-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);

Refaire à partir de drawPixel

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

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

Refaire à partir de drawLine

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

display.fillRect(size\_t coord\_x1, size\_t coord\_y1, size\_t coord\_x2, size\_t coord\_y2, COLOR);

Refaire à partir de drawRect

* 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);

Refaire à partir de drawLine

* 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);

Refaire à partir de drawTriangle

* 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);

Refaire à partir de drawPixel

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

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

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

* void drawBitmap(icons[f][XPOS], icons[f][YPOS], bitmap, w, h, WHITE);

screen.drawBitmap(int16\_t x, int16\_t y, uint8\_t \*bitmap, int16\_t w, i nt16\_t h, uint16\_t color),

void al\_draw\_bitmap(ALLEGRO\_BITMAP \*bitmap, x, float dy, 0)