

## Qu'est-ce que c'est ?

Involt est une application qui permet l'échange de données entre une carte Arduino et un site HTML via le câble USB.



## De quoi Involt est-il constitué ?

Involt est constituée de :

- Une page HTML, à compléter.
- Un programme Arduino, à compléter.
- Un ensemble de fichiers dans des dossiers, à ne pas modifier.

## Comment cela fonctionne-t-il ?

Involt traduit les classes CSS en fonctions, en utilisant du code Javascript.

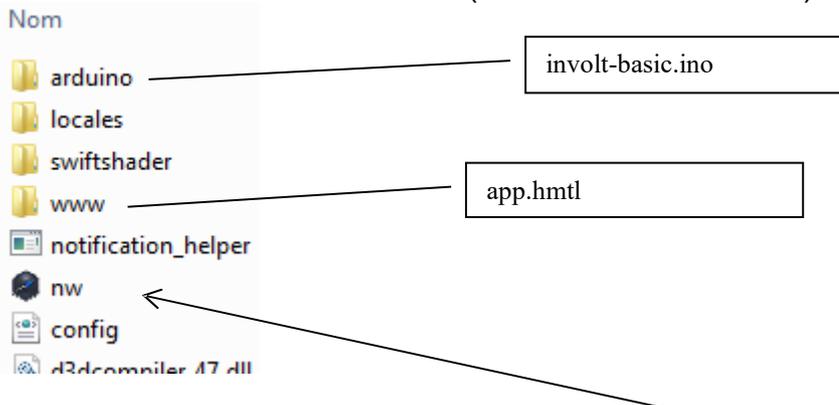
## Quelles sont les connaissances requises ?

Les connaissances de bases en **HTML**, **CSS** et **C pour Arduino** suffisent.

## Que faut-il faire ?

Après avoir installé l'application, vous devrez utiliser le fichier **app.html** pour créer votre page HTML et le fichier **involt-basic.ino** pour écrire votre code Arduino.

Vous devez modifier ces fichiers (sans modifier leur nom) dans l'arborescence suivante.



Le fonctionnement s'effectuera en exécutant le fichier **nw.exe** (puis cliquer sur le COM... et valider)

## Exemples basiques :

1. Envoyer une donnée de la carte Arduino vers la page HTML



il faut utiliser la syntaxe suivante :

**involtSend(pinIndex, value)** en C Arduino et « **ard show** » en HTML

Exemple :

Arduino

```
void loop() {
  involtSend(1, analogRead(A0));
}
```

*Lecture du port analogique A0 et envoi dans le canal n°1*

Page HTML

HTML

```
<div class="ard show A1 "> </div>
```

*Affichage de la valeur du canal n°1*

2. Envoyer des données de la page HTML à la carte Arduino



Pour la page HTML, il faut utiliser la syntaxe suivante :

HTML

```
<div class="ard [element] [pin] [parameters]"></div>
```

Pour l'Arduino : **involtReceive()**;

Exemple pour envoyer une valeur numérique via un clic sur un bouton pour actionner une LED reliée à la broche 5 :

Page HTML	Arduino
<p><i>P5 = '1' si on clique sur le bouton "Click me"</i></p> <p>HTML</p> <pre>&lt;button class="ard button P5 value-123"&gt;Click me&lt;/button&gt;</pre>	<p><i>P5 est dans involtPin[5] ; envoi de la valeur involtPin[5] sur la broche numéro 3 (PWM)</i></p> <pre>void loop() {   involtReceive();   analogWrite(3, involtPin[5]); }</pre>

Reception d'une valeur dans Arduino

Avec la fonction **involtReceive()**, la valeur numérique reçue est stockée dans le tableau : **involtPin[pinIndex]** ; **pinIndex** correspond au nombre indiqué lors de l'envoi précédé de la lettre P : P5 dans notre exemple.

Exemple pour envoyer un nombre via une fenêtre pour modifier une valeur du programme l'Arduino

Page HTML	Arduino
<p><i>demande de saisie d'un nombre</i></p> <pre>&lt;input type="number" class="ard involt-input P5"&gt;</pre>	<p><i>valeur devient le nombre saisi</i></p> <pre>valeur = involtPin[5];</pre>

## Généralités : Pins & Arduino sketch

Afin de coller au plus près avec le monde Arduino, Involt utilise des broches virtuelles (**PINS**) : Il existe trois types de broches : **P**, **S** et **A**.

Il est possible d'en créer autant que l'on veut. Ce sont des variables



### Quand Involt envoie des valeurs à la carte Arduino

HTML

```
<div class="ard [element] [pin] [parameters]"></div>
```

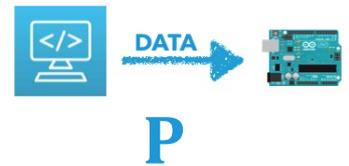
Involt peut envoyer deux types de données :

Pour les **valeurs numériques** depuis Involt, il faut utiliser le préfixe **P**

Les données sont enregistrées dans le tableau `involtPin[]`.

Exemple : Un nombre envoyé avec l'index **P6** sera stocké dans la ligne 6 du tableau `involtPin`

```
involtPin[6].
```



Pour les **chaînes de caractères**, il faut utiliser le préfixe **S**.

Les données sont enregistrées dans le tableau `involtString[]`.

Exemple : Un texte envoyé avec l'index **S8** sera stocké dans la ligne 8 du tableau `involtString`

```
involtString[8].
```



### Quand la carte Arduino envoie des données à Involt

Il existe deux fonctions pour envoyer des données à l'application :

`involtSend(pin, value)` et `involtSendString(pin, string)`.

Les nombres et textes sont stockés dans l'application comme **PIN** de type **A**.

Par exemple : La pin **A0** peut être un nombre `123` ou du texte `word`.

Pour afficher les éléments sur la page HTML il suffit d'écrire les lignes suivantes :



Page HTML	Arduino
<p><i>Affichage des données A0 (canal n°0) et A1 (canal n°1)</i></p> <pre>&lt;div class="ard show A0"&gt;&lt;/div&gt; &lt;div class="ard show A1"&gt;&lt;/div&gt;</pre>	<pre>// Voir commentaires ... void loop(){   involtReceive();   // Lecture analogique du port A0 et envoi sur   // le canal n°0   involtSend(0, analogRead(A0));   delay(2);   // Envoi du mot involt sur le canal n°1   involtSendString(1, "involt");   delay(8); }</pre>

**Rappel : Après la modification des programmes, POUR VALIDER LA COMMUNICATION USB et lancer la page html il faut exécuter le programme nw.**