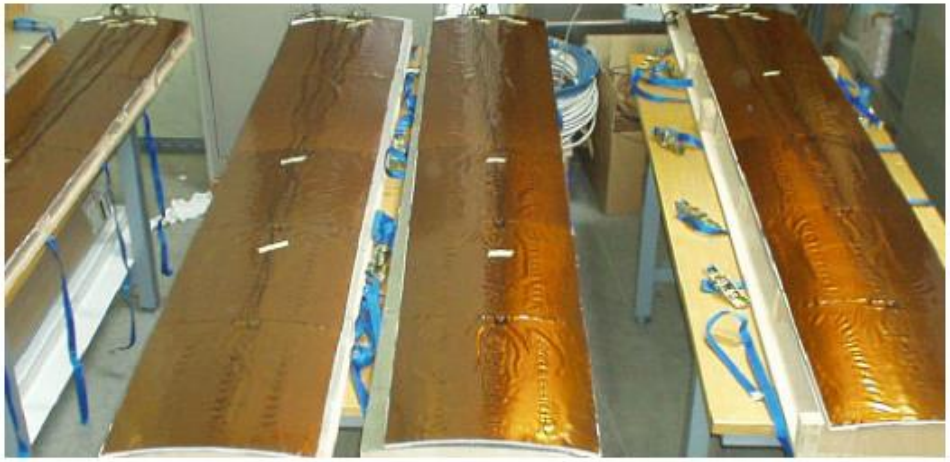
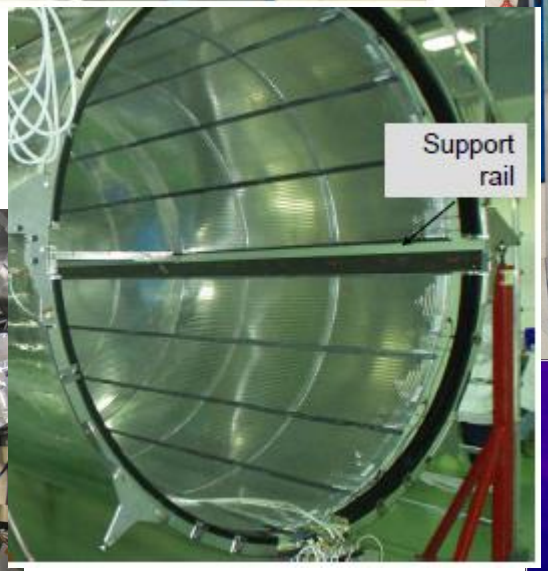
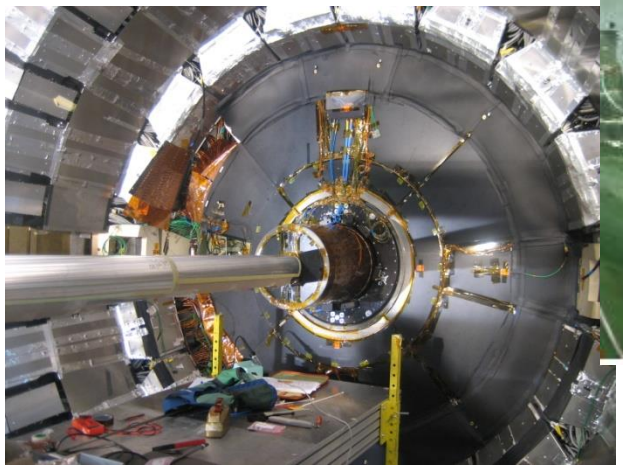
Εισαγωγη στους ανιχνευτες σωματιδιων στο CERN

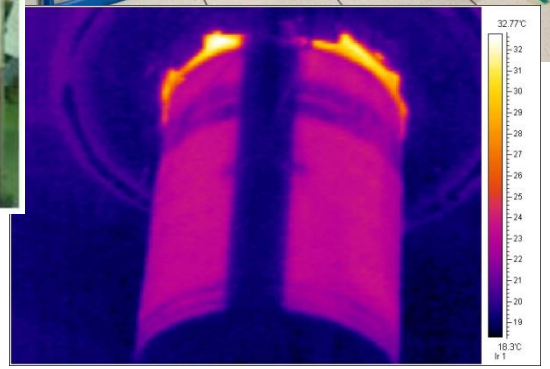
...και ισως μερικες πιθανες ιδεες για τους μαθητες σας

Συστηματα εξυπνων μονωτων(~ -20^{0}C καταγραφεας τροχιων ~ +17^{0}C ECAL)

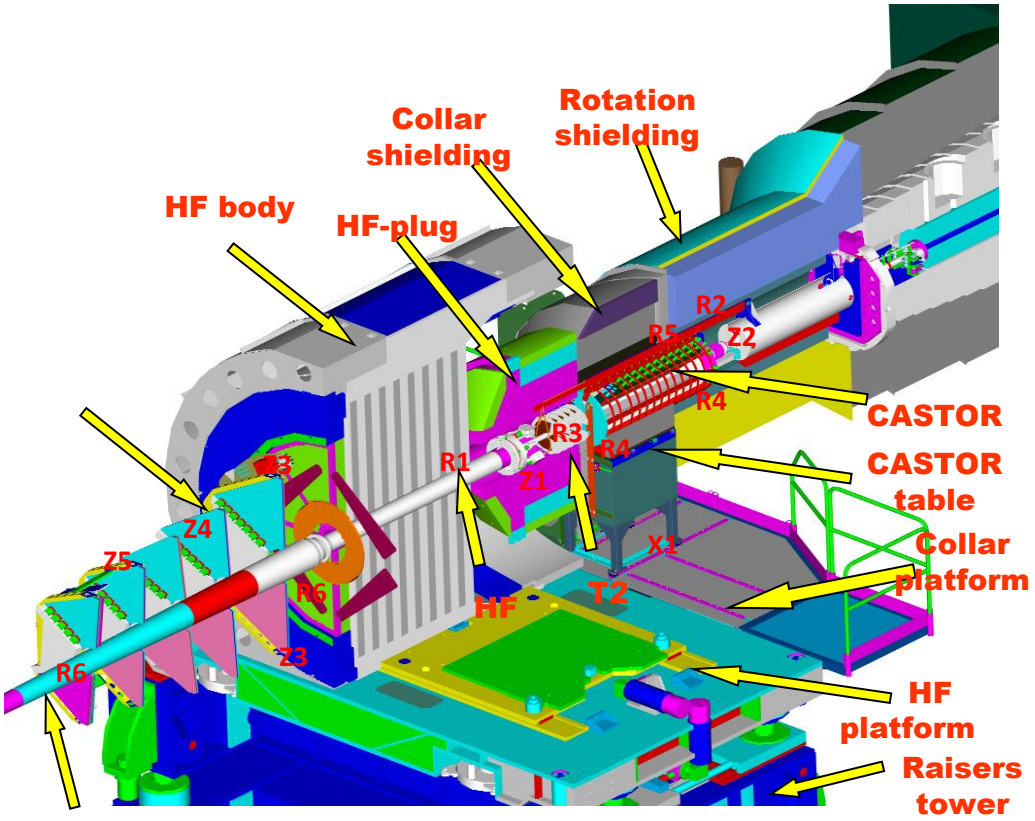




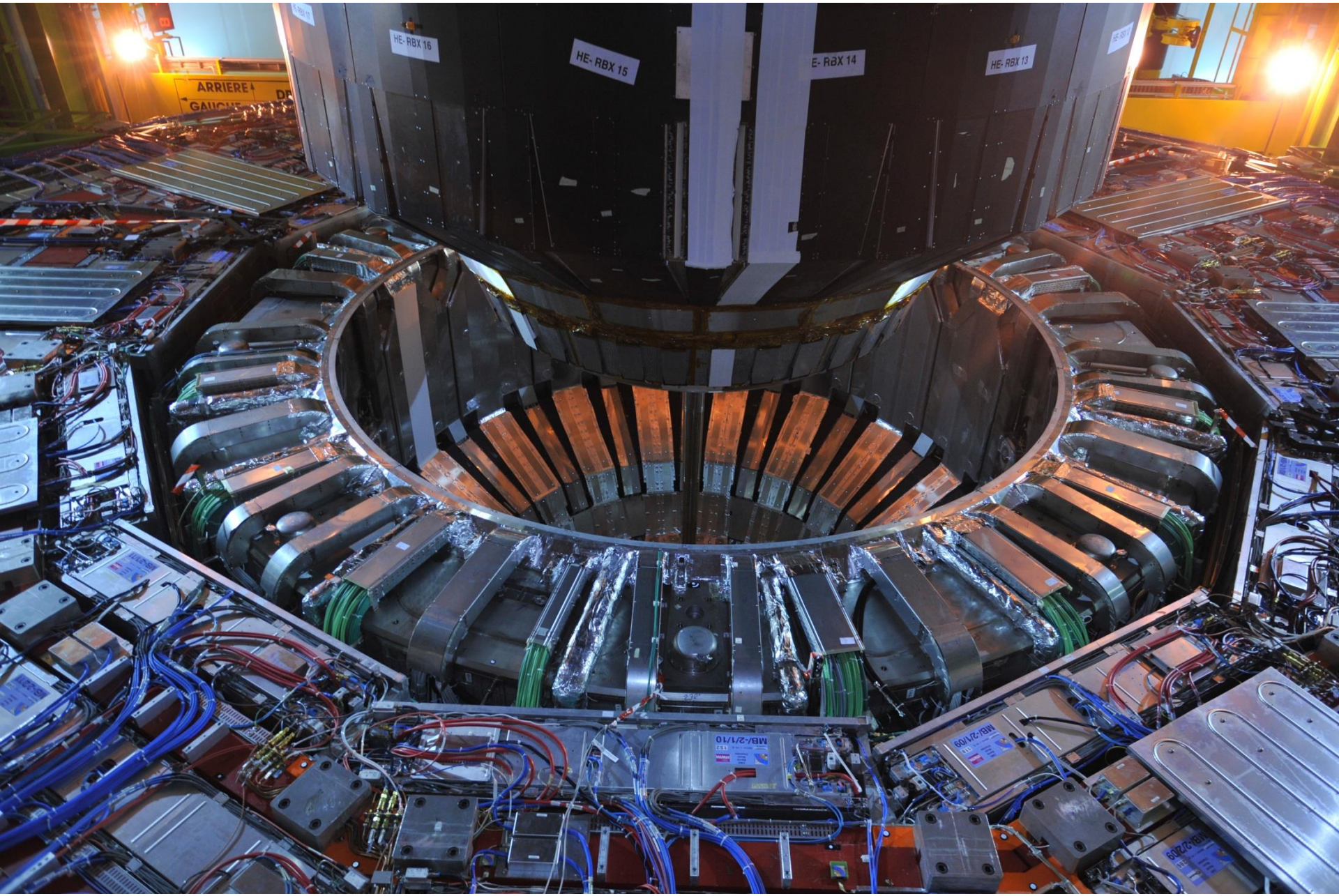




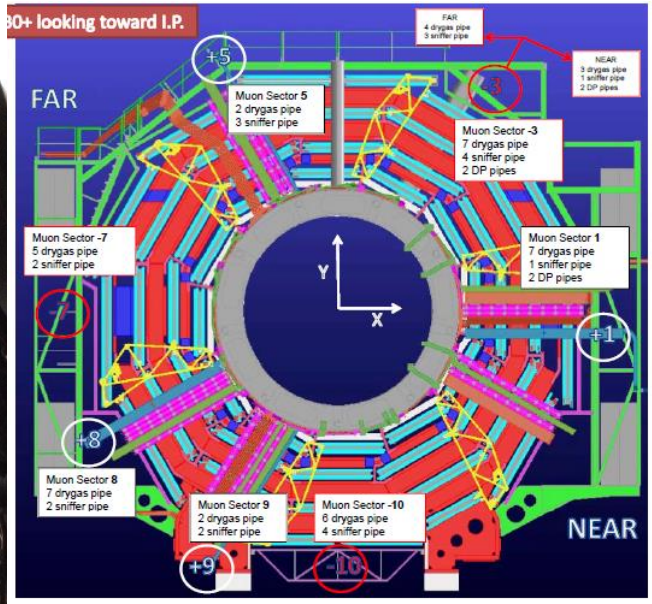
•Συστηματα ελεγχου κινησεων αντικειμενων (υποανιχνευτες και αλλα κοματια “ζυγιζουν” μεχρι 2 τοννους, αισθητηρες)



Τα βαρη των κομματιων μπορουν να αγγιξουν τους 2 τονους και κινουνται με υδραυλικα συστηματα. Ολα αυτα αυτοματα και απο μακρια...μα το προβλημα του να βρεθουν η να αναπτυχθουν αισθητηρες καταλληλοι για τις συνθηκες των πειραματων παραμενει


HE- RBX 16
HE- RBX 15
HE- RBX 14
HE- RBX 13
ARRIERE

•Συστηματα ελεγχου περιβαλλοντος (ξερος αερας, αζωτο, θερμοκρασια)



Κοστος ~150,000 ευρω

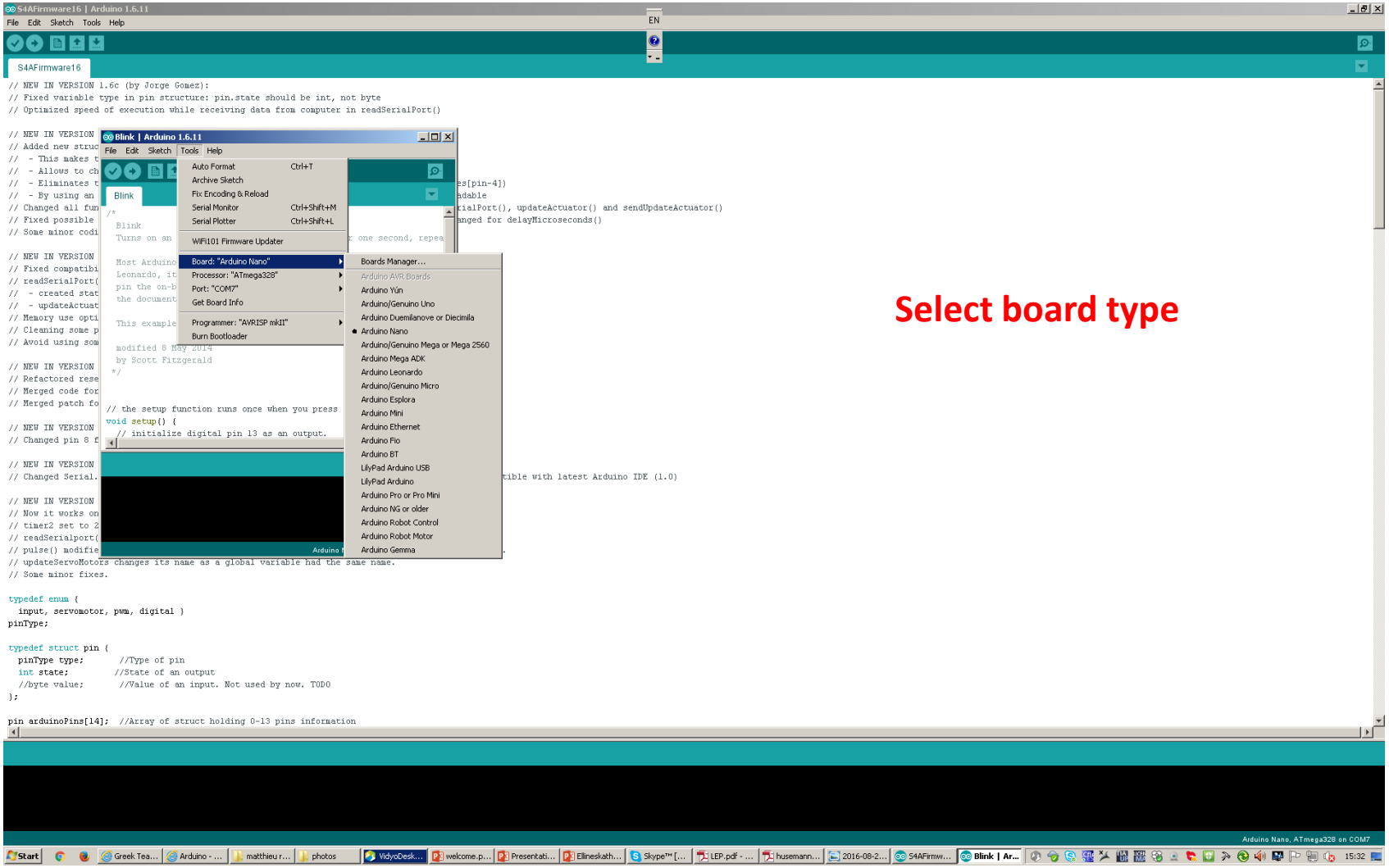
Προστασια του ανιχνευτη τροχιων απο προβλημα υγρασιας η μαλλον σημειου δροσου

We would like to work with you on possible projects that will teach programming to students, (scratch https://scratch.mit.edu/) in the context of learning to measure and control numbers; programming Provided it is done in a way accessible to their age teaches them the logic and the discipline of going about a measurement Scratch is used for programming games, playing with photos, putting together music, etc. I am sure that in this context some of You might have even used it
We would like to deal with the "Scratch for Arduino-S4A" version because it gives you the option of combining simple and visual oriented programming with the Arduino platform that makes electronics easy and students can make things happen.

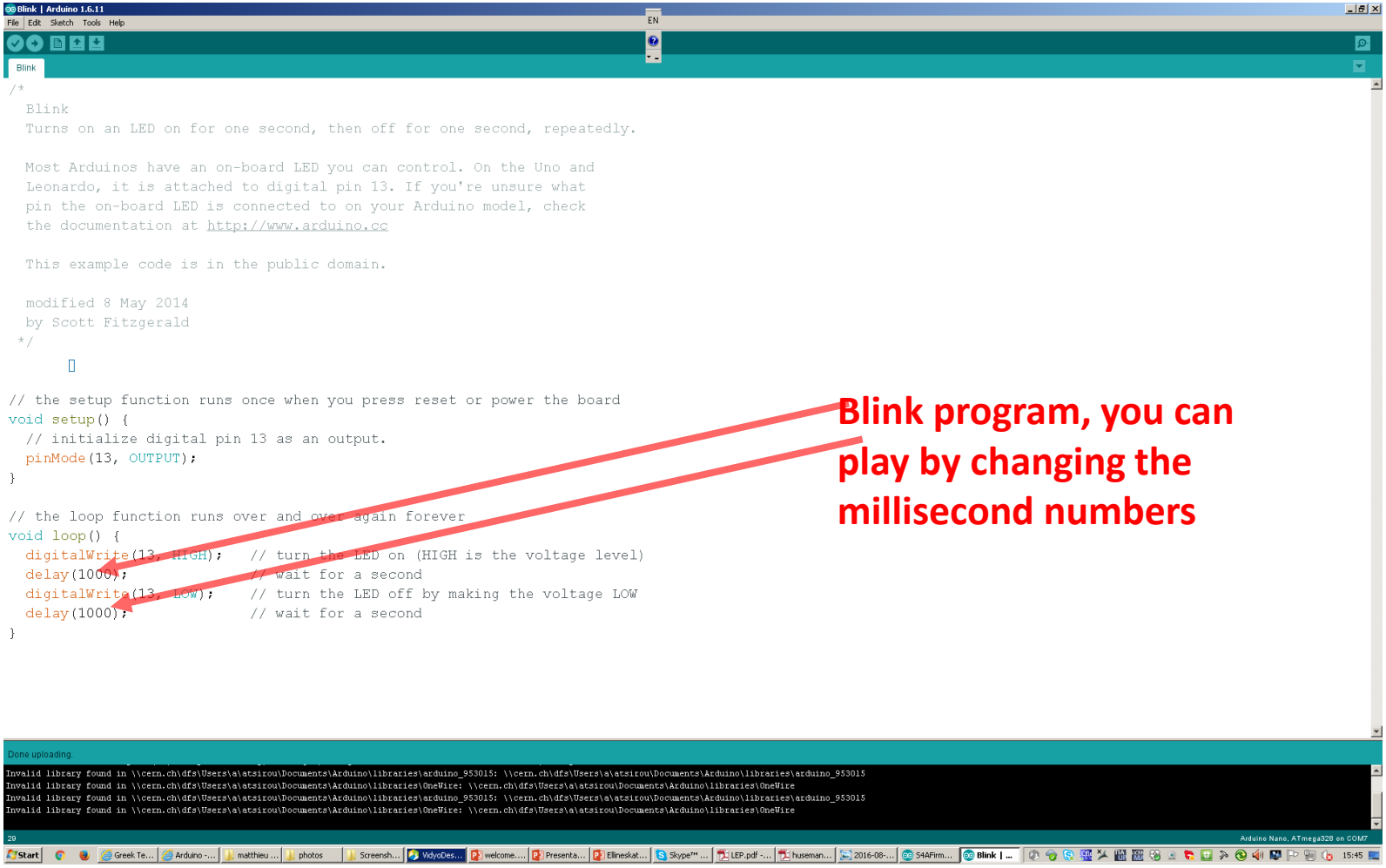
All information about the Arduino platform can be accessed at
https://www.arduino.cc/
From this page you can download the appropriate software, be it Windows, Linux or Mac OS X (Mac OSX Lion or later). You will not deal with this software as you will not be an Arduino developer (or you can do it with some interested students but then you will use C or other "adult" Programming languages…!)

However, you have to use the Arduino software to tell your system WHAT Arduino Platform you are using:…you could start Arduino, go to the menu, select Tools and from there select Boards and click on the board type you have connected (Uno, Mega, nano, Duemilenove, Leonardo,etc)…

Then , to be sure that you are connected to the board, try to run a program called "Blink" that makes an LED already built on your Arduino to "blink" by going to "Menu", "File", "Examples", "Basics", "Blink"..

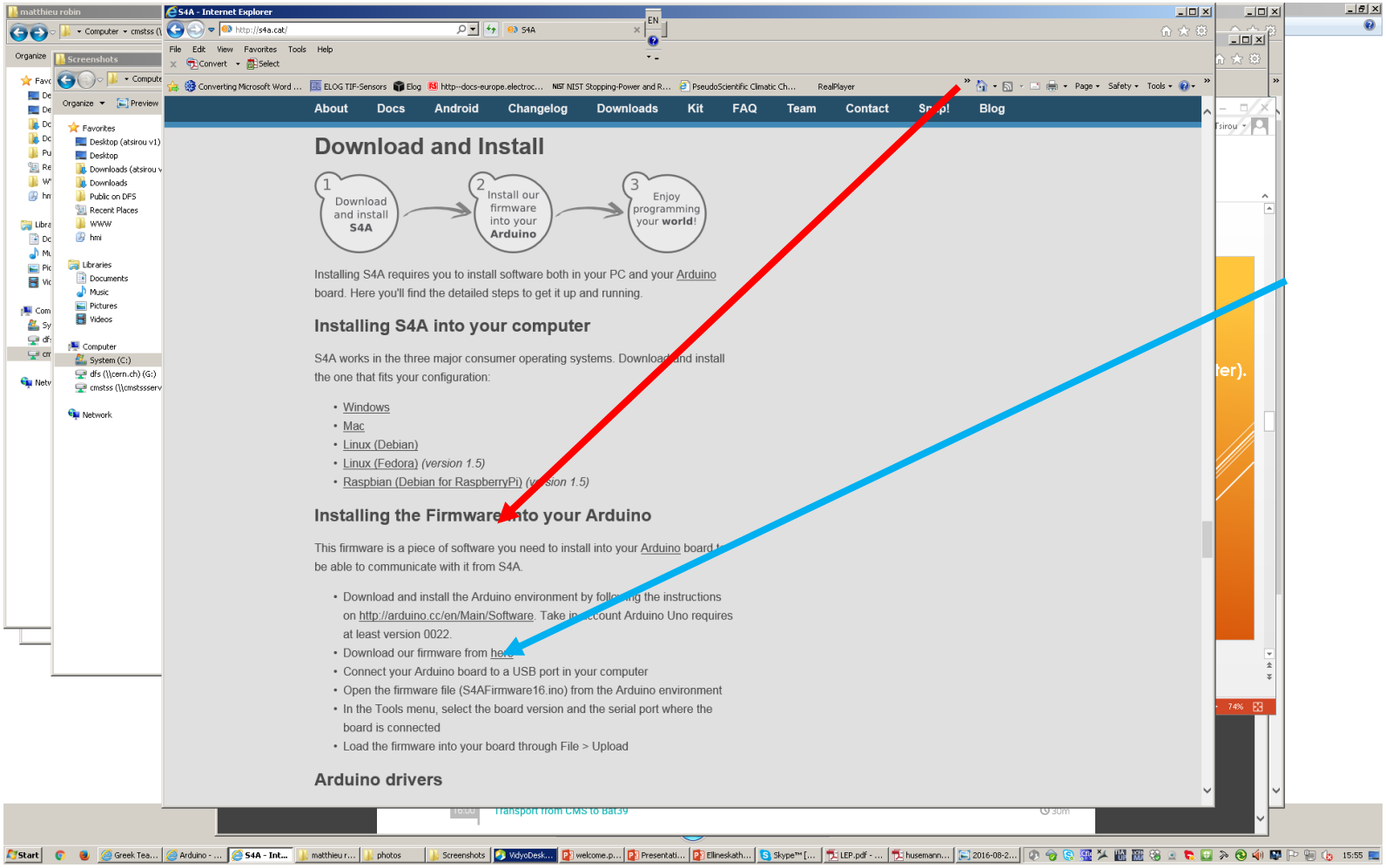


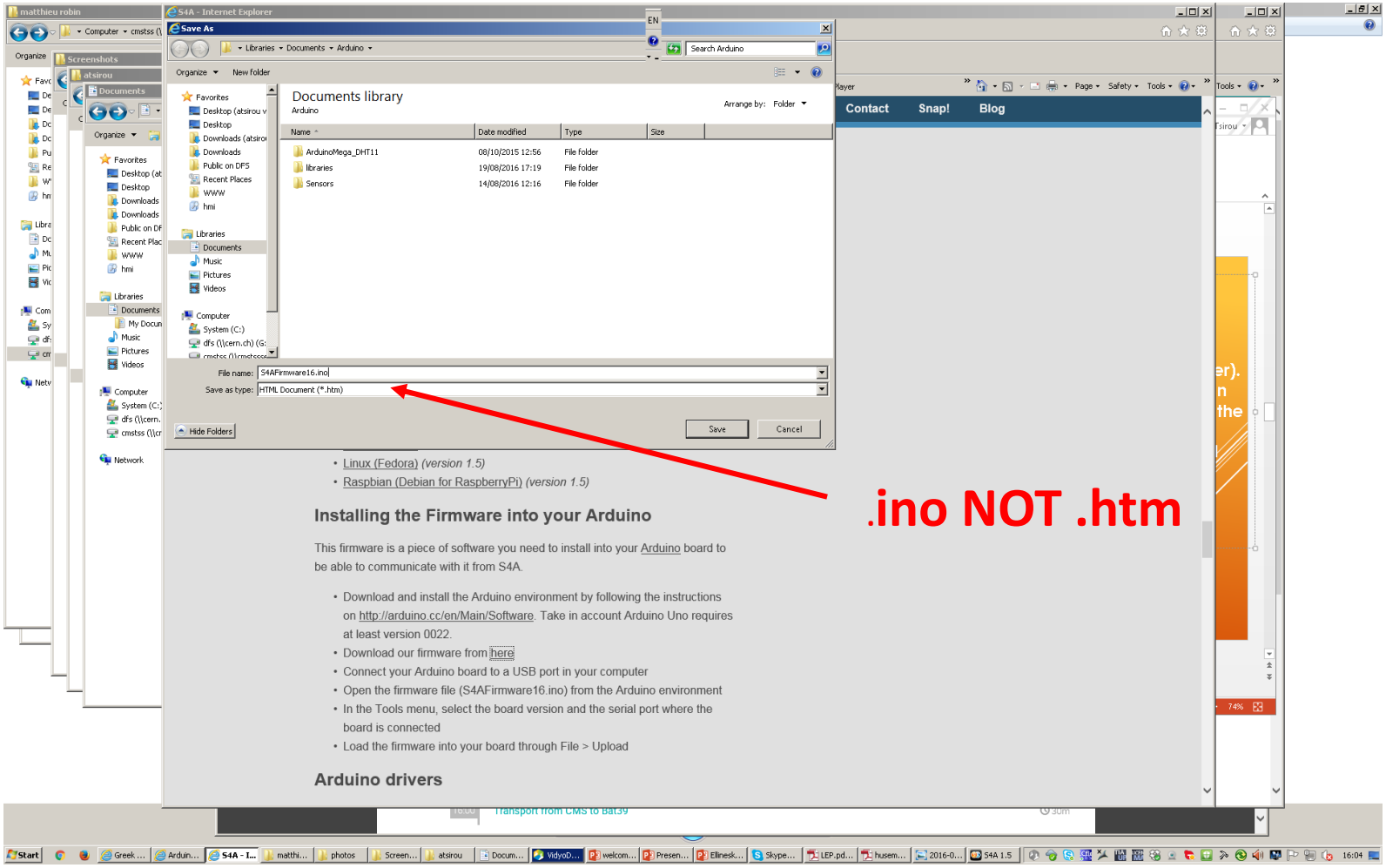
Select board type


Blink | Arduino 1.6.11
File Edit Sketch Tools Help
Blink
/*
Blink
Turns on an LED on for one second, then off for one second, repeatedly.
Most Arduinos have an on-board LED you can control. On the Uno and
Leonardo, it is attached to digital pin 13. If you're unsure what
pin the on-board LED is connected to on your Arduino model, check
the documentation at http://www.arduino.cc
This example code is in the public domain.
modified 8 May 2014
by Scott Fitzgerald
*/
// the setup function runs once when you press reset or power the board
void setup() {
// initialize digital pin 13 as an output.
pinMode(13, OUTPUT);
}
// the loop function runs over and over again forever
void loop() {
digitalWrite(13, HIGH); // turn the LED on (HIGH is the voltage level)
delay(1000); // wait for a second
digitalWrite(13, LOW); // turn the LED off by making the voltage LOW
delay(1000); // wait for a second
}
Blink program, you can play by changing the millisecond numbers
Done uploading.
Invalid library found in \\cern.ch\dfs\Users\a\atsirou\Documents\Arduino\libraries\arduino_953015: \\cern.ch\dfs\Users\a\atsirou\Documents\Arduino\libraries\arduino_953015
Invalid library found in \\cern.ch\dfs\Users\a\atsirou\Documents\Arduino\libraries\OneWire: \\cern.ch\dfs\Users\a\atsirou\Documents\Arduino\libraries\OneWire
Invalid library found in \\cern.ch\dfs\Users\a\atsirou\Documents\Arduino\libraries\arduino_953015: \\cern.ch\dfs\Users\a\atsirou\Documents\Arduino\libraries\arduino_953015
Invalid library found in \\cern.ch\dfs\Users\a\atsirou\Documents\Arduino\libraries\OneWire: \\cern.ch\dfs\Users\a\atsirou\Documents\Arduino\libraries\OneWire
29
Arduino Nano, ATmega328 on COM7
Start
15:45

All i**nformation about the Arduino platform for the Scratch environment can be accessed at http://s4a.cat/**

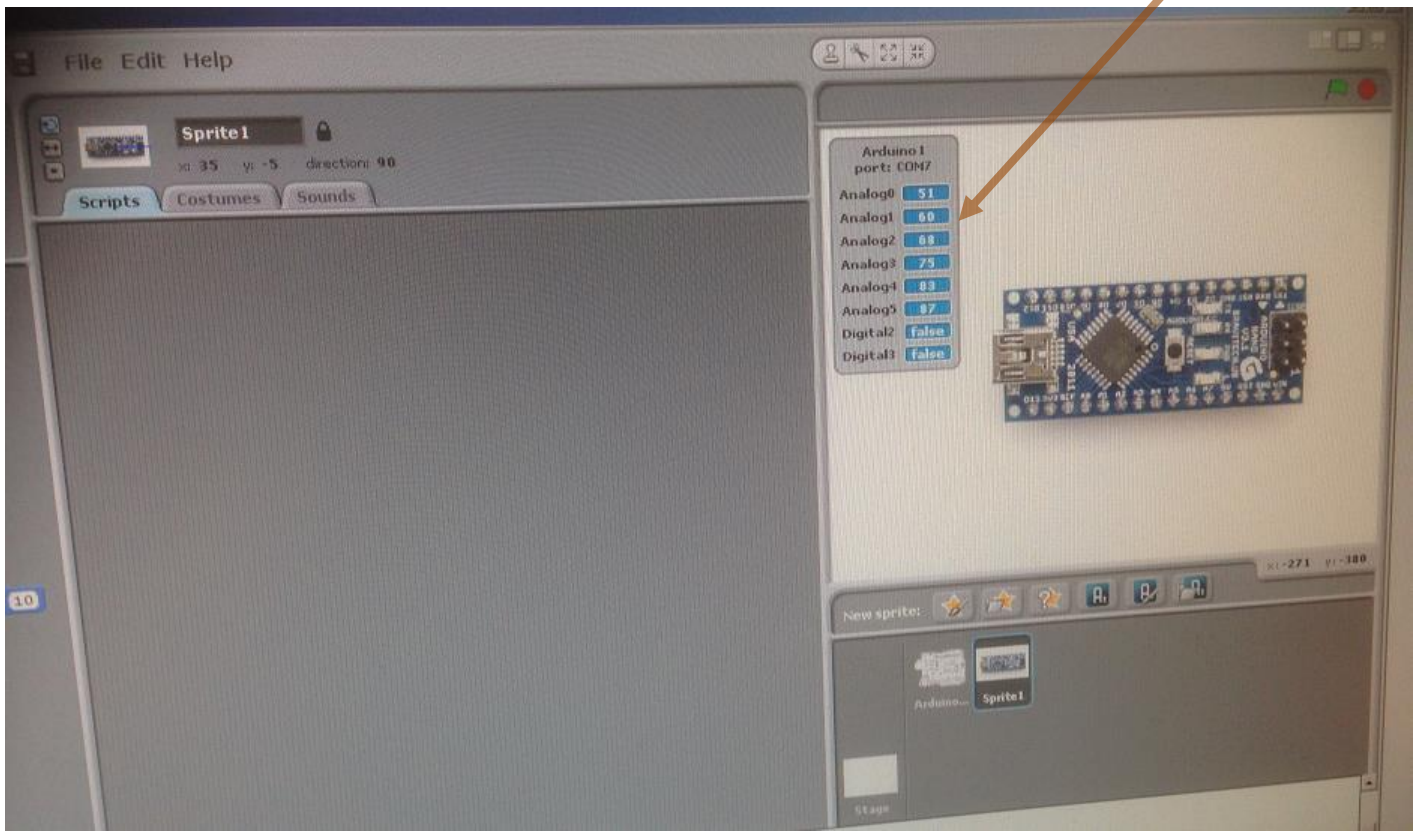
From this page (Downloads) you can download the appropriate software, be it for Windows, Linux or Mac OS X (Mac OSX Lion or later). Make sure that you also get the program that you have to execute in order to "tell" your Arduino platform that it has to work with Scratch; the program is found in "Installing the Firmware into your Arduino", right-click on "here" and save target as a .ino and NOT a .htm file as proposed in your Documents -> Arduino and then run it! You have to do this every time you want to use Scratch for your Arduino (s)


S4A - Internet Explorer
About
Docs
Android
Changelog
Downloads
Kit
FAQ
Team
Contact
Blog
Download and Install
1 Download and install S4A
2 Install our firmware into your Arduino
3 Enjoy programming your world!
Installing S4A requires you to install software both in your PC and your Arduino board. Here you'll find the detailed steps to get it up and running.
Installing S4A into your computer
S4A works in the three major consumer operating systems. Download and install the one that fits your configuration:
Windows
Mac
Linux (Debian)
Linux (Fedora) (version 1.5)
Raspbian (Debian for RaspberryPi) (version 1.5)
Installing the Firmware into 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 into account Arduino Uno requires at least version 0022.
Download our 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


Save As
Documents library
Arduino
Arrange by: Folder
Name
Date modified
Type
Size
ArduinoMega_DHT11
08/10/2015 12:56
File folder
libraries
19/08/2016 17:19
File folder
Sensors
14/08/2016 12:16
File folder
File name: S4AFirmware16.ino
Save as type: HTML Document (*.htm)
Save
Cancel
Contact
Snap!
Blog
.ino NOT .htm
Linux (Fedora) (version 1.5)
Raspbian (Debian for RaspberryPi) (version 1.5)
Installing the Firmware into 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 our 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

When you first “read” the Arduino via Scratch, this is what it looks like:

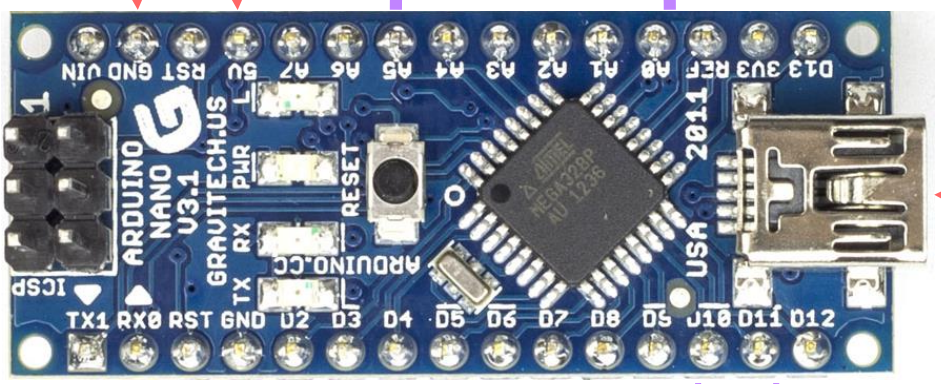
Values that “change”



5V power. For (almost) every sensor

GROUND! Your reference!

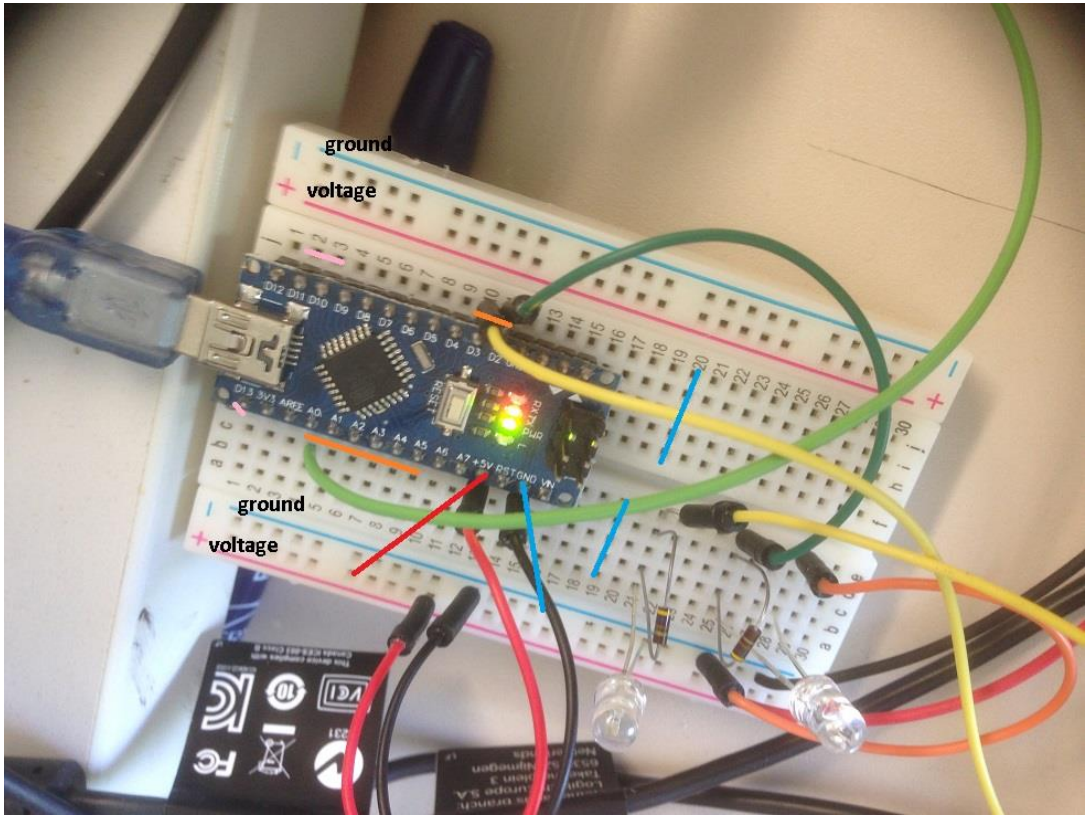
A0-A5 Read analog values in Scratch



D2-D3 Read digital inputs in Scratch

D10-D11-D13 “Drive” digital outputs in Scratch

USB connection with your PC

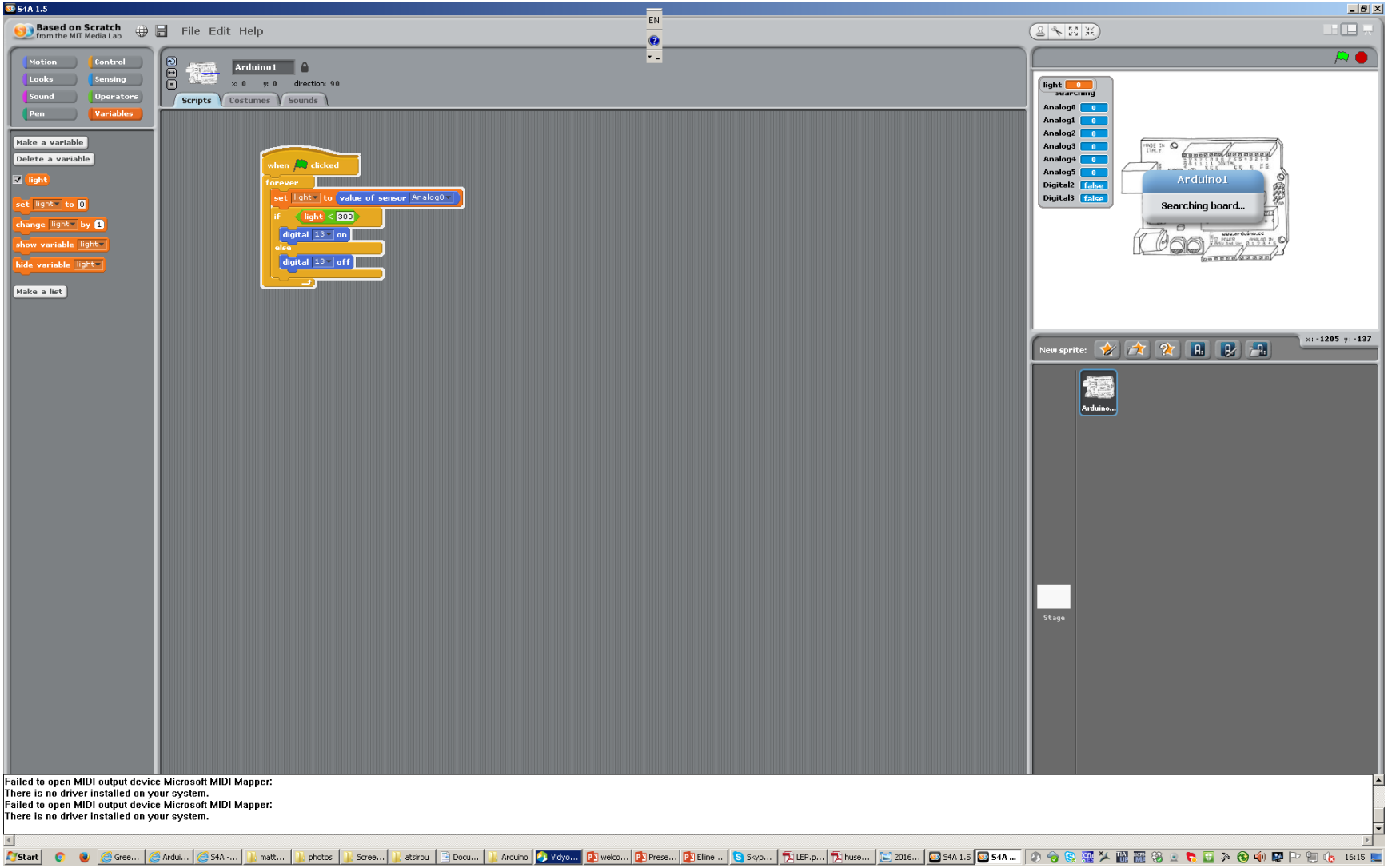

ground
voltage
ground
voltage

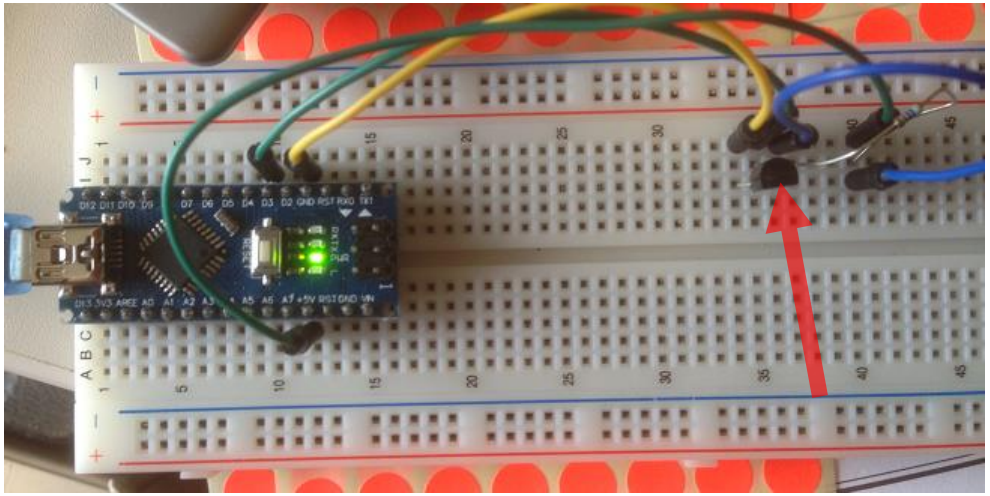
There are many types and models of Arduino.
We will be trying to concentrate using the nano Arduino and Uno which we will use for all the projects we propose to you. Using the Arduino in the Scratch environment you can read “analog” sensors and digital “sensors”.
Sensors are a huge industrial/scientific explosion of our times and CERN is a remarkable sensor consumer/producer.
Sensors provide information about everything and can be read in an analog way like current, voltage, resistance and many others (favour these for learning) or they can be digital and have an integrated microprocessor that gives you directly the numerical measured value.

The analog sensors you can read in Scratch are attached to your Arduino platform (literally)

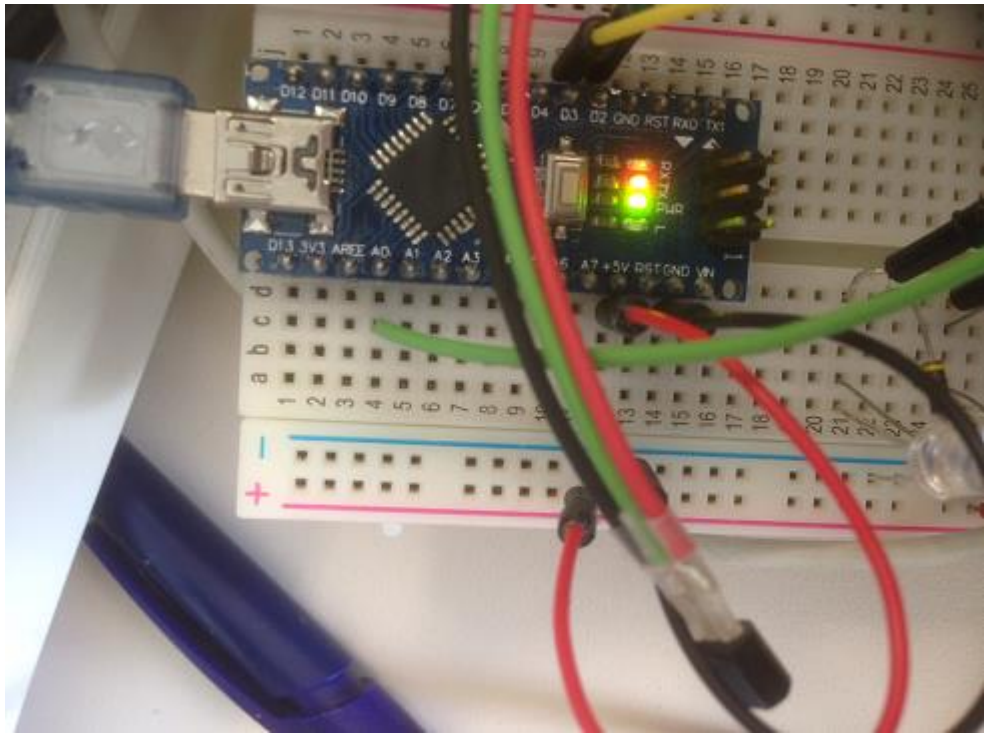
- **Thermometers**
- **Light level (photo resistors)**
- **Humidity**
- **Flow**
- **Pressure**
- **and many others ….**

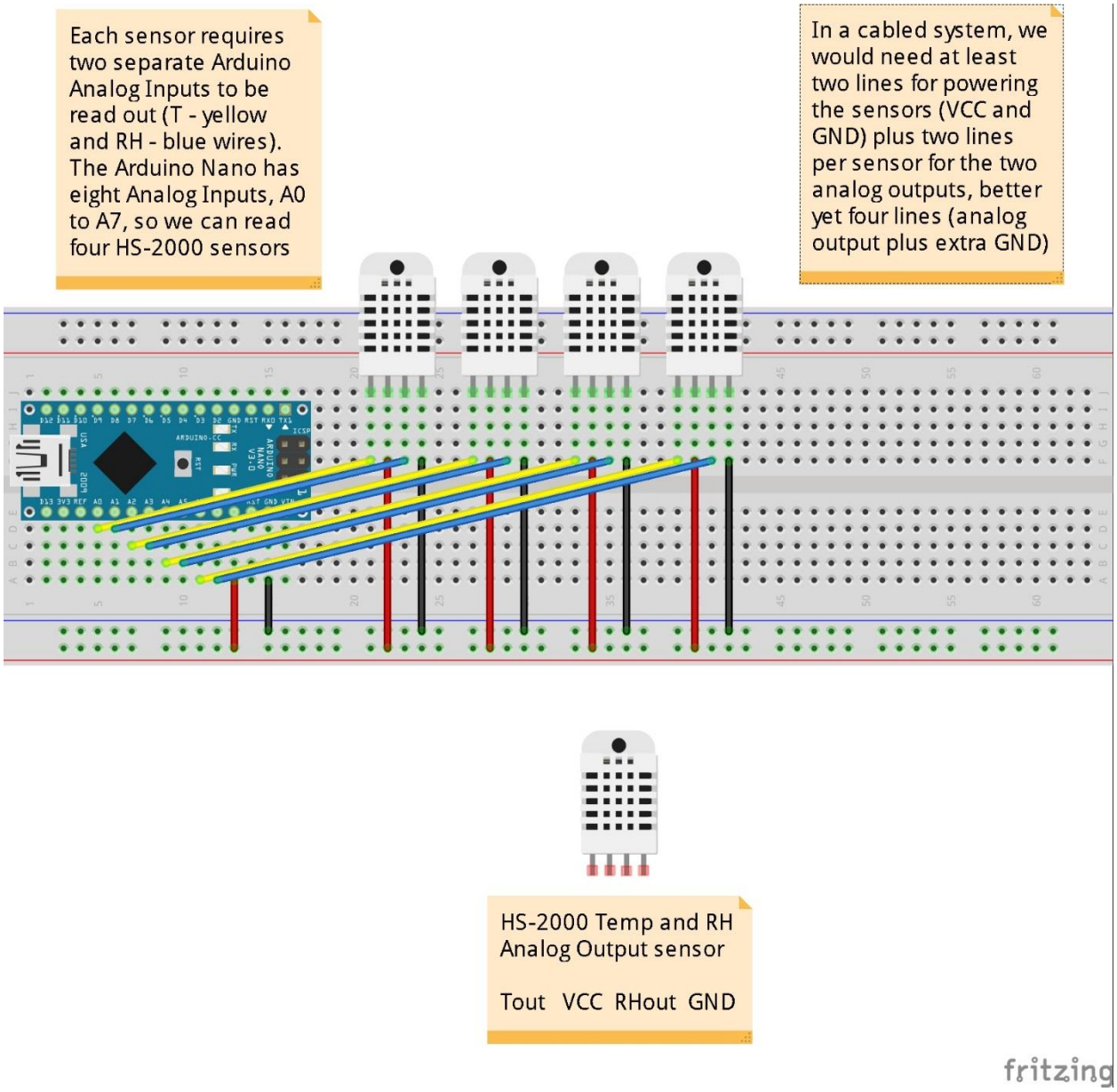
the numerical measured value.


S4A 1.5
Based on Scratch
from the MIT Media Lab
File Edit Help
EN
Motion
Control
Looks
Sensing
Sound
Operators
Pen
Variables
Make a variable
Delete a variable
light
set light to 0
change light by 1
show variable light
hide variable light
Make a list
Arduino1
x: 0 y: 0 direction: 90
Scripts
Costumes
Sounds
when clicked
forever
set light to value of sensor Analog0
if light < 300
digital 13 on
else
digital 13 off
light 0
Analog0 0
Analog1 0
Analog2 0
Analog3 0
Analog4 0
Analog5 0
Digital2 false
Digital3 false
Arduino1
Searching board...
New sprite:
x: -1205 y: -137
Arduino...
Stage
Failed to open MIDI output device Microsoft MIDI Mapper:
There is no driver installed on your system.
Failed to open MIDI output device Microsoft MIDI Mapper:
There is no driver installed on your system.
Start
16:15



The digital sensor connections ; powering and D3 as output
for a temperature sensor, DS18B20 (+125 to -55^{0}C).
You can do similar work with a standard analog sensor the LM35..
This is more educational since they get to do a bit of simple arithmetic before getting the temperature value!


D12 D11 D10 D9 D8
D13 3V3 AREF A0 A1 A2 A3
D4 D3 D2 GND RST RXD TX1
PWR
A7 +5V RST GND VIN



Κοστος ~30 ευρω