

Nassredine Amiche  
Yass Bak  
Tony Ren  
Antoine Bourhis  
Christophe Ye

## **Voiture autonome:** **Livrable intermédiaire**

### **Cahier des charges :**

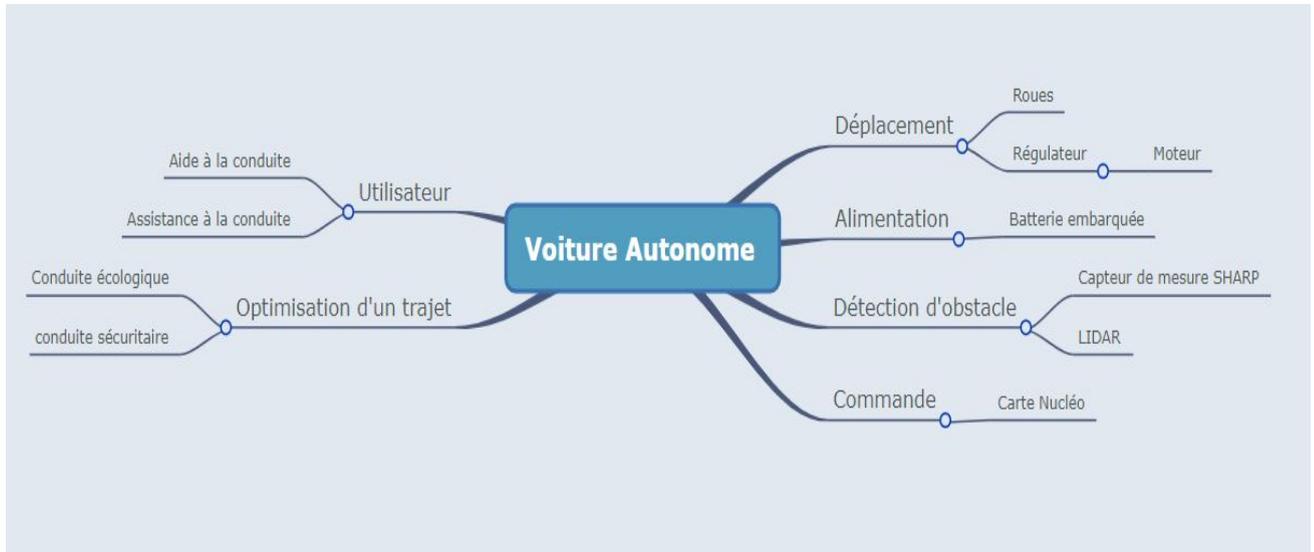
- Voiture qui roule de façon autonome
- Détection d'obstacles
- Esquiver de manière autonome les obstacles sur sa route.
- Arrêt de la voiture lorsque l'obstacle est inévitable.
- (Départ en arrière automatique lorsque la voiture est passée à l'arrêt à cause d'un obstacle inévitable. Seulement si on a le temps)

### **Planning des séances à suivre:**

Séance	Groupe 1	Groupe 2
Séance N°3 (1ère partie)	programmation de l'arrêt de la voiture.	Récupération de l'ancien code et adaptation du programme suite aux essais de la séance n°2.
Séance N°3 (2nde partie)	Détermination des angles optimaux de détection des objets.	Détermination de la vitesse optimale pour éviter les accidents du temps et de l'angle nécessaires pour faire tourner la voiture.
Séance 4	Compréhension du capteur Lidar et des valeurs caractéristiques d'angles et des distances	Adaptation du code pour le capteur Lidar
Séance 5	TESTS	TESTS
Séance n°6	FILM + optimisation de la voiture	Poster + optimisation de la voiture
Séance N°7	Préparation de l'oral	Préparation de l'oral

Séance n°8	Signature chez Tesla	Signature chez Général motors
------------	----------------------	-------------------------------

**Carte conceptuelle :**



**Schéma bloc:**

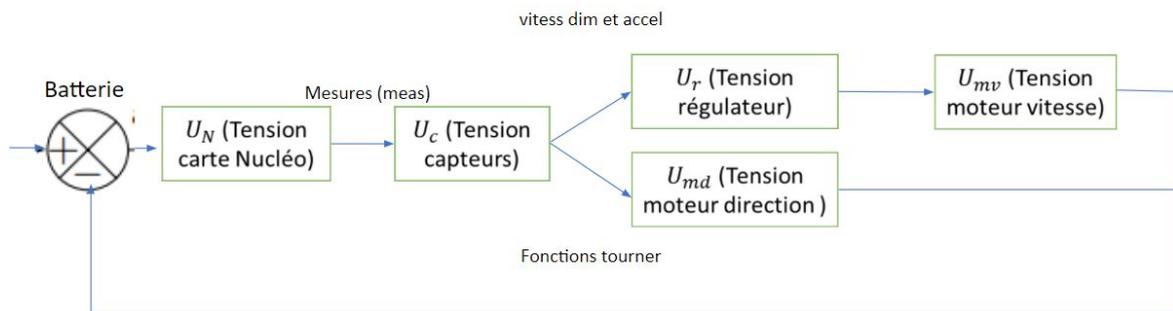


Schéma Bloc : Voiture autonome

**Code + explication:**

On récupère les tensions des capteurs qui nous indiquent la distance à laquelle se trouve l'obstacle. Si l'obstacle est à une distance inférieure ou égale à 40 cm que ce soit à gauche, à droite ou même à l'avant; on se tourne vers la partie libre, c'est-à-dire à gauche ou à droite, et afin de ne pas appliquer plus d'une fois la fonction "tournerg" ou "tournerd", car elle met la voiture à la position initial avant de commencer à tourner de façon continue, on ajoute un compteur qui va se mettre à un dès qu'on applique une de ces fonctions puis quand il repasse dans la grande boucle while(1) il continuera à tourner dans le bon sens.

```
#include "mbed.h"
PwmOut ma_sortie_pwm(D13);
Serial pc(USBTX, USBRX);
AnalogIn analog_in(A0); // L'entrée du premier capteur (celui à l'avant)
AnalogIn analogd_in(A1); // L'entrée du deuxième capteur (celui à droite)
AnalogIn analogg_in(A2); // L'entrée du troisième capteur (celui à gauche)
PwmOut vitess(D5);

void vitessdim(); // la fonction qui permet de diminuer la vitesse
void tournerd(); // la fonction qui permet de tourner à droite
void tournerg(); // la fonction qui permet de tourner à gauche
void accel(); // la fonction qui permet d'accélérer la vitesse

int main()
{
    int p=0;
    int d=0;
    int g=0;
    int pg=0;
    int pd=0;
    double meas;
    double measd;
    double measg;
    ma_sortie_pwm.period_ms(20);
    vitess.period_ms(20);
    vitess.pulsewidth_us(1.59*1000);
    while (1){

        meas = analog_in.read(); // La valeurs du capteur à l'avant
        measd = analogd_in.read(); // La valeurs du capteur à droite
        measg = analogg_in.read(); // La valeurs du capteur à gauche
        pc.printf("Capteur: %lf \n", meas);
        if (measd>0.23 || measg>0.23 || meas>0.3) // S' il détecte un mouvement à l'avant à
gauche ou à droite
        {
            if (measd>0.23 && measg<0.23 && meas<0.3 ) // S' il détecte un mouvement à
droite

        {
```

if (d<1) // on ajoute à chaque fois cette condition qui revient à demander si c'est le premier passage après avoir détecté un objet ou non pour qu'il ne repasse pas par la fonction tournerg car sinon elle remet la voiture à rouler vers le sens directe avant de tourner

```
{
    tournerg();
    d=d+1;
}
else
{ ma_sortie_pwm.pulsewidth_us(0.91*1000) // on continue à tourner à gauche}
}
if (measg>0.23 && measd<0.23 && meas<0.3 ) // S' il détecte un mouvement à gauche

{
    if (g<1) {
        tournerd();
        g=g+1;
    }
    else
    {ma_sortie_pwm.pulsewidth_us(1.65*1000) // on continue à tourner à droite}
}
if (measg>0.23 && measd>0.23 && meas<0.3 ){
ma_sortie_pwm.pulsewidth_us(1.25*1000);
vitess.pulsewidth_us(1.59*1000);
}
if (meas>0.3 && measg<0.23 && measd<0.23 ) // S' il détecte un mouvement devant
{
    if (p<1) {
        tournerd();
        p=p+1;
    }
    else
    {ma_sortie_pwm.pulsewidth_us(1.65*1000) // on continue à tourner à droite}
}
if (meas>0.3 && measg>0.23 && measd<0.23 ) // S' il détecte un mouvement à gauche et devant

{
    if (pg<1) {
        tournerd();
        pg=pg+1;
    }
    else
    {ma_sortie_pwm.pulsewidth_us(1.65*1000) // on continue à tourner à droite}
}
if (meas>0.3 && measd>0.23 && measg<0.23 ) // S' il détecte un mouvement à droite et devant
{
    if (pd<1) {
```

```

    tournerg();
    pd=pd+1;
}
else
{ma_sortie_pwm.pulsewidth_us(0.91*1000) // on continue à tourner à gauche}
}
}

else {ma_sortie_pwm.pulsewidth_us(1.25*1000);
vitesse.pulsewidth_us(1.59*1000);
g=0;
p=0;
d=0;
pg=0;
pd=0; // on remet ici tout les compteur à 0 pour qu'il puisse repasser par les
conditions
}

}
}
void accel()
{
int i;
for (i=0;i<4;i++)
{
    vitesse.pulsewidth_us((1.5+i/100)*1000);
    wait(0.01);
}
}

void tournerd()
{
int i;
vitesse.pulsewidth_us(1.57*1000); // Pour pouvoir vraiment faire ralentir le moteur on est
obligé de mettre le signal a 0... Lorsque que l'on va passer sur le moteur de la voiture, il
faudra sûrement adapter la période du moteur.
for (i=1;i<101;i++){
    ma_sortie_pwm.pulsewidth_us((1.25+0.40*i/100)*1000);
    wait(0.001);
}

}
void tournerg()
{
int i;

```

vitess.pulsewidth\_us(1.57\*1000); // Pour pouvoir vraiment faire ralentir le moteur on est obligé de mettre le signal a 0... Lorsque que l'on va passer sur le moteur de la voiture, il faudra sûrement adapter la période du moteur.

```
    for (i=1;i<101;i++){
        ma_sortie_pwm.pulsewidth_us((1.25-0.34*i/100)*1000);
        wait(0.001);
    }

}

void vitessdim()
{
    int i;
    for (i=0;i<4;i++)
    {
        vitess.pulsewidth_us((1.6-i/100)*1000);
        wait(0.01);
    }
}
```