-lib mixte :

Arduino

Allegro

* Innit
* void drawPixel(size\_t coord\_x, size\_t coord\_y, COLOR) ;

screen.drawPixel(size\_t coord\_x, size\_t coord\_y, COLOR);

void [al\_draw\_pixel](https://www.allegro.cc/manual/al_draw_pixel)(float x, float y, [ALLEGRO\_COLOR](https://www.allegro.cc/manual/ALLEGRO_COLOR) COLOR);

* void drawLine(size\_t coord\_x1, size\_t coord\_y1, size\_t coord\_x2, size\_t coord\_y2, COLOR);

screen.drawLine(size\_t coord\_x1, size\_t coord\_y1, size\_t coord\_x2, size\_t coord\_y2, COLOR);

void al\_draw\_line(float x1, float y1, float x2, float y2, ALLEGRO\_COLOR color, float thickness = 1);

* void drawRect(size\_t coord\_x, size\_t coord\_y, size\_t w, size\_t h, COLOR);

screen.drawRect(size\_t coord\_x, size\_t coord\_y, size\_t w, size\_t h, COLOR);

void al\_draw\_rectangle(float x1, float y1, float x2, float y2, ALLEGRO\_COLOR color, float thickness = 1); (à tester si c’est entre 2 coords ou 1 coord et une taille)

* void fillRect(size\_t coord\_x, size\_t coord\_y, size\_t w, size\_t h, COLOR);

display.fillRect(size\_t coord\_x, size\_t coord\_y, size\_t w, size\_t h, COLOR);

void al\_draw\_filled\_rectangle(float x1, float y1, float x2, float y2, ALLEGRO\_COLOR COLOR);

* void drawTriangle(size\_t coord\_x1, size\_t coord\_y1, size\_t coord\_x2, size\_t coord\_y2, size\_t coord\_x3, size\_t coord\_y3, COLOR);

screen.drawTriangle(size\_t coord\_x1, size\_t coord\_y1, size\_t coord\_x2, size\_t coord\_y2, size\_t coord\_x3, size\_t coord\_y3, COLOR);

void al\_draw\_triangle(float x1, float y1, float x2, float y2, float x3, float y3, ALLEGRO\_COLOR color, float thickness = 1);

* void fillTriangle(size\_t coord\_x1, size\_t coord\_y1, size\_t coord\_x2, size\_t coord\_y2, size\_t coord\_x3, size\_t coord\_y3, COLOR);

screen.fillTriangle(size\_t coord\_x1, size\_t coord\_y1, size\_t coord\_x2, size\_t coord\_y2, size\_t coord\_x3, size\_t coord\_y3, COLOR);

void al\_draw\_filled\_triangle(float x1, float y1, float x2, float y2, float x3, float y3, ALLEGRO\_COLOR COLOR);

* void drawCircle(size\_t coord\_x, size\_t coord\_y, size\_t r, COLOR);

screen.drawCircle(size\_t coord\_x, size\_t coord\_y, size\_t r, COLOR);

void al\_draw\_circle(float x, float y, float r, ALLEGRO\_COLOR COLOR, float thickness = 1);

* void fillCircle(size\_t coord\_x, size\_t coord\_y, size\_t r, COLOR);

screen.fillCircle(size\_t coord\_x, size\_t coord\_y, size\_t r, COLOR);

void al\_draw\_filled\_circle(float x, float y, float r, ALLEGRO\_COLOR color);

* void drawChar(size\_t coord\_x, size\_t coord\_y, char c, size\_t txt\_size, COLOR)

screen.setTextSize(txt\_size);  
screen.setTextColor(COLOR);  
screen.setCursor(x ,y);  
screen.write(c);

al\_draw\_text(font a determiner), COLOR, x, y, 0, \*c)

* void drawString(size\_t coord\_x, size\_t coord\_y, char\* s, size\_t txt\_size, COLOR)

screen.setTextSize(txt\_size);  
screen.setTextColor(COLOR);  
screen.setCursor(x ,y);  
screen.print(s);

al\_draw\_text(font a determiner), COLOR, x, y, 0, s)

* void drawBitmap(size\_t x, size\_t y, uint32\_t bitmap[], w, h);

Refaire avec drawPixel()

Refaire avec drawPixel()

* void fillScreen(uint16\_t COLOR)

screen.fillScreen(uint16\_t color)

JE SAIS AP