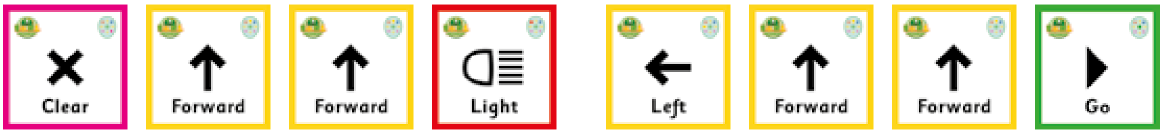
Decomposition is the term used for the process of breaking a problem down into smaller parts.

This activity is similar to activity 5 in that a challenge is set with a specified start, direction and target. This time the children are asked to break the challenge down at the programming stage.

The challenge shown below would be to get E.a.R.L 2 from its starting point to the green square. One way to decompose this problem is shown by the dotted white arrows (stage 1 from start to red rectangle and stage 2 from red rectangle to green square). This isn't the only solution and children should be encouraged to look for other ways to achieve the same result.



A solution could look like this:



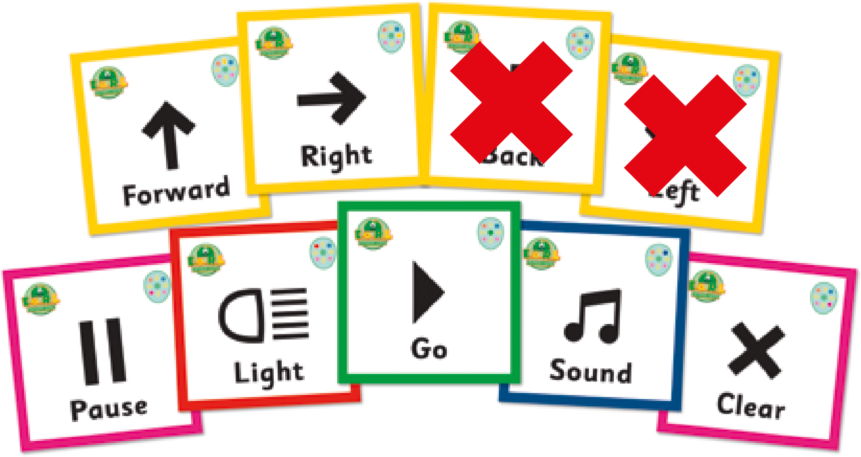
Note. The solution above makes E.a.R.L 2's lights flash between the two parts of the program. This can be helpful in the debugging process, e.g. did the error occur before or after the lights flashed?

Any of the light, sound or pause buttons could be used in this way.

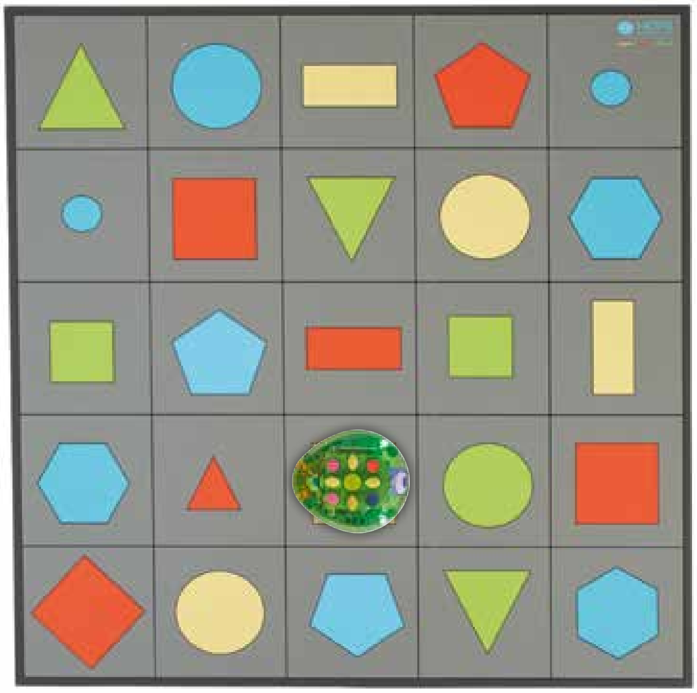
Activity 7 – Introducing more complex challenges

Problem solving

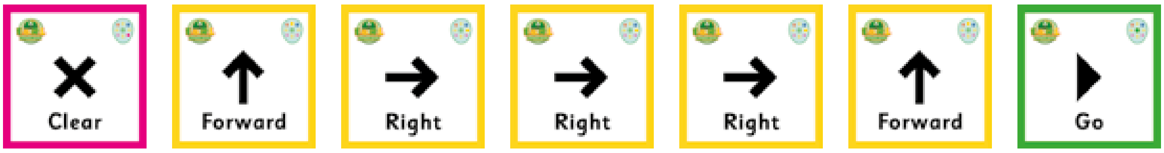
This activity encourages children to further apply logical thinking to solve problems. If E.a.R.L 2 didn't have a left turn or a back button, can it still get anywhere we want it to on the mat?



Some of E.a.R.L 2's buttons could be covered with a small piece of paper and/or some of the programming cards could be removed. Children then set themselves challenges and attempt to solve them using a restricted set of E.a.R.L 2 commands.



Example: Get E.a.R.L 2 to the green circle using the cards shown above:

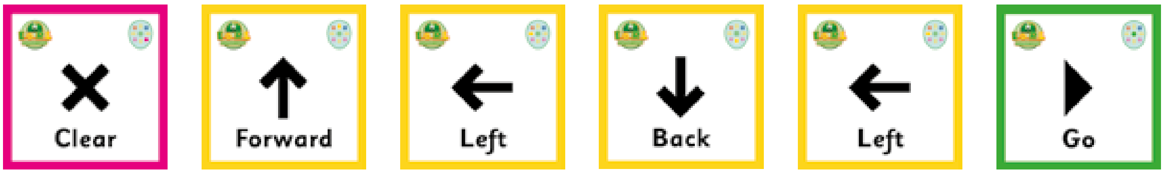


Activity 8 – Introducing prediction

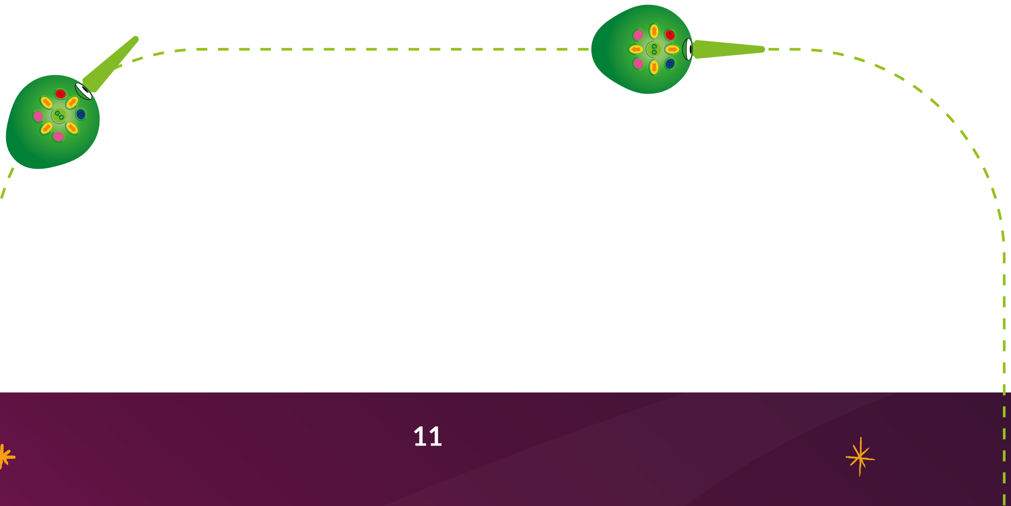
Reading a program and predicting the outcome

This activity will work best when using the programming cards. Some direction cards are mixed up and placed face down. E.a.R.L 2 should be placed towards the middle of a mat. Several cards are picked from the pile, turned face upward and put into a sequence between ‘Clear’ and ‘Go’ cards.

Once a random program has been created the children need to use logical reasoning to work out where E.a.R.L 2 will get to and which way E.a.R.L 2 will face. Once they think they have the answer they can test the program with E.a.R.L 2 and see if they were correct.



As the children develop confidence with this activity they can repeat it using randomly selected command cards to increase the length of the program.



Glossary

Algorithm

A sequence of steps to solve a problem. With E.a.R.L 2 this is the commands that are entered using the buttons on top.

Bug

An error in a program.

Debugging

Fixing errors in a program.

Decomposition

Breaking a problem down into smaller parts.

Initialisation

The initial commands or steps before the main program is executed.

Program

A sequence of instructions that a computer follows.

____

____

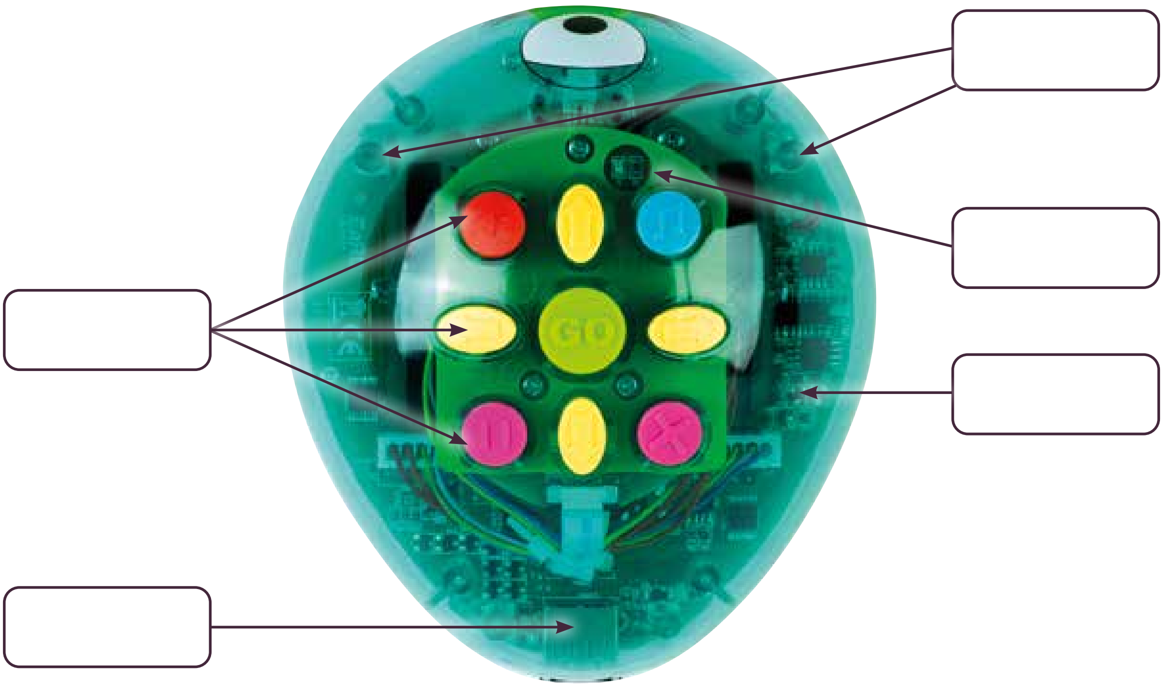
____

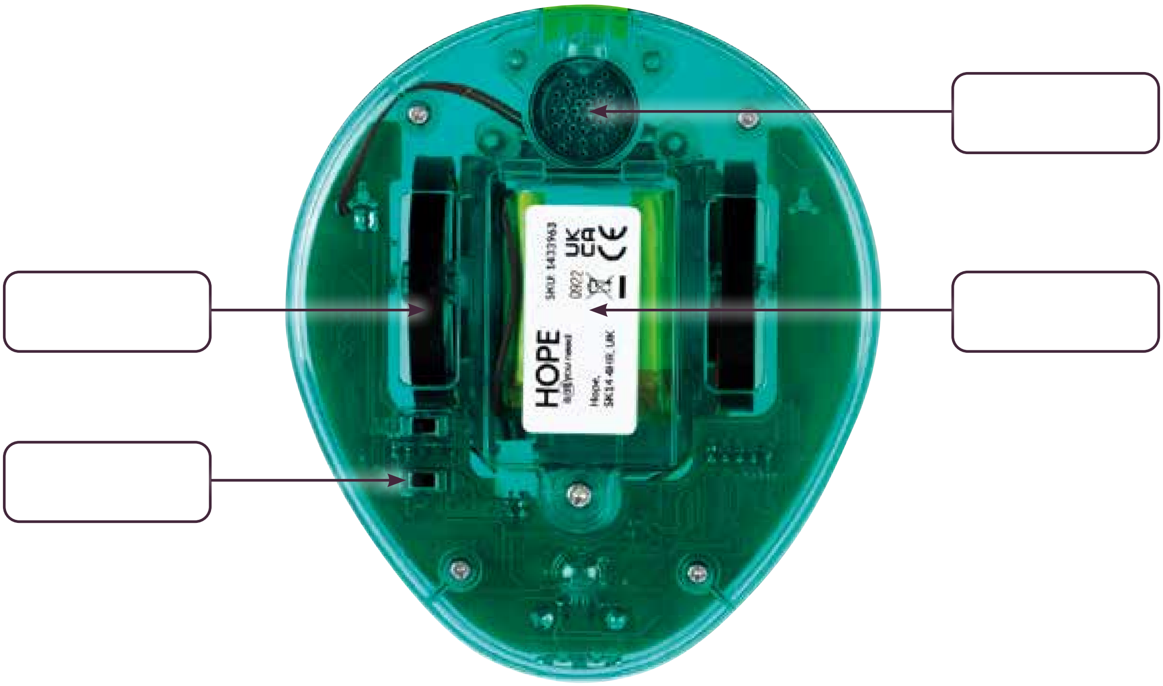
____

____

____

Label E.a.R.L 2 Activity Sheet

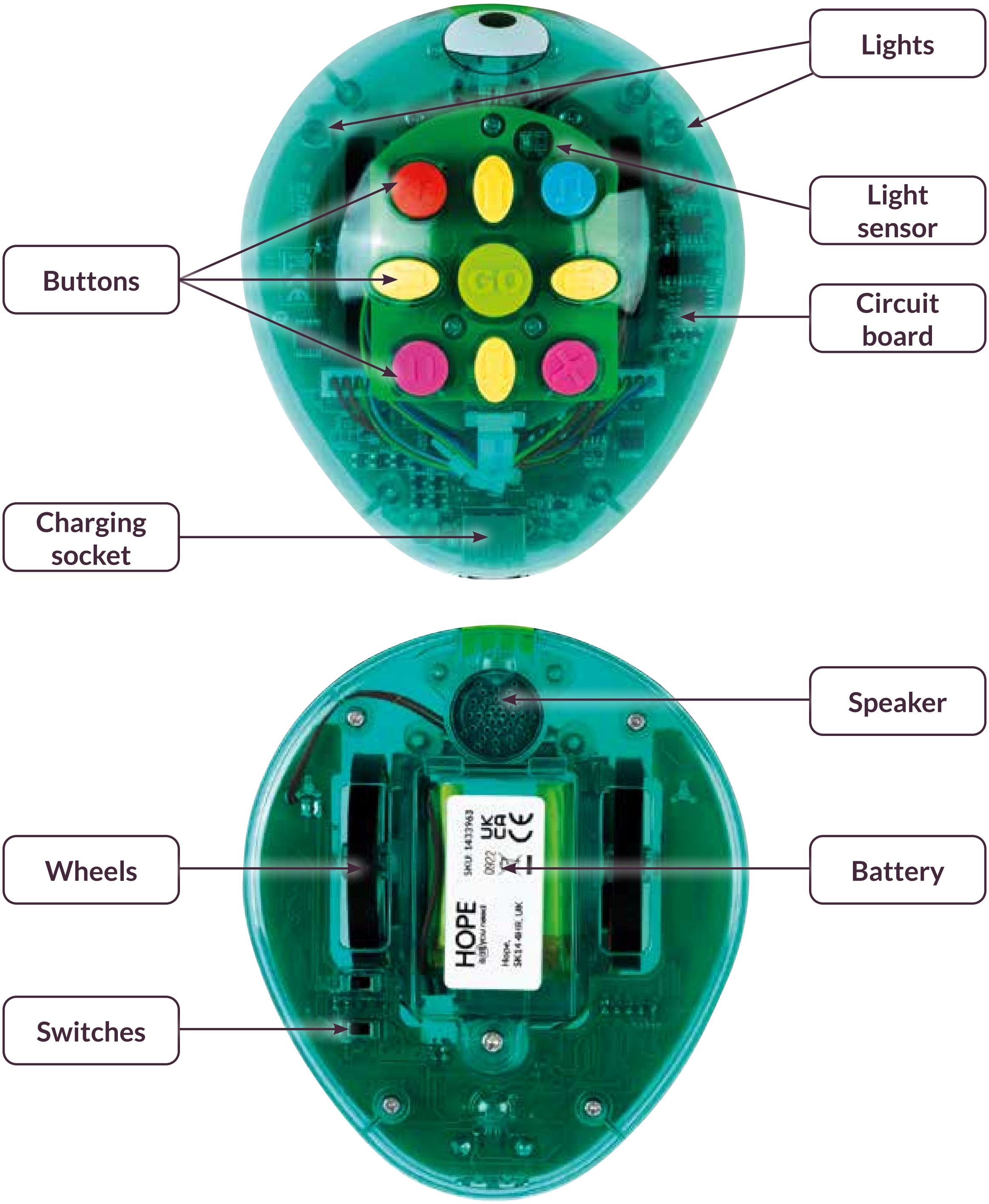




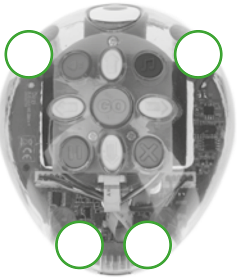
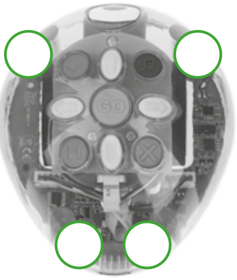
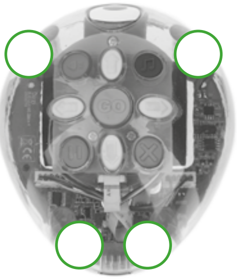
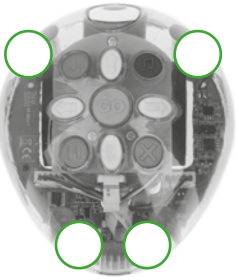
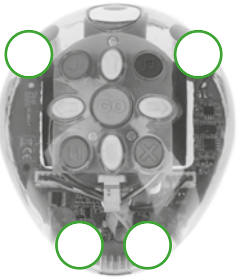
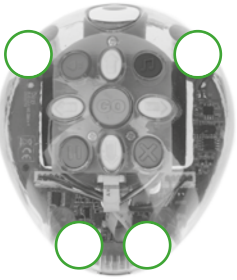
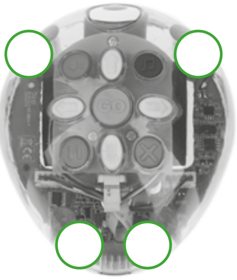
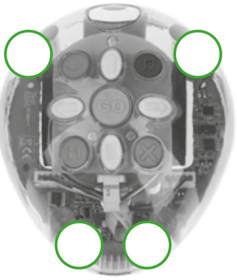
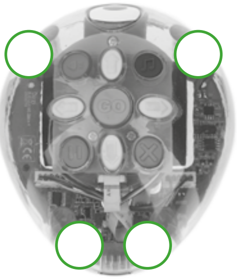
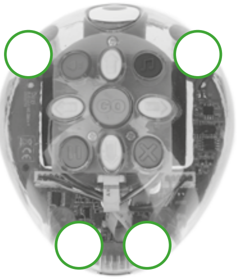
Cut out the words below and correctly label E.a.R.L 2.

Buttons	Switches	Lights	Circuit board
Wheels	Charging socket	Speaker	Battery

E.a.R.L 2 Inside and Out – Diagram



E.a.R.L 2’s Lights Activity Sheet

⇧		💡	
⇩		♫	
⇨		⏸	
⇦		✕	
GO		At the end of a program	

Colour in E.a.R.L 2’s lights for each action.