|  |  |  |
| --- | --- | --- |
| **Smart Greenhouse with the BBC micro:bit** | | |
| Designing and programming a system that can keep conditions in a greenhouse ideal for growing food | | |
| **Subject(s):** Design and Technology, Computing, Engineering  **Approx time:** 80-120 minutes |  | **Key words / Topics:**   * BBC micro:bit * blocks * design brief * greenhouse * growing food * programming * moisture level * temperature control |
| **Stay safe**  Whether you are a scientist researching a new medicine or an engineer solving climate change, safety always comes first. An adult must always be around and supervising when doing this activity. You are responsible for:    • ensuring that any equipment used for this activity is in good working condition  • behaving sensibly and following any safety instructions so as not to hurt or injure yourself or others    Please note that in the absence of any negligence or other breach of duty by us, this activity is carried out at your own risk. It is important to take extra care at the stages marked with this symbol: ⚠ | | |
| **Suggested Learning Outcomes** |  |  |
| * To understand the conditions needed for food plants to grow well in a greenhouse. * To be able to design a programmable system that can control the temperature and soil moisture levels within a greenhouse. | | |
| **Introduction** |  |  |
| This is one of a series of resources to support the use of the BBC micro:bit in Primary Design & Technology, Computing and Engineering lessons. This resource focusses on learners designing, programming and using a programmable system to measure and control temperature and soil moisture level in a greenhouse. | | |
| **Purpose of this activity**  In this activity learners will make use of the BBC micro:bit to design and create a programmable system that can control the temperature and soil moisture levels in a ‘smart’ greenhouse. They will analyse a design brief and design criteria before taking one of two routes through the activity – designing and producing the programming themselves or using a pre-written program that they can download straight onto their device and/or edit as they go through.  This activity could be used as a main lesson activity to teach about how programmable devices can be used to solve problems associated with food production, within D&T, Computing and/or Engineering. It could also be used as part of wider STEM-based scheme of learning focussed on how programming can be used to embed intelligence into products and systems. | | |

|  |  |  |
| --- | --- | --- |
| **Activity** |  | **Teacher notes** |
| **Introduction (5-10 minutes)**  Teacher to introduce the activity and safety notes. Teacher to explain that learners will be producing a programmable system to control the temperature and soil moisture levels in a ‘smart’ greenhouse.  **Starter activity (10-20 minutes)**  Teacher to show the starter program to learners. Ask them to look at the code and explain what they think it does. Finally, get them to simulate the code on their own computer to see what it actually does.  **Design context, brief and criteria (5-10 minutes)**  Explain to learners that the design context is what sets the scene for the project that they will be working on. Share and discuss the context given on presentation slide 4. Then discuss the design brief shown in slide 5 and the design criteria on slide 6.  Teacher to hand out the resources and equipment required.  **Producing the programmable device (30-40 minutes depending on route taken) ⚠**  Learners can be taken through this activity in one of two different ways. Route 1 (presentation slide 7) or route 2 (slide 8). Servo motors will also be required to physically open the greenhouse window and actuate the watering of the plants.  Ensure and demonstrate safe practices when working with electronic devices e.g. ensure polarity of battery leads are correct, do not touch exposed wires, remove power if components are overheating etc.  Route 1  Learners to follow the instructions on slide 7 to open a new project. They should then write their own code to satisfy the needs of the design brief. They should simulate this and check that it works prior to downloading. The teacher could show them or provide access to the example program plant:bit to help them, or take them through how to write the code step by step.  Route 2  Learners to follow the instructions on slide 8 to open the plant:bit example program on screen. They should then simulate this to check it works and download to their micro:bit. The main focus in this instance will be on testing and using the device to control the conditions within the greenhouse.  **Testing the System (20-30 minutes)**  Learners to attach the required external input and output devices to the micro:bit – moisture probe and servo motor actuators. If it is not possible to use physical output devices they could simulate the inputs and outputs to the system.  Learners to test their system to check that it works as expected and meets the needs of the brief. |  | This activity could be undertaken as individuals, in pairs, or in small groups.  **Micro:bit versions**  This resource works with both V1 and V2 of the BBC micro:bit. With both V1 and V2, servo motors will be required for actuating the outputs. Moisture probes will be needed to measure soil moisture levels (see resource notes below).  **Starter**  Teacher could deliver this starter using the PR of the PRIMM (Predict-Run-Investigate-Modify-Make) process:   * Predict – show the code and ask students what it does. * Run – enter the code and run it in the simulator and then on the micro:bit and observe what it actually does.   The starter is a small aspect of the main program. It demonstrates how to measure the moisture of soil in a plant pot. The learners will need to use croc clips attached to nails or unfolded paper-clips stuck into the plant soil. As part of this starter, learners must measure the reading for both dry and wet soil, and work out where to set the DRY\_GT threshold to (a figure where if the number is greater than this number, we assume the plant is dry and needs watering).  **Design context and brief**  Conditions for plants to grow – learners could discuss the right temperature, soil moisture and/or light level needed for the plants to grow.  Explain that the plant will need to be kept not too warm or not too cold. Similarly, the soil they are planted in must be not too wet or not too dry.  **Producing the programmable device**  Learners can be taken through this in one of two ways:   * Route option 1 – learners themselves create the code to meet the design brief, with help as required from the teacher. In this instance the main focus is the programming activity. * Route option 2 – The example code is treated as pre-written and downloaded onto the micro:bit ‘as is’. In this instance the main focus is on the overall system and its use in context.   To support simulation, the teacher should show  learners how to use the analog pin simulation. This is achieved by clicking on the mouse on the P pin and sliding it up and down, while holding the mouse button to change the analog reading, and sliding the thermometer up and down to change the temperature.  **Testing the System**  For opening a window and/or watering the plants, servo motors could be connected to the micro:bit to move a scale model of a window up and down, and/or knock a small bucket of water onto the soil if it is too dry.  Learners could also set this up as a test rig, with assistance as required from the teacher.  Servo motors used must be suitable for use with the BBC micro:bit. For example, the MonkMakes servo kit for micro:bit, or the servo kit for micro:bit from Kitronik  **Example programs**   * Starter example program - This program is for the starter activity (presentation slide 3). * Plant:bit program – In its simplest form the program could simply show the readings of temperature and moisture level, without any element of output control. Every 15 sec, scroll T+ temperature(C) M + moisture (0..1023) – the moisture measured by two crocodile clips on P0+GND and two nails in the soil.   An EVERY 15000 ms block should be used; it is not necessary to introduce variables at this stage, but learners should think about how they want to present the two readings every 15 seconds. The -example shows a scrolling setup, but there are other ways this could be presented. Learners will also learn more about temperature and moisture readings - they could take measurements and work out the best threshold values to use for later.  For a medium level of challenge, the program can be progressed by introducing a simple control aspect. If temp > hot, screen shows open window, lift optional servo P1 up to show window open; if temp < cold, screen shows closed window, drop optional servo P1. When it is too hot, the window is opened; when it is not hot, the window is closed. At this point teachers should choose if a real servo is used, or if the display is just going to show whether the window is open or closed; the display aspect is simple because a single value of open/closed could be represented by two icons. It could become a D&T focussed project here by using a scaled down model wooden house and actually physically attaching a servo to a small flap that opens and closes like a window.  For a much harder level of challenge, learners will need to water the plant with real water and also for a non-servo setup, work out how to display the status of both the window and the water delivery system. If moisture > dry, screen shows watering, move servo P2 to simulate tipping a water jug; if moisture < wet, screen shows not watering, return servo P2 to simulate levelling watering jug.  For a servo setup, the teacher could makes this into a D&T focussed project by working out how to mount and tip a small bucket of water into the plant pot.  For a non-servo project, the teacher needs to show learners how they could display two items at once (the -example file here shows an icon of a plant pot, with the left top LED plotted if the window is open, and the right top LED plotted if the water delivery system is on). Teachers could do a class-wide ideation session here to gather ideas from students on different ways to display two status values at the same time on the screen, and even show the example code at the end of this to see how it works and to analyse and add in the student ideas to get it working. |

|  |  |  |
| --- | --- | --- |
| **Differentiation** |  |  |
| **Basic** |  | **Extension** |
| * Provide the plant:bit program as a set program that can be downloaded ‘as is’, or edited as required. * Produce the program in sections, with the teacher demonstrating how to do these one or two blocks of code at a time. * Give learners a partially completed program for them to add the missing blocks of code. |  | * Think about how the system could also measure the light level in the greenhouse. * Think about how the system could change the light level in the greenhouse if it is too bright or too dark. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Resources** |  | | **Required files** icon-docicon-pdficon-ppt |
| * BBC micro:bits (V1 or V2) and associated USB download cables * 3 V power supplies for micro:bits (e.g. 2 x AAA battery packs) * Servo motors for opening the window and watering the plants (e.g. MonkMakes servo kit for micro:bit, or the servo kit for micro:bit from Kitronik). * Moisture probes (these could be created by using crocodile clips and two nails to go into the soil) * Computers with access to the internet |  | | icon-ppt Primary Presentation – BBC Micro:bit Smart Greenhouse  Example HEX programs:   * Starter example program * Plant:bit example program |
| **Additional websites** |  | |  |
| * **BBC Micro:bit homepage:** <https://makecode.microbit.org> * **How to upload HEX files to the program editor:** <https://support.microbit.org/support/solutions/articles/19000065686> * **Rayla Horticultural Society – Growing vegetables in a greenhouse:** <https://www.rhs.org.uk/vegetables/growing-in-greenhouse> * **Forbes – The best things to grow in a greenhouse:** <https://www.forbes.com/home-improvement/outdoor/best-things-to-grow-in-greenhouse/> * **CGTN – Greenhouse technology:** <https://newseu.cgtn.com/news/2023-09-12/How-new-greenhouse-technology-could-transform-UK-food-production-1mZJXZh2Mso/index.html> | | | |
| **Related activities (to build a full lesson)** |  | |  |
| **Starters** (Options)   * Use a mind map or spider chart to analyse the design brief and criteria. * Use the information on presentation slide 3 to analyse the starter program and how it works. | | **Plenary**   * Evaluate how well the device performed during testing. What went well and what could be improved? * Self/peer assess the completed system against the requirements of the design criteria. | |

|  |
| --- |
| **The Engineering Context** |
| * Programmable systems are an integral part of the world we live in today. Almost all electronic devices, from smartphones to washing machines to complex aircraft control systems, rely on programmable devices for them to function. It is therefore vital for systems engineers to develop skills in using programming to embed intelligence into electronic systems. * Food production is an excellent context to explore how programmable systems can be used to solve engineering, social and environmental problems. |

|  |  |
| --- | --- |
| **Curriculum links** | |
| **England: National Curriculum**  Design & Technology KS2   * apply their understanding of computing to program, monitor and control their products.   Computing 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. | **Northern Ireland Curriculum**  The World Around us KS2   * Plants and plant growth. * Design and make models. * The effects of adding components to simple   circuits. |
| **Scotland: Curriculum for Excellence**  Technologies   * TCH 2-09a, TCH 2-12a * TCH 2-13a, TCH 2-14a, TCH 2-15a | **Wales: National Curriculum**  Primary – Science and Technology   * Design thinking and engineering offer technical and creative ways to meet society’s needs and wants. * Computation is the foundation for our digital world. |

|  |  |  |
| --- | --- | --- |
| **Assessment opportunities** | | |
| * Formal teacher assessment of completed programs and systems. * Self/peer assessment of completed systems against the design criteria. * Informal assessment of practical skills used. | | |
|  |  |  |