



Premiers déplacements du robot RAPE

Les mouvements : avance jusqu'à bloquer sur un objet.

Mise en place d'un blocage sur l'avance avec les fins de course avant ou la cellule analogique.

Mise en place de paramètre sur le filtre de la cellule.

Mise en place de la Led bleue pour indiquer le blocage.

Table des matières

Premiers déplacements du robot RAPE	1
Introduction :	1
Rappel sur la cellule analogique	1
Filtrage de la cellule avant. Explications du fonctionnement du filtre :	2
Modifications pour rendre le programme plus facile à modifier :	2
Gestion de la led bleue.....	3
Modification du grafctet.....	3

Introduction :

Dans ce cours nous allons faire des mouvements rectilignes à notre robot, nous mettrons sur sa trajectoire un objet. La cellule ou le pare choc avant doivent arrêter le mouvement . La cellule sera filtrée par une tempo(voir le cours sur les tempos timer4 dans les bases) pour éviter les erreurs sur la lecture. La fonction LED elle sera modifiée.

Rappel sur la cellule analogique

cm	blanc lice	fond noir granuleux
abscisse	blanc	noir
40	100	100
30	194	194
25	215	220
20	276	261
15	334	360
10	503	470
5	644	644
3	595	601

600 (valeur utilisée dans le prog) = 5cm du pare choc avant jusqu'à l'obstacle.

**Programme** : test_cycle_roue_continu_P4_arret_obstacle

C'est le même programme que le programme précédent mais on ajoute la cellule centrale ainsi que les capteurs avant.

Filtrage de la cellule avant. Explications du fonctionnement du filtre :

```
//detection en marche avant et filtre de la cellule centrale

Détection de la cellule analogique sur avant robot qui déclenche la tempo

//la cellule détecte en dessous de 5cm une valeur supérieure à 600 => une tempo est lancée

Si la cellule revient à une valeur inférieure à 600 alors le start_tpo est remis à zéro sinon

start_tpo_filtre_det_av_ctrl = (capteur_central_avant_AN0 >= 600) ;

if (start_tpo_filtre_det_av_ctrl & fm_bit_10ms) {cpt_filtre_av++;}

si la tempo dépasse la valeur de consigne on monte le bit de blocage

if (cpt_filtre_av >= 5) {blocage_par_dectecteur_central=1 ; cpt_filtre_av=0;} //

si la valeur est inférieure à 600 on remet le compteur à 0

if ( !start_tpo_filtre_det_av_ctrl) {cpt_filtre_av=0;}

si la valeur est inférieure à 600 on remet le bit de blocage à 0 ;

if (capteur_central_avant_AN0 < 600) {blocage_par_dectecteur_central = 0;}

// blocage par cellule ou détecteur mécanique = détection d'un obstacle

if (fm_det_AV_ID || fm_blocage_det_cent) {blocage =1;}

//remis à zéro lors de l'enregistrement de la position
```

Test à réaliser : voir la vidéo

Modifications pour rendre le programme plus facile à modifier :

On voudrait avoir un paramètre pour la détection de la cellule ainsi qu'un paramètre sur la tempo cellule.

On ajoute le paramètre pour la cellule centrale: val_det_cellu_central

On ajoute le paramètre de la consigne pour la tempo :

Les deux valeurs seront fixées dans le début programme lors de la mise en service du pic.

Création des consignes en integer non signé 0→65535 .

```

//*****définition des mots et bits utilisés dans notre programme *****
int compt_cod_ga;
int compt_cod_dr;
int consigne_depla_dr;
int consigne_depla_ga;
int en_cours_de_mouvement;
unsigned long mem_opt_droit;
unsigned char copie_etapes_graf_cycle, etapes_graf_cycle;
int signed ecart_roue =0;
unsigned int val_det_cellu_central, cons_tpo_filtre_cell_central;
//*****

```



Consignes chargée avec une valeur

```
//consignes de réglage du robot
cpt_filtre_av = 0;
val_det_cellu_central= 600;// distance avant detection de la cellule centrale entre 0 et 80cm .600 =5cm
cons_tpo_filtre_cell_central =5 ;// temps en ms du filtrage de la cellule centrale
```

Modification du prog

```
//detection en marche avant et filtre de la cellule centrale
start_tpo_filtre_det_av_ctrl = (capteur_central_avant_ANO >= val_det_cellu_central) ;//génère un fm cde avance un pas à 'blocage
if (start_tpo_filtre_det_av_ctrl & fm_bit_10ms) { cpt_filtre_av++;}
if (cpt_filtre_av>= cons_tpo_filtre_cell_central ) {blocage_par_detecteur_central=1 ; cpt_filtre_av=0;} //
if ( !start_tpo_filtre_det_av_ctrl) {cpt_filtre_av=0;}
if (capteur_central_avant_ANO < val_det_cellu_central) (blocage_par_detecteur_central = 0;)
// blocage par cellule ou détecteur mécanique
if (fm_det_AV_ID || fm_blocage_det_cent) {blocage =1;}//remis à zéro lors de l'enregistrement de la position
.....
```

Gestion de la led bleue

```
sortie_led_bleue =(!det_AV_ID || (blocage_par_detecteur_central && fm_bit_100ms));
```

Dans cette ligne, on allume la led bleue si on a un blocage par détection centrale avec un clignotement et fixe si c'est un des détecteurs mécaniques du pare choc avant.

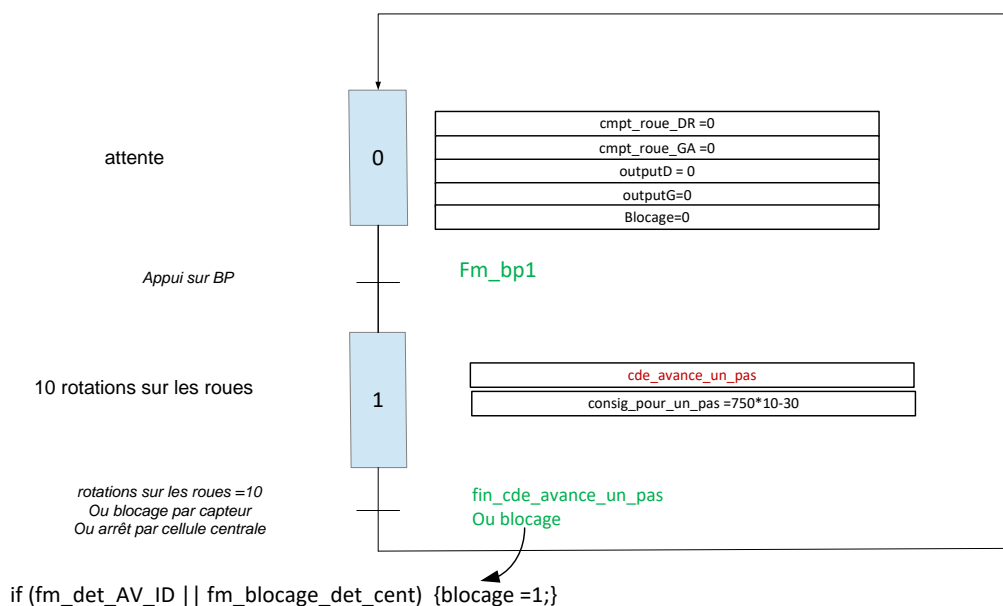
L'entrée étant toujours à 1, on met le détecteur inversé avec « ! » devant le détecteur.

Modification du grafctet

On a repris le grafctet du programme 3 en y ajoutant le blocage.

Test_cycle_roue avec arrêt sur obstacle

Prog 4





11/2023

Lorsque l'on appui sur le BP, le robot avance de 10 rotations de roue maximum. Si le robot rencontre un obstacle il s'arrête soit parce que le détecteur TOR mécanique indique un obstacle soit parce que la cellule à détecter un objet devant le robot.