USER GUIDE

Integrating DIY electronics into an eCraft2Learn project

eCraft2Learn

TABLE OF CONTENTS

Overview

eCraft2Learn is an EU funded project centred around researching, designing, piloting and validating an ecosystem based on digital fabrication and making technologies for creating computer-supported artefacts.

You can learn more about the eCraft2Learn Project in Teacher Guide 1 Introducing eCraft2Learn which can be downloaded at: https://project.ecraft2learn.eu/introducing-ecraft2learn/

The eCraft2Learn ecosystem primarily provides and supports the following digital tools and technology:

- Electronic components - for creating electronics circuits.
- Arduino Uno - for connecting to electronic components and programming of artifacts.
- Tinkercad circuits - for planning and simulation of circuit design ideas.
- Snap4Arduino - for programming the Arduino.
- Snap! - for programming AI (artificial intelligence) into projects.

Together these physical components and digital tools can be utilised for experimenting, learning and implementing DIY electronics into student projects.

This guide has been created to assist you to get started with creating and programming electronic artifacts using Arduino and electronic components. The guide will take you through the following:

- Learning about DIY electronics
- Planning and simulating electronic circuits
- Working with electronic components
- Connecting and programming an Arduino
- Adding artificial intelligence to artifacts

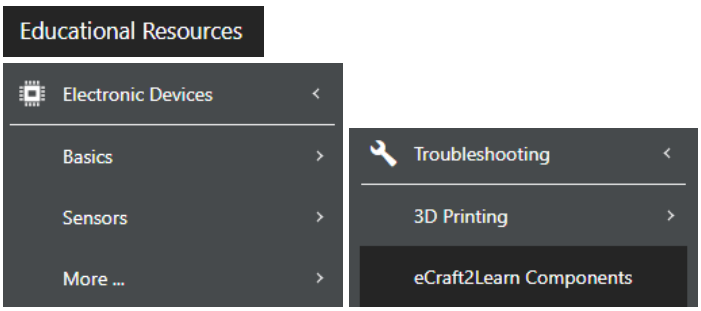
Throughout the guide, you will be signposted to other more detailed documentation from the eCraft2Learn academic research and project pilots. These documents provide a more detailed information and support for DIY electronics.

Learning about DIY electronics

Prior to planning any electronic circuit, it is important that students understand the various electronic components available to them and how they work. The eCraft2Learn ecosystem provides a range of documentation and open education resources to help educators and students to learn about DIY electronics.

EDUCATION RESOURCES

In the UUI students can access a series of education resources to learn about the various electronic components. Click on the Educational Resources Link in the top right of the interface, then click on Electronic Devices and the Troubleshooting links in the menu to access these resources.



In the electronic devices menu, the components are categorised into three sub-menus; Basics, Sensors and More… Selecting one of the sub menu's opens up a list of available resources. Each resource contains information to help the students learn more and many of the resources contain links to further resources available on the internet such as video tutorials.

The resources in the Troubleshooting menu will help if you are experiencing difficulties in using the technology.

PROJECT DOCUMENTATION FOR TEACHERS

In addition to the open education resources available in the UUI. The eCraft2Learn project has produced various research reports including a Teacher Training Manual which includes a series of 3D printing activities to get you started and a User Manual for the UUI.

A list of the main electronic components and tools can be found in
Teacher Guide 3 Setting up a eCraft2Learn learning environment which can be downloaded at:
https://project.ecraft2learn.eu/setting-up-an-ecraft2learn-learning-environment/

You can access technical training documentation and activities for Arduino and DIY electronics in Project Report D3.4 Section 3.5 at:
https://project.ecraft2learn.eu/wp-content/uploads/2018/05/eCraft2Learn_D3.4_M16_Manual-of-Craft-and-Project-based-Learning-STEAM-Training-for-Teachers.pdf

You can access further support for 3D printing in the UUI user manual in Project Report D4.5 Section 7 at:
https://project.ecraft2learn.eu/wp-content/uploads/2019/01/D4.5-User-manual-for-programming-of-computer-supported-artefacts-with-integrated-debugger-and-3d-modelling-simulation-and-printing-%E2%80%93-the-unified-user-interface-approach.pdf

Planning and simulating electronic circuits

Students should consider their use of electronics when planning their artifacts. At a very basic level this could be as simple as recognising what function is required and which input and output components will help them to achieve these outcomes.

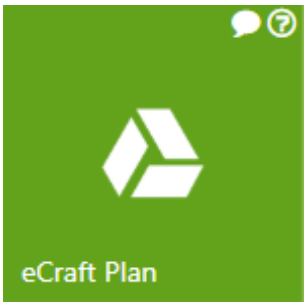
For example in the lighthouse exemplar project:

Required Function: The light at the top of the lighthouse should flash on and off when it gets dark.

Input device: A light sensor (LDR) to sense the light levels.

Output device: A Light Emitting Diode (LED) programmed by the Arduino to flash if the light level drops below a certain level.

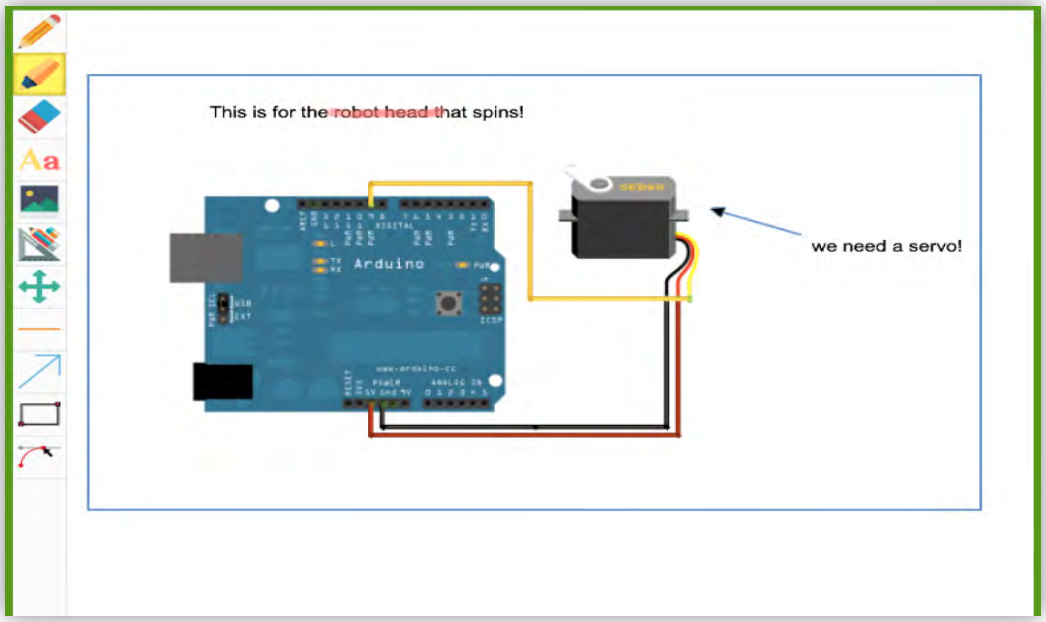
There are two tools in the UUI that will help students to plan and simulate their electronic circuits, eCraft Plan and Tinkercad Circuits.



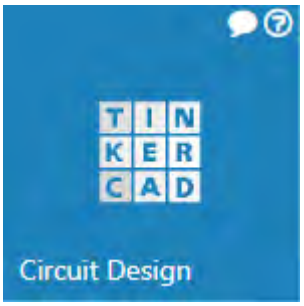
eCraft Plan

eCraft Plan is a sketching tool that provides tools for drawing different shapes and items. These include freehand drawing tools, erasor, text tool, shapes tool and image import functionality.

The image import functionality is particularly useful for planning electronic circuits and adding notes to annotate them.



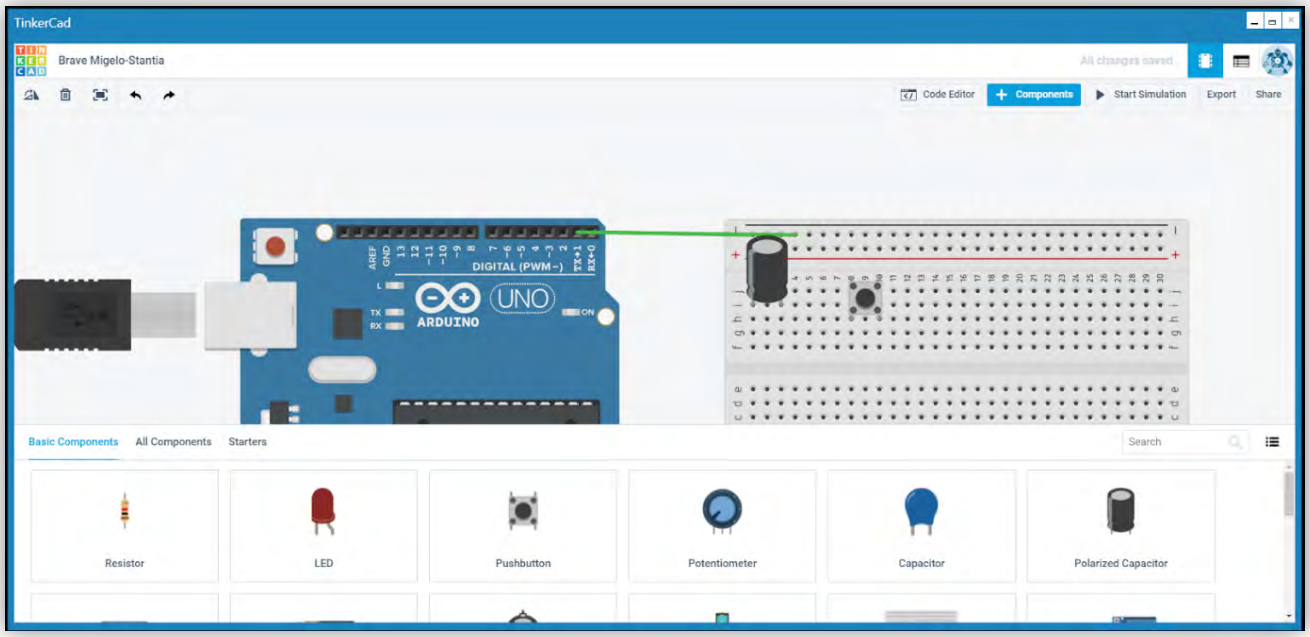
You can learn more about the eCraft Plan tool and how to use it in Project Report D4.5 Section 4.1 at:
https://project.ecraft2learn.eu/wp-content/uploads/2019/01/D4.5-User-manual-for-programming-of-computer-supported-artefacts-with-integrated-debugger-and-3d-modelling-simulation-and-printing-%E2%80%93-the-unified-user-interface-approach.pdf



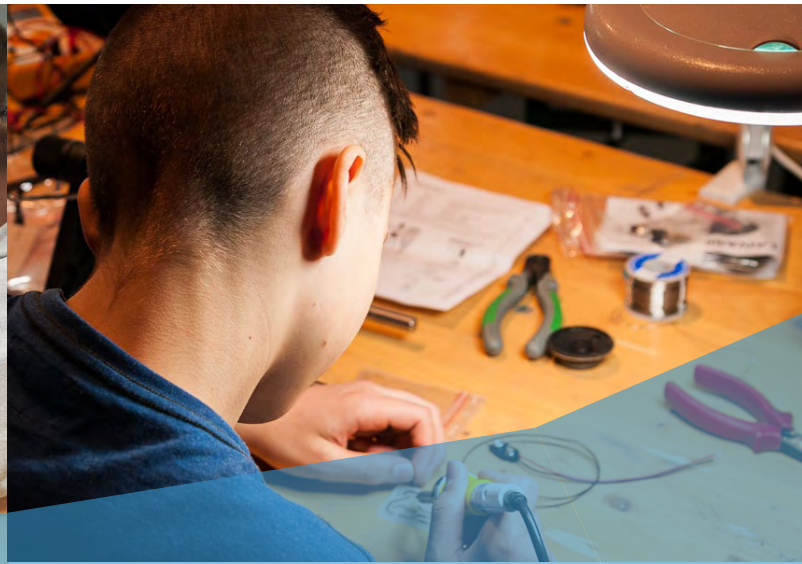
Tinkercad Circuits

Tinkercad Circuits can be used for simulating different electronic elements including the Arduino Uno microcontroller.

This is a useful tool tool for designing circuits prior to physically making them as the circuit design can be simulated and easily edited until a working circuit design has been developed.

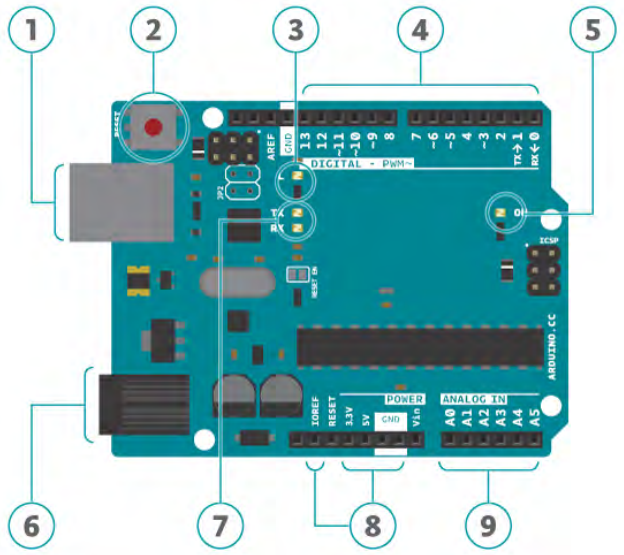


You can learn more about how Tinkercad operates in Project Report D4.5 Section 6.1 at:
https://project.ecraft2learn.eu/wp-content/uploads/2019/01/D4.5-User-manual-for-programming-of-computer-supported-artefacts-with-integrated-debugger-and-3d-modelling-simulation-and-printing-%E2%80%93-the-unified-user-interface-approach.pdf



Working with an Arduino and electronic components

Arduino boards are small computers with which you can read information from a variety of sensors as well as control lights, motors and other things. Arduino UNO is the best board to get started with electronics and coding. If this is your first experience tinkering with the platform, the UNO is the most robust board you can start playing with. The UNO is the most used and documented board of the whole Arduino family. The diagram below shows the various elements of an Arduino Uno board.



1. USB socket: When uploading a program to the UNO it is done through a USB cable.
   The USB cable is connected to a computer and to the USB socket.
   The cable also provides power to the UNO board.

2. Reset button: When the reset button is pressed, the program uploaded to the board is restarted.

3. On-board LED: This LED is connected to digital pin 13. You can turn the LED on or off
   by programming pin 13.

4. Digital pins: There are 14 digital pins, numbered from 0 to 13.

5. ON LED: This LED is turned on when the UNO is supplied with power.

6. Power socket: To provide the UNO board with power you can connect a battery or adapter
   to the power socket. The recommended input voltage is 7-12V.

7. TX and RX LED:s: The TX LED blinks when data is sent by the UNO over the serial port.
   The RX LED blinks when the UNO receives data over the serial port.

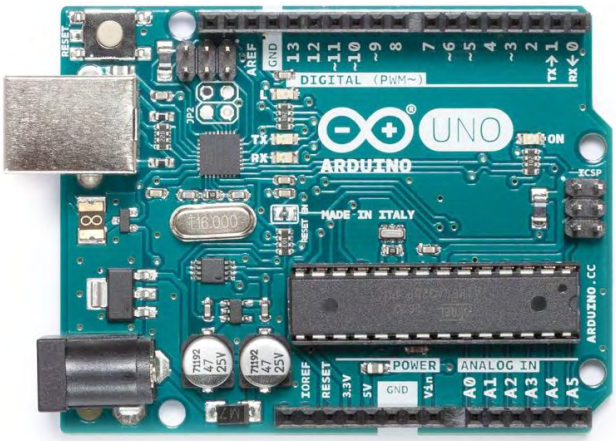
8. Power and ground pins: The power pins are marked IOREF, 3.3V and 5V.
   The ground pins are marked with GND.

9. Analogue pins: There are six analogue pins numbered from A0 to A5.

CONNECTING AN ARDUINO TO ELECTRONIC COMPONENTS

In order to create circuits each group of students will need access to an Arduino Uno board, a number of different electronic components, a breadboard and jumper wires.

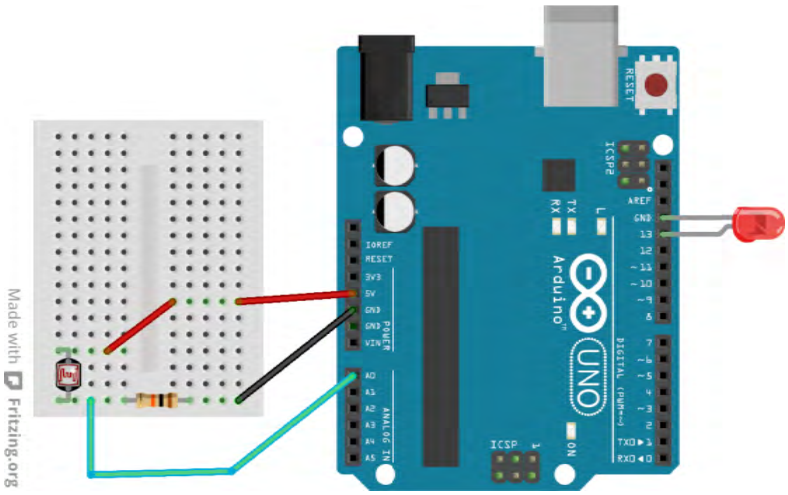
Arduino boards have a number of pins that are numbered and grouped by functionality. On the Arduino UNO, shown in the image below, there are a group of digital pins and a group of analogue pins.



The digital pins are a group of 14 pins (numbered 0 to 13), you can connect digital inputs and outputs to these pins.

The analogue pins are a group of 6 pins (labelled A0 to A5), you can connect analogue sensors to these pins.

Electronic components and circuits can be connected directly to the analogue and digital pins as shown in the example below. This can be done either plugging a component directly into the board - as shown on the left of the diagram by connected the LED between digital pins 13 and Ground. Alternatively jumper wires can be used to connecting the Arduino to components in a breadboard or another assembled circuit. The left of the diagram below shows a simple light sensor circuit assembled on breadboard using an LDR and a resistor. The circuit is powered by connecting the board to the 5V and Ground pins on the Arduino and an analogue input is being taken from the sensor by connecting a jumper wire from the circuit to pin A0 on the Arduino.



You can access technical training documentation and activities for Arduino and DIY electronics including a hands-on practical exercise in Project Report D3.4 Section 3.5 at:
https://project.ecraft2learn.eu/wp-content/uploads/2018/05/eCraft2Learn_D3.4_M16_Manual-of-Craft-and-Project-based-Learning-STEAM-Training-for-Teachers.pdf

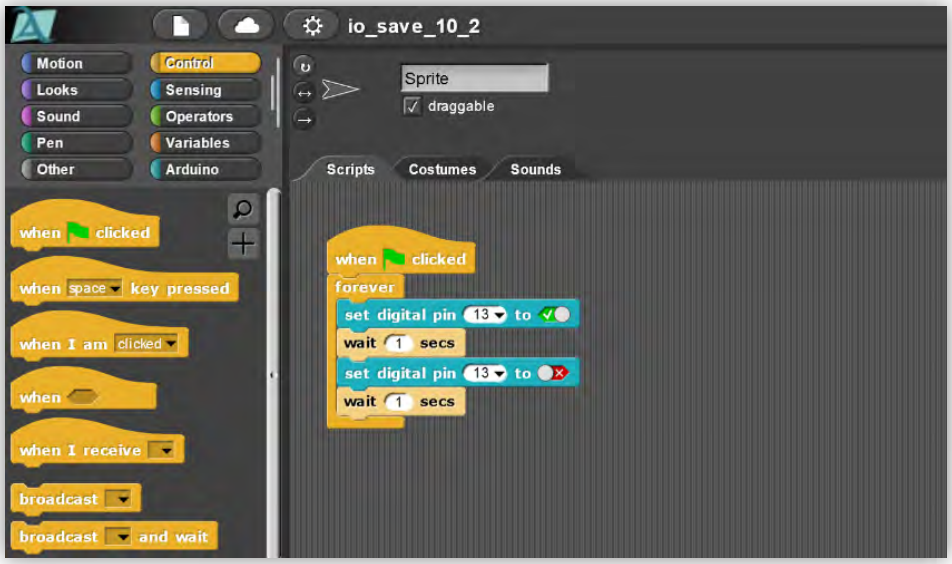
Connecting and programming an Arduino

Since Arduino boards, unlike computers, does not have a screen or a keyboard, you need to use external software running on a different computer to write programmes for the Arduino. This software can be the text-based Arduino IDE (Integrated Development Environment) or the visual programming environment of Snap4Arduino. In the eCraft2Learn project, it is recommended that you use Snap4Arduino to program the Arduino boards. This is available through the Unified User Interface (UUI).

PROGRAMMING IN SNAP4ARDUINO

Similarly to Snap!, in Snap4Arduino different sets of blocks are arranged in a library of different groups, collected in the top left panel of the editor and represented by clickable buttons of different colours. Once a group is chosen, you can add commands to your program by dragging and dropping the blocks from the left bottom panel to the centre of the editor. Snap4Arduino includes an additional group of blocks, labelled as 'Arduino', through which it is possible to connect the software environment with the external hardware, to send commands and to read pin values. In fact these blocks are commands that you can insert into the program: these commands will be sent to the Arduino board when the program is running.

The program shown below is an example of how Snap4Arduino can be used to blink the embedded led connected to the pin 13 of the Arduino board.

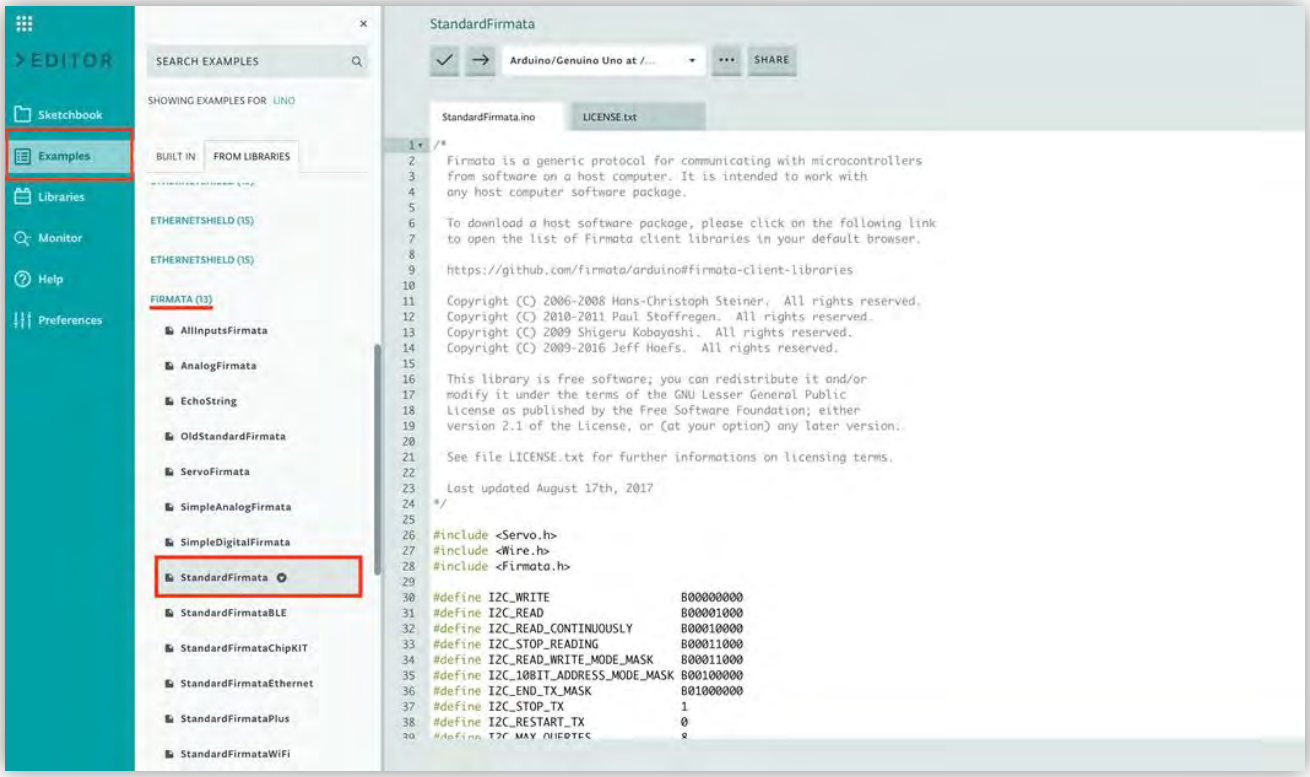


CONNECTING TO SNAP4ARDUINO

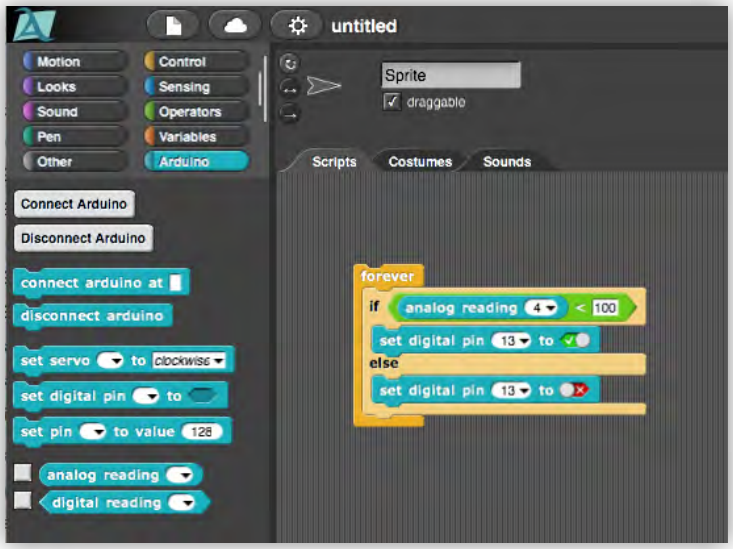
In order to connect an Arduino board to Snap4Arduino and to make communication between them available for Snap! programs, it is required to pre-load a program onto the Arduino board. This is a specific 'Firmata' software which will run on Arduino and communicate typically through the USB connection.

To upload the Firmata: close Snap4Arduino if open, connect the Arduino board to a USB port on the computer, and open the Arduino IDE environment clicking on the Arduino IDE tile in the UUI.

In the Tools menu, select the board version and the serial port where the board is connected. After that, go to “Examples”-> “From Libraries”-> “Firmata”-> “Standard Firmata”, select the file “Standard Firmata” (Figure 32) and upload the file clicking on the arrow icon. When the upload finished, Arduino is ready to communicate with Snap4Arduino. Now you can close the Arduino IDE and open Snap4Arduino.



Finally, click on the “Snap4Arduino” tile in the UUI. The Snap4Arduino editor will open in a new UUI window. Click on the “Arduino” set of blocks and select “Connect Arduino”.



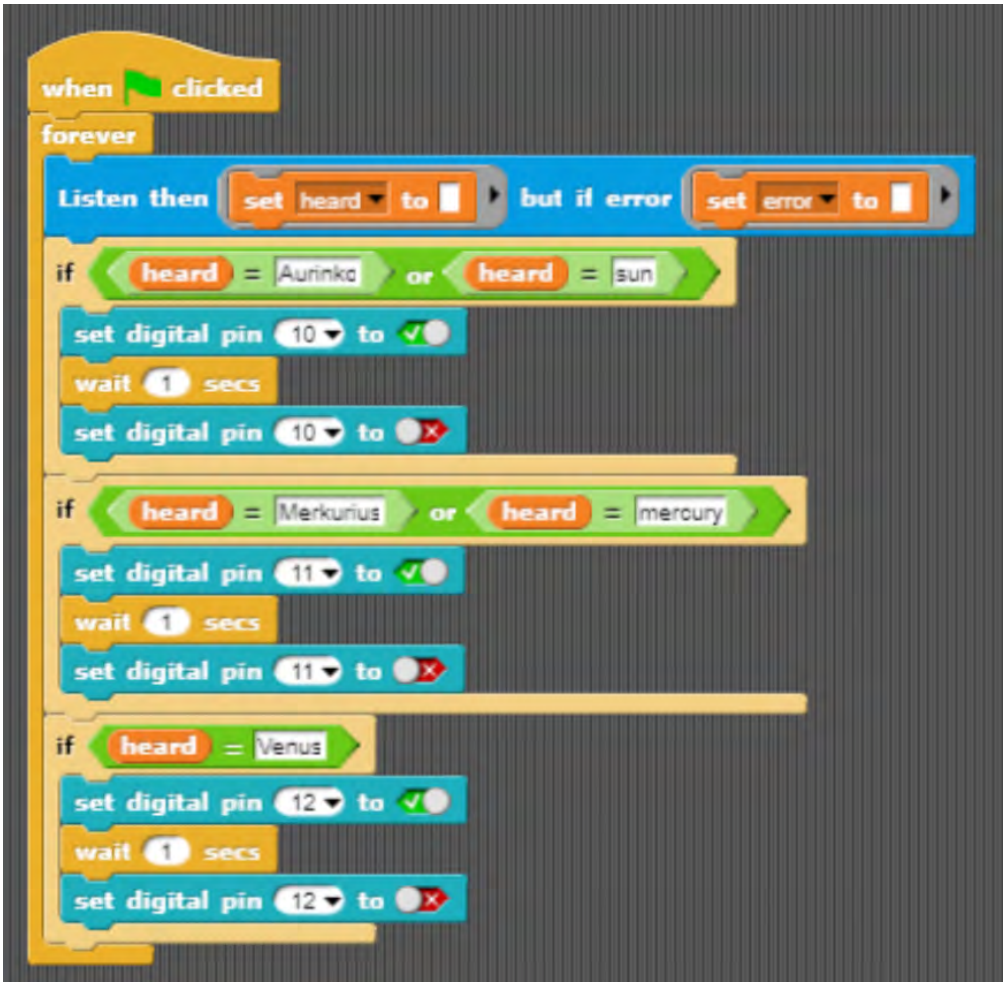
Once the board is connected to Snap4Arduino, programs can be executed in the board.

You can access the full instructions for connecting and running Snap4Arduino in Project Report D4.5 Section 7.2 at:
https://project.ecraft2learn.eu/wp-content/uploads/2019/01/D4.5-User-manual-for-programming-of-computer-supported-artefacts-with-integrated-debugger-and-3d-modelling-simulation-and-printing-%E2%80%93-the-unified-user-interface-approach.pdf

Adding artificial Intelligence (A.I.) to artifacts

Students can build artefacts that rely upon A.I. cloud services and machine learning. This enables them to build artefacts that respond to voice commands, generates speech, recognises images, and more.

The eCraft2Learn Snap! A.I. blocks library includes blocks for training the computer to recognise images, audio and other data. Students can use this for example to build programs where hand or body gestures control how their artefacts move or behave. An artefact can also be trained to behave differently depending upon what is in front of its camera.



Several sample projects using the A.I. extensions to Snap! have been developed and trialled in the eCraft-2Learn project pilots, you can learn more about these projects in
Project Report D5.5 Sections 4.11, 4.16 and 4.17 at:
https://project.ecraft2learn.eu/wp-content/uploads/2019/01/D5.5-Small-scale-case-pilot-report-and-good-practice-videos.pdf

You can also view the best practice video for one of these projects at:
https://www.youtube.com/watch?v=Ipm7DKHeySY&index=11&list=PLgKtrHOACe-J6bvq-ka5ue4ERs142f4De

OTHER PROJECT EXAMPLES AND IDEAS INCLUDE:

- A robot (virtual or real) that responds to voice commands to move and turn.
- A device that listens for the name of a famous person or place and then speaks its description obtained by querying Wikipedia.
- A robot that can send images from its camera to cloud services that respond with a description of what is in front of it and the robot can act differently depending upon what it sees.
- A vehicle that can be constructed that has been trained to drive to the left or right depending upon whether the student in front of a camera is leaning left or right. It can be programmed to stop when it recognises an outstretched hand.
- Different coloured LEDs that can be turned on or off by speaking the name of the colour.
- A box with a camera and speaker can be built so that when a leaf is placed inside it says what plant it is from (after the students have trained it to classify different leaves).
- A gadget with a camera that can be trained to sound an alarm is someone falls down and fails to get up.

The education and training material for students and teachers/coaches on A.I. programming in Snap! and Snap4Arduino can be found in the Education resources that are available through the UUI at:
https://ecraft2learn.github.io

You can learn more about how to import and use the Snap! A.I. block in Project Report D4.5 Section 7.1.3 at:
https://project.ecraft2learn.eu/wp-content/uploads/2019/01/D4.5-User-manual-for-programming-of-computer-supported-artefacts-with-integrated-debugger-and-3d-modelling-simulation-and-printing-%E2%80%93-the-unified-user-interface-approach.pdf

eCraft2Learn

https://project.ecraft2learn.eu/

office@ecraft2learn.eu

01/2017 – 12/2018

This project has received funding from the European Union's Horizon 2020 Coordination & Research and Innovation Action under Grant Agreement No 731345.