

# Finitions et nouvelles améliorations

La version du code de la GUI a évolué progressivement. *In fine*, la boucle `draw()` comprend le rafraîchissement des éléments graphiques, et l'analyse des différentes variables qui définissent des conditions mutuellement exclusives afin d'engager des actions par l'envoi de message :

```
void draw() {

    DrawPanelTitle("1 - USB setup", 4,4,632,120);
    // ...
    btn_serial_up.Draw(panelColor);
    // ...
    DrawPanelTitle("2 - Activate", 4,128,140,200);
    btn_serial_Home.Draw(panelColor);
    DisplayXZ(posX, posZ);
    DisplayEndStop(isUp);

    bkm_moving.Set(isMoving);
    bkm_moving.Draw();

    bkm_ready.Set(isReady);
    bkm_ready.Draw();

    // ... et enfin :

    if((isAuto) && (isReady)) {
        waitStart = currentTime();
        waitEnd = waitStart + pauseDuration;
        isWaiting = (currentTime() < waitEnd );
        isReady = false;
    }
    if(isWaiting) {
        isWaiting = (currentTime() < waitEnd );
    }
}
```

```

if((isAuto) && (posNum < Xtargets.length) && !isMoving && !isWaiting && !isUp) {
    serial_port.write("IZ\n");
}
if((isAuto) && (posNum < Xtargets.length) && !isMoving && !isWaiting && isUp) {
    String cmdToSend = "GX " + str(Xtargets[posNum]) + "\n";
    // println(cmdToSend);
    serial_port.write(cmdToSend);
    posNum++;
}
}

```

Autrement dit, j'ai (encore) écrit une machine à états, sans le dire. Si j'avais été rationnel, j'aurais pu gagner du temps et éviter de tâtonner : que ça serve de leçon ! ☐☐ (coucou les étudiants d'ingénierie chimique MU5CI803 ☐☐)

A ce stade, pour améliorer la robustesse, la maintenabilité et la lisibilité du code, il faudrait :

- écrire la table de vérité, tracer le GRAFCET, etc. ☐☐
- écrire et réduire la fonction d'état☐☐
- remplacer la série de `if()` et implémenter un `switch ... case()`

C'est indispensable de fiabiliser car actuellement les pauses et montées/descentes ne sont pas toujours dans le bon ordre (!?). En outre, il faut "durcir" la commande GX : vérifier que l'argument fourni est dans les bornes accessibles (c'est fait actuellement avec 70mm et 1000mm), et, au fur et à mesure de l'avancée, vérifier que l'on n'a pas dépassé la valeur ciblée (je ne comprends pas comment cela peut arriver, mais cela arrive).

Enfin, on peut/doit améliorer GZ pour éviter de descendre sans limite, s'il n'y a rien sous le capteur (ou enfin ajouter le deuxième fin de course ?)

Il reste quelques finitions matérielles à réaliser :

- ~~la butée de carotte et réflecteur pour le TOF X.~~
- ~~les supports des TOF~~
- ~~le support du MUX~~
- le boîtier

La modification du support du TOF Z semble diminuer la variabilité de la mesure, et permet d'éviter d'arriver dans la zone où la mesure devient trop incertaine. Toutefois, cela ne résoud pas le problème d'écart de mesure entre surface sombre et surface claire... La solution est peut-être une mesure de la réflectance ? Mais à quel moment ? Avec quel capteur ?