

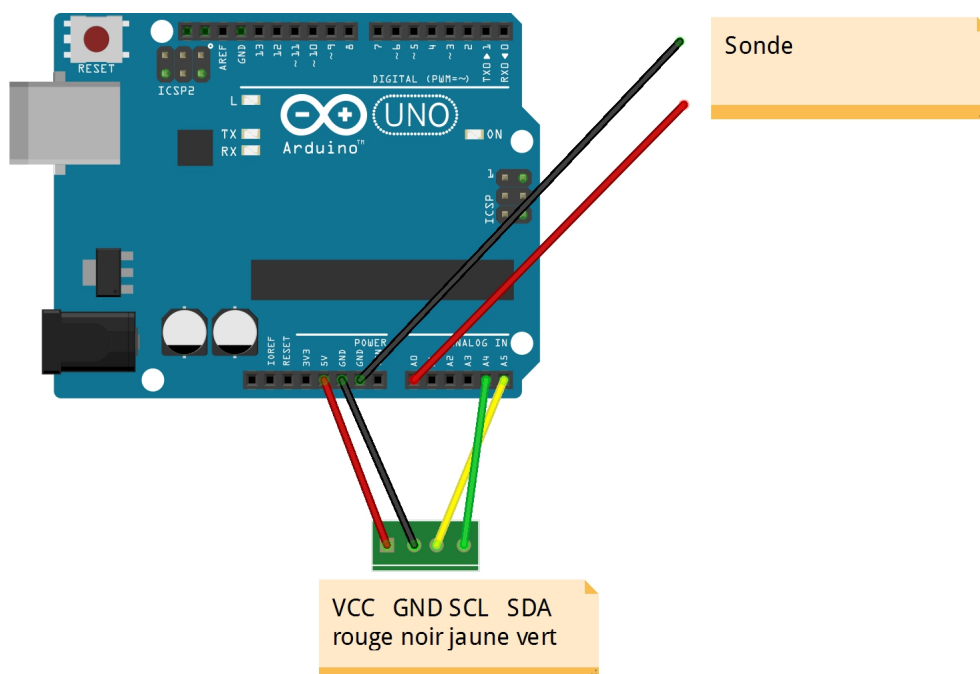
# Ecran Oled, mini oscilloscope

Oled LCD afficheur 0.96" I2C 128x64



Pric indicatif hors taxe, livraison comprise (Décembre 2014) : 4 €

## Schéma



fritzing

## Description

Cet écran a une définition de 128 pixels en largeur et de 64 pixels en hauteur.

Il est "monochrome" sachant que les 16 premières lignes sont de couleur jaune et les 48 dernières de couleur bleue. Il existe des modèles où les 64 lignes sont bleues.

L'interface est de type i2c. Cet écran fonctionne avec la bibliothèque u8glib:

[U8GLIB page principale](#)

Le code décrit ci-après permet de mesurer et de visualiser un signal basse-fréquence ( < 60 hz)



La valeur affichée en Hz est la fréquence de rafraîchissement de l'écran et non pas celle du signal.

**ATTENTION : les entrées analogiques de l'Arduino ne supportent pas une tension supérieure à 5 V.**

## Code

```
#include "U8glib.h"
#define maxW (128) // largeur ecran en pixels
#define maxH (63) // hauteur ecran en pixels
#define maxG (47) // hauteur max graphique
U8GLIB_SSD1306_128X64 u8g(U8G_I2C_OPT_NO_ACK); // type ecran
int sensorPin=A0; // broche analogique pour la sonde
int sensorValue=0; // valeur brute
int col=0; // colonne courante
char tension[16]; // nombre de caracteres par ligne
int ty[maxW]; // valeurs des tensions
int ref=5; // tension de reference
int diviseur=1024/ref; // diviseur de tension
unsigned long freq=0; // nombre de rafraichissements d'ecran realises
void draw(void)
{
  int i;
  u8g.drawStr(5,14,tension);
  for(i=0;i<maxW;i++) u8g.drawPixel(i,ty[i]);
}
void setup(void)
{
  int i;
  for(i=0;i<maxW;i++) ty[i]=maxH;
  u8g.begin();
  u8g.setFont(u8g_font_unifont);
}
void loop(void)
{
  u8g.firstPage();
```

```
do
{
  sensorValue=analogRead(sensorPin);
  ty[col]=maxH-map(sensorValue,0,1023,0,maxG);
  col++;
  if(col>=maxW) col=0;
  freq++;
  sprintf(tension,"%03d Hz %01d.%03d V",(int)(freq*1000/millis()),
  sensorValue/diviseur,
  sensorValue%diviseur);
  draw();
}
while(u8g.nextPage());
}
```