

Société Chimique de France
Division Enseignement-Formation
Mercredi 27 Mars 2019

Assemblons des « briques »
pour concevoir nos systèmes !

Alain Demolliens
Lycée Carnot - Dijon
ademolliens@mac.com




```
Temperature_Modele | Arduino 1.8.8
Temperature_Modele
1 #include <Wire.h>
2 #include <SoftwareSerial.h>
3 #include <OneWire.h>
4 #include <DallasTemperature.h>
5
6 // Les utilitaires pour les programmes
7 #define ONE_WIRE_BUS 4
8 #define ONE_WIRE_BUS 7
9 #define ONE_WIRE_BUS 7
10 #define ONE_WIRE_BUS 7
11 #define ONE_WIRE_BUS 7
12 // variables locales au programme
13 long delay = 1000; // Delai en ms entre 2 appels de la fonction job
14 bool controle = false; // à vrai si delai = 0
15 bool debug = false;
16
17 // variables spécifiques au systeme étudié
18 #define ONE_WIRE_BUS 7
19 #define ONE_WIRE_BUS 7
20 #define ONE_WIRE_BUS 7
21 #define ONE_WIRE_BUS 7
22 #define ONE_WIRE_BUS 7
23 #define ONE_WIRE_BUS 7
24 #define ONE_WIRE_BUS 7
25 #define ONE_WIRE_BUS 7
26 #define ONE_WIRE_BUS 7
27 #define ONE_WIRE_BUS 7
28 #define ONE_WIRE_BUS 7
29 #define ONE_WIRE_BUS 7
30 #define ONE_WIRE_BUS 7
31 #define ONE_WIRE_BUS 7
32 #define ONE_WIRE_BUS 7
33 #define ONE_WIRE_BUS 7
34 #define ONE_WIRE_BUS 7
35 #define ONE_WIRE_BUS 7
36 #define ONE_WIRE_BUS 7
37 #define ONE_WIRE_BUS 7
38 #define ONE_WIRE_BUS 7
39 #define ONE_WIRE_BUS 7
40 #define ONE_WIRE_BUS 7
41 #define ONE_WIRE_BUS 7
42 #define ONE_WIRE_BUS 7
43 #define ONE_WIRE_BUS 7
44 #define ONE_WIRE_BUS 7
45 #define ONE_WIRE_BUS 7
46 #define ONE_WIRE_BUS 7
47 #define ONE_WIRE_BUS 7
48 #define ONE_WIRE_BUS 7
49 #define ONE_WIRE_BUS 7
50 #define ONE_WIRE_BUS 7
51 #define ONE_WIRE_BUS 7
52 #define ONE_WIRE_BUS 7
53 #define ONE_WIRE_BUS 7
54 #define ONE_WIRE_BUS 7
55 #define ONE_WIRE_BUS 7
56 #define ONE_WIRE_BUS 7
57 #define ONE_WIRE_BUS 7
58 #define ONE_WIRE_BUS 7
59 #define ONE_WIRE_BUS 7
60 #define ONE_WIRE_BUS 7
61 #define ONE_WIRE_BUS 7
62 #define ONE_WIRE_BUS 7
63 #define ONE_WIRE_BUS 7
64 #define ONE_WIRE_BUS 7
65 #define ONE_WIRE_BUS 7
66 #define ONE_WIRE_BUS 7
67 #define ONE_WIRE_BUS 7
68 #define ONE_WIRE_BUS 7
69 #define ONE_WIRE_BUS 7
70 #define ONE_WIRE_BUS 7
71 #define ONE_WIRE_BUS 7
72 #define ONE_WIRE_BUS 7
73 #define ONE_WIRE_BUS 7
74 #define ONE_WIRE_BUS 7
75 #define ONE_WIRE_BUS 7
76 #define ONE_WIRE_BUS 7
77 #define ONE_WIRE_BUS 7
78 #define ONE_WIRE_BUS 7
79 #define ONE_WIRE_BUS 7
80 #define ONE_WIRE_BUS 7
81 #define ONE_WIRE_BUS 7
82 #define ONE_WIRE_BUS 7
83 #define ONE_WIRE_BUS 7
84 #define ONE_WIRE_BUS 7
85 #define ONE_WIRE_BUS 7
86 #define ONE_WIRE_BUS 7
87 #define ONE_WIRE_BUS 7
88 #define ONE_WIRE_BUS 7
89 #define ONE_WIRE_BUS 7
90 #define ONE_WIRE_BUS 7
91 #define ONE_WIRE_BUS 7
92 #define ONE_WIRE_BUS 7
93 #define ONE_WIRE_BUS 7
94 #define ONE_WIRE_BUS 7
95 #define ONE_WIRE_BUS 7
96 #define ONE_WIRE_BUS 7
97 #define ONE_WIRE_BUS 7
98 #define ONE_WIRE_BUS 7
99 #define ONE_WIRE_BUS 7
100 #define ONE_WIRE_BUS 7
```

IDE Arduino

Programmation

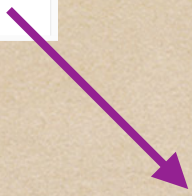


Ordinateur



Arduino

Alimentation

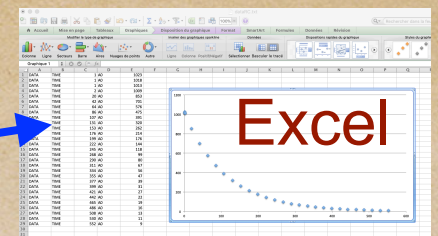



```
Temperature_Modele
1 #include <Wire.h>
2 #include <SoftwareSerial.h>
3 #include <Arduino.h>
4 #include <DallasTemperature.h>
5
6 // Adresse de la DS18B20
7 #define ONE_WIRE_BUS 4
8
9 // Adresse du DS18B20
10 #define ONE_WIRE_BUS 4
11
12 // Variables globales
13 long delay = 1000; // Délai en ms entre 2 appels de la fonction job
14 bool controle = false; // à vrai si delay = 0
15 bool debug = false;
16
17 // variables spécifiques au système étudié
18 #define ONE_WIRE_BUS 4
19 #define ONE_WIRE_BUS 4
20 #define ONE_WIRE_BUS 4
21 #define ONE_WIRE_BUS 4
```

IDE Arduino

```
Serial
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
```

Moniteur série



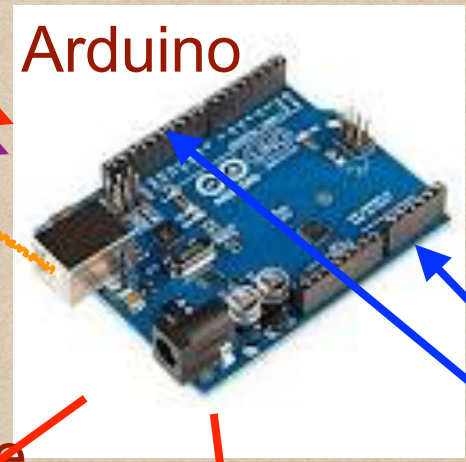
Excel

```
Python
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
```

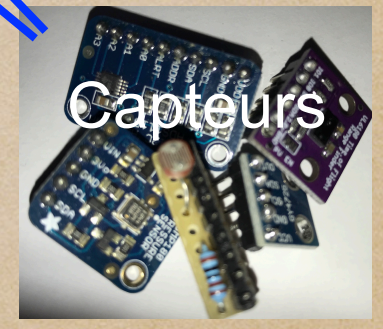
Python



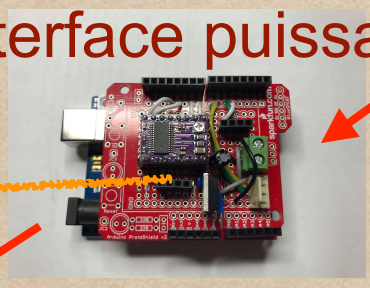
Ordinateur



Arduino



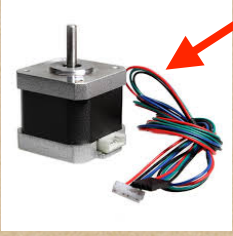
Capteurs



Interface puissance



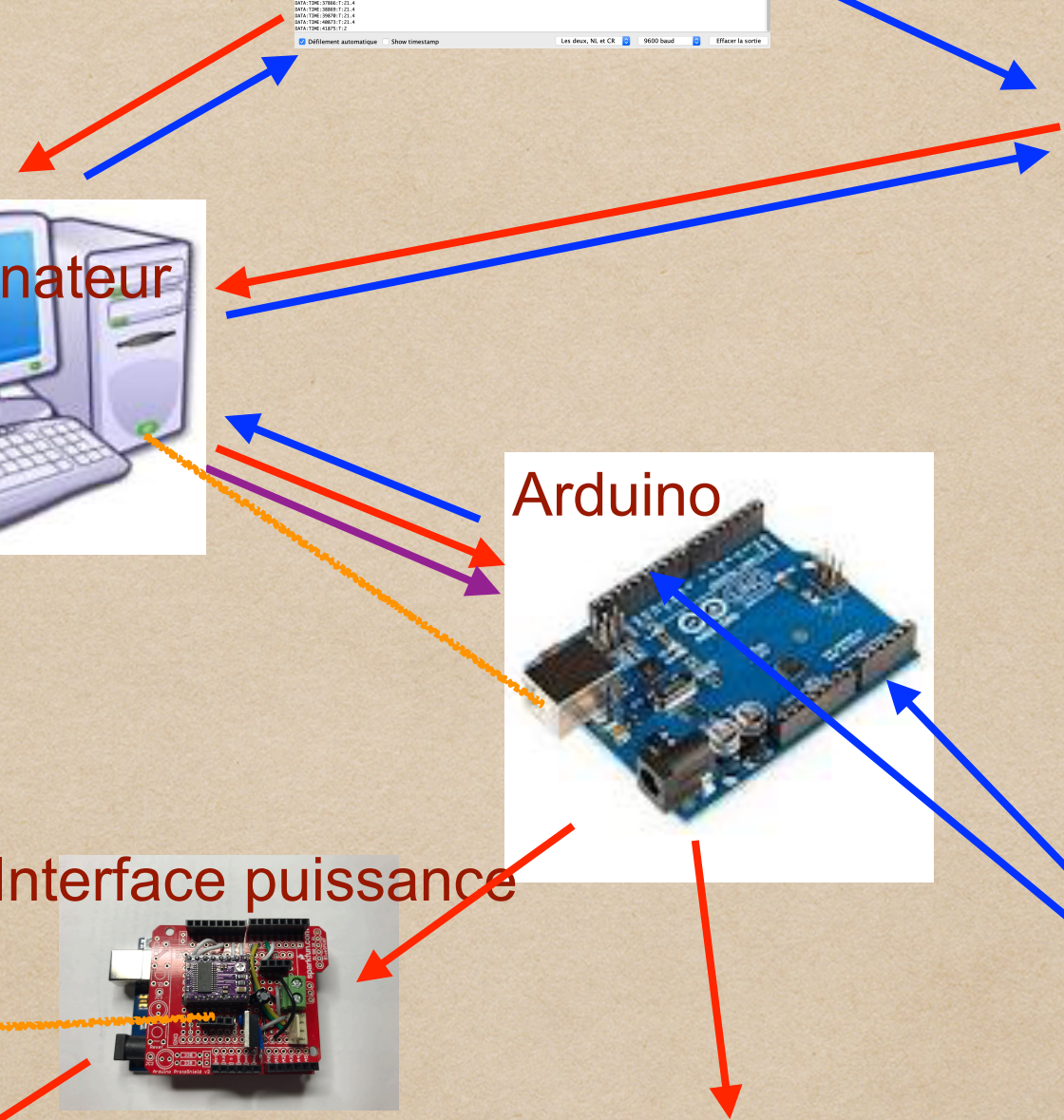
15 V



> 5 V, 100 mA



< 5V, < 100 mA

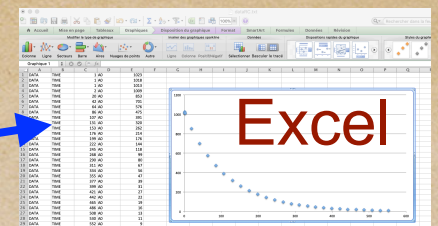



```
Temperature_Moche | Arduino 1.6.8
Temperature_Moche
1 #include <Wire.h>
2 #include <SoftwareSerial.h>
3 #include <OneWire.h>
4 #include <DallasTemperature.h>
5
6 // Adresse de la sonde DS18B20
7 #define ONE_WIRE_BUS 7
8
9 OneWire oneWire(ONE_WIRE_BUS);
10 DallasTemperature sensors(oneWire);
11
12 // variables spécifiques au système étudié
13 Adresse ONE_WIRE_BUS ?
14 bool controle = false; // à vrai si delay = 0
15 bool debug = false;
16
17 // variables spécifiques au système étudié
18 Adresse ONE_WIRE_BUS ?
19 OneWire oneWire(ONE_WIRE_BUS);
20 DallasTemperature sensors(oneWire);
```

IDE Arduino

```
Serial Monitor (Arduino/Genuino Uno)
5
DATA TIME 12066 F:22.4
DATA TIME 12067 F:22.3
DATA TIME 12068 F:22.2
DATA TIME 12069 F:22.1
DATA TIME 12070 F:22.0
DATA TIME 12071 F:21.9
DATA TIME 12072 F:21.8
DATA TIME 12073 F:21.7
DATA TIME 12074 F:21.6
DATA TIME 12075 F:21.5
DATA TIME 12076 F:21.4
DATA TIME 12077 F:21.3
DATA TIME 12078 F:21.2
DATA TIME 12079 F:21.1
DATA TIME 12080 F:21.0
DATA TIME 12081 F:20.9
DATA TIME 12082 F:20.8
DATA TIME 12083 F:20.7
DATA TIME 12084 F:20.6
DATA TIME 12085 F:20.5
DATA TIME 12086 F:20.4
DATA TIME 12087 F:20.3
DATA TIME 12088 F:20.2
DATA TIME 12089 F:20.1
DATA TIME 12090 F:20.0
DATA TIME 12091 F:19.9
DATA TIME 12092 F:19.8
DATA TIME 12093 F:19.7
DATA TIME 12094 F:19.6
DATA TIME 12095 F:19.5
DATA TIME 12096 F:19.4
DATA TIME 12097 F:19.3
DATA TIME 12098 F:19.2
DATA TIME 12099 F:19.1
DATA TIME 12100 F:19.0
```

Moniteur série



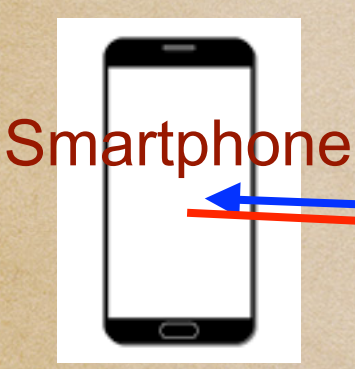
Excel

```
Python
# Python code for data processing and visualization
import time
import math
import matplotlib.pyplot as plt

# Example code snippet
def calculate_average(values):
    return sum(values) / len(values)

# Plotting example
plt.plot([1, 2, 3, 4, 5], [10, 8, 6, 4, 2])
plt.show()
```

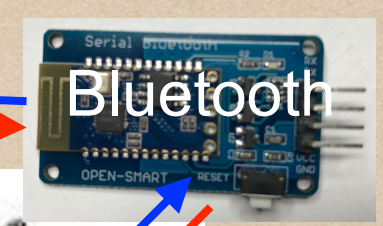
Python



Smartphone



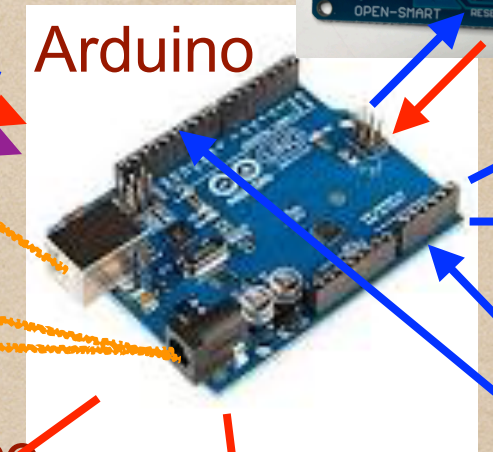
Ordinateur



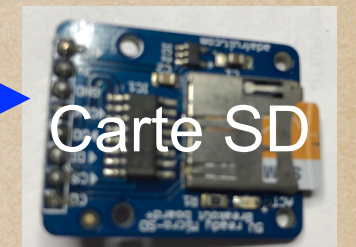
Bluetooth



LCD



Arduino



Carte SD

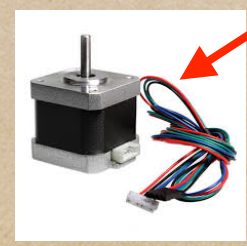
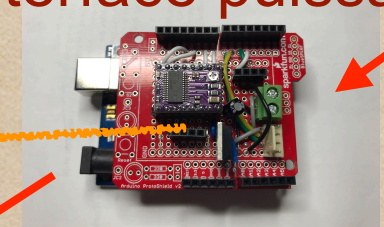


9-12 V

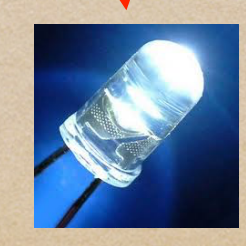


15 V

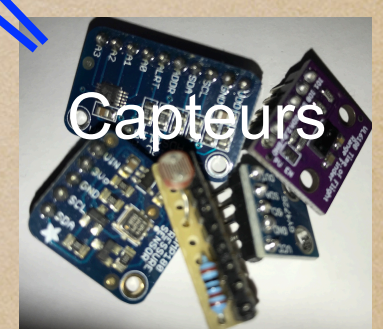
Interface puissance



> 5 V, > 100 mA



< 5V, < 100 mA



Capteurs

Pourquoi faire ?

Pour quoi faire ?

Pourquoi faire ?

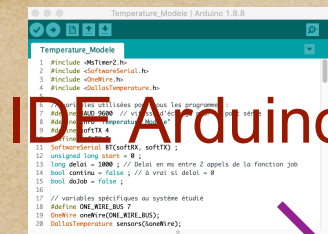
- parce que le programme va nous l'imposer !
- par curiosité ;
- pour renouveler quelques pratiques expérimentales ;
- pour motiver les élèves ;
- etc...

- pour réaliser des acquisitions automatiques ;
- pour réaliser des systèmes ouverts ;
- pour essayer, modestement, de proposer à l'échelle d'une paillasse de lycée quelques « techniques » du monde de l'industrie ou de la recherche...

Pour quoi faire ?

- des acquisitions automatiques pour traitement/
modélisation ultérieur en python :
=> température, masse, ddp...
- des systèmes d'acquisition/contrôle à moindre coût :
=> burette automatique
- mettre les fluides en mouvement :
=> réalisation de systèmes ouverts...
- proposer des systèmes plus complexes :
=> banc de traction, calorimétrie isotherme...
- proposer quelques solutions pour les TIPE...

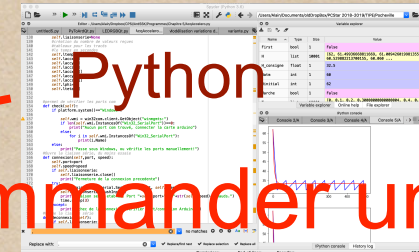
Comment faire ?



IDE Arduino



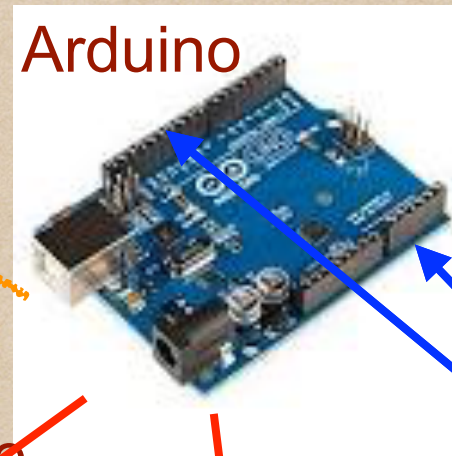
Moniteur série



Python



Ordinateur

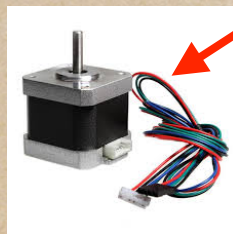
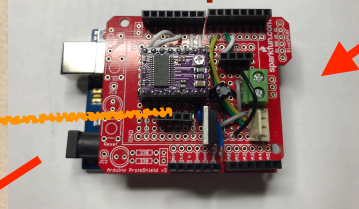


Arduino



15 V

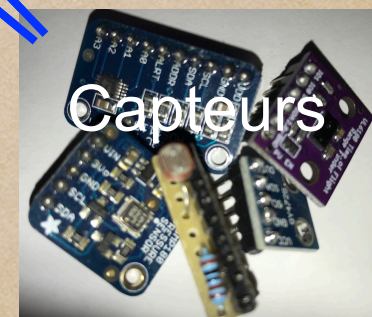
Interface puissance



> 5 V, > 100 mA



< 5V, < 100 mA



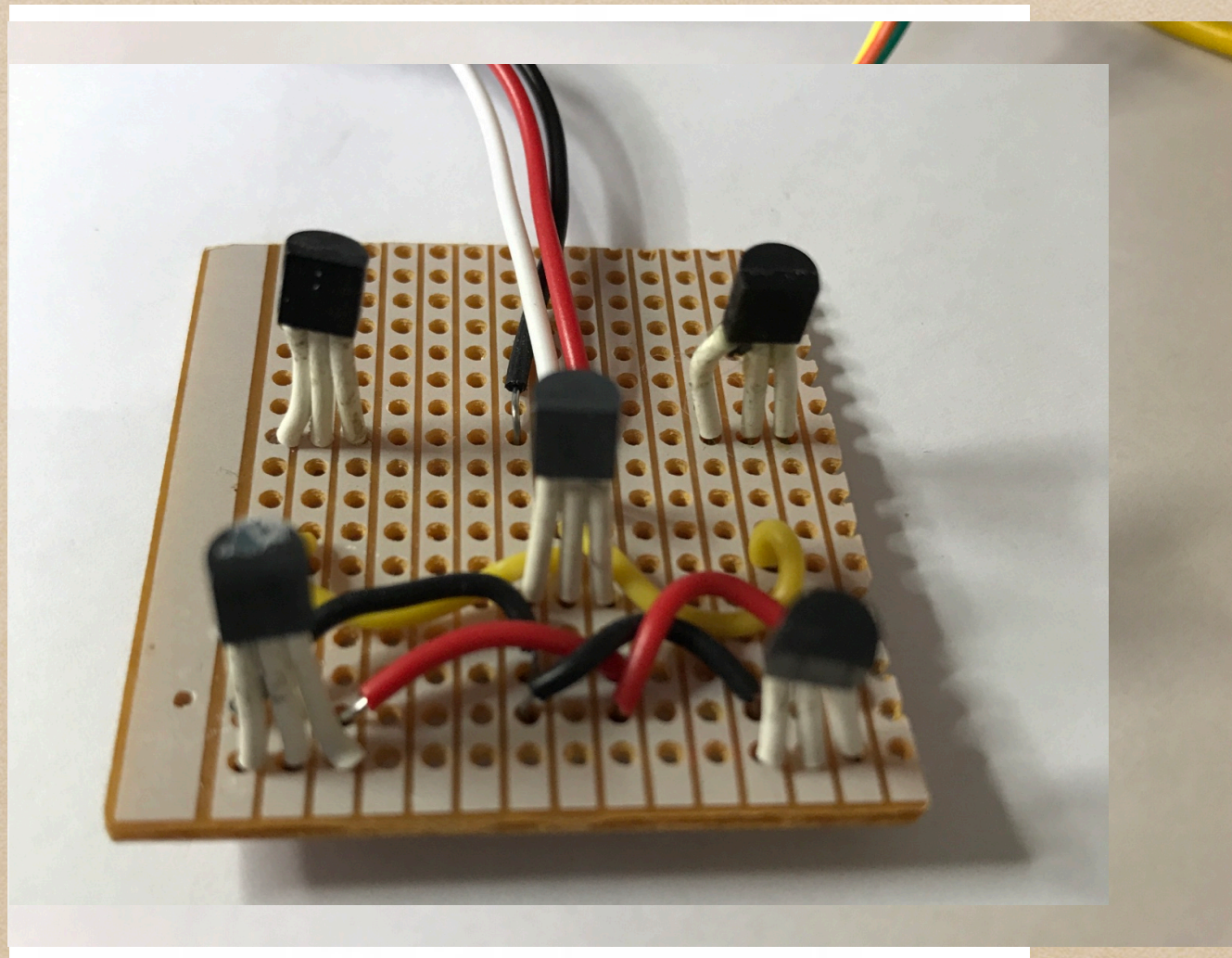
Capteurs

Apprendre à commander une LED

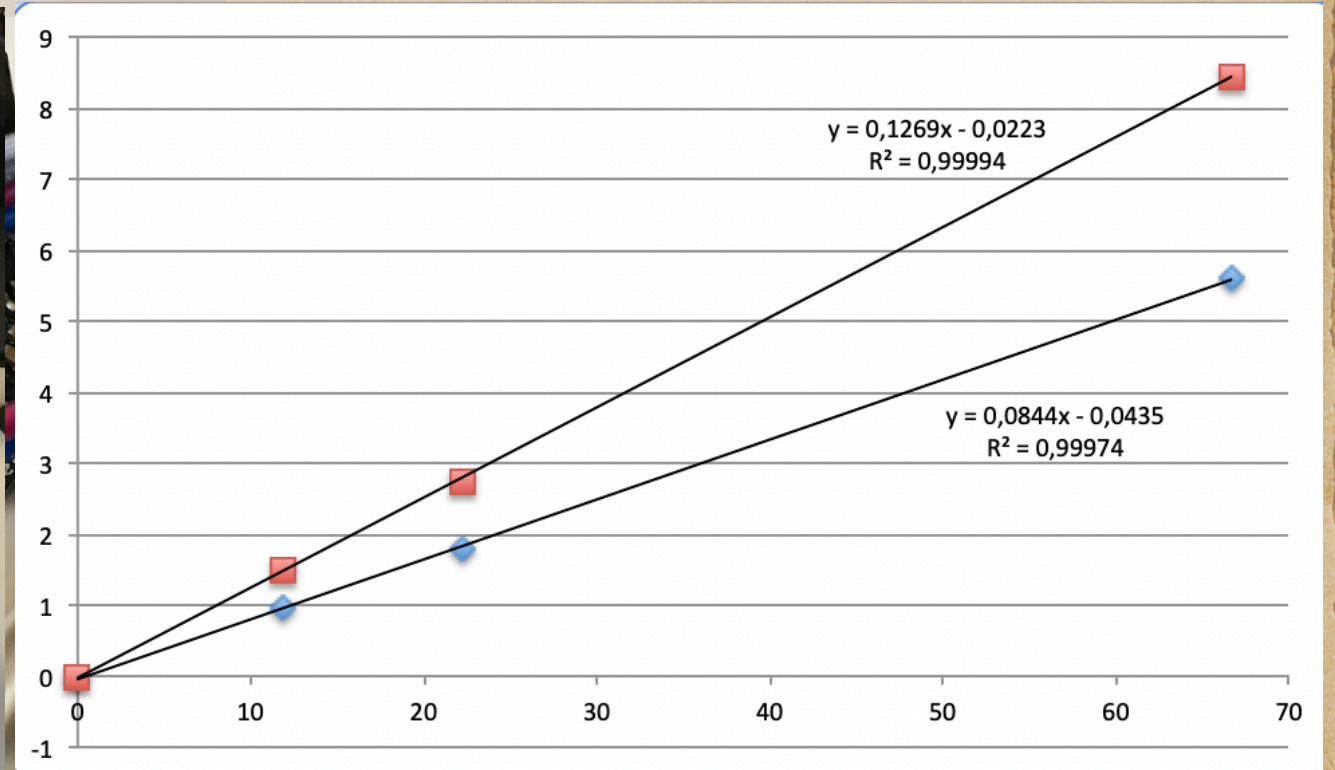
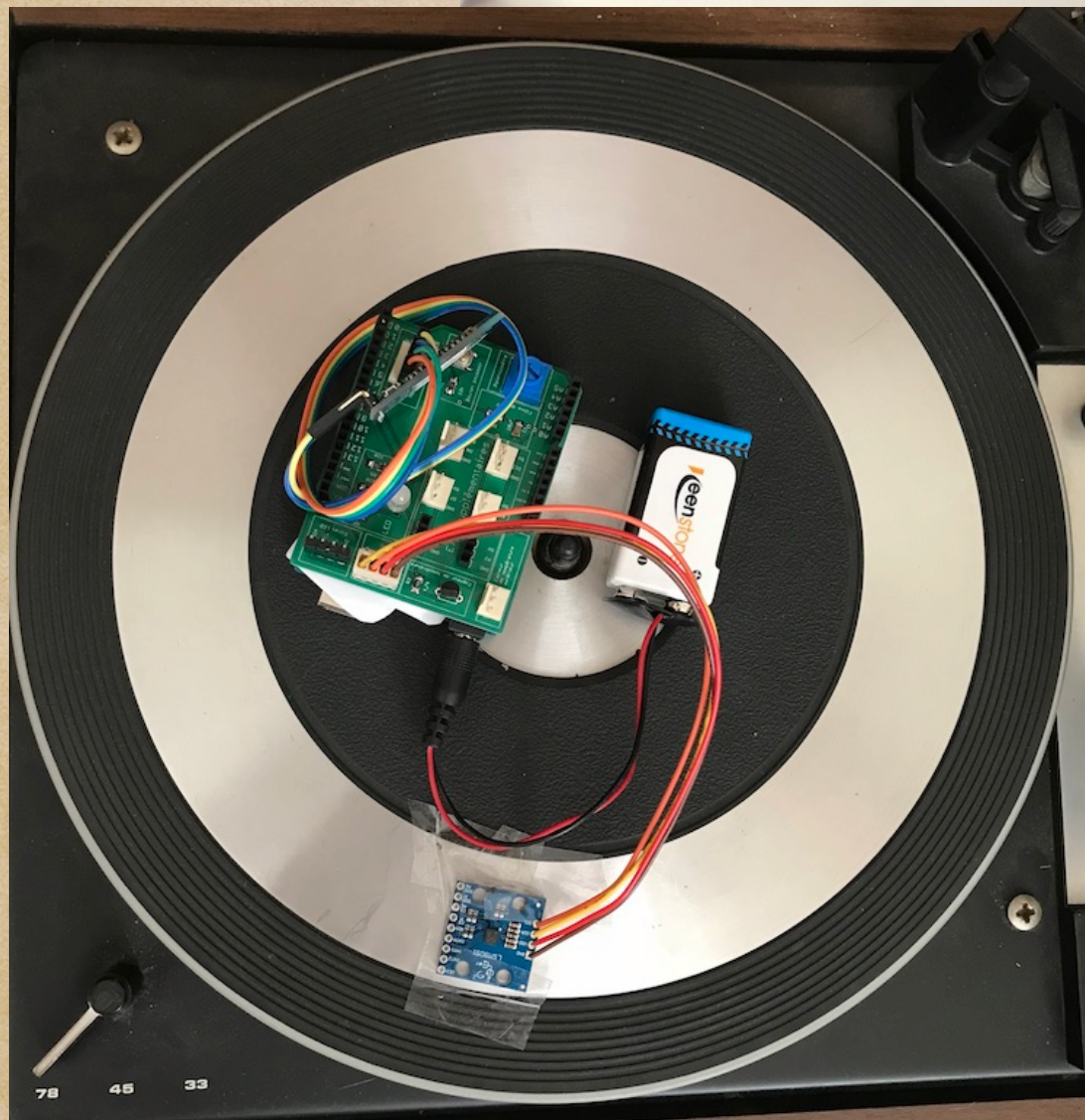
Apprendre à faire une acquisition

Un capteur...

... différents conditionnements



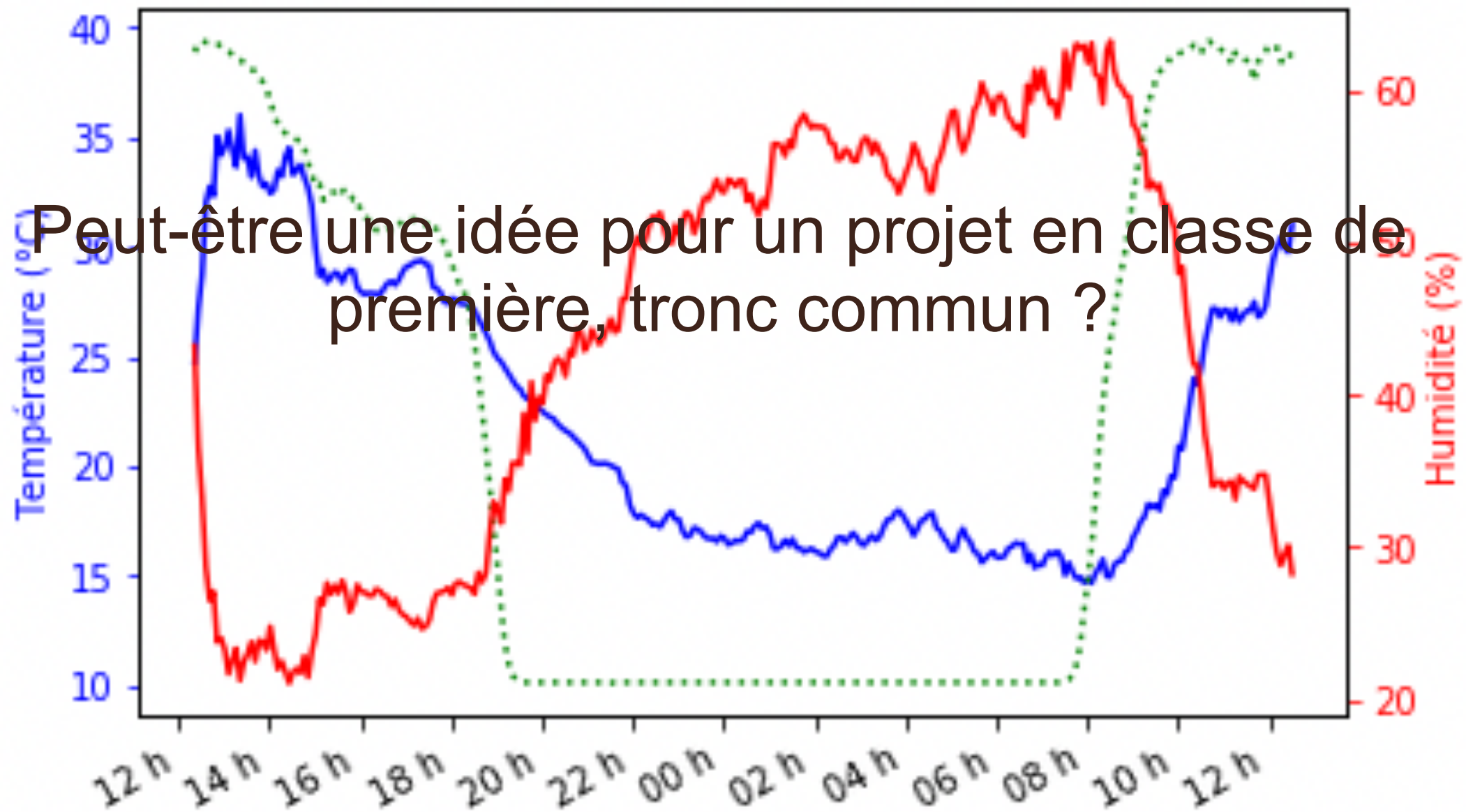
Un capteur... ... différentes mises en oeuvre



Accélération fonction de ω^2
pour différentes valeurs de R

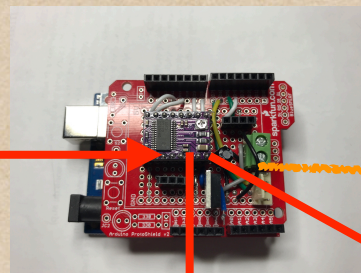
Plusieurs capteurs...

... une mini station météo !



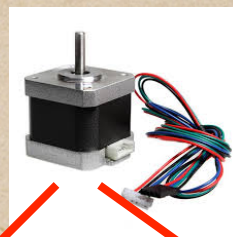
Peut-être une idée pour un projet en classe de première, tronc commun ?

Interface de puissance



15-24 V

Moteur pas à pas



Circuit de « puissance »

- relais
- transistor
- ...

Pompe péristaltique

Vis d'Archimède

=> mouvement de translation

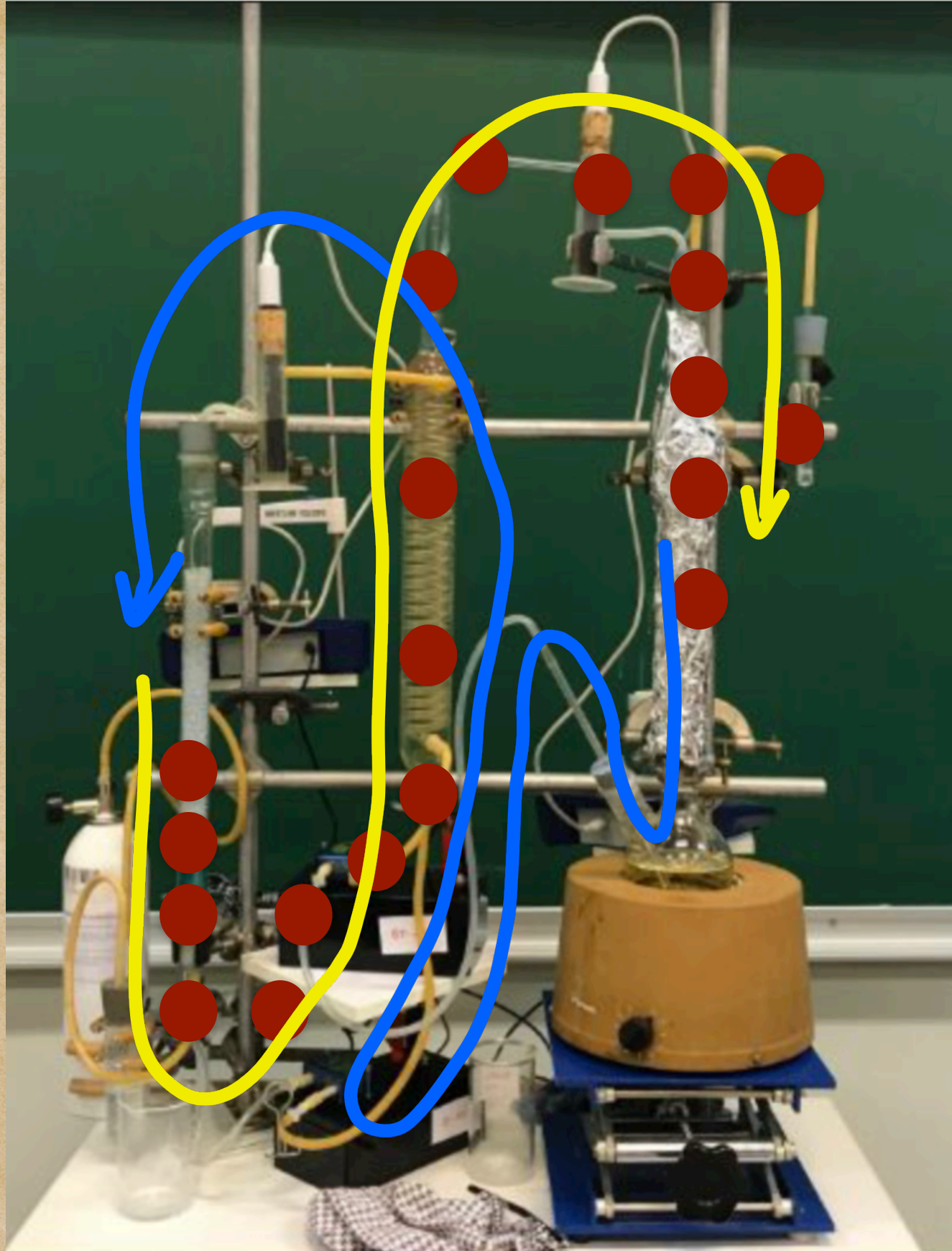
=> banc motorisé

Pompe péristaltique



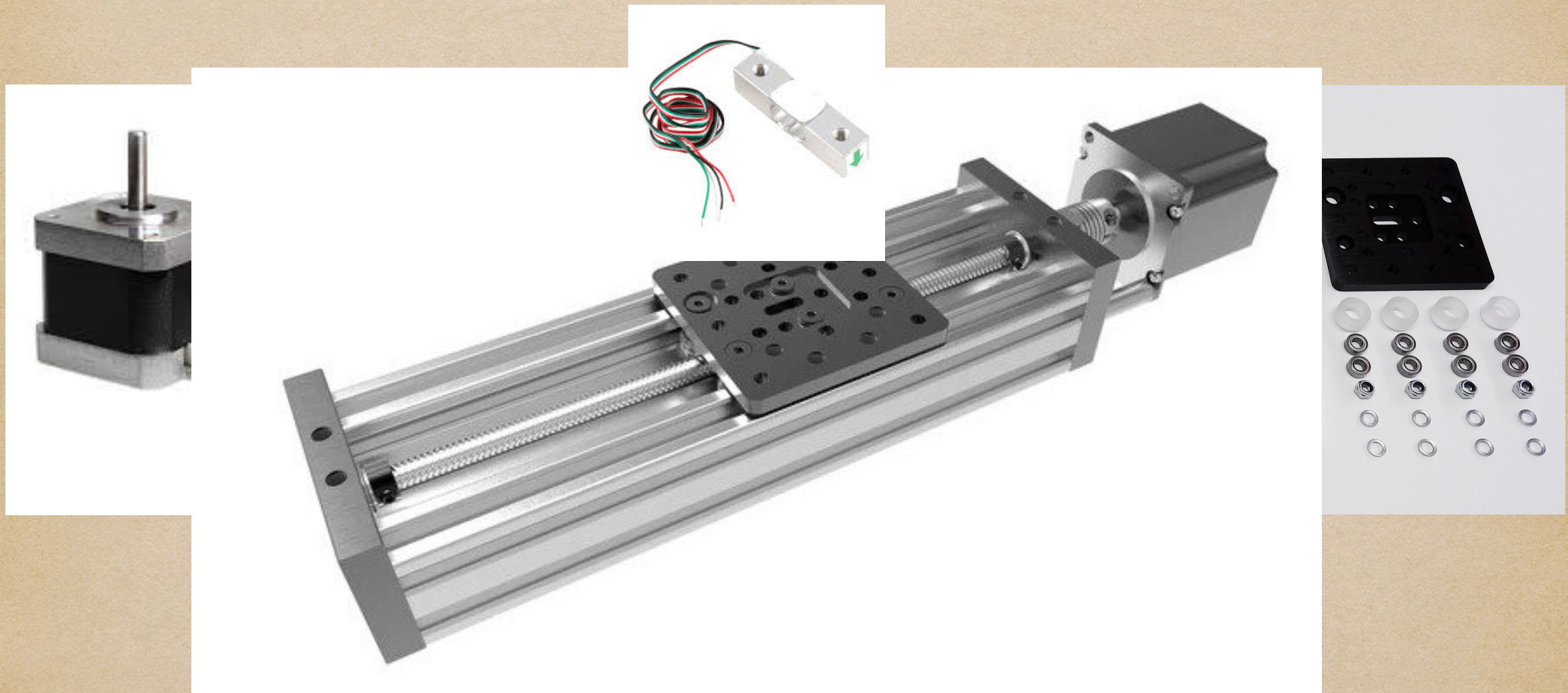


Extraction
continue
liquide-liquide



Modélisation
unité captage
de CO₂

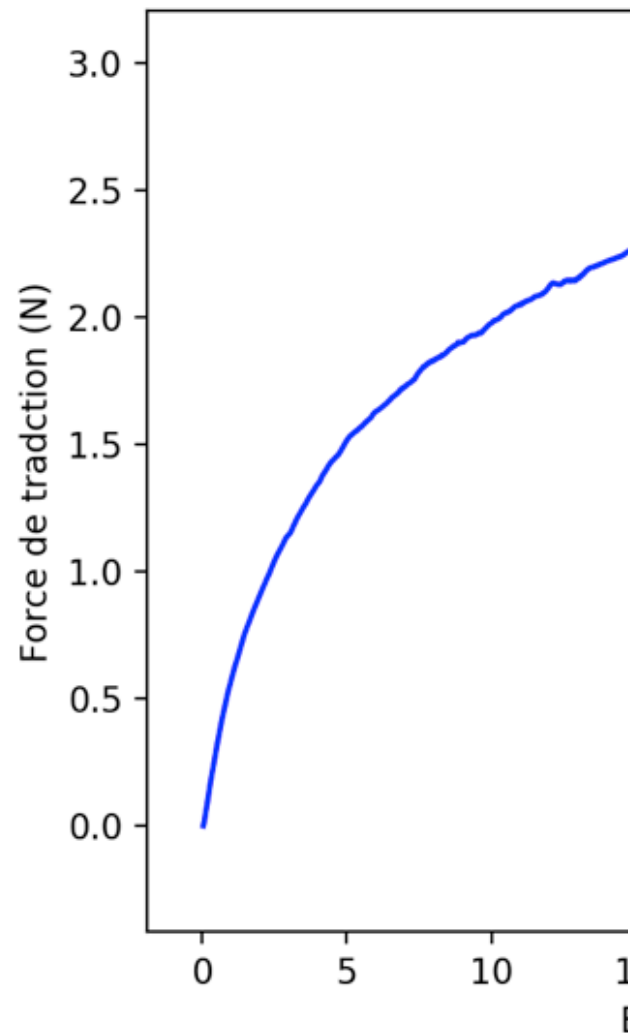
Banc de translation



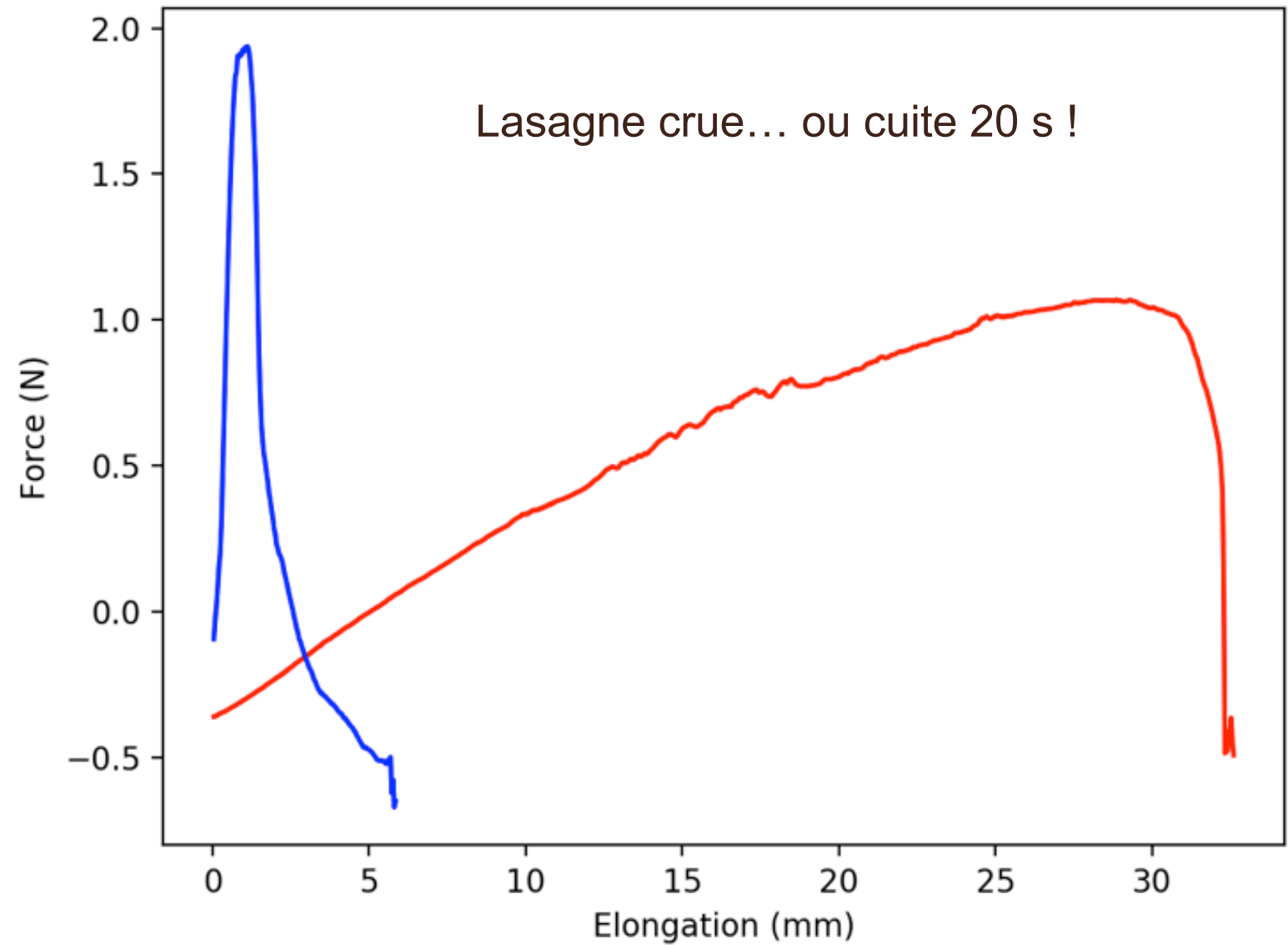
On commande le moteur pas à pas :
200 pas => 1 tour => translation de 8 mm

Banc de traction

Courbe de traction d'un bioplastique



Lasagne crue... ou cuite 20 s !



On mesure

Ce n'est pas toujours le pays des Bisounours !

Dans la vie au lycée, l'intérêt pédagogique

- Activité d'apprentissage et envahissante !
- Relative fragilité des élèves
- Maintien de la « motivation »
- « Rigidité » de l'encadrement

Coût du matériel

Encadrement des élèves

«Yapluka!»