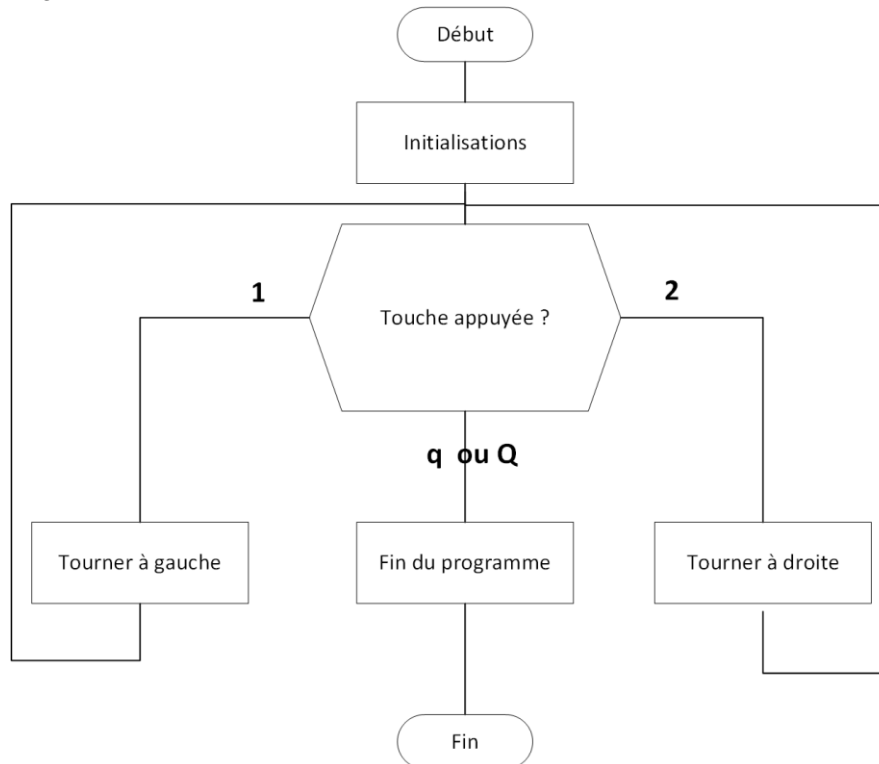


Commander un servomoteur à partir du clavier

Cahier des charges : Lorsque le programme est lancé, le servomoteur se positionne au centre de sa course. L'appui sur la touche 1 (gauche) et 2 (droite) déplace l'axe du servomoteur vers les limites relevées manuellement. La sortie du programme se fait avec les touches **q** ou **Q**. Le servomoteur est alors repositionné au centre.

Organigramme :



Ouvrez l'éditeur de texte nano :

```
pi@raspberrypi ~ $ nano cde_servo.sh
```

Saisissez le script destiné à commander la rotation du servomoteur à partir du clavier.

```
#!/bin/bash
# Script cde_servo.sh
# Commande d un servomoteur avec les touches du clavier
# Ce script utilise les commandes en ligne de wiring pi
#
# Au lancement du programme le servomoteur se positionne au milieu
# Les touches 1 et 2 font tourner le moteur jusqu à la butée
# 1 à gauche
# 2 à droite
# q ou Q pour quitter
```

```
# Effacer l'écran
clear

echo "Pour faire se déplacer l'axe du servomoteur"
echo "utilisez les touches 1 et 2 "
echo "Pour quitter le programme, utilisez la touche Q"

# Définir les limites du servomoteur
gauche_max=250
droite_max=60
milieu=150

# Position de départ du servomoteur
position=150

# Incrément lors de l appui sur touche
avance=5

# Définir la fonction gauche
gauche() {
    # Est ce que le moteur est en butée gauche ?
    if [ $position -lt $gauche_max ]
    then
        position=$(( $position + $avance ))
        gpio -g pwm 18 $position
    fi
}

# Définir la fonction droite
droite() {
    # Est ce que le moteur est en butée gauche ?
    if [ $position -gt $droite_max ]
    then
        position=$(( $position - $avance ))
        gpio -g pwm 18 $position
    fi
}

# Configuration du PWM
# Passer le GPIO 18 en sortie, mode PWM hardware
gpio -g mode 18 pwm

# Utiliser le mode PWM Mark/space
gpio pwm-ms

# Configurer pwmClock et pwm Range
gpio pwmc 192
gpio pwmr 2000

# Positionner le servo au milieu
gpio -g pwm 18 $position
```

```

# Boucle infinie de lecture des touches clavier
while :
do
    # La fonction read lit l entrée standard
    # -t 60 est un time out qui laisse une seconde pour appuyer sur une
    touche
    # -n 1 lit un caractère
    # -s ne renvoie pas le caractère reçu vers l'écran
    # La valeur lue est rangée dans la variable touche
    read -t 60 -n 1 -s touche

        case $touche in
            1)
                # Appel de la fonction gauche
                gauche
                ;;
            2)
                # Appel de la fonction droite
                droite
                ;;
            q)
                echo "Fin du script"
                # Mettre le moteur au milieu
                gpio -g pwm 18 $milieu
                exit 0
                ;;
            Q)
                clear
                echo "Fin du script"
                # Mettre le moteur au milieu
                gpio -g pwm 18 $milieu
                exit 0
                ;;
            *)
                echo "Choix incorrect"
                ;;
        esac
done

```