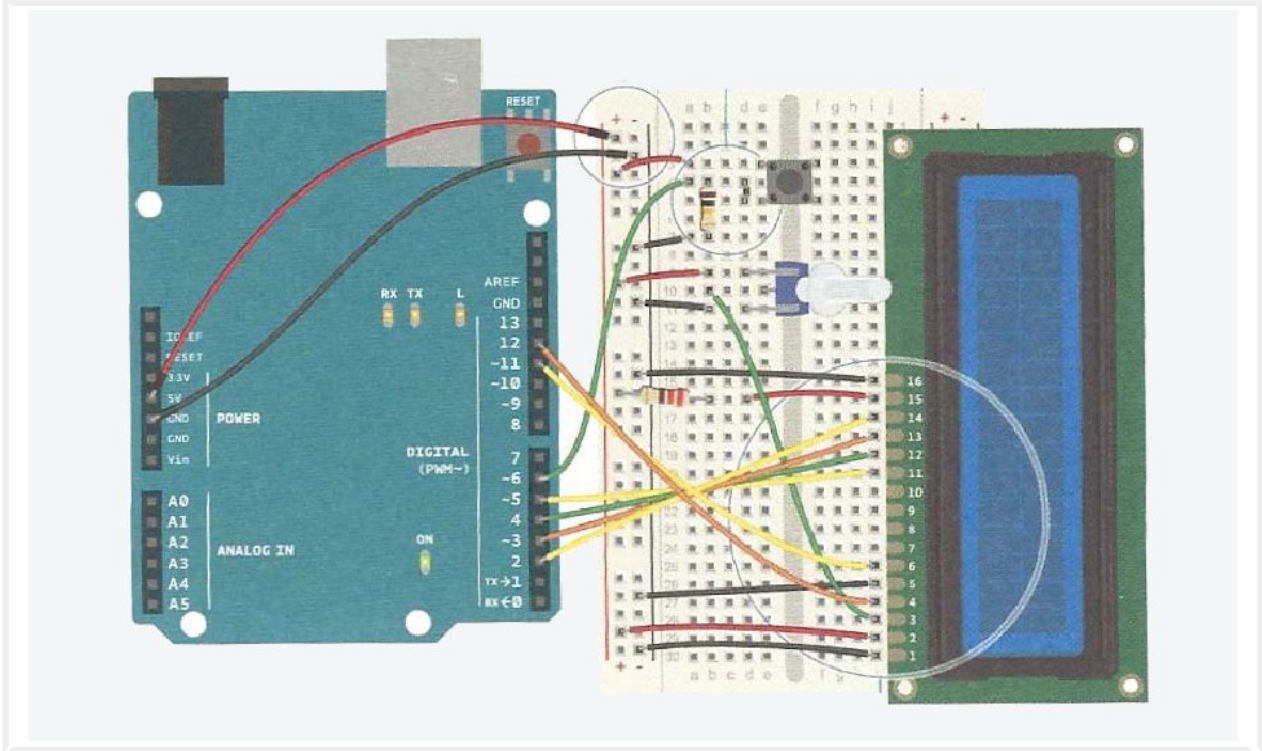


Ecran LCD

(Afficher un chiffre sur un écran LCD)



Liste des composants

- . 1 écran LCD
- . 1 bouton poussoir
- . 1 potentiomètre de 10 k Ω
- . 1 résistance de 10 k Ω
- . 1 résistance de 220 Ω
- . 1 plaque d'essais
- . Fils de connexion

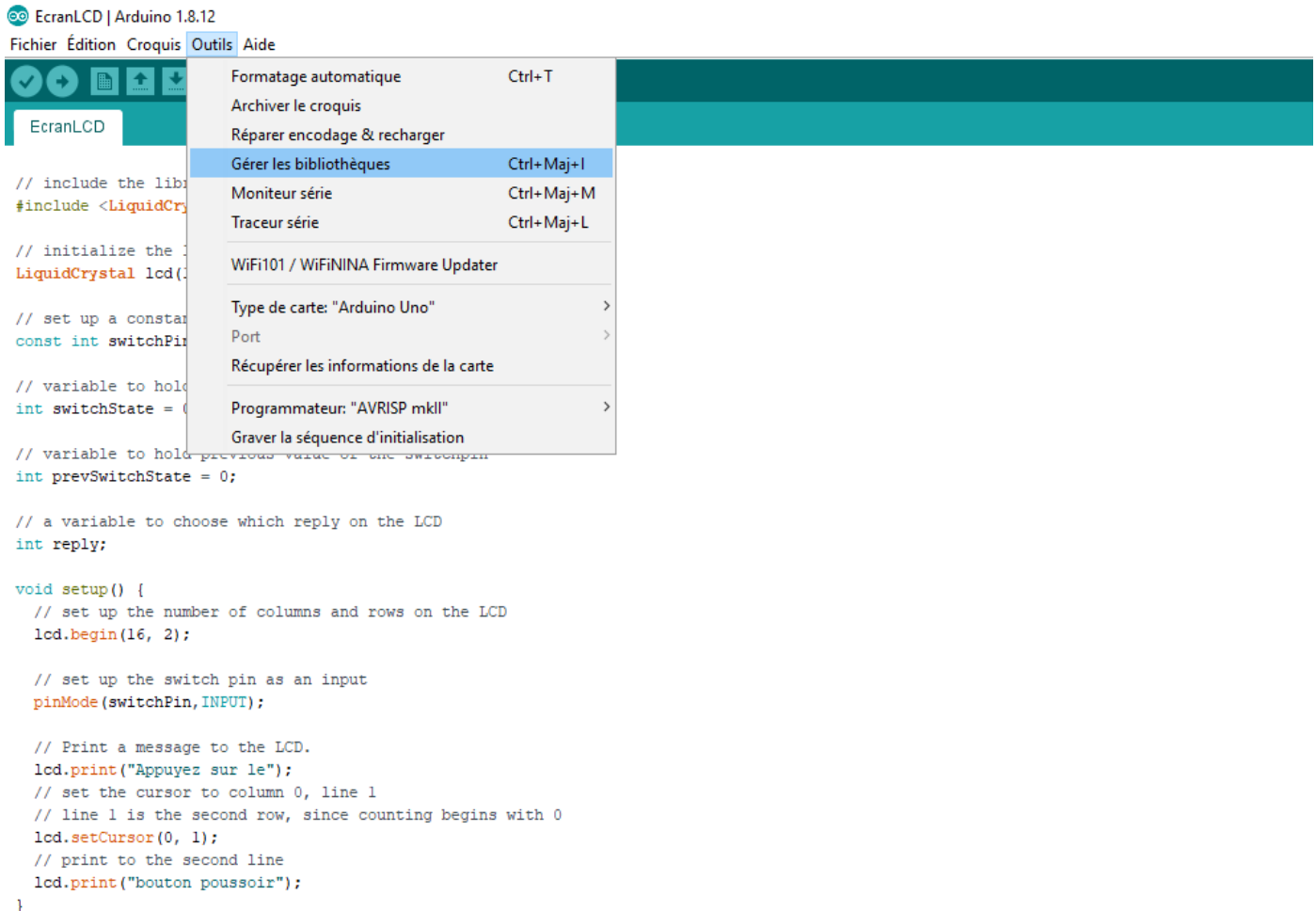
Objectif

L'objectif de cette activité est d'afficher un chiffre choisi au hasard sur un écran LCD après un appui sur un bouton poussoir.

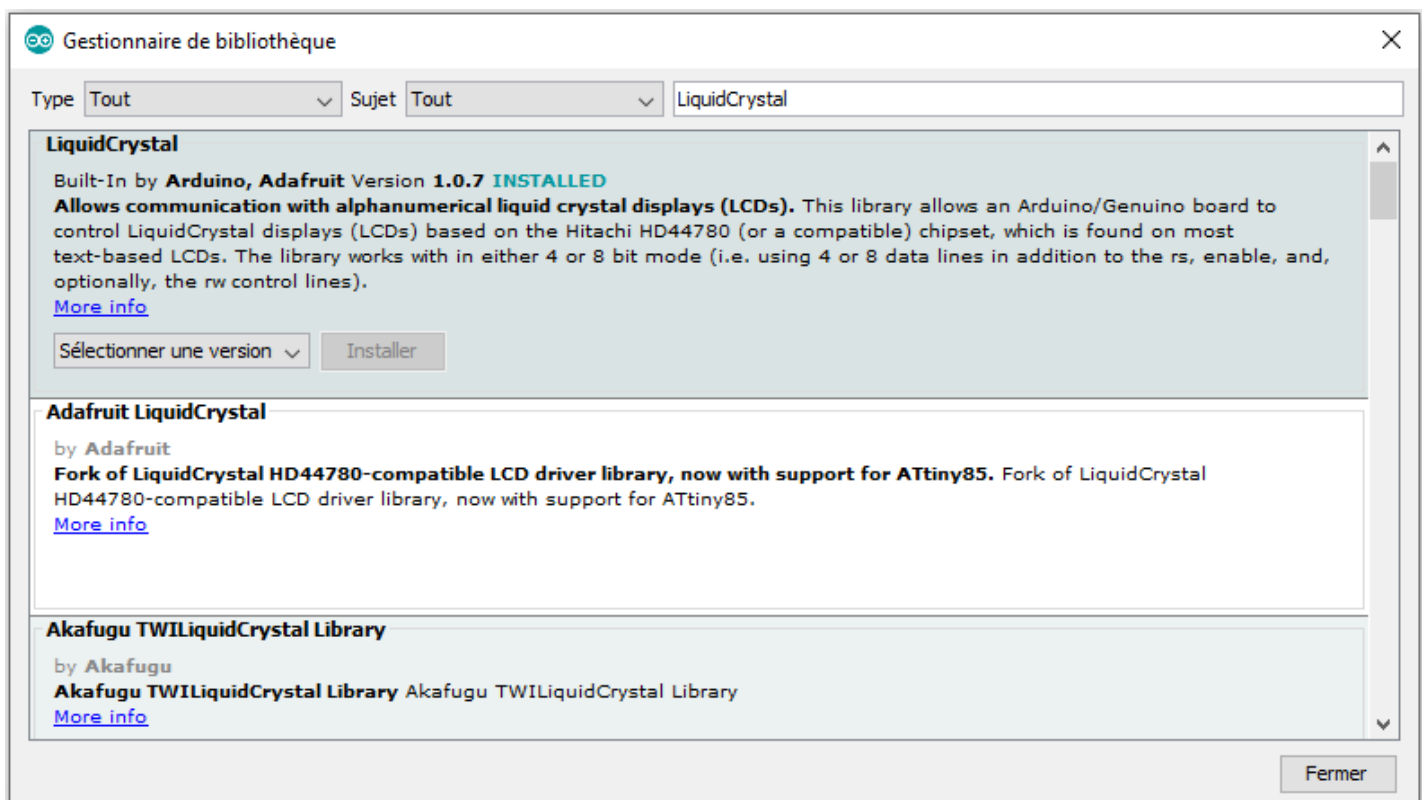
L'écran à cristaux liquides utilisé comporte deux rangées de 16 caractères, d'où son nom de LCD 1602 et le potentiomètre du montage permet de régler sa luminosité.

Le code de l'activité nécessite l'installation au préalable de la librairie " **LiquidCrystal** " d'Adafruit.

Afin d'ajouter une librairie à l'IDE Arduino, il faut aller dans le menu « **Outils ->Gérer les bibliothèques** » :



Il suffit ensuite de rechercher et d'ajouter la librairie " **LiquidCrystal** " d'Adafruit:



. Le programme

Voici le code de l'activité :

EcranLCD

```
// include the library code:
#include <LiquidCrystal.h>

// initialize the library with the numbers of the interface pins
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);

// set up a constant for the switchPin
const int switchPin = 6;

// variable to hold the value of the switchPin
int switchState = 0;

// variable to hold previous value of the switchpin
int prevSwitchState = 0;

// a variable to choose which reply on the LCD
int reply;

void setup() {
  // set up the number of columns and rows on the LCD
  lcd.begin(16, 2);

  // set up the switch pin as an input
  pinMode(switchPin, INPUT);

  // Print a message to the LCD.
  lcd.print("Appuyez sur le");
  // set the cursor to column 0, line 1
  // line 1 is the second row, since counting begins with 0
  lcd.setCursor(0, 1);
  // print to the second line
  lcd.print("bouton poussoir");
}

void loop() {
  // check the status of the switch
  switchState = digitalRead(switchPin);

  // compare the switchState to its previous state
  if (switchState != prevSwitchState) {

    if (switchState == HIGH) {
      // randomly chose a reply
      reply = random(10);
      // clean up the screen before printing a new reply
      lcd.clear();
    }
  }
}
```

```

// set the cursor to column 0, line 0
lcd.setCursor(0, 0);
// print some text
lcd.print("Chiffre:");
// move the cursor to the second line
lcd.setCursor(0, 1);

switch(reply) {
case 0:
    lcd.print("Zero");
    break;
case 1:
    lcd.print("Un");
    break;
case 2:
    lcd.print("Deux");
    break;
case 3:
    lcd.print("Trois");
    break;
case 4:
    lcd.print("Quatre");
    break;
case 5:
    lcd.print("Cinq");
    break;
case 6:
    lcd.print("Six");
    break;
case 7:
    lcd.print("Sept");
    break;
case 8:
    lcd.print("Huit");
    break;
case 9:
    lcd.print("Neuf");
    break;
}
}
}
// save the current switch state as the last state
prevSwitchState = switchState;
}

```

Déroulement du programme :

– 1. Insertion des bibliothèques :

- . Insertion de la librairie " LiquidCrystal " d'Adafruit
- . Initialisation de la librairie avec les broches utilisées pour l'écran LCD

– 2. Déclaration des constantes et variables :

- . const int switchPin = 6 (constante nombre entier correspondant à la broche du bouton poussoir)

. int switchState = 0 (variable nombre entier pour stocker la valeur du potentiel de la broche du bouton poussoir)

. int prevSwitchState = 0 (variable nombre entier pour stocker l'ancienne valeur du potentiel de la broche du bouton poussoir)

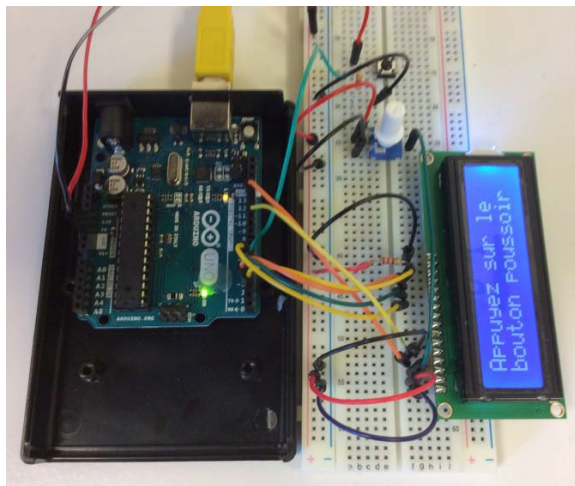
. int reply (variable nombre entier correspondant au chiffre qui doit être affiché)

– 3. Initialisation des entrées et sorties :

. **Initialisation du nombre de lignes et de colonnes de l'écran LCD,**

. **Initialisation de la broche du bouton poussoir en entrée,**

. **Affichage de la consigne « Appuyer sur le bouton poussoir » sur l'écran LCD.**



– 4. Fonction principale en boucle :

–> **Lecture de la valeur de la broche du bouton poussoir,**

–> **Si le bouton poussoir est appuyé, un chiffre est tiré au sort,**

–> **Affichage du chiffre sur l'écran LCD.**

