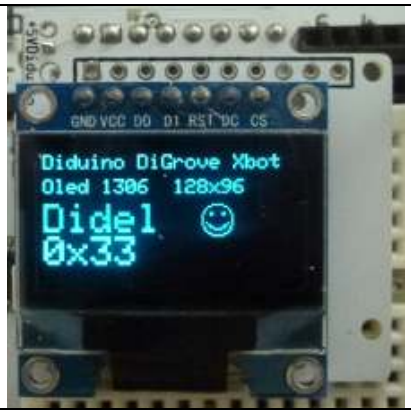
www.didel.com/xbot/Oled.pdf

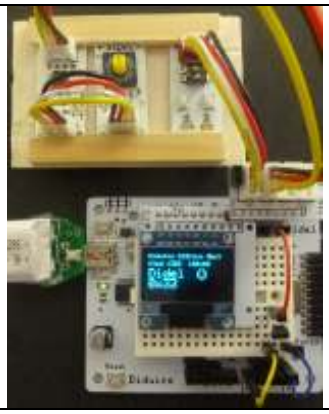
Didel and Boxtec support two Oled displays, not only nice, but essential for debugging robotic applications. They use the same minishield adapter (SSD 1331 at the end of this document). Compared to vendor's examples, pins have been redefined for compatibility between the two displays, and for leaving pin 13 and its led free for debugging.

Oled 128x64 blue SSD1306 on PORTB (pins 8-12)

French version www.didel.com/xbot/Oled1306.pdf

This Oled is available from ElecRow and Didel. Its resolution is 128x64 pixels, blue only. Adafruit, Seedstudio have similar devices in different packages, not compatible with Didel Minishield.
Power 5V, 40 mA.
3.5V 30 mA
Serial interface, 5 lines.

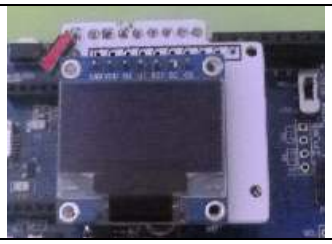




http://www.elecrow.com/display-c-77/lcds-oleds-c-77_88/096-oled-128x64-blue-p-751.html

Didel minishield on pins 8-12.
Diduino/Arduino compatible.
Kidule option (on pins 14-18)

SSD1306 / SSD1331 pinout compatible

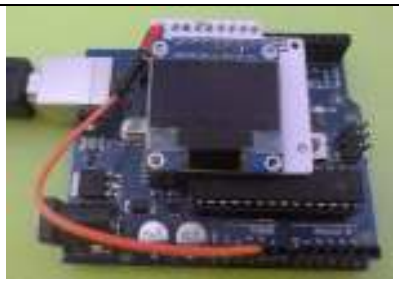


minishield on pins 8-13

Kidule compatible

On Diduino, power is from the power pin replacing the Aref pin (solder drop installed)

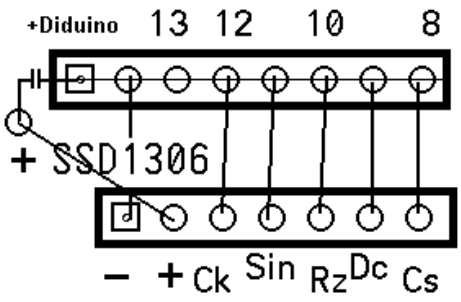
On Arduino boards, add a wire to power the module (3 to 5V).



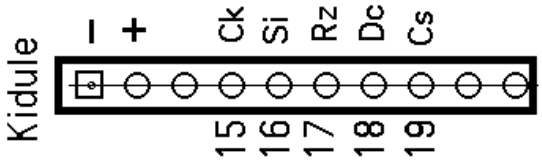
```
Signals SSD1306 blue
  AdaFruit
#define MOSI   9 - D1
#define CK    10 - D0
#define DC    11
#define CS    12
#define RZ    13
```

```
Didel minishield
#define CK   12
#define MOSI 11
#define RZ   10
#define DC    9
#define CS    8
```

On Arduino Uno the +5V must be brought from the power connector on the other side of the board.




```
 Didel kidule
#define sclk   15
#define mosi   16
#define rst    17
#define dc     18
#define cs     19
```



Installation

unzip www.didel.com/xbot/Oled1306.zip

Transfer the 2 files under arduino/libraries/ Put our demo programms in a working file.

Simple demo program – dots and text

```
// TestOled128x64 blue   SSD1306
#include <SPI.h>
#include <Wire.h>
#include <Adafruit_GFX.h>
#include <Adafruit_SSD1306.h>

#define CK   12
#define MOSI 11
#define RZ   10
#define DC    9
#define CS    8
Adafruit_SSD1306 display(MOSI, CK, DC, RZ, CS);


// Color definitions - to be compatible
// BLACK WHITE hidden declare for
//   compatibility with 1331
#define BLUE    WHITE
#define RED     WHITE
#define GREEN   WHITE
#define CYAN    WHITE
#define MAGENTA WHITE
#define YELLOW  WHITE

void setup(void) {
  Serial.begin(9600);
  display.begin();
}
```

```
void loop() {
//  display.clearDisplay();

  // draw single pixels
  display.drawPixel(10, 10, WHITE);
  display.drawPixel(11, 11, WHITE);
  display.drawPixel(12, 12, GREEN);
  display.drawPixel(62, 62, GREEN);
  display.drawPixel(63, 63, GREEN);
  display.drawPixel(126, 62, RED);
  display.drawPixel(127, 63, RED);
  delay(500);

  // text display tests
  display.setTextSize(1);
  display.setTextColor(WHITE);
  display.setCursor(0,0);
  display.println("Kidule");
  display.println("Oled");
  delay(500);
  display.setTextColor(WHITE);
  display.setTextSize(2);
  display.print("Didel");
  display.display();
  delay(500);
// while (1) {}
}
```

Drawing a picture is cumbersome and difficult to correct if you do not use two #define

```
#define P display.drawPixel
#define W WHITE
  byte xx= 80; byte yy= 24;
    P(xx+5,yy,W);P(xx+6,yy,W);P(xx+7,yy,W);P(xx+8,yy,W);P(xx+9,yy,W);P(xx+10,yy,W);
   P(xx+3,yy+1,W);P(xx+4,yy+1,W);                    P(xx+11,yy+1,W);P(xx+12,yy+1,W);
  P(xx+2,yy+2,W);                                                    P(xx+13,yy+2,W);
 P(xx+1,yy+3,W);                                                       P(xx+14,yy+3,W);
 P(xx+1,yy+4,W); P(xx+4,yy+4,W);P(xx+5,yy+4,W);  P(xx+10,yy+4,W);P(xx+11,yy+4,W);  P(xx+14,yy+4,W);
P(xx+0,yy+5,W); P(xx+4,yy+5,W);P(xx+5,yy+5,W);   P(xx+10,yy+5,W);P(xx+11,yy+5,W);  P(xx+15,yy+5,W);
P(xx+0,yy+6,W);                                                                    P(xx+15,yy+6,W);
P(xx+0,yy+7,W);                                                                    P(xx+15,yy+7,W);
P(xx+0,yy+8,W);                                                                    P(xx+15,yy+8,W);
P(xx+0,yy+9,W);  P(xx+2,yy+9,W);                          P(xx+13,yy+9,W);     P(xx+15,yy+9,W);
P(xx+0,yy+10,W); P(xx+3,yy+10,W);P(xx+4,yy+10,W); P(xx+11,yy+10,W);P(xx+12,yy+10,W); P(xx+15,yy+10,W);
P(xx+1,yy+11,W); P(xx+4,yy+11,W);P(xx+5,yy+11,W); P(xx+10,yy+11,W);P(xx+11,yy+11,W); P(xx+14,yy+11,W);
 P(xx+1,yy+12,W);  P(xx+6,yy+12,W);P(xx+7,yy+12,W);P(xx+8,yy+12,W);P(xx+9,yy+12,W); P(xx+14,yy+12,W);
  P(xx+2,yy+13,W);                                                                  P(xx+13,yy+13,W);
   P(xx+3,yy+14,W);P(xx+4,yy+14,W);                        P(xx+11,yy+14,W);      P(xx+12,yy+14,W);
    P(xx+5,yy+15,W);P(xx+6,yy+15,W);P(xx+7,yy+15,W);P(xx+8,yy+15,W);P(xx+9,yy+15,W);P(xx+10,yy+15,W);

display.display();  transfer the buffer to display
```

Adafruit/ Elecrow demos - Just need to change the pinout definition.

Primitives GFX library

```
display.setCursor(x,y);
display.setTextColor(color);
display.setTextSize(i);  //i= 1 6x8  i=2 12x16  i=3 24x32
display.print("abcd");
display.print(val);    //char,byte int long
display.println(val,HEX); //(val,BIN)
```

remove leading zeros

```
display.drawPixel(x,y,color);
display.drawLine(x1,y1,x2,y2,color);
```

others: `drawRect fillRect drawCircle fillCircle drawTriangle drawRoundRect`
see Adafruit GFX graphic Library

Useful debug functions
Set coordinate and color first
Display a number with leading zero suppression

```
display.print(val);   // decimal value
display.print(val,HEX);
display.print(val,BIN);
```

Didel functions for no leading zero suppression

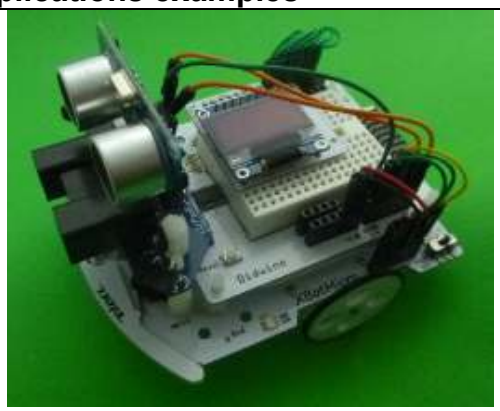
```
P8Hex (val); P16Hex (val); P32Hex (val);
P8Bin (val); P16Bin (val);
Example of graphic functions
LineHoriz (hauteur);
LineVertic (pos);
```

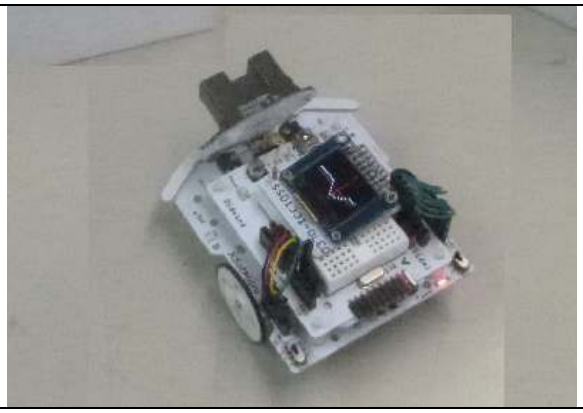
Display also on Terminal if #define Terminal

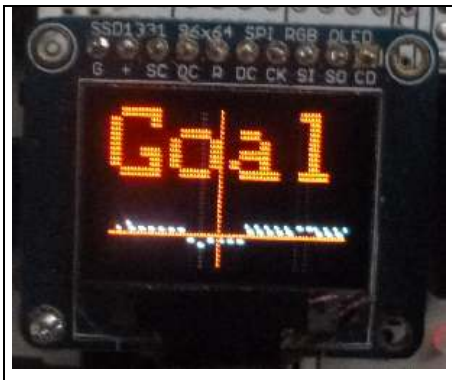
```
// OlibNbres.h
// --- librairie
byte x,y;
char ConvNibble (byte nn){ // 4 bit hexa → Ascii
  char cc;
  if (nn<10) cc = nn + '0';
  else cc = nn-10 + 'A';
  return cc;
  }
void P8Hex (byte hh) {
  display.setCursor(x,y);
  char cc = ConvNibble (hh >> 4);// hh not modif
  Serial.print (cc); display.print (cc);
  cc = ConvNibble (hh & 0x0F) ;
  display.print (cc);
  x+=2*6+4;
}
```

```
void P8Bin (byte bb) {
  display.setCursor(x,y);
  for (byte i=0;i<8;i++) {
    if (bb&0x80)
 {Serial.print ('1'); display.print ('0'); }
    else
 {Serial.print ('0'); display.print ('0'); }
    bb <<= 1;
  }
}
void P16Bin (unsigned int bb) {
  display.setCursor(x,y);
  for (byte i=0;i<16;i++) {
     if (bb&0x8000)
{Serial.print ('1'); display.print ('0'); }
     else
{Serial.print ('0');display.print ('0'); }
     bb <<= 1;
  }
}
```

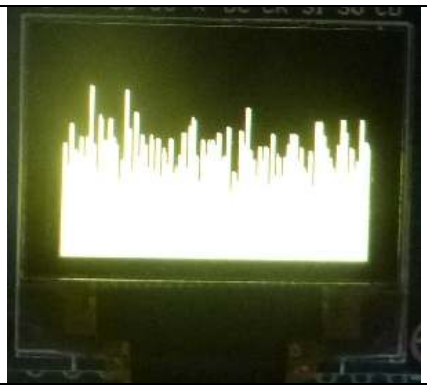
Applications examples



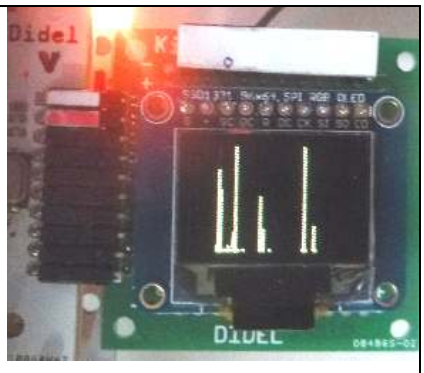




Curves



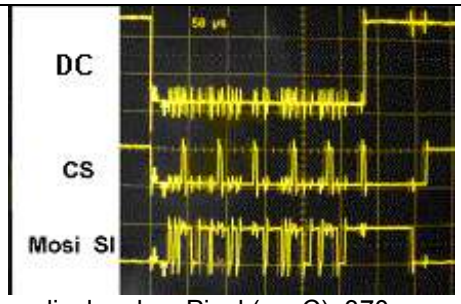
Histogram (random numbers)



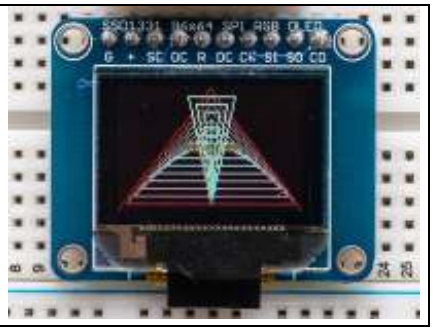
Histogram of measures

Oled 96x64 color SSD1331 and minishied

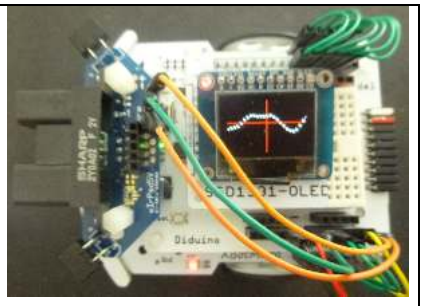
The SSD1331 Oled is available from Boxtec and Adafruit. Its resolution is 96x64 pixels, RGB, 16 levels of color.
OLED Breakout Board - 16-bit Color 0.96" w/ microSD holder [45070].



display.drawPixel (x,y,C) 370 us



Didel Minishield adapter is compatible with this circuit and is available from Boxtec (bare PCB, no connectors).



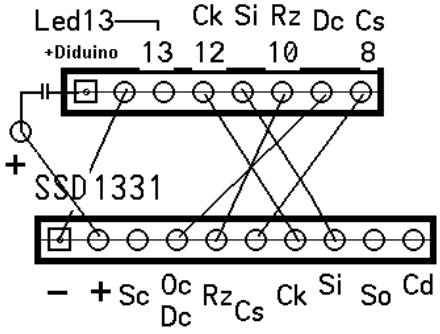
Installation

unzip www.didel.com/xbot/Oled1331.zip
Transfer the 2 files under arduino/libraries/ Put our demo programms in a working file.

Demo software from Adafruit can be used by changing only the #define.

Signals SSD1331 coul	Didel minishield
#define sclk 13	#define sclk 12
#define mosi 11	#define mosi 11
#define cs 9	#define rst 10
#define rst 10	#define dc 9
#define dc 8	#define cs 8

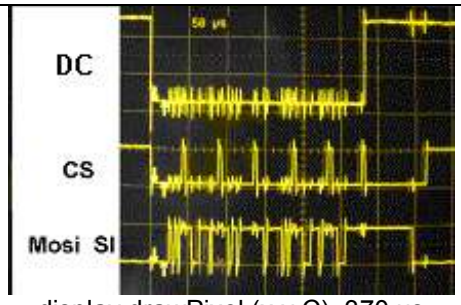
CS Rz DC Ck Si



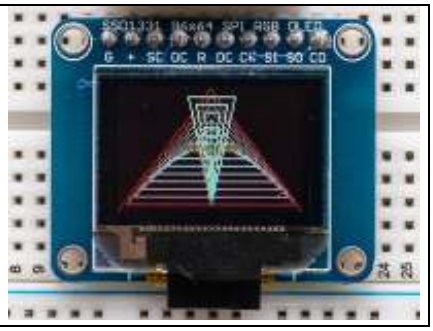
The SD card of the SSD1331 needs one more signal to be accessed. A jumper wire will be used to connect to the documented signal.

This Oled is available from Boxtec and Adafruit. Its resolution is 96x64 pixels, RGB, 16 levels of color.

Didel Oled minishield is compatible.
PCB only available for 4 CHF.



display.drawPixel (x,y,C) 370 us



Software incompatibilities between 1306 and 1331

Only on SSD1306	Only on SSD1331
`display.clearDisplay();` // clears the screen and buffer	`display.fillScreen(couleur);`

`display.display();` // required after text and lines (after initialisation, display Adafruit logo) `..(color);`	`...(WHITE);`

Our demo software is on the previously mentionned zip.

SSD1331

```
// TestOled96x64.ino  color
#include <SPI.h>
#include <Adafruit_GFX.h>
#include <Adafruit_SSD1331.h>


#define sclk 12
#define mosi 11
#define cs   8
#define dc   9
#define rst  10
Adafruit_SSD1331 display(cs, dc, mosi, sclk,
rst);
//Adafruit_SSD1331 display = Adafruit_SSD1331(cs, dc,
mosi, sclk, rst);
// Color definitions
#define BLACK     0x0000
#define BLUE      0x001F
#define RED       0xF800
#define GREEN     0x07E0
#define CYAN      0x07FF
#define MAGENTA   0xF81F
#define YELLOW    0xFFE0
#define WHITE     0xFFFF

void setup(void) {
  Serial.begin(9600);
  display.begin();
  display.fillScreen(BLACK);
}
```

```
void loop() {
//  display.clearDisplay();

  // draw single pixels
  display.drawPixel(10, 10, WHITE);
  display.drawPixel(11, 11, WHITE);
  display.drawPixel(12, 12, GREEN);
  display.drawPixel(62, 62, GREEN);
  display.drawPixel(63, 63, GREEN);
  display.drawPixel(126, 62, RED);
  display.drawPixel(127, 63, RED);
  delay(500);

 // text display tests
  display.setTextSize(1);
  display.setTextColor(WHITE);
  display.setCursor(0,0);
  display.println("Kidule");
  display.println("Oled");
  delay(500);
  display.setTextColor(WHITE);
  display.setTextSize(2);
  display.print("Didel");
  delay(500);
// while (1) {}
}
```

jdn 140830/141208