

# ARDUINO

## C'est quoi donc ?



**Julien VILLEMEJANE**



Paris-Saclay



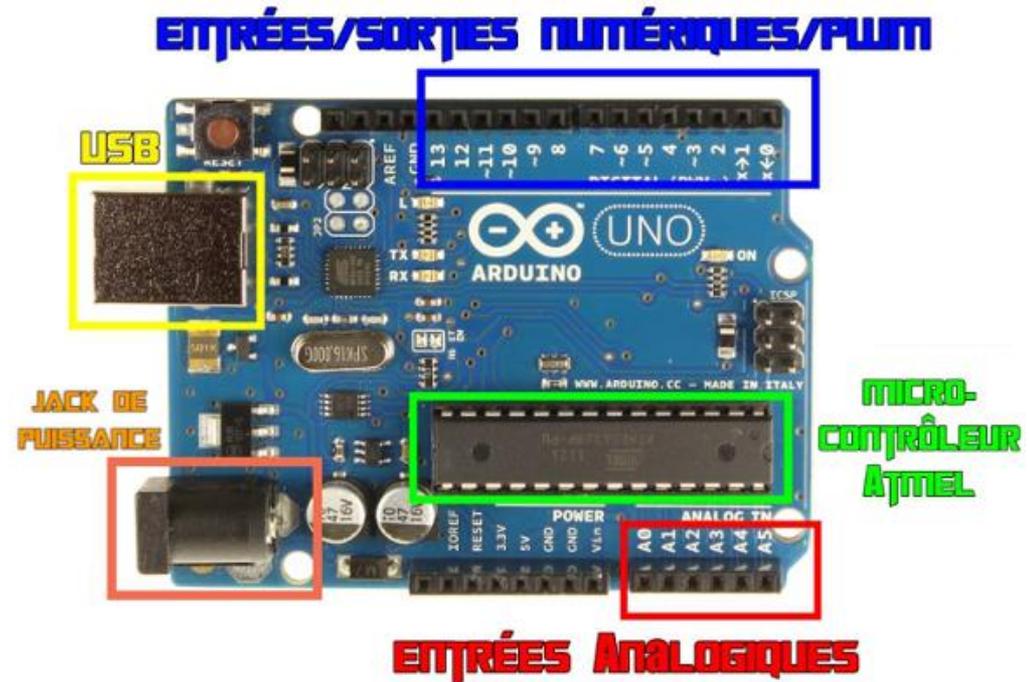
Saint-Étienne



Bordeaux

# Qu'est-ce qu'une carte Arduino ?

- Microcontrôleur
- Alimentation
- Téléversement
- Entrées / Sorties



Paris-Saclay



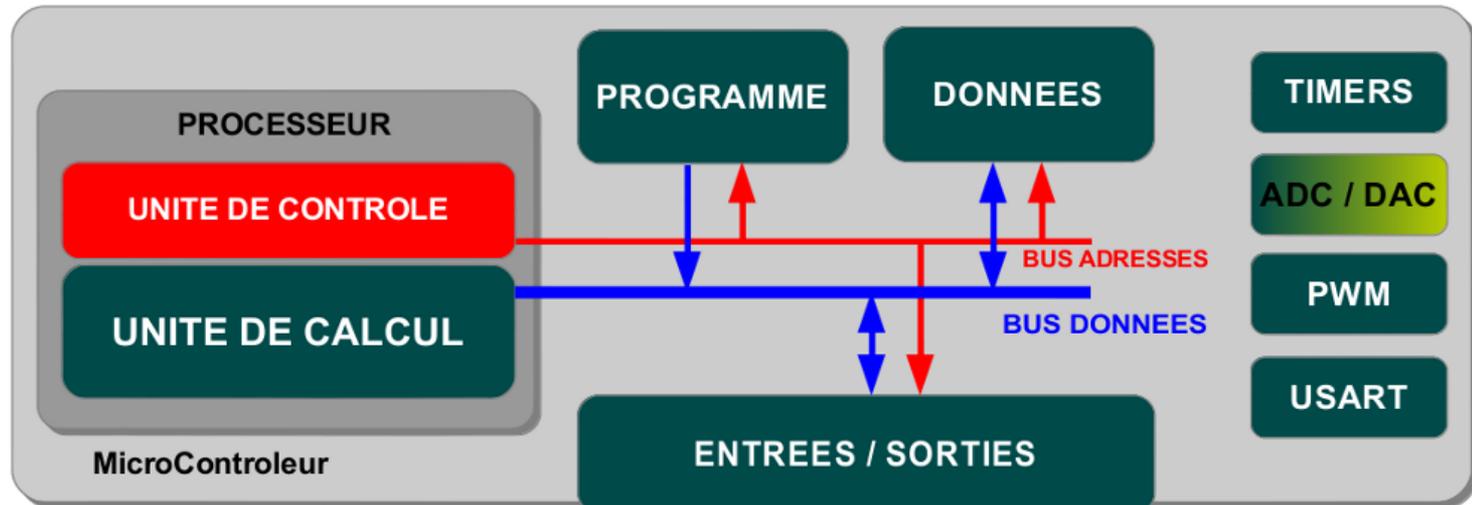
Saint-Étienne



Bordeaux

# Qu'est-ce qu'un microcontrôleur ?

- Processeur spécialisé
- Entrées / Sorties réactives



Paris-Saclay



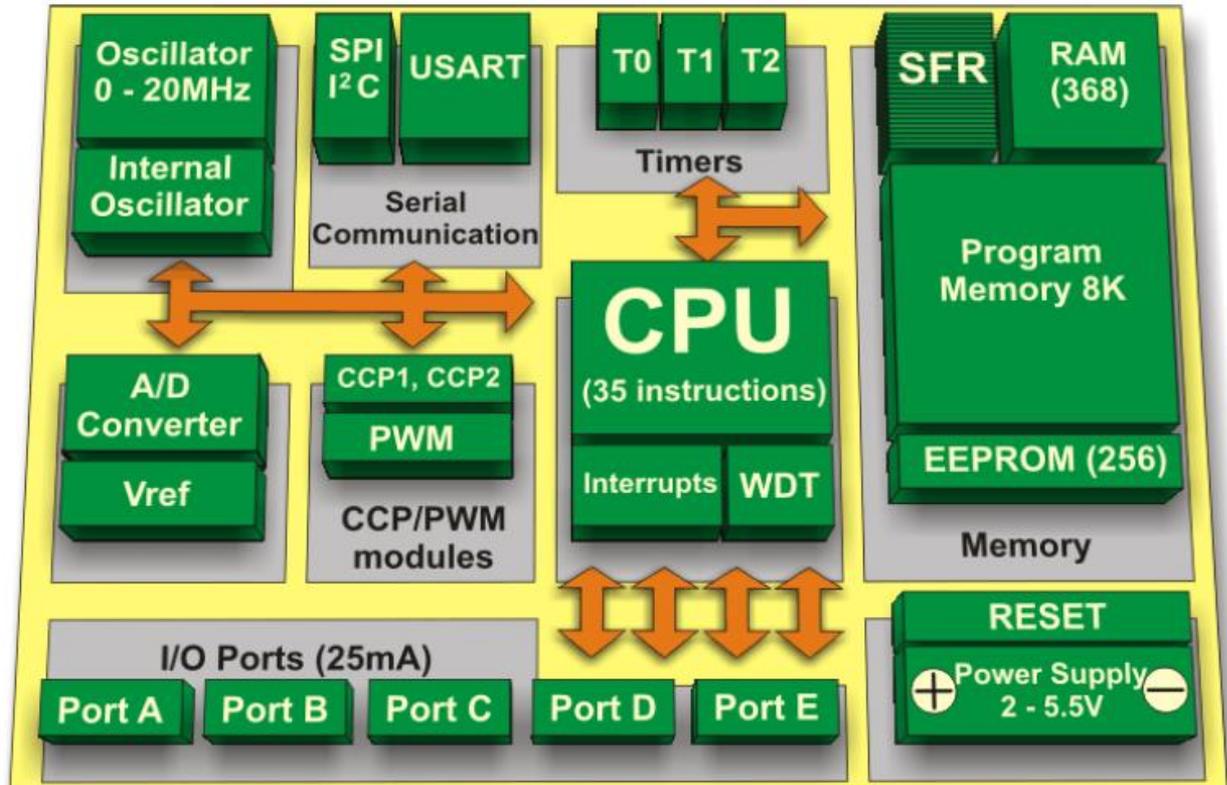
Saint-Étienne



Bordeaux

# Qu'est-ce qu'un microcontrôleur ?

- Processeur spécialisé
- Entrées / Sorties réactives



<http://learn.mikroe.com/> - PIC16F887



Paris-Saclay



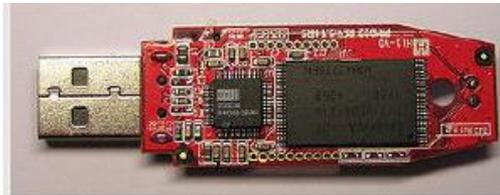
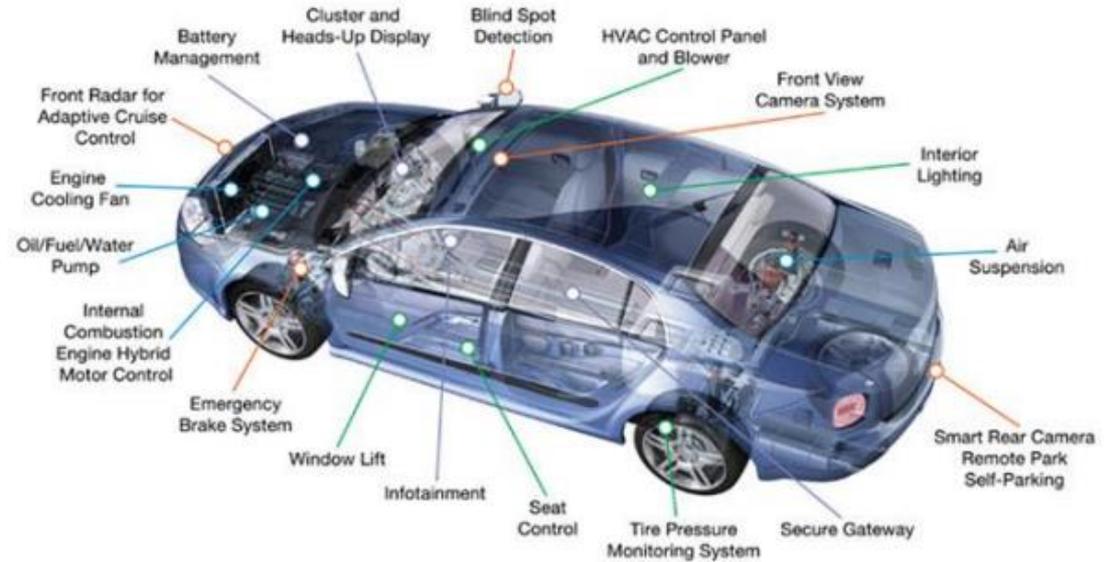
Saint-Étienne



Bordeaux

# Où sont-ils utilisés ?

- Système embarqué
- Capteurs intelligents



Paris-Saclay



Saint-Étienne

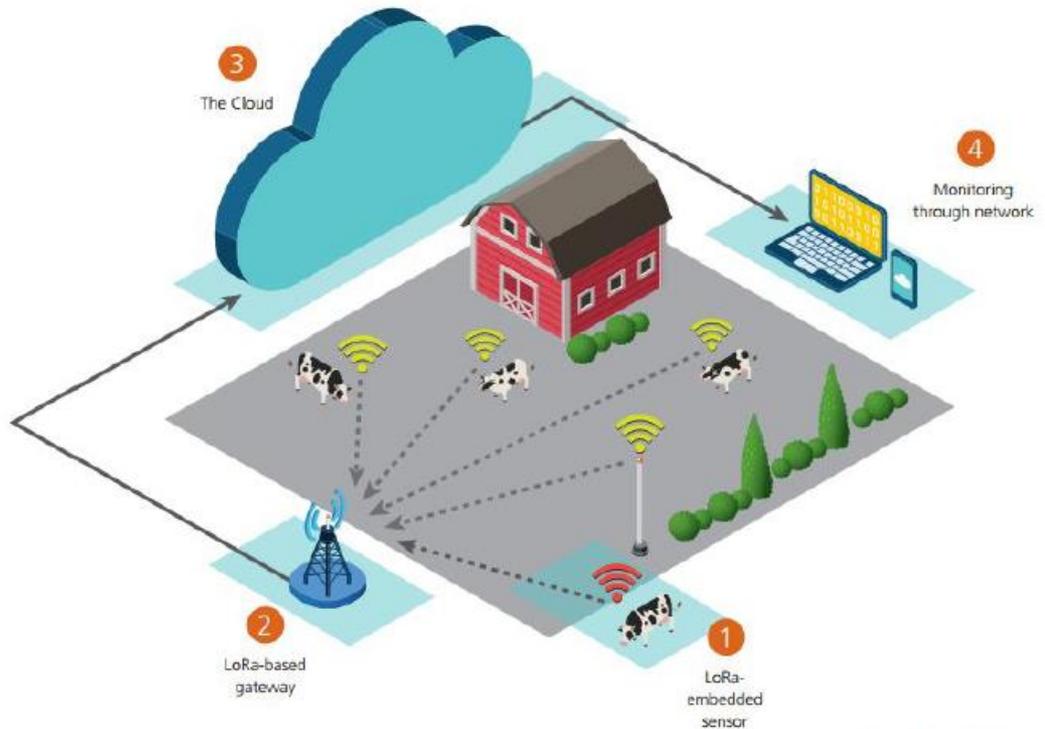


Bordeaux

# Où sont-ils utilisés ?

- Système embarqué
- Capteurs intelligents

## LoRA - LoNG RaNGE WIRELESS PLATFORMS



<http://www.semtech.com>



Paris-Saclay



Saint-Étienne

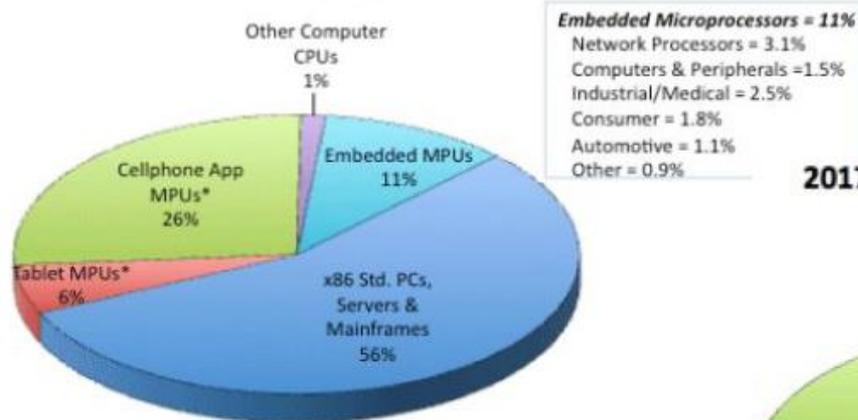


Bordeaux

# Et par rapport à un PC ?

- Processeurs généralistes vs Processeurs embarqués

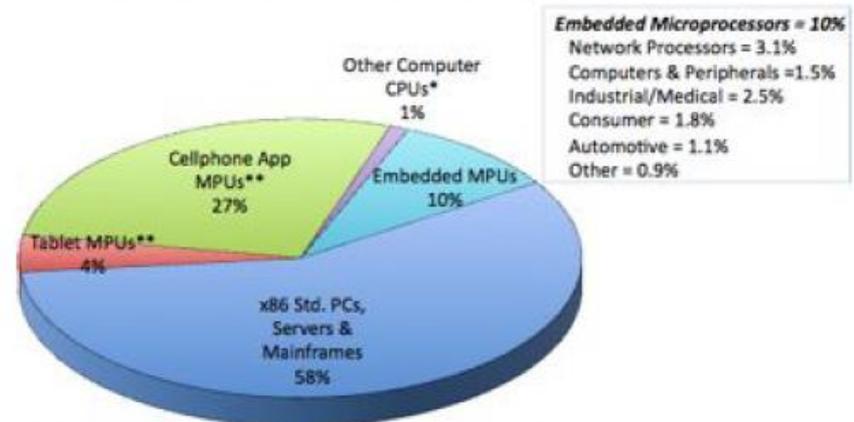
2013 MPU Sales by Applications (Fcst, \$61.0B)



\*Includes ARM-based and x86 processors.

Source: IC Insights

2017 MPU Sales by Applications (Fcst, \$66.6B)



\*Covers ARM and other RISC MPUs in servers and workstations.

\*\*Includes ARM and x86 mobile application processors.

Source: IC Insights



Paris-Saclay



Saint-Étienne



Bordeaux

# Et par rapport à un PC ?

- Processeurs généralistes vs Processeurs embarqués

	PC standard	ARDUINO Uno
Fréquence	2 GHz	16 MHz
Core / Architecture	4 / 64 bits	1 / 8 bits
Consommation	100 à 500 W	< 1 W
Entrées/Sorties	/	6 Analog / 13 Digital
Ports extension	USB, PCI...	SPI, I2C, RS232
RAM	4 Go	2 ko



Paris-Saclay



Saint-Étienne



Bordeaux

- Processeurs généralistes vs Processeurs embarqués

	PC standard	ARDUINO Uno
Fréquence	2 GHz	16 MHz
Core / Architecture	4 / 64 bits	1 / 8 bits
Consommation	100 à 500 W	< 1 W
Entrées/Sorties	/	6 Analog / 13 Digital
Ports extension	USB, PCI...	SPI, I2C, RS232
RAM	4 Go	2 ko

- Et combien de temps met une application à se lancer sur un PC ?
- Est-ce une durée fixe ?



Paris-Saclay



Saint-Étienne



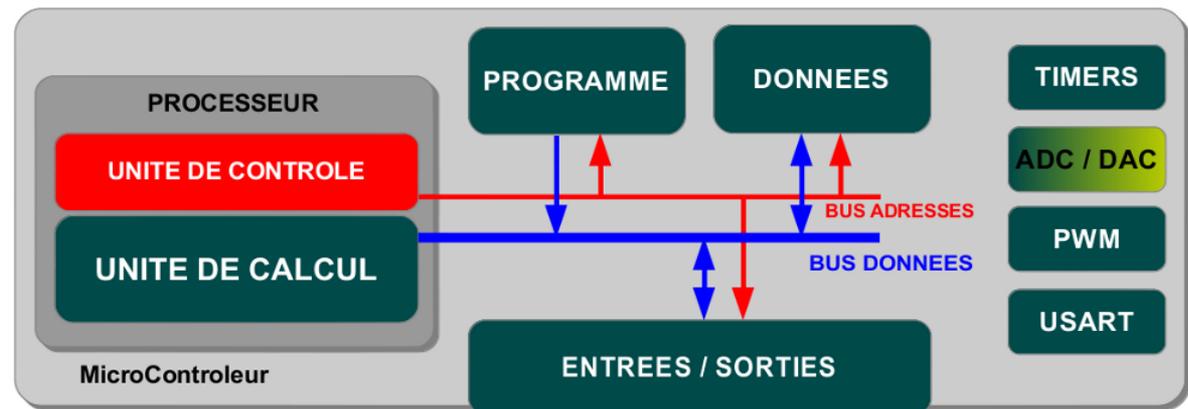
Bordeaux

# Revenons à Arduino

- Accès à des entrées/sorties réactives



ARDUINO Uno
16 MHz
1 core / 8 bits
< 1 W
6 Analog / 13 Digital
SPI, I2C, RS232
2 ko



- Grande communauté
- Shields (extensions)



Paris-Saclay



Saint-Étienne



Bordeaux

# Comment ça marche une carte Arduino ?

- Programme
- IDE = Interface de Développement
- Téléversement



```

Blink | Arduino 1.0
File Edit Sketch Tools Help

Blink
Turns on an LED on for one second, then off for one second, repeats.

This example code is in the public domain.
*/

void setup() {
  // initialize the digital pin as an output.
  // Pin 13 has an LED connected on most Arduino boards:
  pinMode(13, OUTPUT);
}

void loop() {
  digitalWrite(13, HIGH); // set the LED on
  delay(1000);           // wait for a second
  digitalWrite(13, LOW); // set the LED off
  delay(1000);           // wait for a second
}
    
```

Arduino Uno on /dev/ttyACM1

```

PROGRAMINO IDE FOR ARDUINO & GENUINO 1.1.0.1
File Edit Options View Sketch Hardware Web Tools Help

Project Explorer
C:\Program Files (x86)\PROGRAM
Terminal.ino

Terminal.ino | new_sketch.ino
10
11
12 long cnt = 0;
13
14
15 void setup()
16 {
17   // Set baudrate to 19200 baud
18   Serial.begin(19200);
19 }
20
21
22
23
24
25
26 void loop()
27 {
28
29   // Read the analog input A0
30   Serial.print(analogRead(A0));
31   Serial.print(" - ");
32   Serial.println(cnt);
33   delay(200);
34   cnt++;
35
36   // Save data after 50 measurements into the Logfile
37   if(cnt==50)
38   {
39     Serial.println("[#SAVE#Logfile.txt]");
40   }
41
}

Object Explorer
void setup()
void loop()

Arduino Messages Search Results
Verify code please wait...
Last verify at 09:08:37
Sketch uses 2,620 bytes (8%) of program storage space. Maximum is 32,256 bytes.

Caps Lock Disabled Line: 26 of 51 Comport: (COM3) Arduino Mega 2560 Board: Arduino Uno
    
```



Paris-Saclay



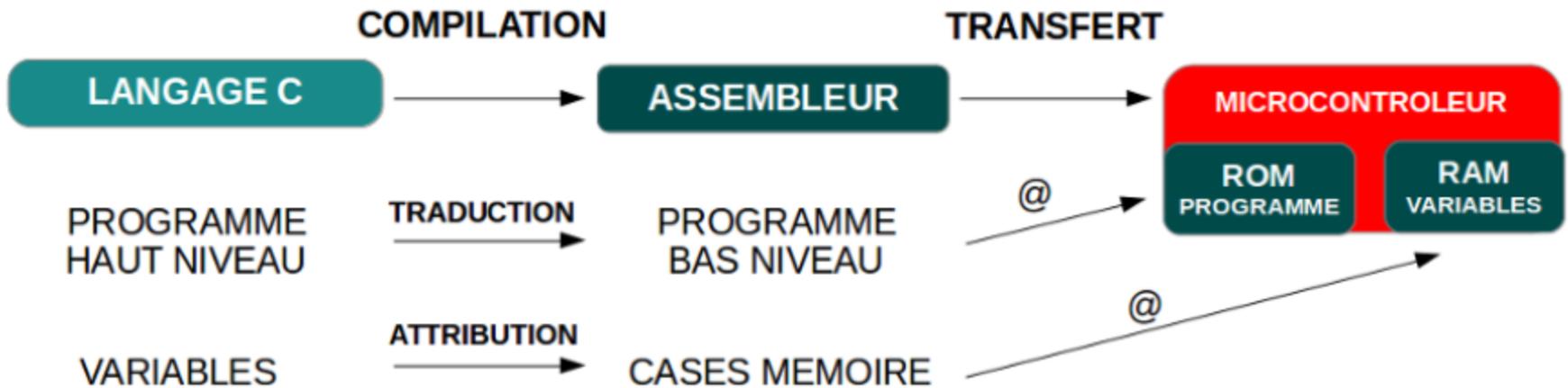
Saint-Étienne



Bordeaux

# Comment ça marche une carte Arduino ?

- Programme
- IDE = Interface de Développement
- Téléversement



Paris-Saclay



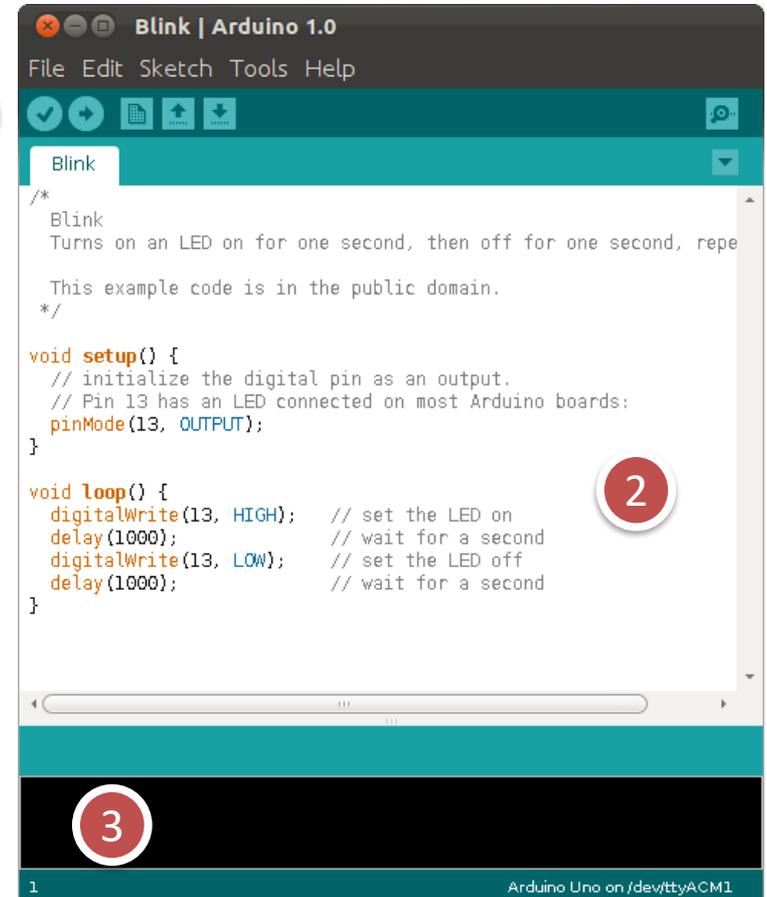
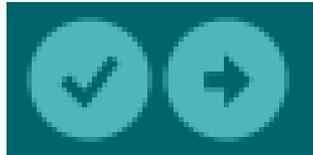
Saint-Étienne



Bordeaux

# Mon premier programme Arduino

- Prise en main interface
- Connexion USB
- Programme **Blink**



```
File Edit Sketch Tools Help
Blink
/*
 * Blink
 * Turns on an LED on for one second, then off for one second, repeatedly.
 *
 * This example code is in the public domain.
 */

void setup() {
  // initialize the digital pin as an output.
  // Pin 13 has an LED connected on most Arduino boards:
  pinMode(13, OUTPUT);
}

void loop() {
  digitalWrite(13, HIGH); // set the LED on
  delay(1000);           // wait for a second
  digitalWrite(13, LOW); // set the LED off
  delay(1000);           // wait for a second
}
```

1

2

3

Arduino Uno on /dev/ttyACM1



Paris-Saclay



Saint-Étienne



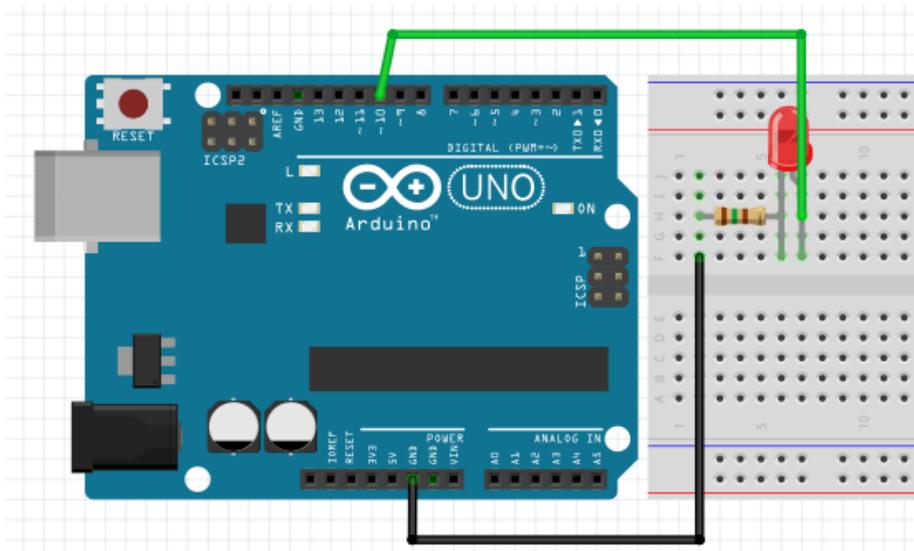
Bordeaux

# Piloter une sortie externe

- Cablage d'une LED sur D10

$$R_{LED} = \frac{V_{ARDUINO} - V_{SEUIL}}{I_{MAX}}$$

- Programme *Blink* modifié



Blink \$

```
void setup() {
  pinMode(10, OUTPUT);
}
void loop() {
  digitalWrite(10, HIGH);
  delay(1000);
  digitalWrite(10, LOW);
  delay(1000);
}
```



Paris-Saclay



Saint-Étienne

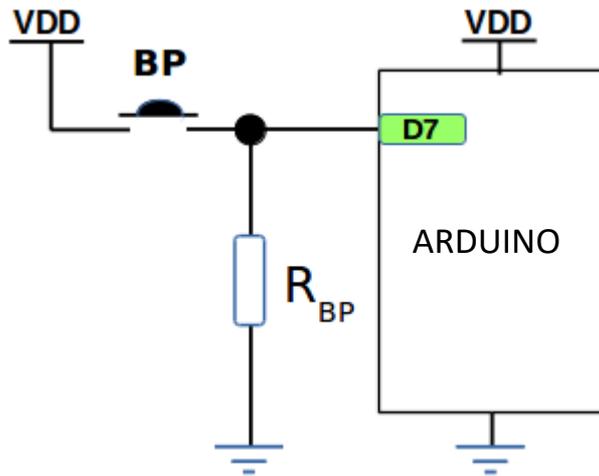


Bordeaux

# Câbler un bouton poussoir

## Récupérer une donnée numérique

- $R_{BP}$  pour protéger l'alimentation



Blink §

```
int val = 0;

void setup() {
  pinMode(13, OUTPUT);
  pinMode(7, OUTPUT);
}

void loop() {
  val = digitalRead(7);
  digitalWrite(13, val);
}
```



Paris-Saclay



Saint-Étienne



Bordeaux

# Récupérer une donnée analogique

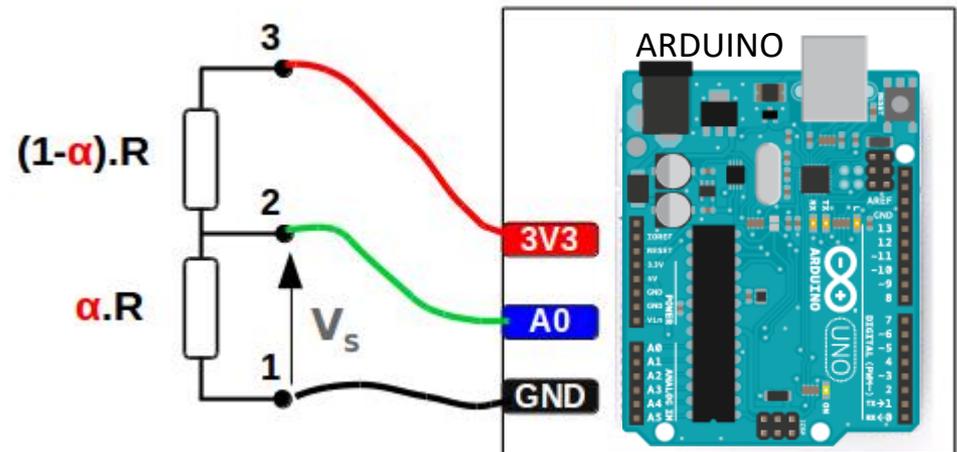
- Câblage potentiomètre

AnalogReadSerial §

```
void setup() {
}

void loop() {
  int sensorValue = analogRead(A0);

  delay(1);
}
```



- Comment vérifier que la donnée est bien convertie ?

- CAN intégré – 10 bits



Paris-Saclay



Saint-Étienne



Bordeaux

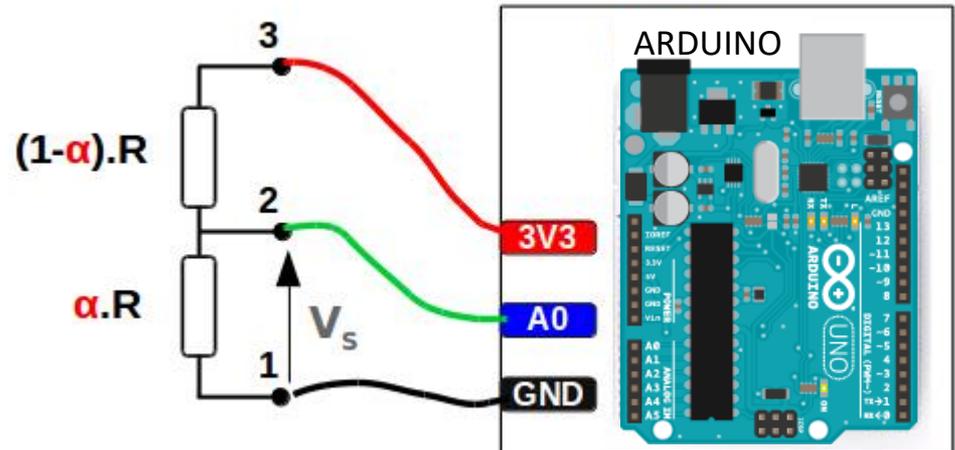
# Récupérer une donnée analogique

- Câblage potentiomètre
- Affichage console + Traceur / **Exemple : Basics/AnalogReadSerial**

## AnalogReadSerial §

```
void setup() {
  Serial.begin(9600);
}

void loop() {
  int sensorValue = analogRead(A0);
  Serial.println(sensorValue);
  delay(1);
}
```



- CAN intégré – 10 bits



Paris-Saclay



Saint-Étienne



Bordeaux

# Récupérer une donnée analogique

- Câblage potentiomètre
- **Affichage console + Traceur** / Exemple : Basics/AnalogReadSerial

AnalogReadSerial §

```
void setup() {  
  Serial.begin(9600);  
}  
  
void loop() {  
  int sensorValue = analogRead(A0);  
  Serial.println(sensorValue);  
  delay(1);  
}
```



Paris-Saclay



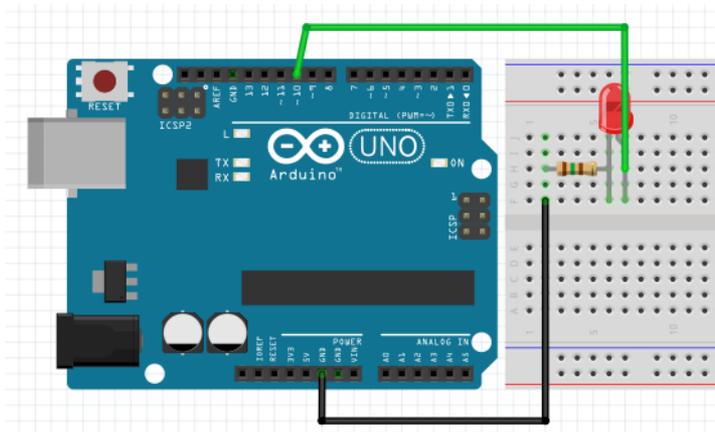
Saint-Étienne



Bordeaux

# Contrôler la luminosité d'une LED

- Modulation de largeur d'impulsions
- Exemple : Basics/Fade



- Quel est le signal appliqué sur la LED ?

## Fade §

```
int led = 9;           // the PWM pin the LED is attached to
int brightness = 0;   // how bright the LED is
int fadeAmount = 5;   // how many points to fade the LED by

// the setup routine runs once when you press reset:
void setup() {
  // declare pin 9 to be an output:
  pinMode(led, OUTPUT);
}

// the loop routine runs over and over again forever:
void loop() {
  // set the brightness of pin 9:
  analogWrite(led, brightness);

  // change the brightness for next time through the loop:
  brightness = brightness + fadeAmount;

  // reverse the direction of the fading at the ends of the fade:
  if (brightness <= 0 || brightness >= 255) {
    fadeAmount = -fadeAmount;
  }
  // wait for 30 milliseconds to see the dimming effect
  delay(30);
}
```



Paris-Saclay

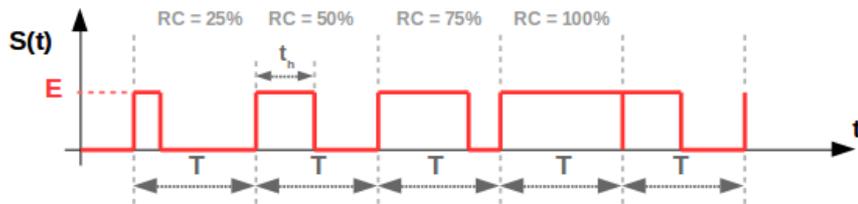
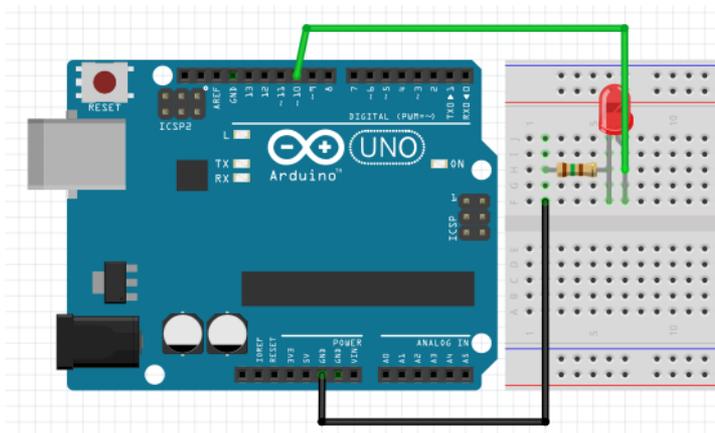


Saint-Étienne



Bordeaux

- Modulation de largeur d'impulsions
- Exemple : Basics/Fade



## Fade §

```
int led = 9;           // the PWM pin the LED is attached to
int brightness = 0;   // how bright the LED is
int fadeAmount = 5;   // how many points to fade the LED by

// the setup routine runs once when you press reset:
void setup() {
  // declare pin 9 to be an output:
  pinMode(led, OUTPUT);
}

// the loop routine runs over and over again forever:
void loop() {
  // set the brightness of pin 9:
  analogWrite(led, brightness);

  // change the brightness for next time through the loop:
  brightness = brightness + fadeAmount;

  // reverse the direction of the fading at the ends of the fade:
  if (brightness <= 0 || brightness >= 255) {
    fadeAmount = -fadeAmount;
  }
  // wait for 30 milliseconds to see the dimming effect
  delay(30);
}
```



Paris-Saclay



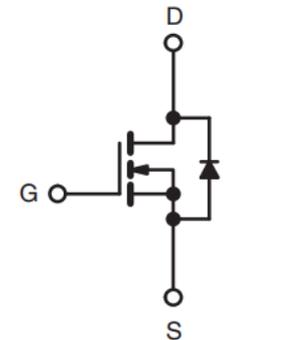
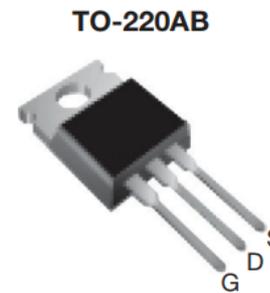
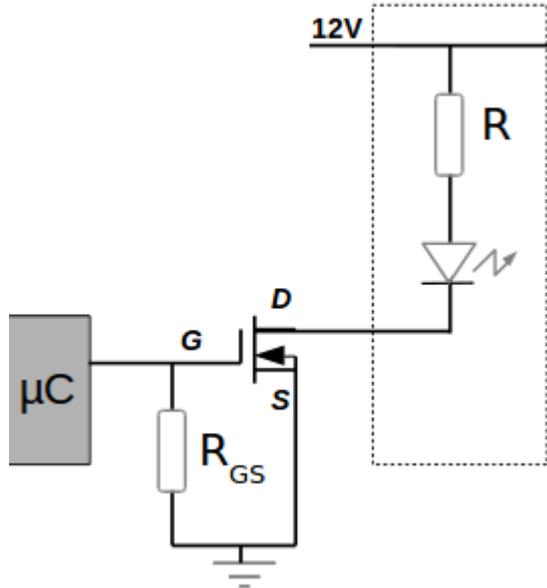
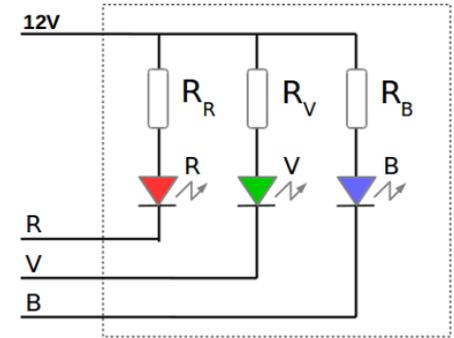
Saint-Étienne



Bordeaux

# Piloter un bandeau de LEDs

- Transistor IRL540 ou BS170
- Bandeau de LED



N-Channel MOSFET



Paris-Saclay



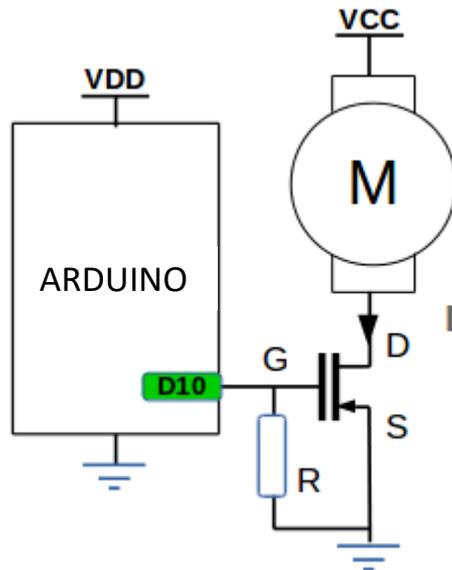
Saint-Étienne



Bordeaux

# Piloter un moteur à courant continu

- Une seule direction / Transistor



- Et pour changer de sens de rotation ?

## Fade §

```
int led = 9;           // the PWM pin the LED is attached to
int brightness = 0;   // how bright the LED is
int fadeAmount = 5;   // how many points to fade the LED by

// the setup routine runs once when you press reset:
void setup() {
  // declare pin 9 to be an output:
  pinMode(led, OUTPUT);
}

// the loop routine runs over and over again forever:
void loop() {
  // set the brightness of pin 9:
  analogWrite(led, brightness);

  // change the brightness for next time through the loop:
  brightness = brightness + fadeAmount;

  // reverse the direction of the fading at the ends of the fade:
  if (brightness <= 0 || brightness >= 255) {
    fadeAmount = -fadeAmount;
  }
  // wait for 30 milliseconds to see the dimming effect
  delay(30);
}
```



Paris-Saclay



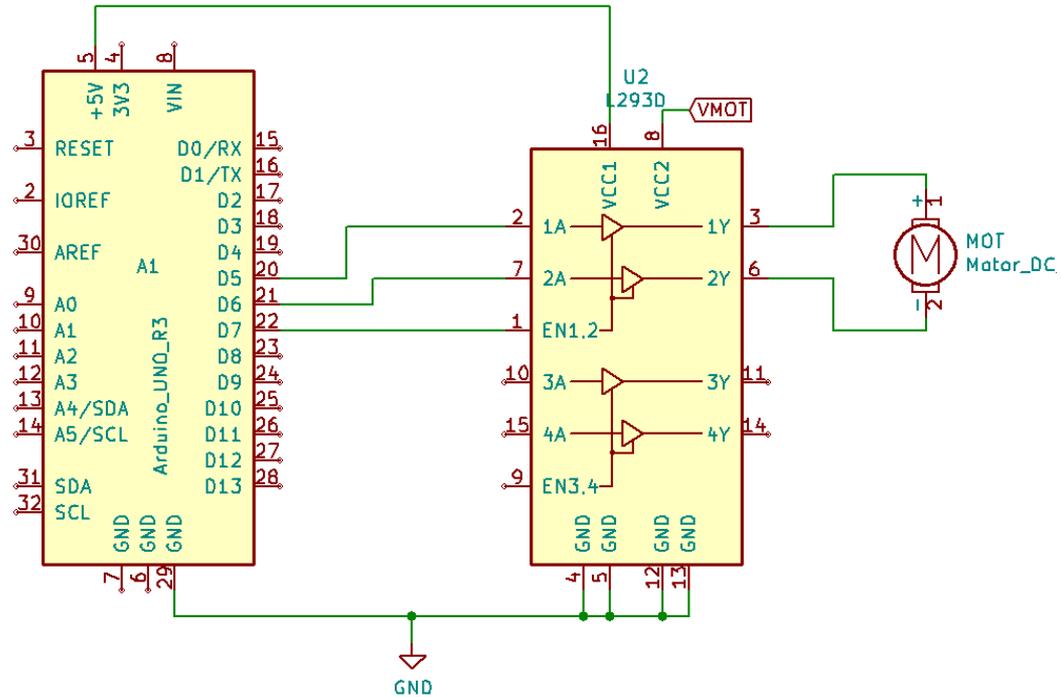
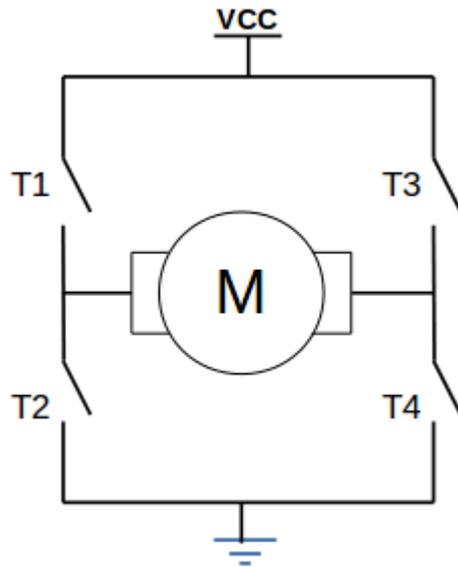
Saint-Étienne



Bordeaux

# Piloter un moteur à courant continu

- Deux directions / Pont en H / L293D



Paris-Saclay



Saint-Étienne



Bordeaux

# Piloter un moteur à courant continu

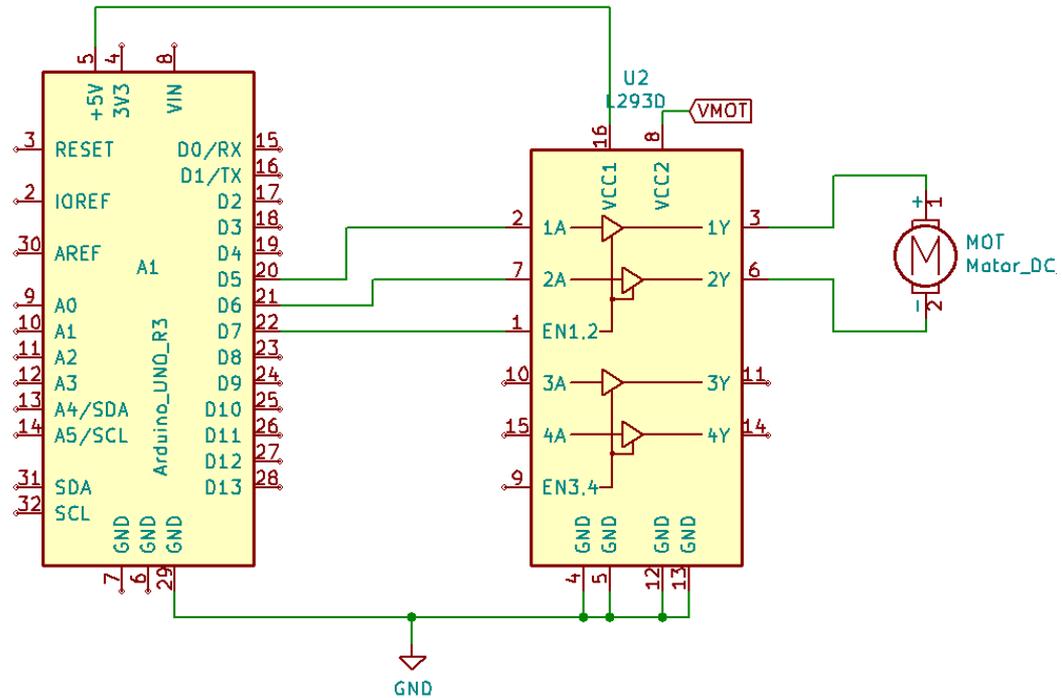
- Deux directions / Pont en H / L293D

MCC8PWM

```

if(etat_activation == 1){
    val_pot = analogRead(A0);
    rapport_cyclique = map(val_pot, 0, 1023, -255, 255);
    if(rapport_cyclique < 0){
        digitalWrite(enableAB, HIGH);
        analogWrite(moteurA, -rapport_cyclique);
        analogWrite(moteurB, 0);
    }
    else{
        digitalWrite(enableAB, HIGH);
        analogWrite(moteurA, 0);
        analogWrite(moteurB, rapport_cyclique);
    }
}
else{
    digitalWrite(enableAB, LOW);
    analogWrite(moteurA, 0);
    analogWrite(moteurB, 0);
}
}

```



Paris-Saclay



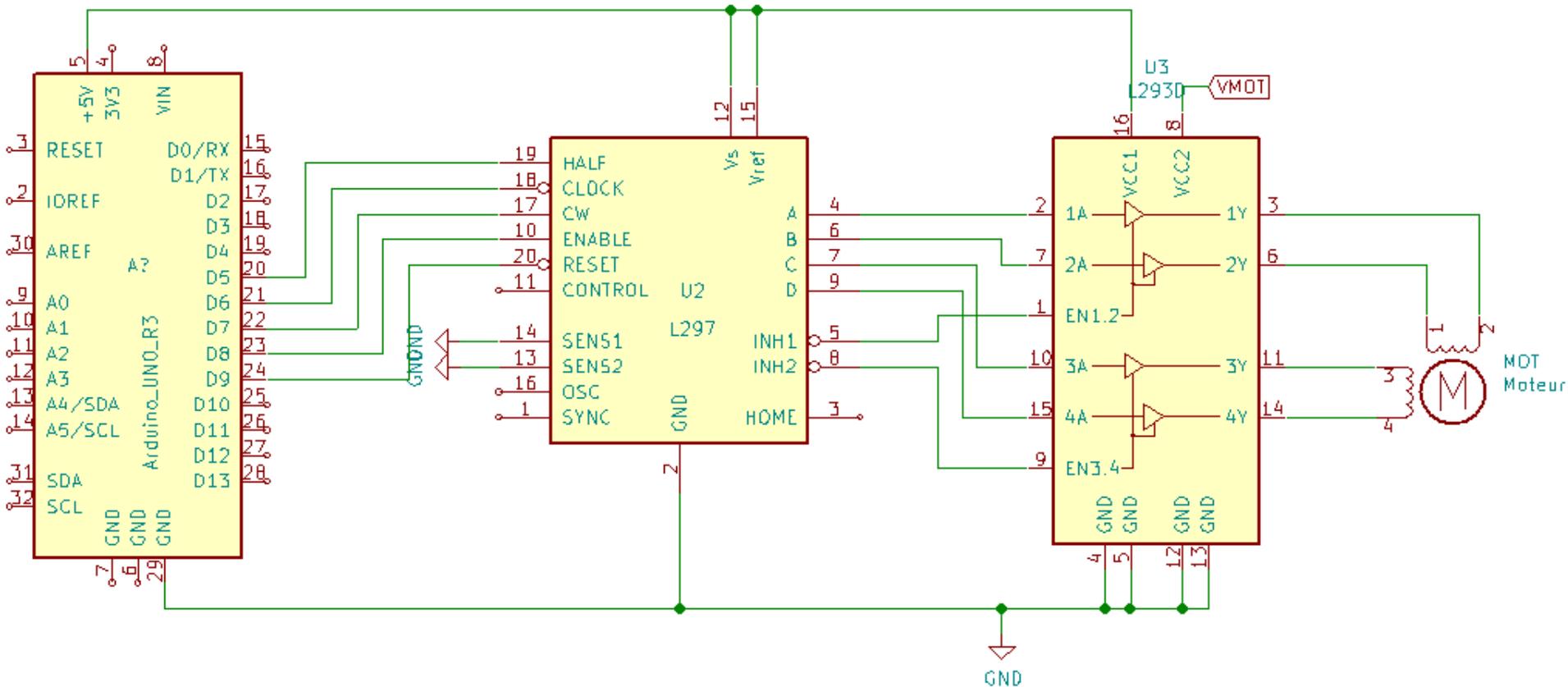
Saint-Étienne



Bordeaux

# Piloter un moteur pas à pas

- Pont en H / L293D
- Driver / L297



Paris-Saclay



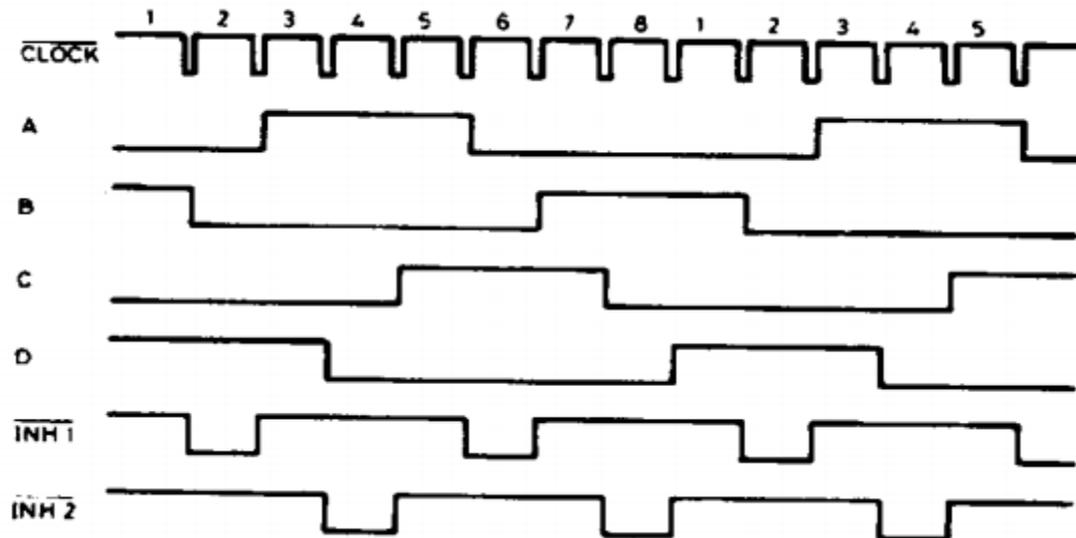
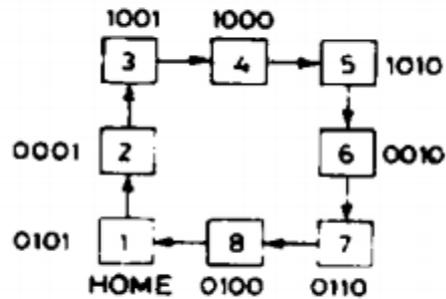
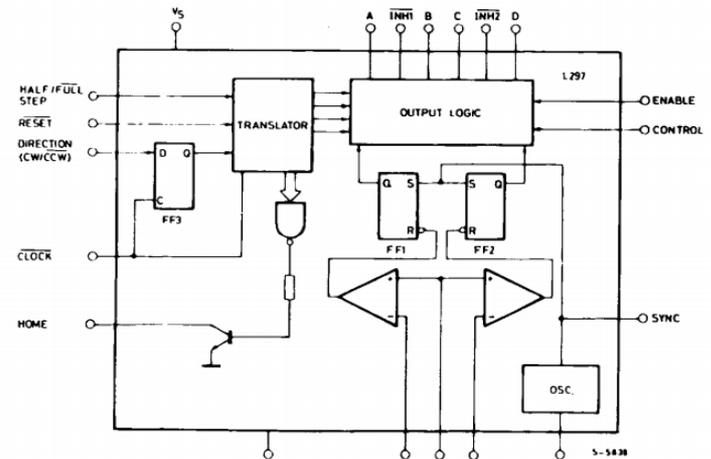
Saint-Étienne



Bordeaux

# Piloter un moteur pas à pas

- Pont en H / L293D
- Driver / L297
- Commande en Demi-pas



Paris-Saclay



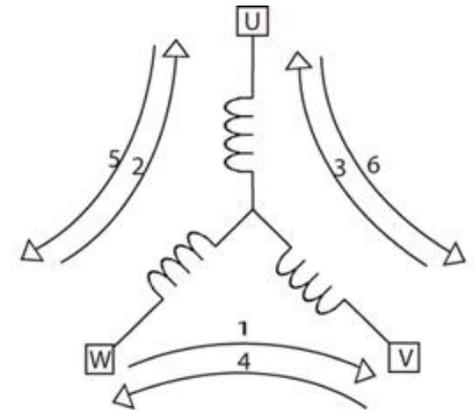
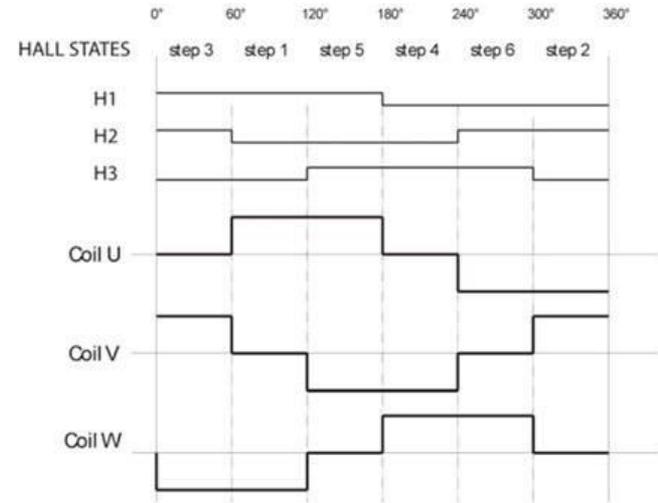
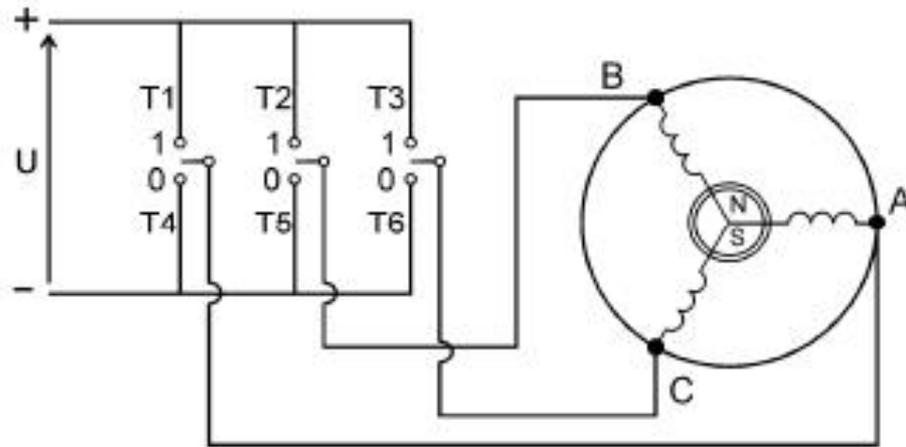
Saint-Étienne



Bordeaux

# Piloter un moteur *brushless*

- Transistors



<http://www.energoelektronika.pl/do/ShowNews?id=1599>

<https://www.digikey.com/en/articles/techzone/2013/mar/an-introduction-to-brushless-dc-motor-control>



Paris-Saclay



Saint-Étienne



Bordeaux

# Piloter un moteur *brushless*

- Pont en H type L293D

## Commande\_BLDC\_TEST

case 1:

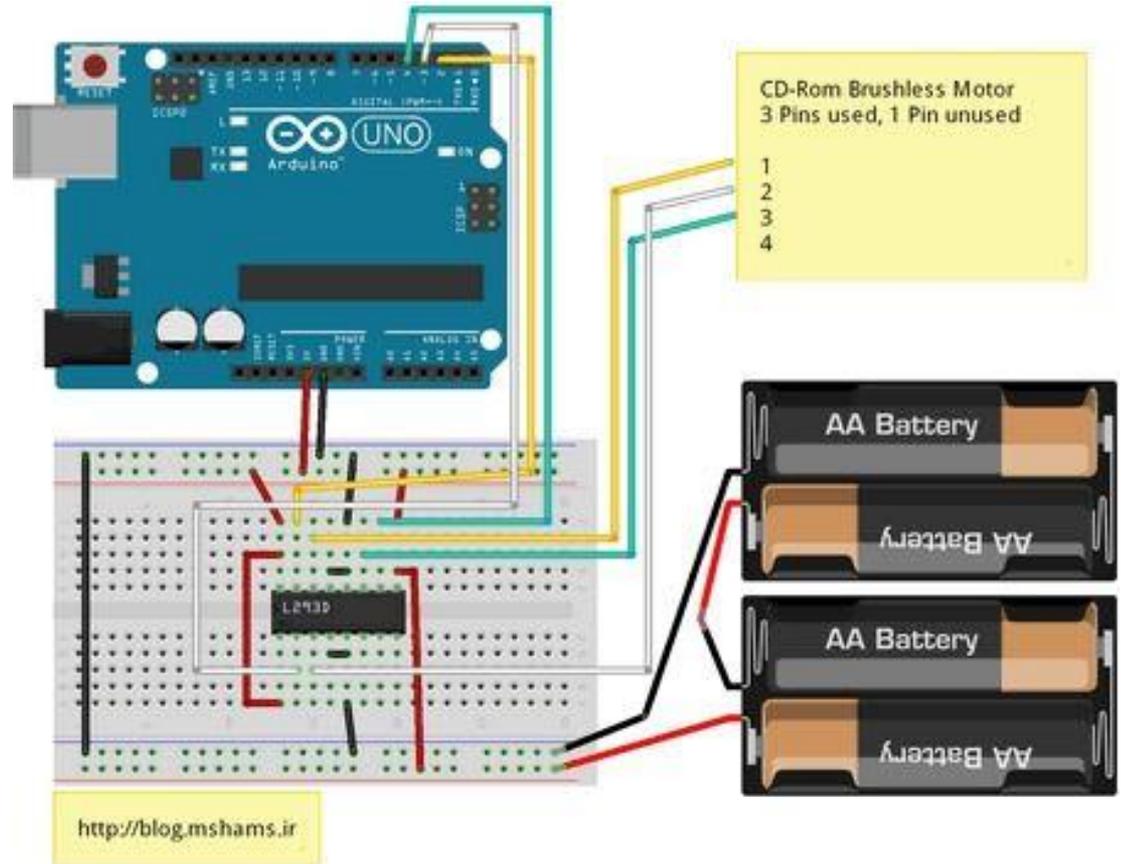
```
digitalWrite(phase1, HIGH);
digitalWrite(phase2, LOW);
digitalWrite(phase3, LOW);
myDelay(stepLength);
break;
```

case 2:

```
digitalWrite(phase1, LOW);
digitalWrite(phase2, HIGH);
digitalWrite(phase3, LOW);
myDelay(stepLength);
break;
```

default:

```
digitalWrite(phase1, LOW);
digitalWrite(phase2, LOW);
digitalWrite(phase3, HIGH);
myDelay(stepLength);
break;
```



Paris-Saclay



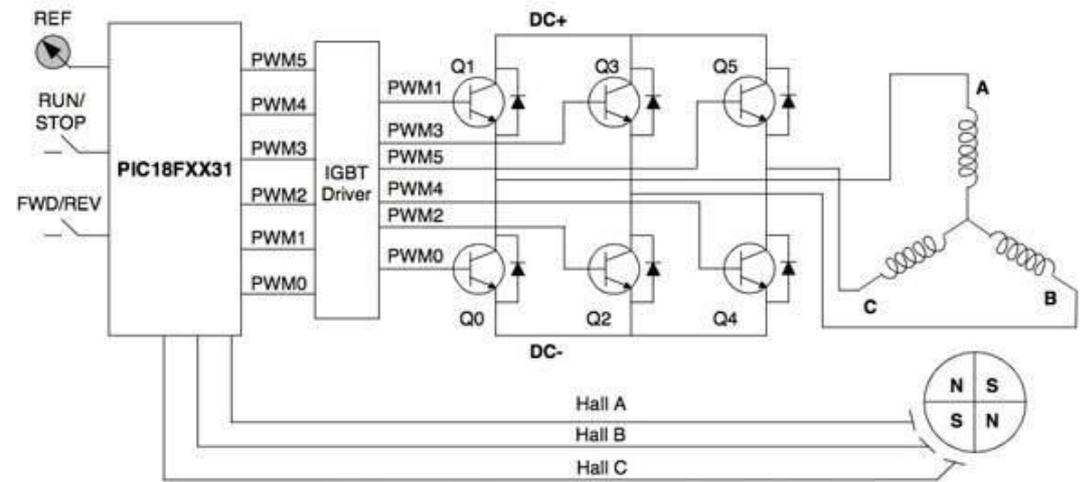
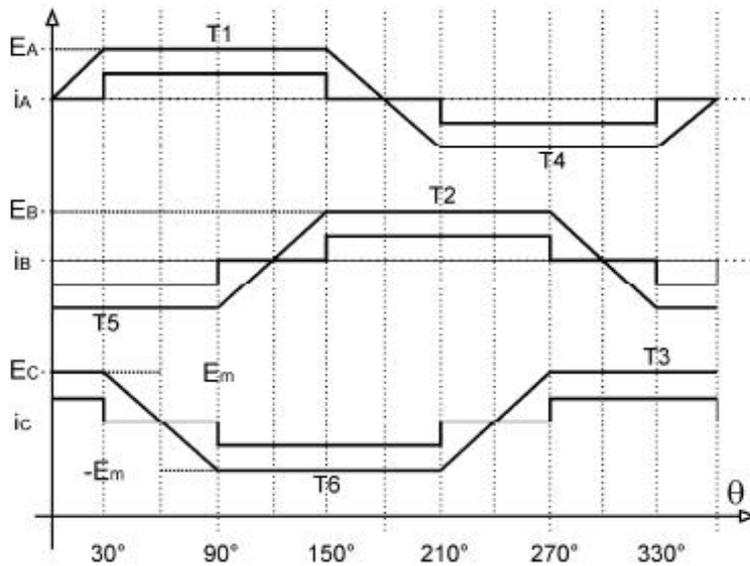
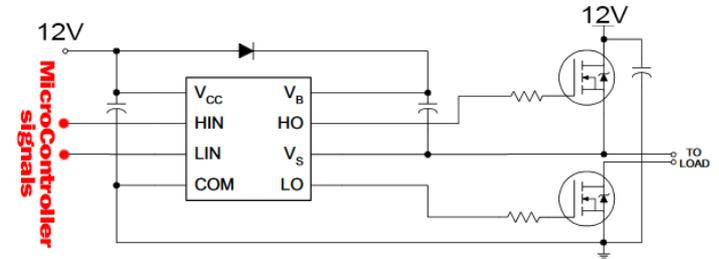
Saint-Étienne



Bordeaux

# Piloter un moteur *brushless*

- Avec drivers de MOS



<https://www.digikey.com/en/articles/techzone/2013/mar/an-introduction-to-brushless-dc-motor-control>

<http://www.energoelektronika.pl/do/ShowNews?id=1599>



Paris-Saclay



Saint-Étienne



Bordeaux

- Avec drivers de MOS

## Commande BLDC

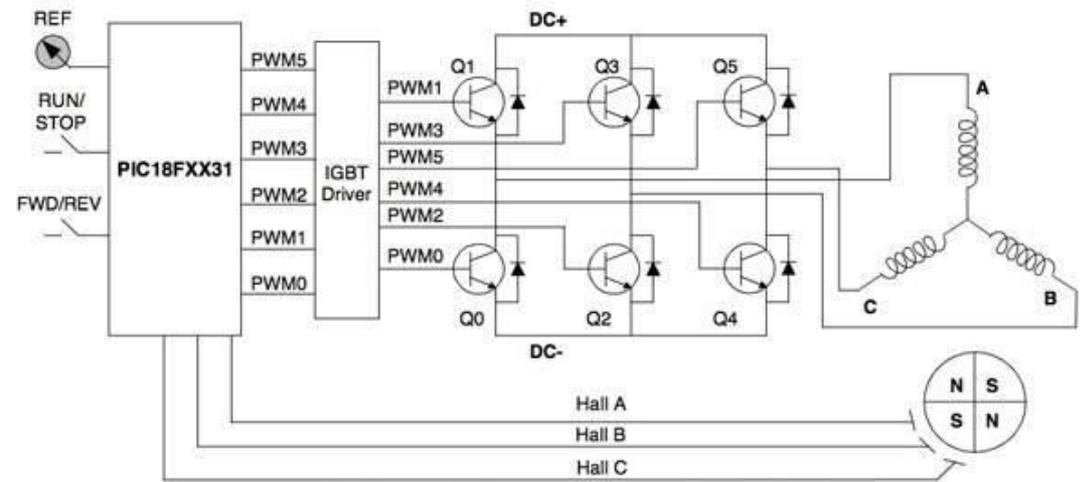
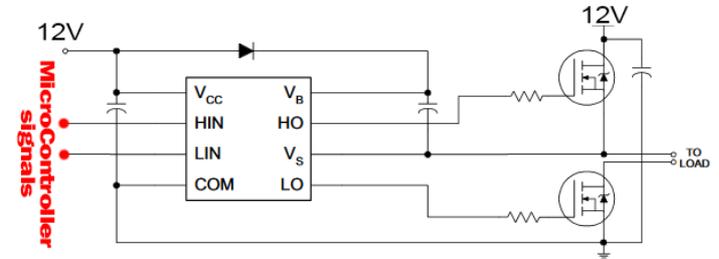
case 1:

```
digitalWrite(AA1, LOW);
digitalWrite(AA2, LOW);
digitalWrite(BB1, LOW);
digitalWrite(CC2, LOW);
digitalWrite(BB2, HIGH);
digitalWrite(CC1, HIGH);
delta = emA-sum;
break;
```

//Phase2 A-B

case 2:

```
digitalWrite(AA2, LOW);
digitalWrite(BB1, LOW);
digitalWrite(CC1, LOW);
digitalWrite(CC2, LOW);
digitalWrite(AA1, HIGH);
digitalWrite(BB2, HIGH);
delta = emC-sum;
```



<https://www.digikey.com/en/articles/techzone/2013/mar/an-introduction-to-brushless-dc-motor-control>

<http://www.energoelektronika.pl/do/ShowNews?id=1599>



Paris-Saclay



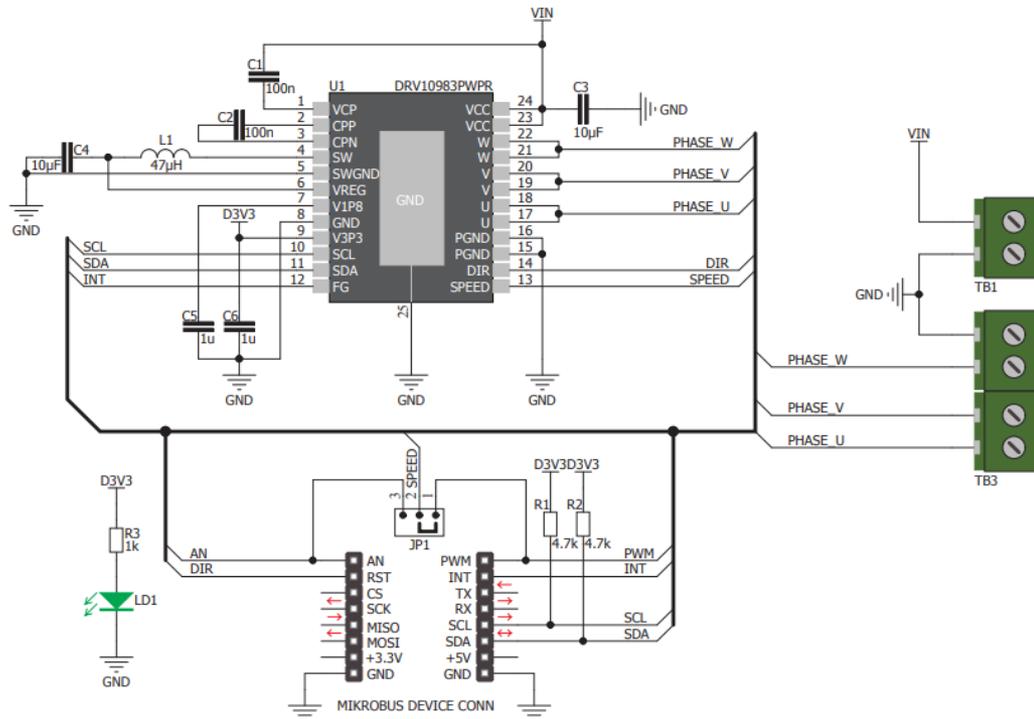
Saint-Étienne



Bordeaux

- Utilisation du module **Brushless 3 click** MikroE
- Commande en **I2C**

Brushless 3 click schematic v100



Paris-Saclay



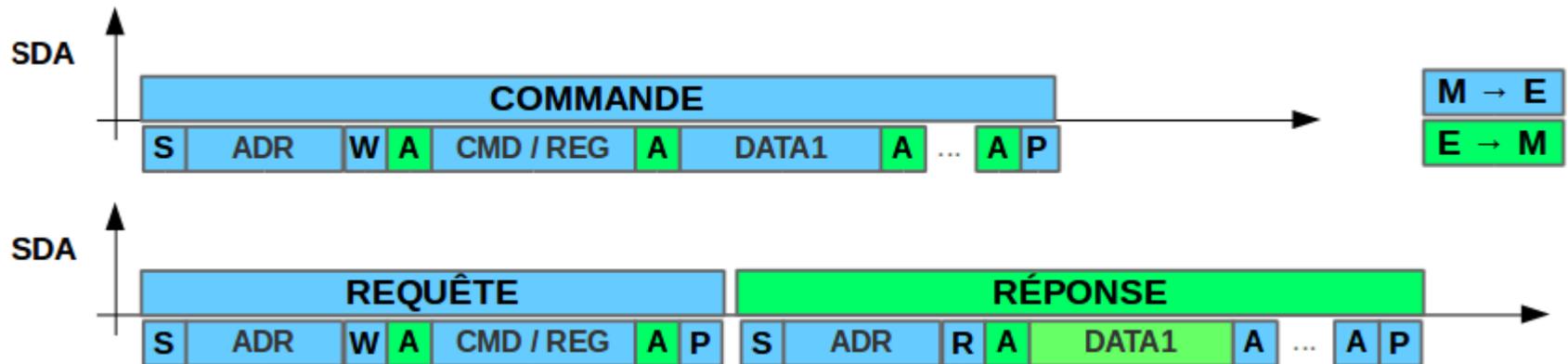
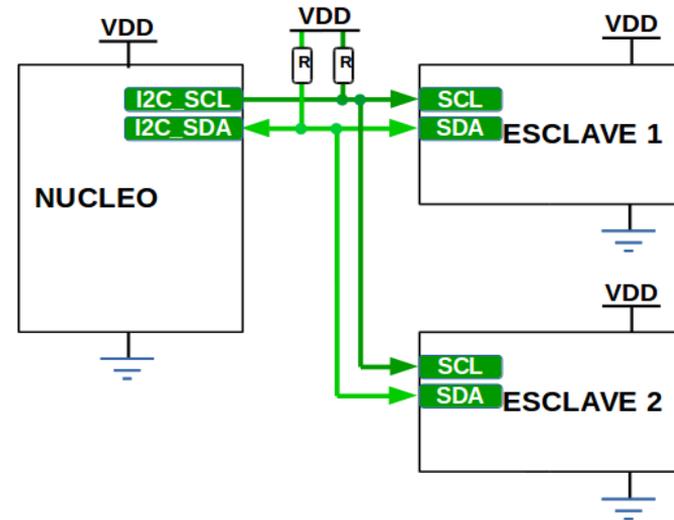
Saint-Étienne



Bordeaux

# Communiquer en I2C

- Réseau de terrain sur 2 fils
- Relation maître/esclaves
- Adressage des esclaves



Paris-Saclay



Saint-Étienne



Bordeaux

- Exemple du **TC1321** / DAC sur carte *Microchip I2C*

- Bibliothèque **WIRE**

TABLE 4-2: TC1321 COMMAND SET (READ\_BYTE AND WRITE\_BYTE)

Command Byte Description		
Command	Code	Function
RWD	00h	Read/Write Data (DATA)
RWCR	01h	Read/Write Configuration (CONFIG)

TABLE 4-3: CONFIGURATION REGISTER (CONFIG), 8-BIT, READ/WRITE

Configuration Register (CONFIG)								
Bit Name	D[7]	D[6]	D[5]	D[4]	D[3]	D[2]	D[1]	D[0]
Bit Function	Reserved (Note 1)							SHDN (Note 2)

Note 1: Always returns '0' when reading

Note 2: 1 = Standby (Shut down) mode  
0 = Normal mode

TABLE 4-4: DATA REGISTER (DATA), 10-BIT, READ/WRITE

Data Register (DATA) for 1st Byte								Data Register (DATA) for 2nd Byte							
D[9]	D[8]	D[7]	D[6]	D[5]	D[4]	D[3]	D[2]	D[1]	D[0]	X	X	X	X	X	X
MSB	X	X	X	X	X	X	X	X	X	LSB	X	X	X	X	X

## Functions

- `begin()`
- `requestFrom()`
- `beginTransaction()`
- `endTransmission()`
- `write()`
- `available()`
- `read()`
- `SetClock()`
- `onReceive()`
- `onRequest()`



Paris-Saclay



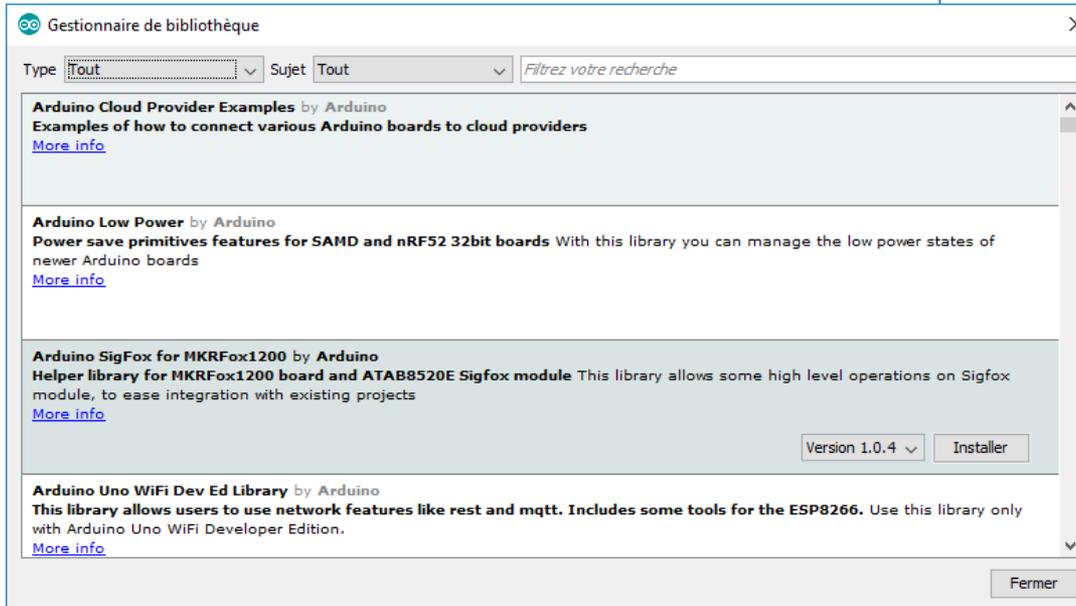
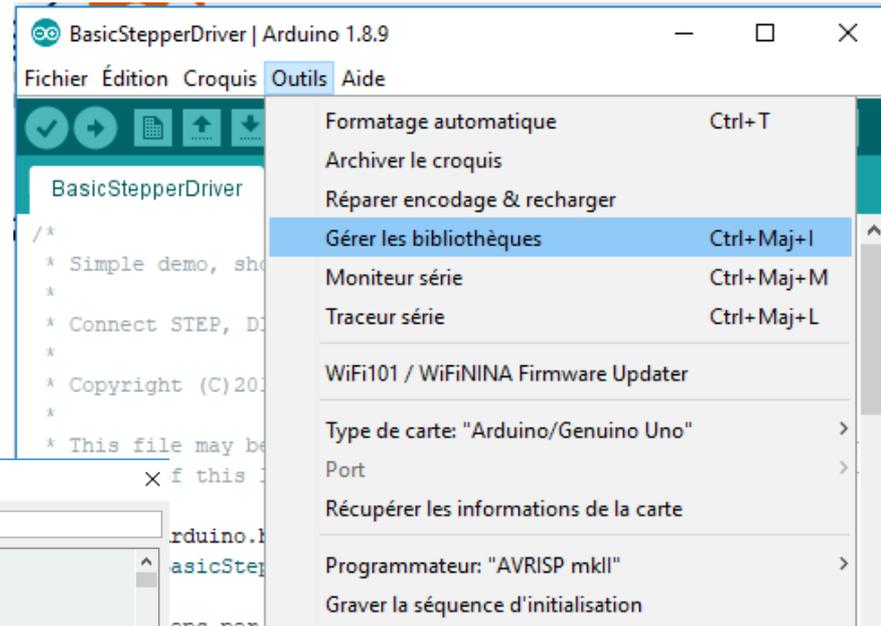
Saint-Étienne



Bordeaux

# Des tonnes de bibliothèques !!

- Brushless Servo
- Stepper Driver / DRV8825



Paris-Saclay



Saint-Étienne



Bordeaux