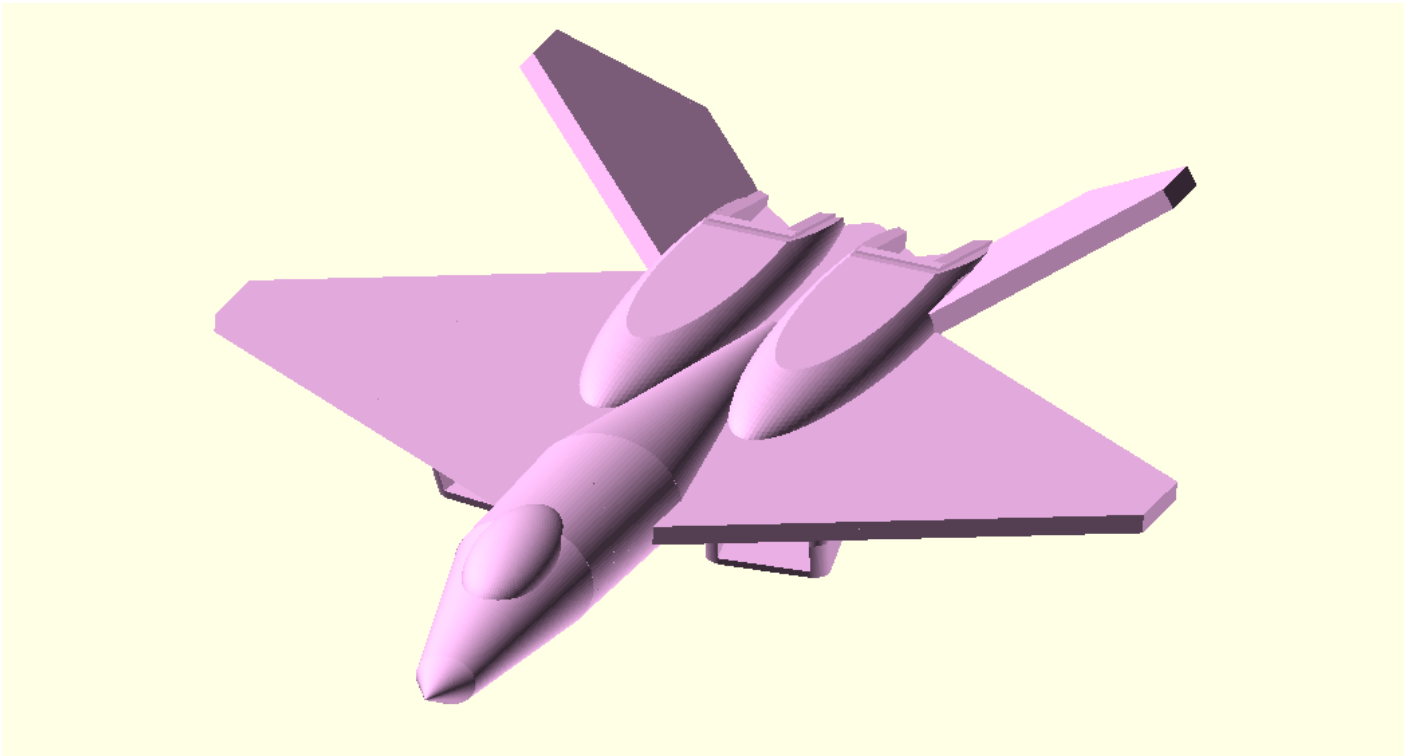


# Northrop YF-23

BRÉMOND Lison, L1 Cursus Master en Ingénierie, groupe A 2023-2024

*lison.bremond@etu.sorbonne-universite.fr*

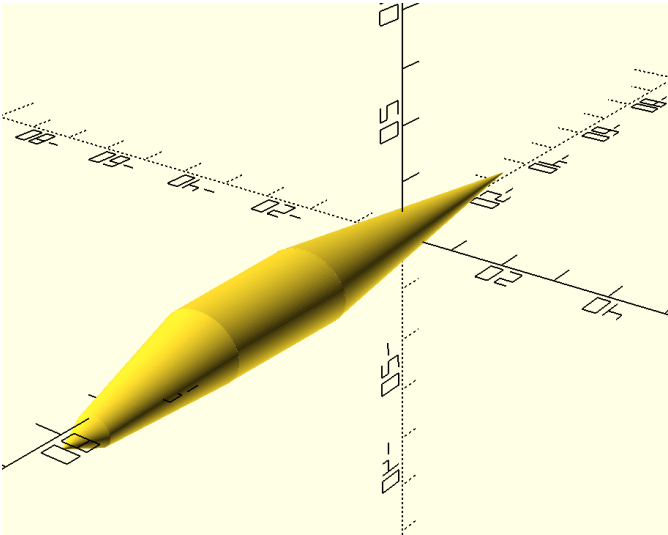
## Projet modélisation 3D YF-23



**Impression en PLA blanc - 4h - 48g avec supports + radeau**

### 1. Fuselage

Le fuselage de la modélisation se compose d'une association de cylindres et de cônes de tailles diverses, de telle sorte à créer le corps principal d'une base de positionnement des éléments à venir :



```
//cylindre avant
translate ([39,0,1]) scale ([1,1,0.8]) rotate ([0,90,0]) cylinder (25,8,7.5,center=true);
//pointe centrale
translate ([-2.5,0,1]) scale ([1.5,1,0.8]) rotate ([0,90,0]) cylinder (39,0.1,8,center=true);
module pointeavant() {
  difference () {
    //long cylindre avant
    translate ([83,0,-1]) scale ([1,1,0.8]) rotate ([0,100,0]) cylinder (25,7.73,3,center=true);
    translate ([67,0,6]) scale ([2,5,5]) cube (5,center=true);
    translate ([97,0,6]) scale ([1,5,5]) cube (5,center=true);
  };
};
translate ([-20.5,0,0.2]) pointeavant();
//pointe tête
translate ([77,0,-2.3]) scale ([1,1,0.8]) rotate ([0,90,0]) cylinder (6,3.2,0.1,center=true);
```

## 2. Ailes et dérive

Pour concevoir les "grandes ailes" nommées ainsi dans le codage, la première tentative fut de venir créer des polyhedrons manuellement en plaçant chaque point selon la forme des ailes. Or, ces formes ne passant pas en format STL pour la suite de l'impression, l'alternative plus longue était de venir placer une différence d'un assemblage de cubes modifiés avec "scale" venant sculpter un autre cube plat de la forme d'une aile. On modifie l'épaisseur de cette même forme avec "scale" en z.

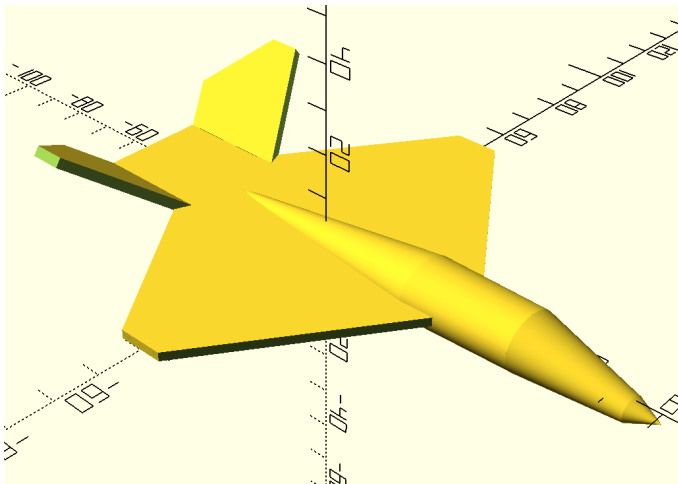
```
//grandes ailes
difference () {
```

```

translate ([0,0,1]) scale ([30,25,0.5]) cube (4,center=true);
translate ([52,30,0]) rotate ([0,0,35]) scale ([6,10,2]) cube (9,center=true);
translate ([52,-30,0]) rotate ([0,0,-35]) scale ([6,10,2]) cube (9,center=true);
translate ([-52,30,0]) rotate ([0,0,-35]) scale ([6,10,2]) cube (9,center=true);
translate ([-52,-30,0]) rotate ([0,0,35]) scale ([6,10,2]) cube (9,center=true);
translate ([0,-70,10]) scale ([6,10,5]) cube (4,center=true);
translate ([0,70,10]) scale ([6,10,5]) cube (4,center=true);
};

```

On procède de la même manière pour la dérive, avec cette fois-ci une inclinaison de  $35^\circ$  selon l'axe y et l'ajout d'une plateforme arrière venant lier les deux ailettes.

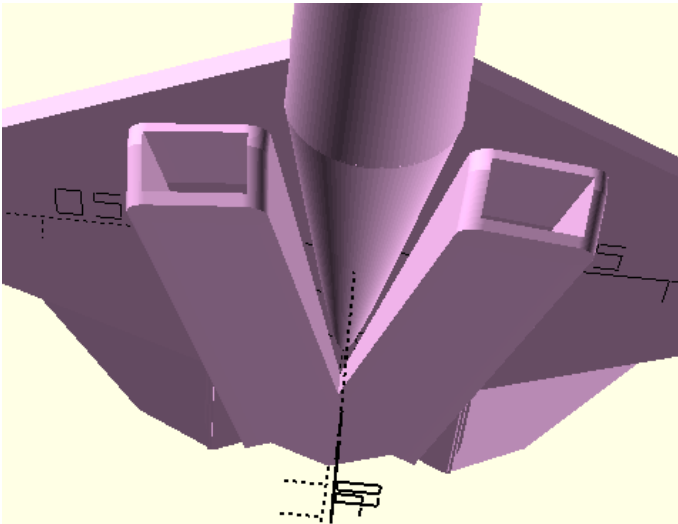


```

//plateforme arrière
translate ([-40,0,1]) scale ([1,1,0.06]) cube (27,0.1,6,center=true);
//dérive
module derive() {
  difference () {
    translate ([-43,25,0]) scale ([4,5,0.25]) cube (9,center=true);
    translate ([-8.5,35,0]) rotate ([0,0,27]) scale ([6,10,2]) cube (9,center=true);
    translate ([-74,35,0]) rotate ([0,0,-32]) scale ([4,7,2]) cube (9,center=true);
    translate ([-50,50,0]) scale ([4,2,2]) cube (9,center=true);
    translate ([-82,10,0]) rotate ([0,0,27]) scale ([6,7,2]) cube (9,center=true);
    translate ([-45,5,0]) scale ([7,2,2]) cube (9,center=true);
  }
};

```

### 3. Entrées d'air



Pour créer des entrées d'air, nous venons placer sous les "grandes ailes" deux parallélépipèdes rectangles aux bords arrondis avec la fonction "minkowski" symétriques et inclinés de  $2^\circ$  selon l'axe y (entrées 1 et 2 nommées dans le code), que l'on déplace sous un module global avec "translate" et "rotate" de  $5^\circ$  selon x et  $10^\circ$  selon y (il faut veiller à changer le signe des angles de rotation et des coordonnées en y pour la symétrie).

--> **Attention** : L'inclinaison des entrée d'air vers le bas implique que leur partie haute traverse le niveau des ailes et dépasse ainsi sur le plan supérieur de l'avion. Il est donc nécessaire de venir faire une différence de ces parties parasites avec une forme suffisamment grande (ici un cube de 8 unités de côté modifié avec "scale") qui ne sera plus visible après la manipulation.

Pour parvenir à creuser les entrées d'air, on procède de la même façon, avec cette fois une fonction "minkowski" comprenant des parallélépipèdes de dimensions 1.6 fois inférieures aux premiers. En créant une différence pour chacun des deux modules et en ajustant la hauteur de ces derniers selon z, nous venons former nos creux.

```
//entrée d'air
module entree1() {
  minkowski ()
  {
    translate ([0,-15,-3.2]) scale ([18,2.2,1.2]) rotate ([0,2,0]) cube (4,center=true);
    cylinder (r=2,h=2,center=true);
  };
};

module entree2() {
  minkowski ()
  {
    translate ([0,-15,-3.2]) scale ([18,2.2,1.2]) rotate ([0,2,0]) cube (2.5,center=true);
    cylinder (r=2,h=2,center=true);
  };
};
```

```

};

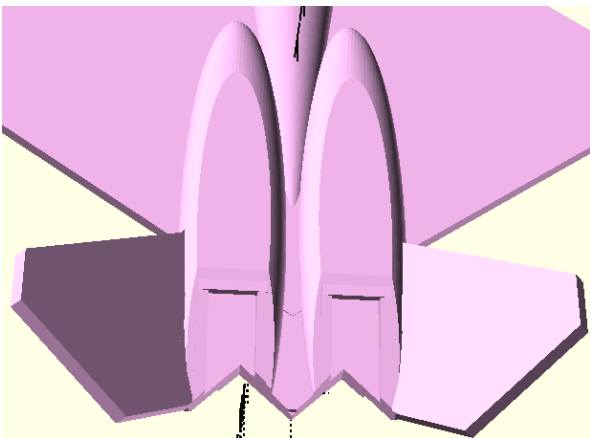
difference () {
translate ([-14,6,3]) rotate ([0,5,-10]) entree1();
translate ([2,3.2,1.7]) rotate ([0,5,-10]) entree2();
translate ([-17,0,6]) scale ([10,7,1.2]) cube (8,center=true);
};

difference () {
translate ([-19,23.5,3]) rotate ([0,5,10]) entree1();
translate ([-0,26.8,1.3]) rotate ([0,5,10]) entree2();
translate ([-17,0,6]) scale ([10,7,1.2]) cube (8,center=true);
};

```

## 4. Partie arrière

Pour toute la partie arrière, on vient ajouter deux sphères que l'on allonge de chaque côté de la pointe précédemment modélisée située au centre du plan supérieur de l'avion. Pour créer des faces plates, on modélise des cubes plats avec "scale" que l'on positionne et abaisse au niveau des deux sphères en créant une différence. Enfin, on modélise une nouvelle fois deux cubes aux extrémités des deux sphères, avec une fonction différence supplémentaire, de sorte à former des creux rectangulaires à l'arrière de l'appareil.



```

difference () {
translate ([-28,8,1.5]) scale ([6,1.2,0.9]) rotate ([260,0,0]) sphere (6,center=true);
translate ([-28,8,7]) scale ([9,1.7,0.5]) cube (6,center=true);
translate ([-60,0,0]) scale ([2,4.5,2]) cube (6,center=true);
translate ([-50,8,6]) rotate ([0,-10,0]) scale ([5,2.5,1]) cube (6,center=true);
//creux
translate ([-47,8,3]) scale ([3,0.8,0.5]) cube (8,center=true);
translate ([-35,0,-2]) scale ([4,4,0.5]) cube (10,center=true);
translate ([-20,0,-5]) scale ([8,4,1]) cube (10,center=true);
//retraits bordures arrières

```

```

translate ([-43,16,2]) scale ([7,1,1]) cube (4,center=true);
};
difference () {
translate ([-28,-8,1.5]) scale ([6,1.2,0.9]) rotate ([260,0,0]) sphere (6,center=true);
translate ([-28,-8,7]) scale ([9,1.7,0.5]) cube (6,center=true);
translate ([-60,0,0]) scale ([2,4.5,2]) cube (6,center=true);
translate ([-50,-8,6]) rotate ([0,-10,0]) scale ([5,2.5,1]) cube (6,center=true);
//creux
translate ([-47,-8,3]) scale ([3,0.8,0.5]) cube (8,center=true);
translate ([-35,0,-2]) scale ([4,4,0.5]) cube (10,center=true);
translate ([-20,0,-5]) scale ([8,4,1]) cube (10,center=true);

```

## 5. Derniers rajouts

Par la suite, nous ajoutons trois rebords rectangulaires sur chaque sphère avec scale, translate et rotate (voir capture d'écran précédente). Puis, nous venons créer un grand module "avion" englobant toutes les modélisations précédentes, et l'incluons dans une nouvelle différence avec deux cubes aplatis à 45° selon z, venant ainsi former les pics triangulaires sur la plateforme arrière formée plus tôt. L'objectif de ce module est de pouvoir sculpter la plateforme, les sphères arrières ainsi que les entrées d'air en une seule fois, en évitant la modélisation de trois fonctions différences par bloc et donc de considérablement allonger le code.

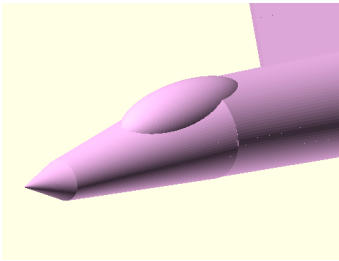
```

//rebords arrières
translate ([-48,11.8,3.2]) rotate ([0,-10,0]) scale ([4,0.3,0.2]) cube (5,center=true);
translate ([-48,-11.8,3.2]) rotate ([0,-10,0]) scale ([4,0.3,0.2]) cube (5,center=true);
translate ([-48,-4.1,3.2]) rotate ([0,-10,0]) scale ([4,0.28,0.2]) cube (5,center=true);
translate ([-48,4.1,3.2]) rotate ([0,-10,0]) scale ([4,0.28,0.2]) cube (5,center=true);
translate ([-38,8,5]) rotate ([0,0,90]) scale ([1.8,0.3,0.2]) cube (5,center=true);
translate ([-38,-8,5]) rotate ([0,0,90]) scale ([1.8,0.3,0.2]) cube (5,center=true);
};

difference () {
avion();
translate ([-63,8,1]) rotate ([0,0,45]) scale ([5,6,2]) cube (4,center=true);
translate ([-63,-8,1]) rotate ([0,0,45]) scale ([6,5,2]) cube (4,center=true);

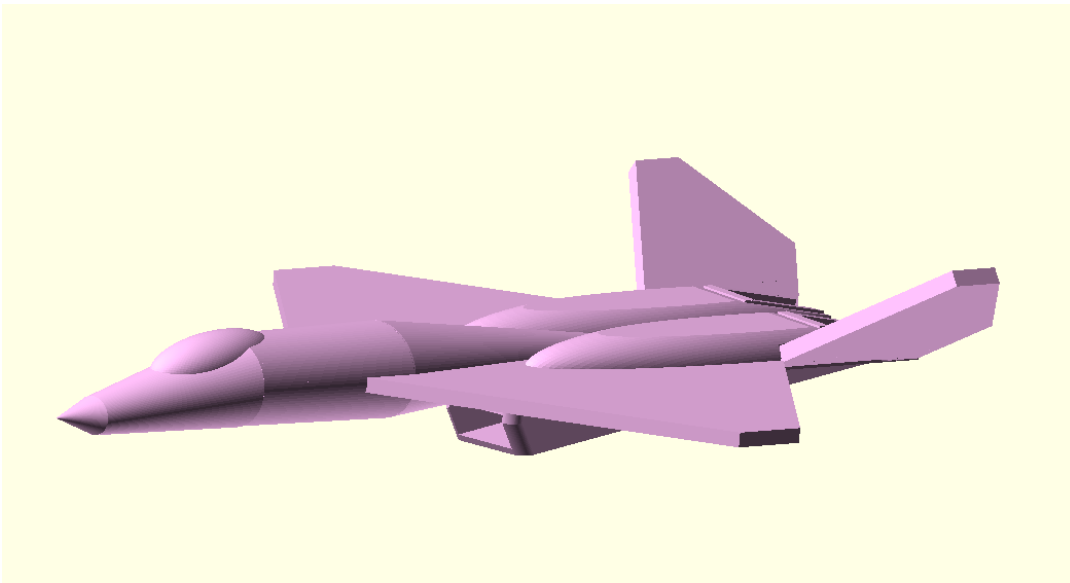
```

Enfin, nous plaçons le cockpit avec une sphère allongée légèrement inclinée :



```
//cockpit  
translate ([55,0,4]) rotate ([0,10,0]) scale ([3.2,1.5,1.2]) sphere (3,center=true);
```

## 6. Code complet



```
$fn=100;  
module total() {  
  module avion() {  
    //cylindre avant  
    translate ([39,0,1]) scale ([1,1,0.8]) rotate ([0,90,0]) cylinder (25,8,7.5,center=true);  
    //pointe centrale  
    translate ([-2.5,0,1]) scale ([1.5,1,0.8]) rotate ([0,90,0]) cylinder (39,0.1,8,center=true);  
    module pointeavant() {  
      difference () {  
        //long cylindre avant  
        translate ([83,0,-1]) scale ([1,1,0.8]) rotate ([0,100,0]) cylinder (25,7.73,3,center=true);  
        translate ([67,0,6]) scale ([2,5,5]) cube (5,center=true);  
        translate ([97,0,6]) scale ([1,5,5]) cube (5,center=true);  
      };  
    };  
    translate ([-20.5,0,0.2]) pointeavant();  
  }  
}
```

```

//pointe tête
translate ([77,0,-2.3]) scale ([1,1,0.8]) rotate ([0,90,0]) cylinder (6,3.2,0.1,center=true);

//plateforme arrière
translate ([-40,0,1]) scale ([1,1,0.06]) cube (27,0.1,6,center=true);
//cockpit
translate ([55,0,4]) rotate ([0,10,0]) scale ([3.2,1.5,1.2]) sphere (3,center=true);

//dérive
module derive() {
difference () {
translate ([-43,25,0]) scale ([4,5,0.25]) cube (9,center=true);
translate ([-8.5,35,0]) rotate ([0,0,27]) scale ([6,10,2]) cube (9,center=true);
translate ([-74,35,0]) rotate ([0,0,-32]) scale ([4,7,2]) cube (9,center=true);
translate ([-50,50,0]) scale ([4,2,2]) cube (9,center=true);
translate ([-82,10,0]) rotate ([0,0,27]) scale ([6,7,2]) cube (9,center=true);
translate ([-45,5,0]) scale ([7,2,2]) cube (9,center=true);
};
};
translate ([0.5,1,-7]) rotate ([35,0,0]) derive();
translate ([0.5,-1,-7]) rotate ([145,0,0]) derive();

//grandes ailes
difference () {
translate ([0,0,1]) scale ([30,25,0.5]) cube (4,center=true);
translate ([52,30,0]) rotate ([0,0,35]) scale ([6,10,2]) cube (9,center=true);
translate ([52,-30,0]) rotate ([0,0,-35]) scale ([6,10,2]) cube (9,center=true);
translate ([-52,30,0]) rotate ([0,0,-35]) scale ([6,10,2]) cube (9,center=true);
translate ([-52,-30,0]) rotate ([0,0,35]) scale ([6,10,2]) cube (9,center=true);
translate ([0,-70,10]) scale ([6,10,5]) cube (4,center=true);
translate ([0,70,10]) scale ([6,10,5]) cube (4,center=true);
};
//entrée d'air
module entree1() {
minkowski ()
{
translate ([0,-15,-3.2]) scale ([18,2.2,1.2]) rotate ([0,2,0]) cube (4,center=true);
cylinder (r=2,h=2,center=true);
};
};

```



```

module entree2() {
  minkowski ()
  {
    translate ([0,-15,-3.2]) scale ([18,2.2,1.2]) rotate ([0,2,0]) cube (2.5,center=true);
    cylinder (r=2,h=2,center=true);
  };
};

difference () {
  translate ([-14,6,3]) rotate ([0,5,-10]) entree1();
  translate ([2,3.2,1.7]) rotate ([0,5,-10]) entree2();
  translate ([-17,0,6]) scale ([10,7,1.2]) cube (8,center=true);
};

difference () {
  translate ([-19,23.5,3]) rotate ([0,5,10]) entree1();
  translate ([-0,26.8,1.3]) rotate ([0,5,10]) entree2();
  translate ([-17,0,6]) scale ([10,7,1.2]) cube (8,center=true);
};

//partie supérieure
difference () {
  translate ([-28,8,1.5]) scale ([6,1.2,0.9]) rotate ([260,0,0]) sphere (6,center=true);
  translate ([-28,8,7]) scale ([9,1.7,0.5]) cube (6,center=true);
  translate ([-60,0,0]) scale ([2,4.5,2]) cube (6,center=true);
  translate ([-50,8,6]) rotate ([0,-10,0]) scale ([5,2.5,1]) cube (6,center=true);
};

//creux
translate ([-47,8,3]) scale ([3,0.8,0.5]) cube (8,center=true);
translate ([-35,0,-2]) scale ([4,4,0.5]) cube (10,center=true);
translate ([-20,0,-5]) scale ([8,4,1]) cube (10,center=true);

//retraits bordures arrières
translate ([-43,16,2]) scale ([7,1,1]) cube (4,center=true);
};

difference () {
  translate ([-28,-8,1.5]) scale ([6,1.2,0.9]) rotate ([260,0,0]) sphere (6,center=true);
  translate ([-28,-8,7]) scale ([9,1.7,0.5]) cube (6,center=true);
  translate ([-60,0,0]) scale ([2,4.5,2]) cube (6,center=true);
  translate ([-50,-8,6]) rotate ([0,-10,0]) scale ([5,2.5,1]) cube (6,center=true);
};

//creux
translate ([-47,-8,3]) scale ([3,0.8,0.5]) cube (8,center=true);
translate ([-35,0,-2]) scale ([4,4,0.5]) cube (10,center=true);
translate ([-20,0,-5]) scale ([8,4,1]) cube (10,center=true);

//retraits bordures arrières

```

```

translate ([-43,-16,2]) scale ([7,1,1]) cube (4,center=true);
};
//rebords arrières
translate ([-48,11.8,3.2]) rotate ([0,-10,0]) scale ([4,0.3,0.2]) cube (5,center=true);
translate ([-48,-11.8,3.2]) rotate ([0,-10,0]) scale ([4,0.3,0.2]) cube (5,center=true);
translate ([-48,-4.1,3.2]) rotate ([0,-10,0]) scale ([4,0.28,0.2]) cube (5,center=true);
translate ([-48,4.1,3.2]) rotate ([0,-10,0]) scale ([4,0.28,0.2]) cube (5,center=true);
translate ([-38,8,5]) rotate ([0,0,90]) scale ([1.8,0.3,0.2]) cube (5,center=true);
translate ([-38,-8,5]) rotate ([0,0,90]) scale ([1.8,0.3,0.2]) cube (5,center=true);
};
difference () {
avion();
    translate ([-63,8,1]) rotate ([0,0,45]) scale ([5,6,2]) cube (4,center=true);
    translate ([-63,-8,1]) rotate ([0,0,45]) scale ([6,5,2]) cube (4,center=true);
};
};
color("#FABBF4") total();

```



*mars 2024*

