

Super Servo Tester

Testez jusqu'à quatre servos, isolés ou in situ



Figure 1. Le quadricoptère pour lequel le Super Servo Tester a été conçu.

Olivier Croiset (France)

J'ai développé le Super Servo Tester présenté ici car le drone sur lequel je travaille produit parfois une désagréable odeur d'électronique brûlée et de vernis de cuivre roussi lors des vols d'essais. J'avais besoin d'un outil pour diagnostiquer le problème et m'aider à en comprendre la cause.

On trouve sur l'Internet de nombreux schémas de circuits permettant de tester un servomoteur seul. Ils servent à vérifier que le servo que vous venez d'acheter fonctionne correctement. Le testeur présenté ici va bien plus loin dans ce concept. Vous pouvez non seulement l'utiliser pour tester jusqu'à quatre servos en même temps, mais aussi pour tester l'unité de commande de vitesse électrique (ESC) ou le calculateur de vol avec ses servos dans de multiples configurations.

Le drone pour lequel j'ai eu besoin du Super Servo Tester est représenté sur la **figure 1**. Il est basé sur [1].

Le b.a.-ba des signaux servo

Pour rappel, un servo est commandé par un signal MLI dont la fréquence (ou taux de répétition des impulsions, PRR) est de 50 Hz. La largeur de l'impulsion (la durée

où le signal est haut) est censée être comprise entre une et deux millisecondes, le reste du temps le signal est bas. Le servo est dans sa position centrale lorsque la largeur de l'impulsion est de 1,5 ms. Lorsque l'impulsion est plus courte, le servo tourne dans un sens (par ex. à gauche) ; lorsqu'elle est plus longue, il tourne dans l'autre sens (par ex. à droite).

Dans un système de vol typique, on doit vérifier cinq paramètres : la durée des impulsions de commande des quatre servos (moteur, roulis, tangage et lacet) et la tension d'alimentation. Pour une conception complexe, par exemple le développement d'un calculateur de vol (d'un drone ou autre modèle RC) ou tout mélangeur d'impulsions, ce testeur permet de voir le résultat de ce mélangeur.

Notez que dans la suite de cet article, le terme « servo » peut être remplacé par ESC ou tout autre dispositif qui produit et/ou reconnaît les signaux servo.

Caractéristiques et capacités

Le Super Servo Tester (SST) mesure la durée des impulsions de commande de jusqu'à quatre servos et fournit des informations sur la qualité de l'alimentation électrique. Il peut être inséré entre le récepteur de télécommande et les servos, entre le récepteur et le calculateur de vol du drone, ou entre le calculateur de vol et les servos.

Les **figures 2 à 6** montrent les configurations possibles.

Modes de fonctionnement

La SST dispose de deux modes de fonctionnement, sélectionnables par un commutateur (à glissière) :

- **Manuel** : Dans ce mode, le SST génère les impulsions pour quatre servos ou pour le calculateur de vol. La largeur des impulsions est commandée par quatre potentiomètres. Dans ce mode, c'est le SST qui alimente les servos (ou l'ESC) qu'on ne doit pas connecter à l'alimentation du modèle. La tension d'alimentation du SST doit être comprise entre 7,5 et 12 V_{CC}.
- **Entrées** : Dans ce mode, on mesure la longueur des impulsions provenant du récepteur ou du calculateur. Les signaux sont également transmis aux sorties du SST pour commander les servos (ou le calculateur de vol). L'alimentation électrique du modèle alimente le SST et les servos (ou les ESC). Celle-ci ne doit pas dépasser 7,49 V et on ne doit pas connecter le SST à sa propre alimentation. De même, les quatre canaux d'entrée doivent être connectés, sinon la LED et le buzzer du SST signaleront un défaut.

Modes d'affichage

L'écran affiche la durée des impulsions sous forme de quatre graphiques à barres ainsi que leur valeur numérique en microsecondes (de 1000 µs à 2000 µs). Les graphiques à barres sont limités à la plage de 1000 µs à 2000 µs, mais la valeur numérique ne l'est pas. Lorsqu'une valeur est hors limites, une boîte est dessinée autour de sa valeur numérique. Dans ce cas, la LED s'allume également et le buzzer, s'il est activé, émet un bip.

L'écran affiche également la valeur de la tension d'alimentation des servos dans l'un ou l'autre des modes de fonctionnement (**Manuel** ou **Entrées**). En effet, la qualité de l'alimentation du modèle est importante pour le calculateur de vol et pour la sécurité du (ou de la) pilote et des personnes admirant ses talents de pilotage. La tension d'alimentation mesurée est encadrée lorsque sa valeur est inférieure à 4,5 V. La LED s'allume également et le buzzer émet un signal sonore, ce qui vous évite de devoir surveiller le testeur en permanence.

La SST propose deux modes d'affichage, empilé et en carré (voir **figure 7**). Le second type est mieux adapté à un quadricoptère. La position à la mise sous tension du potentiomètre P4 détermine le mode d'affichage. Lorsque P4 est tourné à fond vers la gauche avant la mise sous tension, le SST utilise l'affichage empilé. En tournant le potentiomètre P4 vers la droite avant la mise sous tension, c'est l'affichage en carré qui est sélectionné lorsqu'on allume l'appareil.

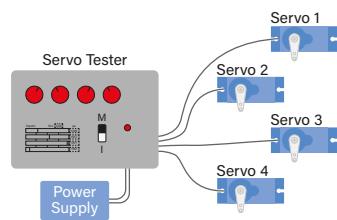


Figure 2. La configuration 1 (mode Manuel), permet de tester simplement quatre servos en même temps. Les servos sont alimentés par le testeur et commandés par les potentiomètres.

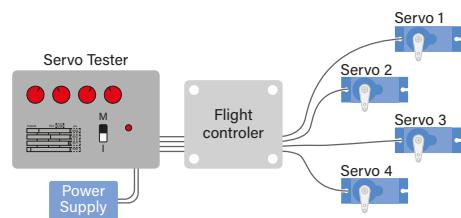


Figure 3. La configuration 2 (mode Manuel) permet de tester un calculateur de vol (de drone) sans émetteur ni récepteur. Les servos et le calculateur de vol sont alimentés par le testeur et commandés par les potentiomètres.

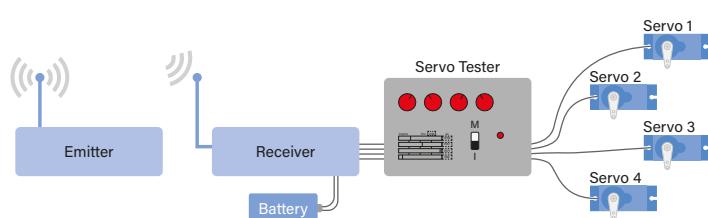


Figure 4. La configuration 3 (Mode Entrées) permet de vérifier l'émetteur et le récepteur. Le testeur, les servos et le calculateur de vol sont alimentés par la batterie du récepteur.

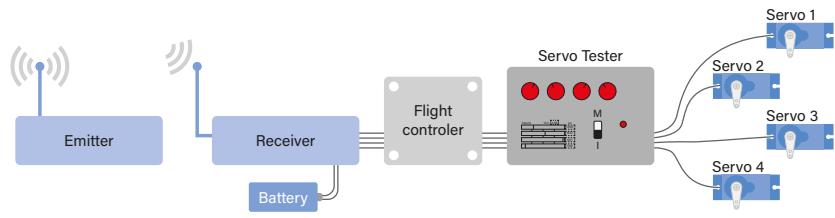


Figure 5. La configuration 4 (Mode Entrées) permet de vérifier le bon fonctionnement de l'émetteur et du récepteur avec le calculateur de vol (dans le cas d'un drone). Le testeur, les servos et le calculateur de vol sont alimentés par la batterie du récepteur.

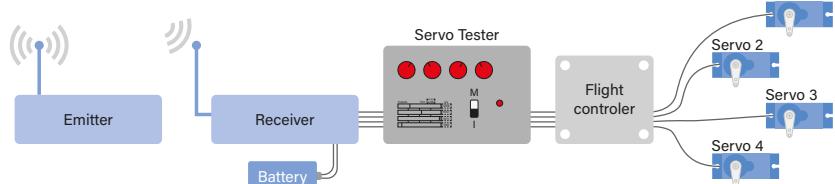
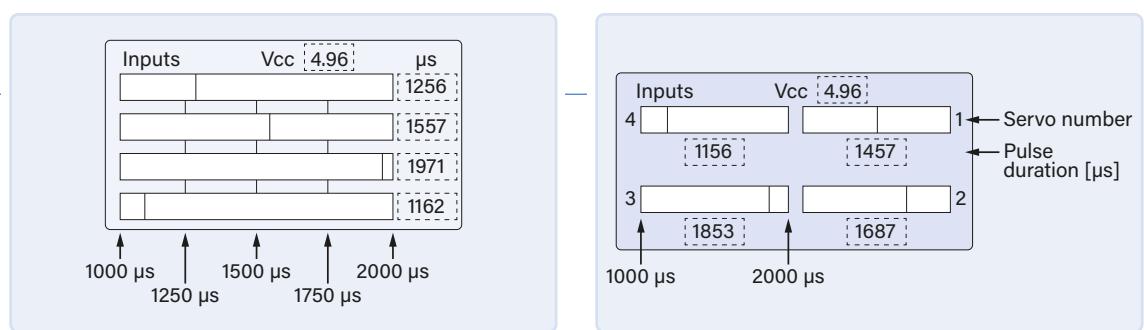


Figure 6. La configuration 5 (Mode Entrées) permet de tester le calculateur de vol avec l'émetteur et le récepteur. Le testeur, les servos et le contrôleur de vol sont alimentés par la batterie du récepteur.

Figure 7. L'écran affiche le mode de fonctionnement (Entrées ou Manuel) sélectionné par le commutateur S1, la tension V_{BATT} (comme V_{CC}) et un graphique à barres avec valeur numérique pour chacun des quatre servos. Trois index permettent d'estimer rapidement la valeur des impulsions reçues.



Circuit

Maintenant que nous savons comment utiliser le testeur, jetons un coup d'œil au circuit. Celui-ci est illustré à la **figure 8**. Il n'est pas trop compliqué car il est principalement constitué de connecteurs.

Le cœur du circuit est un microcontrôleur ATmega328P,

le même que l'on trouve sur une carte Arduino UNO. Sa fréquence d'horloge est déterminée par X1, un cristal de 16 MHz, aidé par C5 et C6. Quatre de ses ports GPIO (PB0 à PB3) sont connectés aux entrées des signaux des servos sur K1. Les sorties de signaux servo (PD5, PD6, PD7 et PB4) sont connectées à K2.

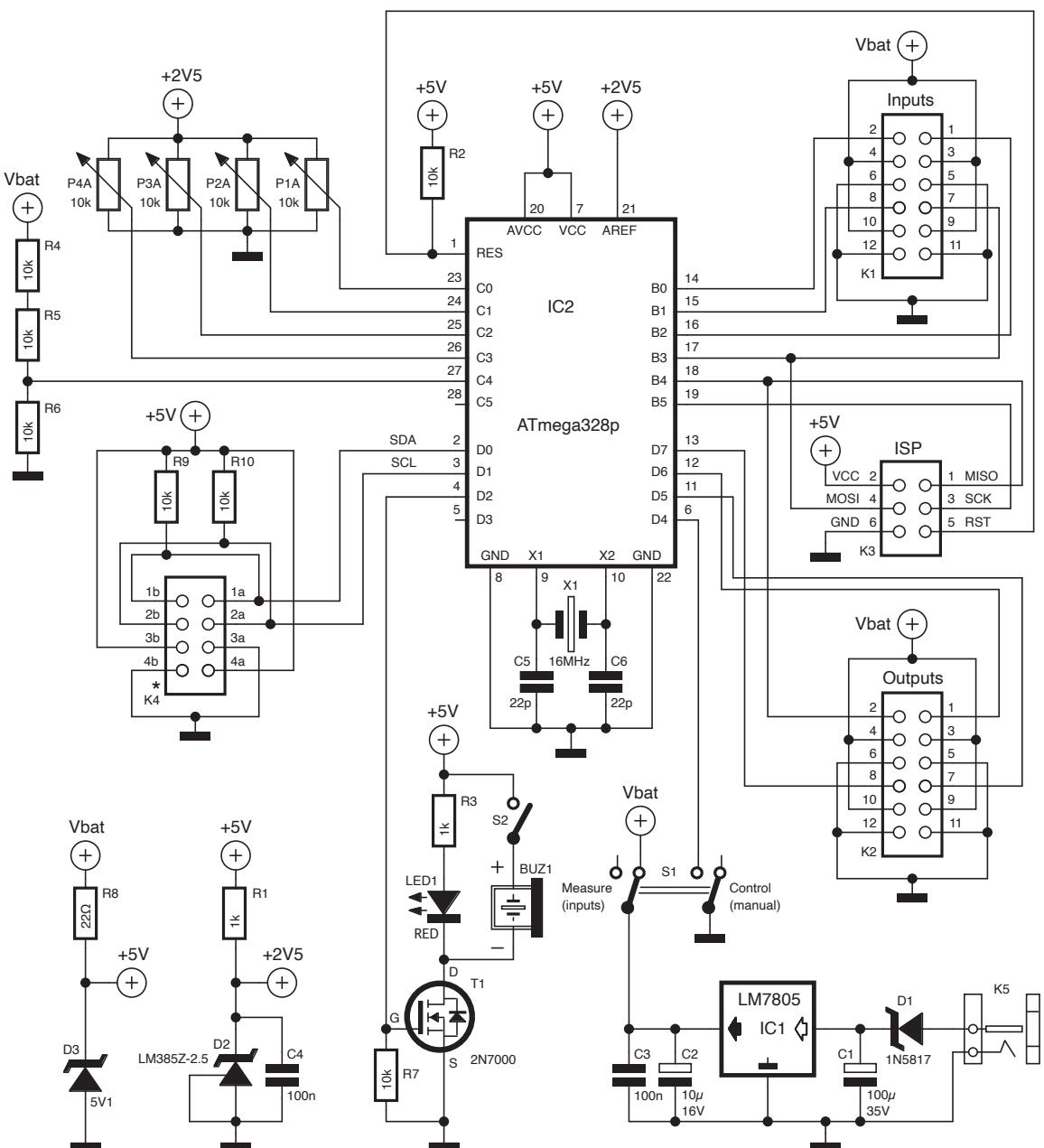


Figure 8. Le Super Servo Tester avec son cerveau ATmega328P entouré de nombreux connecteurs.



Ces deux connecteurs sont câblés pour pouvoir utiliser directement des câbles servo standard. En d'autres termes, les triplets de broches 1-3-5, 2-4-6, 7-9-11 et 8-10-12 correspondent tous à une connexion servo. Ces connexions utilisent souvent un code couleur orange-rouge-marron. L'orange (signal d'impulsion) se connecte à la broche 1 (ou 2 ou 7 ou 8), le rouge (V_{CC}) à la broche 3 (ou 4 ou 9 ou 10) et le marron (GND) à la broche 5 (ou 6 ou 11 ou 12). Bien sûr, il y aura des exceptions à cette règle, alors vérifiez avant de connecter quoi que ce soit.

Entrées analogiques

Les quatre potentiomètres sont connectés aux entrées analogiques de l'UC sur PC0 à PC3. La tension d'alimentation est connectée par un diviseur de tension (R4, R5 et R6) à l'entrée analogique PC4. Le rapport entre (R4+R5) et R6 doit être de 2:1, mais leurs valeurs absolues ne sont pas critiques. L'utilisation de trois résistances de même valeur simplifie leur tri pour plus de précision.

Pour mesurer la tension d'alimentation du microcontrôleur, une tension de référence autre que la tension d'alimentation elle-même est nécessaire pour le convertisseur analogique-numérique (CA/N). L'ATmega328P possède une référence intégrée de 1,1 V, mais cette valeur est un peu basse. J'ai donc utilisé D2, un LM385-2,5, comme référence externe de 2,5 V. Ce composant est plus précis qu'une simple diode Zener à 2 bornes, ce qui améliore la qualité de la mesure de la tension d'alimentation du servo : 1 à 2 % au lieu d'environ 5 % pour une diode Zener ordinaire.

Comme le boîtier de D2 est haut, on peut la replier sur le CI pour limiter la hauteur de l'ensemble. Il est aussi possible de l'utiliser comme plot de soutien de l'écran. La mesure de la tension d'alimentation ne doit pas être supérieure à 7,49 V, car la tension d'entrée maximale de l'UC est de 5 V. Par conséquent, l'alimentation du récepteur et des servos ne doit jamais dépasser cette valeur.

Problèmes d'affichage

L'écran OLED I²C basé sur le SSD1306 est branché sur le connecteur K4. Le port I²C de l'écran n'est pas connecté au port I²C de l'UC, mais à PD0 et PD1. Le bus I²C est émulé par logiciel. En effet, sur l'ATmega328, le bus I²C est partagé avec l'entrée analogique PC4 qui est déjà utilisée pour mesurer la tension d'alimentation. Par conséquent, on ne peut pas ici utiliser le périphérique I²C intégré. R9 et R10 sont des résistances de rappel pour le bus I²C. En théorie, elles devraient avoir une valeur d'environ 4,7 kΩ, mais 10 kΩ fonctionne aussi et permet de gagner une ligne sur la nomenclature, car la plupart des autres résistances ont aussi une valeur de 10 kΩ.

Notez que K4 est dessiné comme un connecteur à 8 broches sur deux rangées, mais sur le CI, vous devez monter une barrette à une rangée de 4 broches dans la position « A » ou « B », pas les deux. La position dépend

de votre écran. Ces écrans ont les broches GND et VCC sur les broches 3 et 4, mais pas toujours dans le même ordre. Les K4A et K4B vous permettent d'utiliser l'un ou l'autre type. L'utilisation d'une double empreinte au lieu de cavaliers (à souder) consomme moins d'espace sur la carte et permet d'économiser deux cavaliers (à souder). L'inconvénient est, bien sûr, que la position de l'ouverture de l'écran dans le boîtier dépend de l'écran car K4A et K4B ne sont pas au même endroit. De plus, selon le fabricant, les dimensions de cet afficheur peuvent varier. Vous devez donc choisir votre écran avant d'usiner le boîtier.

Sélection du mode de fonctionnement

L'interrupteur à glissière S1, de type DPDT (double contact inverseur), sélectionne le mode de fonctionnement du Super Servo Tester. En mode *Manuel*, il connecte la tension d'alimentation de 5 V aux connecteurs des servos. En mode *Entrées*, la tension d'alimentation du SST est V_{BATT} , donc sa propre alimentation de 5 V doit être déconnectée pour éviter les conflits. En fait, vous êtes censé déconnecter l'alimentation du SST dans ce mode, mais comme prudence est mère de sûreté, S1 la déconnecte pour vous. La diode Zener D3 ainsi que R8 garantissent que l'alimentation du reste du circuit dérivée de V_{BATT} ne dépasse pas les valeurs maximales supportées par les autres composants. La position de S1 est lue par PD4, de sorte que l'UC peut choisir le mode de fonctionnement correspondant.

À proprement parler, S1 n'a pas besoin d'être du type DPDT, mais il doit être capable de passer la puissance consommée par jusqu'à quatre servos et même plus s'il y a également un ESC ou un calculateur de vol connecté aux sorties. Les interrupteurs à glissière DPDT appropriés sont souvent moins chers que les types unipolaires, ce qui explique pourquoi ils sont utilisés ici.

Divers

Le port GPIO PD2 de l'UC commande la LED d'alarme et le buzzer. Comme ils sont connectés en parallèle, le MOSFET T1 fournit un peu de puissance supplémentaire pour les alimenter tous les deux sans surcharger l'UC. S2 est un deuxième interrupteur à glissière utilisé pour couper le buzzer, car il peut parfois s'avérer gênant. Cet interrupteur est de petite taille et de faible puissance. L'alimentation de 5 VCC pour le SST en mode *Manuel* est obtenue par un régulateur de tension linéaire classique 7805 (IC1). Veillez à utiliser un type TO220 pour ce composant, car il doit alimenter jusqu'à quatre servos. Pour limiter la hauteur de l'ensemble, on peut monter C1 et C2 couchés.

Enfin, le connecteur K3, de type boîtier, permet de programmer le microcontrôleur sans le retirer du circuit. Il est câblé de la même manière que le connecteur Arduino ICSP (*In-Circuit Serial Programming*).

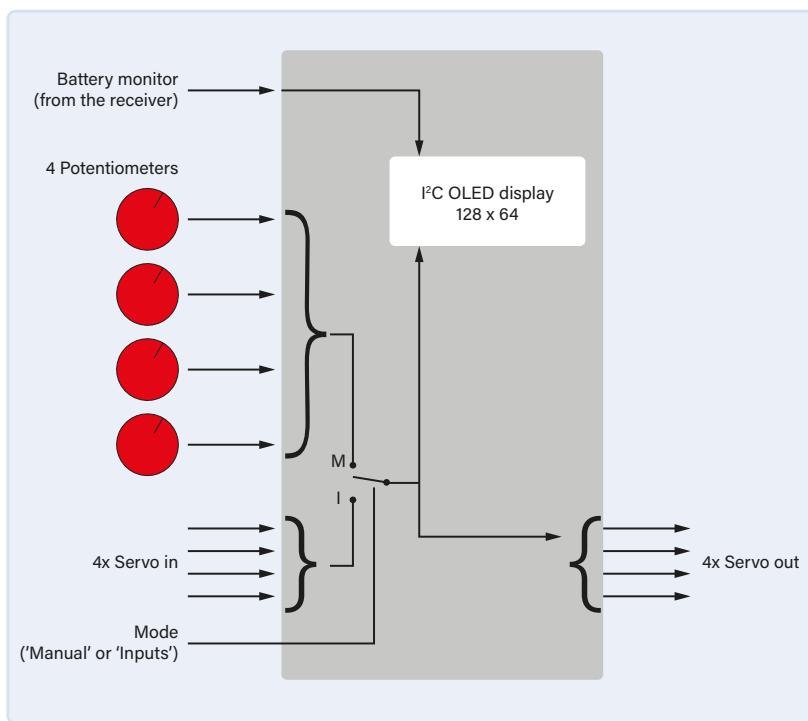


Figure 9. Vue d'ensemble de haut niveau montrant ce qui se passe à l'intérieur du microcontrôleur.

Quelques mots sur le logiciel

J'ai utilisé l'EDI Arduino pour ce projet, ce qui prouve qu'on peut l'utiliser pour des projets complexes. J'ai d'abord réalisé un démonstrateur de faisabilité avec un Arduino UNO et une platine d'expérimentation. Lorsque tout a fonctionné comme prévu, j'ai chargé le programme dans le dispositif final via le connecteur ICSP. Notez que, lorsque vous programmez le microcontrôleur via le connecteur ICSP K3, rien ne doit être relié aux connecteurs K1 et K2 car ils partagent chacun un signal avec K3. Toutes les bibliothèques nécessaires au projet sont incluses dans le téléchargement [3], à l'exception de la bibliothèque Servo, car elle fait partie de l'EDI Arduino. Le projet est organisé en trois fichiers : *Display.ino*, *Entrées.ino* et le fichier principal du croquis. Les noms des fichiers expliquent assez bien ce qu'ils contiennent. Le fichier croquis principal contient les fonctions `setup()` et `loop()` mais aussi les routines d'interruption utilisées pour minoter les impulsions d'entrée. Les changements de niveau des signaux d'entrée produisent des interruptions par changement d'état de broche. On peut alors utiliser la fonction `micros()` pour mesurer la longueur des impulsions. Les largeurs d'impulsion mesurées sont reportées sur la sortie et transmises aux fonctions d'affichage. La **figure 9** présente un aperçu de haut niveau des flux de signaux à l'intérieur du microcontrôleur.

Utilisation de la bibliothèque OLED_I2C

Comme mentionné précédemment, l'écran OLED est commandé par un bus I²C. Sur l'ATmega328P, ce bus est partagé avec le CA/N. PC4 peut être soit la broche SDA soit l'entrée analogique A4. Pour contourner ce problème, le bus I²C est émulé par logiciel. La bibliothèque *OLED_I2C* de Rinky-Dink Electronics [2] le fait et nous permet d'assigner n'importe quel port GPIO au bus I²C.

J'ai dû modifier légèrement le fichier *OLED_I2C.cpp* de cette bibliothèque car une directive `include` incorrecte empêchait sa compilation. Remplacer la ligne 25 :

```
#include "hardware/avr/HW_AVR.h"
```

par

```
#include "HW_AVR.h"
```

Aussi, dans le fichier *HW_AVR.h*, dans la fonction `OLED::update()`, j'ai commenté la ligne 24, comme ci-dessous :

```
//noInterrupts();
```

Ceci est nécessaire car la bibliothèque Servo utilise les interruptions pour générer les signaux MLI à 50 Hz sur les sorties du SST. Elles doivent également rester activées pour mesurer les impulsions entrantes.

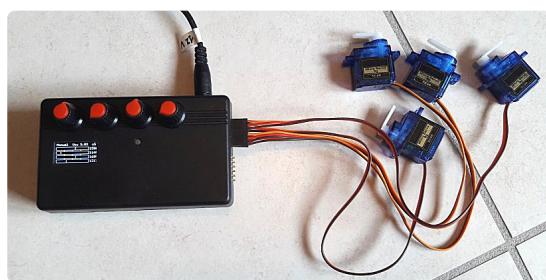


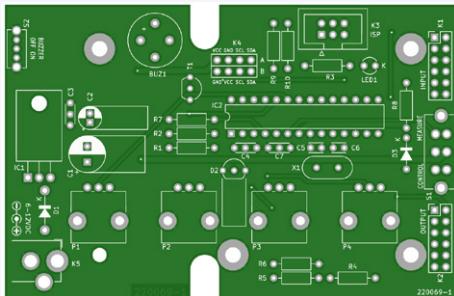
Figure 10. Le prototype de l'auteur commandant quatre servos en mode Manuel.



Figure 11. Le prototype d'Elektor Labs monté dans son boîtier.



LISTE DES COMPOSANTS



Résistances (traversantes, 0,25 W, 5%)

R1, R3 = 1 kΩ

R2, R4, R5, R6, R7, R9, R10 = 10 kΩ

R8 = 22 Ω

P1, P2, P3, P4 = 10 kΩ, linéaire, vertical

Condensateurs (traversants)

C1 = 100 µF, 35 V, diamètre 8 mm, pas 3,5 mm

C2 = 10 µF, 16 V, diamètre 5 mm, pas 2 mm

C3, C4, C7 = 100 nF, pas 2,5 mm ou 5 mm

C5, C6 = 22 pF, pas 2,5 mm

Semiconducteurs (traversants)

D1 = 1N5817

D2 = LM385Z-2.5, TO-92

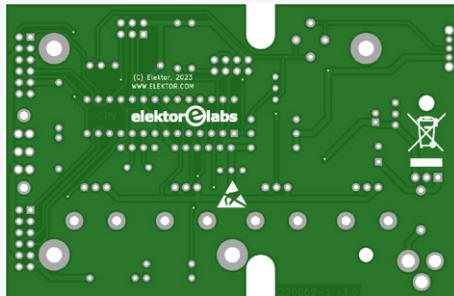
D3 = BZX79-C5V1

IC1 = 7805, TO220

IC2 = ATmega328-P

LED1 = LED, 3 mm, rouge

T1 = 2N7000, TO-92



Divers (traversants)

BUZ1 = Buzzer piézoélectrique avec oscillateur, diam. 12 mm, pas 6,5 ou 7,6 mm

K1, K2 = Connecteur à 12 broches coudées à 90° sur 2 rangées, pas 2,54 mm

K5 = Connecteur baril

K4* = Connecteur à 4 broches, pas 2,54 mm (montage A ou B selon l'écran)

K3 = Connecteur à 6 broches sur 2 rangées, pas 2,54 mm

S1 = Interrupteur à glissière DPDT, 90° (par ex. MFS201N-16-Z)

S2 = Interrupteur à glissière SPDT, 90° (par ex. OS102011MA1)

X1 = Quartz, 16 MHz, HC-49S (profil bas)

Écran OLED, 2,4 cm, 128 x 32 pixels, 4 broches I2 C (par exemple www.elektor.fr/18747)

Elektor PCB 220069-1

Boîtier suggéré : Hammond 1593N

* = voir le texte

Avant la mise sous tension

Si le potentiomètre P1 est à sa position minimale (tourné vers la gauche) à la mise sous tension, l'écran affiche un cadre blanc. Cela permet de positionner la découpe pour l'écran et d'ajuster l'ouverture si nécessaire. Tout le monde ne dispose pas d'une machine CNC à la maison. Tournez P1 vers le centre pour revenir à un fonctionnement normal. De même, si P4 est dans sa position minimale à la mise sous tension, l'affichage utilisera le mode empilé. S'il est dans sa position maximale, c'est le mode en carré qui sera activé. Dans ce deuxième cas, cela signifie que le moteur connecté à la sortie 4 sera à pleine vitesse au démarrage.

C'est le moment de décoller

Concevoir et développer le Super Servo Tester a été amusant et c'est un outil utile à avoir sous la main si vous aimez les drones, les quadrioptères ou les modèles RC en général. Non seulement ce projet m'a permis d'améliorer

mes compétences en programmation, mais je peux enfin sortir mon drone, qui attend impatiemment de décoller !

VF : Denis Lafourcade — 220069-04

Des questions, des commentaires ?

Contactez Elektor (redaction@elektor.fr).



Produits

➤ **Arduino Uno R3 (SKU 15877)**
www.elektor.fr/15877

➤ **Écran OLED, 0,96 pouce, 128x32 pixels (I²C) (SKU 18747)**
www.elektor.fr/18747

LIENS

[1] Site web du drone de Joop Brokking : http://www.brokking.net/ymfc-al_main.html

[2] Bibliothèque OLED_I2C : <http://www.rinkydinkelectronics.com/index.php>

[3] Téléchargements du projet sur Elektor Labs : <https://www.elektormagazine.fr/labs/super-servo-tester>

