Year 6 Summer 2- Short term plan - Microbits and robot cars

Unit Rationale

This unit introduces pupils to the idea of physical computing. This enables pupils to use their computers to interact with sensors, LEDs, robotics, buttons etc on a microcontroller such as the BBC Microbit, rather than just on the computer itself (screen, keyboard, touchpad etc). Pupils are shown a variety of real-world applications and encouraged to use knowledge gained to start building their own projects.

National Curriculum Objectives:	Cross Curricular Links:
• 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	
Trips/Visits:	**Modern Day Links:**
Prior Learning:	**Substantive Knowledge:**
Y1/Y2 - Algorithms & programs Y3 - Creating an animation (Scratch) Y4 - Programming and debugging (Scratch)	facts. what is an algorithm?

Big ideas/Disciplinary Knowledge	What next?
thinking like a computer scientist	Y5 - Designing an adventure game (Scratch) Y6 - Advanced programming including simulations, debugging and action games Y6 - Controlling cars and lights with a Microbit (To be written)

Lesson	WALT	What should the children remember?	Lesson plan and outcome	Key Vocabulary	Key Questions
1	**E-Safety**		See separate e-safety plans		
Lesson 2	**Find out how far a robot car travels and turns**		**Note: This lesson can be done as a separate STEM experiment. You will need to print out experiment sheets and test grids for each group. You will also need metre sticks for measuring distance and paper/pencils.** **Put the program download links on Google Classroom so children can access them.** **Show WALT** **Introduce Move Motor Robot Car** - Show Video - Explain features - Explain extension for makecode editor **Introduce motor commands** - move - stop - spin - pause **Experiments** To learn how to control the robot cars more accurately, they will be doing two experiments and recording the results.	**Move motor** **Sensors** **Extension** **Experiment**	**What sensors does a Move Motor have?** **What do you need to add to the makecode editor to use the Move Motor?** **How can you change the amount of spin/movement that the car performs?**

			This information/experience can then be used in future lessons to predict what values the time/speed variables will need to be rather than just randomly guessing. Explain experiments (see slides and sheets) **Activity** Children to use printed experiment/results sheets in groups to conduct the experiments and record the results. Speed test link https://makecode.microbit.org/S04590-40899-99777-78158 Spin test link https://makecode.microbit.org/S95053-67180-79460-55350 **Quiz** See slide		
Lesson 3	**Accurately drive a robot car around objects**		**NOTE: Pupils will need the experiment results from last lesson to help them work out the values for time/speed. Keep the speed the same throughout the program and change the time only for best results.** **Quiz** See slide **Show WALT** **Work through challenge** - TTYP: Can they create an algorithm? - TTYP: What commands might they use for a program? - TTYP: How will you control movement? **Example**	**Algorithm**	**How accurate was your program?** **What did you need to change/adjust to make it work better?**

			- Use results from last lesson to help control the forward movement (see slide) - Explain that they can use this info to control the spin too. - NOTE: Works best if they keep the speed the same and just change the time. **Challenge** - Set up an object for the car to drive around with a start/finish line. - Use a ruler to measure distances and the information from last lesson to write a program to drive around the object. - Test your program **Plenary: How accurate is your program? What have you had to change?**		
Lesson 4	**Accurately draw shapes with a robot car using loops**		Note: Check the equipment list! **Quiz** **Show WALT** **Teaching (work through slides)** - Remind the children about how programmers use repetition to make their programs more efficient (shorter and simpler) - Introduce repetition command on Makecode - Introduce the pen hole - Explain how we can use our test results from week 1 along with Maths to try and draw more accurate shapes. - Explain the example program **Challenge** - Children to try and draw a square on A3 paper using the move motor car and a sharpie pen (you may need blue tack to hold the pen in place).	**Repetition Loops**	**What lessons have you learned about controlling the cars accurately?** **Whose program was most accurate?**

			Reflection (See slide) - Assess how accurate the shapes were. - Have they tried slowing down the cars for greater accuracy? (e.g. set speed to 5 or 10) **Extension challenges** - Can they draw a rectangle? (repeat x2 long side and short side) - Can they draw a triangle? Repeat x 3 with 120° turn **Plenary** - Reflect on what they have learned about moving the car with greater accuracy - Whose shapes were the most accurate?		
Lesson 5	**Program a robot car to follow a line.**		Note: Check the equipment list! **Quiz** **Show WALT** **Teaching (work through slides)** - Line following sensors - Explain algorithm **Challenge** - Download the line following program https://makecode.microbit.org/S04344-47488-86640-21447 - Children to draw an oval on A3 paper using a thick marker pen, place the car on the line and switch it on. Does it follow the line? **Reflection (See slide)**		Did the car follow the line correctly? Was a faster/slower speed better? Did you get the lights working correctly?

			- Try adjusting the speed of the car. Does the car follow more accurately when the speed is faster or slower? **Teaching** - Explain about the four LED lights - Show the code that is needed to light the lights. **Extension challenges** - Can they amend the program so that the car lights are red when turning right, blue when straight and green when left? **Plenary** - Did the car follow the line correctly? - Was a faster/slower speed better? - Did you get the lights working correctly? - How else could you use the lights with the car? (e.g. think about real cars)		
Lesson 6	**Program a robot car to follow a road and use light/sounds.**		**Introduce sounds?** **Functions?**		