

**Clément FOLLET**  
**Tuteur: Igor SILVA**  
**Dirigé par M. RUI SILVA**

**Stage ERASMUS réalisé à**



## **RAPPORT DE STAGE**

**Sujet: SYSTEME DE CONTROLE D'UN ASCENSEUR**



**UNIVERSITE DES SCIENCES ET  
TECHNOLOGIES DE LILLE  
INSTITUT UNIVERSITAIRE TECHNOLOGIQUE  
DE LILLE "A"**

**Département : Mesures Physiques**

**Juin 2005**



## **REMERCIEMENTS**

Je remercie le M. Rui Silva pour m’ avoir permis d’ effectuer ce stage dans son laboratoire de recherche.

Je remercie M. Igor Silva pour sa pédagogie et ses efforts pour parler français, ainsi que pour sa patience et sa disponibilité.

Je remercie Mme Marina Sousa pour sa bonne humeur et la générosité de m’ avoir permis d’ utiliser son ordinateur pour effectuer des recherches sur internet.

Je remercie le responsable des stages Erasmus de l’ IUT pour nous avoir facilité le départ en stage.

Je remercie les responsables des Bureaux des Relations Internationales de l’ ISEP et de l’ IUT de Lille « A » pour leurs aides administratives.

Je remercie Mme Rosa Pinto pour nous avoir permis de trouver facilement un logement dans le centre-ville de Porto.

Je remercie Rémi da Silva pour son cours sur le langage C.

Je remercie Ms Miguel Machado et Telmo Ferreira, responsables de la salle Internet de l’ ISEP pour leur accueil et leur sens de l’ humour.

Je remercie les employés du restaurant universitaire.

## **SYNTHESE**

Mon stage de fin de DUT s'est déroulé à Porto, deuxième ville du Portugal, où j'ai été accueilli en tant qu'étudiant Erasmus dans le département Física de l'Instituto Superior de Engenharia appartenant à l'Instituto Politécnico de Porto.

Avant mon arrivée, j'avais choisi de prendre comme sujet de stage la simulation d'un système de contrôle de garage. Etant relativement similaire à ce sujet, mon tuteur m'a conseillé d'étudier le système de contrôle d'un ascenseur, celui-ci étant plus intéressant.

Mon projet porte donc sur l'étude d'un système de contrôle d'ascenseur. Ce sujet comprend l'étude d'un microcontrôleur et sa programmation.

J'ai travaillé ces deux mois et demi de stage dans un laboratoire de l'université, avec l'accord de M. Rui Silva et l'aide de M. Igor Silva, afin de mettre au point ce programme; et ai également utilisé de cet période pour profiter des joies et des enrichissements que peuvent apporter un stage Erasmus.

# SOMMAIRE

	page
<b>REMERCIEMENTS</b>	3
<b>SYNTHESE</b>	4
<b>INTRODUCTION</b>	6
<b>I. <u>Présentation des lieux.</u></b>	
1. L'IPP et L'ISEP.....	8
2. Le laboratoire.....	8
<b>II. <u>Le système de contrôle d'un ascenseur.</u></b>	
1. Les objectifs.....	11
2. Présentation du système.	
2.1. Description.....	13
2.2. Les composants.....	14
2.2.1. L'architecture du système.....	14
2.2.1. Le choix des capteurs.....	16
2.2.2. Le microcontrôleur AT89C52.....	17
4. La programmation.	
4.1. Le logiciel $\mu$ Vision.....	21
4.2. Le cahier des charges et les solutions.....	23
4.3. Les choix de programmation.....	27
4.4. Tests et validation.....	34
<b>CONCLUSION</b>	38
<b>ANNEXES</b>	39
<b>BIBLIOGRAPHIE</b>	39

# INTRODUCTION

Le stage que j'ai effectué de début Avril à mi-Juin 2005 est une simulation d'un système de contrôle d'ascenseur. Son approche est purement théorique et ne fait donc appel à aucune simulation réelle.

L'étude du système de contrôle et la programmation ont été progressives, mon tuteur me donnant des explications sur les problèmes rencontrés et m'indiquant ceux à résoudre.

Cette progression est résumée ci-dessous :

La réalisation de la simulation a commencé par une brève étude du système mécanique et électronique de l'ascenseur pour pouvoir imaginer son fonctionnement et ainsi permettre la future programmation.

L'étude des documents constructeurs du microcontrôleur utilisé et sa compréhension a pris une part importante dans la réalisation du stage.

Une fois avoir compris le fonctionnement d'un microcontrôleur, la programmation sur le logiciel  $\mu$ Vision, logiciel de simulation de microcontrôleurs, a pu commencer, notamment avec l'initialisation et la connection du microcontrôleur avec le reste du système.

La programmation a pu ensuite continuer en essayant de respecter scrupuleusement le cahier des charges.

Une fois la philosophie du microcontrôleur comprise, j'ai du apprendre à me servir des interruptions et trouver dans l'architecture du programme un compromis entre celles-ci et le programme déjà écrit, le programme fonctionnant quasiment.

Les grandes parties de ce présent rapport de stage concernant le système de contrôle d'un ascenseur vont donc traiter dans un premier temps de la description du système et de ses composants, puis dans un deuxième temps la programmation.

Ensuite, une conclusion mettra en avant le bilan et les apports du stage.

Mais tout d'abord, parlons brièvement du lieu de travail où j'ai passé ces deux mois et demi de stage Erasmus.

# **PRESENTATION DES LIEUX**

## **1. L'IPP et l'ISEP.**

- L'IPP (Institut Polytechnique de Porto), institution publique d'enseignement supérieur, a été créé en 1985. Il regroupe de nombreuses écoles dont :
  - l'Ecole Supérieure d'Education (ESE),
  - l'Ecole Supérieure de Musique et des Arts du spectacle(ESM),
  - l'Institut Supérieur d'Ingénierie (ISEP),
  - l'Institut Supérieur de la Comptabilité et de l'Administration (ISCAP),
  - l'Ecole Supérieure des Etudes Industrielles et de Gestion (ESEIG),
  - l'Ecole Supérieure de la Technologie et de la Gestion de Felgueiras (ESTGF),
  - l'Ecole Supérieure des Technologies de la Santé de Porto(ESTSP).



- L'ISEP, baptisé de ce nom en 1974, fut autrefois une école puis un institut industriel et, bien qu'aujourd'hui centenaire, elle n'a été intégrée à l'IPP qu'en 1988.

Huit départements font partie de l'ISEP : Mathématiques, Physique, Ingénierie Chimique, Ingénierie Informatique, Ingénierie Electrotechnique, Ingénierie Géotechnique, Ingénierie Civil et Ingénierie Mécanique. De plus, l'institut possède des Laboratoires et des Centres de Recherche.



L'ISEP, c'est aussi 5700 étudiants, 480 professeurs dont un quart d'entre eux sont maître ou docteurs, 110 fonctionnaires, travaillant dans des locaux d'une superficie totale de 40000m<sup>2</sup>.

Dans le but de s'ouvrir à l'Europe l'ISEP a développé, dans le cadre des programmes Erasmus/Socrates, une coopération avec des universités européennes afin de faire découvrir aux étudiants étrangers l'institut, et ainsi se créer des relations, internationales.

L'ISEP est également ouvert à la vie culturelle. C'est une scène d'innombrables activités culturelles: spectacles, concerts, expositions, conférences, cycle de cinéma ...

## **2. Le laboratoire.**

Le laboratoire dans lequel j'ai travaillé durant ces 2 mois ½ fait partie du CIEA, le Centre de Recherche en Ingénierie Appliquée. Ce centre compte une cinquantaine de chercheurs qui travaillent en collaboration avec l'industrie et d'autres institutions afin de

développer la recherche dans les domaines de la dégradation des matériaux, des polymères, ...

# **SYSTEME DE CONTROLE D'ASCENSEUR**

## 1. Les objectifs.

Les objectifs ont été formulés à l'aide d'un cahier des charges, donné à la première réunion par notre tuteur. Le cahier des charges initialement en portugais a été traduit en français avec l'aide de notre tuteur et d'internet. Voici la version française :

# Cahier des charges

## Système de contrôle d'un ascenseur

### 1. Conditions du système

#### 1.1 Objectifs

Le système à développer a pour but de contrôler un ascenseur.

Le projet de ce système devra garantir l'intégrité physique des utilisateurs et gèrera de façon intelligente ses ressources.

#### 1.2 Exigences du système

L'ascenseur doit couvrir 10 étages, soit en montant soit en descendant, tout en étant capable de transporter des personnes et des biens, quelque soit les combinaisons des étages de départ et de destination.

L'ascenseur doit se déplacer avec 2 vitesses différentes. La vitesse maximale sera la vitesse de croisière ou de transport, la vitesse minimale sera quant à elle uniquement utilisée lors du freinage.

Le système doit être capable de gérer simultanément les appels d'au moins 10 utilisateurs. La gestion des appels doit minimiser le nombre de changement de sens. C'est pourquoi l'utilisateur se trouvant à l'extérieur de la cabine de l'ascenseur pourra lors de l'appel indiquer le sens dans lequel il veut aller. Une fois à l'intérieur de la cabine, il pourra sélectionner l'étage désiré. De plus, la charge à transporter devra seulement être la charge utile pour le trajet à parcourir.

Les portes d'accès de la cabine ne pourront s'ouvrir qu'au moment où la cabine de l'ascenseur sera immobilisée.

Une fois la cabine immobilisée, le système de contrôle ouvrira les portes et devra les maintenir ouvertes durant un laps de temps minimum de 5 secondes, garantissant ainsi la sécurité de l'utilisateur durant l'accès à la cabine. Cette période de temps doit être comptée à partir du moment où l'ascenseur se trouve immobilisé dans l'étage d'appel et avec les portes d'accès complètement ouvertes. Le système de contrôle doit être capable d'identifier la présence d'utilisateurs ou d'objets se trouvant dans le cours d'accès à la cabine. Si une présence est détectée, le système de contrôle devra rouvrir complètement les portes d'accès ou les maintenir ouvertes, puis attendre l'arrêt de la détection. Terminée la détection, le système doit attendre 5 secondes supplémentaires avant de fermer les portes et de reprendre son fonctionnement normal.

Pendant le trajet, les portes doivent toujours être maintenu fermées. En aucun cas la cabine de l'ascenseur devra entrer en mouvement sans que les portes d'accès soient complètement fermées.

L'ascenseur doit être capable d'élever une charge maximale de 750 Kg, le poids de la cabine n'étant pas pris en compte. Chaque fois que la charge maximale sera dépassée, le système de contrôle de l'ascenseur actionnera une alarme sonore/lumineuse. Puis il attendra en gardant la

cabine immobilisée que le poids de cette dernière redescende en dessous du seuil toléré, tout en maintenant les portes ouvertes ou en les rouvrant complètement.  
Le système devra indiquer l'étage auquel se trouve la cabine à chaque instant.

### 1.3 Restrictions du système

Le circuit intégré utilisé sera le microcontrôleur 89C51 d'Atmel, celui-ci ayant une fréquence de fonctionnement de 16 MHz.

Le moteur à utiliser pour élever ou abaisser la cabine de l'ascenseur sera contrôlé par un convertisseur externe de 3 bits, ces derniers définissant le sens et les 4 niveaux possibles de vitesse.

La signalisation de l'étage auquel se trouve l'ascenseur devra être effectuée au moyen d'un décodeur 7 segments se composant de 4 bits d'entrée et d'un affichage LCD à 7 segments.

Le capteur de pression dans le plancher de la cabine émettra un signal analogique décodé à 16 niveaux par un convertisseur analogique/numérique. À chaque niveau correspondra la pression exercée par un poids de 50 Kg.

Le signal utilisé pour produire l'alarme sonore/lumineuse d'excès de poids devra être généré par le contrôleur.

On utilisera un moteur pas-à-pas pour commander les portes d'accès à la cabine. La génération des signaux de contrôle des moteurs pas-à-pas utilisés sera effectuée par le système de contrôle de l'ascenseur.

Les capteurs de présence des portes d'accès produiront des signaux binaires.

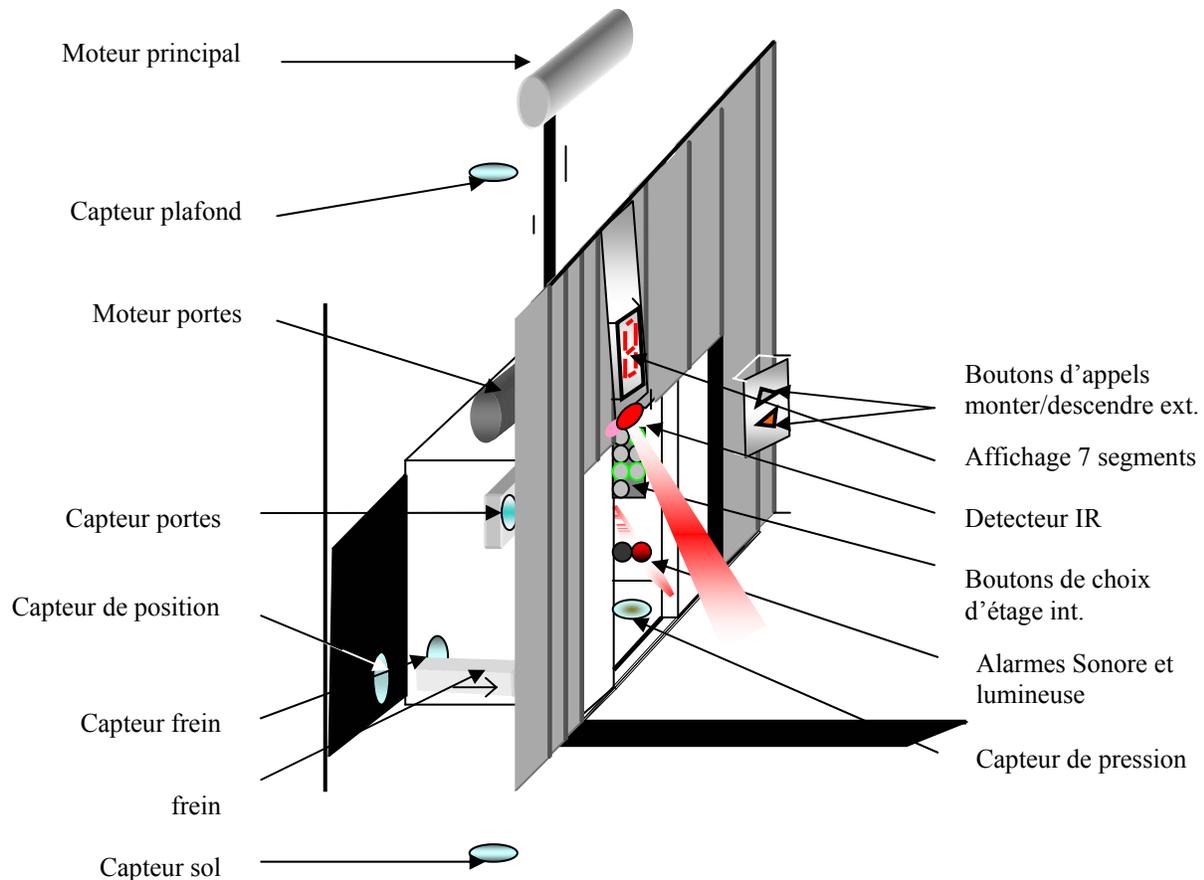
Pour s'assurer que les portes soit complètement fermées ou ouvertes, un capteur produisant un signal de type binaire sera utilisé pour définir le fin-de-course et l'activation du frein.

L'interface avec l'utilisateur sera réalisée à l'aide d'un système de boutons contrôlés par un microcontrôleur externe au système de contrôle de l'ascenseur. Les appels des utilisateurs seront reçus par le système de contrôle de l'ascenseur via un port série avec un baud rate de 9600 baud. Les informations seront transmises suivant un protocole à définir.

Un mécanisme générant un signal de type binaire devra être considéré pour détecter une éventuelle défaillance dans le fonctionnement normal de la cabine de l'ascenseur.

## 2. Présentation du système.

### 2.1. Description.



Le système se compose d'un seul et unique ascenseur couvrant 10 étages et pouvant se déplacer et montant et en descendant afin de transporter des personnes ou des biens quelque soit les combinaisons d'étages de départ et de destination.

Pour se faire, l'ascenseur se déplace à deux vitesses, une vitesse de croisière et une vitesse de freinage. Il a donc fallu placer des capteurs de position en conséquence. Il y a 3 capteurs par étages intermédiaires deux capteurs à chaque étage extrême, ce qui nous fait en tout  $8 \times 3 + 2 \times 2$  soit 28 capteurs de position.

Il y a en plus 2 capteurs aux extrémités du couloir d'élévation de l'ascenseur. Un capteur au sol coïncident avec le capteur de position de l'étage 0, cela veut dire que le capteur sol et ce capteur de position seront actifs au même moment. Un capteur au plafond qui ne coïncide pas avec le capteur de position de l'étage 9.

Le système a aussi besoin de prendre en compte les appels des utilisateurs. Il existe donc des boutons d'appel à chaque étage, un bouton d'appel pour monter ainsi qu'un pour descendre. Sachant que quand nous sommes à l'étage 0 et à l'étage 9 nous ne

pouvons demander qu'un seul sens, cela nous fait au total 18 boutons d'appel extérieurs. N'oublions pas non plus le choix des étages qui se fait à l'intérieur de la cabine grâce une interface de 10 boutons.

Par addition, nous avons donc 28 boutons d'appel extérieurs et intérieurs.

Le système se compose également d'un moteur ayant pour ordre de déplacer la cabine. Ce moteur consommant une grande partie d'énergie et mettant en jeu des couples importants, le nombre de changements de sens sera à limiter, le poids de la cabine (plus les personnes) sera lui aussi à minimiser.

Un deuxième moteur sert à ouvrir et fermer les portes. Un capteur donnera à chaque instant l'information sur l'état des portes (ouvertes/fermées) et contrôlera ainsi leurs ouverture et leurs fermeture.

De plus, il existe un frein servant à maintenir les portes fermées durant le trajet. L'état du frein est donné par un capteur.

Un détecteur de présence faisant partie de la cabine commande la réouverture des portes ainsi que la temporisation.

Des afficheurs 7 segments (un par étage) donneront des informations sur le sens de déplacement de la cabine et sur l'étage, ce qui permet entre autre une estimation du temps d'attente pour l'utilisateur.

Une alarme sonore et une alarme lumineuse servent à faire sortir les personnes de la cabine si un excès de poids où une défaillance du système sont détectés.

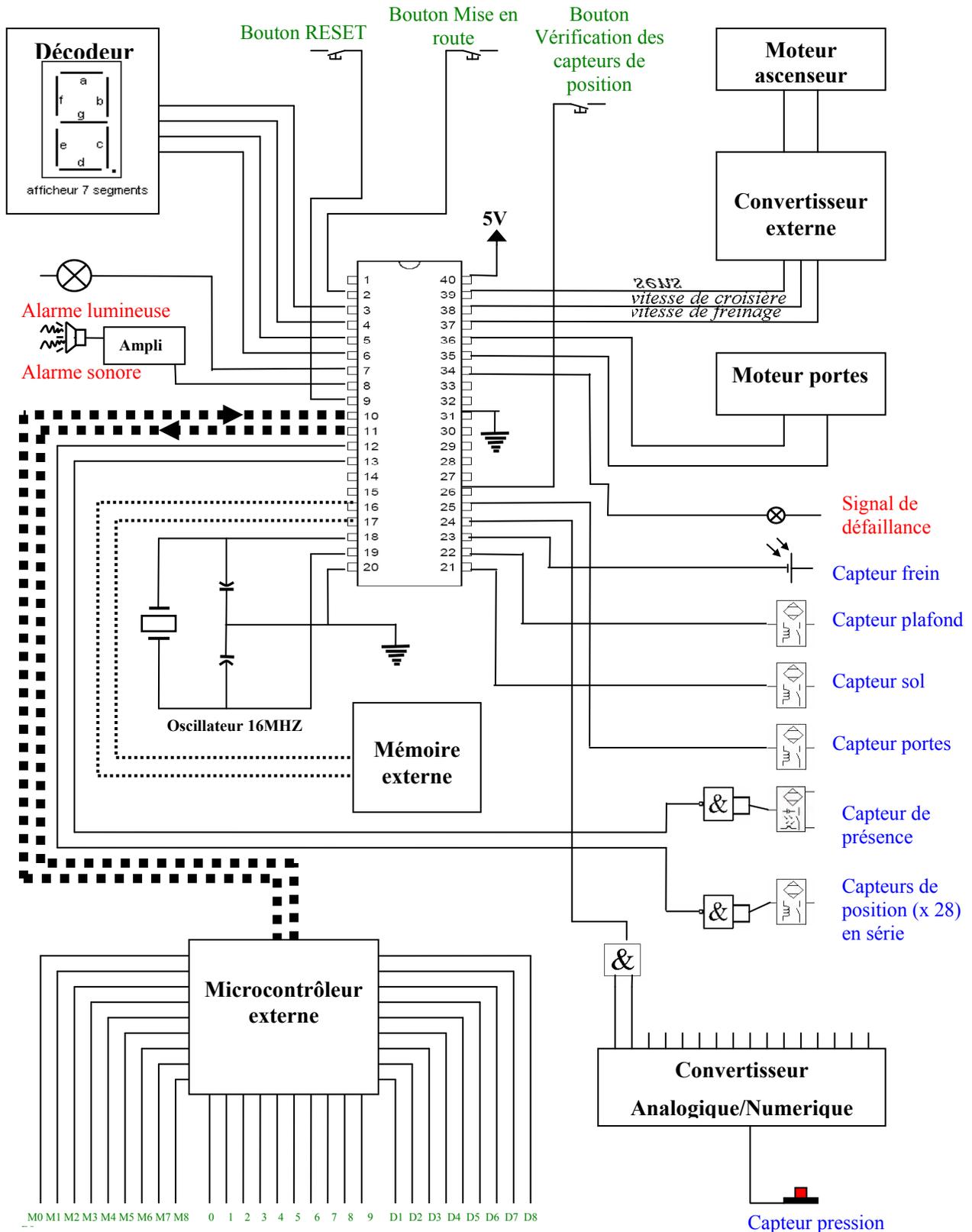
L'excès de poids est détecté par un capteur de pression se trouvant dans le plancher de la cabine.

## **2.2. Les composants.**

### **2.2.1. L'architecture du système.**

Voir map page suivante.

# Map de l'électronique d'un système de contrôle d'ascenseur



## 2.2.2. Le choix des capteurs.

- Les capteurs de detection de la cabine, le capteur portes et le capteur frein.



Les capteurs utilisés pour la detection de la cabine ainsi que le capteur portes et le capteur frein peuvent être des détecteurs de proximité inductifs.

La technologie des détecteurs de proximité inductifs est basée sur la variation d'un champ magnétique à l'approche d'un objet conducteur du courant électrique. Ce type de capteur est donc réservé à la détection sans contact d'objets métalliques comme une cabine d'ascenseur par exemple.

Leur portée est pour les plus courant de 50 mm, ce qui n'est pas suffisant pour un système d'ascenseur, des détecteurs spéciaux peuvent donc être demandés.

Le principal avantage de ces détecteurs est leur durée de vie puisqu'il n'y a aucun contact physique avec l'objet détecté. Autre avantage, ils ne demandent aucun entretien contrairement aux détecteurs de proximité photoélectriques.

- Le capteur de présence.

Un détecteur de proximité photoélectrique peut être utilisé pour la détection des usagers dans le cours d'accès de l'ascenseur. Ce détecteur est de système proximité, c'est à dire qu'il est à la fois émetteur de lumière et récepteur photosensible. Le faisceau infrarouge invisible envoyé est réfracté par une personne ou un objet, un pic sur le signal de reception correspond donc à une obstruction du cours d'accès, ce qui engendre un changement d'état sur le signal de sortie. Il existe d'autres systèmes de détection photoélectriques comme le système barrage avec un émetteur et un récepteur, ou encore le système reflex avec un emetteur-recepteur et un reflecteur. Un faisceau de lumière coupé est alors correspondant à un signal nul à la sortie du recepneur.

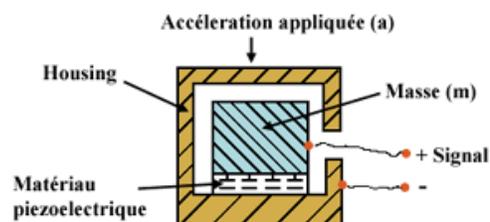


- Le capteur de pression.

Enfin, le capteur utilisé pour mesurer le poids des personnes dans la cabine peut être un capteur piézoélectrique à compression.



Cette technologie repose sur une propriété du crystal de quartz, qui sous l'effet d'une force se polarise et donc crée une différence de potentiel. Cette différence de potentiel étant proportionnelle à la force, le poids peut en être déduit.



### 2.2.3. Le Microcontrôleur et son initialisation.

Ce microcontrôleur de marque ATMEL est un circuit intégré disposant de fonctionnalités aidant à la programmation de systèmes relativement simple tels que celui de l'ascenseur.

Il dispose entre autres de 256 bytes de mémoire interne, de 32 entrées/sorties réparties sur 4 ports, de 3 timer/counters, de 8 sources d'interruption et d'autres fonctionnalités comme le mode Idle ou encore le mode Power-down.

De plus, de la mémoire externe peut être ajouté.

Il coûte 170 €.

#### 3.2.3.1. Quelques fonctions du microcontrôleur.

Différents espaces mémoire sont réservés sur le microcontrôleur pour stocker non seulement le programme et ses variables, mais aussi pour le configurer et enregistrer les états des registres utilisées par ses fonctionnalités.

Ces registres sont stockés dans le SFR(Special Function Register)

Ces registres sont modifiables, une grande partie d'entre eux servent à configurer les fonctions du microcontrôleur, en voici les plus importantes.

Figure 2. SFR Map. (...) Indicates Resident in AT89C52, not in AT89C51.

8 Bytes								
F8								FF
F0	B							F7
E8								EF
E0	ACC							E7
D8								DF
D0	PSW							D7
C8	(T2CON)	(T2MOD)	(RCAP2L)	(RCAP2H)	(TL2)	(TH2)		CF
C0								C7
B8	IP							BF
B0	P3							B7
A8	IE							AF
A0	P2							A7
98	SCON	SBUF						9F
90	P1							97
88	TCON	TMOD	TL0	TL1	TH0	TH1		8F
80	P0	SP	DPL	DPH			PCON	87

- Les interruptions.

Il existe sur le microcontrôleur des interruptions permettant de définir un évènement prioritaire sur un autre. Par exemple les capteurs de position sont prioritaires, l'ascenseur devant savoir à tout moment à quel étage il se trouve, ils ont été reliés à l'entrée P3.2 qui correspond à l'interruption INT0.

Ces interruptions permettent au système d'être guidé par des évènements et simplifient la programmation.

Une interruption a lieu à chaque transition 1-0 de l'entrée concernée, cela provoque un arrêt dans le cours normal d'exécution du programme, la fonction interrupt X est alors appelée. Une fois l'interruption terminée, on revient à l'endroit où le programme avait été interrompu.

On peut autoriser ou interdire les différentes interruptions par la mise à 1 de bits correspondant à chaque source d'interruption dans le registre IE du SFR.

Il existe aussi un registre de priorité qui permet de garder une interruption prioritaire, même si une autre interruption est en cours (voir ci-dessous).

Table 5. Interrupt Enable (IE) Register

(MSB)								(LSB)
EA	-	ET2	ES	ET1	EX1	ET0	EX0	

Enable Bit = 1 enables the interrupt.  
Enable Bit = 0 disables the interrupt.

Symbol	Position	Function
EA	IE.7	Disables all interrupts. If EA = 0, no interrupt is acknowledged. If EA = 1, each interrupt source is individually enabled or disabled by setting or clearing its enable bit.
-	IE.6	Reserved.
ET2	IE.5	Timer 2 interrupt enable bit.
ES	IE.4	Serial Port interrupt enable bit.
ET1	IE.3	Timer 1 interrupt enable bit.
EX1	IE.2	External interrupt 1 enable bit.
ET0	IE.1	Timer 0 interrupt enable bit.
EX0	IE.0	External interrupt 0 enable bit.

User software should never write 1s to unimplemented bits, because they may be used in future AT89 products.

Figure 24. IP: Interrupt Priority Register

(MSB)								(LSB)
-	-	PT2	PS	PT1	PX1	PT0	PX0	

Priority bit = 1 assigns high priority.  
Priority bit = 0 assigns low priority.

Symbol	Position	Function
-	IP.7	reserved
-	IP.6	reserved
PT2	IP.5	Timer 2 Interrupt priority bit.
PS	IP.4	Serial Port Interrupt priority bit.
PT1	IP.3	Timer 1 Interrupt priority bit.
PX1	IP.2	External Interrupt 1 priority bit.
PT0	IP.1	Timer 0 Interrupt priority bit.
PX0	IP.0	External Interrupt 0 priority bit.

User software should never write 1s to unimplemented bits, since they may be used in future AT89 Series products.

- les timer/counters.

Les timer/counters possèdent eux aussi des sources d'interruptions mais ce n'est pas leur principal caractéristique. Ils permettent comme le nom l'indique de compter et sont principalement utilisés pour effectuer des temporisations.

Ils peuvent compter de différentes manières suivant le mode choisi. Le mode utilisé pour la temporisation de 5 secondes est mode 1 qui permet au timer de compter en 16 bit d'une valeur hexadécimale de départ choisie à la valeur 0FFFF. Une fois la valeur atteinte, le drapeau d'interruption se met à 1 et reste à 1. Les autres modes permettent de compter avec 8 bits ou en auto-reload (on recharge une valeur de départ à chaque cycle du timer).

Les timers/counters peuvent aussi être configurés en générateur de baud rate. Cela permet d'avoir une liaison série avec par exemple un autre microcontrôleur pour recevoir des informations. Dans le cas de l'ascenseur, c'est le timer 2 qui se charge de générer le baud rate de 9600 baud pour permettre de recevoir les appels des utilisateurs. Le mode baud rate generator est un mode auto-reload, il a donc fallu calculer une valeur de départ(RCAP2H,RCAP2L) pour obtenir le baud rate voulu.

L'équation est (pour information):

$$RCAP2H,RCAP2L = 65536 - (\text{fréquence de l'oscillateur}/(32 \times \text{Baud Rate}))$$

Remarque: la fréquence d'oscillation de 16MHz est générée par un crystal de quartz. Sa fréquence d'oscillation peut être choisie de 0 à 24 MHz

Toutes ces fonctionnalités sont configurées dans les registres TMOD(Timer/Counter Mode Control Register) et TCON(Timer/Counter Control Register) pour le timer 0 et 1; et dans T2MOD et T2CON pour le timer 2. (voir Annexes)

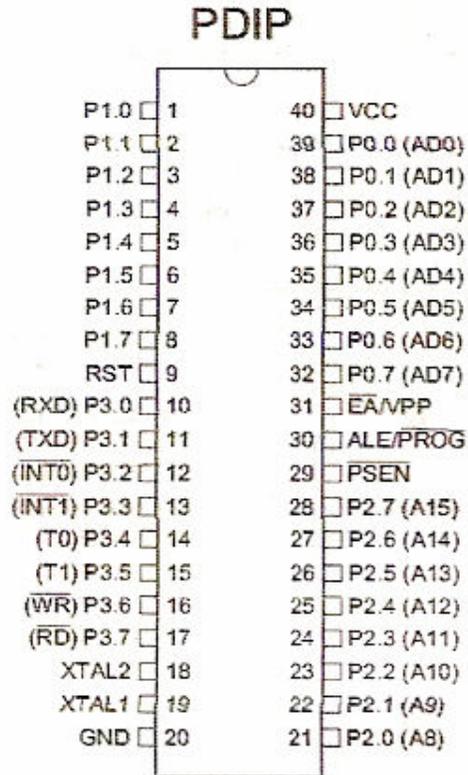
- La mémoire externe.

Pour ajouter de la mémoire externe (64Kbytes) au microcontrôleur, il suffit de mettre la pin EA à la masse et de libérer les pins P3.6 et P3.7 pour faire la communication.

### 3.2.3.2. Le branchement du microcontrôleur.

Le branchement du microcontrôleur s'est fait selon les priorités des événements extérieurs et les fonctions du microcontrôleur (voir branchement page suivante).

## ATTRIBUTION DES ENTREES/SORTIES DU MICROCONTROLEUR AT89C51



Baud rate generator	<b>P1.0</b>	<b>MICROCONTROLEUR AT89C51</b>	<b>VCC</b>	Alimentation
Mise en route	<b>P1.1</b>		<b>P0.0</b>	Sens (montée/descente)
Affichage étage bit 0	<b>P1.2</b>		<b>P0.1</b>	Vitesse de croisière
Affichage étage bit 1	<b>P1.3</b>		<b>P0.2</b>	Vitesse de freinage
Affichage étage bit 2	<b>P1.4</b>		<b>P0.3</b>	Ouverture des portes
Affichage étage bit 3	<b>P1.5</b>		<b>P0.4</b>	Fermeture des portes
Alarme lumineuse	<b>P1.6</b>		<b>P0.5</b>	Signal de défaillance
Signal sonore	<b>P1.7</b>		<b>P0.6</b>	-
Reset	<b>RST</b>		<b>P0.7</b>	-
Reception liaison série	<b>P3.0(RXD)</b>		<b>EA/VPP</b>	Masse
Transmission liaison série	<b>P3.1(TXD)</b>		<b>ALE/PROG</b>	-
Capteurs de position cabine	<b>P3.2(INT0)</b>		<b>PSEN</b>	-
Capteur de présence	<b>P3.3(INT1)</b>		<b>P2.7</b>	-
(temporisations) -	<b>P3.4(T0)</b>		<b>P2.6</b>	-
(signal sonore) -	<b>P3.5(T1)</b>		<b>P2.5</b>	Vérification capteurs posit'
Entrée mémoire externe	<b>P3.6</b>		<b>P2.4</b>	Capteurs portes O/F
Sortie mémoire externe	<b>P3.7</b>		<b>P2.3</b>	Capteur de poids
Horloge	<b>XTAL2</b>		<b>P2.2</b>	Capteur frein
Horloge	<b>XTAL1</b>		<b>P2.1</b>	Capteur plafond
Masse	<b>GND</b>	<b>P2.0</b>	Capteur sol (étage 0)	

## II. La programmation.

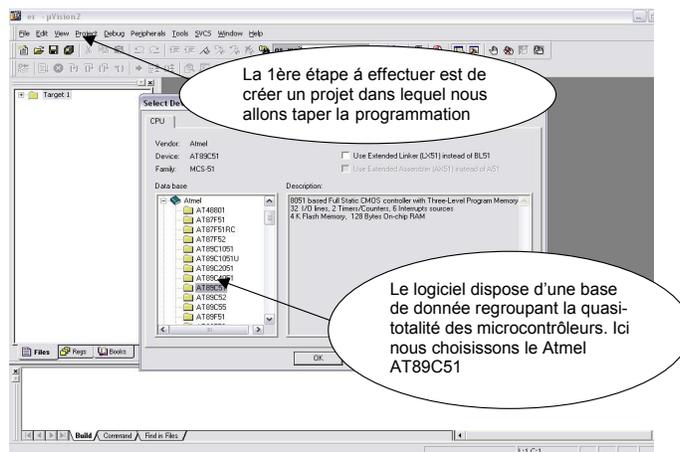
### 1. Le logiciel $\mu$ Vision2 et le langage C.

Nous avons du apprendre à utiliser le logiciel de programmation.

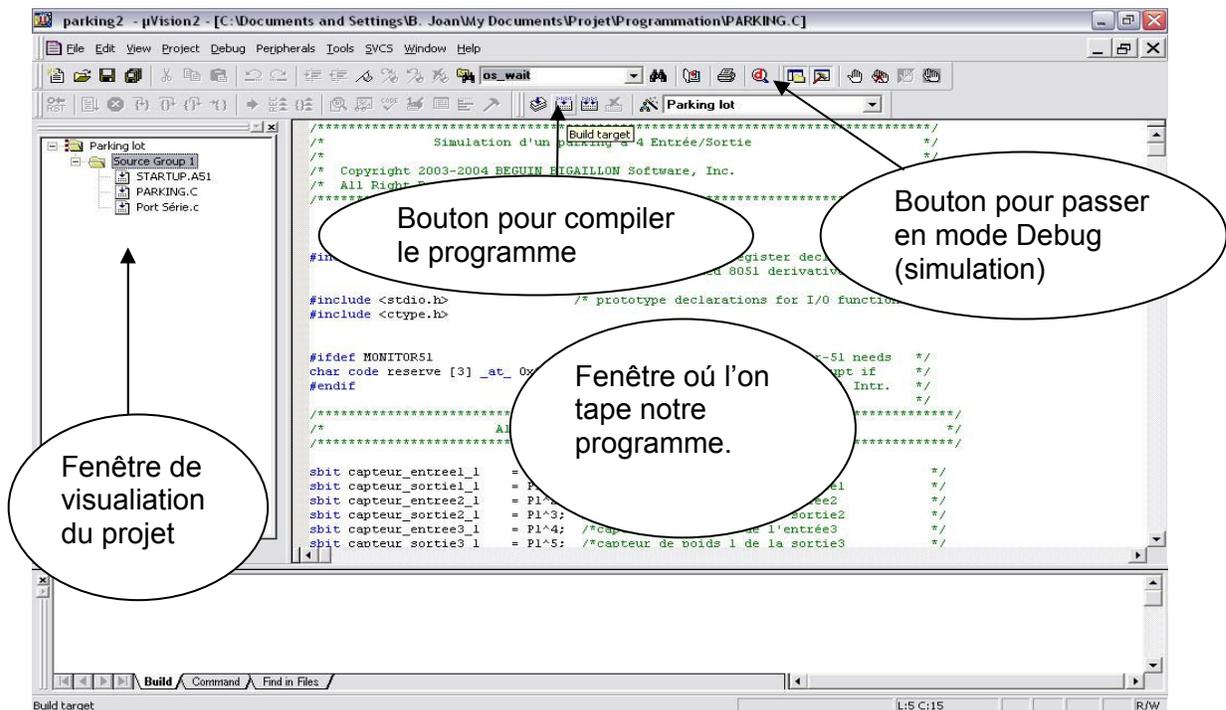
Le logiciel  $\mu$ Vision 2 a été conçu pour permettre de simuler de nombreux microcontrôleurs.

La programmation de celui-ci s'effectue en langage C.

#### 1.1. Création d'un projet et choix du microcontrôleur.

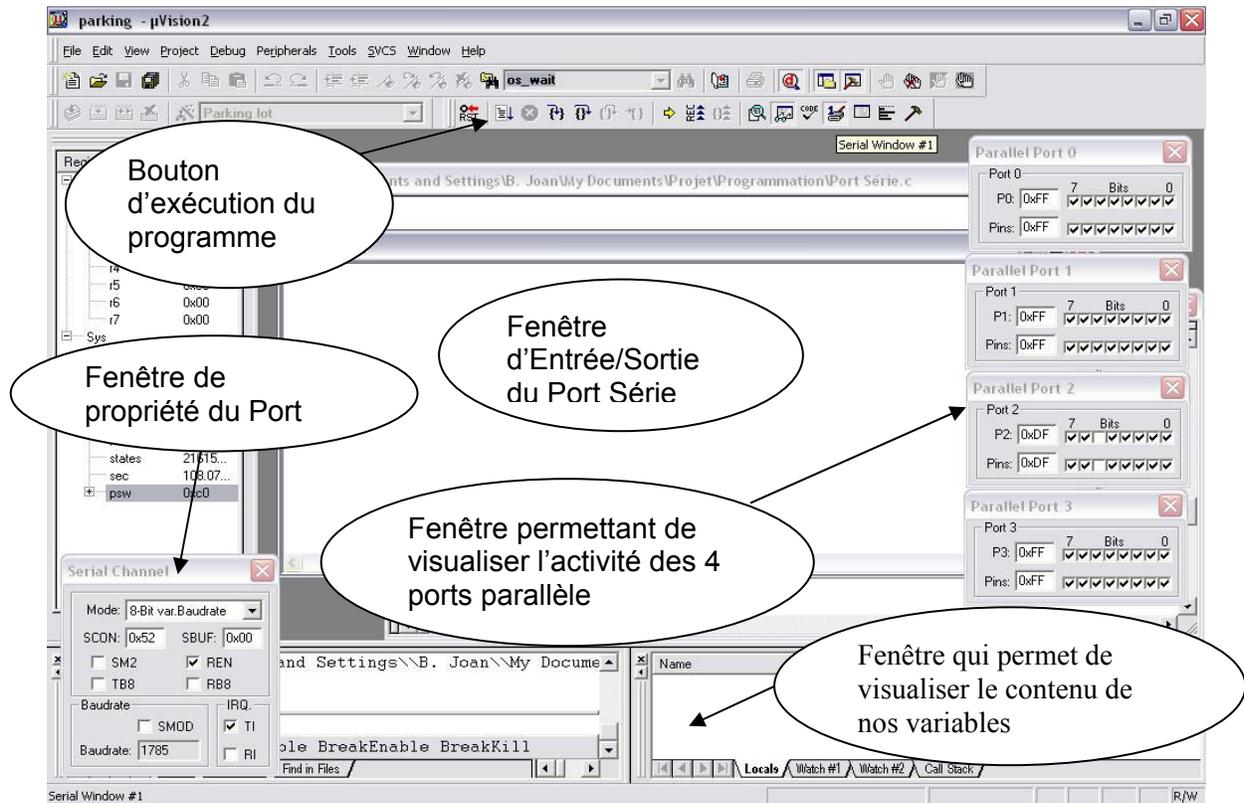


#### 1.2. En mode Programmation.



### 1.3. En mode Simulation.

La particularité de ce logiciel est qu'après la compilation du programme il simule l'activité du microcontrôleur. On va alors pouvoir choisir les bits à activer, voir le déroulement du système en observant la mise à 0 ou à 1 des bits, voir l'évolution du contenu de certaines variables, les informations qui passent par le Port Série, ...



- Le langage C.

Le langage C a été mis au point par D.Ritchie et B.W.Kernighan au début des années 70. Leur but était de développer un langage qui permettrait d'obtenir un système d'exploitation de type UNIX portable.

Le langage C reste un des langages les plus utilisés actuellement car comportant des instructions et des structures de haut niveau tout en générant un code très rapide grâce à un compilateur très performant.

## 2. Le cahier des charges et les solutions.

n°	EXIGENCES	SOLUTIONS
1	L'ascenseur doit couvrir 10 étages, soit en montant soit en descendant, tout en étant capable de transporter des personnes et des biens, quelque soit les combinaisons des étages de départ et de destination.	L'ascenseur couvre les étages 0 à 9, soit 10 étages, en montant et en descendant. Toutes les combinaisons possibles d'étages de départ et d'arrivée sont organisées suivant un ordre logique grâce à deux tableaux.
2	L'ascenseur doit se déplacer avec 2 vitesses différentes. La vitesse maximale sera la vitesse de croisière ou de transport, la vitesse minimale sera quant à elle uniquement utilisée lors du freinage.	Les deux vitesses de croisière et de freinage sont utilisées. La vitesse de freinage est aussi utilisée dans la remise à l'étage 0 de l'ascenseur et pendant la vérification des capteurs.
3	Le système doit être capable de gérer simultanément les appels d'au moins 10 utilisateurs. La gestion des appels doit minimiser le nombre de changement de sens. C'est pourquoi l'utilisateur se trouvant à l'extérieur de la cabine de l'ascenseur pourra lors de l'appel indiquer le sens dans lequel il veut aller. Une fois à l'intérieur de la cabine, il pourra sélectionner l'étage désiré.	Le système est capable de gérer simultanément les appels de 18 utilisateurs, soit le nombre d'appels maximal possible des 18 boutons extérieur. Le nombre de changement de sens est minimisé grâce à deux tableaux, l'un prenant en compte les appels pour monter, l'autre ceux pour descendre. Les étages sélectionnés à l'intérieur de la cabine sont quant à eux répartis dans un troisième tableaux, ce qui permet l'erreur d'un utilisateur de prendre la cabine dans le mauvais sens.
4	De plus, la charge à transporter devra seulement être la charge utile pour le trajet à parcourir.	La charge utile sera respectée grâce aux bon sens des utilisateurs. A chaque étage, il y a des afficheurs 7 segments indiquant les étages à chaque instant, l'utilisateur devra regarder avant d'entrer dans la cabine si celle-ci se déplace dans le sens voulu.
5	Les portes d'accès de la cabine ne pourront s'ouvrir qu'au moment où la cabine de l'ascenseur sera immobilisée.	Il existe un frein qui ne permet l'ouverture des portes qu'au moment où la cabine se trouve en face du cours d'accès. De plus, le programme considère une défaillance du système si la cabine n'est pas immobilisée, c'est à dire lorsqu'il y a un problème avec le moteur d'élévation.
6	Une fois la cabine immobilisée, le système de contrôle ouvrira les portes et devra les maintenir ouvertes durant un laps de temps minimum de 5 secondes, garantissant ainsi la sécurité de l'utilisateur durant l'accès à	Le timer 0 fait la temporisation.

	la cabine. Cette période de temps doit être comptée à partir du moment où l'ascenseur se trouve immobilisé dans l'étage d'appel et avec les portes d'accès complètement ouvertes.	
7	Le système de contrôle doit être capable d'identifier la présence d'utilisateurs ou d'objets se trouvant dans le cours d'accès à la cabine. Si une présence est détectée, le système de contrôle devra rouvrir complètement les portes d'accès ou les maintenir ouvertes, puis attendre l'arrêt de la détection. Terminée la détection, le système doit attendre 5 secondes supplémentaires avant de fermer les portes et de reprendre son fonctionnement normal.	Le capteur de présence crée une interruption.
8	Pendant le trajet, les portes doivent toujours être maintenu fermées. En aucun cas la cabine de l'ascenseur devra entrer en mouvement sans que les portes d'accès soient complètement fermées.	Si les portes n'arrivent pas à se fermer, le programme considérera une défaillance du système. Les portes seront donc obligatoirement fermées pendant le trajet. De plus, le frein les maintiendra fermées.
9	L'ascenseur doit être capable d'élever une charge maximale de 750 Kg, le poids de la cabine n'étant pas pris en compte. Chaque fois que la charge maximale sera dépassée, le système de contrôle de l'ascenseur actionnera une alarme sonore/lumineuse. Puis il attendra en gardant la cabine immobilisée que le poids de cette dernière redescende en dessous du seuil toléré, tout en maintenant les portes ouvertes ou en les rouvrant complètement.	Le capteur de poids déclenche cette procédure.
10	Le système devra indiquer l'étage auquel se trouve la cabine à chaque instant.	Le comptage des capteurs de position donne l'étage.

	<b>RESTRICTIONS</b>	<b>SOLUTIONS</b>
11	Le circuit intégré utilisé sera le microcontrôleur 89C51 d'Atmel, celui-ci ayant une fréquence de fonctionnement de 16 MHz.	Le microcontrôleur utilisé est le 89C52 d'Atmel, celui-ci contenant un timer supplémentaire et 256 bytes de RAM contre 128 bytes pour le C51. Il fonctionne à une fréquence de 16Mhz.
12	Le moteur à utiliser pour élever ou abaisser la cabine de l'ascenseur sera contrôlé par un convertisseur externe de 3 bits, ces derniers définissant le sens et les 4 niveaux possibles de vitesse.	Trois sorties sont utilisées sur le microcontrôleur pour commander le sens, la vitesse de croisière et la vitesse de freinage. Les états logique de ces sorties commandent alors le moteur à l'aide du convertisseur externe.
13	La signalisation de l'étage auquel se trouve l'ascenseur devra être effectuée au moyen d'un décodeur 7 segments se composant de 4 bits d'entrée et d'un affichage LCD à 7 segments.	Quatre sorties sont utilisées sur le microcontrôleur pour l'affichage des étages. Ces pins envoient un code binaire au décodeur permettant l'affichage décimal par les afficheurs 7 segments situés dans la cabine et à chaque étage.
14	Le capteur de pression dans le plancher de la cabine émettra un signal analogique décodé à 16 niveaux par un convertisseur analogique/numérique. À chaque niveau correspondra la pression exercée par un poids de 50 Kg.	Le convertisseur analogique/numérique envoie un code binaire qui est ensuite transformé en un signal binaire suivant le poids des personnes à l'intérieur de la cabine. L'état logique 1 correspond à un excès de poids.
15	Le signal utilisé pour produire l'alarme sonore/lumineuse d'excès de poids devra être généré par le contrôleur.	Le timer 1 du microcontrôleur se chargera de générer ces signaux d'alarme.
16	On utilisera un moteur pas-à-pas pour commander les portes d'accès à la cabine. La génération des signaux de contrôle des moteurs pas-à-pas utilisés sera effectuée par le système de contrôle de l'ascenseur.	Le microcontrôleur a deux sorties commandant le moteur pas-à-pas, une pour l'ouverture, une autre pour la fermeture.
17	Les capteurs de présence des portes d'accès produiront des signaux binaires.	Un mécanisme d'interrupteurs permet l'émission du signal binaire du capteur de présence seulement à l'étage où se trouve l'ascenseur.
18	Pour s'assurer que les portes soit complètement fermées ou ouvertes, un capteur produisant un signal de type binaire sera utilisé pour définir le fin-de-course et l'activation du frein.	Ce signal est primordial pour autoriser l'ascenseur à se déplacer, si ce signal n'est pas à l'état logique 1, le programme considèrera qu'il y a une défaillance et arrêtera l'ascenseur. Le frein devra alors être contrôlé par un agent, celui-ci devant être toujours en bon état de fonctionnement pour garantir la sécurité des utilisateurs.

19	L'interface avec l'utilisateur sera réalisée à l'aide d'un système de boutons contrôlés par un microcontrôleur externe au système de contrôle de l'ascenseur. Les appels des utilisateurs seront reçus par le système de contrôle de l'ascenseur via un port série avec un baud rate de 9600 baud. Les informations seront transmises suivant un protocole à définir.	18 entrées pour les appels extérieurs à la cabine et 10 entrées pour les 10 étages pouvant être sélectionnés à l'intérieur de la cabine sont nécessaires au microcontrôleur externe. Le baud rate utilisé pour la transmission des appels est de 9615 baud, valeur la plus proche possible de la valeur fixée par le cahier des charges. Ce baud rate est généré par le timer 2. Le protocole est défini suivant un tableau expliqué dans le chapitre suivant.
20	Un mécanisme générant un signal de type binaire devra être considéré pour détecter une éventuelle défaillance dans le fonctionnement normal de la cabine de l'ascenseur.	Le programme prend en charge les défaillances possibles dans le fonctionnement de l'ascenseur, un signal est alors généré et un reset est alors nécessaire pour remettre en route l'ascenseur. Ce reset ne peut être effectué que par un agent habilité et connaissant les problèmes que peut rencontrer l'ascenseur.

### 3. Les choix de programmation.

#### 3.1. Envoyer des appels.

Pour simuler les appels sur le logiciel, il a fallu se servir du clavier et du code ASCII, puisque le microcontrôleur ne reçoit lui que le langage binaire. Un protocole a donc été créé, il est résumé dans le tableau suivant :

Touche clavier	Code décimal	Signification
e	101	Un utilisateur est à l'étage 1 et désire descendre
f	102	“ 2 “
g	103	“ 3 “
h	104	“ 4 “
i	105	“ 5 “
j	106	“ 6 “
k	107	“ 7 “
l	108	“ 8 “
m	109	“ 9 “
n	110	Un utilisateur est à l'étage 0 et désire monter
o	111	“ 1 “
p	112	“ 2 “
q	113	“ 3 “
r	114	“ 4 “
s	115	“ 5 “
t	116	“ 6 “
u	117	“ 7 “
v	118	“ 8 “
0	48	Un utilisateur est dans la cabine et désire aller à l'étage 0
1	49	“ “ 1
2	50	“ “ 2
3	51	“ “ 3
4	52	“ “ 4
5	53	“ “ 5
6	54	“ “ 6
7	55	“ “ 7
8	56	“ “ 8
9	57	“ “ 9

Table 2 : Protocole des appels

#### 3.2. Recevoir et mémoriser un appel.

Lorsqu'un appel est fait par un utilisateur, l'appel est de suite enregistré dans un tableau.

```
break;
case 102: /* f */
    appel_descendre_etage[2] = 1;
    i = 2;
break;
case 103: /* g */
    appel_descendre_etage[3] = 1;
    i = 3;
break;
case 104: /* h */
    appel_descendre_etage[4] = 1;
    i = 4;
```

```

r0 0x00
r1 0x00
r2 0x00
r3 0x00
r4 0x00
r5 0x00
r6 0x00
Sys
a 0x00
b 0x00
}
sens = 1; //une fois tous les étages visualisés, on change
//permettre de regarder le tableau "descendre"
}
if(sens == 1){
    k=9;
    for(k==9;k>=0;k--){
        if((appel_descendre_etage[k]==1)|| (choix_int_etage[k]==1);
        trajetCabine());
    }
    sens = 1;
}
appel_descendre_etage [X:0x000004) = { [0]=0x0000,[1]=0x0000,[2]=0x0000,[3]=0x0001,[4]=0x0000,[5]=0x0000,[6]=0x0000,[7]=0x0000,[8]=0x0000,[9]=0x0000

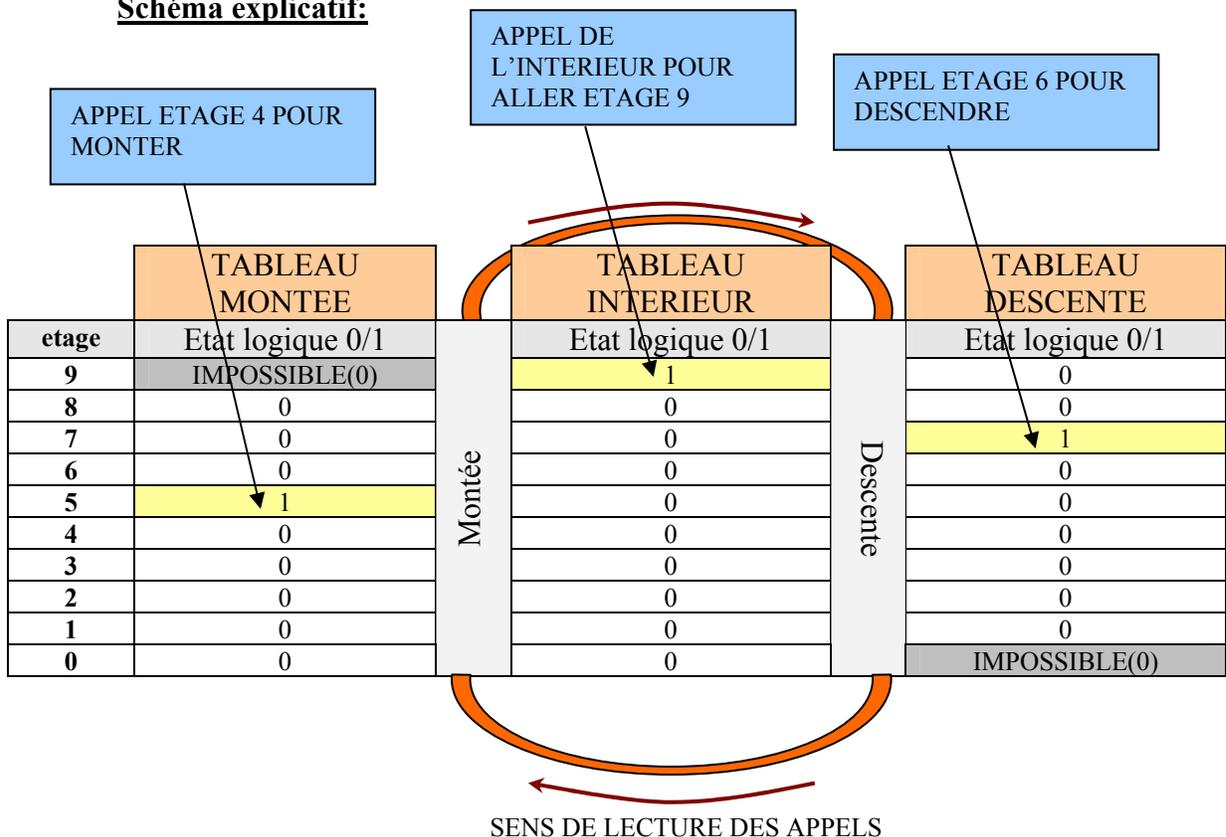
```

### 3.3. Comment ordonner les appels?

Une solution était d'utiliser des structures où l'étage suivant devenait l'étage actuel et l'appel suivant l'étage suivant mais cette solution était plus compliquée à mettre en oeuvre que de dresser trois tableaux, un pour les appels de montée, un pour les appels de descente et un pour les appels intérieurs, où chaque étage pouvaient être activé par une mise à l'état logique 1. Le problème était que ces tableaux demandaient plus d'espace mémoire, la RAM du microprocesseur n'étant que de 256 bytes. L'addition d'une mémoire externe a donc été nécessaire.

A chaque instant, ces tableaux permettent d'enregistrer les appels et de les ordonner pour la lecture avant ou pendant le trajet de l'ascenseur.

**Schéma explicatif:**



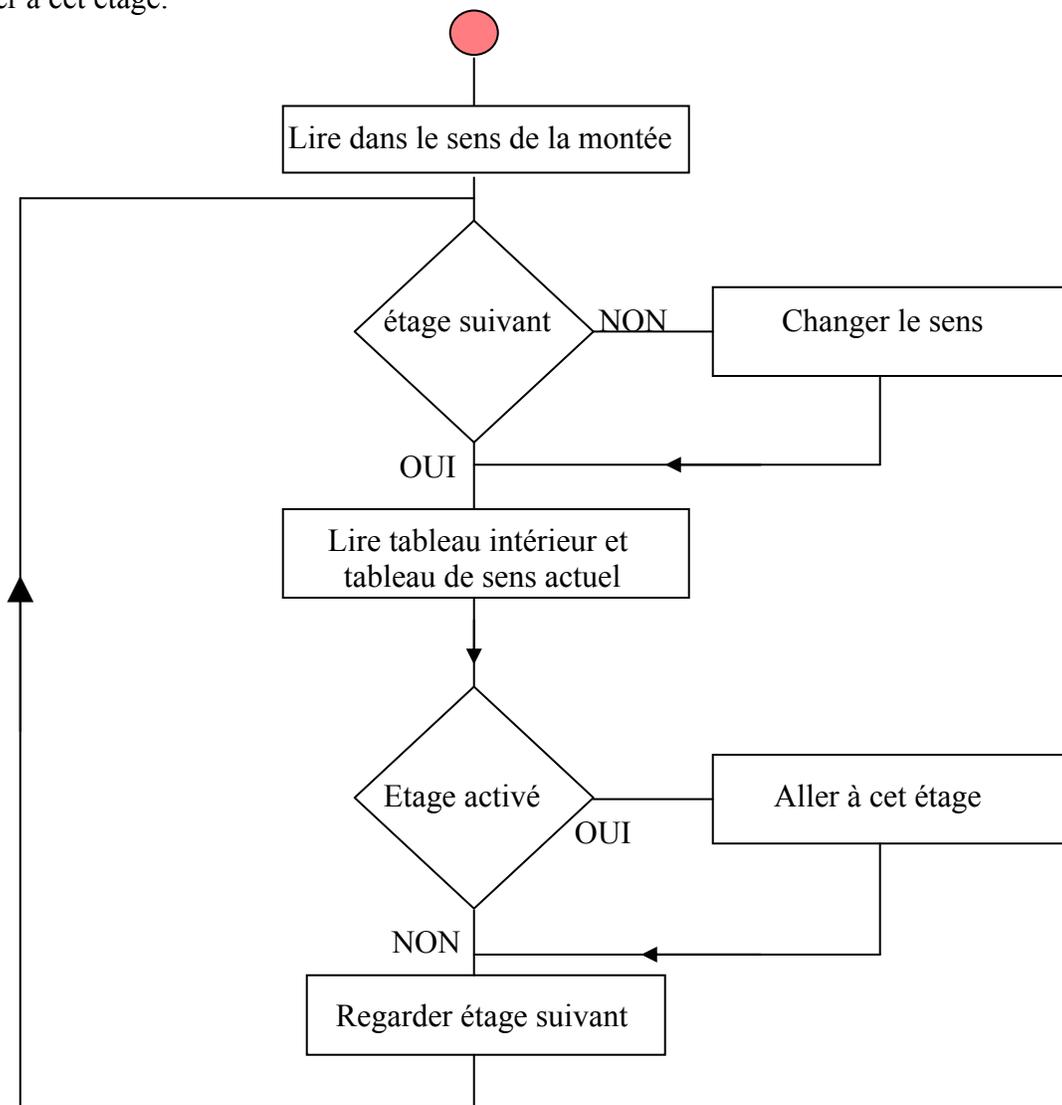
### 3.4. La lecture des appels.

La lecture des appels se commence toujours par la lecture du tableau “montée”, l’ascenseur se trouvant toujours initialement à l’étage 0.

Elle se déroule de la manière suivante (voir fluxogramme) :

Les tableaux montée et intérieur sont lus simultanément, si aucun des deux n’est activé pour cet étage, on regarde l’étage supérieur, et ainsi de suite... Une fois arrivé à l’étage 9, c’est le tableau de descente qui est à son tour lu, toujours simultanément avec le tableau intérieur, jusqu’à l’étage 0, et ainsi de suite...

L’activation d’un étage, pour n’importe quel tableau, signifie que l’ascenseur doit aller à cet étage.



**Figure 1:** fluxogramme de lecture des appels

### 3.5. Sens de déplacement.

Le sens de déplacement se déduit des tableaux à l'exception des déplacements engendrés par un changement de tableau.

Exemple : si l'ascenseur est à l'étage 7 et il n'y a aucun étage demandé aux étages suivants suivant le sens de déplacement de la cabine. La lecture du tableau opposé va avoir lieu et si un étage est trouvé et est supérieur à l'étage 7, le sens n'est pas modifié. C'est ce que j'ai appelé un mouvement inverse, c'est un déplacement de sens opposé au sens du tableau.

### 3.6. Calcul des capteurs de position.

La lecture des tableaux sert à calculer le nombre de capteurs de position à activer pour aller à l'étage choisi. Dès ce calcul effectué, la cabine se déplace et compte le nombre de capteurs activés. Une fois ce nombre de capteurs égal à celui calculé, une nouvelle recherche de l'étage suivant est commandée. Cette recherche commencera de l'étage actuel aux étages supérieurs si le sens est "monter", aux étages inférieurs si le sens est "descendre".

### 3.7. Comment prendre en compte les appels effectués durant le trajet et s'arrêter?

Sachant que monter d'un étage peut prendre environ 3 secondes, si l'ascenseur se déplace de l'étage 0 à l'étage 9, des utilisateurs ont le temps d'effectuer des appels aux étages intermédiaires et l'ascenseur doit répondre à ces demandes le plus vite possible.

C'est pourquoi, à chaque capteur de position pouvant entraîner le freinage de la cabine, l'ascenseur regarde dans le tableau si l'étage supérieur, ou inférieur suivant le sens, est activé. Si c'est le cas, une procédure d'arrêt (mise en vitesse de freinage et demande d'arrêt au prochain capteur) est enclenchée. Si l'étage n'est pas activé, l'ascenseur continue sa progression.

### 3.8. L'accès des personnes à la cabine.

L'accès des personnes commence par l'ouverture des portes, dès que les portes sont ouvertes, on temporise pendant 5s et on ferme les portes. La détection de présence ne peut être effective que lorsque les portes sont en train de se fermer. Dès lors, une interruption a lieu et commande la réouverture des portes et une nouvelle temporisation. Il est donc indispensable de ne détecter aucune présence pendant la fermeture des portes pour pouvoir poursuivre le processus.

Un excès de poids engendre la même procédure excepté qu'un signal lumineux et un signal sonore sont émis, et que l'on utilise pas d'interruption.

### 3.9. L'attente d'un choix d'étage de l'utilisateur.

Quand un utilisateur entre dans la cabine, il doit choisir l'étage où il veut aller. L'ascenseur ne doit pourtant pas obligatoirement attendre que l'utilisateur fasse le choix d'étage avant de continuer à avancer.

Cette attente ne doit avoir lieu qu'à la condition qu'aucun autre appel pour les étages suivants (supérieurs si l'ascenseur monte, inférieurs s'il descend) n'a été fait.

Si un ou plusieurs appels ont été faits, l'attente n'est pas indispensable puisque l'utilisateur a au minimum le temps pour aller à l'étage suivant (avant le capteur pouvant entraîner le freinage) pour faire son choix d'étage.

L'ascenseur compte donc le nombre de choix d'étage fait de l'intérieur de la cabine, ainsi quand celui-ci est égal à 0 et que quelqu'un entre dans l'ascenseur, l'ascenseur attend qu'il choisisse l'étage.

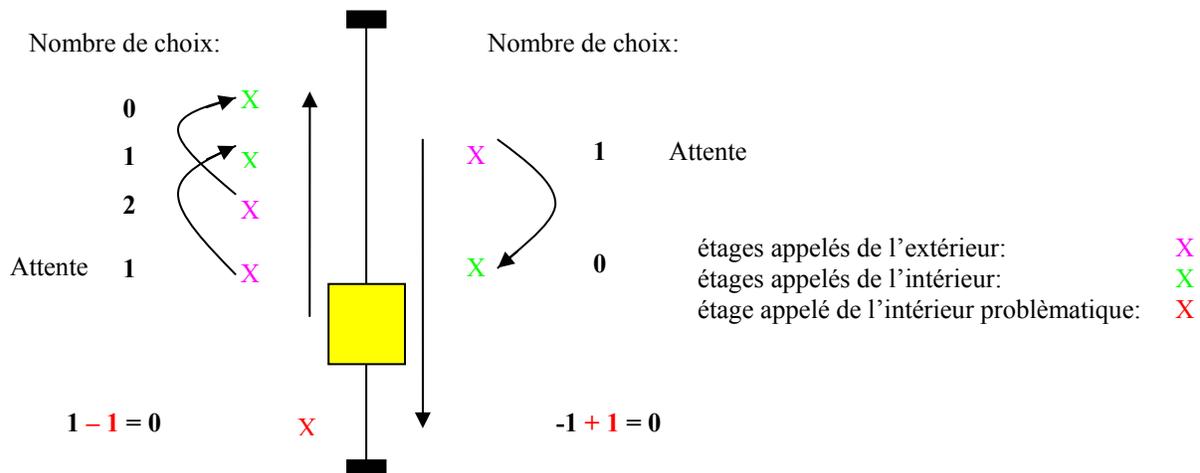
Attention, l'utilisateur ne possède que 10 secondes pour choisir l'étage. Au bout des 10 secondes, l'ascenseur prend automatiquement en compte l'appel suivant. Ceci pour éviter que l'ascenseur ne se bloque si quelqu'un entre et ressort pour tel ou tel raison.

Autre cas possible, un utilisateur entre et choisi un étage déjà dépassé par l'ascenseur.

Ce cas peut arriver quand un utilisateur entre dans l'ascenseur et s'aperçoit qu'il ne va pas dans le voulu, ou encore si quelqu'un appuie malencontreusement sur un bouton d'étage. Alors le nombre de choix d'étage doit être décrémenté pour ne pas perturber le fonctionnement normal de l'ascenseur, et ainsi ne permettre que les attentes de choix d'étage qui sont réellement nécessaires. Le nombre de choix d'étage sera réincrémenté quand l'ascenseur sera arrivé à l'étage demandé (l'ascenseur aura auparavant changé de sens).

#### Schéma explicatif:

Ex : L'ascenseur est en train de monter et un étage inférieur est demandé de l'intérieur de la cabine. Un choix a lieu lorsque l'utilisateur entre dans la cabine. L'incrémentation du nombre de choix a lieu lorsque l'utilisateur choisi un étage. La décrémentation a lieu lors de l'arrivée de l'utilisateur à l'étage demandé.



## Demonstration par le programme :

Exemple: Un utilisateur est au 2ème étage et désire aller au 7ème.

Au départ, aucun choix de l'intérieur de la cabine n'a été fait.

```

| 0x00 | case 55:/* 7 */
| 0x09 | i = 7;
|       | if(choix_int_etage[7] == 0){
|       |     choix_int_etage[7] = 1;
|       | }
| choix_int_etage [%:0x000022] = { [0]=0x0000,[1]=0x0000,[2]=0x0000,[3]=0x0000,[4]=0x0000,[5]=0x0000,[6]=0x0000,[7]=0x0000,[8]=0x0000,[9]=0x0000 }
| 0x00 | break;
| 0x21 | case 56:/* 8 */

```

On fait monter l'ascenseur au premier étage...

```

etage: 0
Un utilisateur est à l'etage 1 et désire monter
etage: 1
tempo 5s:
tempo terminée

```

On va attendre que l'utilisateur fasse le choix de l'étage où il désire aller puisqu'aucun choix n'a été fait auparavant et qu'il a fait un appel pour monter.

```

if((choix > 0) && (choix_int_etage[etage] == 1))
    choix = choix - 1;
//le nombre de choix faits est alors décrémenté arrivé à l'etage choix
if(choix == 0){ //aucun choix:
    t=0; //si il n'y a pas de choix d'etage avant 10 secondes,
    TR0=1; //le système prendra en compte l'appel suivant
    if(sens == 0)
        if(mouvement_inverse==0)
            while((choix == 0) && (appel_monter_etage[etage] == 1)
                && (t < 10000) && (choix_int_etage[etage] == 0))
                choix = choix + 1;
            if(mouvement_inverse==1)
                while((choix == 0) && (appel_descendre_etage[etage] == 1)
                    && (t < 10000));
    if(sens == 1)

```

Une fois le choix fait, on fait monter l'ascenseur au 7ème...

```

case 5b:/* 7 */
i = 7;
if(choix_int_etage[7] == 0){
    choix_int_etage[7] = 1;
    choix = choix + 1;
}
break;
case 5c:/* 8 */

```

Le nombre de choix de l'intérieur de la cabine est incrémenté.

Un utilisateur est dans la cabine et désire aller à l'étage 7

```

etage: 2
etage: 3
etage: 4
etage: 5
etage: 6
etage: 7
tempo 5s:

```

Le nombre de choix est décrémenté (choix = 0) mais comme le 7ème étage n'était pas demandé de l'extérieur, on n'attend pas de choix d'étage d'un utilisateur.

Register	Value
Regs	
r0	0x07
r1	0x00
r2	0x00
r3	0x00
r4	0x00
r5	0x00
r6	0x00
r7	0x00
r8	0x00
r9	0x00
r10	0x00
r11	0x00
r12	0x00
r13	0x00
r14	0x00
r15	0x00

```

if((choix > 0) && (choix_int_etage[etage] == 1))
    choix = choix - 1;
//le nombre de choix faits est alors décrémenté arrivé à l'étage choisi

if(choix == 0){ //aucun choix:
    t=0; //si il n'y a pas de choix d'étage avant 10 secondes.
    TR0=1; //le système prendra en compte l'appel suivant

    if(sens == 0)
        if(mouvement_inverse==0)
            while((choix == 0) && (appel_monter_etage[etage] == 1))
                ;
            //le nombre de choix faits est alors décrémenté arrivé à l'étage choisi
        }
        while((choix == 0) && (appel_descendre_etage[etage] == 1))
            ;
    }
    }

```

Comme il n'y a aucun autre appel, l'ascenseur prendra en compte le prochain appel enregistré.

### 3.10. Effacer les appels.

Les appels sont effacés suivants le sens dans lequel l'ascenseur se déplace. C'est-à-dire que si l'ascenseur monte, c'est l'étage (d'où l'ascenseur va partir) du tableau montée ainsi que du tableau des appels intérieurs qui vont être effacés. L'étage concerné du tableau des appels intérieurs est toujours effacé, quelque soit le sens de déplacement.

A partir où l'ascenseur s'arrête à un étage dans un sens, l'appel de cet étage du même sens est effacé ; même si l'appel a été fait trop tard pendant que l'ascenseur était en vitesse de freinage. C'est pourquoi l'effacement des messages se fait toujours juste après que les portes se soient fermées ou après le choix d'étage de l'utilisateur s'il y en a un.

Lors d'un mouvement inverse (cf paragraphe 2.5. Sens de déplacement), c'est l'étage du tableau de sens opposé au sens du dernier déplacement qui est effacé.

### 3.11. Ajouter de la mémoire externe.

Le microcontrôleur ne possédant que 256bytes de mémoire interne, j'ai eu besoin d'ajouter de la mémoire. Ceci entraîne des coûts supplémentaires mais améliore le rendement du microcontrôleur. Et comparé au coût globale d'un ascenseur avec toutes ses éléments mécanique, ce surcoût est moindre.

### 3.12. Les défaillances et la mise en route.

A chaque mise en route, l'ascenseur redescend à l'étage 0. Puis une vérification des capteurs peut avoir lieu si elle est demandé, cela peut permettre à la maintenance de vérifier après un changement de capteur si celui-ci fonctionne.

Quand l'ascenseur fonctionne, toutes les défaillances de capteurs peuvent être repérées par des temporisations ou par d'autres capteurs, ceci entraîne la descente de l'ascenseur à l'étage 0 et l'ouverture des portes. Si c'est un problème du capteur porte, l'ascenseur reste à l'étage où il est avec les portes ouvertes.

Quand une défaillance à lieu, un RESET est obligatoire pour redémarrer l'ascenseur. La demande de vérification des capteurs doit être faite avant la demande de mise en route pour qu'elle soit effective.

#### **4. Tests et validation.**

##### 4.1. Tableau.

<b>n°</b>	<b>TESTS</b>	<b>VALIDATION</b>
<b>INITIALISATION</b>		
1	Remettre l'ascenseur en position initiale	OUI – Si l'ascenseur ne se trouve pas à l'étage 0 à la mise en route du système, on le fait descendre à l'étage 0.
2	Vérifier le fonctionnement des capteurs de position de la cabine	OUI – A chaque mise en route, une vérification des capteurs de position se produit. Elle est obligatoire. Si un des capteurs est défaillant, un contrôle de la maintenance et un RESET sont nécessaires à la remise en route de l'ascenseur.
<b>LES APPELS</b>		
3	Recevoir les appels	OUI – Les appels sont reçus par le port série via un baud rate de 9615 baud. Des réception d'un code, le drapeau d'interruption de réception du port série s'active, ce qui permet de prendre en compte les appels en temps réel. Seules les interruptions des capteurs de position seront prioritaires car l'ascenseur doit savoir à tout moment à quel étage il se trouve. Comme le microcontrôleur travaille à une fréquence de 16MHz, il est impossible d'avoir deux appels simultanés. Ces appels sont de type binaire (8 bits), codés par le microcontrôleur externe. L'équivalent décimal lu par le programme indique l'étage de l'appel et son sens pour les appels de l'extérieur et l'étage désiré pour les appels de l'intérieur.
4	Simuler les appels	OUI – On simule les appels grâce au clavier, il faut donc faire correspondre les touches disponibles du clavier avec leurs valeurs décimales suivant le code ASCII.
5	Mémoriser les appels	OUI – Les appels sont stockés dans les tableaux de montée et de descente sous forme binaire. Un état logique 1 dans un des tableaux signifie que l'ascenseur doit s'arrêter à l'étage demandé dans le sens demandé. Les appels de l'intérieur de la cabine sont stockés dans un troisième tableau qui enregistre (sous forme binaire également) les étages demandés.
6	Attendre un appel	OUI – L'ascenseur peut avoir besoin d'attendre le choix d'un étage de l'intérieur de la cabine si il n'y a eu aucun étage choisi auparavant. Une constante est alors nécessaire pour savoir s'il faut attendre ou non, cette constante enregistre le nombre d'étages choisis (activés), si cette constante est nulle et qu'un appel avait eu lieu à cet étage, l'ascenseur attend un choix de l'utilisateur.
<b>ACCES A LA CABINE</b>		
7	Temporiser 5 secondes	OUI – 5000 interruptions du timer pour 1 interruption =

		1ms.
8	Ouvrir/fermer les portes d'accès à la cabine	OUI – L'état logique du capteur des portes permet savoir si les portes sont ouvertes ou fermées.
9	Identifier une présence	OUI – Utilisation de l'interruption INT1 commandant la réouverture des portes, l'attente de la fin de la détection et la refermeture des portes.
10	Signaler un excès de poids	OUI – Un excès de poids est signalé par l'état logique 1 du capteur de poids et permet l'activation des signaux d'alarmes, la réouverture des portes, l'attente d'un poids toléré, la désactivation des signaux d'alarme puis la refermeture des portes.
<b>PENDANT LE TRAJET</b>		
11	Savoir à chaque instant où se trouve l'ascenseur et afficher l'étage.	OUI – Grâce aux interruptions des capteurs de position qui permettent le comptage des capteurs et ainsi le comptage des étages, connaissant à chaque trajet l'étage de départ et l'étage d'arrivée.
12	Monter/descendre à un étage voulu	OUI – L'ascenseur sait à quel étage il se trouve, il regarde grâce aux tableaux de montée, de descente ou de choix intérieur quel est l'étage suivant demandé suivant le sens actuel.
13	Changer de sens	OUI – Si l'ascenseur voit que dans le tableau de sens actuel aucun étage supérieur (montée) ou inférieur (descente) est demandé, et qu'aucun choix de l'intérieur pour aller au delà de l'étage actuel (suivant le sens) n'a été fait, le sens est alors changé et le tableau de sens opposé est visualisé.
14	Prendre en compte les appels de tous les étages (mouvement inverse)	OUI – Il se peut que l'ascenseur ai besoin d'aller dans le sens opposé à celui exigé par le tableau. (ex : le système visualisait dans le tableau de sens montée, aucun étages supérieurs demandés, il regarde alors dans le tableau de sens descente et voit que le premier étage demandé est supérieur à l'étage actuel) L'ascenseur doit donc effectuer un nouveau changement de sens; qui n'est pas exactement un changement de sens puisqu' aucune vitesse n'était activée... L'ascenseur continuera donc à se déplacer dans le même sens que précédemment et effectuera le réel changement de sens à l'étage suivant.
15	Minimiser le nombre de changement de sens	OUI – Cette minimisation se fait grâce aux trois tableaux qui permettent d'activer les étages demandés en temps réel.
16	Prendre en compte les étages intermédiaires	OUI – Pendant le trajet, des arrêts aux étages intermédiaires à l'étage précédent et l'étage suivant peuvent être demandés. Le système prend en compte ces demandes en regardant à chaque passage devant un capteur pouvant entraîner un freinage et l'arrêt de la cabine au capteur suivant, si l'étage suivant est demandé. Si c'est le cas et que le sens demandé est le même que la cabine, la procédure d'arrêt à l'étage suivant est enclenchée.
<b>LES DEFAILLANCES DU SYSTEME</b>		
17	Repérer une défaillance	OUI – Toutes les défaillances de capteurs, exceptées celles des capteurs de présence et de poids, peuvent être repérées par le programme. Le programme peut effectuer à chaque mise en route une vérification des

		capteurs de position en se déplaçant du capteur de sol à l'étage 0 jusqu'au capteur du plafond pour vérifier si le nombre de capteurs ayant été activés est bien celui espéré. De plus, les défaillances des autres capteurs (capteur sol, capteur plafond et capteur portes) peuvent être repérées grâce à des temporisations qui considèrent au bout d'un certain temps de non réponse du capteur que celui-ci est défaillant. Une défaillance du moteur contrôlant les portes peut aussi être détectée grâce à ces temporisations. Par contre, une défaillance du moteur principal ne peut être repérée, un système extérieur doit être prévu pour pallier à ces défaillances et ainsi garantir la sécurité des usagers.
18	Arrêter le système lors d'une défaillance.	OUI – Si une défaillance se produit, un signal de défaillance est produit. Si c'est un problème dans le fonctionnement des portes, les portes se ouvrent pour laisser les gens sortir. L'ascenseur reste alors bloqué à cet étage. Si c'est un problème du moteur principal, la mécanique du moteur doit pouvoir stopper la cabine.
19	Relancer le système	OUI – Il existe une entrée RESET sur le microcontrôleur. Lorsque qu'une défaillance a lieu, le système est stoppé et attend que la maintenance règle le problème. Le système ne peut se réinitialiser et fonctionner qu'après un RESET et une mise en route.

## 4.2. Test final.

Ce test prend en compte toutes les situations critiques rencontrées par l'ascenseur lors de son fonctionnement. L'ascenseur se trouve initialement à l'étage 0.

**Table 1**

0	1	2	3	4	5	6	7	8	9	
2 M 4	0	1 3 8	4 M	0 7 M	4 6	5 M	6 7 M	1 8	7 D	3
2 D					6 9 D					
<b>Ordre des arrêts prévu:</b> 0, 2, 4, 6, 7, 8, 9, 7, 4, 3, 2, 1, 0, 3, 4, 5, 6, 1										

X M signifie: un utilisateur est à l'étage X et désire monter;

X D signifie: un utilisateur est à l'étage X et désire descendre;

X signifie: un utilisateur est à l'intérieur de la cabine et veut aller à l'étage X;

**en rouge:** premier appel (cabine immobile étage 0);

**en vert:** appels effectués lors de la montée;

**en bleu:** appels effectués lors de la descente;

**en mauve:** appels effectués lors de la 2ème montée;

.....: capteurs de position pouvant entraîner le freinage et l'arrêt à l'étage supérieur lors de la montée;

.....: capteurs de position pouvant entraîner le freinage et l'arrêt à l'étage inférieur lors de la descente;

\_\_\_\_\_ : capteurs de position pouvant entraîner l'arrêt de la cabine;

Ce test simule les situations suivantes:

- 21 appels;
- appels pour monter et pour descendre;
- appels couvrants les 10 étages;
- appels faits de l'extérieur;
- appels faits de l'intérieur;
- appels intermédiaires;
- appels pour descendre lors de la montée;
- appels pour monter lors de la descente;
- 2 appels de l'extérieur pour le même étage et le même sens;
- 2 appels de l'extérieur pour le même étage mais de sens opposés;
- 2 appels de l'intérieur pour le même étage;
- 1 appel d'étage ne correspondant pas au sens de l'appel extérieur;
- 1 appel d'étage fait trop tard;
- 1 appel d'ascenseur fait trop tard;
- 1 appel d'ascenseur pour l'étage 6 fait trop tard avec arrêt à cet étage dû à un appel fait de l'intérieur;
- 1 mouvement inverse dans chaque sens.

On peut voir sur le justificatif de validation (en annexe) du test que l'ordre des arrêts a été respecté.

# CONCLUSION

## 1. Bilan.

Le programme issu de ces deux mois et demi de stage fonctionne, mais l'architecture du programme n'est pas comme il le faudrait. Les interruptions n'ont pas toutes été utilisées correctement car le programme étant quasiment en fonctionnement, des modifications auraient été osées.

Maintenant que je sais comment se servir de ces interruptions, si j'ai l'occasion de reprogrammer un microcontrôleur, je saurai m'y prendre à l'avance pour créer des fonctions gérant totalement le programme en interruptions.

Ce type de programmation n'est pas si compliqué mais demande de l'expérience.

## 2. Les Apports.

Ce stage m'a apporté beaucoup sur le plan des connaissances en programmation de microcontrôleur.

Notre tuteur voulant nous faire travailler comme des ingénieurs, je réalise mieux maintenant le dynamisme intellectuel et les masses de connaissances, ainsi que le travail que demande le métier d'ingénieur.

Pour finir, ce stage Erasmus m'a surtout donné un regard extérieur sur le monde, le système, ou encore sur la France et m'a ouvert l'esprit sur les différences Nord-Sud par exemple. Il m'a aussi appris à relativiser. Un stage à l'étranger apporte énormément de choses...

## 3. Ouverture.

Malgré que ce stage ai été très intéressant, je ne pense pas me destiner à de longues études scientifiques mais plutôt à des études dans l'Audiovisuel, puisque c'est ce que je veux faire depuis quelques années déjà. Mais les métiers de l'Audiovisuel demandent aussi de l'initiative, un dynamisme intellectuel et des masses de connaissances... Ce sont les qualités qui mènent au professionnalisme et je pense les avoir tout au moins entrevues grâce à ce stage.

## ANNEXES

- Documentation AT89C52
- Justificatif de validation du test final
- Les fluxogrammes
- Programme
- Cahier des charges en portugais

## BIBLIOGRAPHIE

**Introduction au langage C (DUT GEII), Agnès Priou**

<http://www.keil.com>  
<http://perso.wanadoo.fr/michel.hubin/capteurs>  
<http://www.key2cad.com>  
<http://stielec.ac-aix-marseille.fr/cours/hu/detecteurs.htm>  
[http://www.cyber.uhp-nancy.fr/demos/MAIN-007/chap\\_deux/capteurs.html](http://www.cyber.uhp-nancy.fr/demos/MAIN-007/chap_deux/capteurs.html)  
<http://membres.lycos.fr/rosemarie/Micro.htm>  
<http://www.optiminfo.com/cartesm.htm>

