PEEK
PURPLE EARLY ENGINEERING KIT

HANDBOOK

A Product of the Purple Future Trust.

Purple Future Trust

Supported by the

ROYAL ACADEMY OF ENGINEERING

You will learn

- Coding with Scratch and Arduino
- Basic Electricity and Electronics
- Basic Electrical Safety
- Fundamentals of Sensors and Actuators
- Entry-level Robotics

S4A SCRATCH ARDUINO

Comes with Scratch for Arduino installed!

Take a PEEK into the world of Engineering

Contact us:
Purple future Trust:
info@purpletrust.org |
+263772584350

purpletrust.org

INTRODUCTION

The **PEEK** is for children aged seven and up, teachers and parents who may or may not have a technical background. It introduces the concepts of electronics and programming in a fun and interactive way. This handbook provides training and guidance in early-stage concepts of the ' **E** ' in STEM, Engineering. The PEEK is our contribution to the drive towards STEM-literacy for the Southern African Region, and Africa as a whole.

The handbook is updated frequently. The most recent update will be available on the Purple Future Trust website http://www.purpletrust.org. Parents and Guardians are encouraged to subscribe to our mailing list to receive updates on new manual releases, new tutorials, PEEK-related events, and other exciting news.

We encourage parents to use the PEEK to facilitate parent-child bonding, which accelerates learning.

OUR MISSION

Our mission is to integrate technology with the playground, for you to learn as you play (to learn almost effortlessly!). Learning about technology becomes an extension of your playtime. Our kit unlocks your ' inner inventor ' and gives you an early advantage to creative thinking in the technology space.

ACKNOWLEDGEMENTS

Purple Future Engineers in Harare, Zimbabwe, developed this handbook via an iterative process of consultation with a sample group of beneficiaries. The document was extensively reviewed internally. It is in the process of being reviewed by the Ministry of Science and Technology, as well as the Ministry of Primary and Secondary Education. Special thanks goes to all those who support this project.

Purple Future Trust Engineers and Trustees:
Eng. Grace Pisirai; Eng. Ian Nyasha Mutamiri

Trustees:
Douglas Makonyere, Linda Khumalo, Ethni Suliali

Content Developer:
Samuel Tsokoroko

Consultant:
Eng. Andrew Mutamiri (Asmasol Consult)

Writers:
Samuel Tsokoroko, Ian Mutamiri, Masimba Gamira

Graphics:
Masimba Gamira (Massiv Digital Agency)

Supported by the Royal Academy of Engineering's Africa Prize for Engineering Innovation

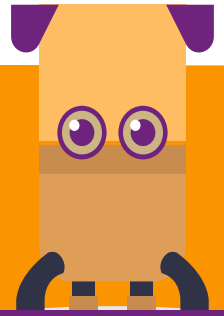
ROYAL ACADEMY OF ENGINEERING

ABOUT THIS GUIDE

The PEEK guide is an essential component of the PEEK Kit. The guide contains lessons or tutorials that will help you learn how to design interactive physical systems using software and hardware that can sense and response to the environment around you. This is called Physical Computing. According Wikipedia, Physical Computing means building interactive physical systems by the use of software and hardware that can sense and respond to the analogue world. Physical computing has helped humankind come up with systems that make our lives better, such as

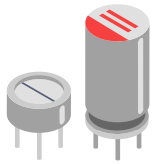
- **Traffic control systems,**
- **Agriculture automation processes,**
- **Robots**

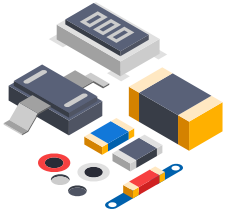
The PEEK comes with this guide, as well as a USB stick or memory card containing code files, diagrams, and other helpful material. You can also download additional files off our website as and when they are available (http://www.purpletrust.org).



WHAT YOU WILL LEARN

This handbook contains 5 beginner tutorials to get you started. You will find more on our website, http://www.purpletrust.org. You will gain strong foundation skills on physical computing. After going through this handbook, you should be able to read, write, or edit code.





PACKING LIST - PEEK COMPONENTS

1	Arduino Uno + USB Cable
2	Breadboard
3	Resistor
4	Jumper
5	Motor Driver
6	Motor
7	Potentiometer
8	Flame Sensor
9	HDX Vibrator
10	Thermistor
11	Servo Motor
12	LDR
13	LCD Crystal Display
14	LED
15	Buzzer
16	Push-Button

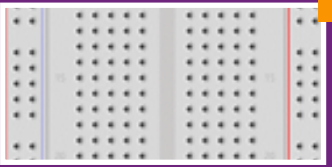
The PEEK comes with several electronic components. Familiarize yourself with the items contained in the kit. Identify the component, and read about it below.

1

Arduino Uno + USB Cable

Arduino Uno is a microcontroller board used to create interactive electronics objects.

2



Breadboard

A breadboard is used to create circuits and connect different sensors and actuators to the Arduino board through jumper wires, and electronic components

3

Resistor

A resistor is an electrical component that limits or regulates the flow of electrical current in an electronic circuit.

4

Jumper

A jumper is a short length of conductor used to close, open or bypass part of an electronic circuit.

5

Motor Driver (1293)

L293D is a motor driver IC used to control motors by amplifying the current of weak signals from the Arduino.

6

Motor

Motor is an electrical device that converts electrical energy into mechanical energy.

7

Potentiometer

Potentiometer is a variable resistor with a third adjustable terminal.

8

Flame Sensor

Flame sensor is one kind of detector which designed for detecting the presents of flame and it is very sensitive to light.

9

HDX Vibrator

HDX vibrator is a used to detect vibration of some kind.

PACKING LIST - PEEK COMPONENTS cont'd

10

Thermistor

Thermistor is a type of resistor whose resistance is dependent on temperature.

12

Light Dependant Diode (LDR)

An LDR is a component that has a (variable) resistance that changes with the light intensity that falls upon it. This allows them to be used in light sensing circuit.

14

LED

Light-emitting diode (LED) is a two-lead semiconductor light source. It is a p - n junction diode that emits light when activated. When a suitable current is applied to the leads, electrons are able to recombine with electron holes within the device, releasing energy in the form of photons

16

Push Button

A push-button (also spelled pushbutton) or simply button is a simple switch mechanism for controlling some aspect of a machine or a process.

11

Servo Motor

A servo motor is a rotary actuator that allows for precise control of angular or linear position, velocity and acceleration.

13

LCD Crystal Display

A LCD Crystal display is a liquid-crystal display used to display limited graphics.

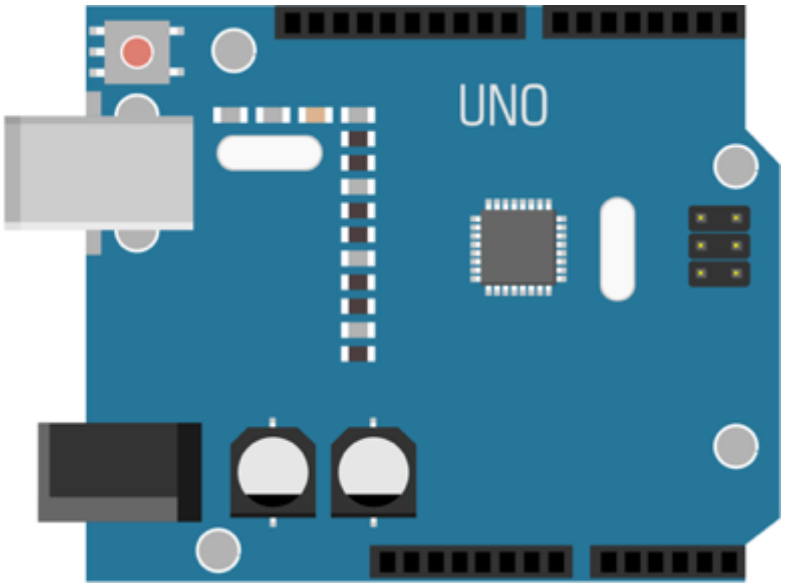
15

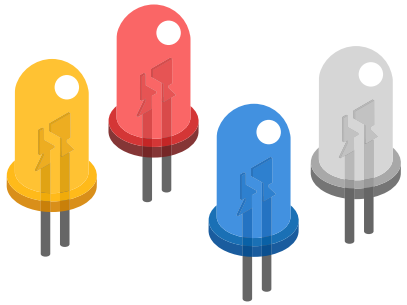
Buzzer

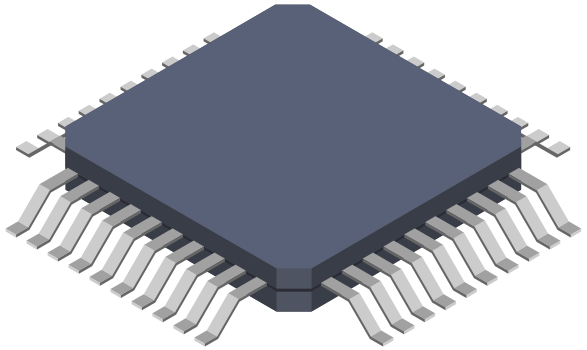
A buzzer or beeper is an audio signalling device, which may be mechanical, electromechanical, or piezoelectric. Typical uses of buzzers and beepers include alarm devices, timers, and confirmation of user input such as a mouse click or keystroke.

PUTTING PARTS TOGETHER

You can connect various sensors and actuators to an Arduino. You can also tell or program an Arduino to do a job in S4A, Scratch for Android. This is called coding, or programming.







WHAT IS AN ARDUINO?

An Arduino is an open-source microcontroller development board. Simply put, you can use the Arduino to read sensors and control things like motors and lights. You can upload programs to this board, and the board can then interact with things in the real world. With this, you can make devices that respond and react to the world at large.

Think about the many different things you can do. If there is something that is in any way controlled by electricity, the Arduino can interface with it somehow. Even if it is not controlled by electricity, you can probably still use things which are (like motors and electromagnets), to interface with it.

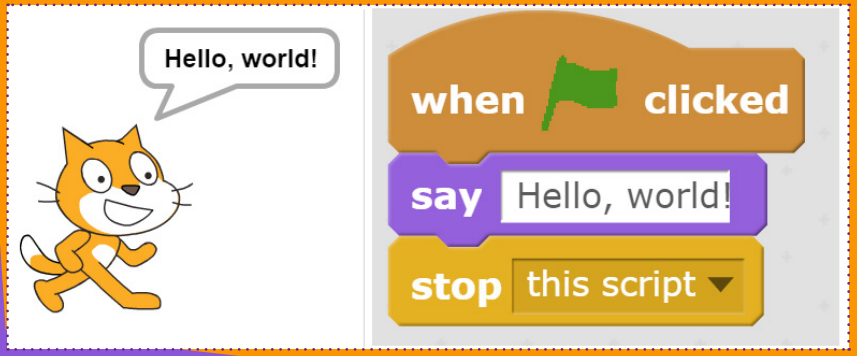
SCRATCH

WHAT IS SCRATCH

Scratch is a computer program that is used to make other computer programs. You 'write' programs (or code) in Scratch using blocks. This is why Scratch is called a Visual Programming Language. Scratch is developed by the MIT Media Lab and is used in many parts of the world.

SCRATCH BLOCKS

Blocks are puzzle-piece shapes that are used to create code in Scratch. The blocks connect to each other vertically like a jigsaw puzzle, where each data type has its own shape and a specially shaped slot for it to be inserted into. A series of connected blocks is called a script.



Scratch has many kinds of blocks, but for starters, we will introduce you to the following:

1. **Event**: Event blocks control events and the triggering of scripts.
2. **Control**: Control blocks control scripts.
3. **Sensing**: Sensing blocks detect things.
4. **Operators**: Operator blocks perform math functions.

SCRATCH FOR ARDUINO (S4A)

Scratch for Arduino (S4A, in short) is a special kind of Scratch, which was modified to interact with our Arduino board. The Citilab Smalltalk Team developed it in 2010.

We will be making Scratch programs and uploading them to the Arduino.

You will need a computer to make Scratch programs.

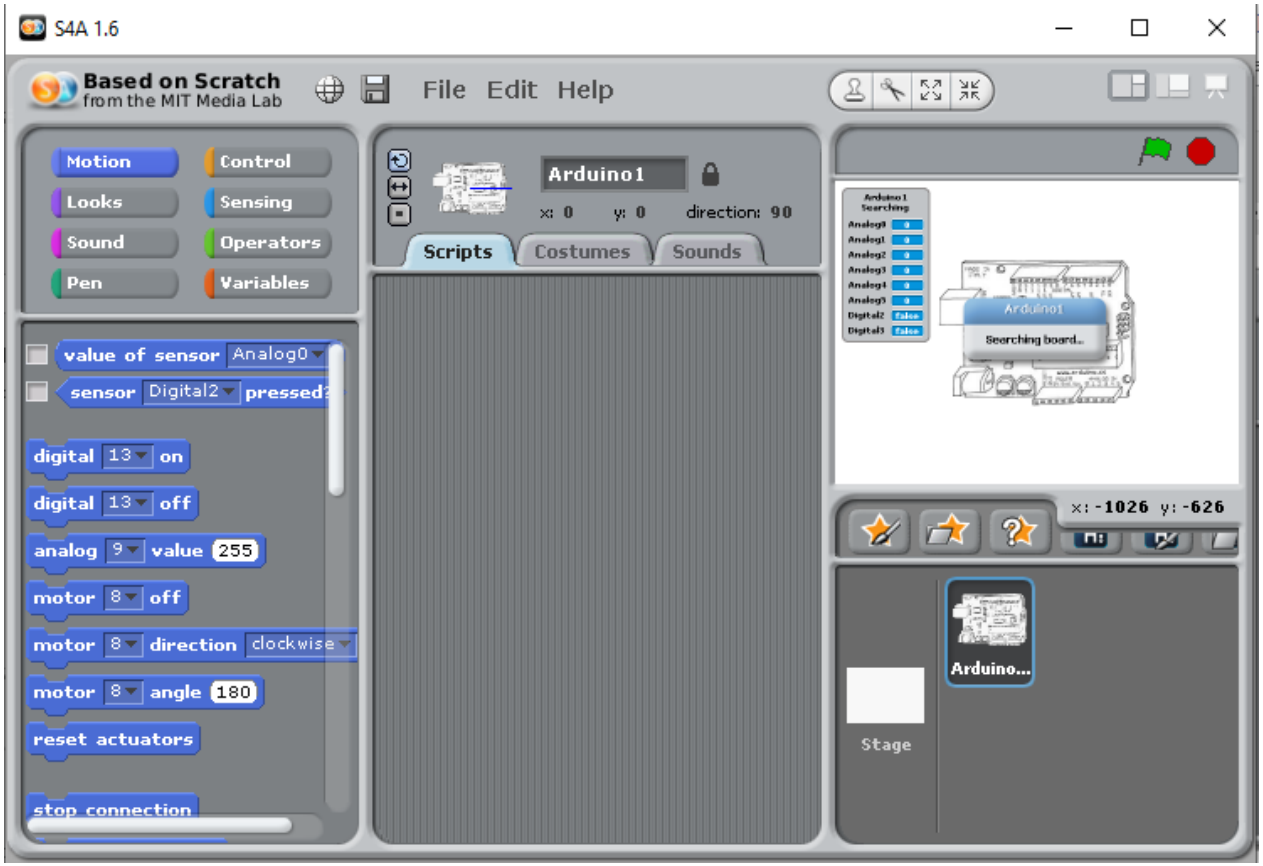
DOWNLOAD AND INSTALL

Installing S4A requires you to install software on your computer. Download S4A by visiting http://s4a.cat/

If you don't have an internet connection, look for the S4A folder on your PEEK USB Stick or Memory Card, and install it from there.

Download and install the appropriate S4A version for your Operational System. There are 3 available programs: For MAC, Windows and Linux (including Raspberry Pi)

Figure 1: This is what S4A (version 1.6) looks like.



Ohm's Law

Ohm 's law is a basic but very important law in electronics. It is a physics law that states that current passing through a conductor is proportional to the input voltage.

- **Voltage (V)**: is the measure of electrical potential. - It is measured in **Volts (V)**
- **Current (I)**: is the amount of flow through a conductive material. -It is measured in Amperes or **Amps (A)**
- **Resistance (R)**: is the material 's opposition to the flow of electric current. -It is measured in **Ohms (Ω)**

LET'S CODE!

Let ' s begin with some basic S4A examples, complete with the Arduino schematics.

INTRODUCTION: S4A OVERVIEW - BLINKING A LIGHT

Used Blocks-Sensing

Wait | Digital OFF | Digital ON

Learning Outcomes

Get to know your Arduino
Become familiar with the S4A interface.
Learn how to drag, drop and change blocks.

Used Blocks - Control

When Green Flag Clicked

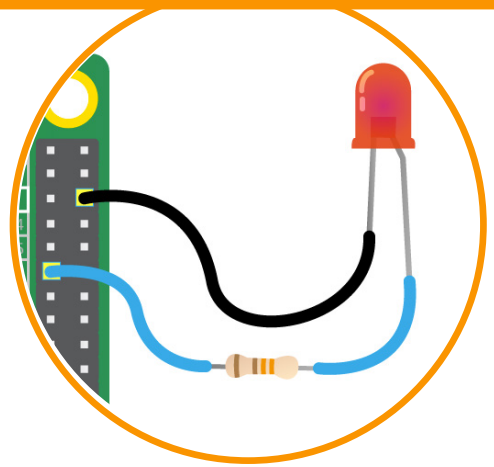
Level:
Beginner

The simplest program we are going to write is for turning on and off an LED light. This is for us to get used to the S4A interface, and how to use blocks.

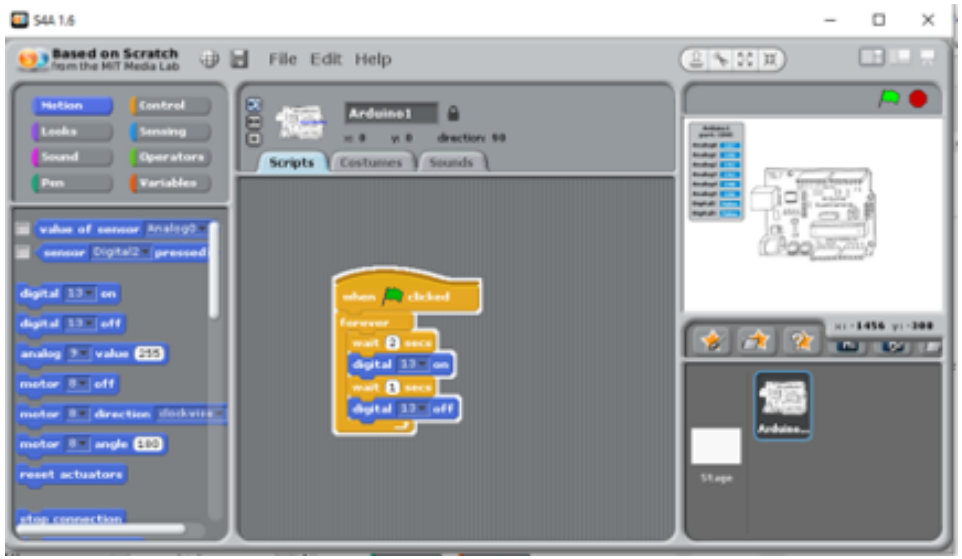
CONNECTING THE LED

Pick an LED from the kit.

Insert the longer wire into pin 13, and the shorter one into the ground pin as below:



THE S4A INTERFACE



BLOCK TYPES

When you start S4A on your computer, it will look like the picture above. On the top left-hand side, you will find 8 block types. Right below, you will find the blocks corresponding to the block type selected. To add a block to the Scripts tab (middle section of the interface), click (and hold) the block, drag it to the Scripts area, and drop it there.

FOLLOW THESE STEPS

STEP 1

Select the Control block types.

STEP 2

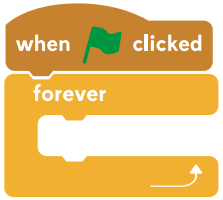
Click and drag

to the Scripts area.

With this block, you tell the Arduino what to do when the green flag is clicked.

STEP 3

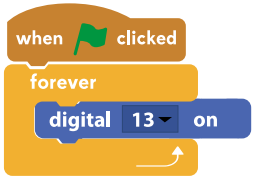
Find the forever block under Control blocks, and place it just underneath the When Green Flag Clicked block. The forever block repeats whatevew blocks are inside it. You should have:



STEP 4

Find the 

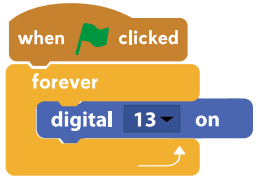
block under Motion blocks and place it underneath the forever block. This turns whatever is connected on the specified pin (in our case, the LED on pin 13) on. You should have:



STEP 5

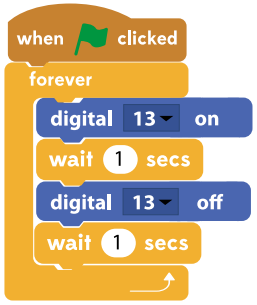
Find the 

block under Control blocks and place it underneath the digital on block. The wait block makes our program pause for the number of seconds specified. You should have:



STEP 6

Use the same method to build something like this: Make sure you pick the digital off block after the first wait.



Make sure you pick the digital off block after the first wait.

RUNNING YOUR PROGRAM

Make sure your Arduino is connected to the USB port of your computer. Press the green flag and the LED should turn on and off repeatedly, with one second breaks in-between.

NEXT STEPS

Try to add different combinations of digital on/off and wait blocks with different times. Have fun!

PROGRAM 1: BLINKING AN LED LIGHT

Level: Beginner

Learning Outcome

Learn how to use the breadboard (how to connect components onto the breadboard and Arduino).

Used Blocks

Control

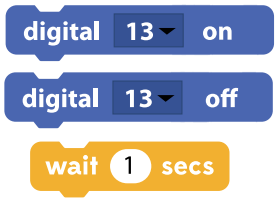
When Green Flag Clicked

Sensing

Digital ON

Digital OFF

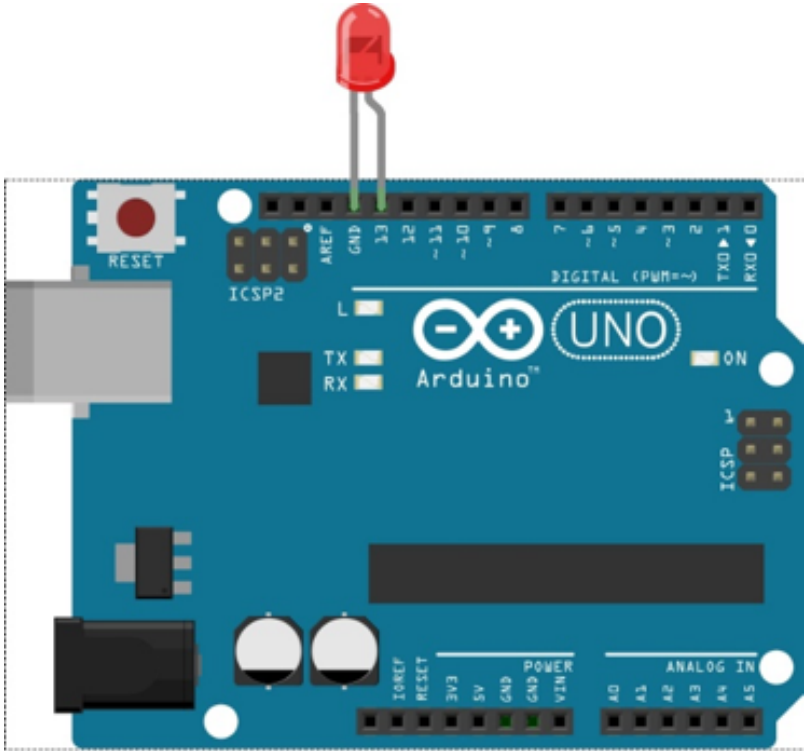
Wait



You Will Need

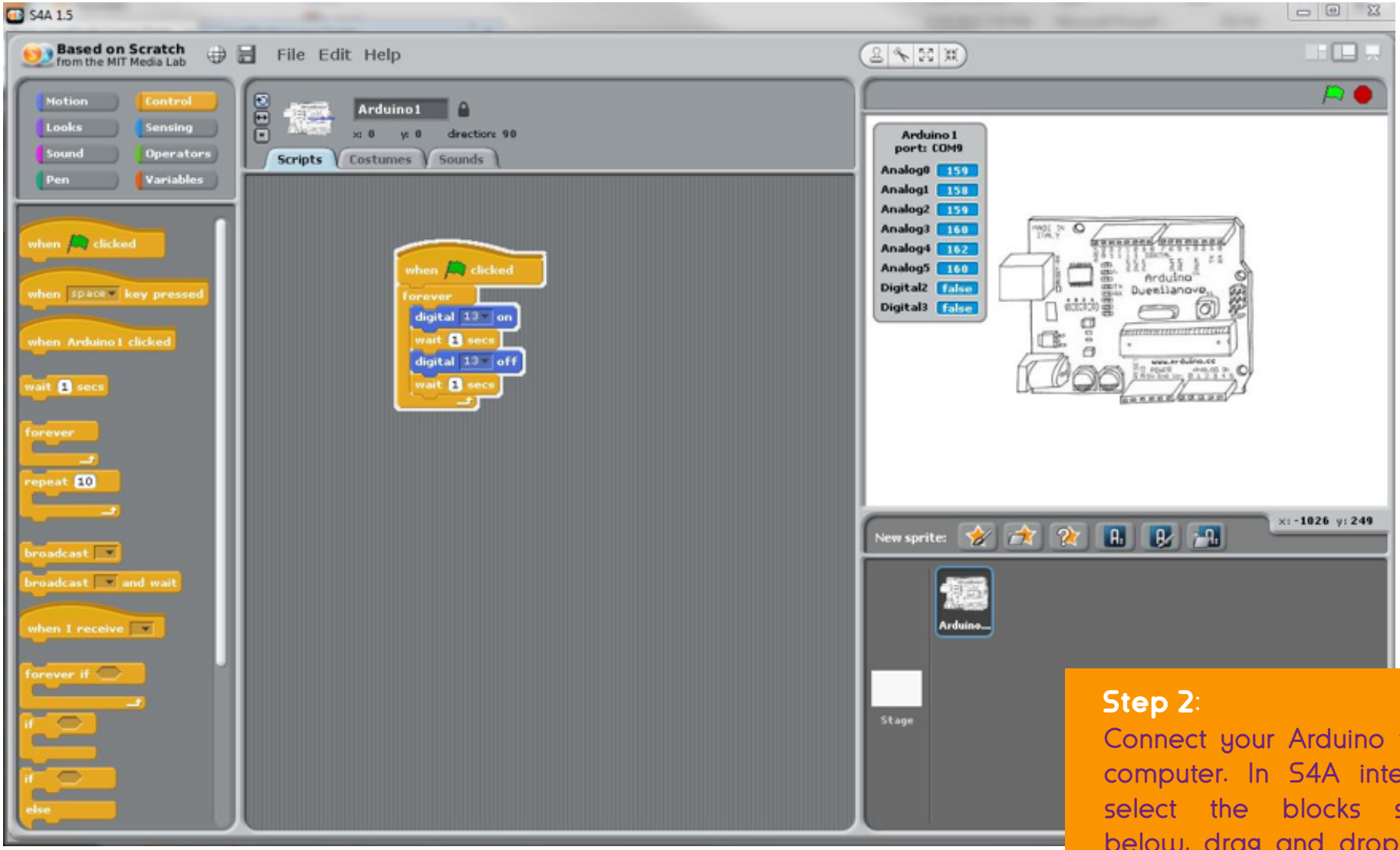
- Breadboard
- LED
- Resistor (220Ω)

STEPS



Step 1:

Connect a LED and a 220Ω Resistor on the breadboard and with Arduino according to the schematic below.



Step 2:
Connect your Arduino to the computer. In S4A interface, select the blocks shown below, drag and drop them onto the scripts area.

PROGRAM 2: TWO LEDs

Level: Beginner

In the following program the two LEDs with light up one after the other.

Learning Outcome

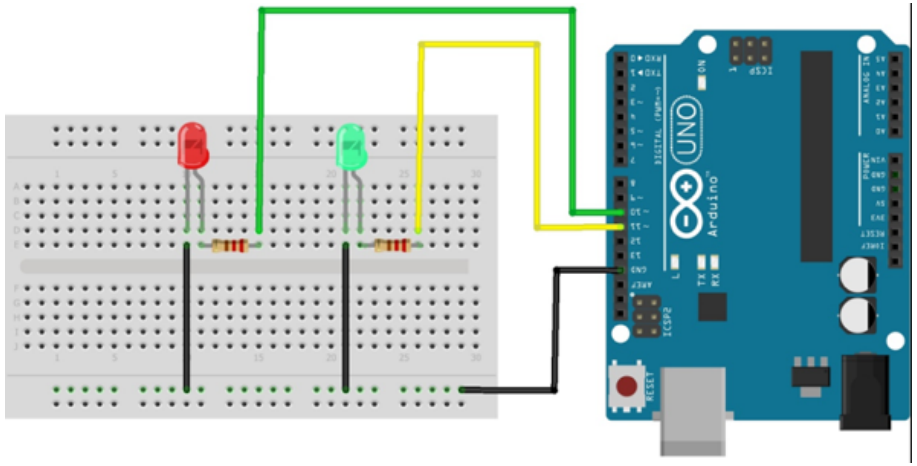
Learn how to connect two LEDs onto the breadboard and to the Arduino.

Used Blocks

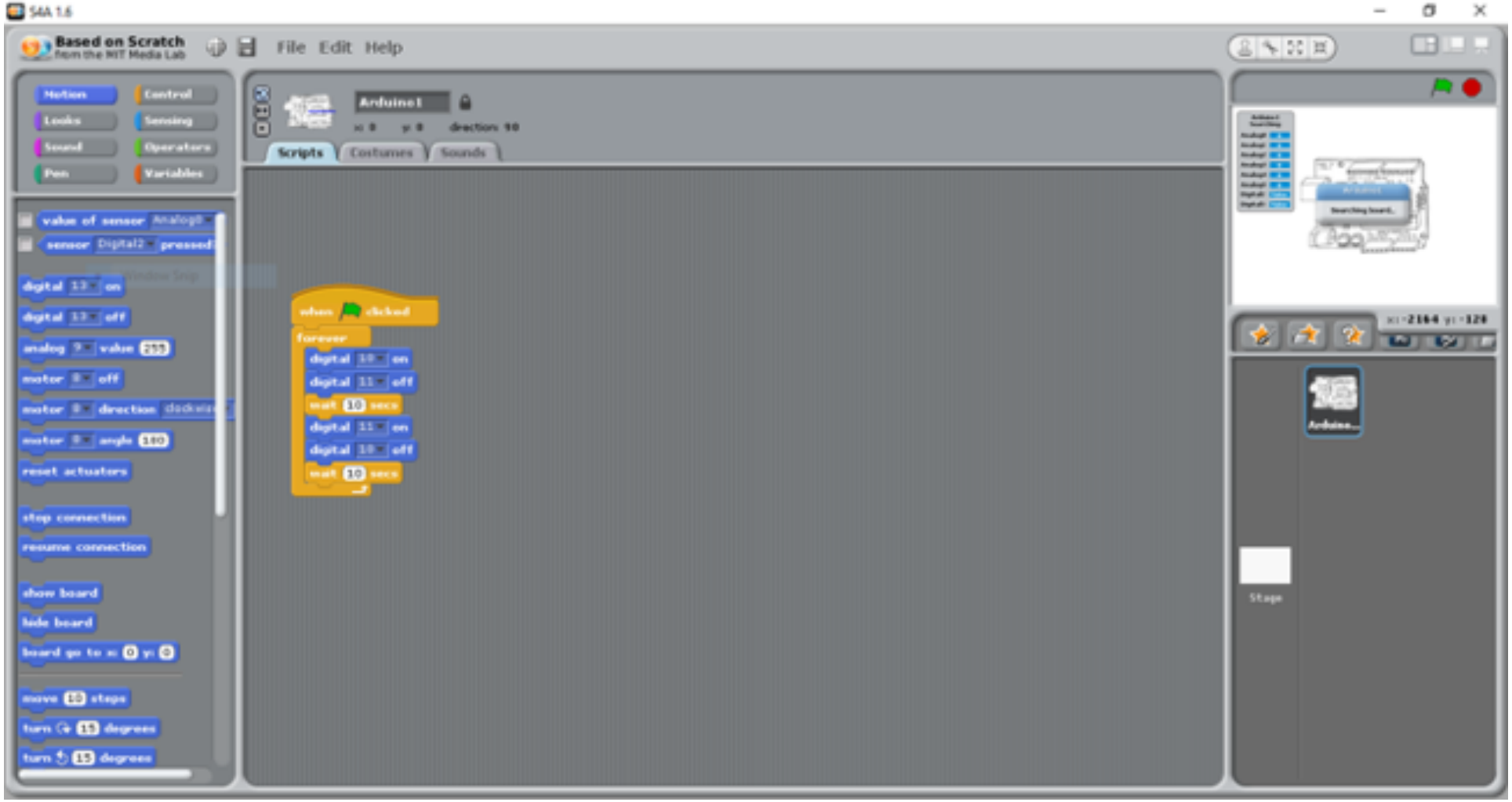
You Will Need

- Breadboard
- LED
- Resistor (220Ω)

Connect as shown here



In this exercise we will mount two LEDs controlled by a button. The objective is turn on the LEDs on whenever the button are pressed, and turn it off when not pressed. Connect the Aduino , Button and LEDs as shown in the diagram above . Code the following code, drag and drop blocks in the scripting area as follows:


S4A 1.6
Based on Scratch
from the MIT Media Lab
File Edit Help
Motion
Control
Looks
Sensing
Sound
Operators
Pen
Variables
Arduino1
Scripts
Costumes
Sounds
when clicked
forever
digital 10 on
digital 11 off
wait 10 secs
digital 11 on
digital 10 off
wait 10 secs
value of sensor Analog0
sensor Digital2 pressed
digital 13 on
digital 13 off
analog 9 value 255
motor 0 off
motor 0 direction
motor 0 angle 180
reset actuators
stop connection
resume connection
show board
hide board
board go to x: 0 y: 0
move 10 steps
turn 15 degrees
turn 15 degrees
Stage

PROGRAM 3 : BUTTON + LED

Level: Beginner

In the following program when you push the button, the LED with light up.

Learning Outcome

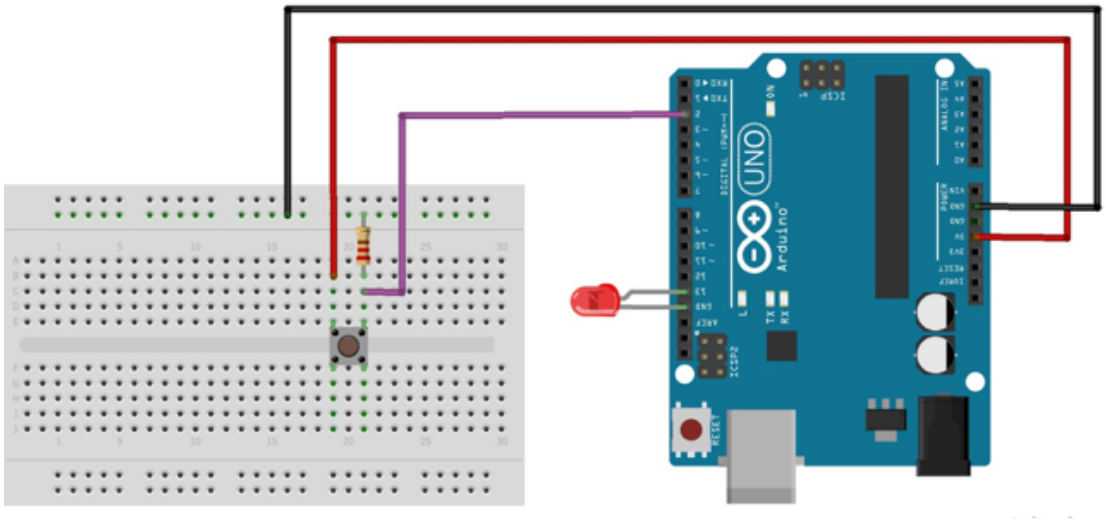
Learn how to connect a pushbutton and LED onto the breadboard and to the Arduino.

Used Blocks

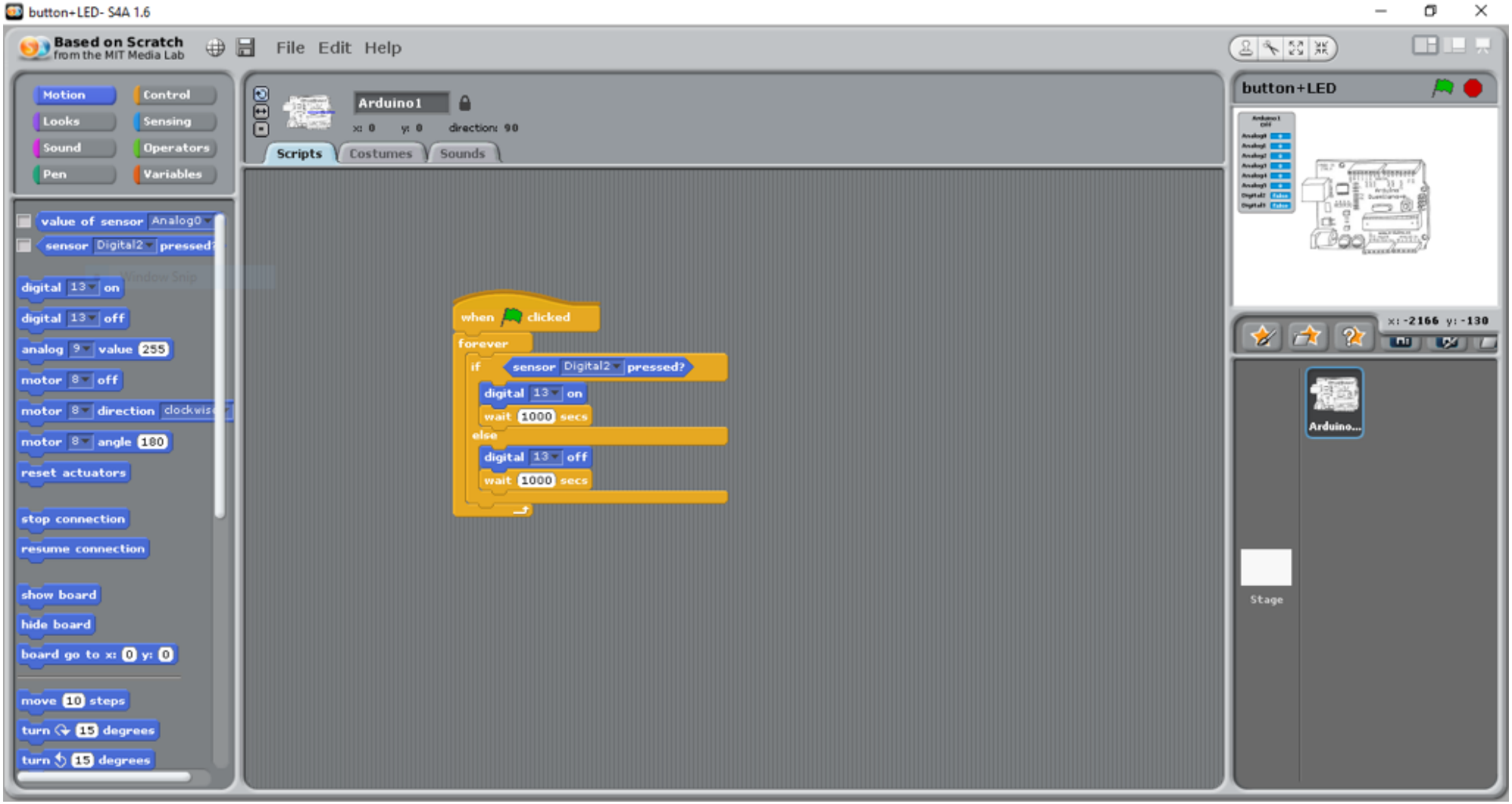
You Will Need

Breadboard
LED
Resistor (220Ω)

Connect as shown here

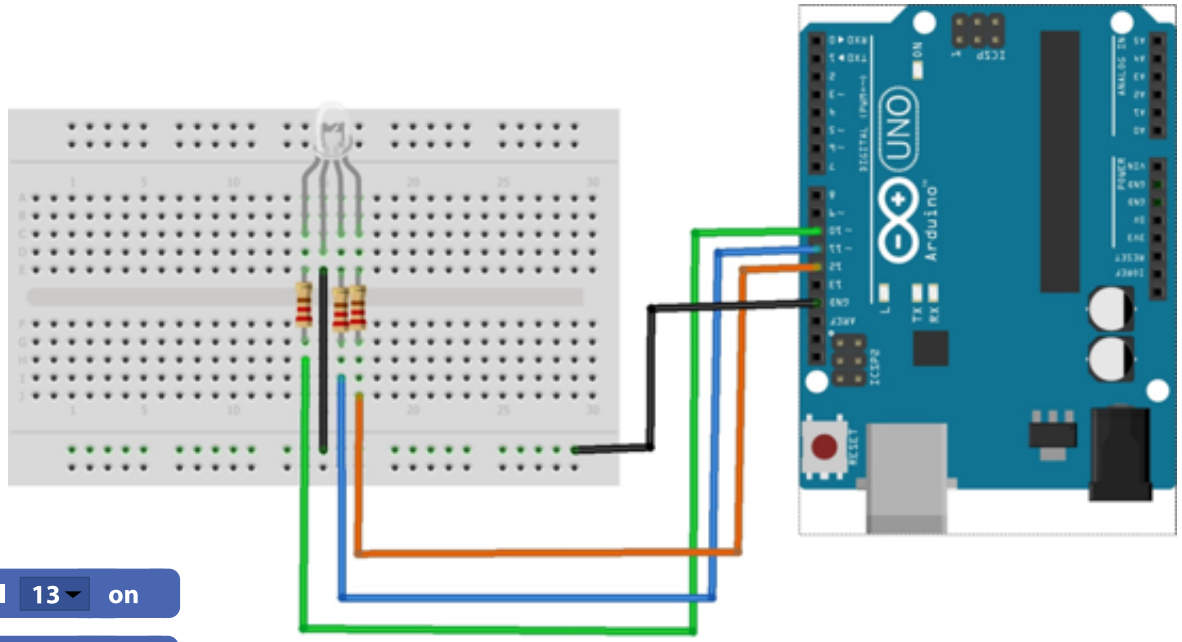


In this exercise we will mount a LED and control it by a button. The objective is turn on the LED on whenever the button are pressed, and turn it off when not pressed. Connect the Arduino , Button and LED as shown in the diagram above . Code the following code, drag and drop blocks in the scripting area as follows:

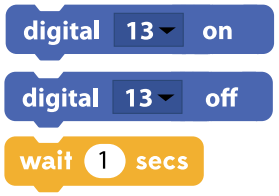

button+LED- S4A 1.6
Based on Scratch
from the MIT Media Lab
File Edit Help
Motion
Control
Looks
Sensing
Sound
Operators
Pen
Variables
value of sensor Analog0
sensor Digital2 pressed?
digital 13 on
digital 13 off
analog 9 value 255
motor 8 off
motor 8 direction clockwise
motor 8 angle 180
reset actuators
stop connection
resume connection
show board
hide board
board go to x: 0 y: 0
move 10 steps
turn 15 degrees
turn 15 degrees
Arduino1
x: 0 y: 0 direction: 90
Scripts
Costumes
Sounds
when clicked
forever
if sensor Digital2 pressed?
digital 13 on
wait 1000 secs
else
digital 13 off
wait 1000 secs
button+LED
x: -2166 y: -130
Arduino...
Stage

PROGRAM 4 : RGB LED

Level: Beginner

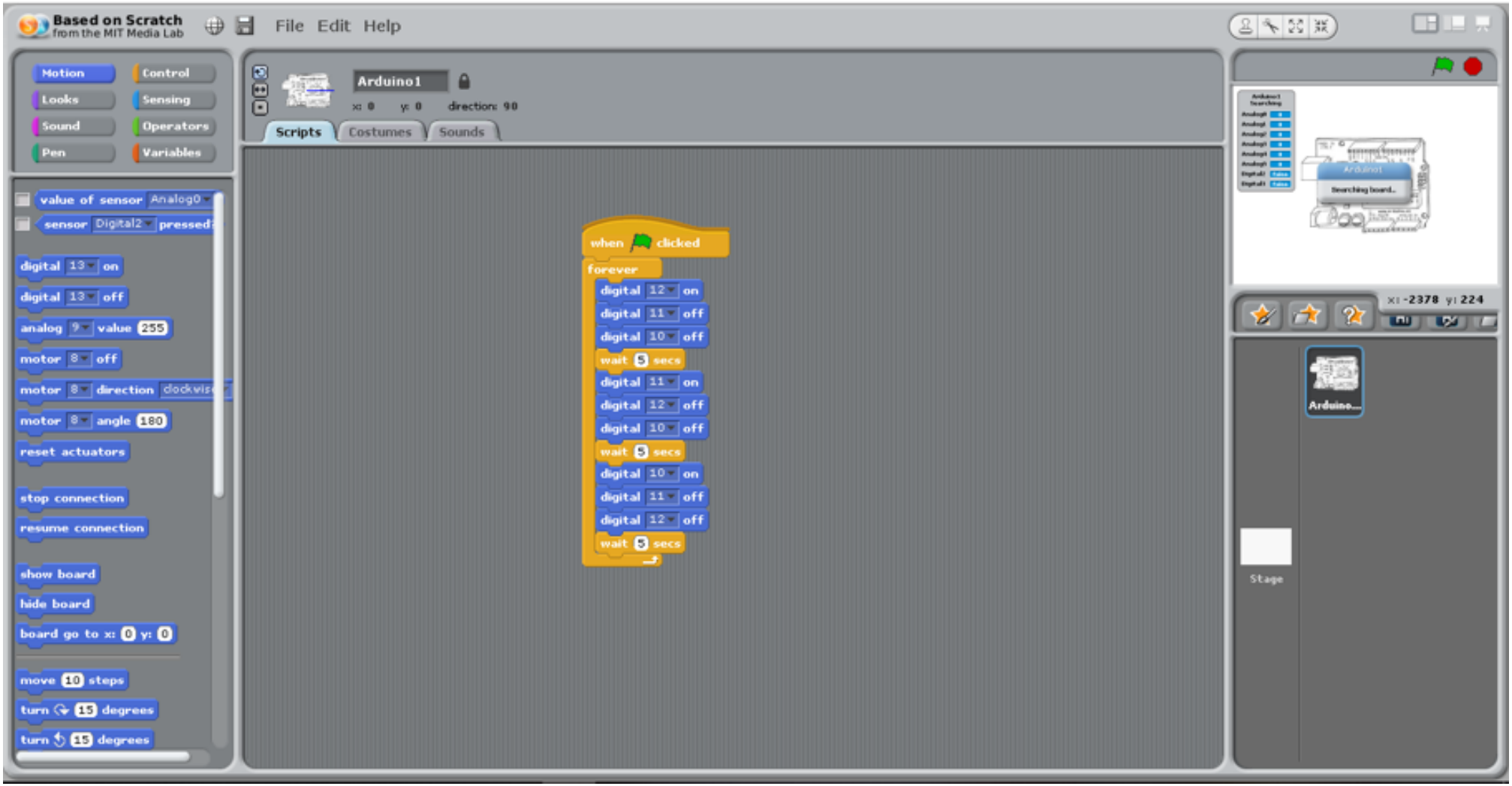


Used Blocks



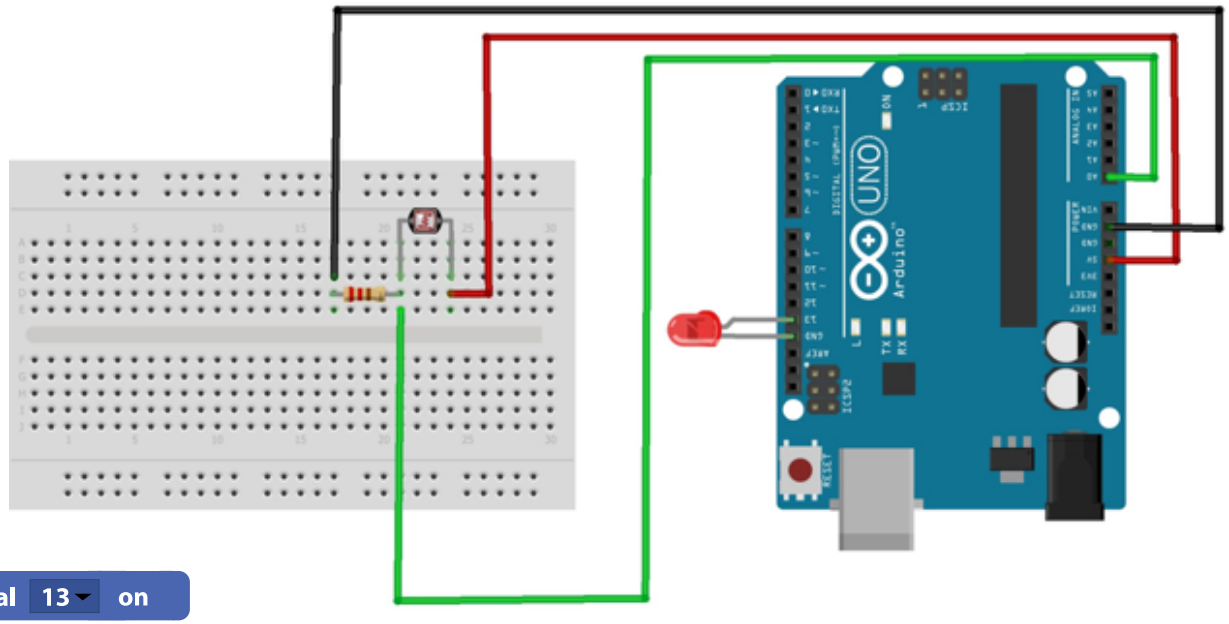
You Will Need

The RGB LEDs provide a full spectrum of color (in contrast to normal ones, which give only one color), resulting in a light composed of three primary colors. Depending on the intensity of each color, we can get a lot of different shades. It is very important to know which color controls each LED leg:


Based on Scratch
from the MIT Media Lab
File Edit Help
Motion
Control
Looks
Sensing
Sound
Operators
Pen
Variables
value of sensor Analog0
sensor Digital2 pressed
digital 13 on
digital 13 off
analog 9 value 255
motor 8 off
motor 8 direction
motor 8 angle 180
reset actuators
stop connection
resume connection
show board
hide board
board go to x: 0 y: 0
move 10 steps
turn 15 degrees
turn 15 degrees
Arduino1
x: 0 y: 0 direction: 90
Scripts
Costumes
Sounds
when clicked
forever
digital 12 on
digital 11 off
digital 10 off
wait 5 secs
digital 11 on
digital 12 off
digital 10 off
wait 5 secs
digital 10 on
digital 11 off
digital 12 off
wait 5 secs
x: -2378 y: 224
Arduino...
Stage

PROGRAM 5 : LDR + LED

Level: Beginner



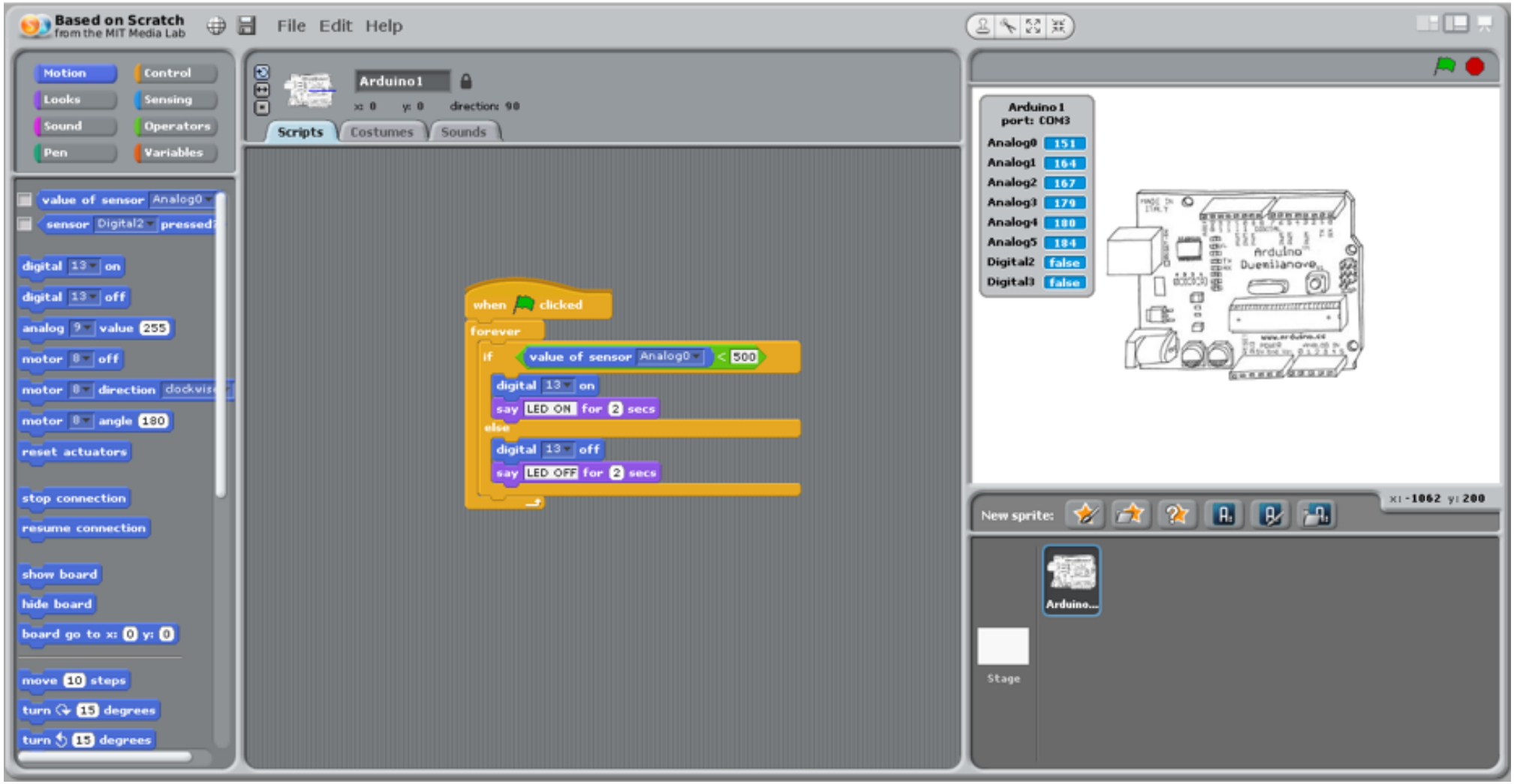
Used Blocks

You Will Need

- Breadboard
- LED
- Resistor (220Ω)
- LDR

Now we will replace the button by a LDR (Light Dependent Resistor). The objective is turn on the LED when the LDR is covered. Turn on a LED when LDR sensor is covered Automatic Night Light (I) .CdS - LDR (Cadmium Sulfide - Light Dependent Resistor) or photocell sensor .Its resistance is inversely dependent on the amount of light falling on it. As shown above

Code as shown below


Based on Scratch
from the MIT Media Lab
File Edit Help
Motion
Control
Looks
Sensing
Sound
Operators
Pen
Variables
value of sensor Analog0
sensor Digital2 pressed
digital 13 on
digital 13 off
analog 9 value 255
motor 0 off
motor 0 direction
motor 0 angle 180
reset actuators
stop connection
resume connection
show board
hide board
board go to x: 0 y: 0
move 10 steps
turn 15 degrees
turn 15 degrees
Arduino1
x: 0 y: 0 direction: 90
Scripts
Costumes
Sounds
when clicked
forever
if value of sensor Analog0 < 500
digital 13 on
say LED ON for 2 secs
else
digital 13 off
say LED OFF for 2 secs
Arduino 1
port: COM3
Analog0 151
Analog1 164
Analog2 167
Analog3 179
Analog4 180
Analog5 184
Digital2 false
Digital3 false
Arduino
Duemilanove
New sprite:
x: -1062 y: 200
Arduino...
Stage

MORE PROGRAMS

We release more programs online regularly. Subscribe to our mailing list on http://purpletrust.org to get more exciting and challenging projects.

Program levels available

Beginner

Intermediate

Advanced

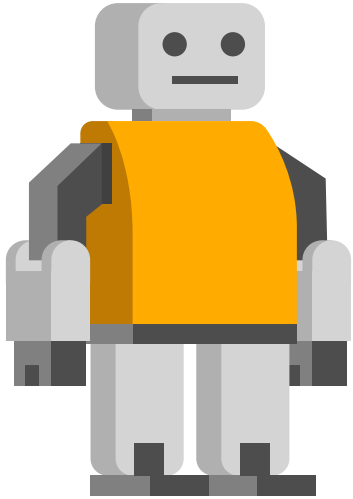
Projects

Four Reasons to get Serious About Online Training

Subject matter experts (SMEs) are usually not trainers. Subject matter experts are often called upon to create training content or contribute to capacity building project components. This is a classic challenge in development. But if SMEs can build skills in online training and facilitation and that can make a huge difference for your project.

Source: TechChange

The Journey to Self-Reliance is paved in peer-to-peer learning.



With peer-to-peer learning

- Students receive more time for individualized learning.
- Direct interaction between students promotes active learning.
- Peer teachers reinforce their own learning by instructing others.
- Students feel more comfortable and open when interacting with a peer.
- Peers and students share a similar discourse, allowing for greater understanding.
- Peer teaching is a financially efficient alternative to hiring more staff members.
- Teachers receive more time to focus on the next lesson.

Source: OpenColleges

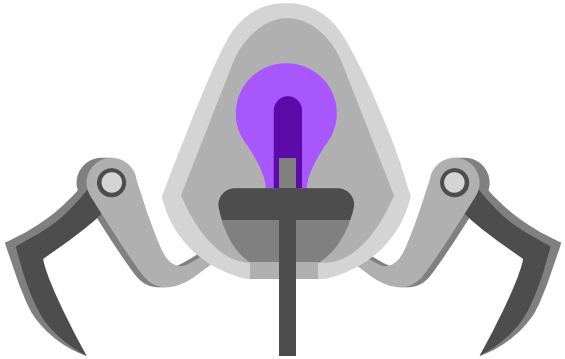
Educational tools and technologies change fast.
The edtech landscape changes fast. We 'll make sure you 're up to date with these changes.

SAFETY PRECAUTIONS

General Safety

Before working on any electronics, consider following these basic safety precautions to help reduce any hazards.

- Remove any electronic equipment you are testing or working on from the power source.
- Never assume the power circuit is off. Test and test again (with a voltmeter for example) to confirm.
- Remove fuses and replace them only after the power to the circuit is disconnected.
- Do not connect power to a circuit until you are done working on it and rechecked the work.
- Always ensure that all electronics equipment is properly grounded.
- If it is damaged, replace it. For instance, replace cables instead of repairing with insulating tape.
- Always use the right electronics repair and maintenance tools.
- Always return covers after removing them to reduce the risk of electric shock.
- Make sure your circuit is not overloaded.
- Always have safety equipment like a fire extinguisher, a basic first aid kit and a mobile phone nearby.



Personal Safety

It is important to ensure that you are safe when working on electronic circuits. Here are some personal safety precautions to keep in mind:

- Always keep your work area dry.
- Always work in a well-ventilated area.
- Do not wear flapping or loose clothing when working.
- Do not work with metallic jewellery on your hands like watches, rings and bracelets.
- Do not use bare hands to remove hot parts.
- Always wear non-conductive shoes.
- Always wear insulator gloves in your hands when carrying out repairs.
- Always remove power to a circuit before connecting alligator clips.
- Always wear safety goggles.
- Be careful when handling large capacitors as they can still hold high voltage even after you have disconnected the circuit from power.

SAFETY PRECAUTIONS cont'd

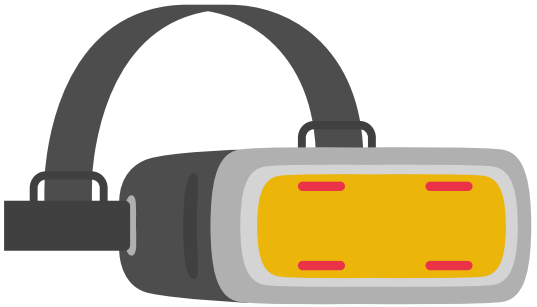
Fire Safety Precautions

When working with electronic equipment, there is often a risk of fire caused by a short circuit or other reason. Follow these precautionary steps:

- Avoid anything that would cause a fire around your working area like paper, cloth or other combustible materials.
- Look out for damaged wire insulation, overheating of electronic equipment, damaged circuit boards and corrosive components like batteries.
- If there is a burning smell on your electronic equipment, disconnect the power source.
- If there is a fire, use a non-conducting dry powder or CO_2 fire extinguisher.
- Always check your circuit to be sure that everything is okay after repairs or maintenance before connecting power.

Electric Shock

One of the major hazards when working with electronic equipment is electric shock. To avoid this, you should take a few safety precautions, including:



- Always read safety procedures that come with every electronic equipment you 're about to test or work on.
- Recheck all wires for bad connections
- Always make sure that all parts of electronic equipment are well-mounted to prevent accidents.
- Keep electronic equipment away from water and other liquids
- Always check for signs of wear, defects and fraying on electronic equipment cables, cords and connectors.
- Use special safety rubber gloves and shoes.

Testing Equipment

With the increasing use of electronics in homes and workplaces, safety is becoming more and more important to consumers and service experts. It may be important to invest in the right testing equipment for your work that ensures that you are able to carry out your job safely.

Contact us on info@purpletrust.org if you are interested in ordering some testing equipment.

Troubleshooting (Advanced)

Get help from your parent.

Installing the Firmware into your Arduino.

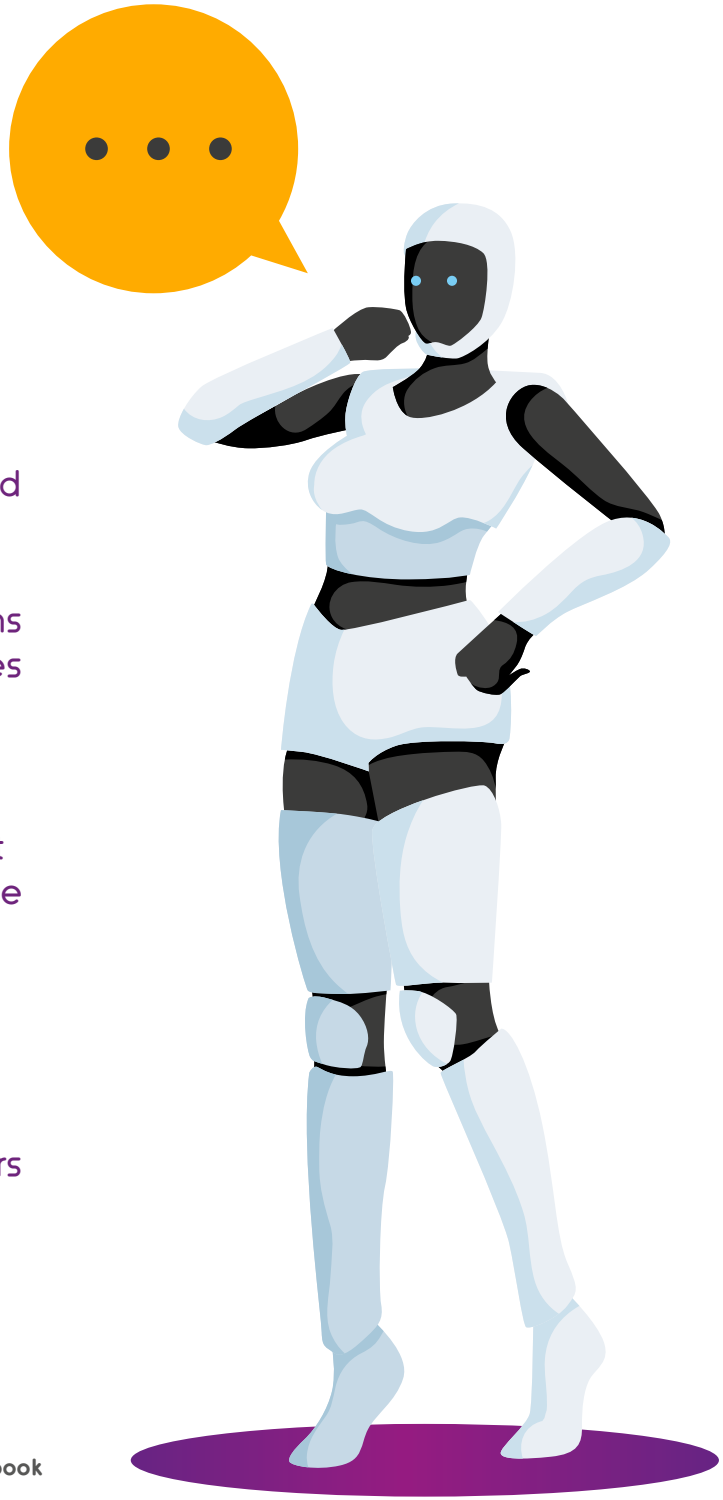
If S4A is unable to ‘ talk ’ to your Arduino.

This firmware is a piece of software you need to install into your Arduino board to be able to communicate with it from S4A.

- Download and install the Arduino environment by following the instructions on http://arduino.cc/en/Main/Software. Take in account Arduino Uno requires at least version 0022.
- Download firmware from here
- Connect your Arduino board to a USB port in your computer
- Open the firmware file (S4AFirmware16.ino) from the Arduino environment
- In the Tools menu, select the board version and the serial port where the board is connected
- Load the firmware into your board through File > Upload

Arduino drivers

If you are a Microsoft Windows user, you may need to install the Arduino drivers into your computer:
Arduino drivers for Microsoft Windows



PEEK
PURPLE EARLY ENGINEERING KIT

Purple Future Trust Purple Early Engineering Kit (PEEK) Handbook

CONTACT

1 Adylinn Road, Harare, Zimbabwe
+263 77 258 4350
info@purpletrust.org