

PROJET

Commande passage a niveau en zone de stockage

Gérer un passage à niveau en zone de stockage.

Contraintes:

Deux capteurs

Obligation de quitter les capteurs en entrée et sortie du passage à niveau.

Le dossier comprend:

Programmes Arduino (V1 et V2).

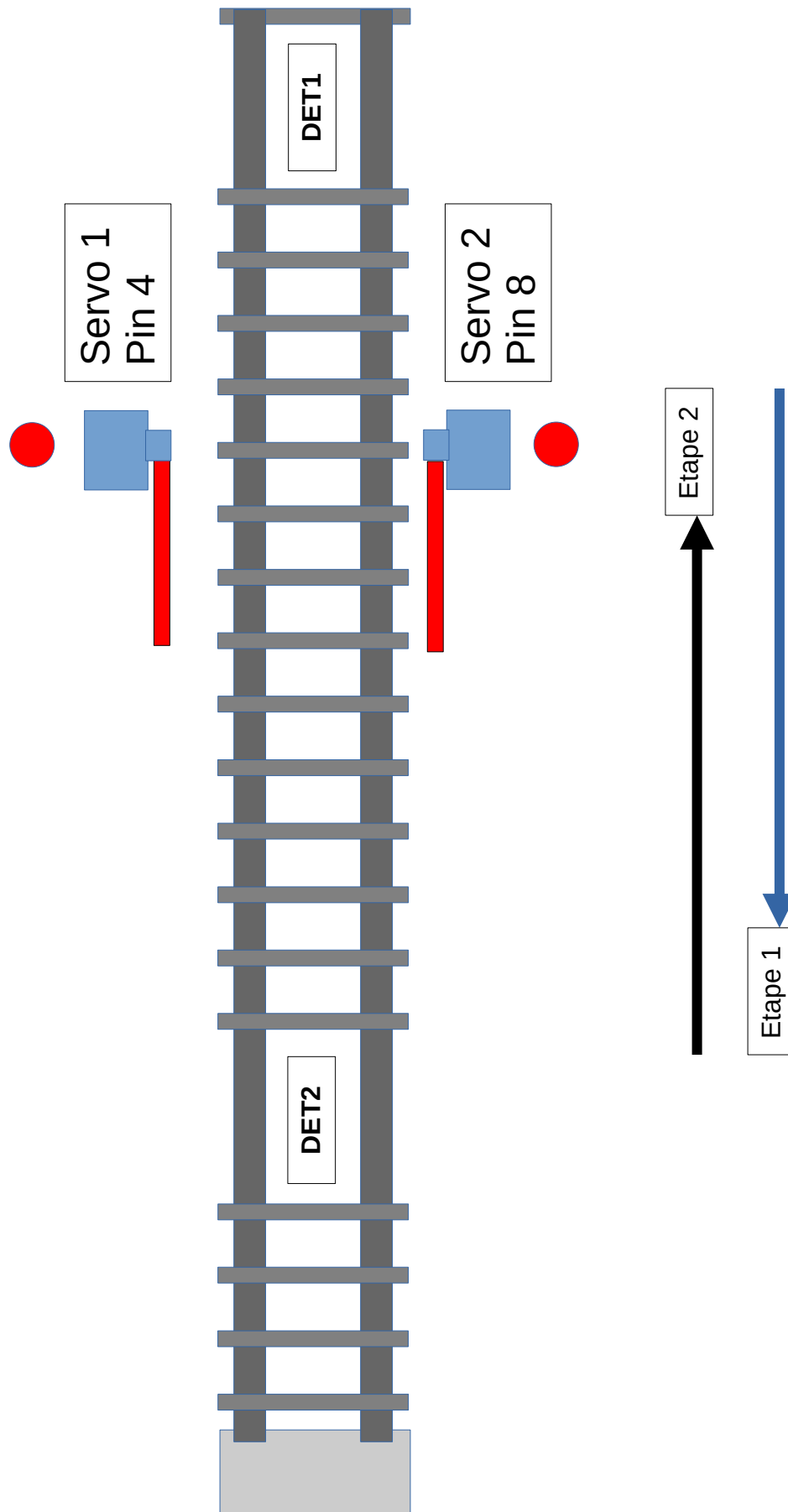
Schéma de câblage en PDF.

Diverses documentations.

Liste du matériel.

Dossier Kicad en Gerber et PDF.

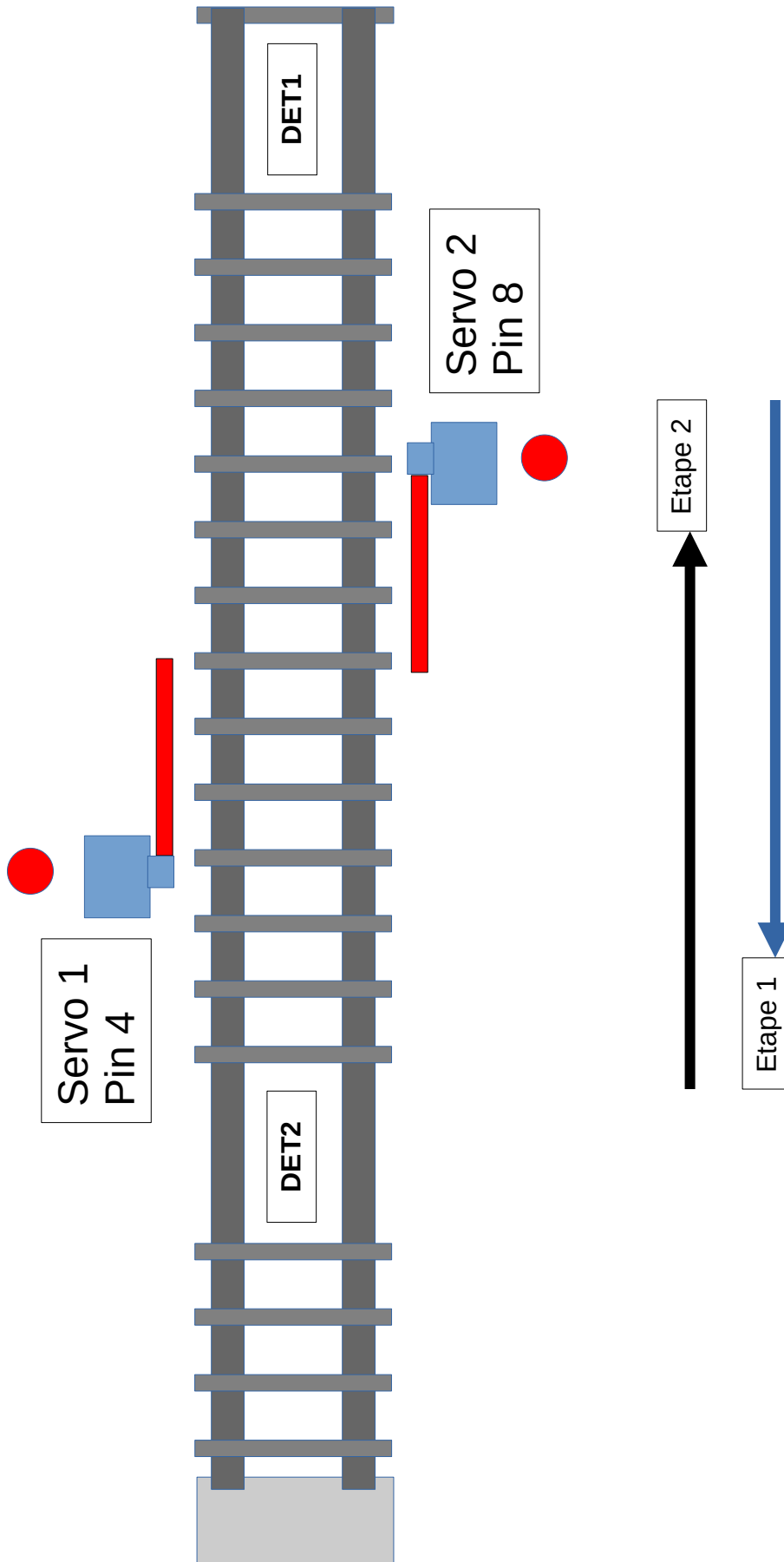
Passage a niveau avant zone de stockage



Pour chaque déplacement, ne pas
rester sur le capteur

programme_v1

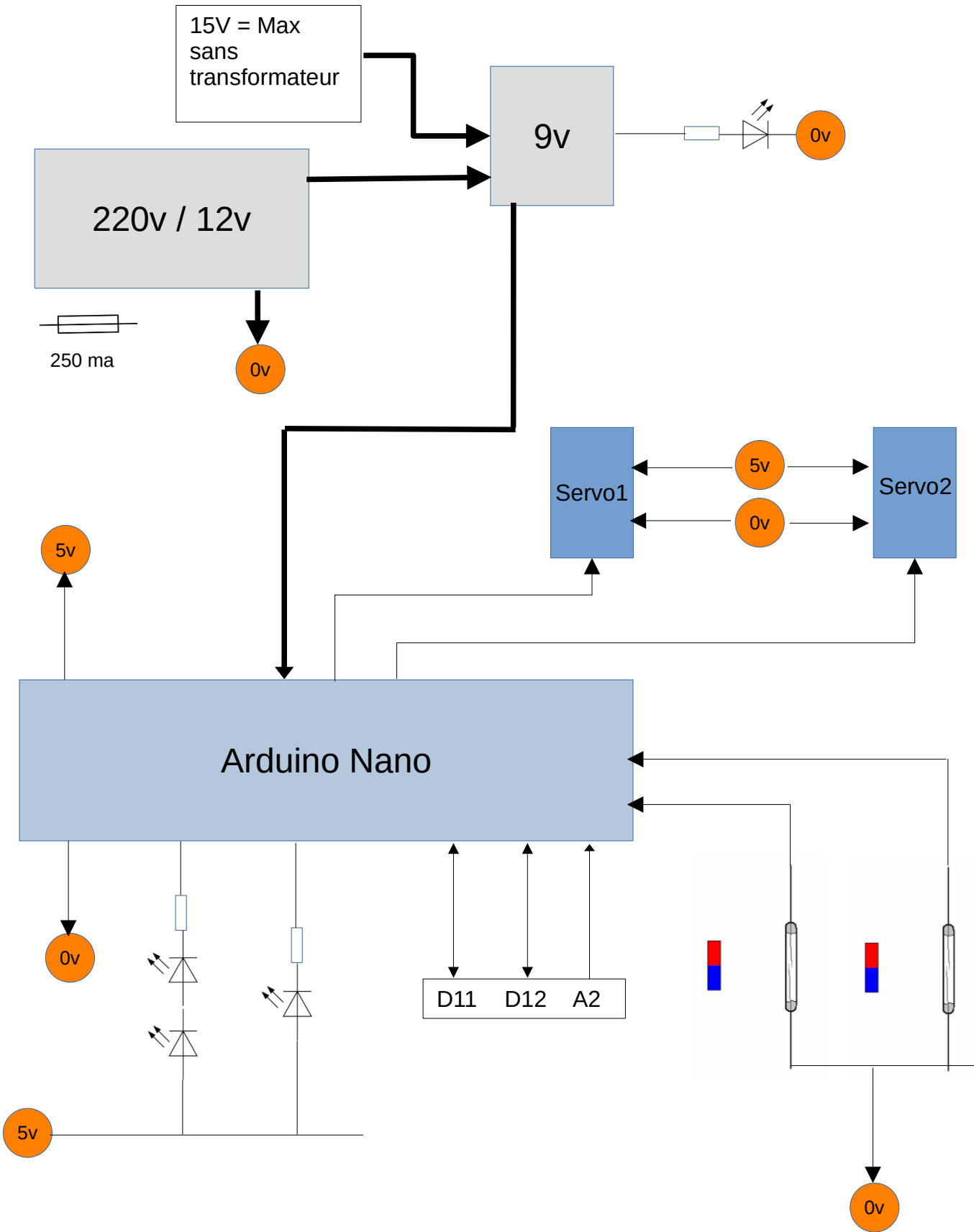
Passage a niveau avant zone de stockage



Pour chaque déplacement, ne pas
rester sur le capteur

programme_v2

Arduino – Commande passage a niveau en zone de stockage

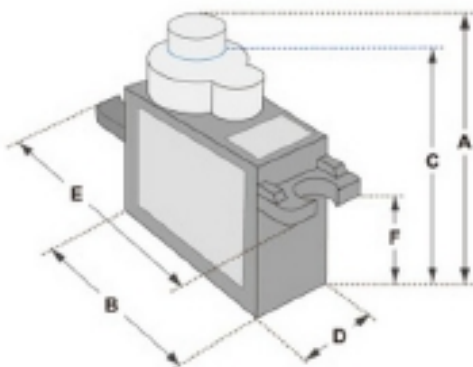


SERVO MOTOR SG90

DATA SHEET



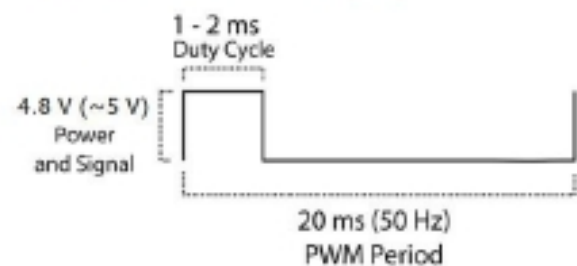
Tiny and lightweight with high output power. Servo can rotate approximately 180 degrees (90 in each direction), and works just like the standard kinds but smaller. You can use any servo code, hardware or library to control these servos. Good for beginners who want to make stuff move without building a motor controller with feedback & gear box, especially since it will fit in small places. It comes with a 3 horns (arms) and hardware.



Position "0" (1.5 ms pulse) is middle, "90" (~2ms pulse) is middle, is all the way to the right, "-90" (~1ms pulse) is all the way to the left.

Dimensions & Specifications	
A (mm) :	32
B (mm) :	23
C (mm) :	28.5
D (mm) :	12
E (mm) :	32
F (mm) :	19.5
Speed (sec) :	0.1
Torque (kg-cm) :	2.5
Weight (g) :	14.7
Voltage :	4.8 - 6

PWM=Orange (⏏)
Vcc=Red (+)
Ground=Brown (-)



Plus d'info :[ICI](http://www.servosmotors.com/)

Câblage SG90

Marron	Noir	GND
Rouge	Rouge	+5v
Orange	Jaune	cde

Librairie Servo

La bibliothèque Servo prend en charge jusqu'à 12 moteurs sur la plupart des cartes Arduino et 48 sur l'Arduino Mega. Sur les cartes autres que la Mega, l'utilisation de la bibliothèque désactive la fonctionnalité `analogWrite()` (PWM) sur les broches 9 et 10, qu'il y ait ou non un servo sur ces broches.

Sur le Mega, jusqu'à 12 servos peuvent être utilisés sans interférer avec la fonctionnalité PWM ; l'utilisation de 12 à 23 moteurs désactivera le PWM sur les broches 11 et 12.

Plus d'info : [ICI](#)

`attach()`

`servo.attach(pin)`

`servo.attach(pin, min, max)`

Attachez la variable servo à une broche. Notez que dans Arduino IDE 0016 et versions antérieures, la bibliothèque Servo prend en charge les servos sur seulement deux broches : 9 et 10.

servo: une variable de type Servo

Pin: le numéro de la broche à laquelle le servo est attaché

Min: min (facultatif) : la largeur d'impulsion, en microsecondes, correspondant à l'angle minimum (0 degré) sur le servo (par défaut : 544)

Max : (optional): the pulse width, in microseconds, corresponding to the maximum (180 degree) angle on the servo (defaults to 2400)

write()

servo.write(angle)

servo: a variable of type Servo

angle: the value to write to the servo, from 0 to 180

Écrit une valeur sur le servo, contrôlant l'arbre en conséquence. Sur un servo standard, cela définira l'angle de l'arbre (en degrés), déplaçant l'arbre vers cette orientation. Sur un servo à rotation continue, cela définira la vitesse du servo (0 étant la pleine vitesse dans un sens, 180 étant la pleine vitesse dans l'autre et une valeur proche de 90 signifiant aucun mouvement)

writeMicroseconds()

servo.writeMicroseconds(us)

Écrit une valeur en microsecondes (us) sur le servo, contrôlant l'arbre en conséquence. Sur un servo standard, cela définira l'angle de l'arbre. Sur les servos standard, une valeur de paramètre de 1 000 correspond à fond dans le sens inverse des aiguilles d'une montre, 2 000 à fond dans le sens des aiguilles d'une montre et 1 500 au milieu.

A noter que certains constructeurs ne suivent pas cette norme de très près et que les servos répondent souvent à des valeurs comprises entre 700 et 2300. N'hésitez pas à augmenter ces points finaux jusqu'à ce que le servo ne continue plus à augmenter sa portée. Notez cependant que tenter de faire passer un servo au-delà de ses points limites (souvent indiqué par un grognement) est un état à courant élevé et doit être évité.

Les servos à rotation continue répondront à la fonction writeMicrosecond d'une manière analogue à la fonction d'écriture.

read()

servo.read()

Lire l'angle actuel du servo (la valeur passée au dernier appel à write())

attached()

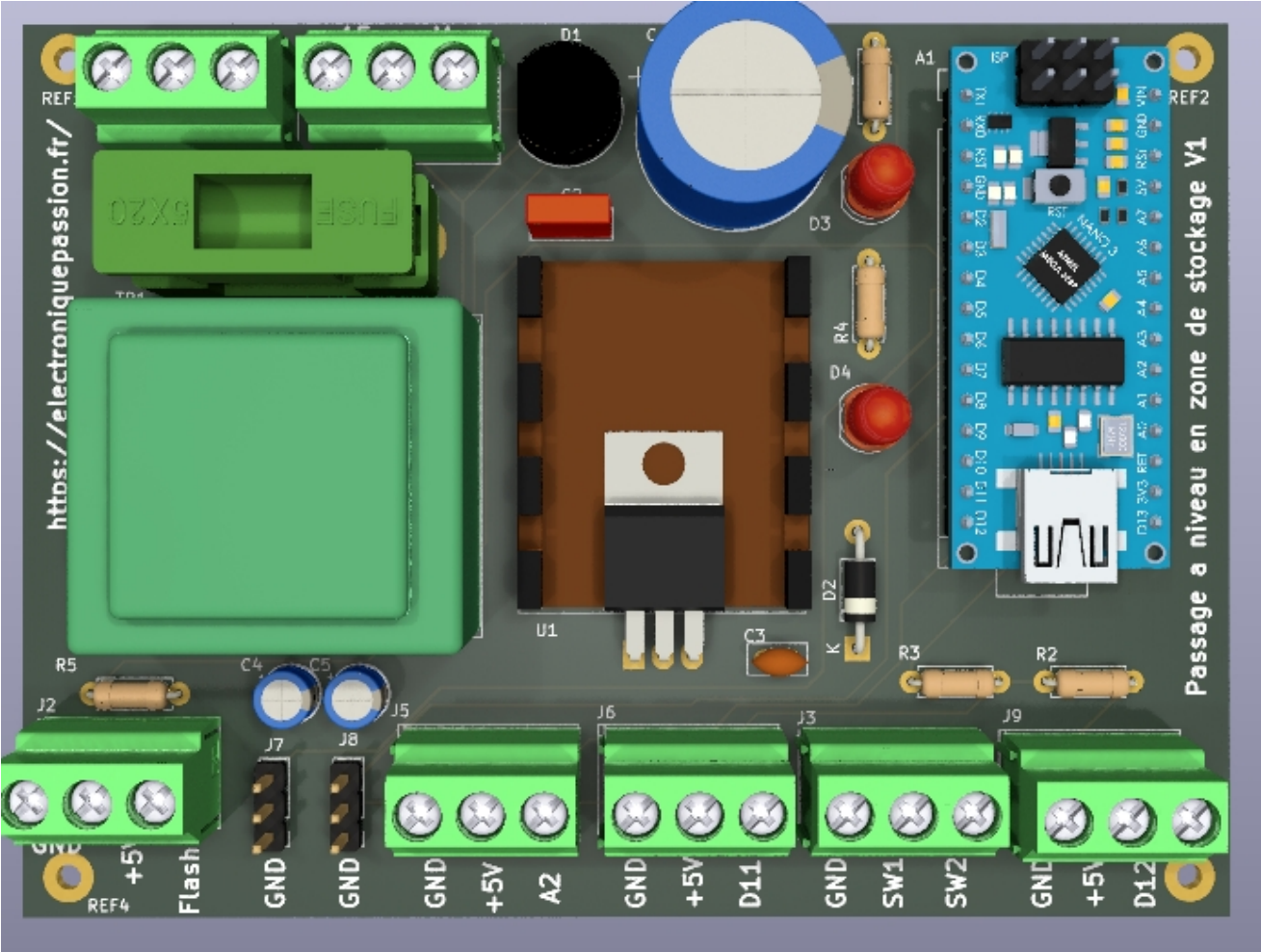
servo.attached()

Vérifiez si la variable servo est attachée à une broche..

detach()

servo.detach()

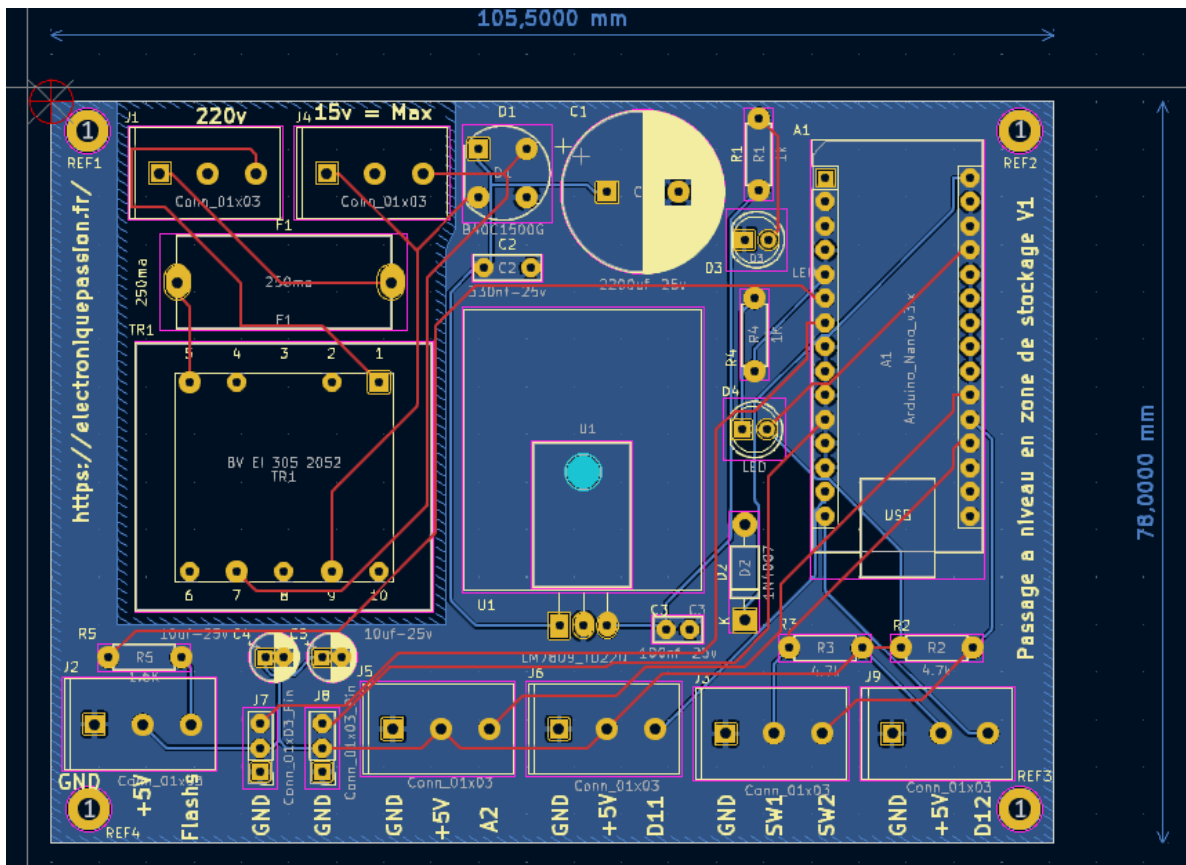
Détachez la variable Servo de sa broche. Si toutes les variables servo sont détachées, alors les broches 9 et 10 peuvent être utilisées pour la sortie PWM avec analogWrite().



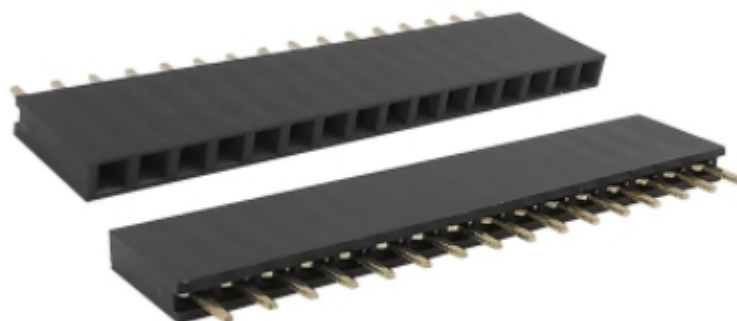
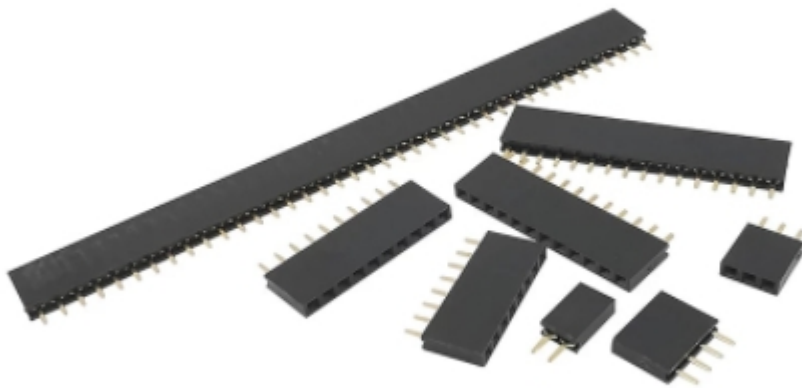
Possibilité de réaliser le circuit imprimé chez : <https://jlcpcb.com> ou autre.

Le fichier .rar dans le dossier :
PCB_Gerber_Passage_A_niveau_Stockage contient l'ensemble des fichiers
Gerber

Mettre de la pâte thermique entre le radiateur et le 7809



<https://fr.aliexpress.com/item/4000852908837.html?spm=a2g0s.9042311.0.0.41a46c372WkNYm>



Matériels utilisés:

Contact ILS

Servo Moteur SG90

Sources :

<https://www.locoduino.org/spip.php?article62>

<https://www.volta.ma/comment-controler-les-servomoteurs-avec-arduino/arduino/>

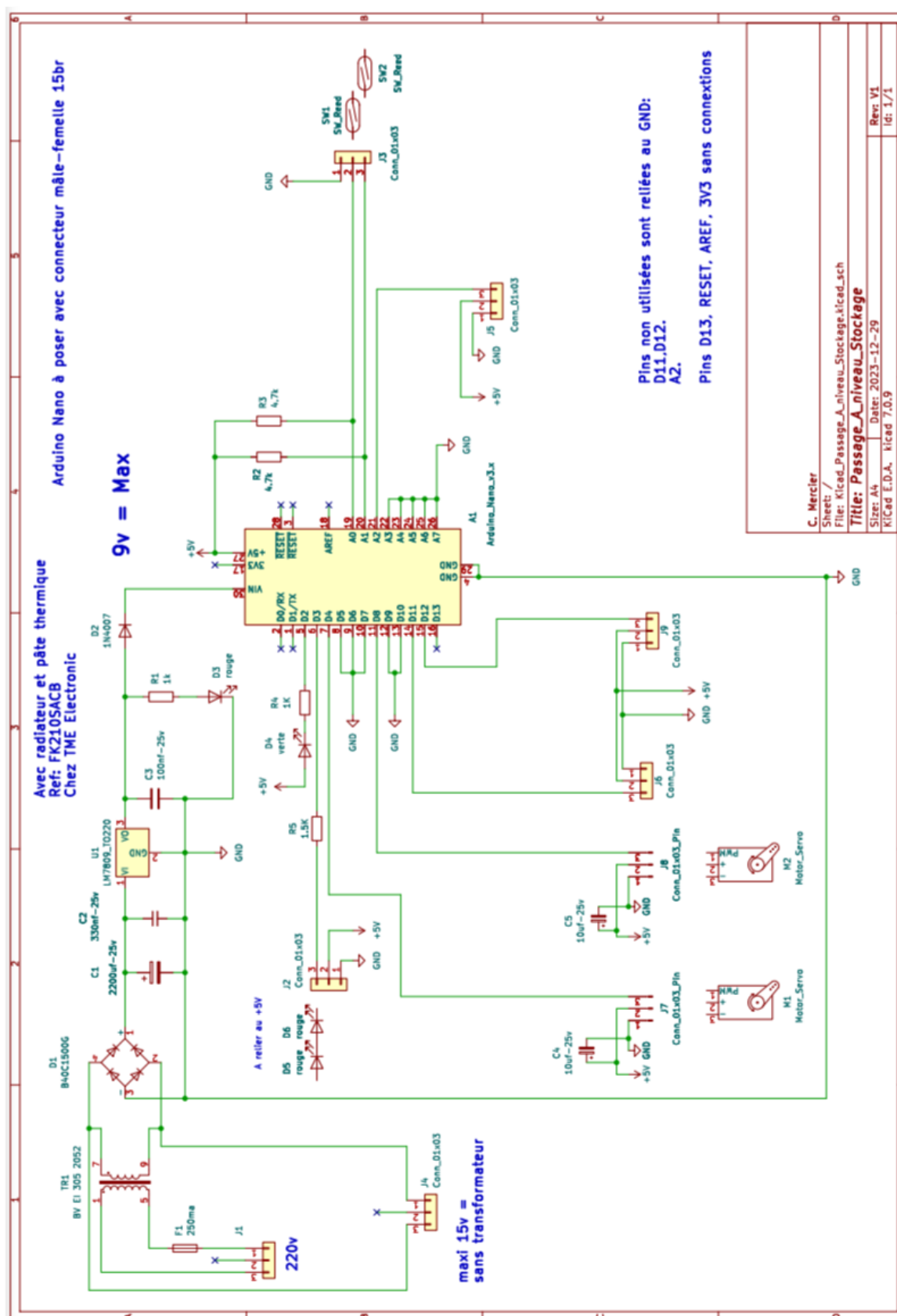
<https://www.carnetdumaker.net/articles/controler-un-servomoteur-avec-une-carte-arduino-genuino/>

<https://eskimon.fr/tuto-arduino-602-un-moteur-qui-a-de-la-tête-le-servomoteur>

Librairie:

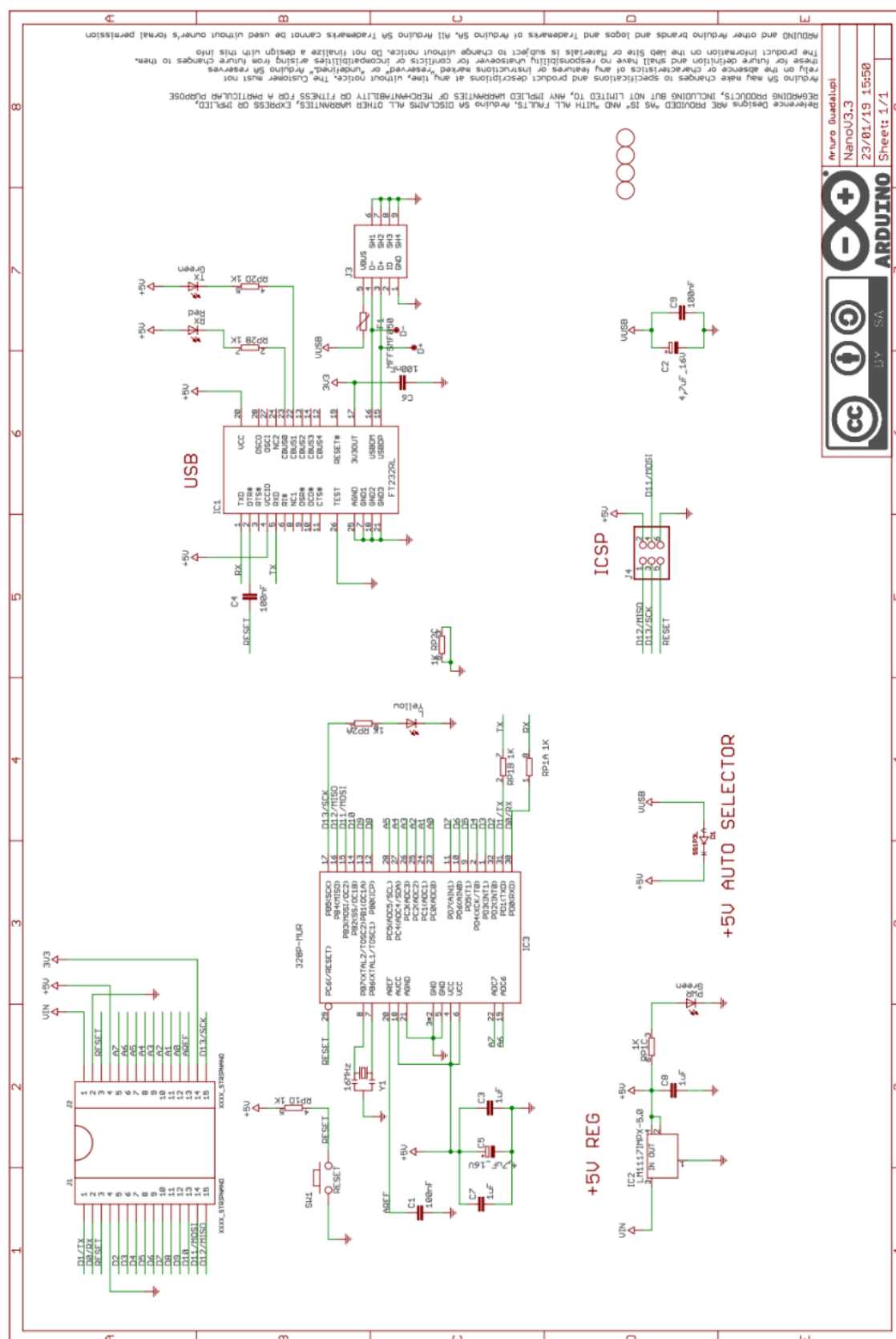
servo

Arduino – Commande passage a niveau en zone de stockage



Fichier PDF dans le dossier :
Schema_Passage_A_niveau_Stockage

Arduino – Commande passage a niveau en zone de stockage

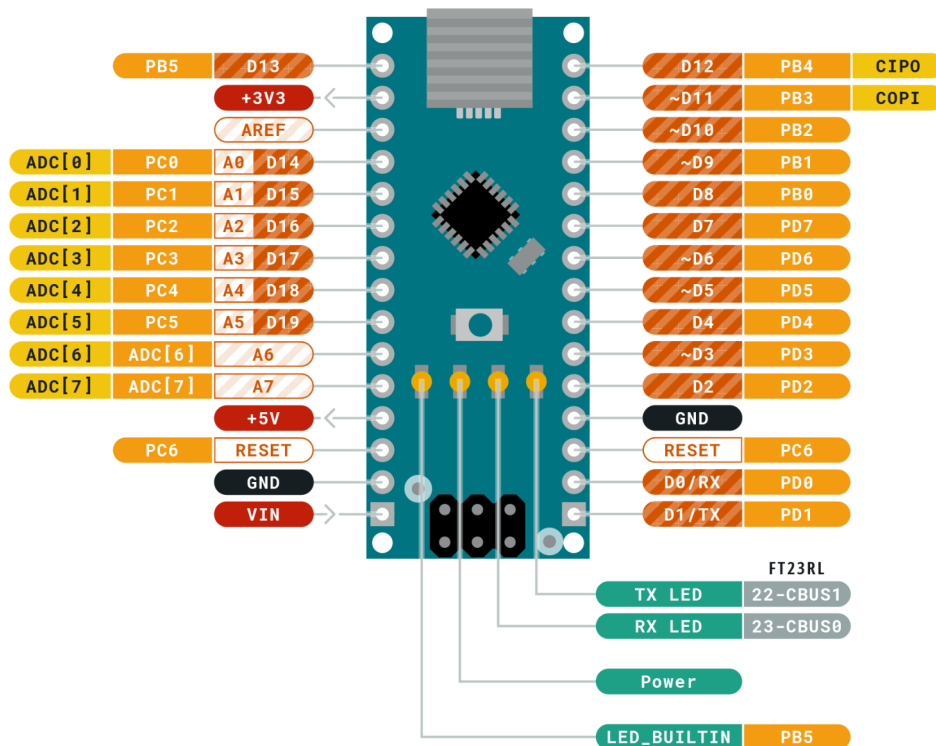


<https://store.arduino.cc/products/arduino-nano>

Arduino – Commande passage a niveau en zone de stockage



**ARDUINO
NANO**



Ground	Internal Pin	Digital Pin	Microcontroller's Port
Power	SWD Pin	Analog Pin	
LED	Other Pin	Default	



This work is licensed under the Creative Commons Attribution-ShareAlike 4.0 International License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-sa/4.0/> or send a letter to Creative Commons, PO Box 1866, Mountain View, CA 94042, USA.

<https://store.arduino.cc/products/arduino-nano>

Dossiers:

Projet _Passage à niveau en zone de stockage



Programme V1



Programme V2



Schéma passage à niveau



Schéma Nano



Liste Materiels



PCB_PDF



PCB Gerber

Chez JICPCB nécessite l'envoi du dossier:

PCB_Gerber_Passage_A_niveau_Stockage.rar