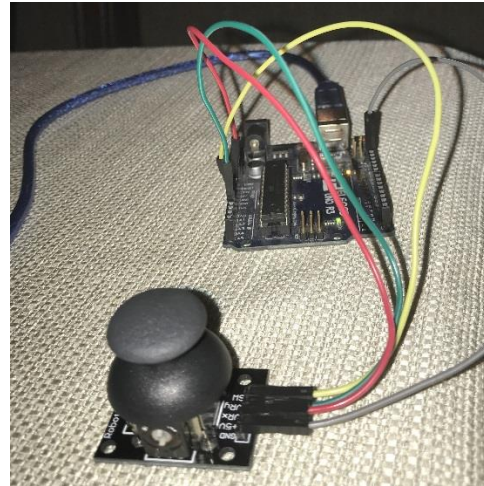
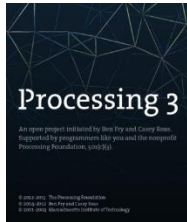


DIGITAL BRIDGE



Joystick connecté à Arduino

Leçon 3

But : Un autre exemple de l'utilisation des signaux analogues est de faire varier la position d'une boule sur l'écran avec l'aide d'un Joystick. Un joystick est en réalité constitué de deux potentiomètres capables de faire varier les coordonnées cartésiennes (X Y) d'un point. Dans cette leçon nous ferons usage du moniteur de série pour visualiser les valeurs X Y dans la console. Les commandes **Serial.begin(9600)** et **Serial.print(paramètre)** seront introduites dans cette leçon.

Processing est un software de présentation graphique et sera utilisé abondamment dans cette classe pour visualiser les codes Arduino. **A ce stade recopier tout simplement le code** et l'explication sera donné au fur et à mesure que nous évoluons avec les leçons prochaines.

Serial.begin(9600)

Cette instruction permet d'initier la communication entre la carte Arduino et le Moniteur de série, ou entre la carte Arduino et tout autre équipement (ordinateur, smartphone, ou autres logiciels). Le débit de la communication (Baud rate) est de 9600 dans cet exercice. Le débit est variable et il est typique d'utiliser 9600 ou 115200 avec Arduino.

Serial.begin est toujours placé dans le bloc de la fonction **void setup()**

Serial.print(paramètre)

L'instruction **Serial.print** rend visible dans la console du Moniteur de série les valeurs des variables transmises par le port de communication. Dans le cas de l'exercice suivant les signaux analogues **xValue** et **yValue** peuvent être observés dans la console. Ce sont ces valeurs qui sont transmises sous forme d'une variable de type **String** au code de Processing. En bref, c'est **Serial.print(paramètre)** qui passent les valeurs X et Y provenant du joystick à la console et au code de Processing.

NOTE : Fermer la fenêtre du Moniteur de série avant de lancer (run) Processing. La carte Arduino ne peut communiquer qu'avec un seul moniteur à la fois.

Code Arduino : Déplacer une balle sur l'écran avec le joystick

```
int xValue = 0; // Signal entrant Analog X VRX ==> A0
int yValue = 0; // Signal entrant Analog Y VRY==> A1
// Branchez + 5V ==> 5V et GND ou (-) ==> GND
void setup()
{
  Serial.begin(9600); // Vitesse du débit du port de communication
}

void loop()
{
  // Points de raccordement des signaux analogues entrant A0 & A1

  xValue = analogRead(A0);
  yValue = analogRead(A1);

  // Affichage des valeurs X & Y dans le Moniteur de serie
  Serial.print(xValue,DEC);
  Serial.print(",");
  Serial.print(yValue,DEC);
  Serial.print(".");

  delay(100);
}
```

Code Processing : Visualise la balle sur l'écran avec le joystick

```
import processing.serial.*; // importe la bibliotheque de communication serie

Serial myPort; // definit l'Objet Serial

// Definition des variables
int x;
int y;
int b;
int index1 = 0;
PFont f;
String portName;
String positionX = "";
String positionY = "";
String data = "";
void setup()
{
```

```

size(512, 512); // Dimension de l'ecran en pixels
portName = "COM3";
myPort = new Serial(this, portName, 9600); // Commence la communication en serie
myPort.bufferUntil('.'); // Lit les donnees jusqu'au caractere '.'

f = createFont("Arial", 16, true); // style du caractere des ecrits
textFont(f, 16);
}

void draw()
{
  fill(100);
  clear();
  fill(200,128,50); // Couleur de la boule (R,G,B)

  ellipse(x/2, 512-(y/2), 25, 25); // Position et dimension de la boule

  text("AnalogX="+ (1023-x) + " AnalogY="+ (1023-y), 10, 20);
}

void serialEvent(Serial myPort)
{
  // Lecture des donnees du port de communication en serie et separation ',' et '.'
  data = myPort.readStringUntil('.');
  data = data.substring(0, data.length()-1);

  index1 = data.indexOf(",");
  positionX = data.substring(0, index1);
  positionY = data.substring(index1+1, data.length());

  // convertit les String en variables Integer
  x = int(positionX);
  y = int(positionY);
}

```