Installer Arduino et Fritzing

Sources

https://www.arduino.cc/

http://fritzing.org/home/

Attention Les commandes sont données avec aptitude comme gestionnaire de paquet (idem que **apt** ou **apt-get**) L'exemple d'installation est donné pour du 64bit (AMD64) et pour la dernière version d'Arduino disponible (à ce jour la 1.8.8) A adapter à vos besoins.

1/ Ajouter votre utilsateur aux groupes tty et dialout

1.1/ Ajouter votre utilsateur aux groupes tty et dialout

```
$ sudo addgroup $USER dialout
$ sudo addgroup $USER tty
```

Se delogguer et se relogguer pour que les modifications soient prises en compte.

1.2/ Vérification de la configuration

Pour vérifier que ces commandes soient bien prises en compte :

```
$ groups
ragnarok tty uucp dialout cdrom floppy sudo audio dip video plugdev netdev
```

Doit faire apparaître **tty** et **dialout**

Et l'Arduino va communiquer via ce device : Exemple pour une Arduino UNO R3

```
ls -l /dev/ttyACM*
crw-rw---- 1 root dialout 166, 0 janv. 17 20:22 /dev/ttyACM0
```

Attention ce device n'apparaîtra que si la carte Arduino est branchée. Pas forcement un 0 comme chiffre à la fin de ttyACM, peut être un 1, 2, 3... Dépend de la configuration. Mais ce device doit appartenir au groupe **dialout**.

Et cette commande, également carte branchée, doit faire apparaître votre carte

```
$ lsusb | grep Arduino
Bus 001 Device 003: ID 2341:0043 Arduino SA Uno R3 (CDC ACM)
```

Fork d'Arduino Il existe sur le marché des forks de cartes d'Arduino ... Pour un fork d'une carte UNO R3, la configuration est identique mais la carte va communiquer via un autre device comme ttyUSB0 ou 1, 2, 3...

Exemple pour une ATMEGA328P

```
ls -l /dev/ttyUSB*
crw-rw----+ 1 root dialout 188,  0 janv. 18 21:51 /dev/ttyUSB0
```

Lors du lancement de l'IDE d'Arduino, le port série sera donc différent et la carte aussi (ici on choisira une Arduino Duemilanone or Diecimila)

2/ Installer l'IDE d'Arduino

2.1/ Installer Arduino depuis le dépôt officiel de votre distribution

Disponible dans les dépôts, cette version est fonctionnelle mais ce n'est pas la plus à jour.

```
$ aptitude show arduino
Paquet : arduino
Version : 2:1.0.5+dfsg2-4.1
État: non installé
Priorité : optionnel
Section : electronics
Responsable : Scott Howard <showard@debian.org>
Architecture : all
Taille décompressée : 1 657 k
Dépend: default-jre | java6-runtime, libjna-java, librxtx-java (>=
2.2pre2-3),
        arduino-core (= 2:1.0.5+dfsg2-4.1)
Recommande: extra-xdg-menus, policykit-1
Description : EDI et bibliothèques pour carte d'expérimentation AVR
 Arduino est une plate-forme libre pour prototypages électroniques, basée
sur du
 matériel et logiciel flexibles et faciles d'utilisation. Elle est destinée
aux
 artistes, concepteurs, passionnés et toute personne intéressée par la
création
 d'objets ou d'environnements interactifs.

 Ce paquet installera l'environnement de développement intégré qui permet
 d'écrire des programmes, de vérifier le code, de le compiler et de le
 transférer sur la carte de développement Arduino. Des bibliothèques et des
 exemples de code seront également installés.
Site : http://www.arduino.cc
Étiquettes: field::electronics, role::program
```

L'installer

```
sudo aptitude install arduino
```

2.2/ Installer la dernière version de l'IDE d'Arduino

Si vous désirez bénéficier de la dernière version de l'IDE d'Arduino. Il faudra réaliser une installation manuelle.

Récupérer et installer la dernière version d'Arduino

```
$ wget https://www.arduino.cc/download.php?f=/arduino-nightly-linux64.tar.xz
$ tar xvf download.php\?f\=%2Farduino-nightly-linux64.tar.xz
$ mv arduino-nightly/ ~/arduino-1.8.8
cd ~/arduino-1.8.8
./install.sh
```

Attention ne pas supprimer le dossier arduino-1.8.8



A ce jour la version d'Arduino la plus récente est la 1.8.9

3/ Installer Fritzing

3.2 Qu'est-ce que Fritzing ?

Fritzing est un logiciel libre de conception de circuit imprimé qui permet de concevoir de façon entièrement graphique le circuit et d'en imprimer le typon. Il permettra, entre autres, de dessiner vos branchements en la breadboard et la carte Arduino. Il génère également des vues schématiques et des circuits imprimés (permet également leur commande...). Il peut également servir d'IDE pour programmer vos cartes Arduino.

```
$ aptitude show fritzing
Paquet : fritzing
Version : 0.9.3b+dfsg-9
État: installé
Automatiquement installé: non
Priorité : optionnel
Section : electronics
Responsable : Enrique Hernández Bello <ehbello@gmail.com>
Architecture : amd64
Taille décompressée : 8 558 k
Dépend: libc6 (>= 2.14), libgcc1 (>= 1:3.0), libgit2-27 (>= 0.26.0), libgl1,
        libqt5concurrent5 (>= 5.0.2), libqt5core5a (>= 5.11.0~rc1),
libqt5gui5
        (>= 5.3.0), libqt5network5 (>= 5.0.2), libqt5printsupport5 (>=
5.0.2),
```

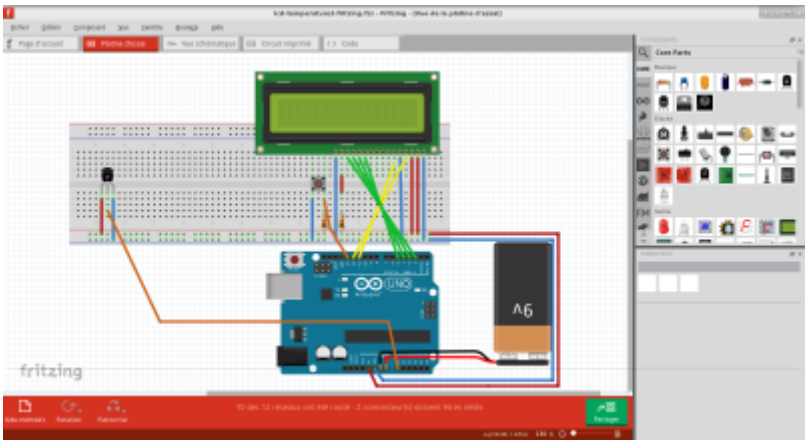
```
        libqt5serialport5 (>= 5.6.0), libqt5sql5 (>= 5.0.2), libqt5svg5 (>=
        5.6.0~beta), libqt5widgets5 (>= 5.11.0~rc1), libqt5xml5 (>= 5.0.2),
        libstdc++6 (>= 5.2), zlib1g (>= 1:1.1.4), fritzing-data (=
        0.9.3b+dfsg-9), libqt5sql5-sqlite, fritzing-parts
Description : Easy-to-use electronic design software
 Fritzing is an open source project designed to help one transition from a
 prototype to a finished project. Aimed at users who want to produce or
document
 circuits and experiments, one starts by building a physical prototype, then
 recreating it with Fritzing's graphical editor. From there one can generate
a
 schematic, PCB artwork, and PCB production files.
Site : http://fritzing.org
Étiquettes: field::electronics, implemented-in::c++, interface::graphical,
            interface::x11, role::program, scope::application,
uitoolkit::qt,
            x11::application
```

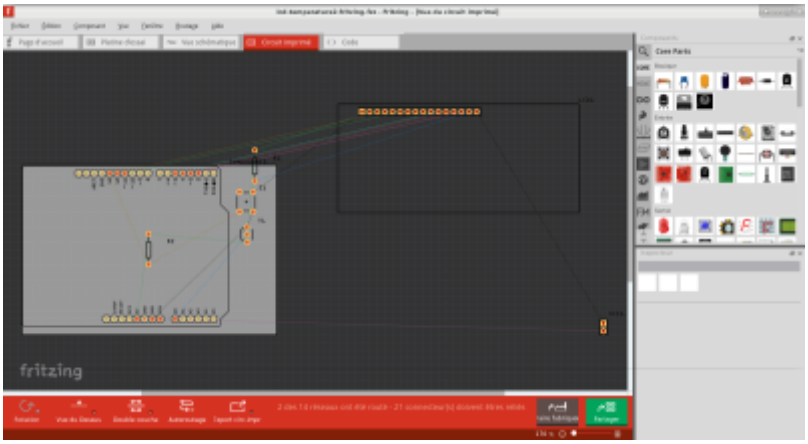
Et l'installer

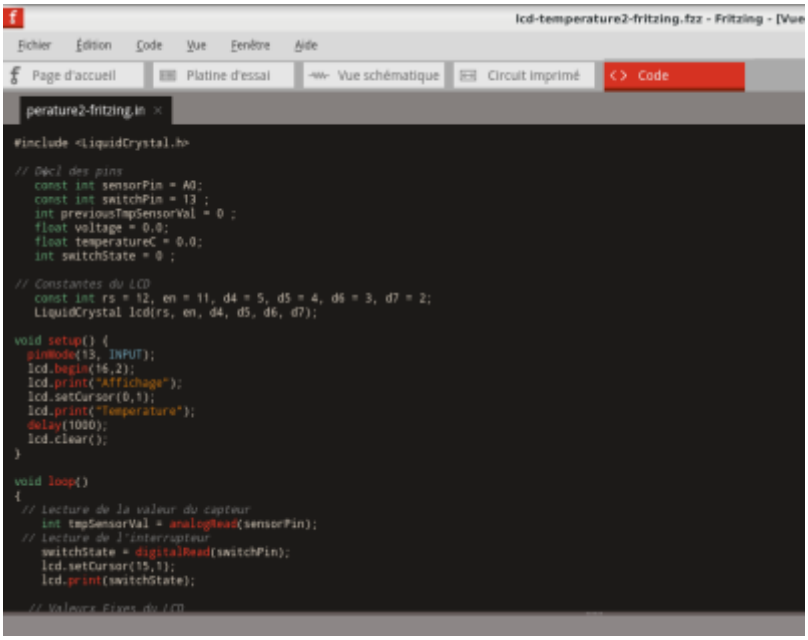
```
$ sudo aptitude install fritzing
```

3.2 En images

Pour se faire une idée, ça donne ce genre de chose :

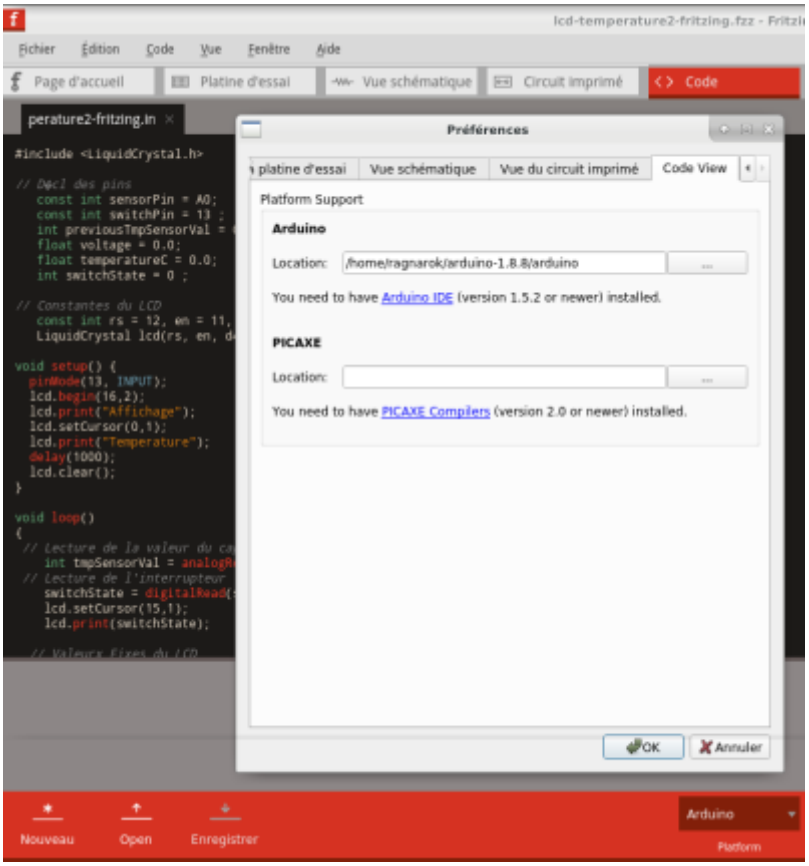






3.3 Configurer Fritzing comme IDE Arduino

Lancer Fritzing, puis **Fichier**, **Préférences**

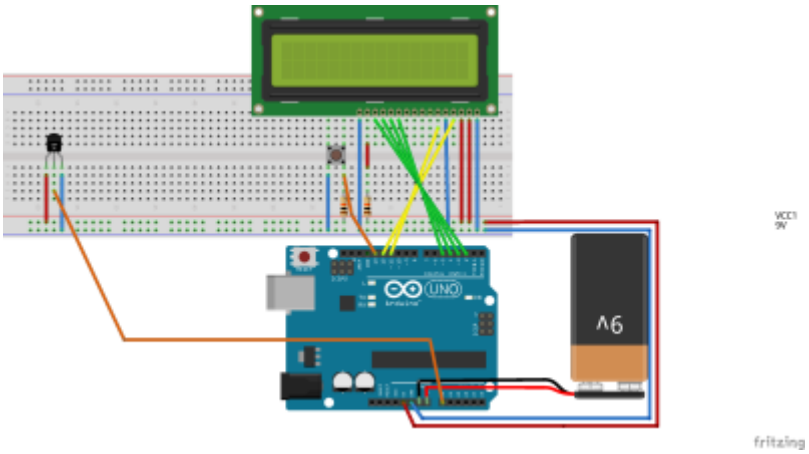


Indiquer comme Location l'endroit dossier contenant le programme Arduino.

4. Exemples

Très simple un écran LCD, un capteur de température TMP36, un switch. Réalisation d'un thermomètre. Pour économiser l'énergie, l'écran ne s'affiche que si on appuie sur le switch. (pas obligé d'utiliser une pile. On peut utiliser l'alimenter en 5V via la prise USB du PC. Dans ce cas brancher : 5V de la carte au + de la breadboard et la masse au - de la breadbord.)

Montage :



Et le code associé :

```
#include <LiquidCrystal.h>

// Décl des pins
   const int sensorPin = A0;
   const int switchPin = 13 ;
   int previousTmpSensorVal = 0 ;
   float voltage = 0.0;
   float temperatureC = 0.0;
   int switchState = 0 ;

// Constantes du LCD
   const int rs = 12, en = 11, d4 = 5, d5 = 4, d6 = 3, d7 = 2;
   LiquidCrystal lcd(rs, en, d4, d5, d6, d7);

void setup() {
  pinMode(13, INPUT);
  lcd.begin(16,2);
  lcd.print("Affichage");
  lcd.setCursor(0,1);
  lcd.print("Temperature");
  delay(1000);
 // Reinitialisation LCD
    lcd.clear();
}

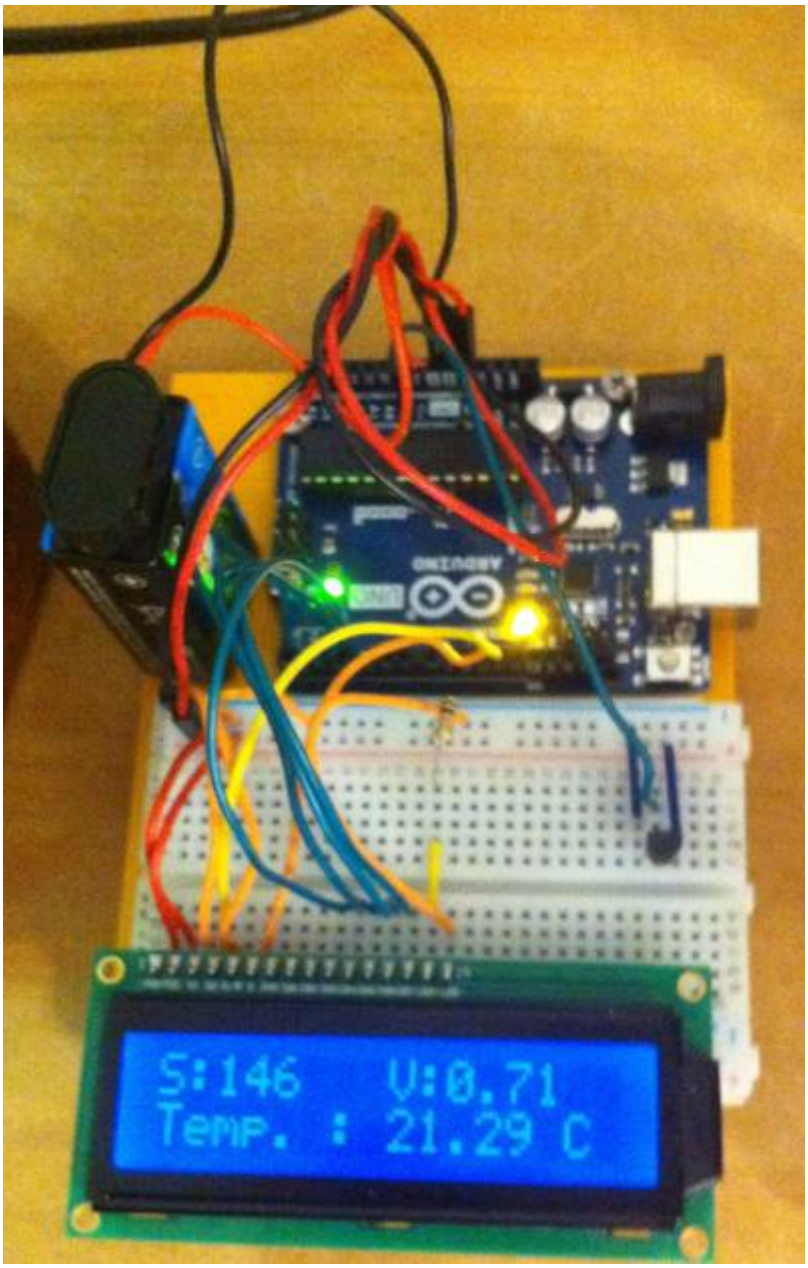
void loop()
{
 // Lecture de la valeur du capteur
    int tmpSensorVal = analogRead(sensorPin);
 // Lecture de l'interrupteur
    switchState = digitalRead(switchPin);
    lcd.setCursor(15,1);
    lcd.print(switchState);

  // Valeurx Fixes du LCD
        lcd.setCursor(8,0);
        lcd.print("V:");
        lcd.setCursor(0,1);
        lcd.print("Temp.:");
        lcd.setCursor(12,1);
        lcd.print("C");

  // Si changement capteur, on (re)calcule la temperature
       if (tmpSensorVal != previousTmpSensorVal )  {
       // Calcul du volatge et conversion en degré Celsius
          voltage = tmpSensorVal * 5.0;
          voltage /= 1024.0;
          temperatureC = (voltage - 0.5) * 100 ;
          previousTmpSensorVal = tmpSensorVal ;
          }
  // Affichage sur demande (appui switch)
```

```
       if (switchState == 1) {
           // Affichage LCD
              lcd.display();
              lcd.setCursor(0,0);
              lcd.print(tmpSensorVal);
              lcd.setCursor(10,0);
              lcd.print(voltage);
              lcd.setCursor(7,1);
              lcd.print(temperatureC);
              delay(10000);
           } else { // On éteint le LCD
              lcd.noDisplay();
           }
delay(500);
}
```

Photo :



From:
https://cbiot.fr/dokuwiki/ - **Cyrille BIOT**

Permanent link:
https://cbiot.fr/dokuwiki/arduiro:installer-arduino-fritzing-linux

Last update: **2019/04/03 14:13**