Key Vocabulary and Definitions:

|  |  |
| --- | --- |
| accelerometer | A sensor that detects movement. |
| algorithmAlgorithm - Free seo and web icons | A set of step-by-step instructions. |
| Bluetooth | A way that devices can be connected to each other. Typically, a Bluetooth connection will be wireless. |
| flashingTransfer Clipart Images | Free Download | PNG Transparent Background -  Pngtree | The process of transferring a program to a micro:bit. It is called flashing because the program is copied to the micro:bit’s flash memory. |
| LED  | This stands for light-emitting diode. The micro:bit display is made of 25 LEDs. |
| microphone | A piece of hardware that can be used to input audio. |
| processor | Sometimes called the ‘brains’ of a computer. The processor receives the inputs, runs the programs and gives outputs. |
| program | A set of instructions written in code that performs a given task. |
| touch sensor | A device capable of detecting when it is touched. |
| USB data cable  | Allows the transfer of data between a computer and peripheral devices. USB is short for universal serial bus. |

Progression

In this unit, pupils will use physical computing to explore the concept of selection in programming through the use of the micro:bit.

Pupils will be introduced to a microcontroller and will learn how to program components through the application of their existing programming knowledge.

National Curriculum:

**Computing**

* 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

**Internet safety**

Use technology safely, respectfully, and responsibly; recognise acceptable/unacceptable behaviour.



Teaching Sequence

Programming A – Selection in Physical Computing

1. To create a program and transfer this to a micro:bit
2. To create a simple animation to learn about sequence and simple loops.
3. To code the micro:bit to make different outputs happen depending on different inputs.
4. To understand how inputs, outputs and computer code work together to make control systems.
5. To use the accelerometer via the ‘on shake’ block to start a code running.

Online Safety – Online Bullying

1. I can recognise online bullying can be different to bullying in the physical world and can describe some of those differences.
2. I can describe how what one person perceives as playful joking and teasing might be experienced by others as bullying.
3. I can explain how anyone can get help if they are being bullied online and identify when to tell a trusted adult.
4. I can identify a range of ways to report concerns and access support both in school and at home about online bullying.
5. I can explain how to block abusive users and describe the helpline services which can help people experiencing bullying.

Blooms Taxonomy – Specific Verbs to Use in Lesson Aims

Knowledge: Describe, find, identify, list, locate, name, recognise, retrieve Comprehension: Classify, compare, explain, infer, interpret, paraphrase, summarise Application: Carry out, implement, use Analysis: Deconstruct, Organise, outline, structure Synthesis: Construct, design, devise, invent, make, plan, produce, Evaluation: Appraise, assess, choose,

## Subject knowledge

This unit focuses on physical computing, which allows learners to control real-life projects through the construction of programs. When learners undertake physical computing, they write programs that control real-world objects, like LEDs and motors, using a computer. The tangible effect of seeing the commands that they entered into a computer being carried out on a physical item, rather than on screen, can be highly motivational for learners. Physical computing also offers the opportunity to take a more project-based approach to learning, and allows learners to make choices about the purpose, design, and program of their product.



### Levels of abstraction

When programming, there are four levels that can help describe a project (known as ‘levels of abstraction’). Research suggests that this structure can support learners in understanding how to create a physical computing project or standalone program and how it works:

* Task — this is what is needed
* Design — this is what it should do
* Build — this is how it is done
* Running the code — this is what it does

Spending time at the ‘Task’ and ‘Design’ levels before engaging in writing code aids learners in assessing the ‘do-ability’ of their programs and reduces a learner’s cognitive load during programming. Learners will move between the different levels throughout the unit, and this is highlighted within each lesson plan.

### Selection

When designing programs, there are often points where a decision must be made. These decisions are known as ‘selection’, and are commonly implemented in programming using ‘if’ statements. Selection is used to control the flow of actions in algorithms and programs by checking whether a condition (see above) has been met. If it has been met, the identified actions will be carried out. When selection is used in programs, infinite loops (see above) are often used to instruct the device to check the condition repeatedly. Without using loops, the condition would only be checked once following the sequence of the code.