Intro

Physical Programming within the Minor Built Digital

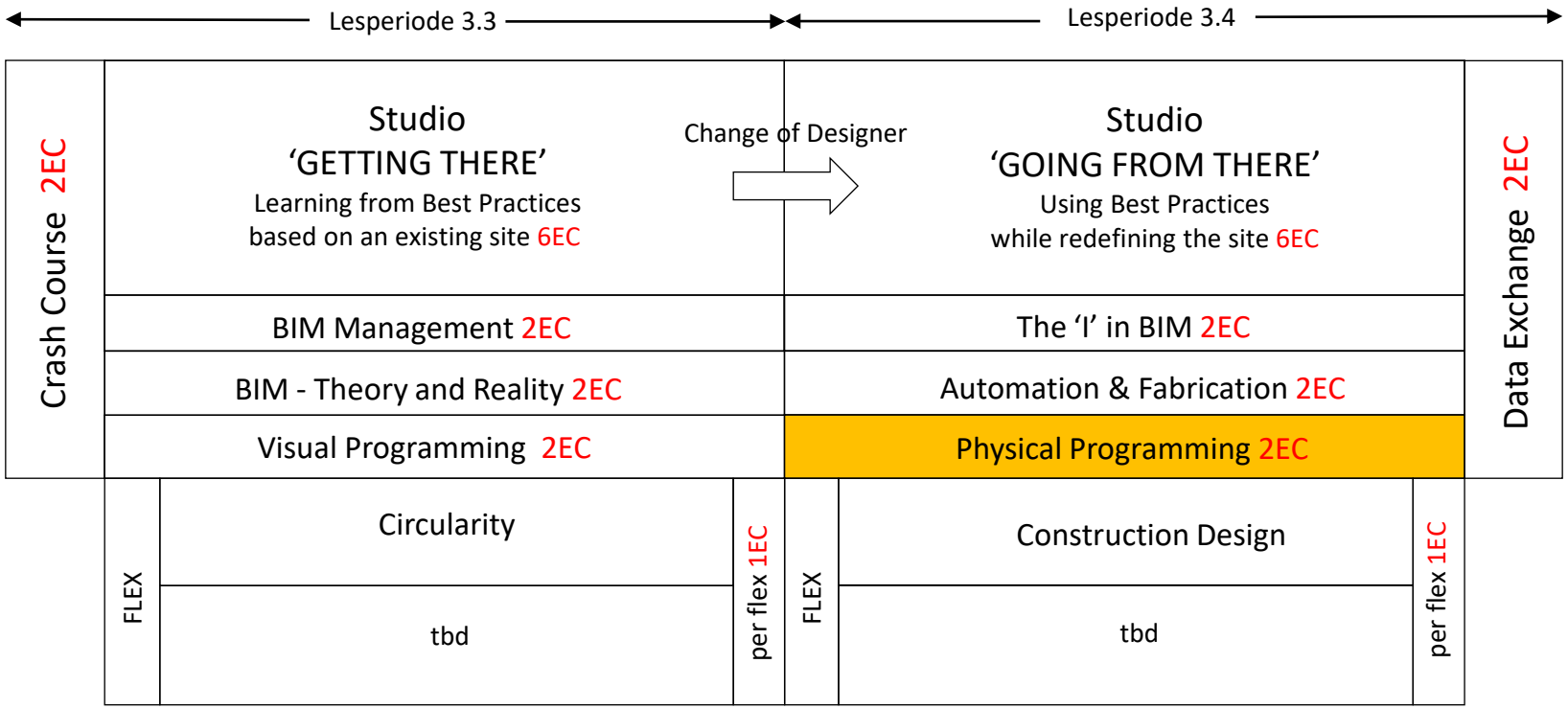
26.05.2023

Hanzehogeschool Groningen
University of Applied Sciences

Hanzehogeschool Groningen
University of Applied Sciences
BuilT Digital

Structure

of the minor ‘Built Digital’



just one elective per student
and quarter mandatory!

Hanzehogeschool Groningen
University of Applied Sciences
BuilT Digital

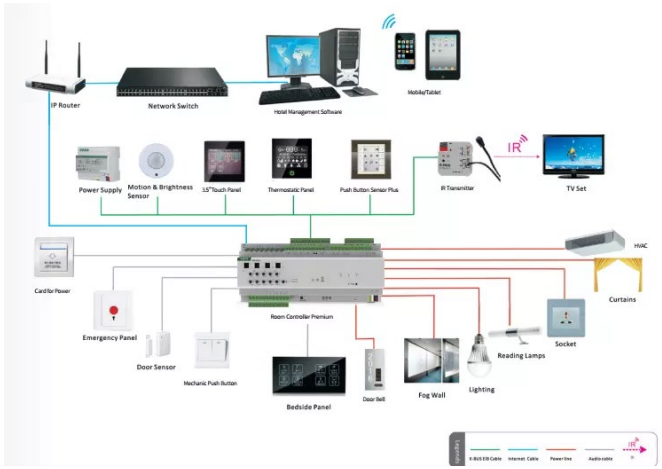
What do they have in common today?

Hanzehogeschool Groningen
University of Applied Sciences
BuilT Digital

What will they have in common tomorrow?

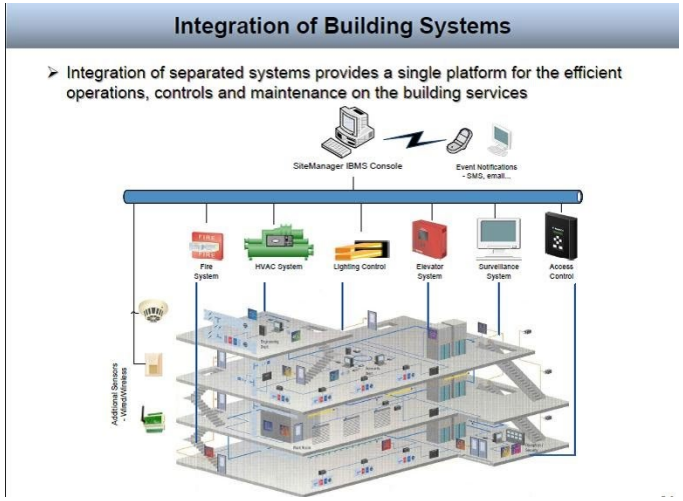
Hanzehogeschool Groningen
University of Applied Sciences
BuilT Digital

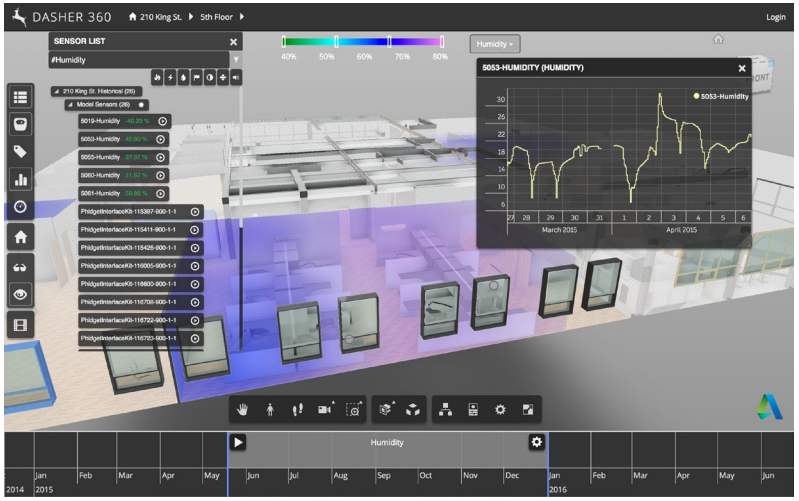
Home Domotica or BUS systems



Hanzehogeschool Groningen
University of Applied Sciences
BuilT Digital

FM





Hanzehogeschool Groningen
University of Applied Sciences
BuilT Digital

Platform possibilities

Raspberry Pi and Arduino are two very popular boards among electronics DIY builders, hobbyists and even professionals. Raspberry Pi and Arduino are quite different boards. While Arduino is aimed at quick programming and circuit prototyping, Raspberry Pi acts as a learning tool for Computer Programming (but you can find Raspberry Pi is several DIY Projects as well). Each board has its own advantages and disadvantages.

Arduino

The Arduino was developed by Massimo Banzi Et Al. in Ivrea, Italy. Arduino is a simple electronics prototyping tool with open-source hardware and software. Arduino is essentially a **Microcontroller development board** using which you can Blink LEDs, accept inputs from Buttons, read data from Sensors, control Motors and many other “Microcontroller” related tasks.

The most popular Arduino board is the Arduino UNO, which is based on ATmega328P Microcontroller from Atmel (now Microchip). Coming to the software side of Arduino, all Arduino boards can be programmed in C and C++ programming languages using a special software called Arduino IDE. The Arduino IDE consists of all the toolchains for editing source code, compiling and programming the Microcontroller on the Arduino board.

Raspberry Pi

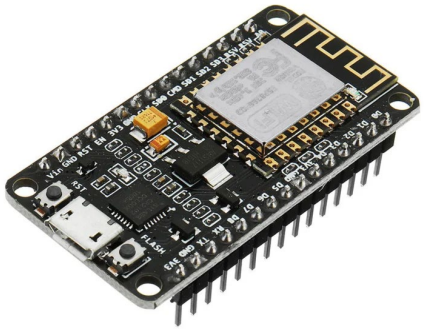
The Raspberry Pi was developed by Eben Upton at the University of Cambridge in the United Kingdom with the aim of teaching and improving programming skills of students in developing countries. While Arduino is a Microcontroller based development board, the Raspberry Pi is a **Microprocessor (usually an ARM Cortex A Series) based board that acts as a computer.**

You can connect several peripherals like a Monitor (through HDMI or AV Port), Mouse and Keyboard (through USB), connect to internet (through Ethernet or Wi-Fi), add a Camera (through the dedicated Camera Interface), just like we do to our desktop computer.

There are a lot more Microcontrollers and Microprocessors out there.

Hanzehogeschool Groningen
University of Applied Sciences
BuilT Digital

We will use the **Arduino & ESP Platforms** in this course



Hanzehogeschool Groningen
University of Applied Sciences
BuilT Digital

What is an Arduino?

Arduino is a microcontroller.

Arduino Logo 

Arduino is an open-source electronics prototyping platform based on flexible, easy-to-use hardware and software.

It's intended for artists, designers, hobbyists,
and anyone interested in creating interactive objects or environments.

You can download the latest version at the following source:
http://www.arduino.cc/

Hanzehogeschool Groningen
University of Applied Sciences
BuilT Digital

a tool

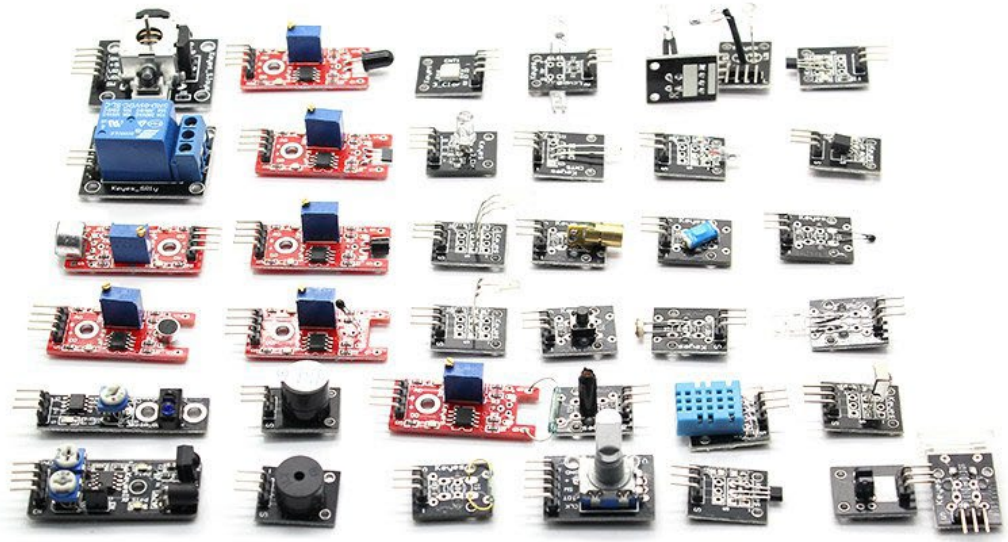
or the leatherman of physical input and output:

Hanzehogeschool Groningen
University of Applied Sciences
BuilT Digital

Inputs

Some examples of inputs:

Could be a temperature sensor, a motion sensor, a distance sensor, a switch and so forth.



Hanzehogeschool Groningen
University of Applied Sciences
BuilT Digital

outputs

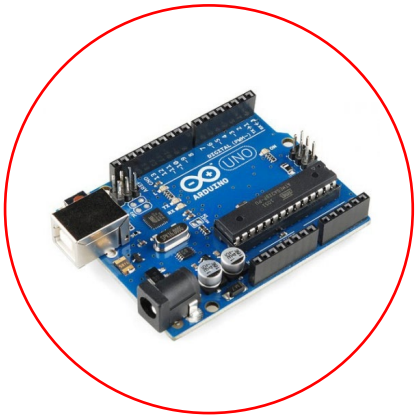
Some examples of outputs would be a light, a screen, a motor and so forth.

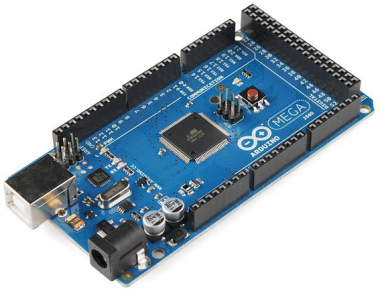
flavors

Arduinos come in different flavours.

Most people starting off go for the UNO board.

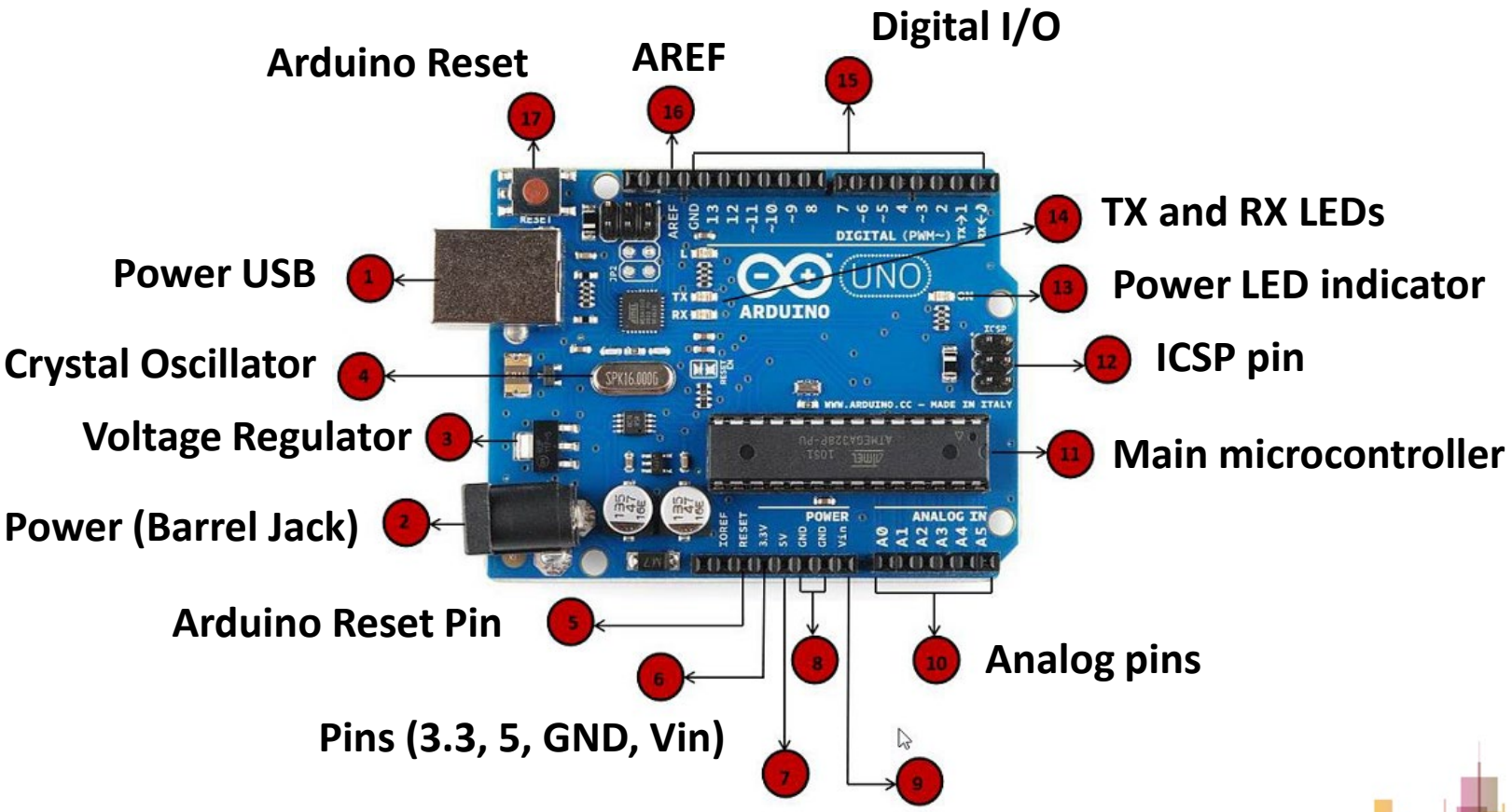
It’s current revision is the third (R3).





Hanzehogeschool Groningen
University of Applied Sciences
BuilT Digital

the classic UNO platform

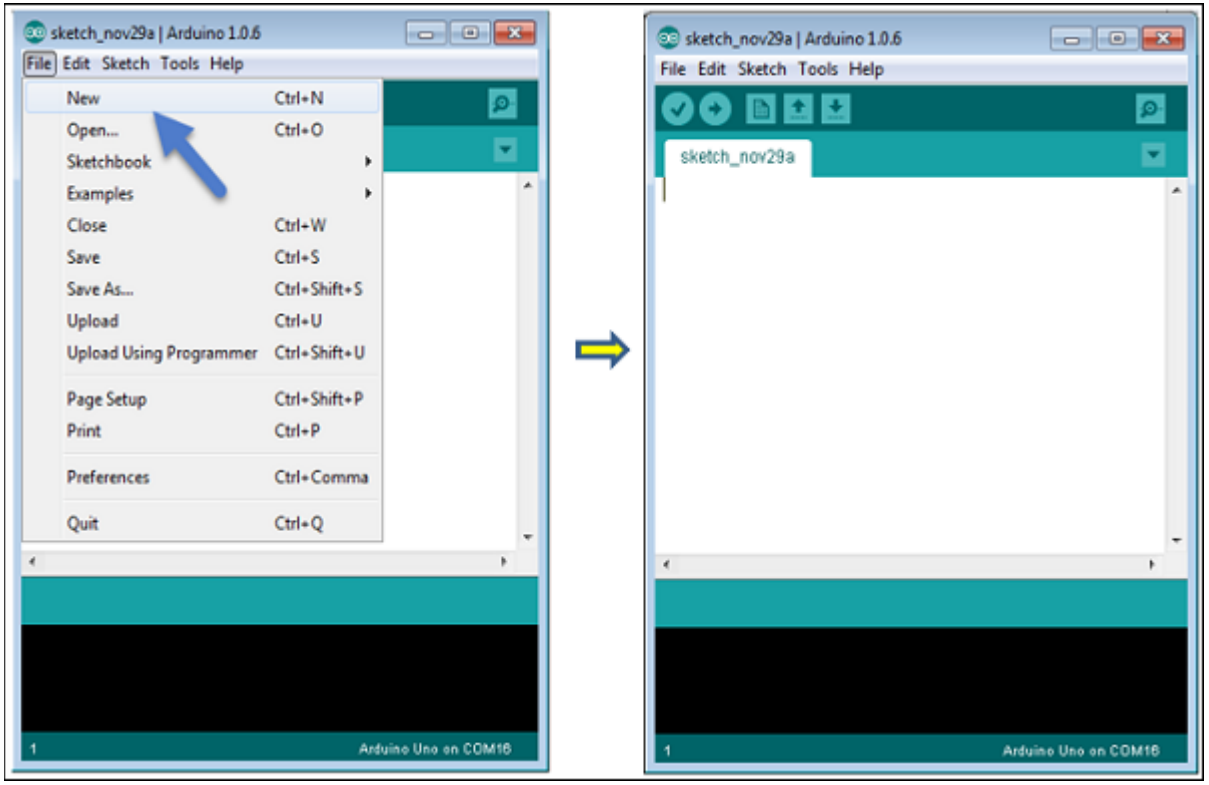


Hanzehogeschool Groningen
University of Applied Sciences
BuilT Digital

the connection

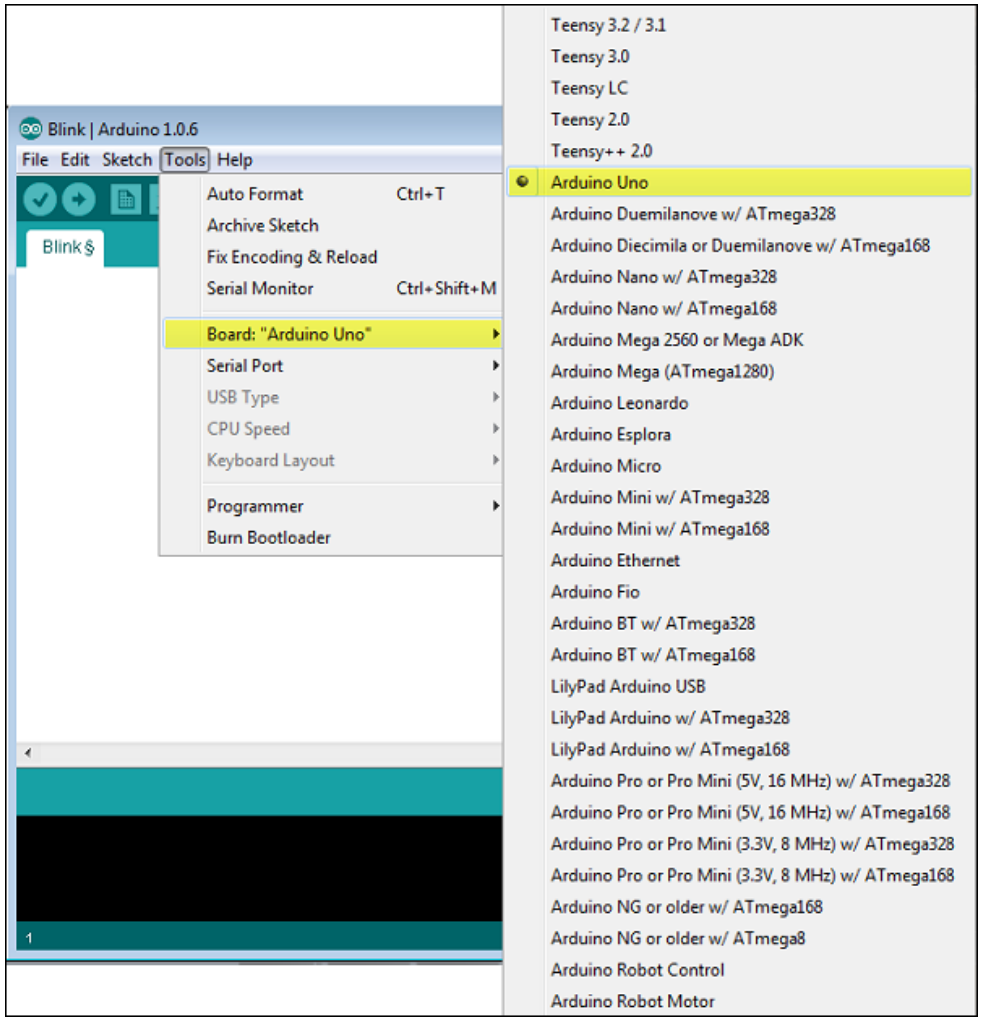
Hanzehogeschool Groningen
University of Applied Sciences
BuilT Digital

the IDE 'Integrated Development Environment'

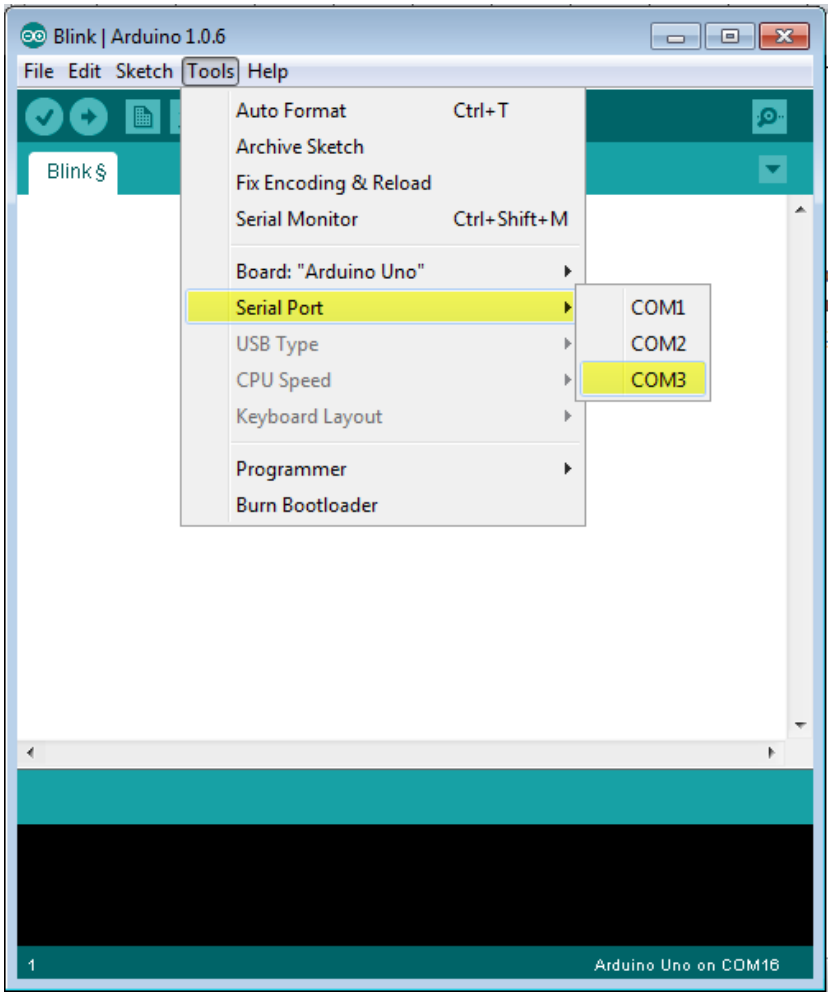


Hanzehogeschool Groningen
University of Applied Sciences
BuilT Digital

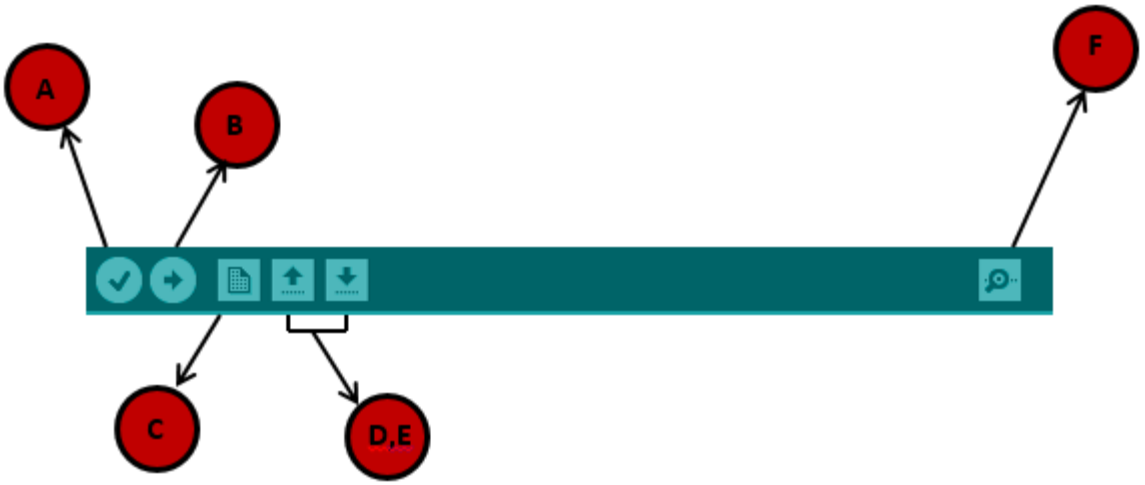
select your board



select your serial port



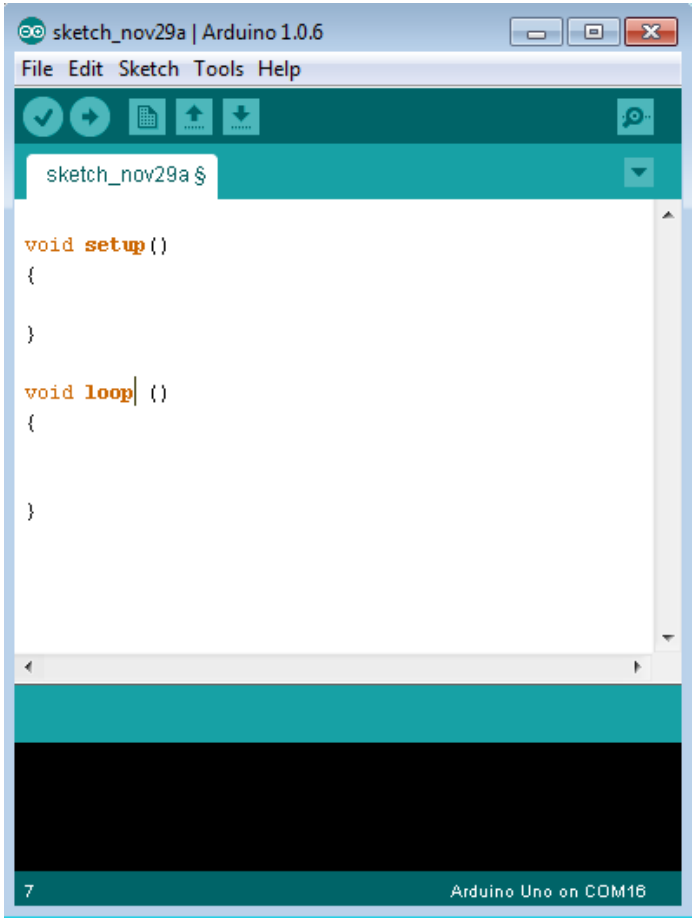
test & upload your sketch (program)



A – Used to check if there is any compilation error.
B – Used to upload a program to the Arduino board.
C – Shortcut used to create a new sketch.
D – Used to directly open one of the example sketch.
E – Used to save your sketch.
F – Serial monitor used to receive serial data from the board and send the serial data to the board.

Hanzehogeschool Groningen
University of Applied Sciences
BuilT Digital

sketch structure



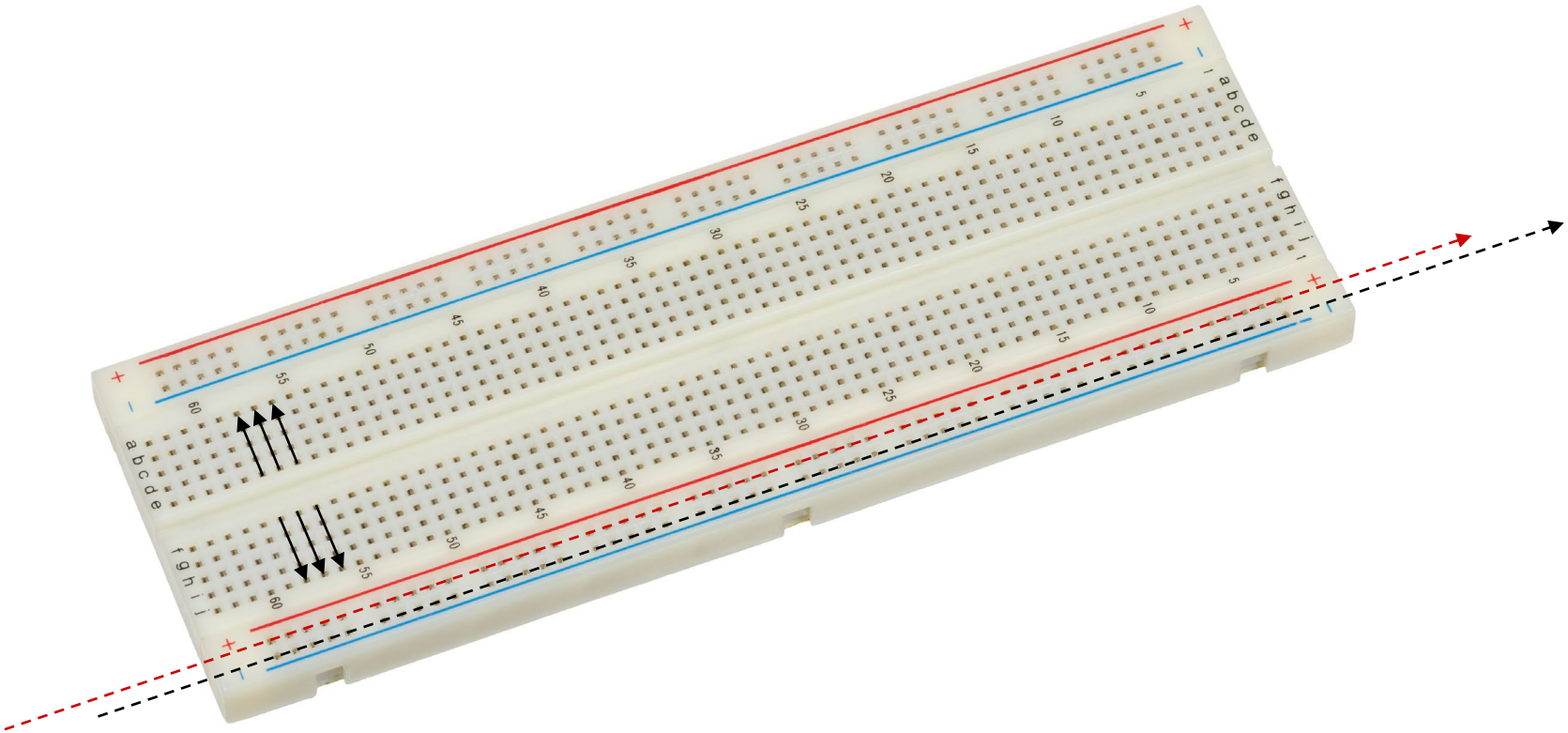
- The **setup()** function is called when a sketch starts. Use it to initialize the variables, pin modes, start using libraries, etc. The setup function will only run once, after each power up or reset of the Arduino board.

- After creating a **setup()** function, which initializes and sets the initial values, the **loop()** function does precisely what its name suggests, and loops consecutively, allowing your program to change and respond. Use it to actively control the Arduino board.

Hanzehogeschool Groningen
University of Applied Sciences
BuilT Digital

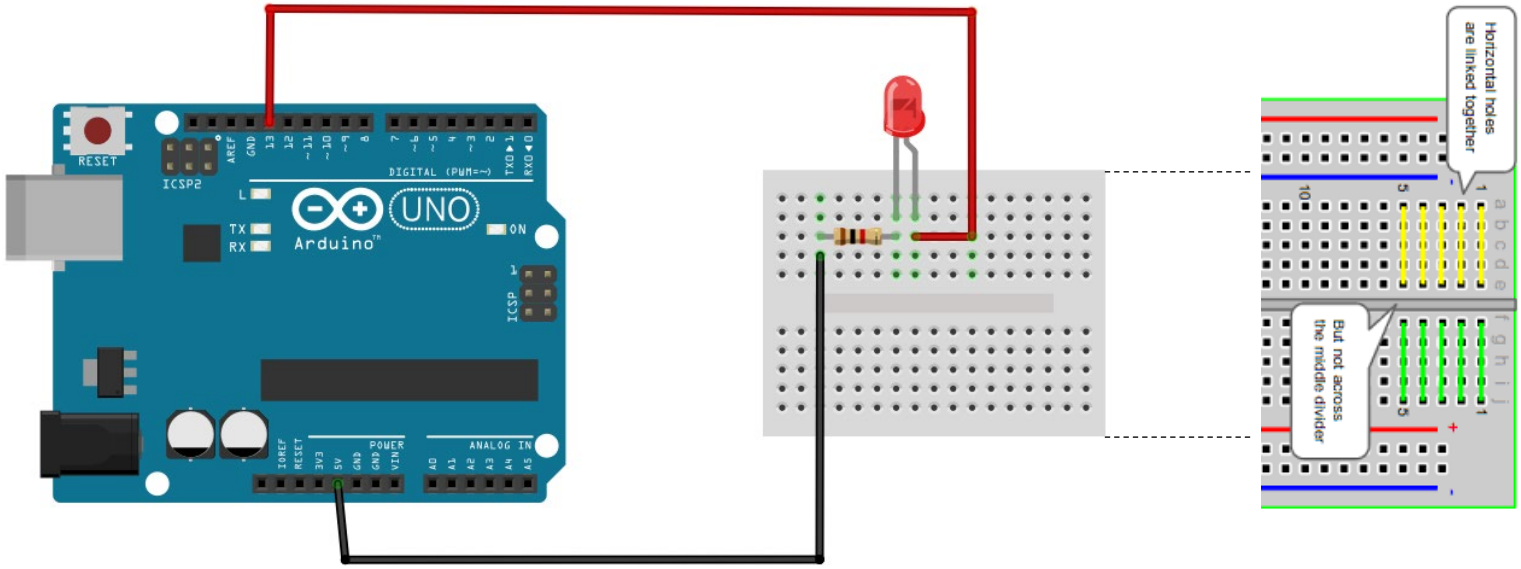

Salad?

Breadboard is your friend



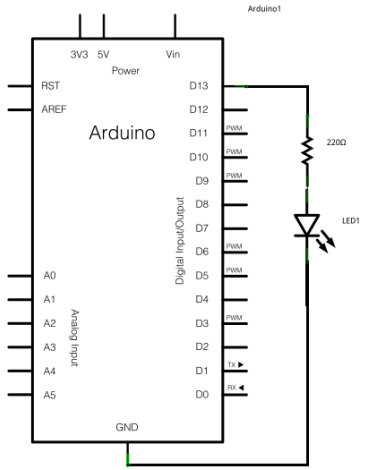
Hanzehogeschool Groningen
University of Applied Sciences
BuilT Digital

Blink



Hanzehogeschool Groningen
University of Applied Sciences
BuilT Digital

Blink (simple)




```
void setup() {

  pinMode(13, OUTPUT);      // initialize digital pin 13 as an output.

}


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
}
```

Hanzehogeschool Groningen
University of Applied Sciences
BuilT Digital

Blink (improved)

```
// Pin number where the LED is connected
int ledPin = 13;

// Time interval in milliseconds
int interval = 1000;


void setup() {
  // Initialize the LED pin as an output
  pinMode(ledPin, OUTPUT);
}


void loop() {
  // Turn on the LED
  digitalWrite(ledPin, HIGH);

  // Wait for the specified interval
  delay(interval);

  // Turn off the LED
  digitalWrite(ledPin, LOW);

  // Wait for the specified interval
  delay(interval);
}
```

Hanzehogeschool Groningen
University of Applied Sciences
BuilT Digital

Blink (using functions)

Part 01

```
// Pin number where the LED is connected
int ledPin = 13;

// Time interval in milliseconds
int interval = 1000;

void setup() {
  // Initialize the LED pin as an output
  pinMode(ledPin, OUTPUT);
}


void loop() {
  // calling function 01
  blink_01();
  }
```

Hanzehogeschool Groningen
University of Applied Sciences
BuilT Digital

Blink (using functions)

Part 02

```
void blink_01()
{
  while( true ) {
      digitalWrite(ledPin, HIGH);
      delay(500);
      digitalWrite(ledPin, LOW);
      delay(500);
  }
}

void blink_02()
{
  for (int i = 0; i < 5; i++)
  {
      digitalWrite(ledPin, HIGH);
      delay(500);
      digitalWrite(ledPin, LOW);
      delay(500);
  }
}
```

LOOP FUNCTIONS:

Option 01:

```
while(condition){
    // code goes here
}
```

Option 02:

```
for(initialization; condition; iteration) {
   // code goes here
}
```

Hanzehogeschool Groningen
University of Applied Sciences
BuilT Digital

Switch LED
by serial commands
(numbers)

```
// Pin assignments
const int ledPin = 13;

void setup() {
  Serial.begin(9600);   // Initialize serial communication
  pinMode(ledPin, OUTPUT); // Set LED pin as output
  digitalWrite(ledPin, LOW); // Initialize LED to be off
}

void loop() {
  if (Serial.available() > 0) {
    char command = Serial.read(); // Read the incoming command

    if (command == '1') {
      digitalWrite(ledPin, HIGH); // Turn on the LED
      Serial.println("LED is ON");
    }
    else if (command == '0') {
      digitalWrite(ledPin, LOW); // Turn off the LED
      Serial.println("LED is OFF");
    }
  }
}
```

Hanzehogeschool Groningen
University of Applied Sciences
BuilT Digital

Switch LED
by serial commands
(strings)

```
// Pin assignments
const int ledPin = 13;

void setup() {
  Serial.begin(9600);   // Initialize serial communication
  pinMode(ledPin, OUTPUT); // Set LED pin as output
  digitalWrite(ledPin, LOW); // Initialize LED to be off
}

void loop() {
  if (Serial.available() > 0) {
    String command = Serial.readStringUntil('\n'); // Read the incoming command

    if (command.equalsIgnoreCase("on")) {
      digitalWrite(ledPin, HIGH); // Turn on the LED
      Serial.println("LED is ON");
    }
    else if (command.equalsIgnoreCase("off")) {
      digitalWrite(ledPin, LOW); // Turn off the LED
      Serial.println("LED is OFF");
    }
  }
}
```

Hanzehogeschool Groningen
University of Applied Sciences
BuilT Digital

Hello Minor (Comment & Setup)

Part 01

```
// this code was written by Boris Baehre (b.bahre@pl.hanze.nl) based on...

/*
 This code is listening to serial inputs,
 switches a led on or off on user commands typed into the serial monitor
 and will respond with serial data and a called function to blink the led.
 05/2023, MBD_VI, SOFE, Hanzehogeschool, Groningen, NL
*/

// Pin assignments
const int ledPin = 13;

void setup() {
  Serial.begin(9600);   // Initialize serial communication
  pinMode(ledPin, OUTPUT); // Set LED pin as output
  digitalWrite(ledPin, LOW); // Initialize LED to be off
  while (!Serial);
  Serial.println("Type 'on' or 'off' or 'Hello Arduino!'");
}
```

Hello Minor (Loop)

Part 02

```
void loop()
{
 if (Serial.available() > 0) {
  String command = Serial.readStringUntil('\n'); // Read the incoming command

  if (command.equalsIgnoreCase("on")) {
   digitalWrite(ledPin, HIGH); // Turn on the LED
   Serial.println("LED is ON");
  }

  else if (command.equalsIgnoreCase("off")) {
   digitalWrite(ledPin, LOW); // Turn off the LED
   Serial.println("LED is OFF");
  }

   else if (command.equalsIgnoreCase("Hello Arduino!")) {
   digitalWrite(ledPin, LOW); // Turn off the LED
   Serial.println("Hello member of MBD!");
   blink_01();
  }
 }
}
```

Hanzehogeschool Groningen
University of Applied Sciences
BuilT Digital

Hello Minor (Functions)

Part 03

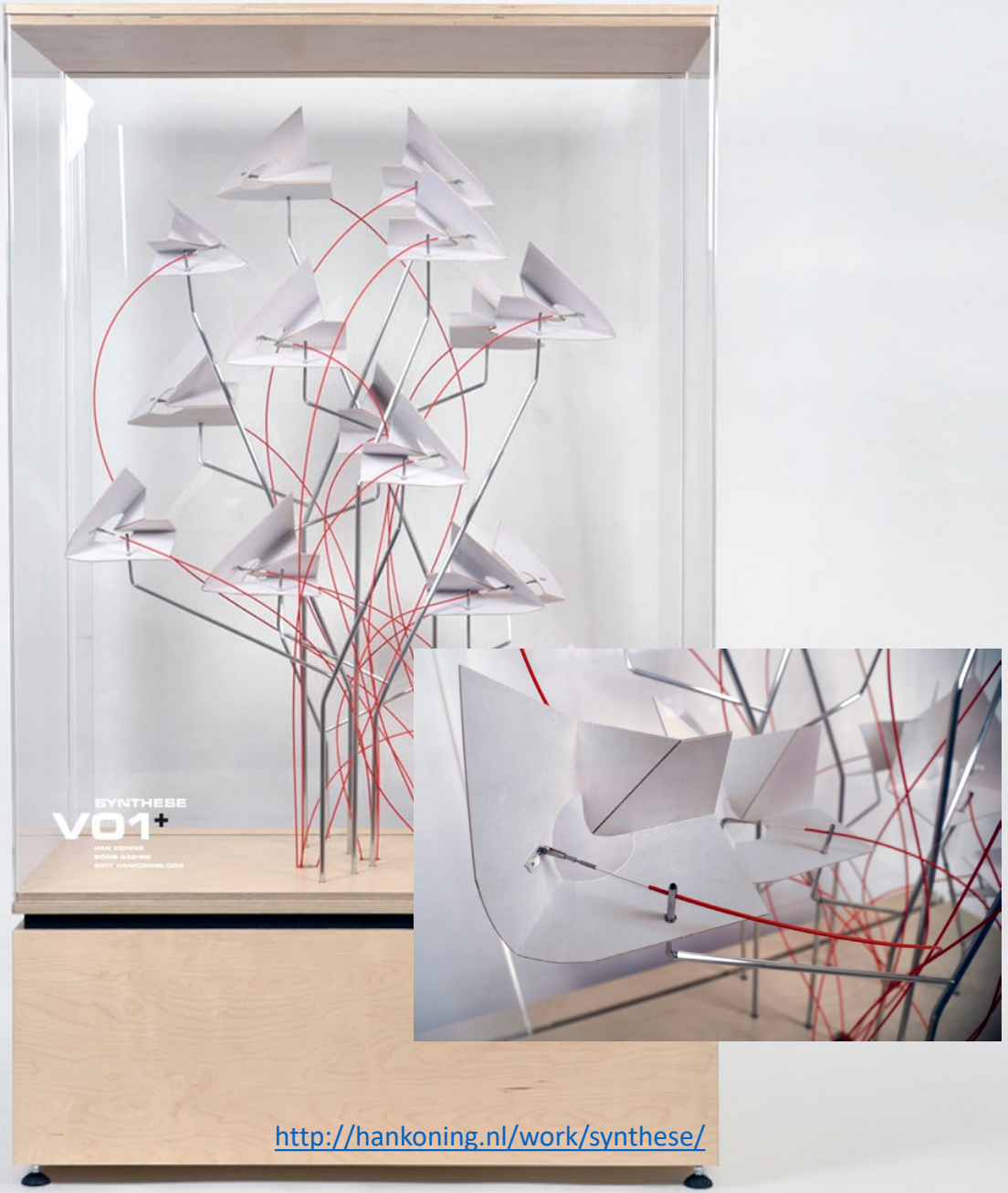
```
void blink_01()
{
  while( true ) {
      digitalWrite(ledPin, HIGH);
      delay(500);
      digitalWrite(ledPin, LOW);
      delay(500);
  }
}

void blink_02()
{
  for (int i = 0; i < 5; i++)
  {
      digitalWrite(ledPin, HIGH);
      delay(500);
      digitalWrite(ledPin, LOW);
      delay(500);
  }
}
```

Hanzehogeschool Groningen
University of Applied Sciences
BuilT Digital

The long https://vimeo.com/149964228

Synthese



http://hankoning.nl/work/synthese/

http://arduinoarts.com/2014/05/9-amazing-projects-where-arduino-art-meet/

...


Hanzehogeschool
Groningen
University of Applied Sciences
BuilT Digital