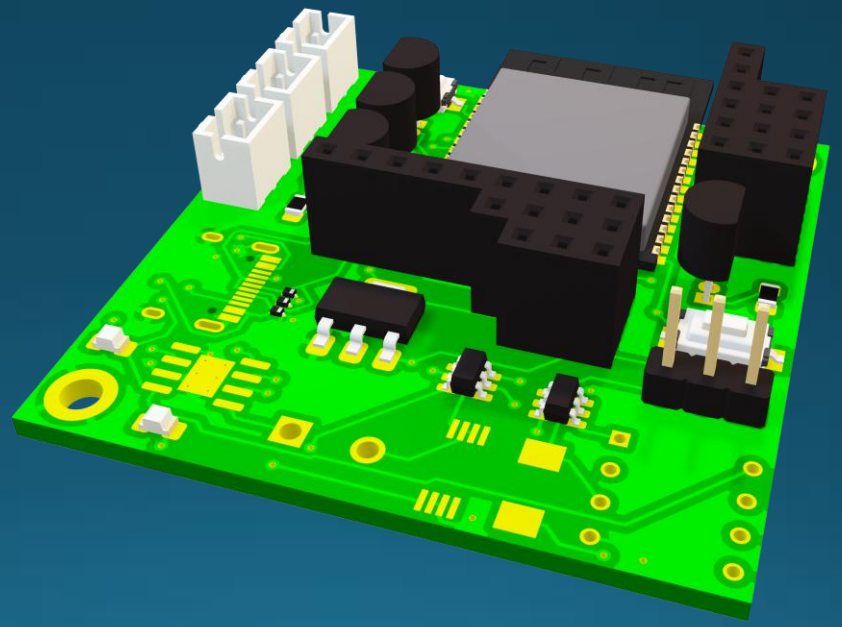
Introduction à l'électronique pratique

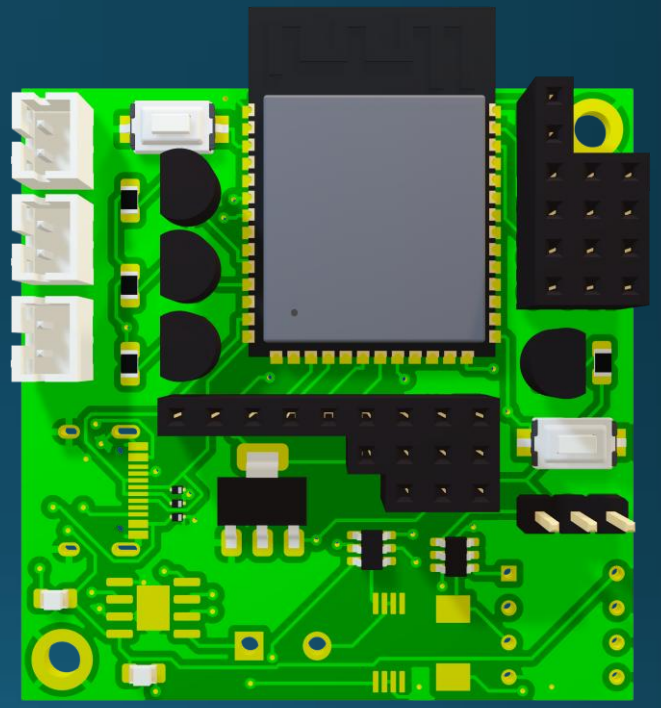


Sonny BASSO et Florian KLEIN

Plan d’aujourd’hui

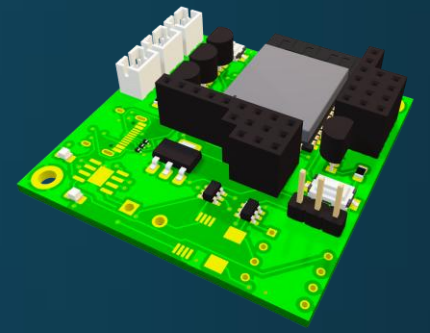
- Les bases
  - Loi d’Ohm
  - Composants principaux
  - Diviseur de tension

- Faire un projet électronique de A à Z
  - Le kit Botanique



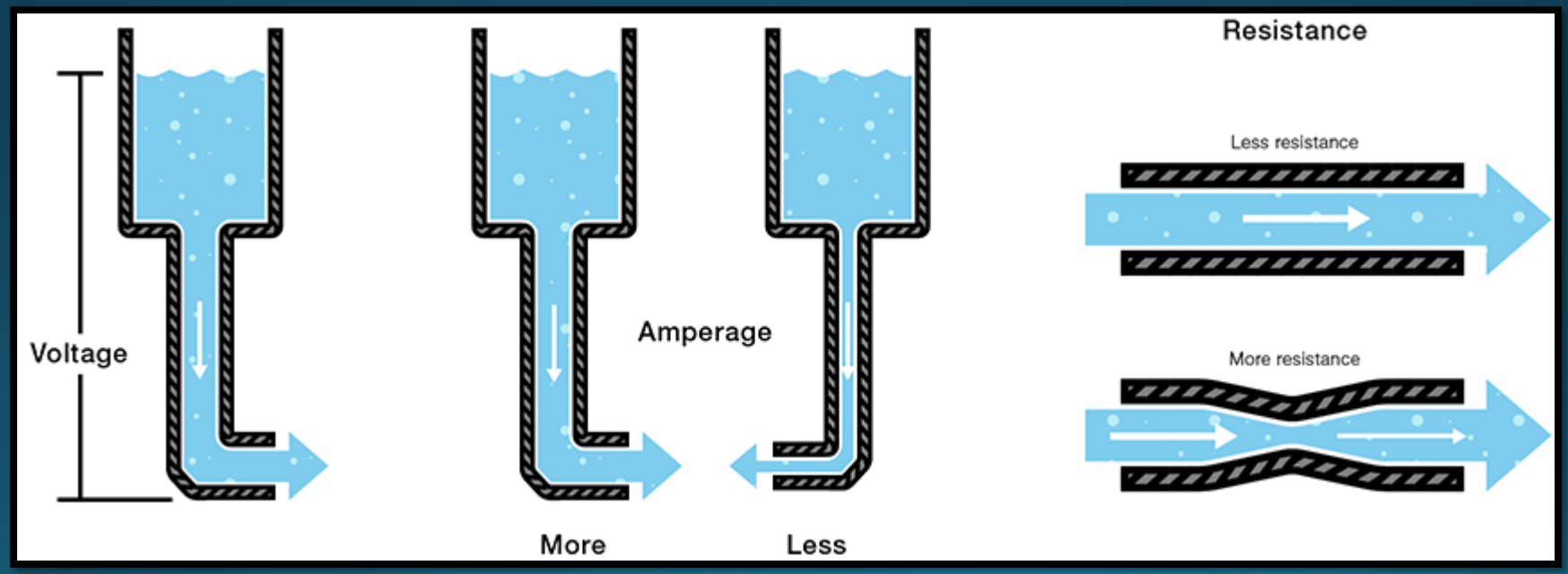
Les bases

Loi d'Ohm



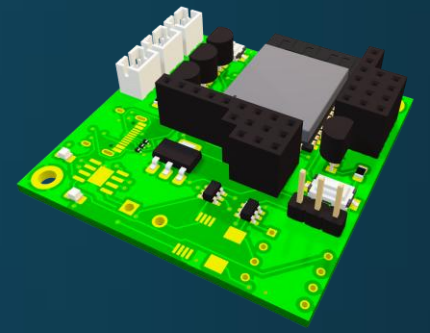
Loi d'Ohm : $U = R \cdot I$

$P = U \cdot I$



Les bases

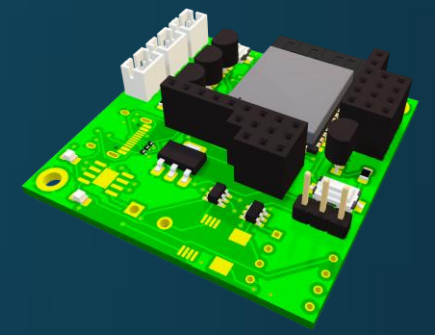
Composants principaux

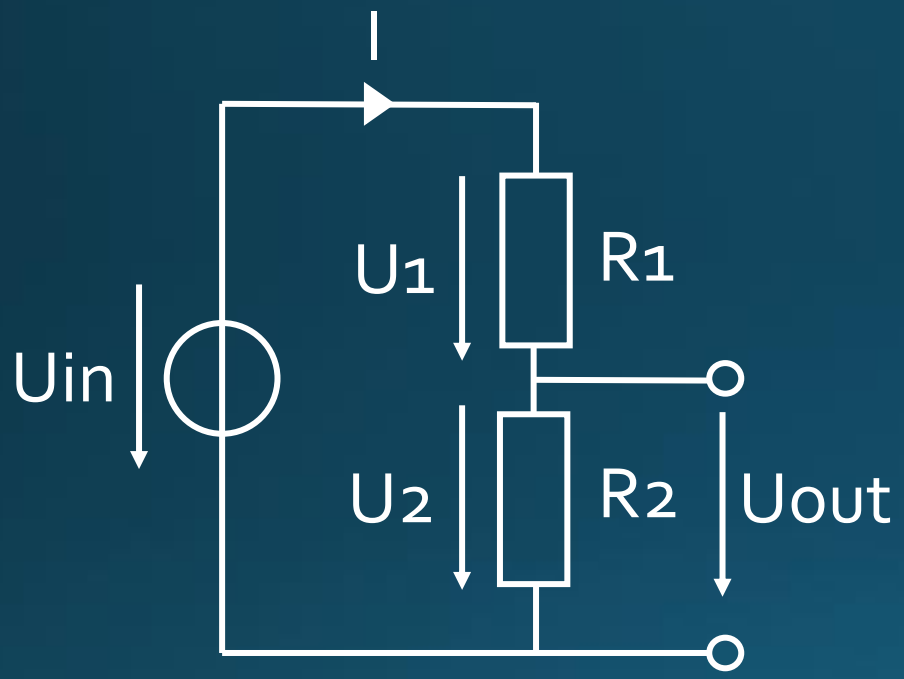


COMPOSANT	RÔLE	EXEMPLE CONCRET
Résistance	Limiter le courant	LED
Diode / LED	Laisser passer le courant dans un seul sens	Indicateur
Condensateur	Stocker une charge	Anti-bruit, temporisation
Transistor	Interrupteur commandé	Commander un moteur
Potentiomètre	Résistance variable	Contrôle de luminosité
Bouton / interrupteur	Entrée manuelle	Déclenchement d'action
Capteur (LDR, DHT, etc.)	Mesurer une grandeur	Température, lumière
Diode Schottky	Protection	Anti-retour courant
Régulateur de tension	Fournir tension stable	LM7805, AMS1117

Les bases

Diviseur de tension

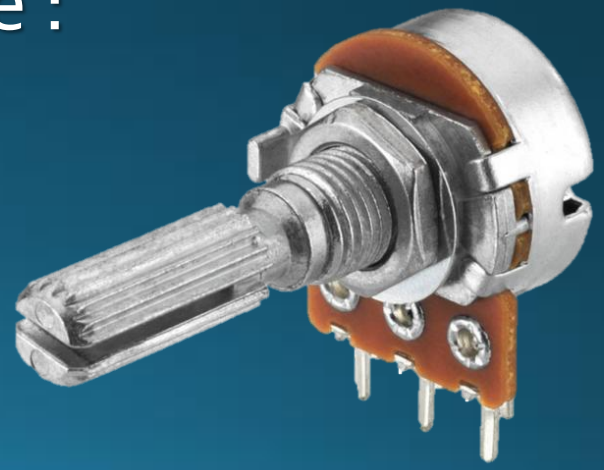




$I = U_{in}/(R_1+R_2)$

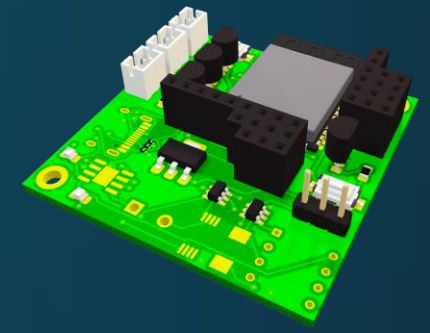
$U_{out} = I \cdot R_2 = U_{in} \cdot R_2/(R_1+R_2)$

Potentiomètre :



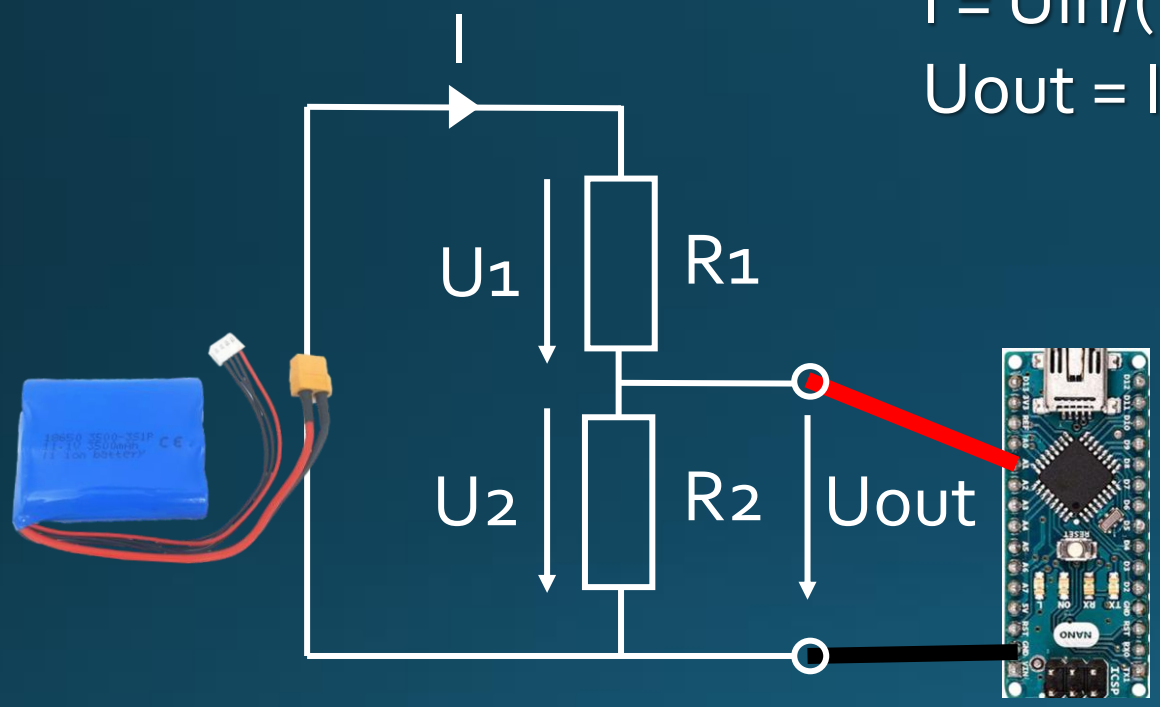
Les bases

Diviseur de tension



$$I = U_{in}/(R_1+R_2)$$

$$U_{out} = I \cdot R_2 = U_{in} \cdot R_2/(R_1+R_2)$$



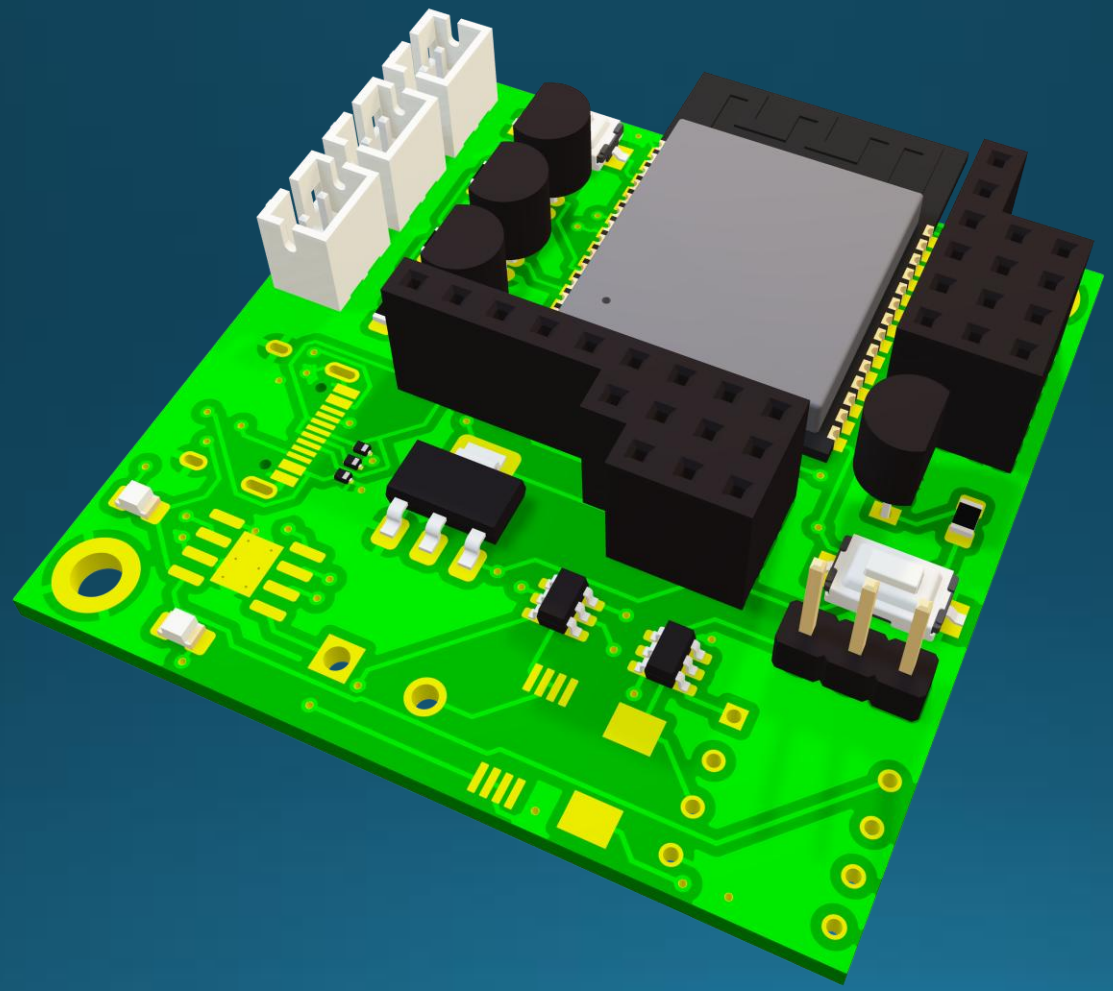
Li-ion 3s : 9.6V-12.6V

Tension analogique d'entrée arduino nano : 0V-5V

→ Rescale la tension pour qu'elle soit entre 0V et 5V→R1=20kohm et R2=10kohm

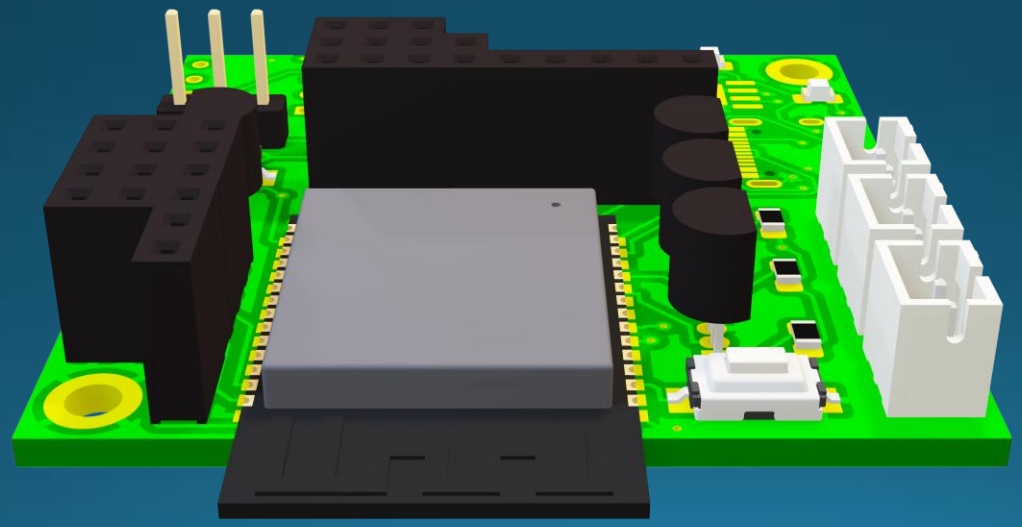
- 12.6V→4.2V
- 9.6V →3.2V

Projet de A à Z : Kit Bota



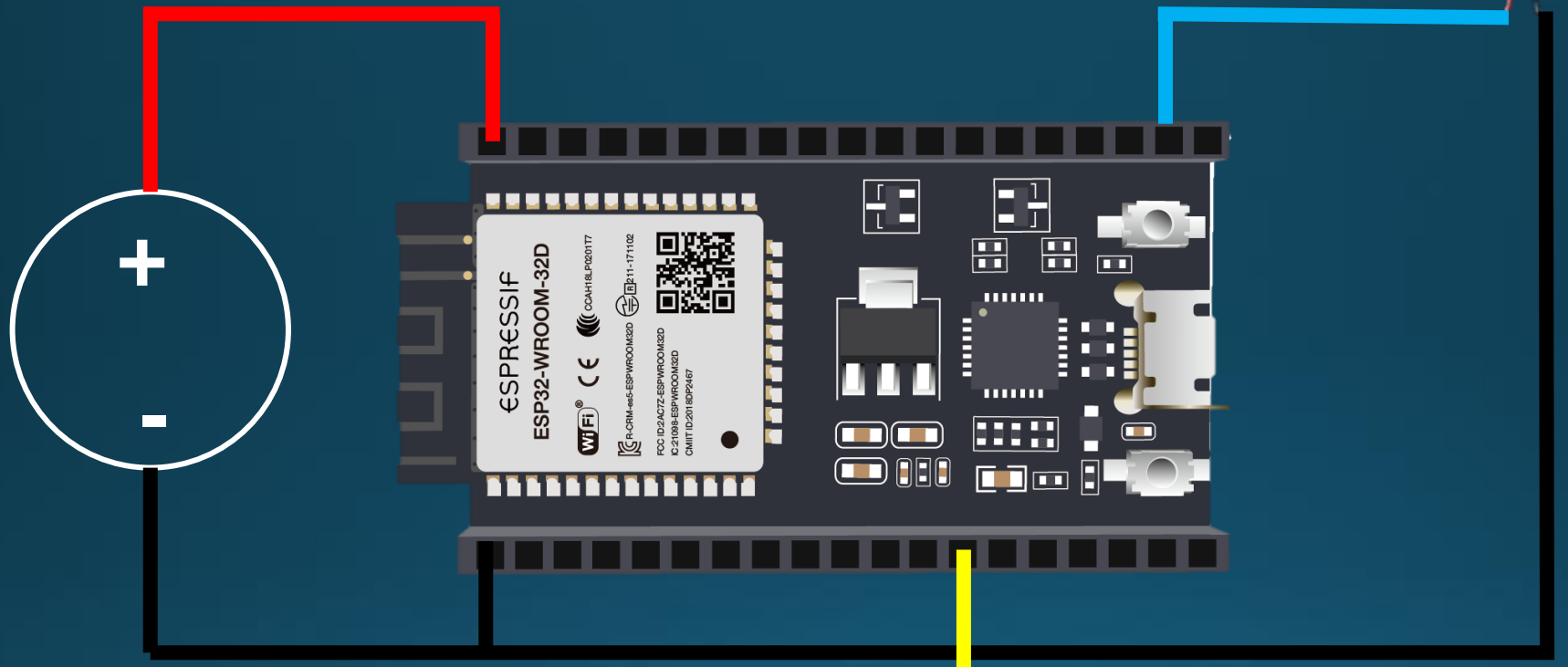
Projet de A à Z : Kit Bota

Idée | Processeur | Capteur | Actuateur | Alim | Code | Tests | Produit fini



Concept / Idée

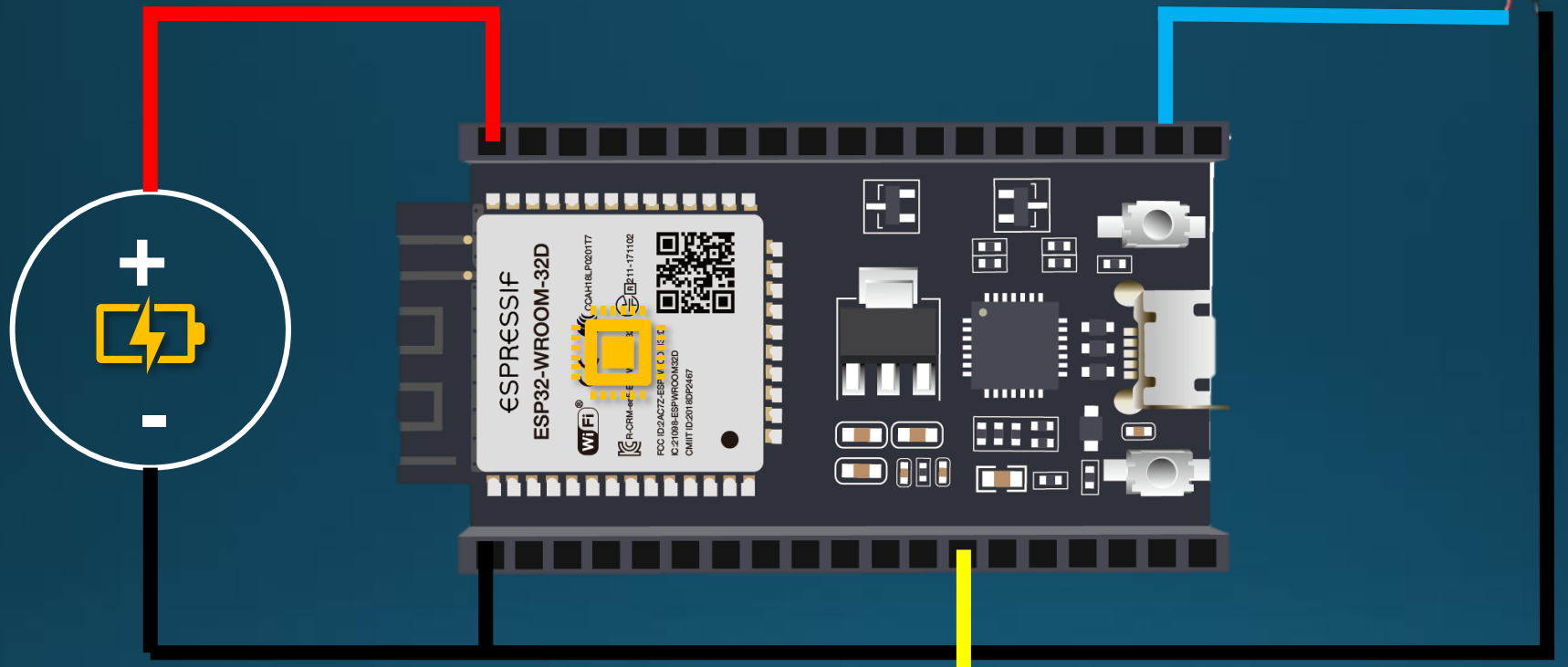
Circuit type



- Gestion autonome d’une plante
- Relevé de données et arrosage automatique

Concept / Idée

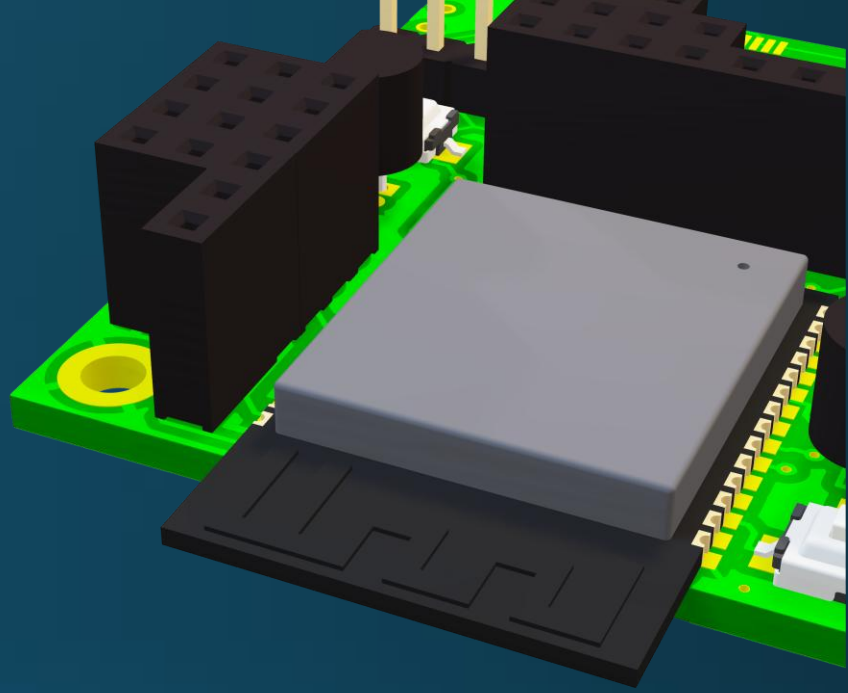
Circuit type



- Gestion autonome d’une plante
- Relevé de données et arrosage automatique

Microcontrôleur

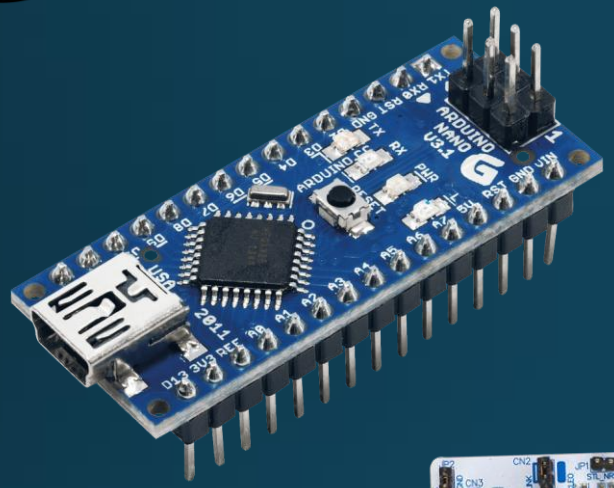
Généralités

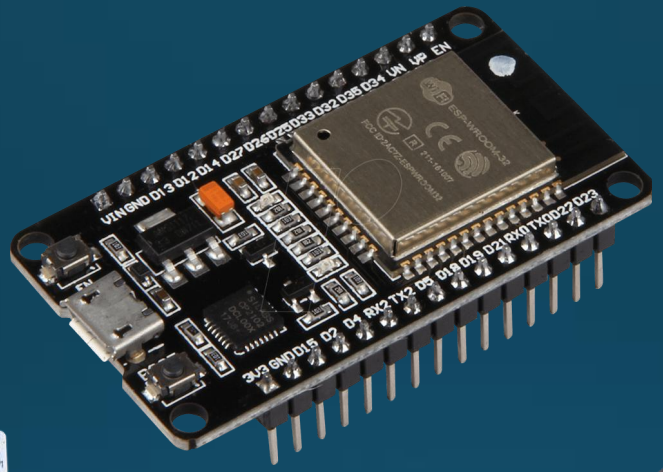


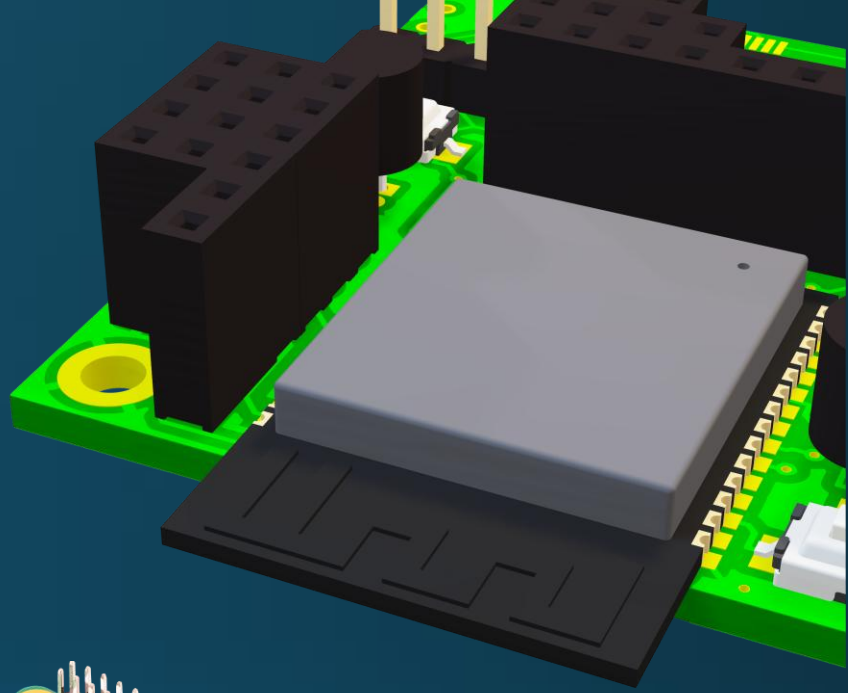
- Cerveau du système
- Calculs
- Traitement des données
- Communication
- Etc

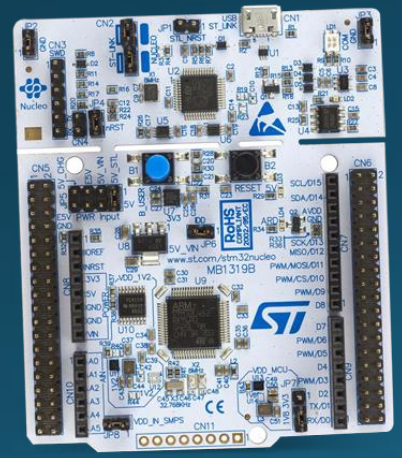
Microcontrôleur

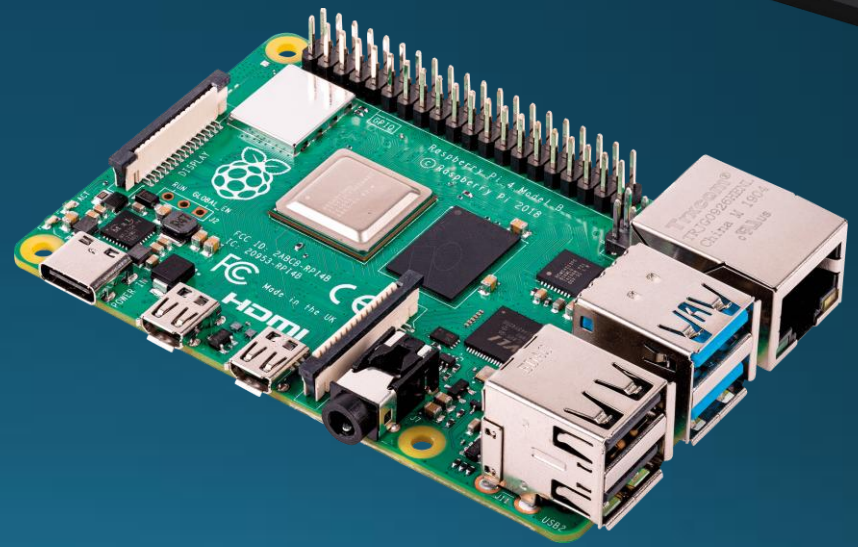
Généralités





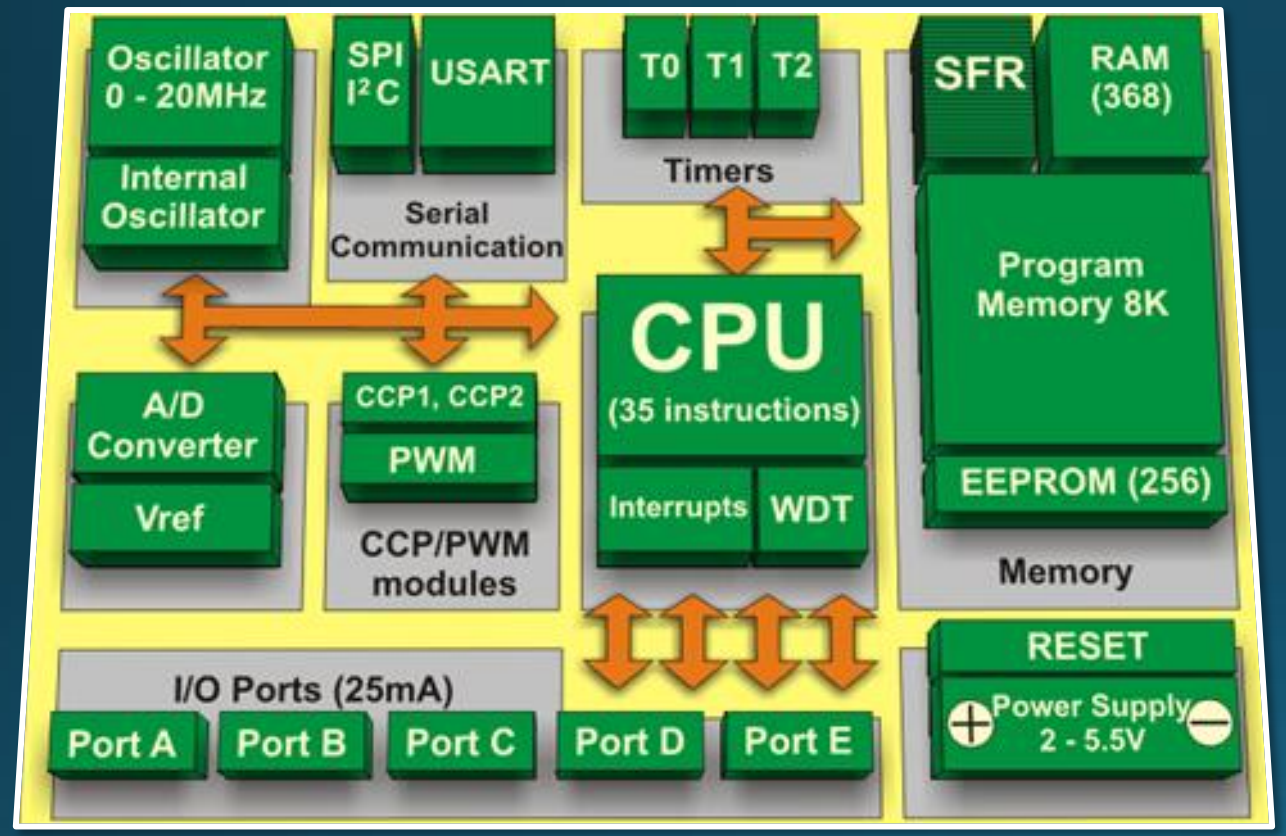


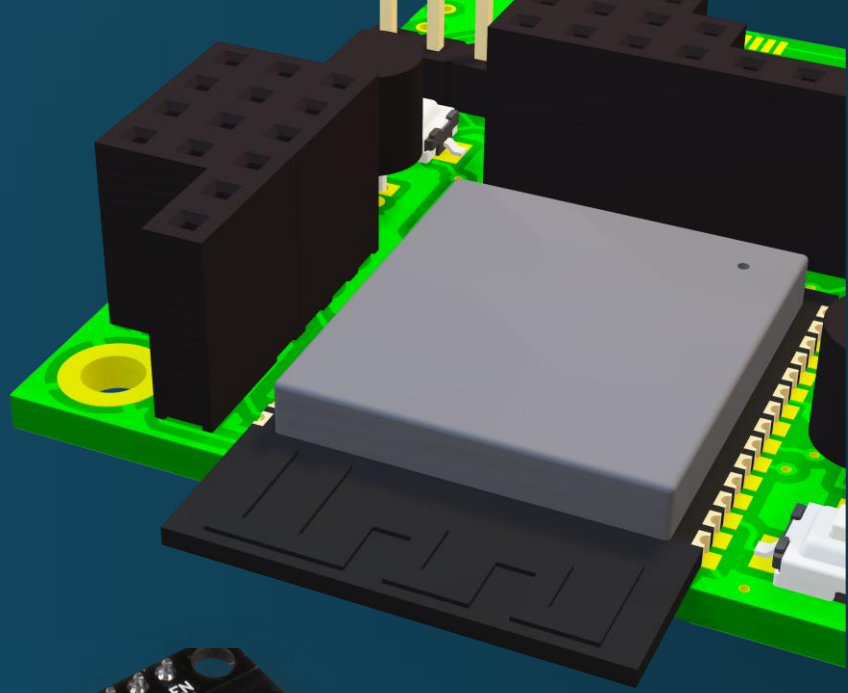


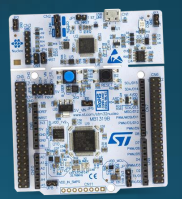


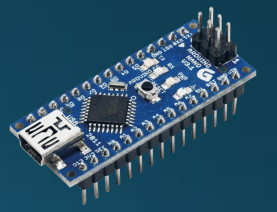
Microcontrôleur

!= Microprocesseur



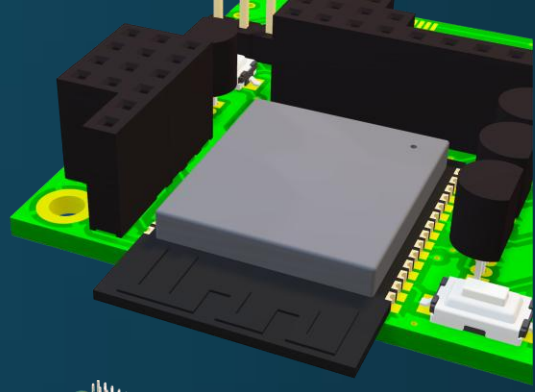


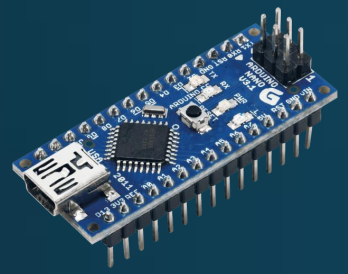




Microcontrôleur

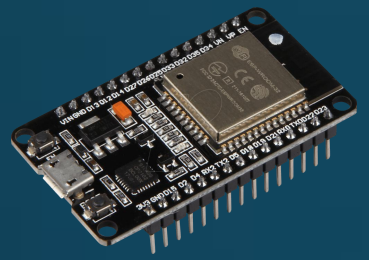
Comment choisir ?





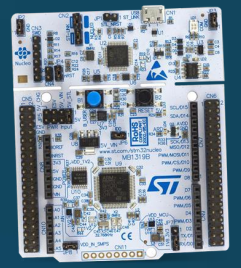
Arduino
(ATmega)

- Très simple à utiliser
- Enorme communauté
- Support pédagogique
- Librairies
- IDE intuitif



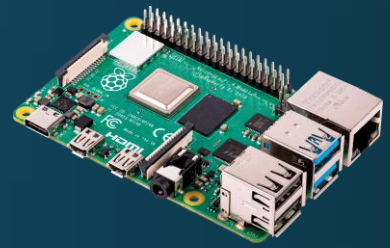
ESP32

- + performant
- Wifi + Bluetooth
- Double cœur
- + de GPIO, ADC, PWM
- Compatible Arduino



STM32

- Très performant
- Faible consommation
- Vaste gamme de choix
- IDE professionnels (STM32CubeIDE…)

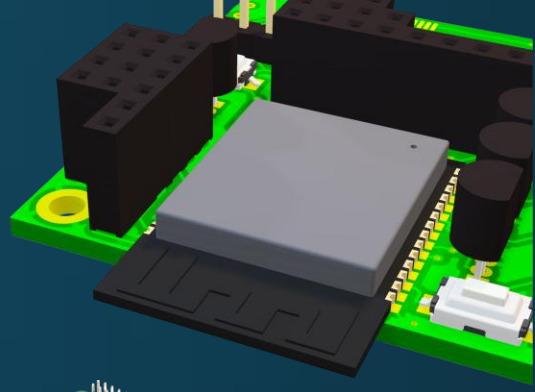


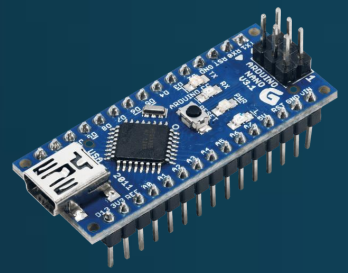
Raspberry Pi

- Véritable ordinateur
- Rapport puissance/prix
- Double cœur (même 4)
- Beaucoup de GPIO…
- Bonne documentation

Microcontrôleur

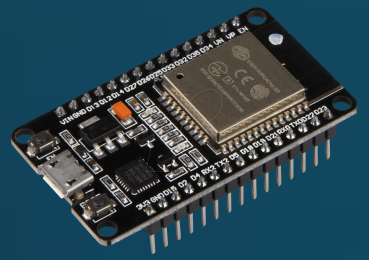
Comment choisir ?





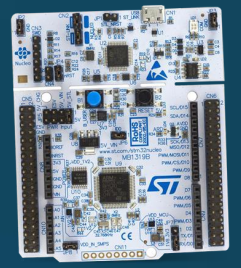
Arduino
(ATmega)

- Peu de puissance
- Pas de WiFi/BLE
- Conso moyenne
- Mémoire et RAM limitées



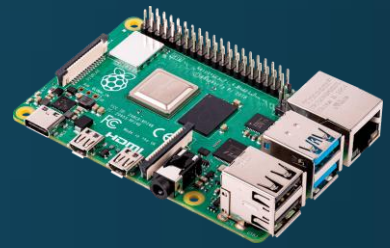
ESP32

- + complexe à configurer
- Conso + élevée



STM32

- Courbe d'apprentissage
- Config plus complexe
- Communauté moins large

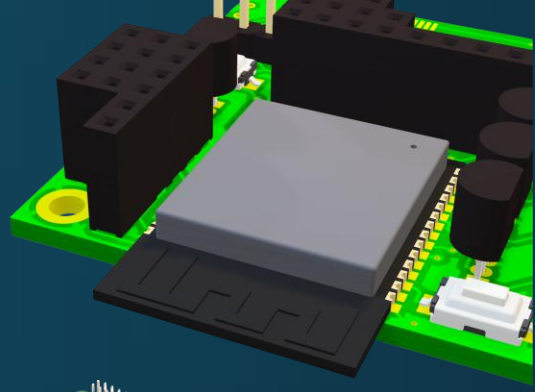


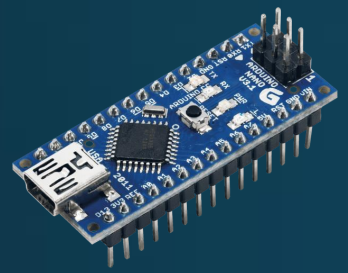
Raspberry Pi

- Moins de librairies
- Pas de DAC intégré
- Codé en python (à vous de voir si c'est bien ou pas…)

Microcontrôleur

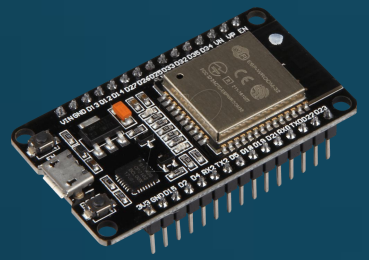
Comment choisir ?





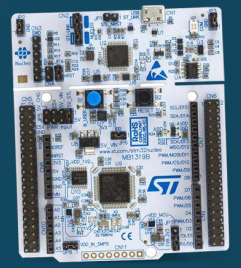
Arduino
(ATmega)

- Débutants
- Education
- Projets simples



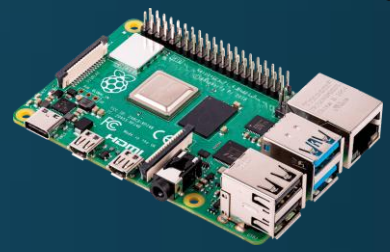
ESP32

- Projets connectés
- Plus évolués



STM32

- Projets exigeants
- Temps réel

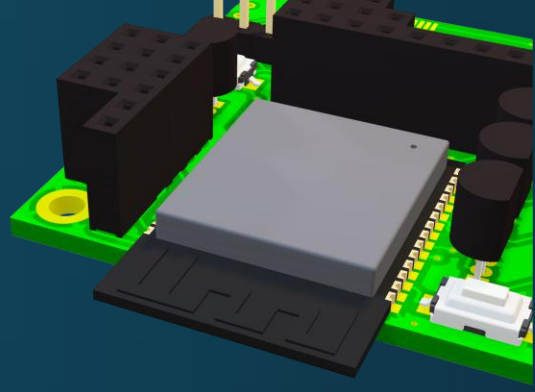


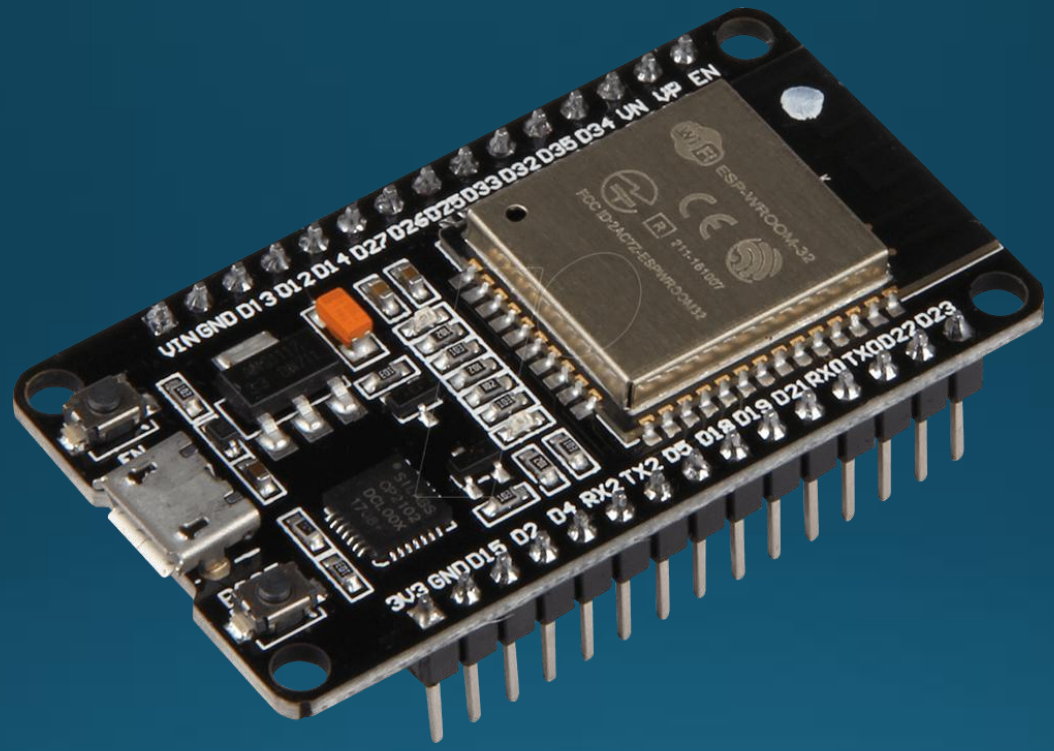
Raspberry Pi

- Logique complète embarquées
- Interface USB
- Optimisation
- Multi-tâche

Microcontrôleur

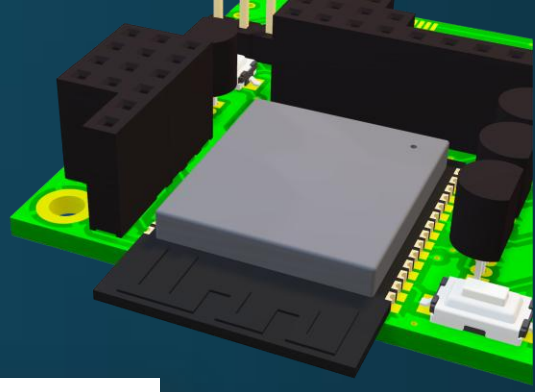
Comment choisir ?

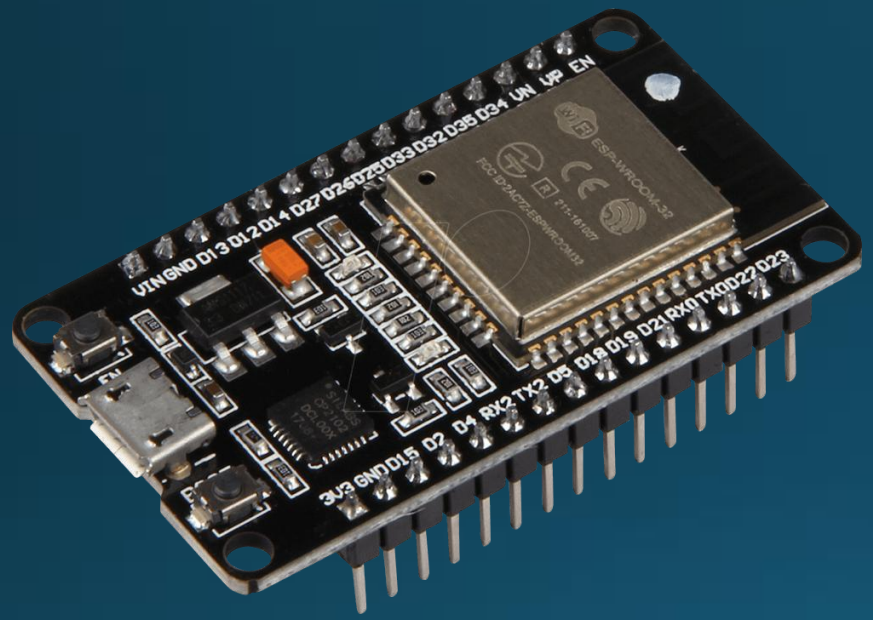


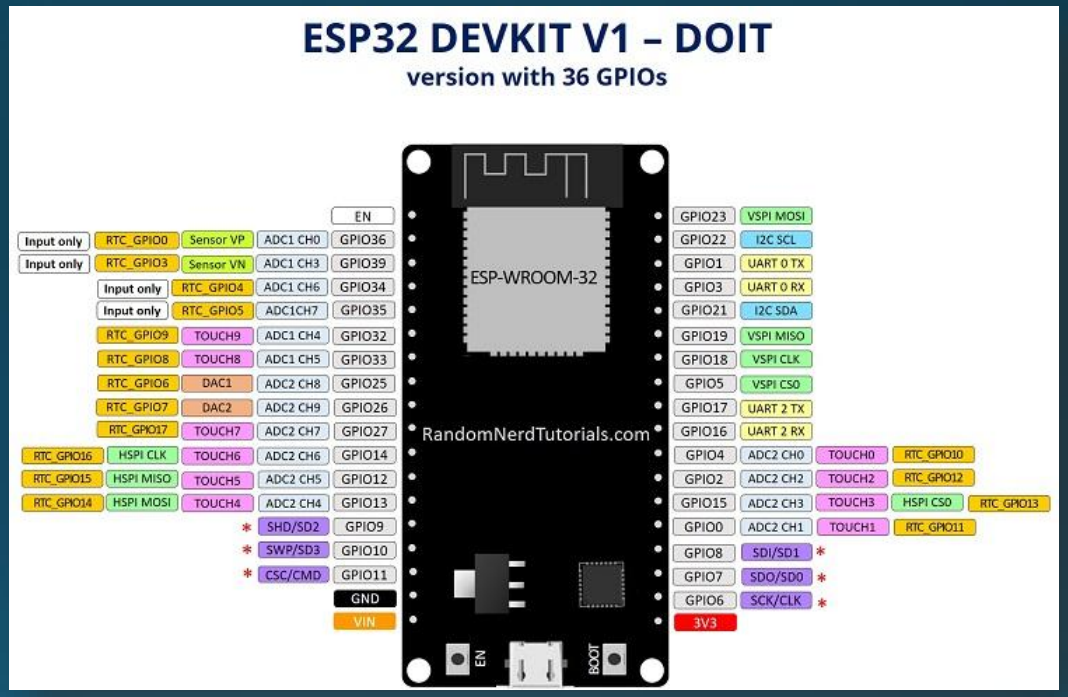


Microcontrôleur

Comment choisir ?

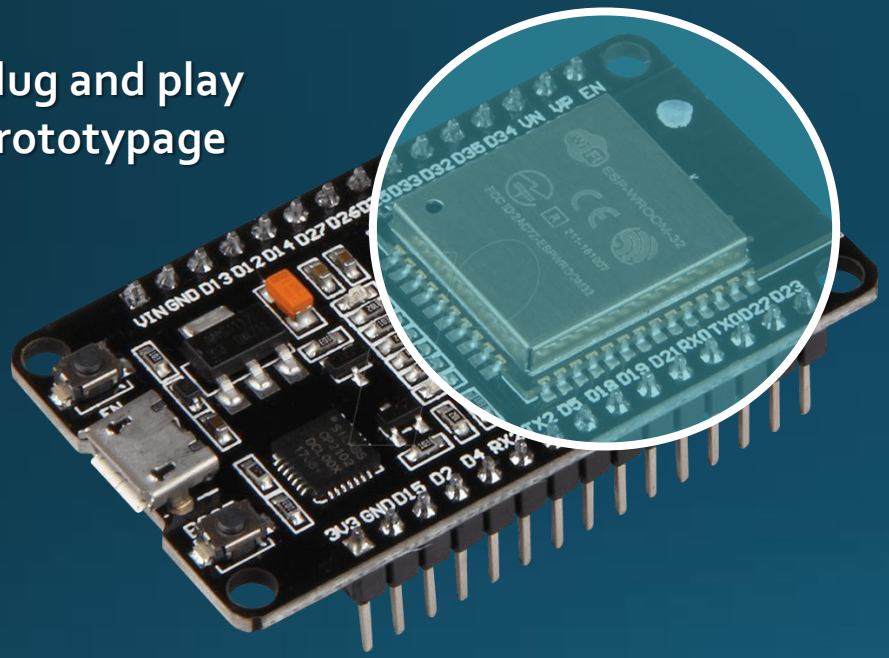




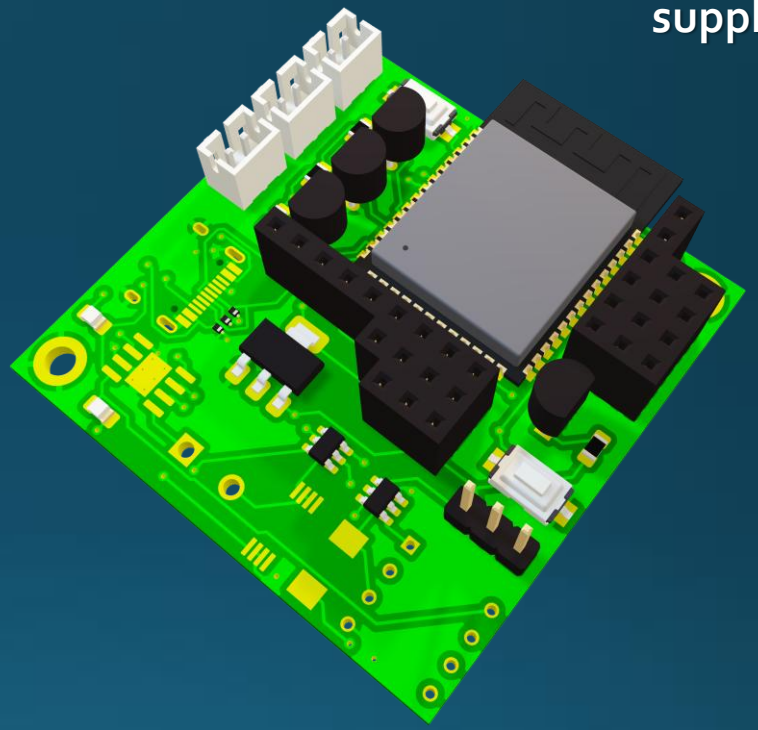


Microcontrôleur
Comment choisir ?

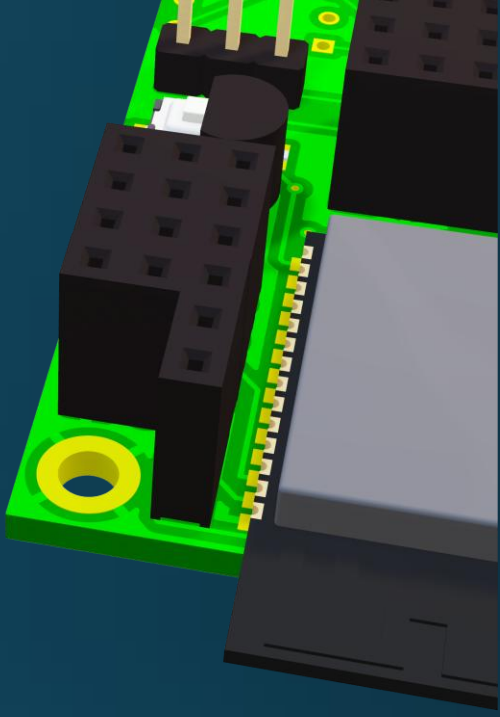
- Board
  - Plug and play
  - Prototypage



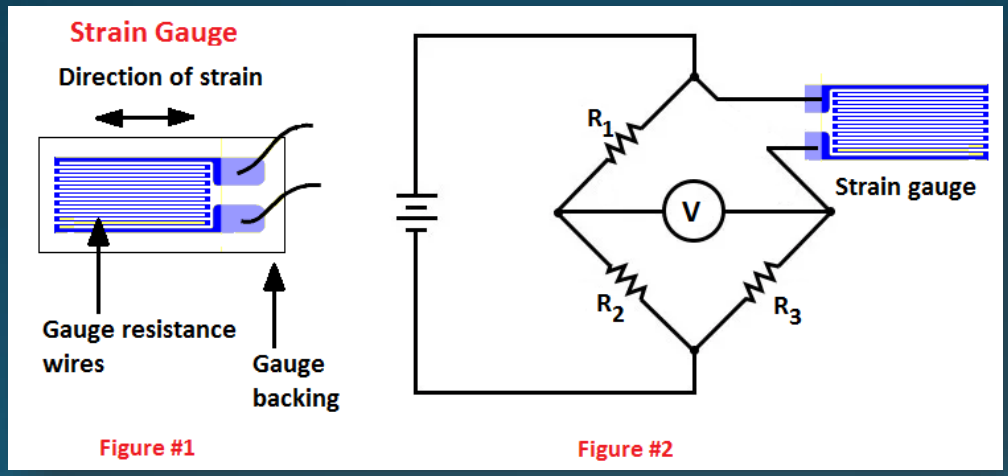
- Chip only
  - Plus opti
  - Plus complexe
  - Circuits supplémentaires



Capteur
Généralités

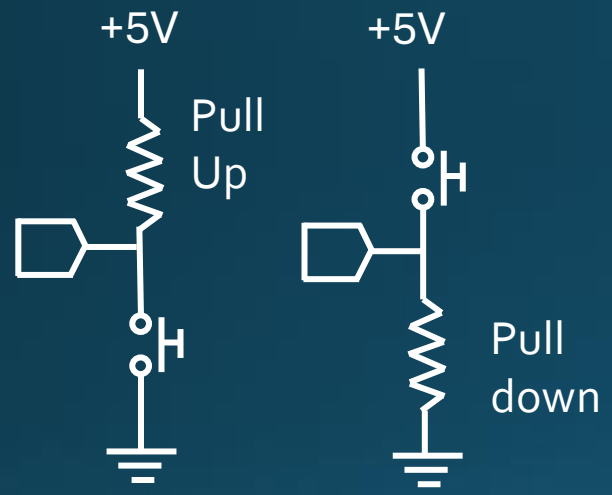


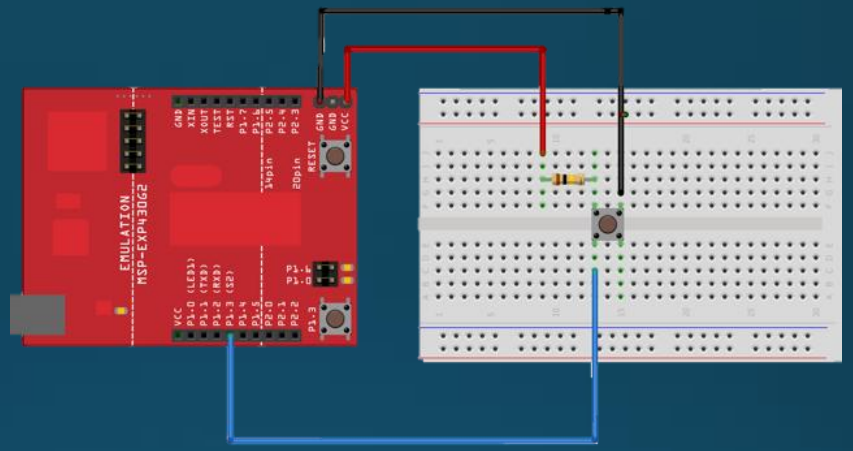
- Mesure de l'environnement
- Domaine physique → Domaine électrique
- Différentes technologies de capteurs

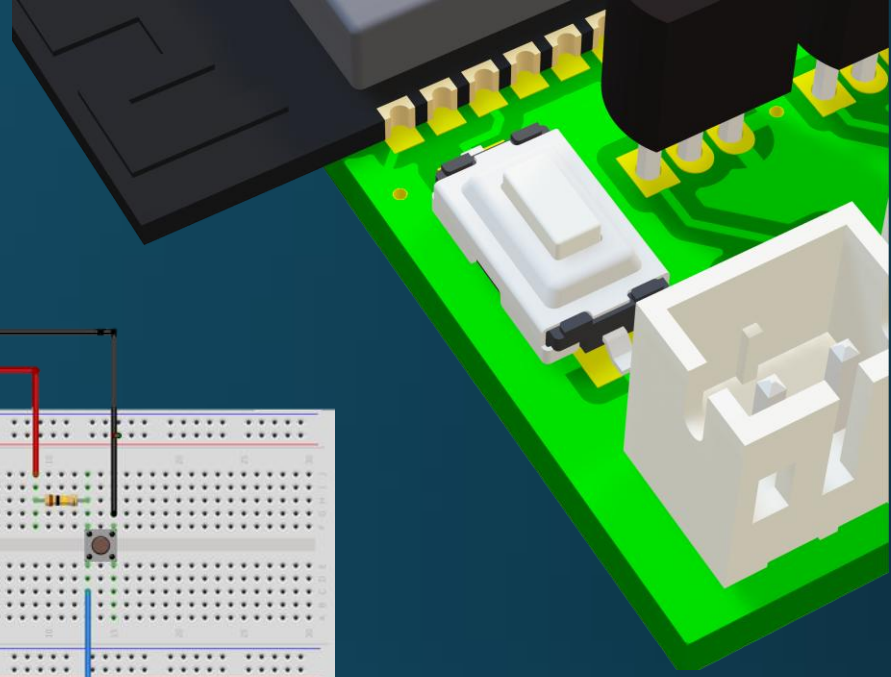


Capteur

Pull-up / Pull-down

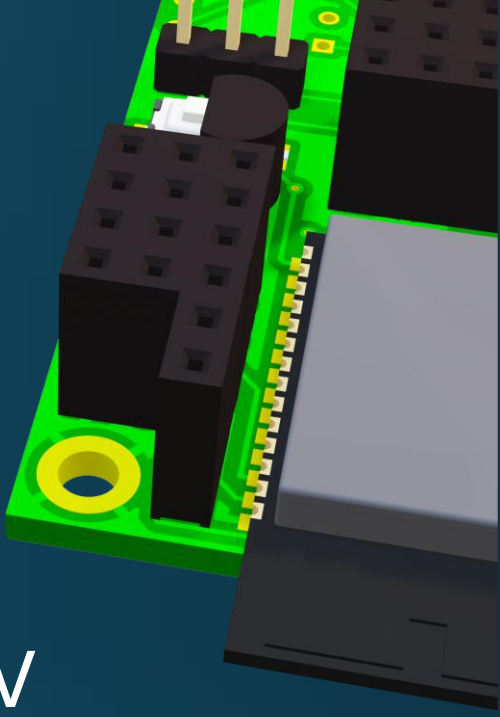






- Eviter un potentiel flottant
- Tirer le courant à Vcc ou à la masse

Capteur
Analogique



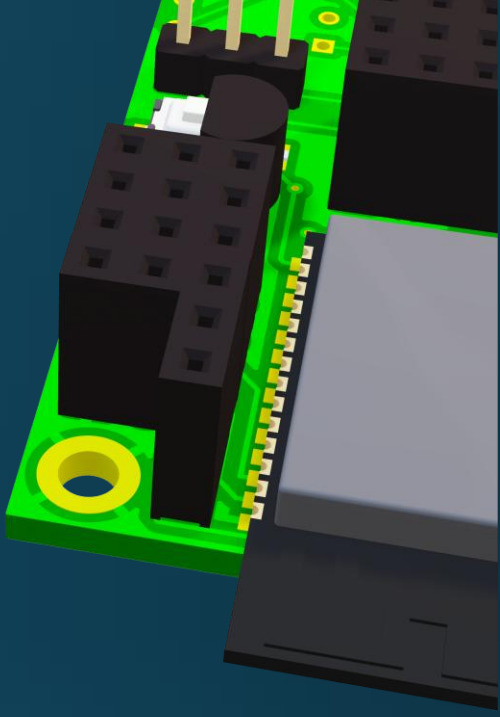
GND = 0 V

VCC = 5 V

0 V < AOUT < 5 V

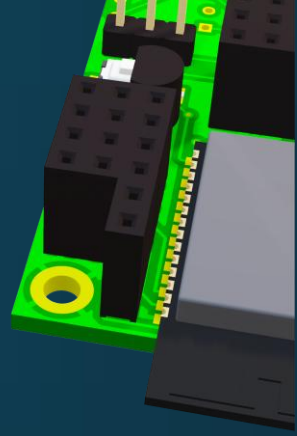
ADC nécessaire

Capteur
Numérique



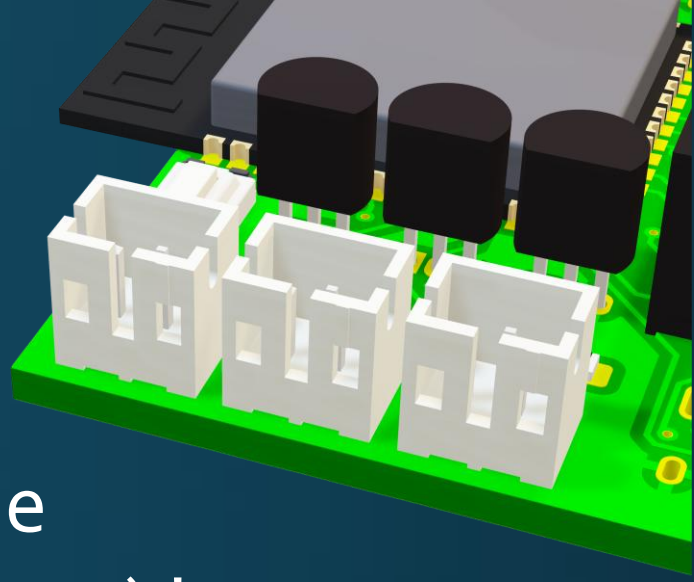
- Signal discret (propre ADC)
- Digital I/O Pins
- Meilleure précision
- Peu sensible au bruit
- Protocols:
  - Serial (UART)
  - I2C
  - SPI

Capteur
Numérique

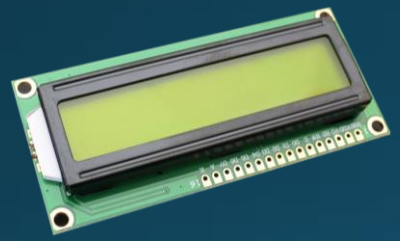


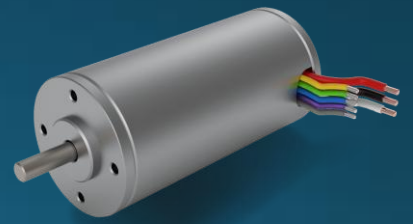
	Serial (UART)	I2C	SPI
Principe	Communication point à point	Bus série synchronisé Adressage d'esclaves	Bus série synchrone maître-esclaves
Pins	TX, RX	SDA, SCL	MOSI, MISO, SCK, SS
Vitesse	Moyenne	Lente	Rapide
Pros	Simple, universel	Multi-périphériques	Stable, simple config
Cons	1 esclave / ligne	Sensible au bruit	Plus de fils

Actuateur
Généralités

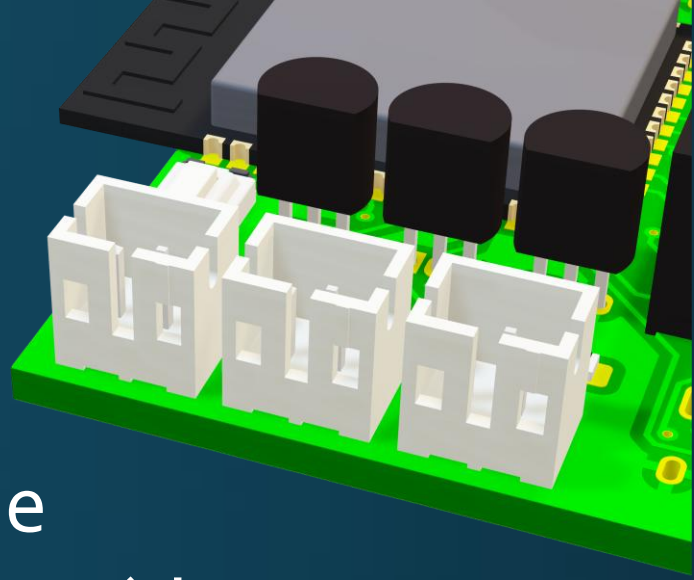


- Agit sur l'environnement
- Domaine électrique → Domaine physique
- LIRE LES DATASHEETS (circuit type, Vin etc) !

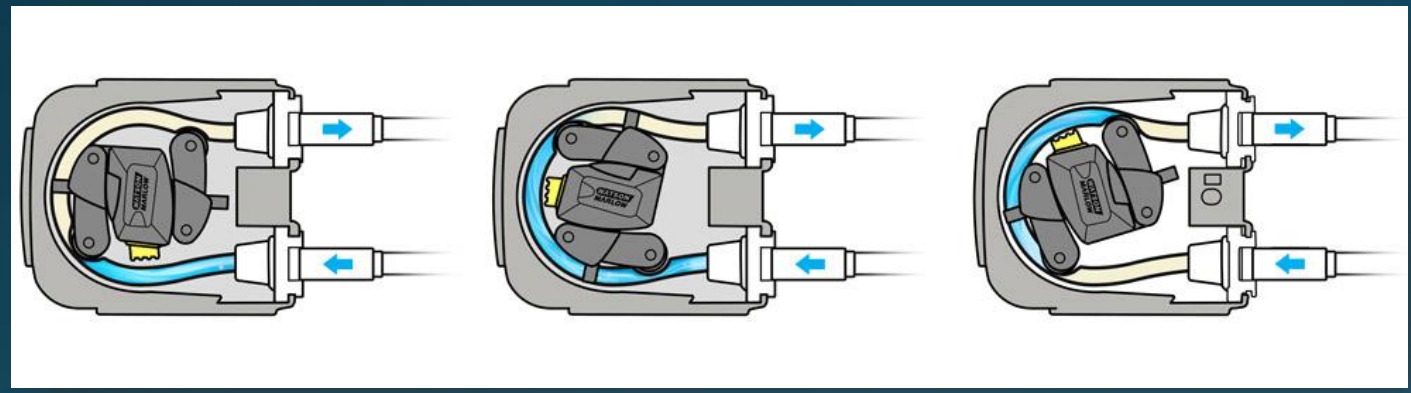




Actuateur
Généralités

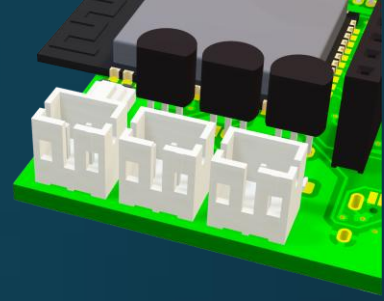


- Agit sur l'environnement
- Domaine électrique → Domaine physique
- LIRE LES DATASHEETS (circuit type, Vin etc) !



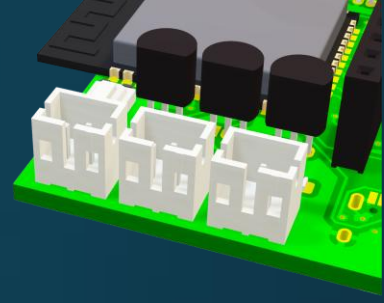
Actuateur
Moteur

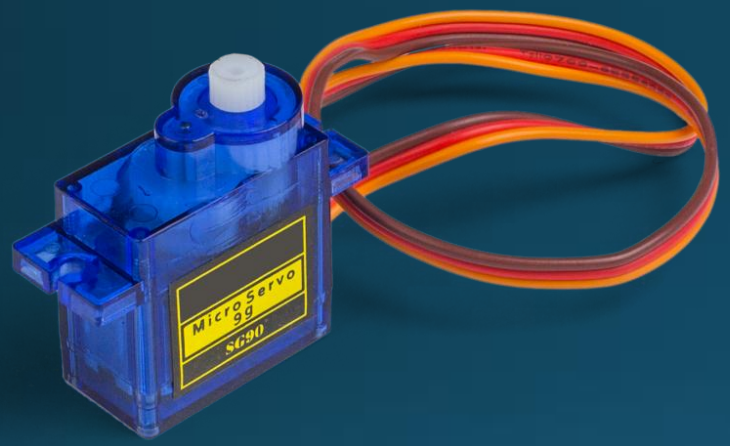
Démo moteurs 11/11



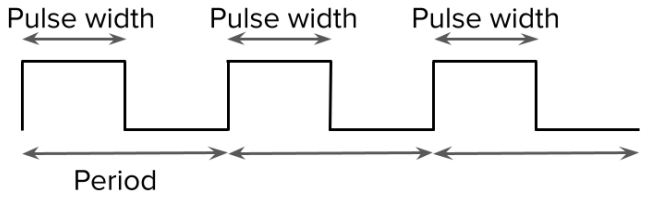
	Moteur DC à balais (brushed)	Moteur DC sans balais (brushless)	Moteur pas-à-pas	Servo moteur
Image				
Avantages	Pas cher, facile à contrôler, bon couple à faible vitesse	Durable, efficace, silencieux	Positionnement précis, stable à faible vitesse	Précision de position, bon ratio couple/poids
Inconvénients	Usure des balais, bruit, entretien	Contrôle complexe (besoin d'ESC), coût élevé	Complexe à haute vitesse, consommation élevée, poids élevé	Rotation limitée (sauf pour les servos 360°) et lente
Application	Jouets, outils électriques, petits appareils	Drones, ventilateurs…	Imprimantes 3D, CNC, robots	Bras robotiques, articulations, roues de contrôle de direction

Actuateur
Moteur





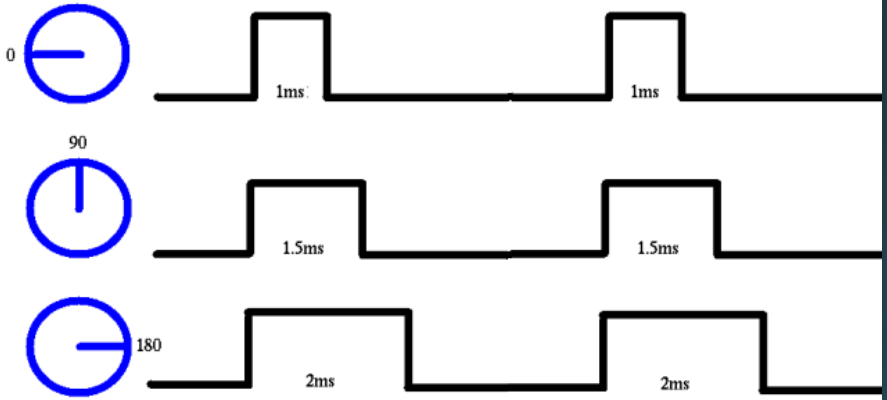
PWM



$$\text{Duty cycle (\%)} = 100 \times \frac{\text{Pulse width}}{\text{Period}}$$

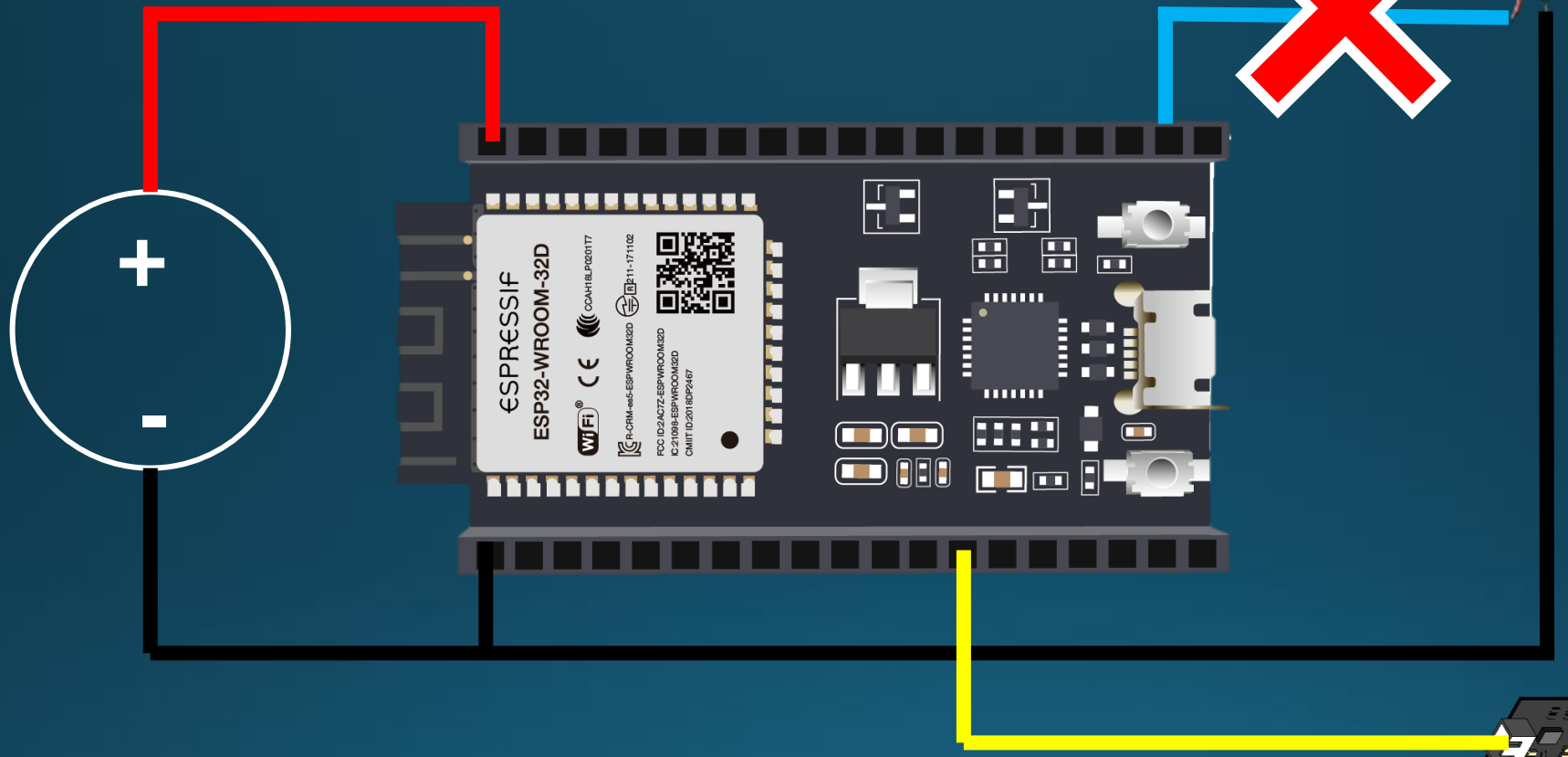
1 ms → 1 kHz

1 us → 1 MHz



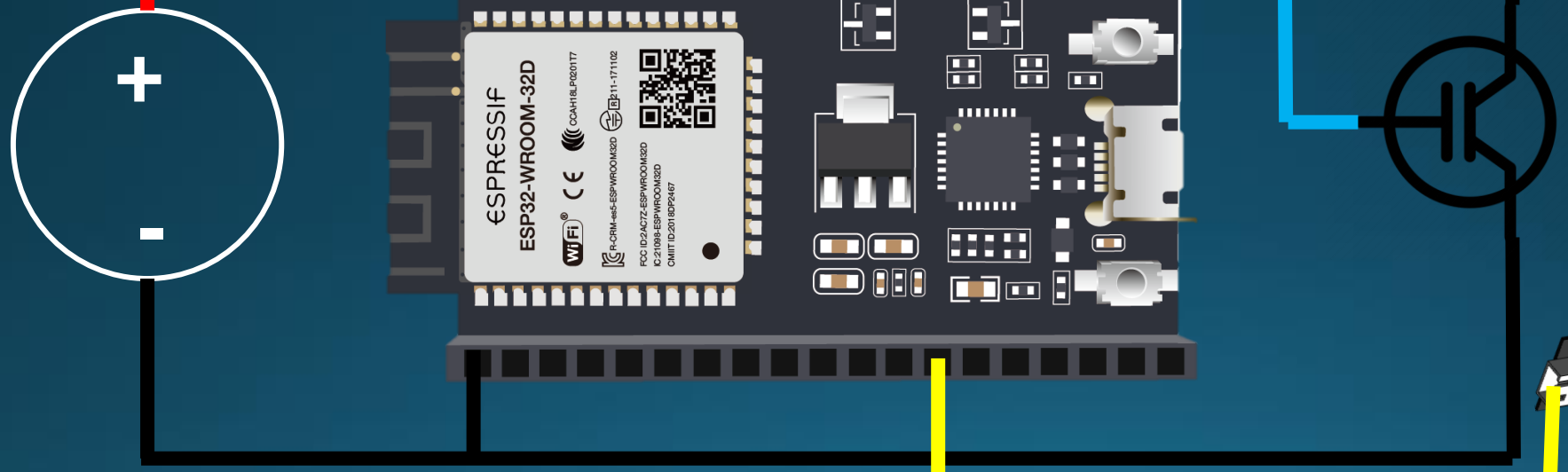
Actuateur

Retour sur le circuit type



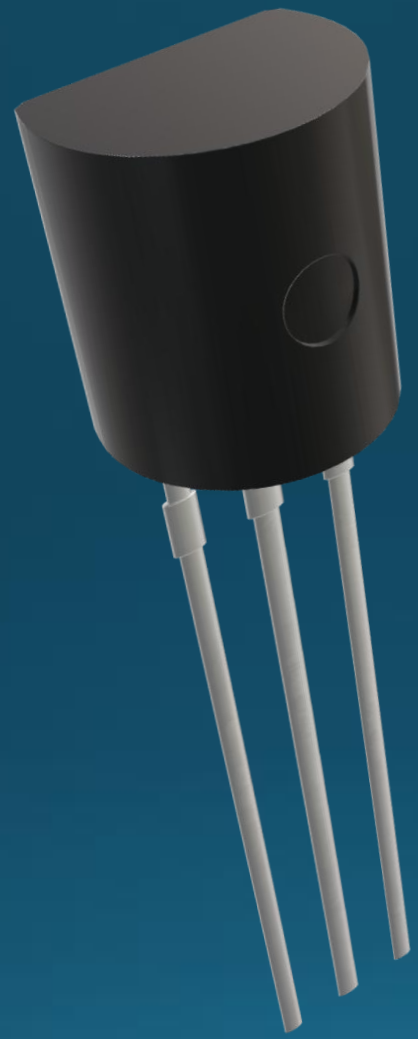
Actuateur

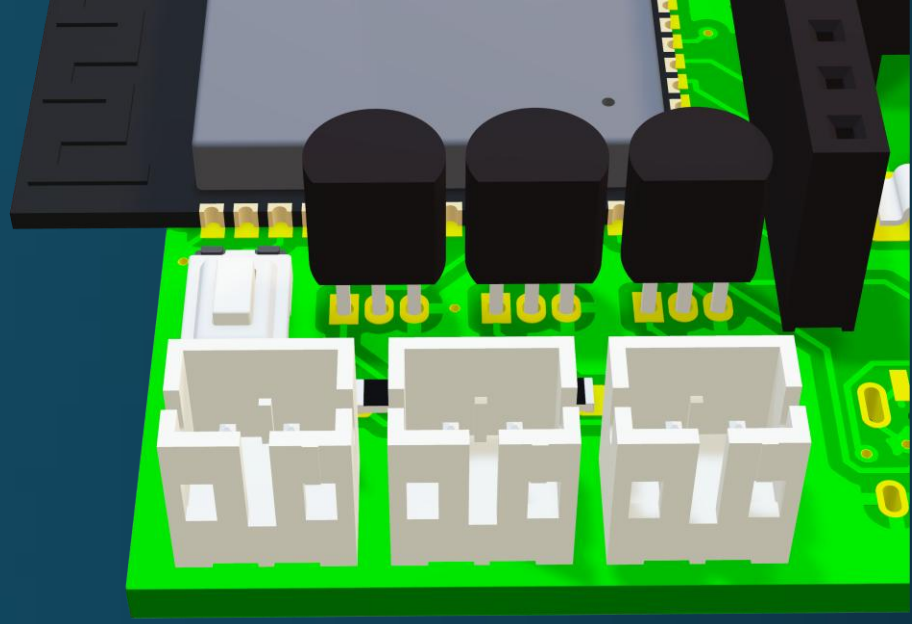
Retour sur le circuit type



Actuateur

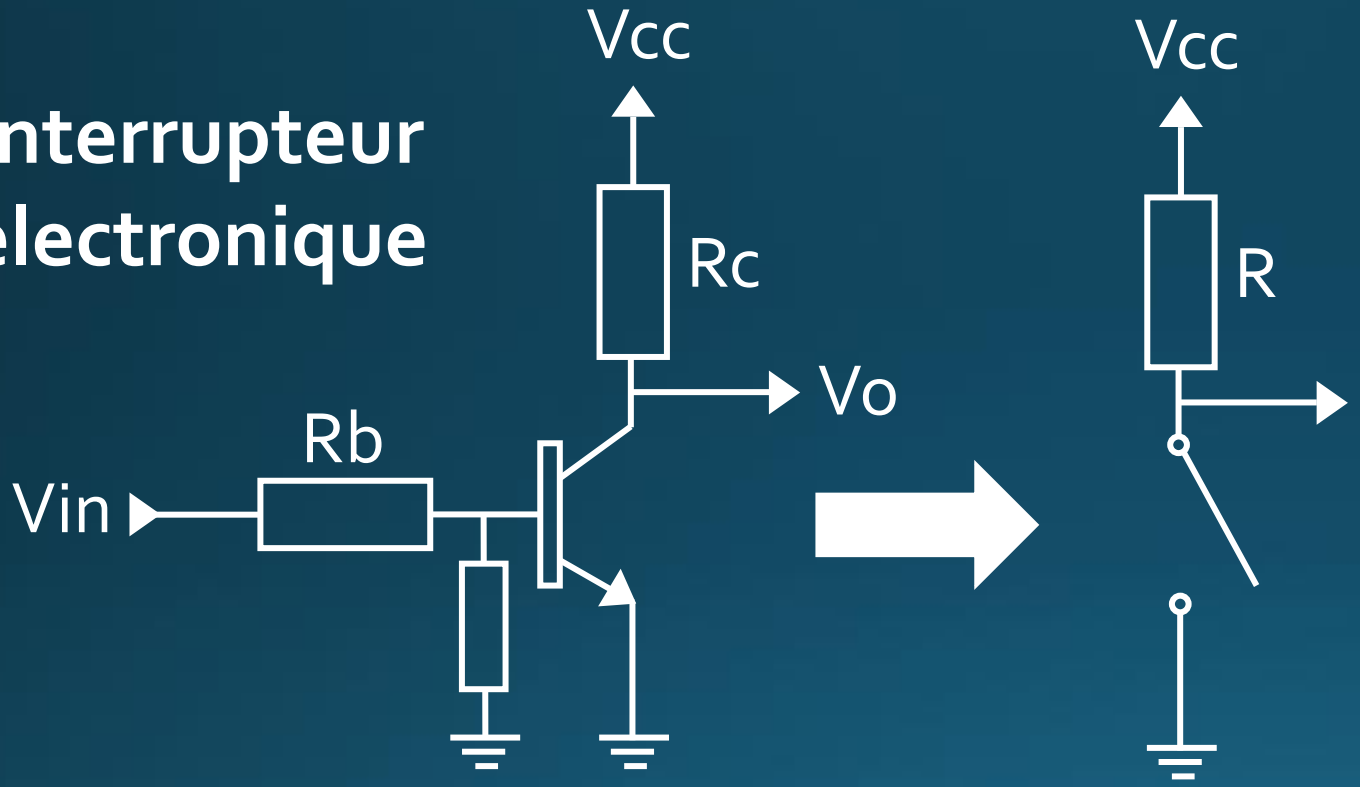
Transistor

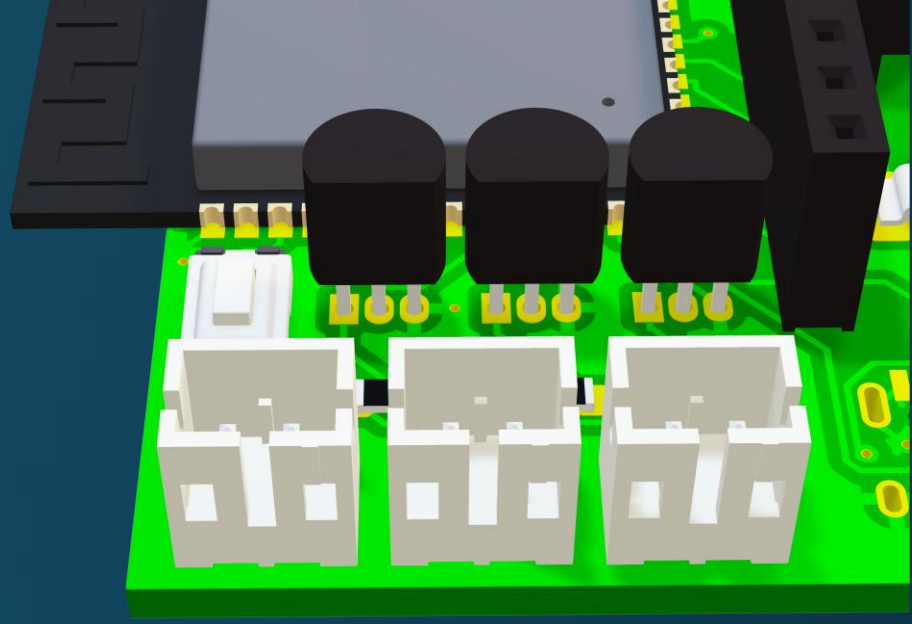


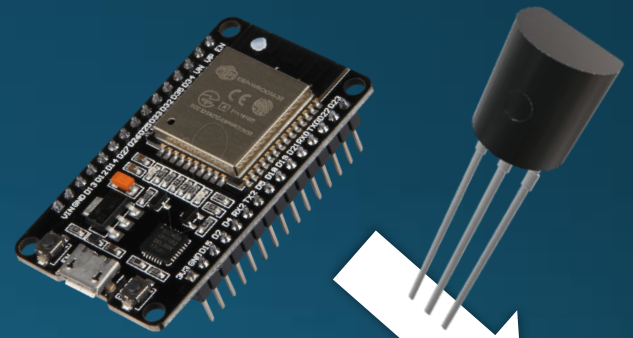


Actuateur
Transistor

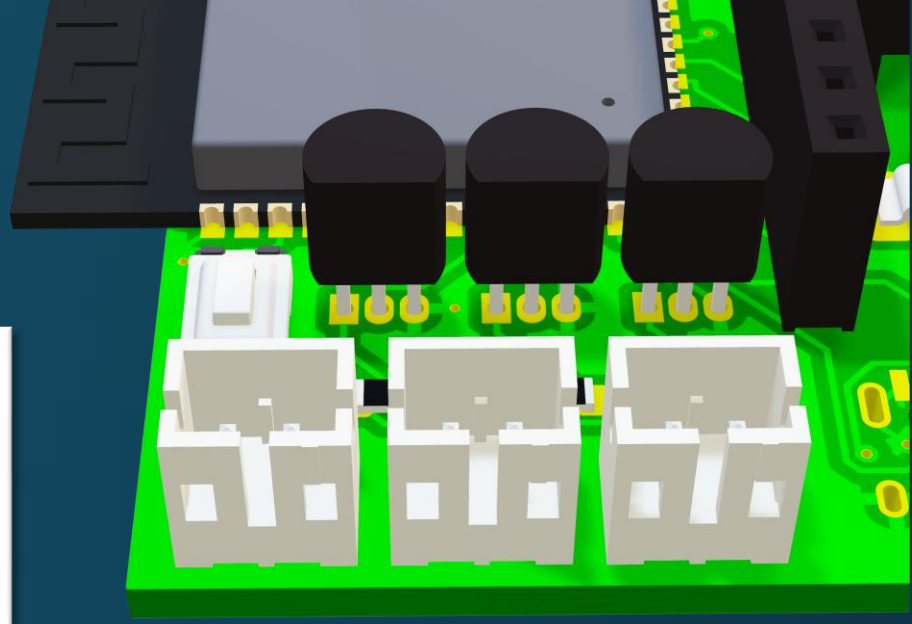
Interrupteur électronique

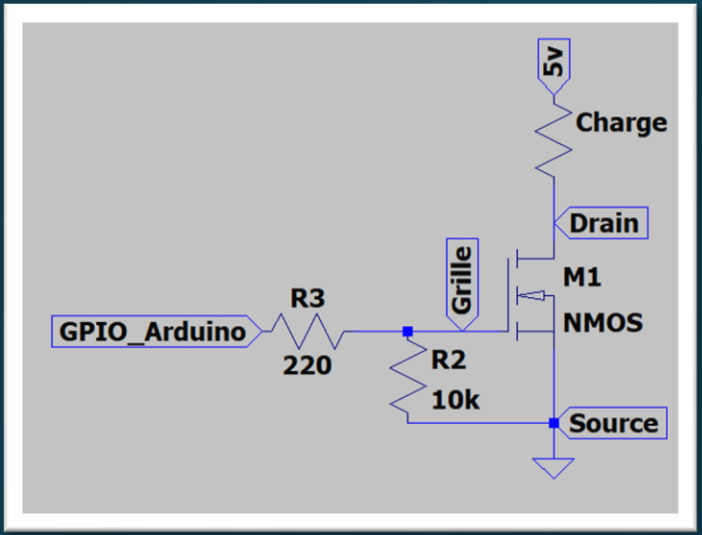


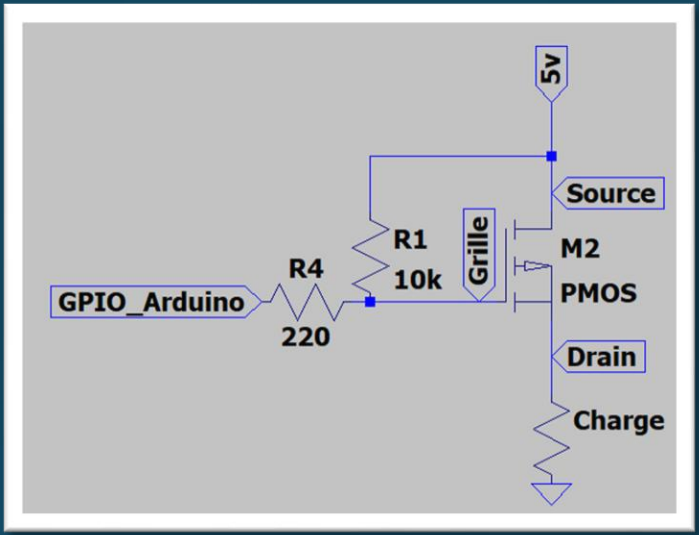




Actuateur
Transistor

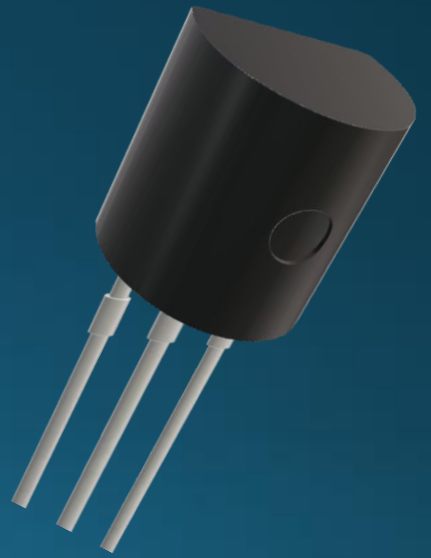




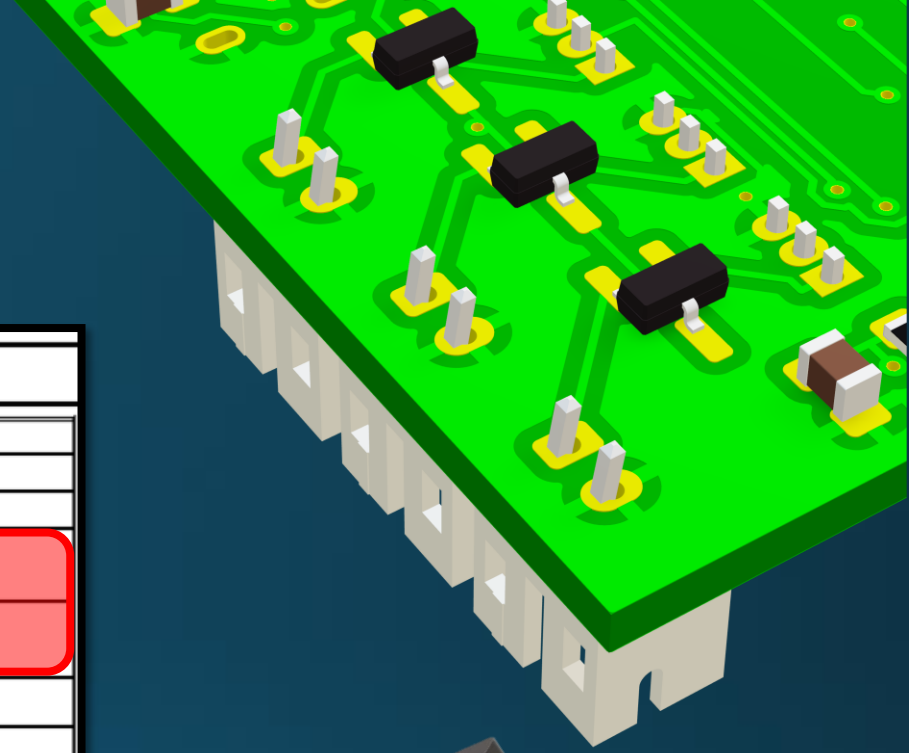


GPIO Arduino	Charge
LOW	OFF
HIGH	ON

GPIO Arduino	Charge
LOW	ON
HIGH	OFF



Actuateur
Transistor

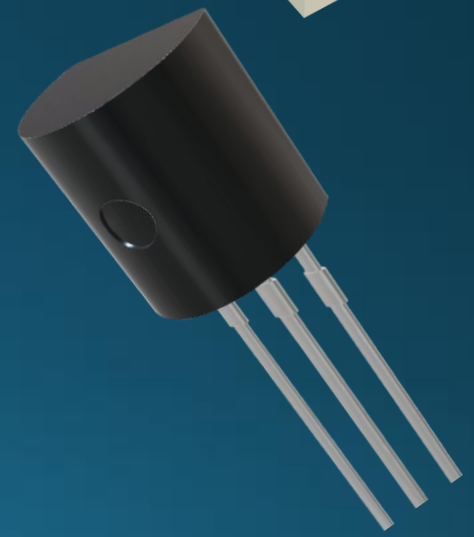


Maximum Ratings (@T_A = +25°C, unless otherwise specified.)

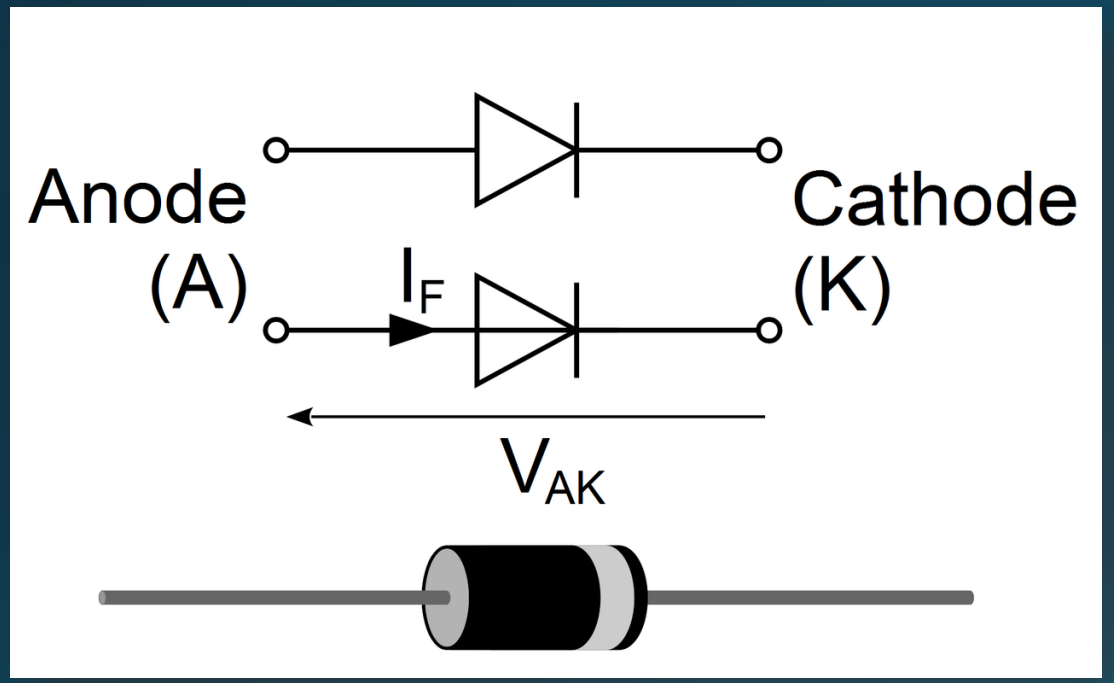
Characteristic			Symbol	Value	Unit
Drain-Source Voltage			V_{DSS}	130	V
Gate-Source Voltage			V_{GSS}	±20	V
Continuous Drain Current (Note 6) V_{GS} = 10V	Steady State	T_A = +25°C T_A = +70°C	I_D	1.0 0.8	A
	t<10s	T_A = +25°C T_A = +70°C	I_D	1.2 1.0	A
Pulsed Drain Current (10µs Pulse, Duty Cycle ≦1%)			I_{DM}	3.3	A
Maximum Body Diode Continuous Current (Note 6)			I_S	1.0	A

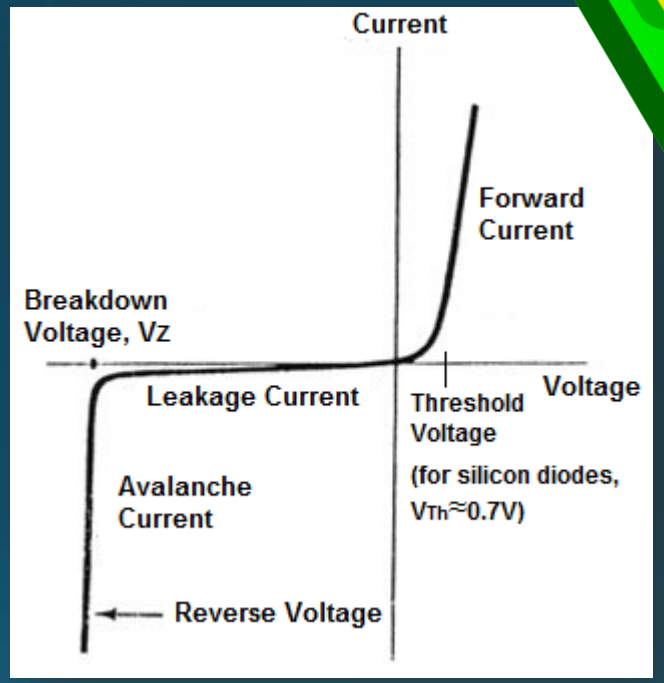
Electrical Characteristics (@T_A = +25°C, unless otherwise specified.)

Characteristic	Symbol	Min	Typ	Max	Unit	Test Condition
OFF CHARACTERISTICS (Note 7)						
Drain-Source Breakdown Voltage	BV_{DSS}	130	—	—	V	V_{GS} = 0V, I_D = 250µA
Zero Gate Voltage Drain Current	I_{DSS}	—	—	100	nA	V_{DS} = 120V, V_{GS} = 0V
Gate-Body Leakage	I_{GSS}	—	—	±100	nA	V_{GS} = ±20V, V_{DS} = 0V
ON CHARACTERISTICS (Note 7)						
Gate Threshold Voltage	$V_{GS(TH)}$	2.0	2.7	4.0	V	V_{DS} = V_{GS}, I_D = 250µA
Static Drain-Source On-Resistance	$R_{DS\ (ON)}$	—	0.41	0.75	Ω	V_{GS} = 10V, I_D = 2.0A
		—	0.43	0.85		V_{GS} = 6.0V, I_D = 2.0A
Diode Forward Voltage	V_{SD}	—	0.8	1.2	V	V_{GS} = 0V, I_S = 1.0A

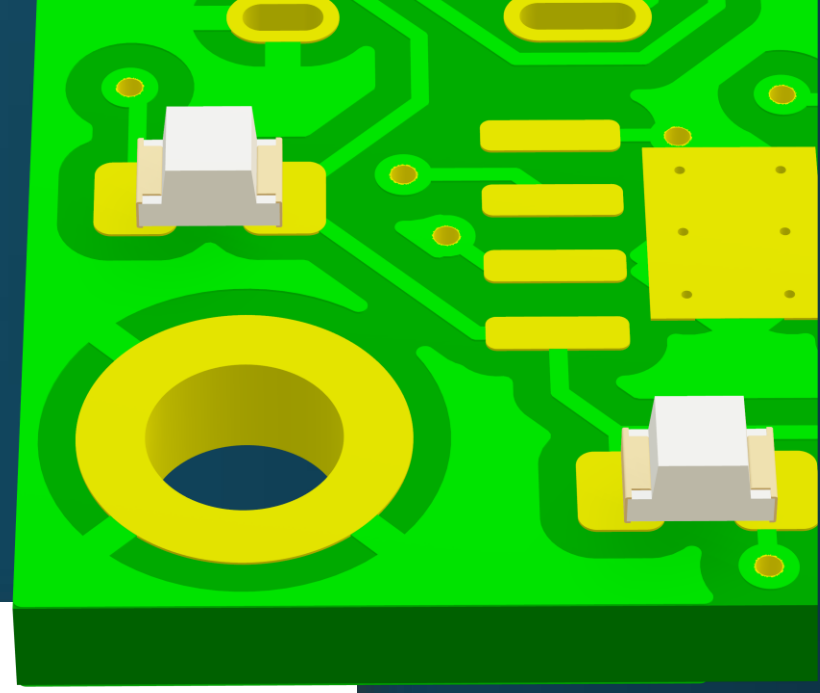


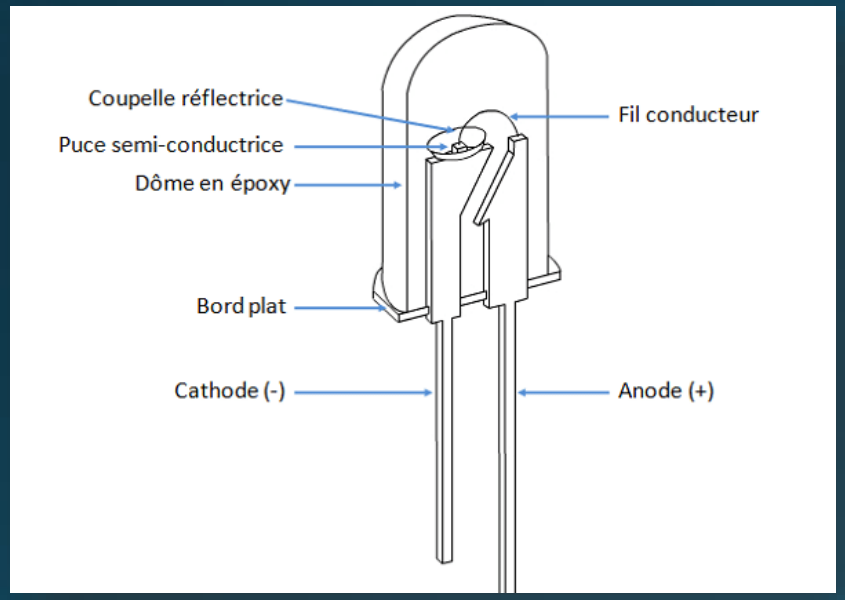
Actuateur
Diode et LED

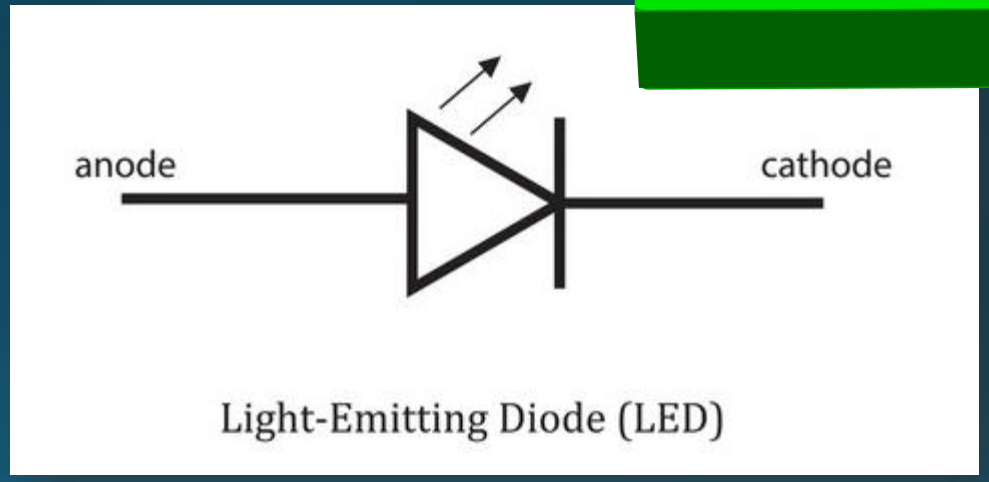




Actuateur
Diode et LED

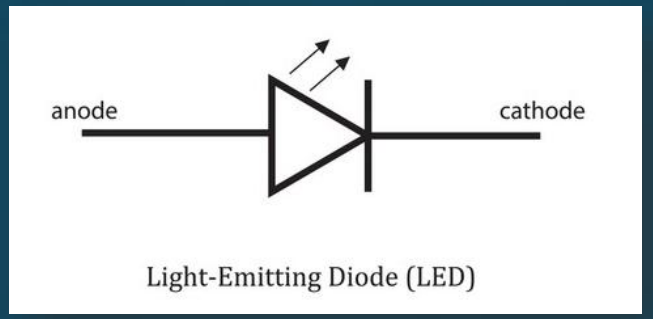


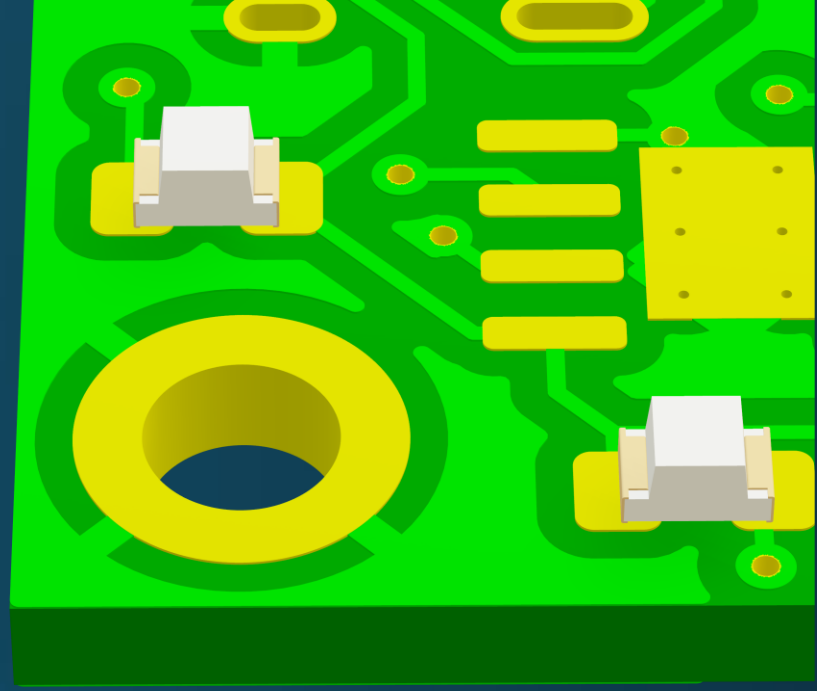




Light-Emitting Diode (LED)

Actuateur
Diode et LED



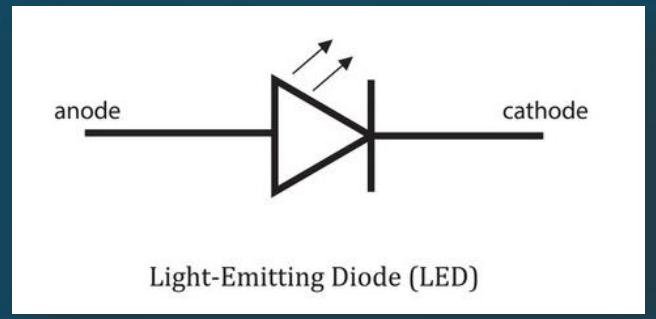


- Ne pas directement brancher à Vcc
- Résistance pour limiter le courant
- Lire datasheet de la LED
- Calculer la valeur de la résistance :

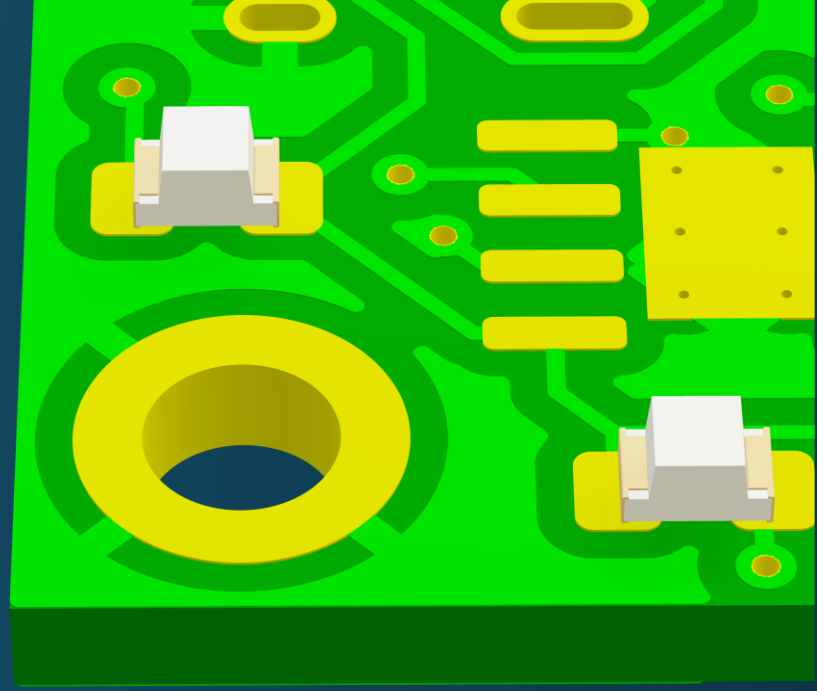
$$R = \frac{V_{alim} - V_f}{I_f}$$

- Tension directe Vf
- Courant nominal If
- Arrondir à la valeur standard de résistance la plus proche (vers le bas idéalement)

Actuateur
Diode et LED



Light-Emitting Diode (LED)



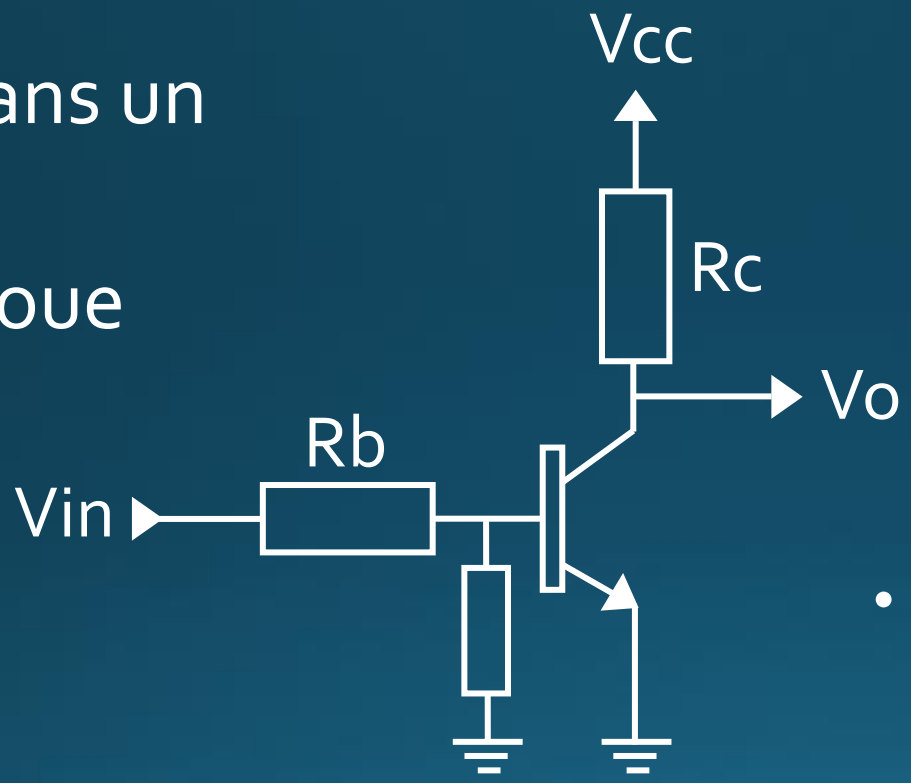
$$R = \frac{V_{alim} - V_f}{I_f}$$

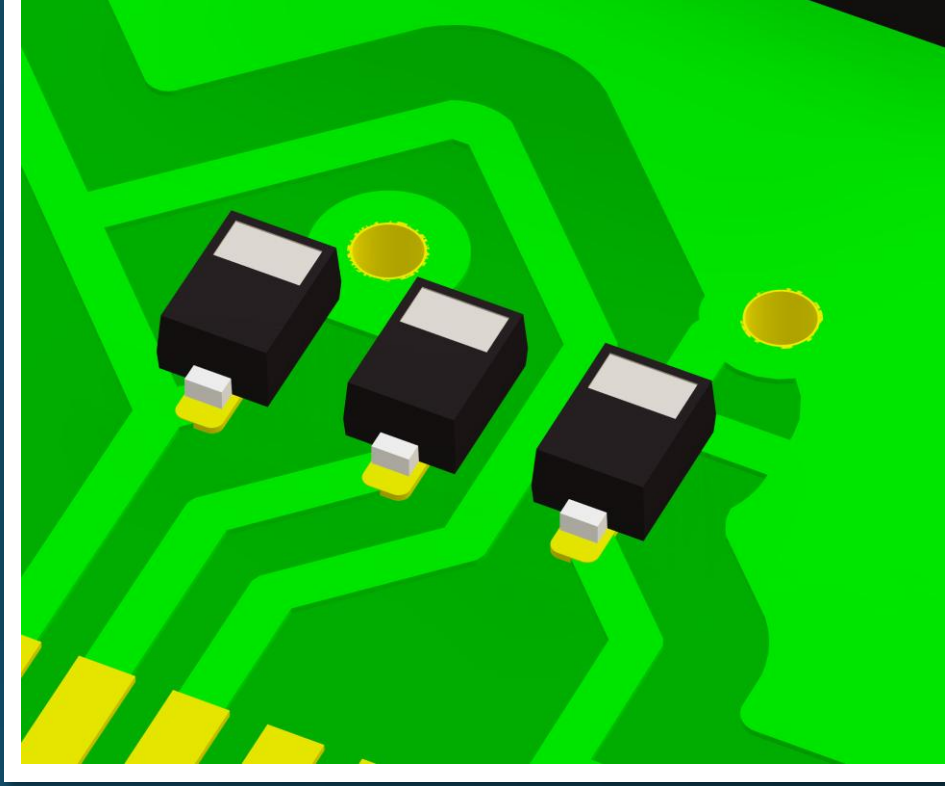
Properties		Test conditions	Value			Unit
			min.	typ.	max.	
Peak Wavelength	λ_{Peak}	20 mA		640		nm
Dominant Wavelength	λ_{Dom}	20 mA		635		nm
Luminous Intensity	I_V	20 mA	120	160		mcd
Forward Voltage	V_F	20 mA		2	2.4	V
Spectral Bandwidth	$\Delta\lambda$	20 mA		17		nm
Reverse Current	I_{REV}	5 V			10	µA
Viewing Angle Phi 0°	$2\theta_{50\%}$	20 mA		40		°

$$R = \frac{5-2.4}{0{,}02} = 130\ \Omega \approx 150\ \Omega$$

Actuateur
Diode et LED

- Courant dans un seul sens
- Diode de roue libre

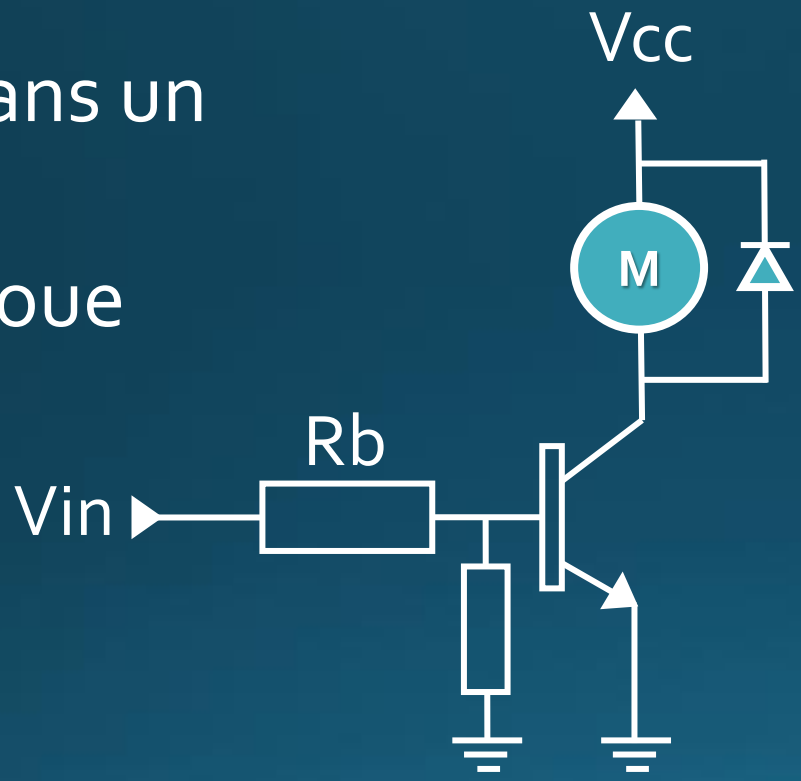


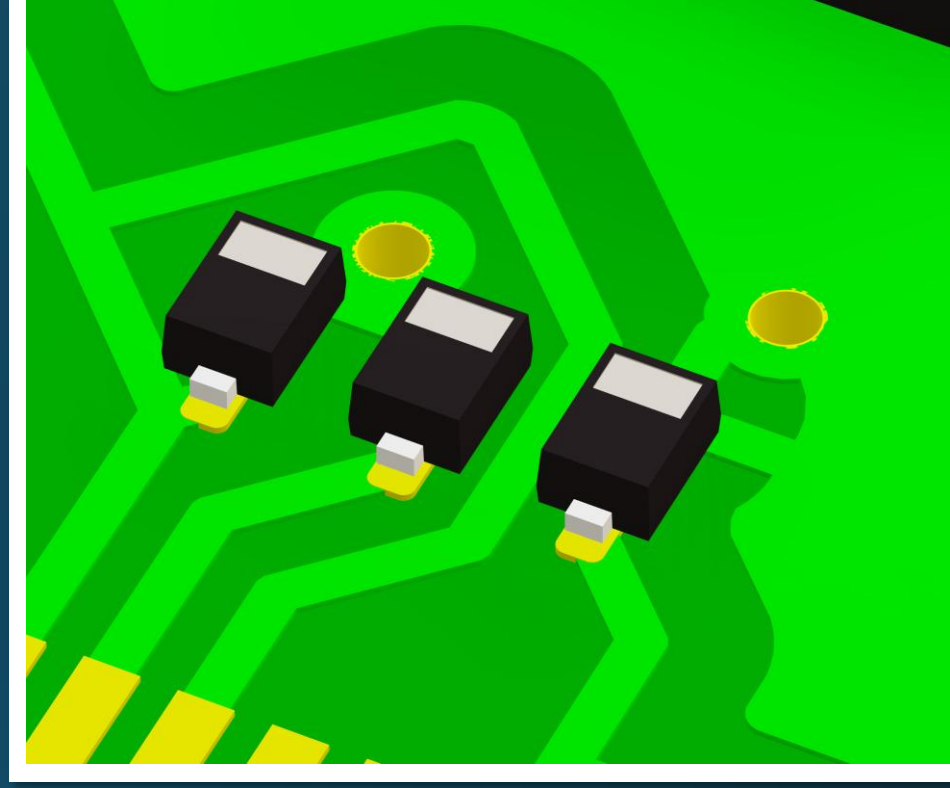


- Moteurs basse tension (5-12V)
- Diode Schottky (ex: 1N5819)

Actuateur
Diode et LED

- Courant dans un seul sens
- Diode de roue libre

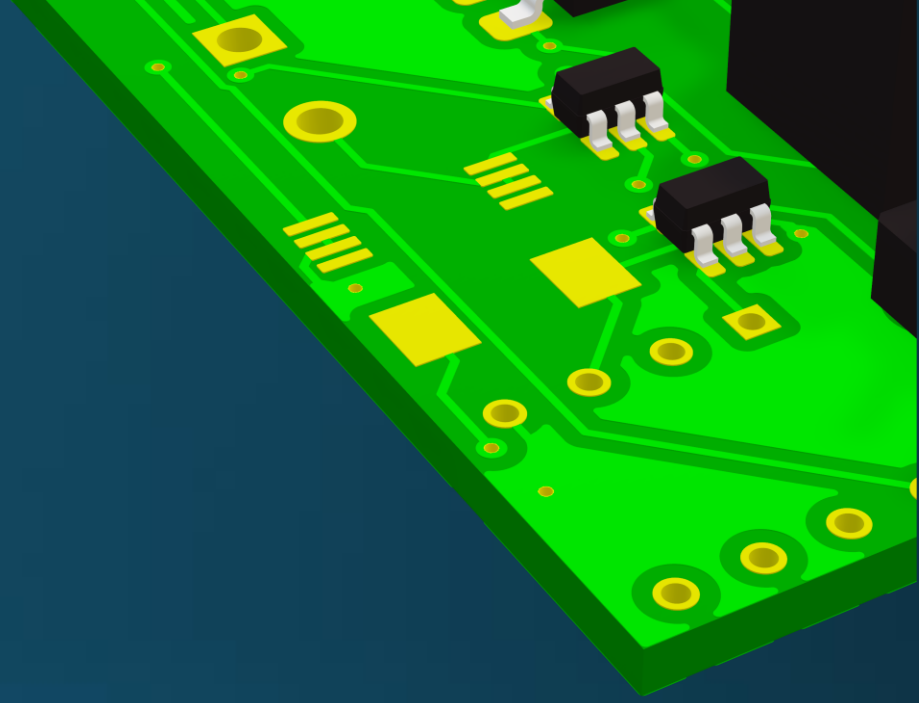




- Moteurs basse tension (5-12V)
- Diode Schottky (ex: 1N5819)

Alimentation

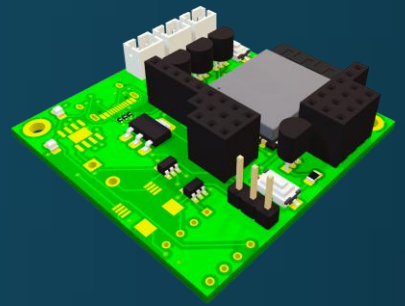
Généralités



- Fournir la puissance nécessaire (tension et courant) au système
- Prendre en considération tous les composants

Alimentation

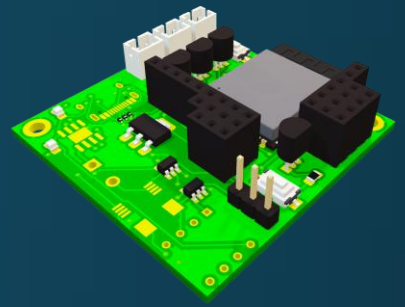
Différentes sources



Outils de labo

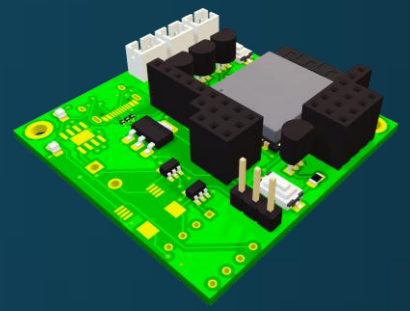
Alimentation

Différentes sources



- Composition de cellules (série et/ou parallèle)
- BMS (Battery Management System) nécessaire
- Fusibles !
- Plus d’infos lors de la démo Batterie (18/11)

Code
Bootloader

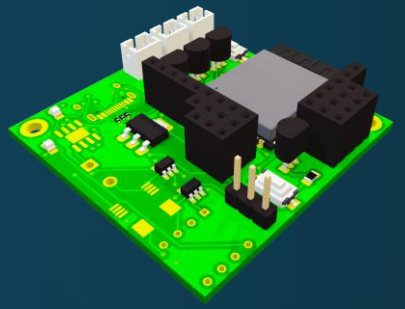


- Programme de configuration (une fois au début)
- Initialisation principale (Horloge, RAM, périphériques de base)
- Configuration de la réception d'un programme utilisateur via protocole de communication (USB, UART, SPI)
  - L'écrire dans la mémoire FLASH
  - Le lancer au démarrage
- Equivalent à un OS pour un ordinateur

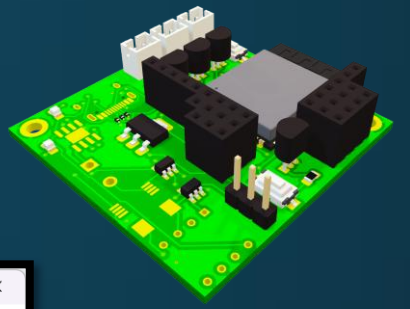
- Soit déjà installé (Arduino, ESP)
- Soit utiliser un programmateur externe (ST-link par exemple)

Code

Bootloader

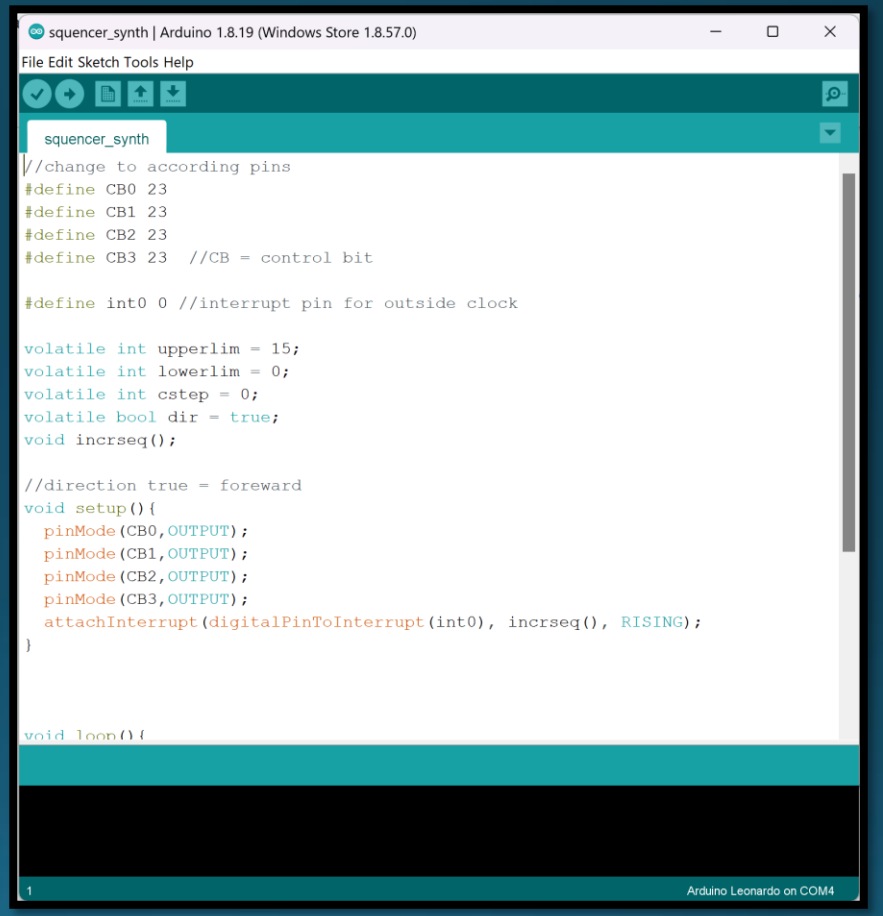


Code
Arduino IDE

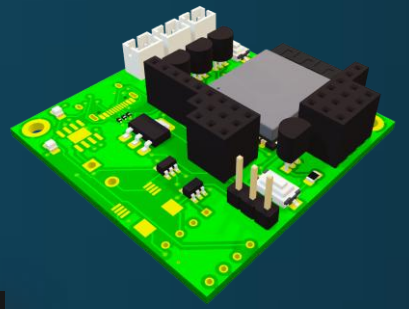


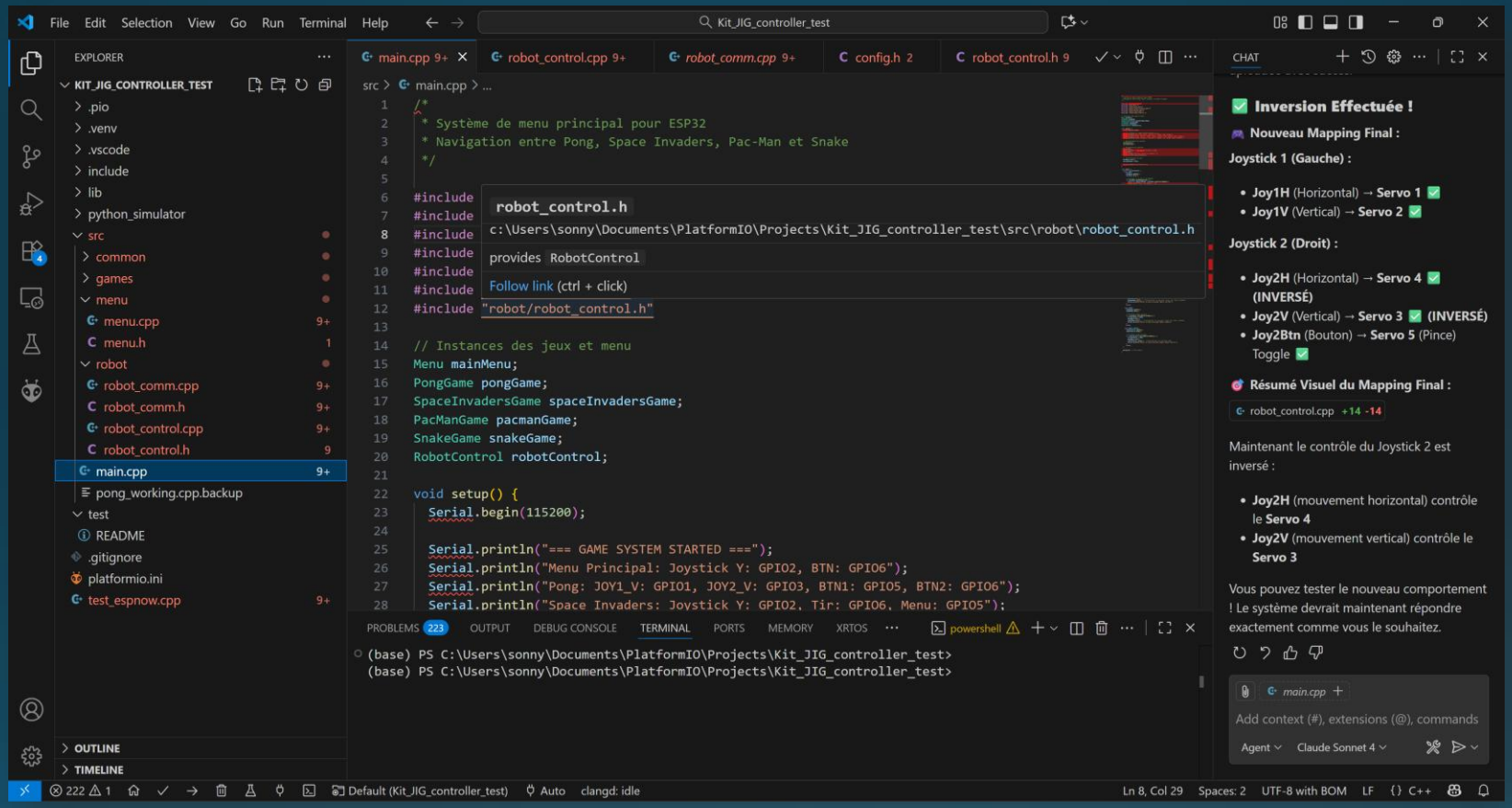
- Programme en C
- Librairies
- Connecte l'Arduino/ESP à ton ordi en USB et upload le programme

- Powersupplies: for now with USB
- Later course via Vin (external)

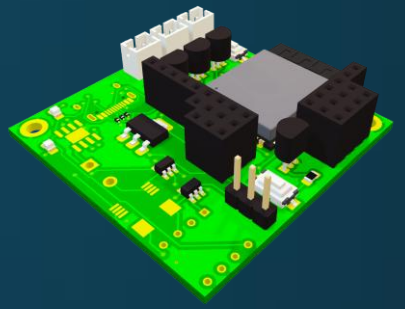


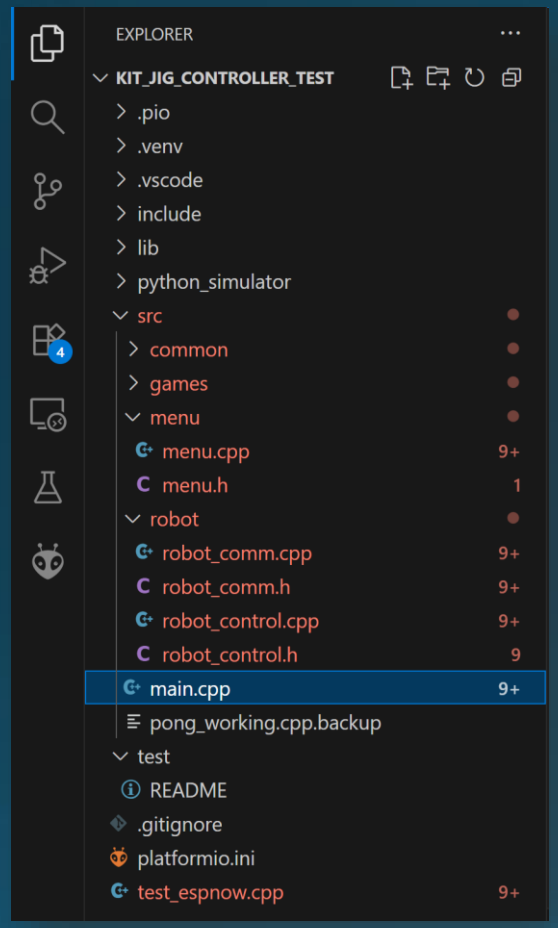
Code
VSCode




File Edit Selection View Go Run Terminal Help
Kit_JIG_controller_test
EXPLORER
KIT_JIG_CONTROLLER_TEST
.pio
.venv
.vscode
include
lib
python_simulator
src
common
games
menu
menu.cpp
menu.h
robot
robot_comm.cpp
robot_comm.h
robot_control.cpp
robot_control.h
main.cpp
pong_working.cpp.backup
test
README
.gitignore
platformio.ini
test_espnow.cpp
OUTLINE
TIMELINE
main.cpp 9+
robot_control.cpp 9+
robot_comm.cpp 9+
config.h 2
robot_control.h 9
src > main.cpp > ...
/*
* Système de menu principal pour ESP32
* Navigation entre Pong, Space Invaders, Pac-Man et Snake
*/
#include
#include
#include
#include
#include
#include "robot/robot_control.h"
robot_control.h
c:\Users\sonny\Documents\PlatformIO\Projects\Kit_JIG_controller_test\src\robot\robot_control.h
provides RobotControl
Follow link (ctrl + click)
// Instances des jeux et menu
Menu mainMenu;
PongGame pongGame;
SpaceInvadersGame spaceInvadersGame;
PacManGame pacmanGame;
SnakeGame snakeGame;
RobotControl robotControl;
void setup() {
Serial.begin(115200);
Serial.println("=== GAME SYSTEM STARTED ===");
Serial.println("Menu Principal: Joystick Y: GPIO2, BTN: GPIO6");
Serial.println("Pong: JOY1_V: GPIO1, JOY2_V: GPIO3, BTN1: GPIO5, BTN2: GPIO6");
Serial.println("Space Invaders: Joystick Y: GPIO2, Tir: GPIO6, Menu: GPIO5");
PROBLEMS 223
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
MEMORY
XRTOS
powershell
(base) PS C:\Users\sonny\Documents\PlatformIO\Projects\Kit_JIG_controller_test>
(base) PS C:\Users\sonny\Documents\PlatformIO\Projects\Kit_JIG_controller_test>
CHAT
Inversion Effectuée !
Nouveau Mapping Final :
Joystick 1 (Gauche) :
Joy1H (Horizontal) → Servo 1
Joy1V (Vertical) → Servo 2
Joystick 2 (Droit) :
Joy2H (Horizontal) → Servo 4 (INVERSÉ)
Joy2V (Vertical) → Servo 3 (INVERSÉ)
Joy2Btn (Bouton) → Servo 5 (Pince) Toggle
Résumé Visuel du Mapping Final :
robot_control.cpp +14 -14
Maintenant le contrôle du Joystick 2 est inversé :
Joy2H (mouvement horizontal) contrôle le Servo 4
Joy2V (mouvement vertical) contrôle le Servo 3
Vous pouvez tester le nouveau comportement ! Le système devrait maintenant répondre exactement comme vous le souhaitez.
main.cpp
Add context (#), extensions (@), commands
Agent
Claude Sonnet 4
Default (Kit_JIG_controller_test)
Auto
clangd: idle
Ln 8, Col 29
Spaces: 2
UTF-8 with BOM
LF
C++

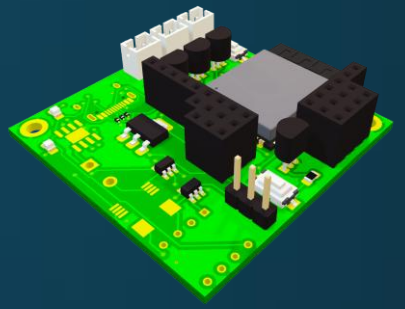
Code
VSCode





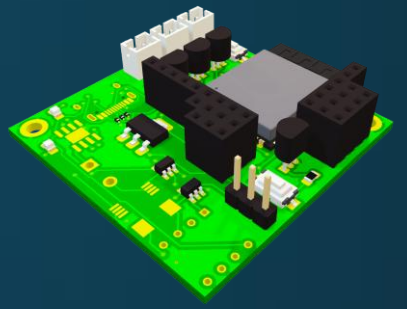
- Workspace
- Organisation du projet
- Arborescence des étapes
- Lien avec Github

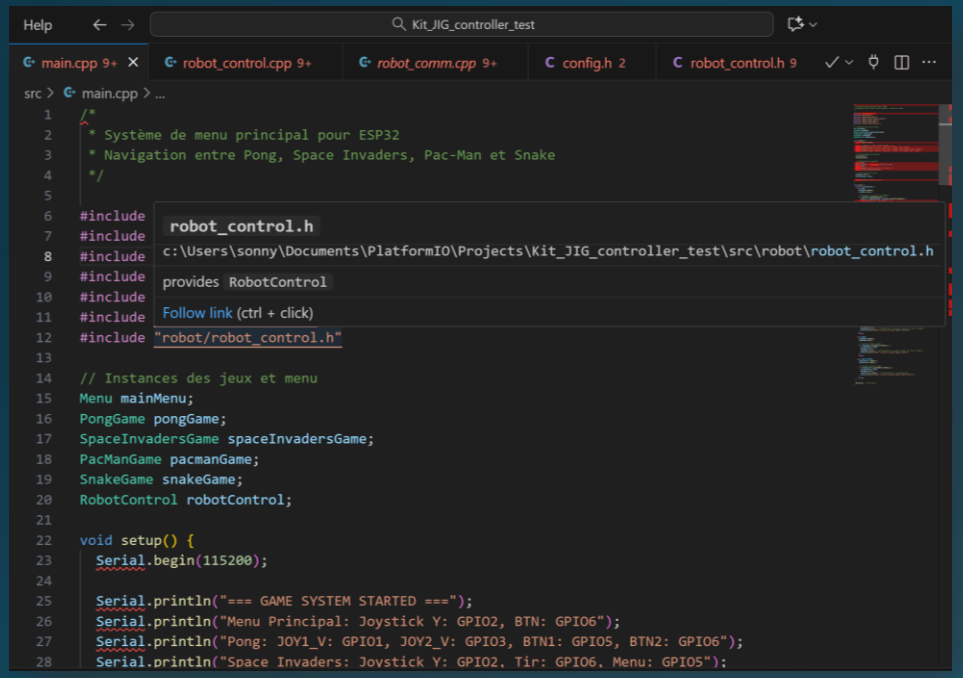
Code
VSCode



- PlatformIO
- Compatibilité avec Arduino/ESP
- Coder comme sur Arduino IDE

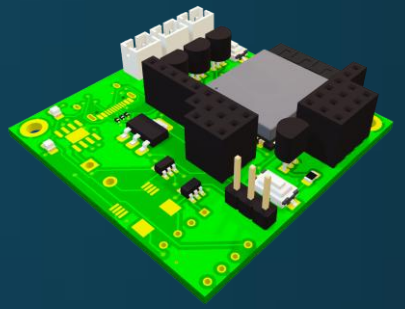
Code
VSCode

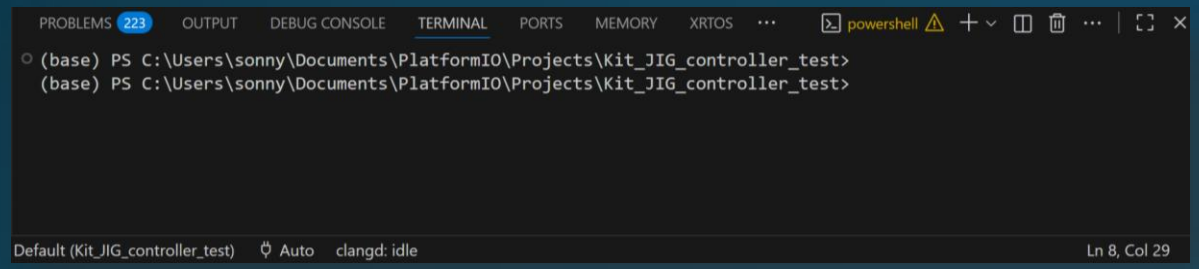




- Interface de programmation
- Raccourcis intuitifs

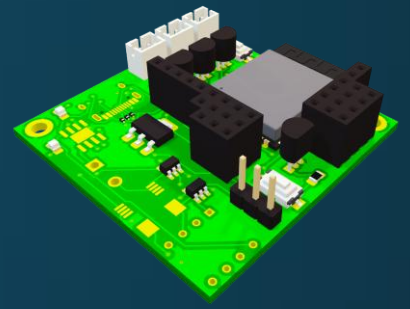
Code
VSCode





- Terminal
- Console output
- Problems

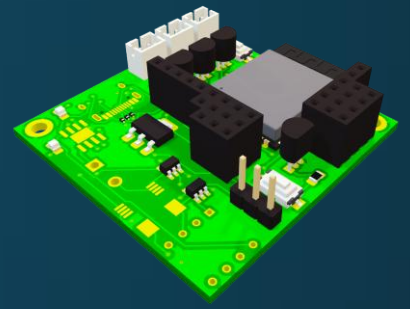
Code
VSCode



- IA intégrée
- Mode Agent
- Monsieur Sonnet

Tests

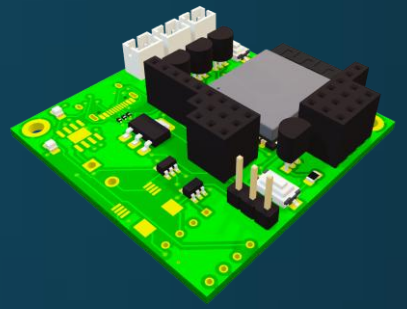
Généralités



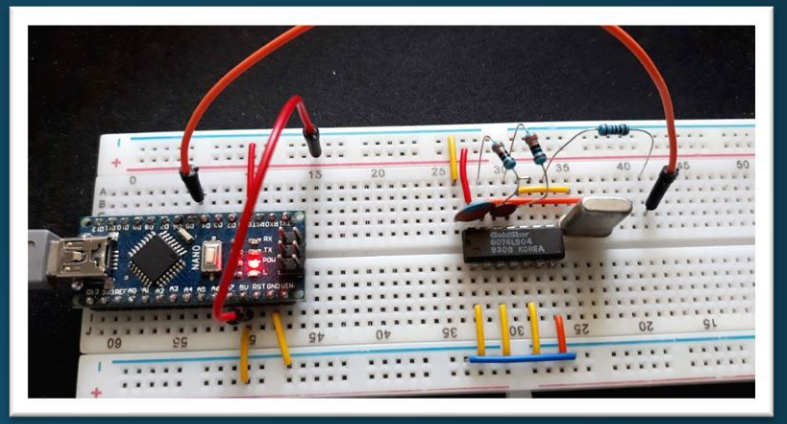
- Important de tester le circuit
  - Risques de court-circuits
  - Ne pas endommager un composant
- Prototyper puis tester
- Maîtrise des outils de labo principaux

Tests

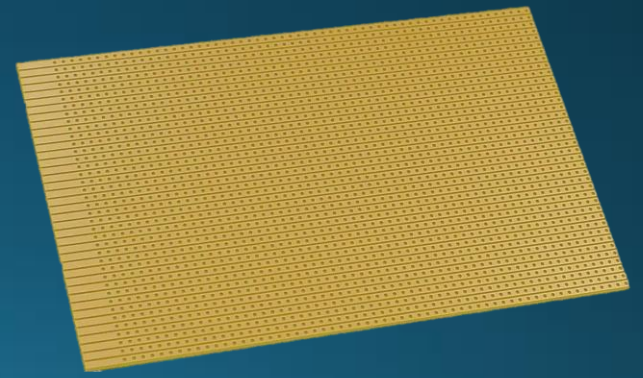
Breadboard



- Pas de soudure
- Prototypage rapide
- Simple à modifier, mesurer et déverminer



- Version petit circuit : Stripboard

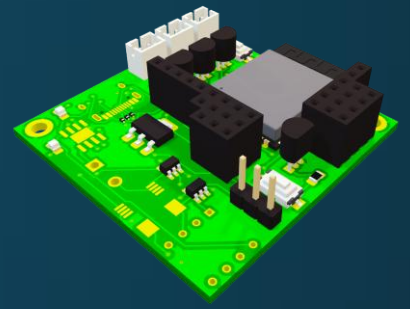


Tests

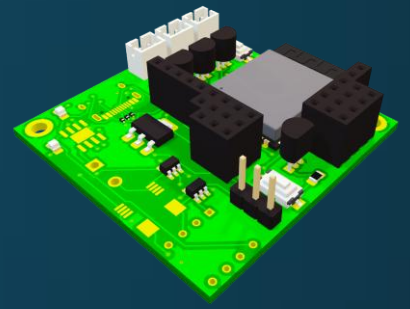
Outils de labo

Tests

Outils de labo



Tests
Outils de labo

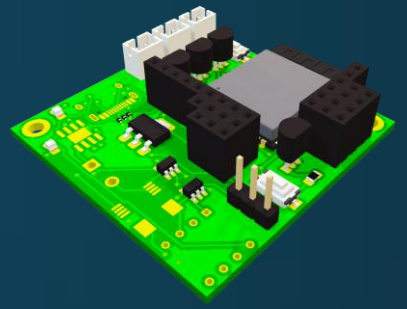


Alimentation

- Générateur de tension
- Limiteur de courant
  - Attention au courant max que peut tirer votre circuit
- Attention saut de tension
  - Allumer puis brancher

Tests

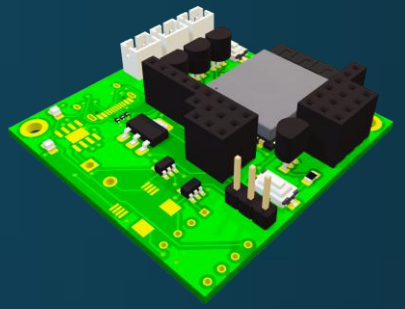
Outils de labo



- Générateur de fonctions
- Création d'un signal
  - Carré, triangulaire, sinusoïdal…

Tests

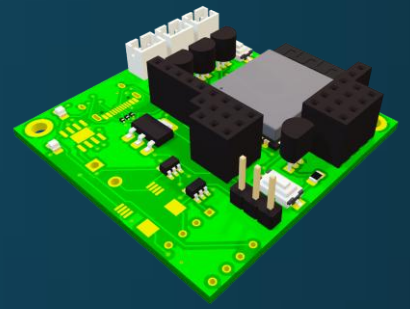
Outils de labo



- Oscilloscope
- Mesure des signaux
- Plusieurs channels
- Réglages
  - Echelles (voltage, courant, temps)
  - Trigger: début du signal sur l'écran
  - etc

Tests

Outils de labo



Driver logic:

180 degrees:

T=0.5ms -- -- Rotate 0 °

T=1.0ms -- -- Rotate 45 °

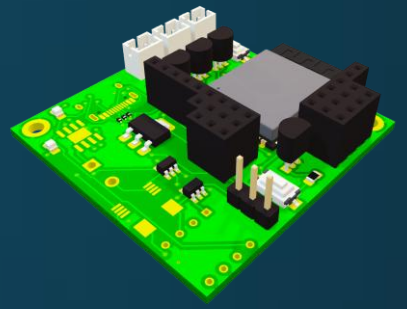
T=1.5ms -- -- Rotate 90 °

T=2.0ms -- -- Rotate 135 °

T=2.5ms -- -- Rotate 180 °

Tests

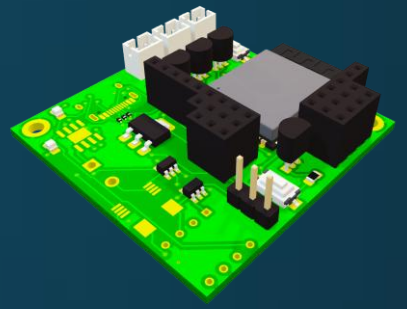
Outils de labo



- Multimètre
- Mesure des grandeurs électriques
  - Tension
  - Courant
  - Résistance
  - Capacité
  - Etc
- Mode Court-Circuit

Tests

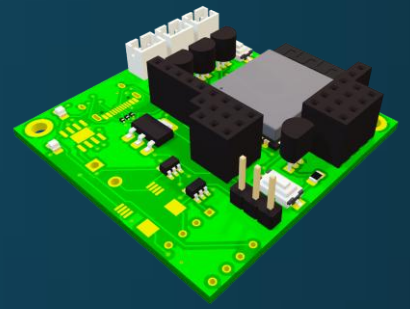
Outils de labo



- Attention aux connexions
- Utiliser les bons pins en fonction de la mesure
- Mesurer une tension si vous êtes branchés en courant = circuit cramé et court-circuit

Tests

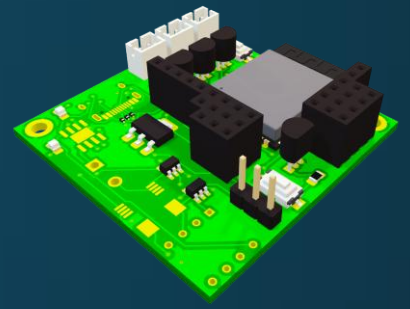
Outils de labo



- Fer à souder
- Ca sert à souder…
- Connexions plus clean
- Moins de bruit qu’une breadboard

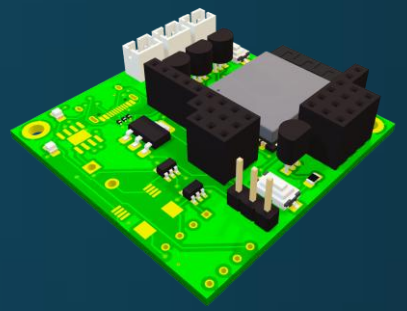
Tests

Erreurs fréquentes

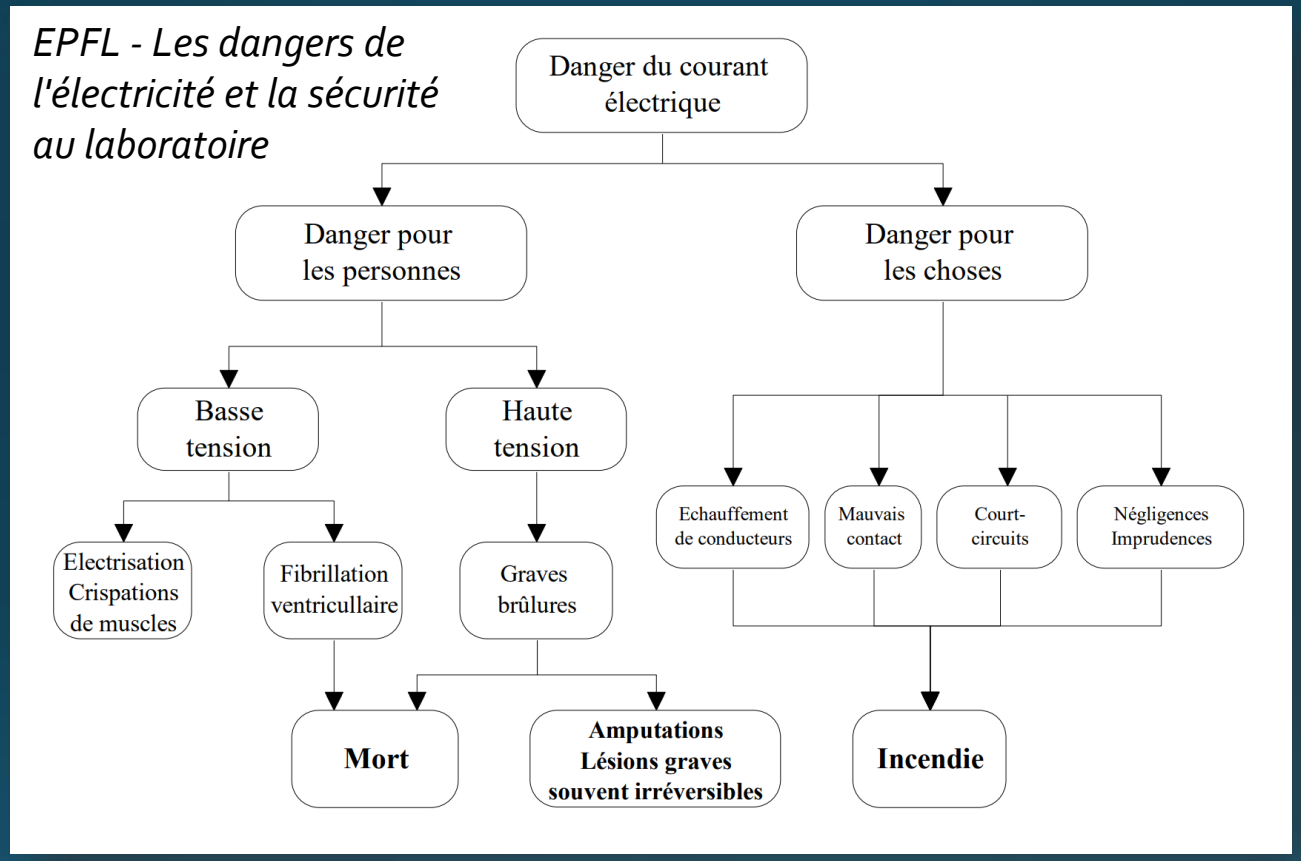


- LED à l'envers
- Masse non commune
- Court-circuit
- Alimentation insuffisante
- Mauvaise lecture de valeur de résistance (code couleur)
  - Bonne pratique de remesurer
- Bruit sur les signaux analogiques
- Capteur alimenté en 5 V alors qu'il attend 3,3 V
  - DATASHEET
  - Toujours tester avec un multimètre avant de brancher

Tests
Dangers



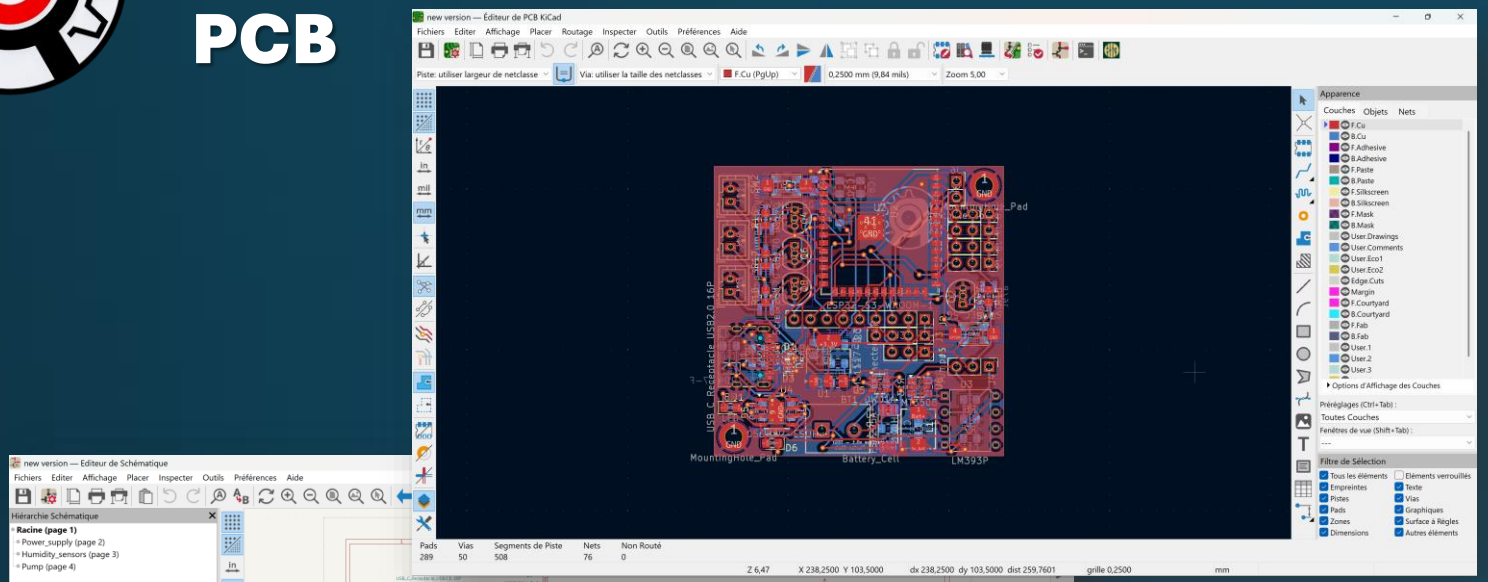
EPFL - Les dangers de l'électricité et la sécurité au laboratoire

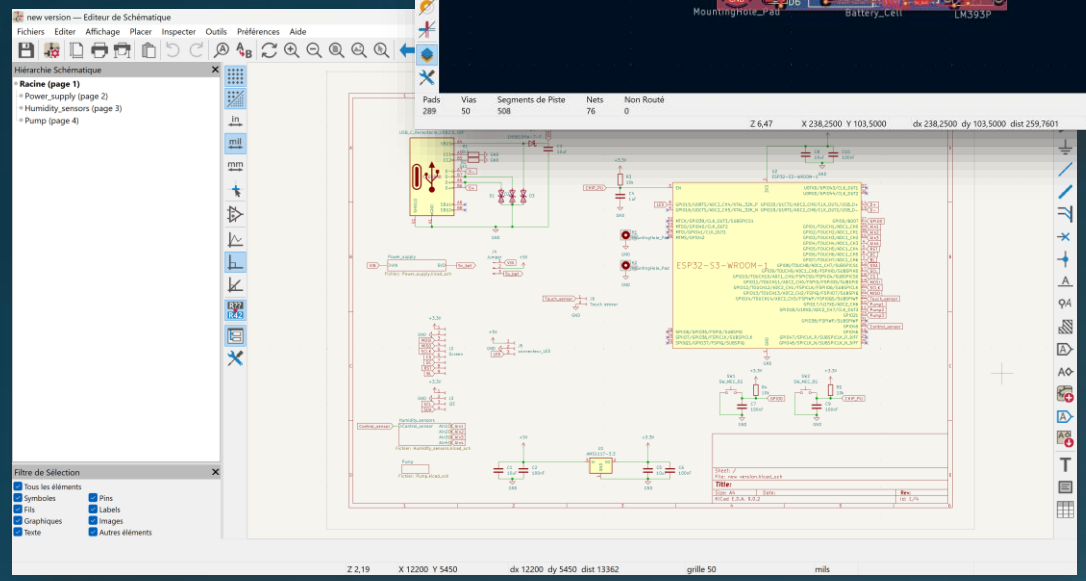


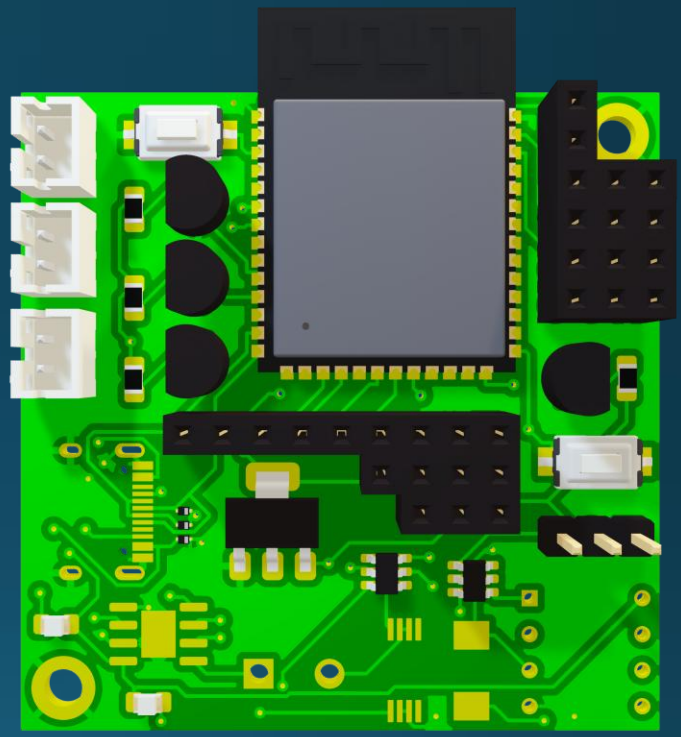
Produit fini

PCB

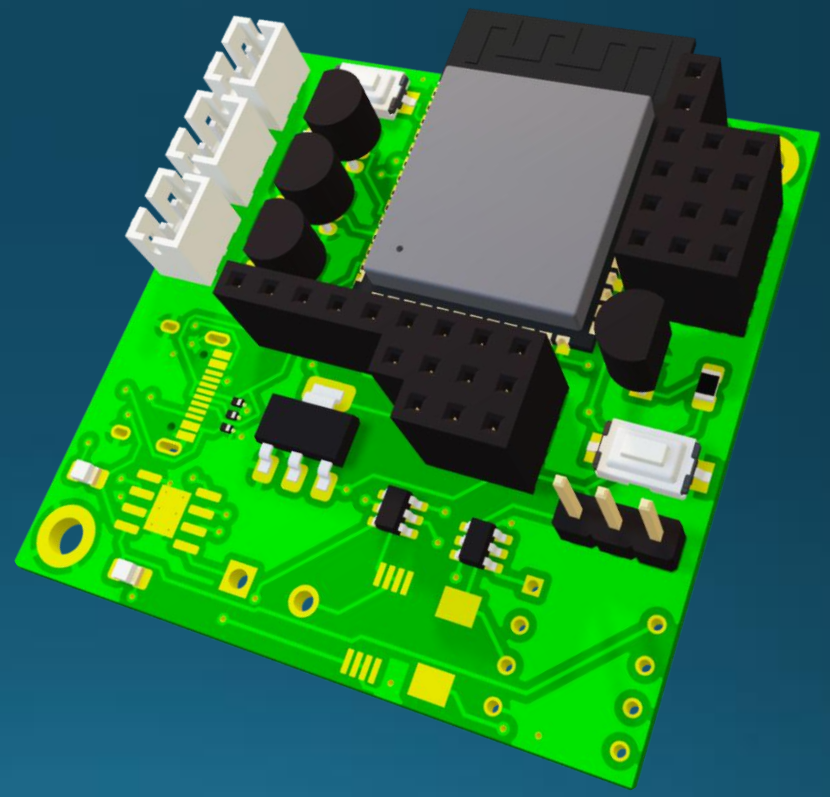
Démo PCB design 04/11







Merci de votre attention



Démo PCB design

04/11

Fin des inscriptions Grand Concours 31/10