Création d’une chaise de bureau connectée à réglage électrique et mémoire de position



TO52

Rapport

Quentin Chambefort

Nahil Zamiati

Table des matières

[Introduction 3](#_Toc534214016)

[1. Présentation Du projet 3](#_Toc534214017)

[a. Expression du besoin 3](#_Toc534214018)

[b. Concept 3](#_Toc534214019)

[c. Ouvertures 3](#_Toc534214020)

[2. Cahier des charges 3](#_Toc534214021)

[a. Expression du besoin 3](#_Toc534214022)

[b. Difficultés prévues & contraintes du projet 4](#_Toc534214023)

[c. Choix des technologies 4](#_Toc534214024)

[3. Réalisation du projet 4](#_Toc534214025)

[a. Descriptif de l’existant 5](#_Toc534214026)

[b. Conception électronique 5](#_Toc534214027)

[c. Conception informatique 9](#_Toc534214028)

[d. Conception mécanique 10](#_Toc534214029)

# Introduction

Parler de l’idée du projet

# Présentation Du projet

## Expression du besoin

Expliquer l’idée qu’on passe quand même pas mal de temps assis sur une chaise en tant que dev

Donc autant qu’elle soit confortable

Expliquer que le manque de possibilité de réglage rend les chaises de bureau moins « polyvalentes » (matter un film dans une chaise de bureau, quelle plaie)

Faire des stats de position, connaître la position préférée des utilisateurs en fonction des applis qu’ils utilisent

## Concept

Chaise de bureau ultra-confortable

Réglage électrique et automatique

Mémoires de position pour utilisation par différents utilisateurs

Intégrée dans un univers d’objets connectés, type « smart home »

Réagit selon l’utilisation

## Ouvertures

Utilisation comme fauteuil de salon, s’adapte à l’utilisation

Utilisation dans les voitures, selon le type de route (autoroute vs nationales)

# Cahier des charges

## Expression du besoin

* Le siège doit être confortable.
* Le siège doit pouvoir se régler de manière manuelle à l’aide de boutons physiques/interface à l’écran. (Application en réseau local)
* Le siège doit pourvoir être contrôlé de manière automatique selon des paramètres utilisateurs (mémoire de position)
* Le siège doit pourvoir être multi-utilisateur, et associer une série de paramètres suivant l’utilisateur l’utilisant.
* Le siège doit pouvoir s’intégrer dans un environnement d’objets connectés (assistant vocal, application bureau, etc…)
* Le siège doit pouvoir détecter si une personne est assise dessus.
* Le siège doit proposer des API de développement.
* Le siège doit fonctionner sans fil.
* Le siège doit pouvoir se connecter à un réseau wifi.

## Difficultés prévues & contraintes du projet

Choix du matériel : carte de programmation adéquate

Prise en main des composants électroniques

Prise en main des assistants vocaux

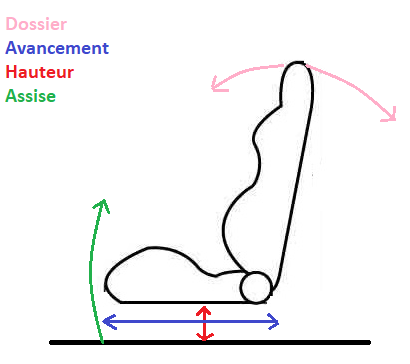
Partie conception mécanique et adaptation de pièces sur le châssis du siège.

Découverte des API pour les différents assistants vocaux sur le marché.

# Réalisation du projet

## Descriptif de l’existant

Un siège avec 4 moteurs pour faire bouger 4 mouvements différents :

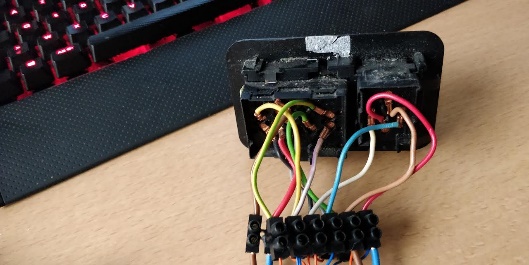


Le mouvement « avancement » consiste à faire avancer le siège sur un mouvement linéaire horizontal

Le mouvement « hauteur » consiste à faire monter le siège selon un mouvement linéaire vertical

Le mouvement « assise » consiste à lever l’extrémité avant de l’assise (sous les genoux)

Le mouvement « dossier » consiste à incliner le dossier



Les 4 moteurs sont connectés directement à des boutons situés sur la tranche du siège pour les piloter directement

[schéma électrique des boutons et des moteurs]

## Choix des technologies

D’abord Arduino MEGA pour nombre d’I/O disponibles, connaissance préalable de l’environnement ARDUINO

Puis ESP8266 E12 (wifi intégré de base sans rajout de modules)

Contraintes de l’ESP8266 : moins d’I/O, fonctionne en 3.3v

Avantages : meilleures performances, wifi intégré

Pour pallier le manque d’I/O :

Utilisation de registres à décalage en série (SN74HC595N 8bit) et de multiplexeurs (CD4051BE)

Avantages SN74HC595N : Mise en série des SR avec le même nombre d’entrées

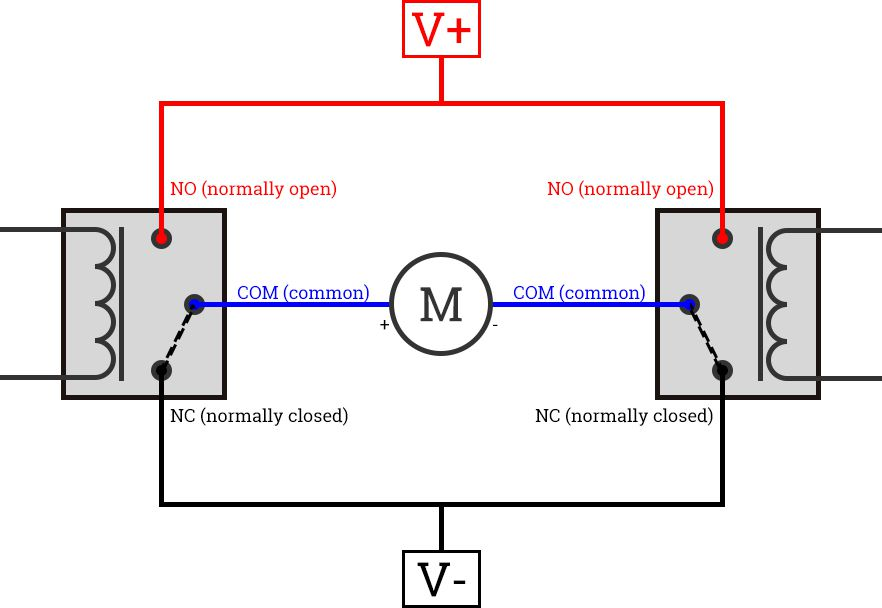
Avantages CD4051BE : utilisation de 2 CD4051BE avec 1 shiftreg, permet de brancher 16 entrées analogiques sur le seul ADC de l’ESP8266

Utilisation d’un écran LCD et d’un bouton type « rotary encoder » pour implémenter une IHM (facilité d’utilisation pour l’utilisateur et facilité d’implémentation de l’écran)

## Conception électronique

* + 1. Contrôle des moteurs

Afin de contrôler les moteurs, leur alimentation a été « hi-jacké » directement au niveau du bloc boutons, et relié à un bloc 8 relais, pouvant être piloté par une carte programmable pour la suite du projet.

Le câblage a été réalisé comme expliqué sur le schéma suivant :

Ce type de câblage permet donc de contrôler les moteurs dans un sens, comme dans l’autre, en ouvrant les relais un à un.

* + 1. Choix des composants

Premier test avec Arduino Mega



Suite avec ESP8266 E12



Utilisation de registres à décalage SN74HC595N

Explication du bidule

Explication de pourquoi qu’on l’a utilisé

[Documentation](http://www.ti.com/lit/ds/symlink/sn74hc595.pdf) ici

Coût en I/O : 3 (coût indépendant du nombre de registres)

Gain en O : 24

Utilisation de multiplexeur CD4051BE

Explication du bidule

Screen de la doc

On l’a utilisé car manque d’I/O

Branché sur un SR

Coût en I/O : 1 (intégralement câblé sur un registre à décalage)

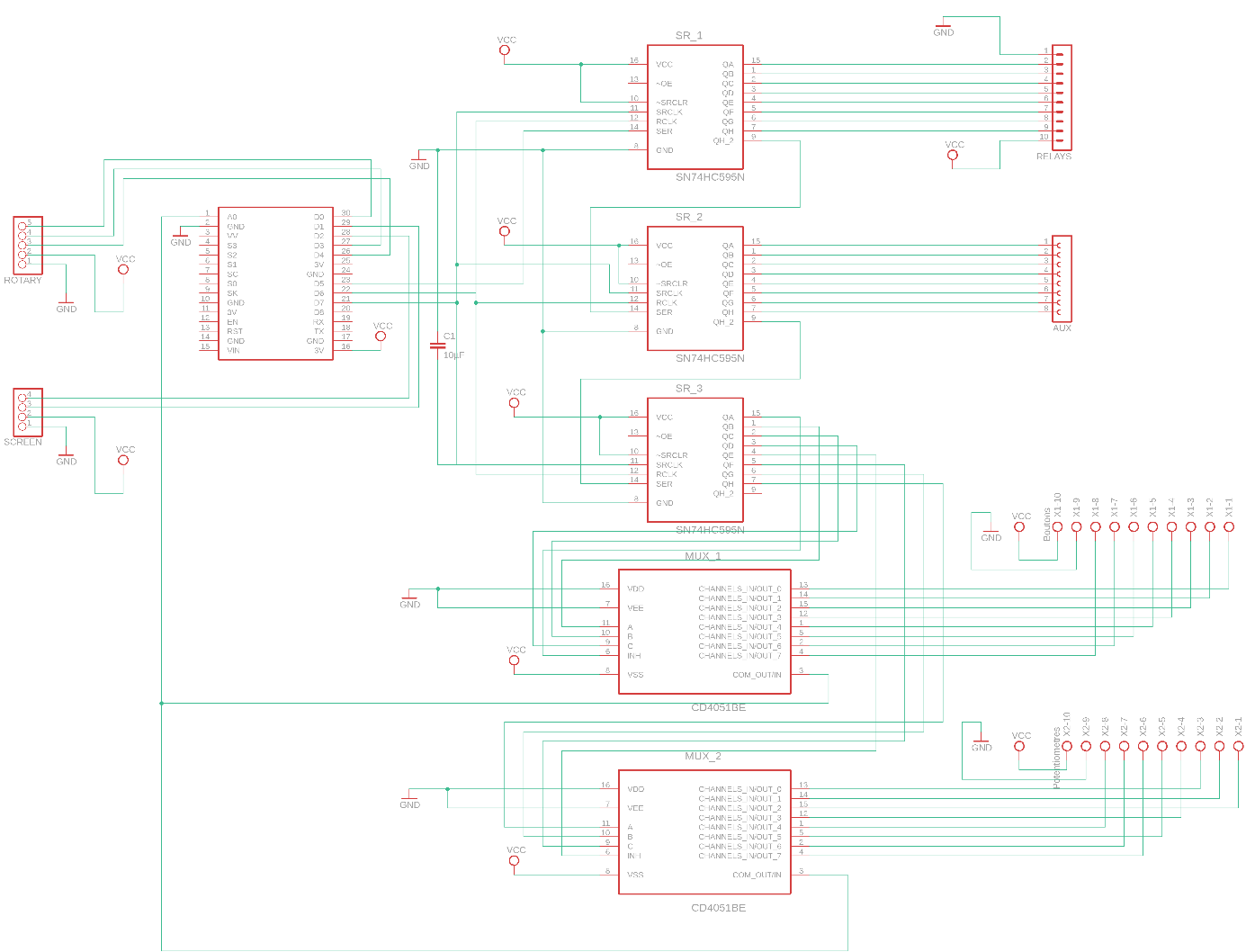
Gain en I : 16

Potentiomètres analogiques

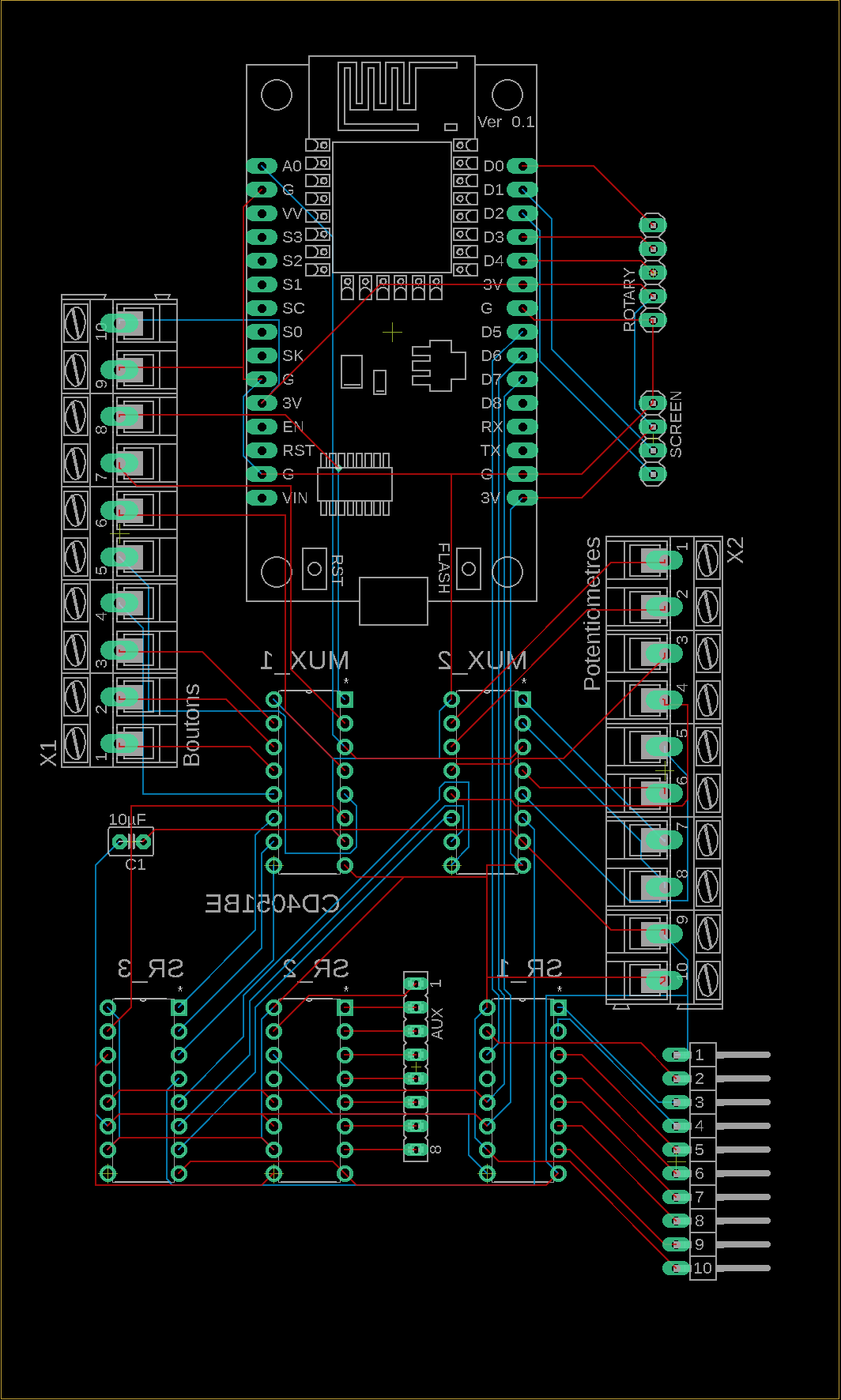
Rotary encoder (avec bouton switch)

Ecran LCD 2x16 caractères



* + 1. Câblage
    2. Création d’un PCB

Afin de rendre le système plus résistant mécaniquement, le design d’un PCB s’est imposé, afin de pouvoir connecter les différents composant de manière plus efficace et plus pérenne, le tout dans un encombrement minimum afin d’être fixé sur le châssis métallique du siège.



## Conception informatique

Langage : c++

Création librairies

Motor

ShiftReg

CustomMux

Menu

* User
* Position

## Conception mécanique

Une des parties les plus complexes vis-à-vis de nos compétences aura été la conception mécanique d’adaptateurs pour fixer les capteurs de position nécéssaires à la fonctionnalité « mémoire de position ». Jusqu’ici, deux adaptateurs ont été créés : l’adaptateur pour le capteur « avancement » et l’adaptateur pour le mouvement « hauteur »

