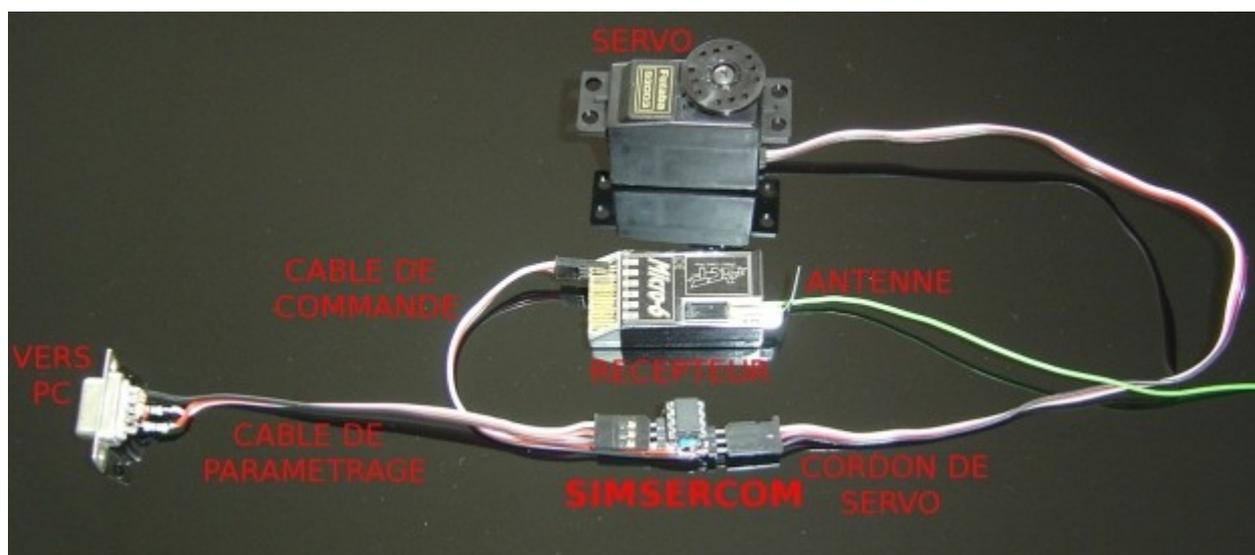


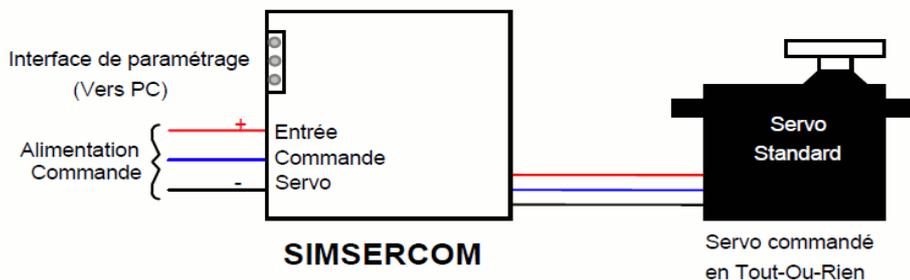
# MANUEL D'UTILISATION DE SIMSERCOM



## Table des matières

|   |    |
|---|----|
| <u>I. PRESENTATION</u> .....  | 2  |
| <u>II. SCHEMA DE SIMSERCOM</u> .....  | 2  |
| <u>III. NOMENCLATURE</u> .....  | 3  |
| <u>IV. CIRCUIT IMPRIME</u> .....  | 3  |
| <u>IV.1. CONNECTIQUE</u> .....  | 3  |
| <u>V. LES 3 TYPES DE COMMANDE POSSIBLES</u> .....   | 4  |
| <u>V.1. COMMANDE PAR UN SIMPLE INTERRUPTEUR</u> .....   | 4  |
| <u>V.2. COMMANDE PAR SORTIE TRANSISTORISEE</u> .....  | 5  |
| <u>V.3. COMMANDE PAR UNE VOIE D'UN RECEPTEUR RC</u> .....   | 5  |
| <u>VI. CORDON DE PARAMETRAGE</u> .....  | 6  |
| <u>VII. COMMANDES ACCEPTEES EN MODE TERMINAL</u> .....  | 6  |
| <u>VIII. REALISATION</u> .....  | 7  |
| <u>IX. PARAMETRAGE A L'AIDE DE SIMSERCOM.EXE</u> .....  | 8  |
| <u>X. MENTIONS LEGALES</u> .....  | 10 |
| <u>XI. INFORMATIONS UTILES A LA PROGRAMMATION DU PIC</u> .....  | 11 |
| <u>XII. QUELQUES EXEMPLES D'UTILISATION</u> .....   | 11 |
| <u>XII.1. Commandé par un interrupteur ou une sortie d'un récepteur ou d'un décodeur Multi-Switch</u> ..... | 11 |
| <u>XII.2. Commandé par un oscillateur basse fréquence</u> .....   | 11 |
| <u>XII.3. Commandé par un photo-transistor</u> .....  | 12 |
| <u>XIII. REVISIONS DU DOCUMENT</u> .....  | 12 |

## I. PRESENTATION



Le montage [SIMSERCOM](#) basé sur un micro-contrôleur PIC 12F675 ou PIC 12F6269 de chez Microchip permet de fixer les limites des positions extrêmes d'un servo standard ainsi que la vitesse de déplacement du palonnier entre ces 2 positions.

La commande est : soit une sortie Tout-Ou-Rien (simple interrupteur, contact de relais ou sortie Multi-Switch), soit une sortie voie d'un récepteur RC.

[SIMSERCOM](#) est une version ultra-simplifiée (5 composants : IC1, C1, R1, D1 et D2) du montage [UNISERCOM](#) tout en conservant 2 des 3 modes de fonctionnement de ce dernier.

Le principal défaut de [UNISERCOM](#) est l'encombrement lié à l'utilisation des 3 résistances ajustables utilisées pour les réglages des butées et de la vitesse.

Les résistances ajustables sont encombrantes, hé bien, supprimons-les !

Oui, mais comment [SIMSERCOM](#) permet-il de faire les réglages de butées et de vitesse ?

[SIMSERCOM](#) utilise une interface série RS232 ultra-simplifiée qui avait été développée et validée avec succès pour le montage [P&B](#).

[SIMSERCOM](#) se paramètre donc à l'aide d'un PC par une liaison série RS232 en mode terminal. Cela signifie qu'il n'y a pas obligatoire de disposer d'un programme spécifique à [SIMSERCOM](#) pour être paramétré.

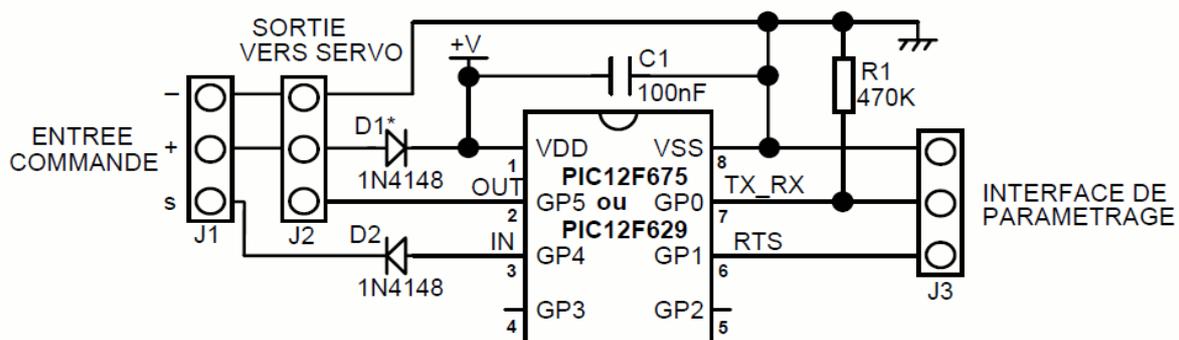
Sous Window\$, il suffit d'utiliser l'application "simsercom.exe" disponible sur le site de l'auteur : <http://p.loussouarn.free.fr>. Il s'agit d'un programme graphique convivial qui permet de paramétrer très facilement [SIMSERCOM](#) sans avoir besoin de connaître la messagerie de commande.

Si vous ne voulez pas installer "simsercom.exe" sur votre PC, il vous suffit d'utiliser le terminal "HyperTerminal" déjà inclus à Window\$ XP ou "TeraTerm" pour Window\$ Vista et 7.

Sous Linux, le terminal "Minicom" fera parfaitement l'affaire.

Aujourd'hui, presque tout le monde dispose d'un PC, c'est pourquoi, cette solution est d'actualité.

## II. SCHEMA DE SIMSERCOM



D1\*: Nécessaire pour une alimentation en 6V.

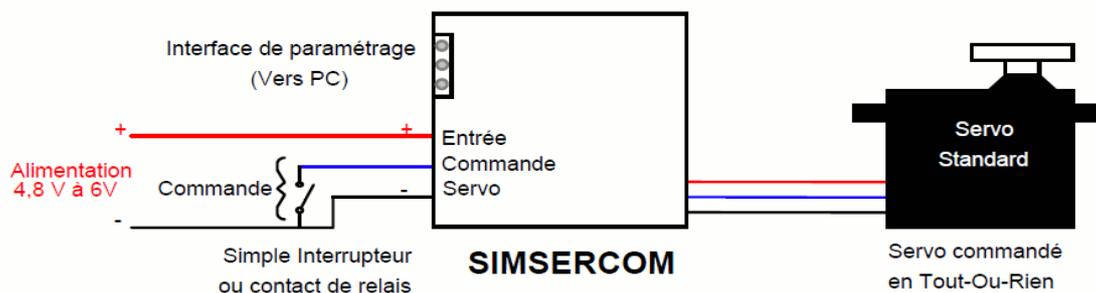
Si l'alimentation est inférieure ou égale à 5,5V, D1 est remplacée par un strap.



## V. LES 3 TYPES DE COMMANDE POSSIBLES

| Utilisation | Signal de commande   | Positions A  | Positions B   |
|-------------|--|--|---|
| 1           | Simple interrupteur ou contacts de relais  | Contact fermé  | Contact ouvert  |
| 2           | Sortie transistorisée de type Tout-Ou-Rien :<br>- collecteur ouvert<br>- push-pull<br>- totem-pole<br>- inverseur CMOS<br>- Multi-switch | $0 < \text{Tension} < 1,5 \text{ V}$                   | $2,5\text{V} < \text{Tension} < 24\text{V}$           |
| 3           | Signal d'une voie proportionnelle issue d'un récepteur RC  | Impulsion $< 1,5 \text{ ms}$<br>(au-dessous du neutre) | Impulsion $> 1,5 \text{ ms}$<br>(au-dessus du neutre) |

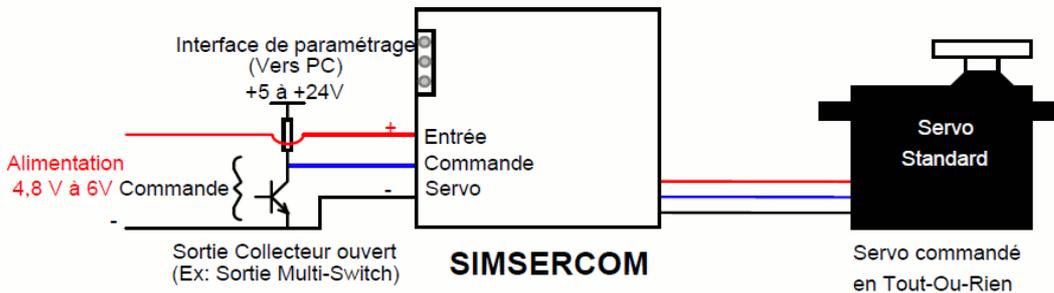
### V.1. COMMANDE PAR UN SIMPLE INTERRUPTEUR



Dans cette configuration, il n'y a pas de radio-commande : le servo est commandé par un simple interrupteur ou un contact de relais : le montage est autonome et alimenté par une batterie externe (4,8 à 6 V).

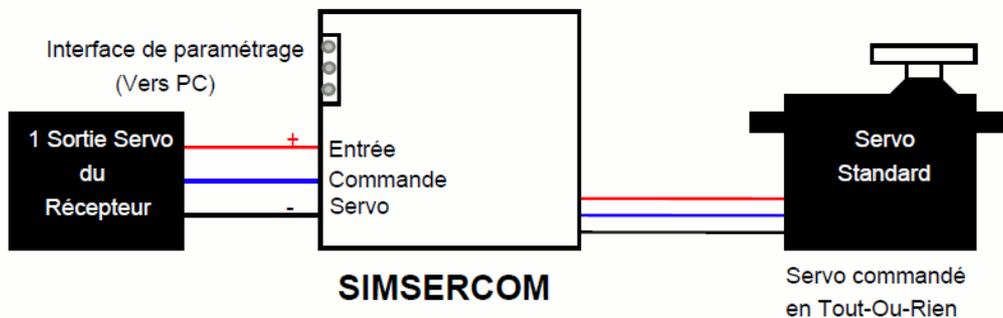
- L'interrupteur ou le contact de relais est fermé :  
→ le servo va lentement en position extrême A.
- L'interrupteur ou le contact de relais est ouvert :  
→ le servo va lentement en position extrême B.

## V.2. COMMANDE PAR SORTIE TRANSISTORISEE



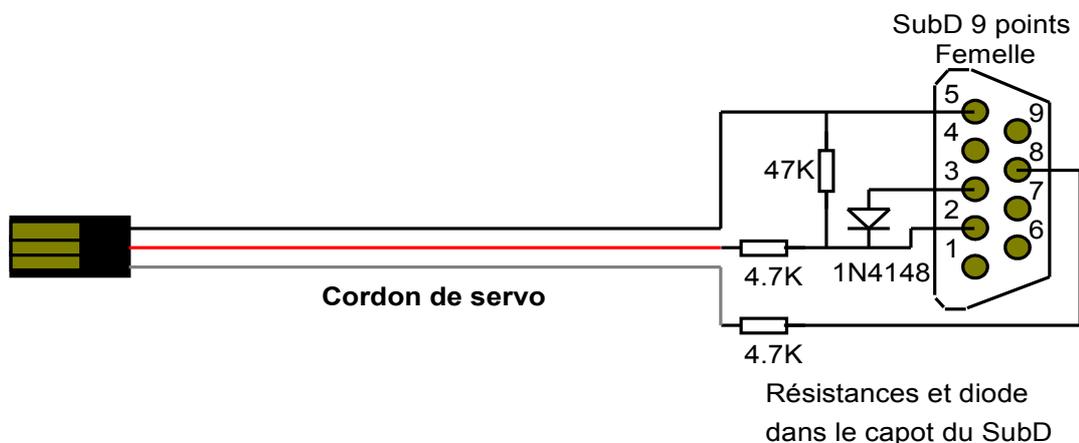
- Lorsque le transistor à collecteur ouvert est saturé, il y a un niveau logique 0 au niveau de la commande, le servo va progressivement en position extrême A.
- Lorsque le transistor à collecteur ouvert est bloqué, la résistance de tirage au + (pull up) assure un niveau logique 1 au niveau de la commande, le servo va progressivement en position extrême B. La tension de polarisation de la résistance de tirage peut sans problème atteindre 24V, puisque l'entrée du PIC est protégée par une diode : l'utilisation d'une sortie d'un Automate Programmable Industriel (API) est donc possible avec [SIMSERCOM](#).

## V.3. COMMANDE PAR UNE VOIE D'UN RECEPTEUR RC



- Si la largeur des impulsions en sortie du récepteur RC est inférieure à 1,5 ms (manche au-dessous du neutre), il s'agit d'une commande pour atteindre la position extrême A,
- Si la largeur des impulsions en sortie du récepteur RC est supérieure à 1,5 ms (manche au-dessus du neutre), il s'agit d'une commande pour atteindre la position extrême B.
- Le montage est alimenté par le récepteur, et de ce fait, ne nécessite pas d'alimentation externe.

## VI. CORDON DE PARAMETRAGE



Le cordon de paramétrage associé à un PC et à un programme ‘Terminal’, il permet d’ajuster finement :

- La position extrême A du servo,
- La position extrême B du servo,
- La vitesse du servo pour aller de la position A à la position B et de la position B à la position A.

Une fois les paramètres ajustés et sauvegardés dans la mémoire EEPROM du micro-contrôleur PC, ce cordon ne sert plus à rien, il doit être retiré du montage [SIMSERCOM](#).

Il est bien sûr possible de retoucher le paramétrage ultérieurement sans avoir à recharger le programme (fichier HEX) dans le PIC. Il suffit juste de réutiliser le cordon de paramétrage pour entrer les nouveaux paramètres de fonctionnement.

Ce cordon de paramétrage n’est pas spécifique à [SIMSERCOM](#), il est désormais utilisé pour tous les montages de l’auteur qui se paramètrent par une liaison série RS232 : c’est devenu un ‘standard’.

Par exemple, ce cordon est déjà utilisé pour paramétrer les montages [P&B](#) de l’auteur.

## VII. COMMANDES ACCEPTEES EN MODE TERMINAL

Dans la table ci-dessous:

- "A" désigne la 1<sup>ère</sup> position extrême du servo,
- "B" désigne la 2<sup>e</sup> position extrême du servo,
- "S" désigne la vitesse du servo.

| COMMANDE | ACTION  | REMARQUE                        |
|----------|---|---------------------------------|
| A=DDDD   | Pour le servo, programme la consigne de largeur d’impulsion de servo correspondant à la position extrême A. | Unité : $\mu$ s<br>Réponse : OK |
| A+       | Pour le servo, incrémente de 1 $\mu$ s la consigne de position extrême A.                                   | Réponse : OK                    |
| A-       | Pour le servo, décréméte de 1 $\mu$ s la consigne de position extrême A.                                    | Réponse : OK                    |
| A?       | Pour le servo, interroge la consigne courante de largeur  | Réponse:                        |

|        |   |                                       |
|--------|---|---------------------------------------|
|        | d'impulsion de servo correspondant à la position extrême A.   | A=DDDD en $\mu\text{s}$               |
| B=DDDD | Pour le servo, programme la consigne de largeur d'impulsion de servo correspondant à la position extrême B.   | Unité : $\mu\text{s}$<br>Réponse : OK |
| B+     | Pour le servo, incrémente de 1 $\mu\text{s}$ la consigne de position extrême B.   | Réponse : OK                          |
| B-     | Pour le servo, décréméte de 1 $\mu\text{s}$ la consigne de position extrême B.  | Réponse : OK                          |
| B?     | Pour le servo, interroge la consigne courante de largeur d'impulsion de servo correspondant à la position extrême B.  | Réponse:<br>B=DDDD en $\mu\text{s}$   |
| S=DDDD | Pour le servo, programme la consigne vitesse pour aller de la position extrême A à la position extrême B et de la position extrême B à la position extrême A. | Unité : $\mu\text{s}$<br>Réponse : OK |
| S+     | Pour le servo, incrémente de 1 $\mu\text{s}$ la consigne de vitesse   | Réponse : OK                          |
| S-     | Pour le servo, décréméte de 1 $\mu\text{s}$ la consigne de vitesse  | Réponse : OK                          |
| S?     | Pour le servo, interroge la consigne courante de vitesse  | Réponse:<br>S=DDDD en $\mu\text{s}$   |
| R      | Enregistre les paramètres courants dans la mémoire EEPROM du micro-contrôleur   | Réponse : OK                          |

#### Notes importantes:

- Les commandes agissent en dynamique, c'est-à-dire, qu'il n'est pas nécessaire d'arrêter puis de remettre sous tension [SIMSERCOM](#) pour que les nouveaux paramètres prennent effet,
- Une fois les paramètres ajustés, ne pas oublier d'envoyer la commande **R** (**R**ecord=**E**n**R**egistrer) pour les sauvegarder dans la mémoire EEPROM.
- Plus on augmente la valeur de **S** (**S**peed=**V**ite**S**se), et plus le servo est rapide : en fait il s'agit du pas d'incrément ou de décrément qui est appliqué au servo toutes les :
  - 20 ms environ (période fixée par l'émetteur si l'entrée de [SIMSERCOM](#) est une voie du récepteur),
  - 30 ms si l'entrée de [SIMSERCOM](#) est une commande TOR(Tout-Ou-Rien)

## VIII. REALISATION

1. Faire le circuit imprimé décrit au §IV,
2. Charger le programme "**simsercom.hex**" dans la mémoire Flash du PIC : c'est l'opération la plus délicate. Pour obtenir le fichier "**simsercom.hex**", contacter l'auteur sur le site: [RC NAVY](#)
3. Réaliser le cordon de paramétrage tel que décrit au §VI, et le connecter sur [SIMSERCOM](#) et sur un port série (port COM) du PC.  
Sur les PC récents, il n'y a plus de port COM RS232. La solution consiste à investir dans un convertisseur USB→RS232. On en trouve pour quelques € sur ebay, par exemple.
4. Mettre [SIMSERCOM](#) sous tension,
5. Sous Window\$, pour paramétrer [SIMSERCOM](#) (régler la valeur des positions extrêmes et la vitesse), 2 solutions sont possibles:
  - Utiliser l'interface graphique conviviale "**simsercom.exe**": aller directement au §IX,
  - Utiliser le programme HyperTerminal intégré à Window\$ XP ou TeraTerm pour Window\$

VISTA et Window\$ SEVEN:  
continuer ci-dessous.

6. Pour Window\$ XP, lancer l'HyperTerminal :  
(Démarrer→Programme→Accessoire→Communication→Hyperterminal)
  - Ajuster les paramètres de communication à 4800, n, 8, 1
  - Activer le contrôle de flux : Hardware
7. Ajuster les positions extrêmes à l'aides des commandes définies dans la table au §VII,
8. Une fois les réglages terminés, ne pas oublier d'envoyer la commande **R** pour sauvegarder les paramètres.

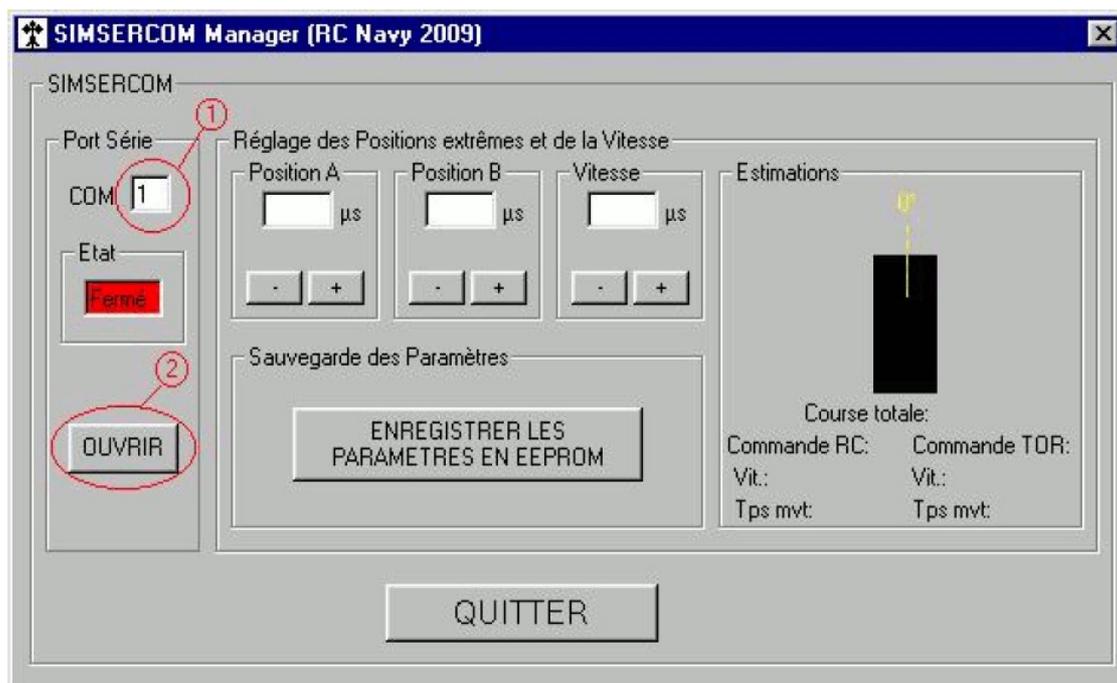
## IX. PARAMETRAGE A L'AIDE DE SIMSERCOM.EXE

Télécharger le paquetage d'installation "**simsercom.zip**" sur le site de l'auteur et installer-le sur votre PC Window\$.

Lancer l'application "**simsercom.exe**" par :

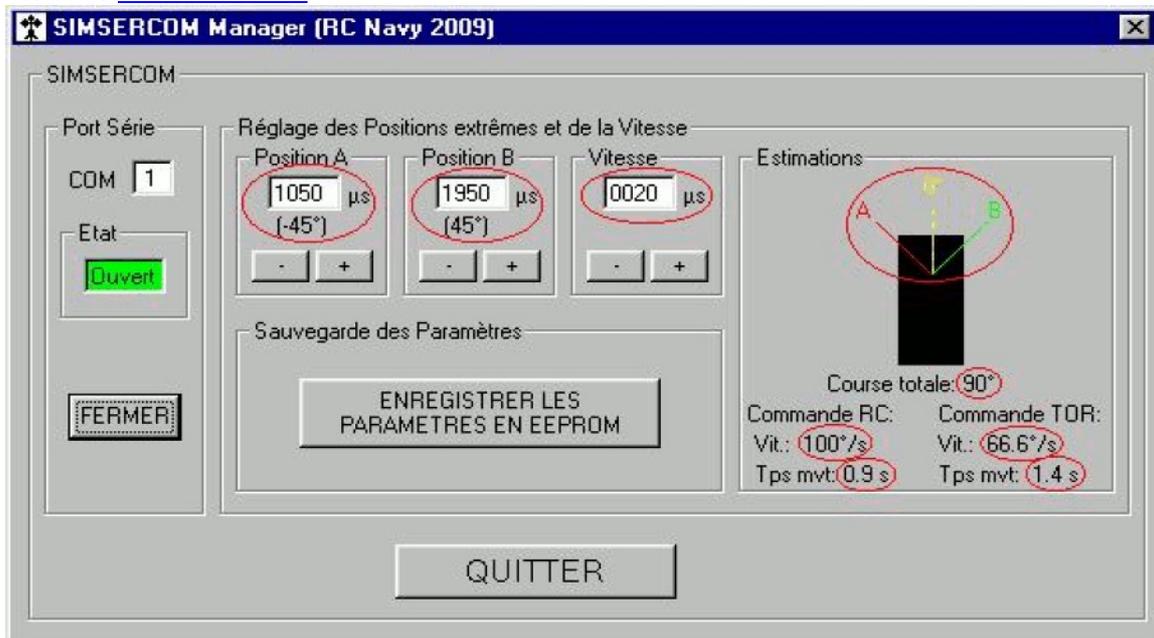
**Démarrer** → **Programmes** → **SIMSERCOM** → **SIMSERCOM**

La fenêtre suivante apparaît :



- Brancher le cordon de paramétrage entre [SIMSERCOM](#) et le port série RS232 (port COM) du PC,
- Mettre sous tension SIMSERCOM,
- Dans la fenêtre graphique, dans la zone(1), entrer le numéro du port série RS232 (port COM),
- Cliquer sur "OUVRIR" (2),

La communication s'établit et les champs entourés en rouge se remplissent avec les paramètres déjà présents dans [SIMSERCOM](#):

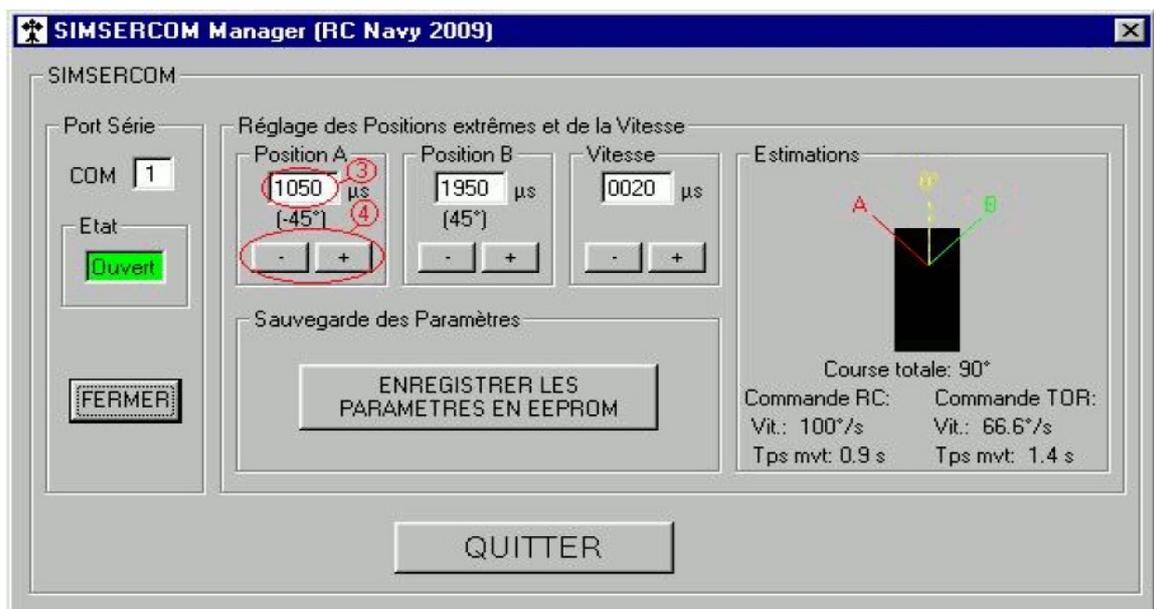


Pour régler grossièrement la valeur de position extrême A, entrer la valeur désirée dans la zone (3), puis confirmer en appuyant sur la touche "Entrée" du clavier du PC.

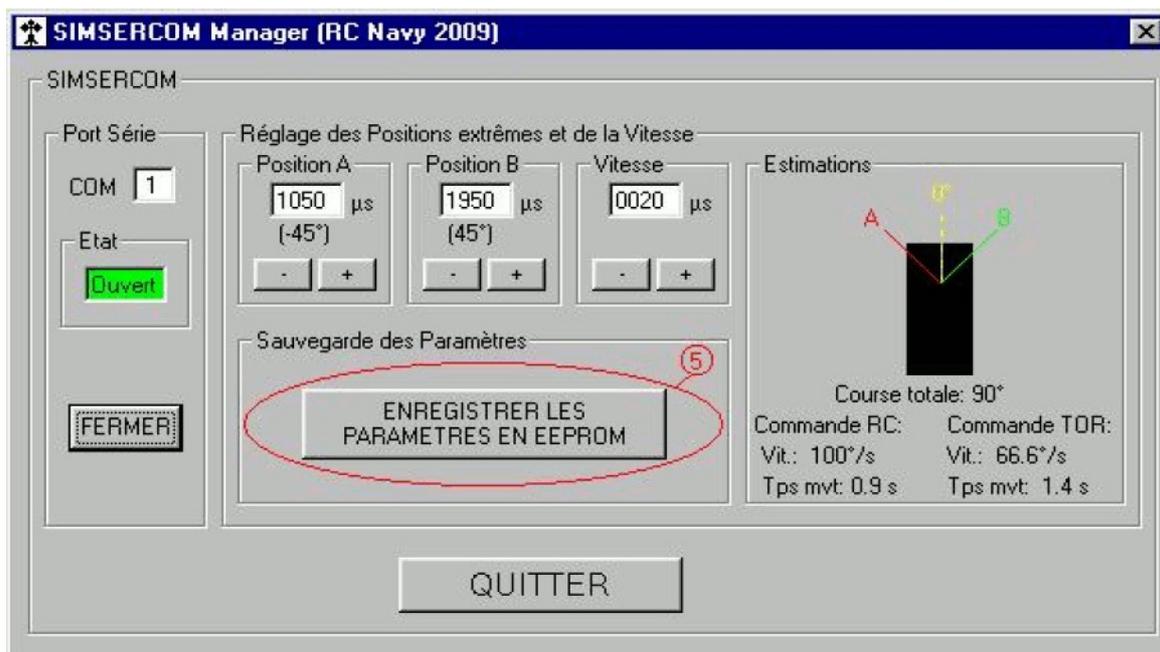
Pour affiner le réglage, utiliser les touches – et + de la zone (4).

Si l'ordre de commande courant de [SIMSERCOM](#) est un ordre pour aller en position A, le SERVO se positionne en temps réel et la position A est mise à jour dans le cadre "Estimations".

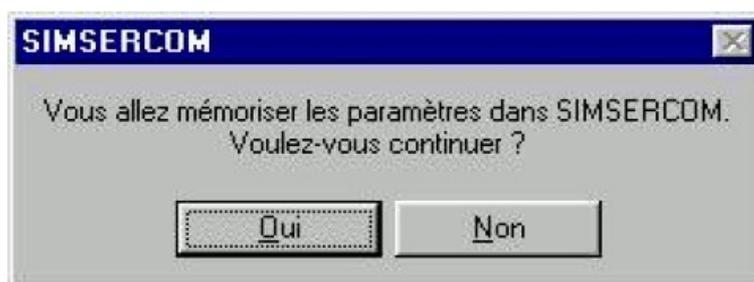
Le réglage de la position extrême B et de la Vitesse est similaire.



Une fois que les positions extrêmes A et B ainsi que la Vitesse de déplacement ont été correctement paramétrées, il ne faut pas oublier de les enregistrer dans la mémoire EEPROM de [SIMSERCOM](#). Pour cela, il suffit de cliquer sur la touche de la zone (5) :



Répondre Oui à l'invite de la fenêtre suivante :



Si tout s'est bien passé, la fenêtre de confirmation suivante apparaît :



[SIMSERCOM](#) est désormais autonome, le cordon de paramétrage peut être déconnecté.

En coupant l'alimentation de [SIMSERCOM](#) puis en le remettant sous tension, on constate que SIMERCOM a bien mémorisé les paramètres.

## X. MENTIONS LEGALES

L'auteur n'est en aucun cas responsable des dommages qui pourraient découler de la mauvaise utilisation ou d'un éventuel dysfonctionnement du montage [SIMSERCOM](#). Il appartient donc à l'utilisateur d'en mesurer et d'en assumer les risques.

## XI. INFORMATIONS UTILES A LA PROGRAMMATION DU PIC

Afin de limiter le nombre de composant du montage, c'est l'oscillateur RC interne (à 4MHz) du PIC12F675 ou du PIC12F629 qui est utilisé.

Les dispersions de fabrication imposent une calibration (un réajustement) de l'oscillateur qui est différente d'un PIC à l'autre.

Le fabricant Microchip a prévu un registre (une case mémoire du PIC) dans laquelle, au démarrage, on vient écrire une valeur destinée à réajuster l'oscillateur afin que la fréquence soit 4 MHz à 1% près.

Pour chaque PIC, Microchip donne cette valeur (valeur à charger dans le registre OSCCAL) et la stocke à la fin de la zone mémoire programme, c'est-à-dire à l'adresse 0x3FF dans le cas du PIC12F675 et du PIC12F629.

Aussi, avant de charger le fichier HEX, il est indispensable de faire une lecture du contenu de la mémoire programme du PIC neuf afin de noter la valeur présente à l'adresse 0x3FF (normalement 0x34XX).

Le programme HEX peut alors être chargé dans le buffer de l'outil de programmation (Ex : IC-PROG).

Une fois chargé, il est nécessaire de forcer la valeur présente à l'adresse 0x3FF à la valeur précédemment relevée (0x34XX).

Le composant peut maintenant être programmé (chargement du fichier HEX).

Une relecture du composant PIC permettra de vérifier qu'à l'adresse 0x3FF se trouve bien la valeur 0x34XX.

Il est indispensable que cette valeur commence par 0x34, parce qu'il s'agit d'une instruction RETLW qui retourne dans W la valeur XX destinée à être chargée dans le registre OSCCAL du PIC.

**Dans le cas contraire, le programme ne démarrera pas.**

L'opération de calibration de l'oscillateur interne est réalisée automatiquement au démarrage du programme.

## XII. QUELQUES EXEMPLES D'UTILISATION

### ***XII.1. Commandé par un interrupteur ou une sortie d'un récepteur ou d'un décodeur Multi-Switch***

- Ouverture/Fermeture lente d'une porte ou d'une trappe,
- Rotation d'une tourelle de canon,
- Montée/Descente du train d'atterrissage d'un avion/hélicoptère,
- Montée/Descente d'un pylône pour planeur,
- Etc...

### ***XII.2. Commandé par un oscillateur basse fréquence***

- En connectant la sortie d'un oscillateur astable (basé sur un 555 par exemple), à l'entrée commande du montage [SIMSERCOM](#), il est possible d'animer un essuie-glace à une vitesse réaliste. Ne pas en abuser : risque d'usure prématurée du potentiomètre de recopie du servo.

### ***XII.3. Commandé par un photo-transistor***

- En connectant un photo-transistor à l'entrée commande du montage [SIMSERCOM](#), il est possible de faire une vanne de gaz sécurisée pour les amateurs de modélisme vapeur : Il suffit d'éclairer le photo-transistor avec une lampe de poche afin de positionner le palonnier du servo pour ouvrir la vanne de gaz et d'allumer le brûleur. Dès que le brûleur est allumé, il est possible d'éteindre la lampe de poche car la flamme éclaire le photo-transistor. Si le brûleur s'éteint, le photo-transistor n'est plus éclairé et [SIMSERCOM](#) ferme automatiquement la vanne d'arrivée de gaz.

## **XIII. REVISIONS DU DOCUMENT**

- 22/03/2009 : création
- 06/09/2009 : correction table des commandes et du débit RS232 (4800 bauds),
- 13/09/2009 : ajout utilisation de l'interface graphique de paramétrage (programme PC) sous Window\$,
- 22/04/2012 : correction du câblage du cordon de paramétrage (la résistance de 47K est câblé entre le point 5 et le point 2 du SubD 9 points et non entre le point 5 et le point 3).