



HomePage  
Référence  
Générateur de Code  
Vidéos

ACCUEIL

**ARDUINO**

> S'installer  
> Débuter  
> Approfondir  
> Tous les programmes  
> **ATELIERS ARDUINO**

**ATELIERS Python + Qt**

PROCESSING  
MATERIEL  
OUTILS  
TESTS  
Plus...

**VIDEOS**

COMPRENDRE  
REFERENCES  
CODER  
TECHNIQUE  
GNU/LINUX  
LIENS  
Rien à voir

Atelier Arduino et Robotique  
de Ploemeur

Ma Boutique virtuelle  
Toutes les pages

..

**Vous aimez ce site ?  
Soutenez-le !**

**Faire un don**



Par compte Paypal ou carte bancaire  
**Vous pouvez donner dès 1€ !**

## Recevoir une chaîne de caractères depuis le PC via le port série USB

Par X. HINAULT - Page créée le : 29/1/2011.

```

/dev/ttyACM0
Arduino| Send
Ascii Caractere 1 = 65
Ascii Caractere 2 = 114
Ascii Caractere 3 = 100
Ascii Caractere 4 = 117
Ascii Caractere 5 = 105
Ascii Caractere 6 = 110
Ascii Caractere 7 = 111
Ascii Caractere 8 = 10
Saut de ligne recu
Chaîne recue=Arduino
Autoscroll Newline 115200 baud

```

Sur cette page... (Masquer)

1. Présentation
2. Matériel Nécessaire
  - 2.1 L'espace de développement Arduino
  - 2.2 Le matériel suivant pour réaliser le montage associé
3. Instructions de montage
4. Le schéma théorique du montage
5. Le circuit du montage
6. Fonctionnement du programme
  - 6.1 Structure globale du programme
  - 6.2 Déroulement du programme
7. Explication du programme
8. Mise en oeuvre du programme
  - 8.1 Préparation du montage et programmation de la carte Arduino :
  - 8.2 Préparation du Terminal côté PC dans le logiciel Arduino
  - 8.3 Fonctionnement
9. Le programme complet en langage Arduino

### 1. Présentation

Le programme reçoit une chaîne de caractère se terminant par un saut de ligne (depuis le Terminal série ou une interface Processing) sur le port série USB. Cette chaîne reçue est affichée dans le Terminal. Ce programme utilise la nouvelle classe String du langage Arduino.

Ce programme utilise les fonctionnalités suivantes :

- Utilise la connexion série vers le PC

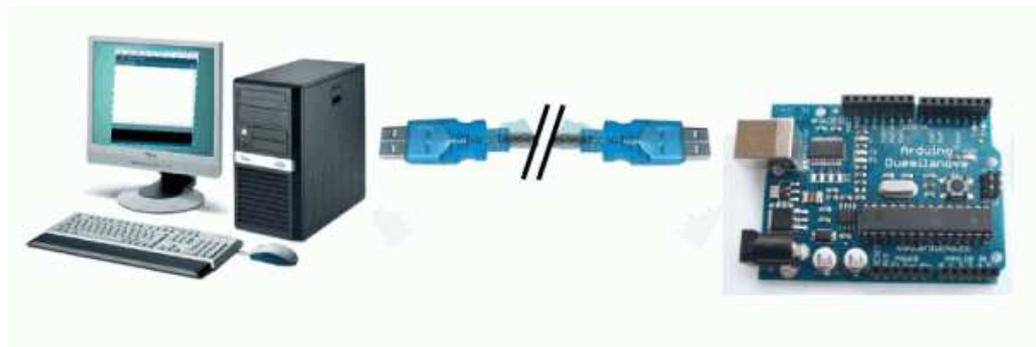
Ressources utiles associées à ce programme :

- La [librairie Serial](#) - pour les communications séries entre la carte Arduino et l'ordinateur ou d'autres composants

### 2. Matériel Nécessaire

#### 2.1 L'espace de développement Arduino

- ... pour éditer, compiler le programme et programmer la carte Arduino.



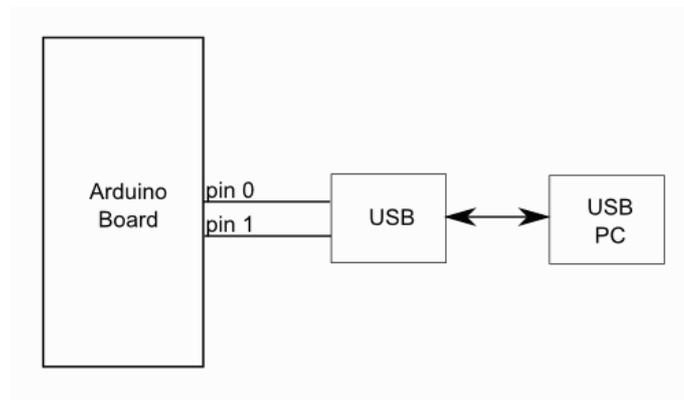
## 2.2 Le matériel suivant pour réaliser le montage associé

- aucun matériel supplémentaire.

## 3. Instructions de montage

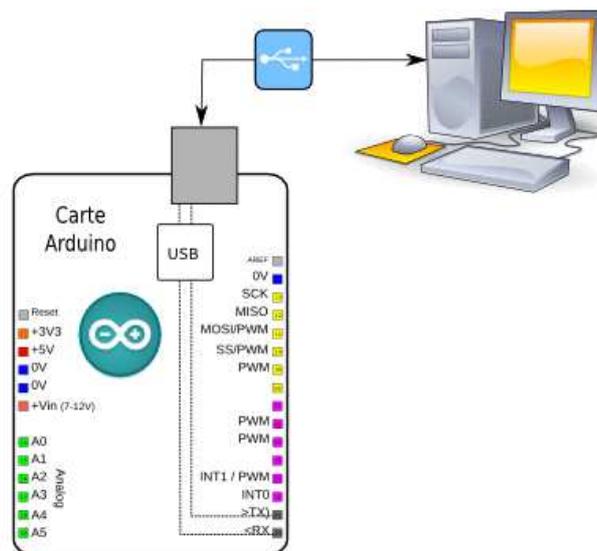
- La connexion série vers le PC utilise les broches 0 et 1 (via le câble USB)

## 4. Le schéma théorique du montage



Le schéma théorique du montage (cliquer pour agrandir)

## 5. Le circuit du montage



Le schéma du montage à réaliser (cliquer pour agrandir)

## 6. Fonctionnement du programme

### 6.1 Structure globale du programme

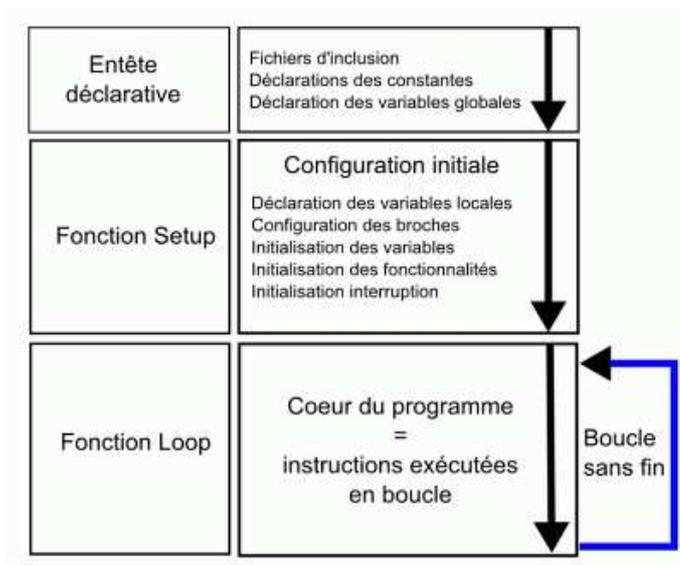
Ce programme simple comprend :

- une entête déclarative
- une partie « configuration » qui ne sera exécutée qu'une fois (fonction `setup()` )
- une partie constituée d'une boucle sans fin que le programme répètera à l'infini (fonction `loop()` ) : c'est le coeur du programme.

## 6.2 Déroulement du programme

Le programme se déroule de la façon suivante :

- Après avoir pris en compte les instructions de la partie déclarative,
- puis après avoir exécuté la partie configuration ( fonction `setup()` ),
- le programme bouclera sans fin ( fonction `loop()` ), exécutant de façon répétée le code compris dans la boucle sans fin.



## 7. Explication du programme

- Voir les commentaires du programme pour les explications.

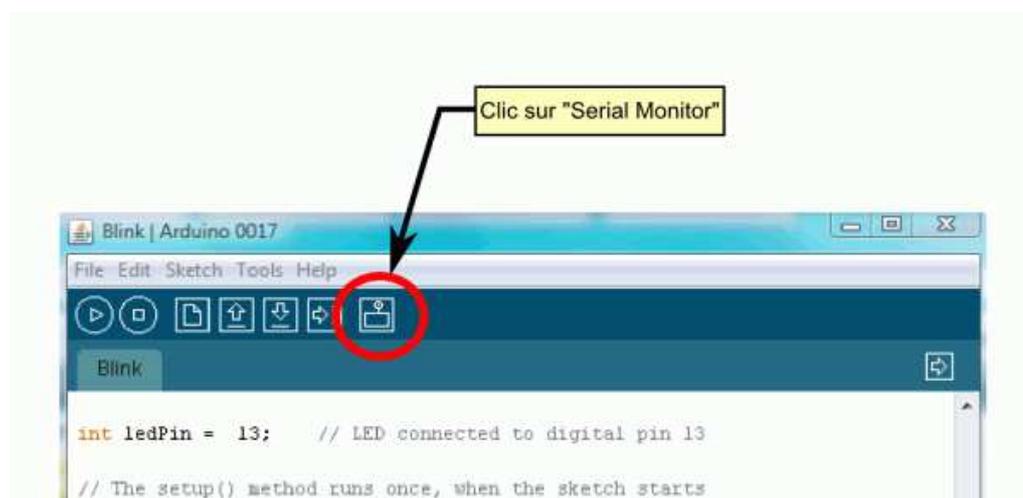
## 8. Mise en oeuvre du programme

### 8.1 Préparation du montage et programmation de la carte Arduino :

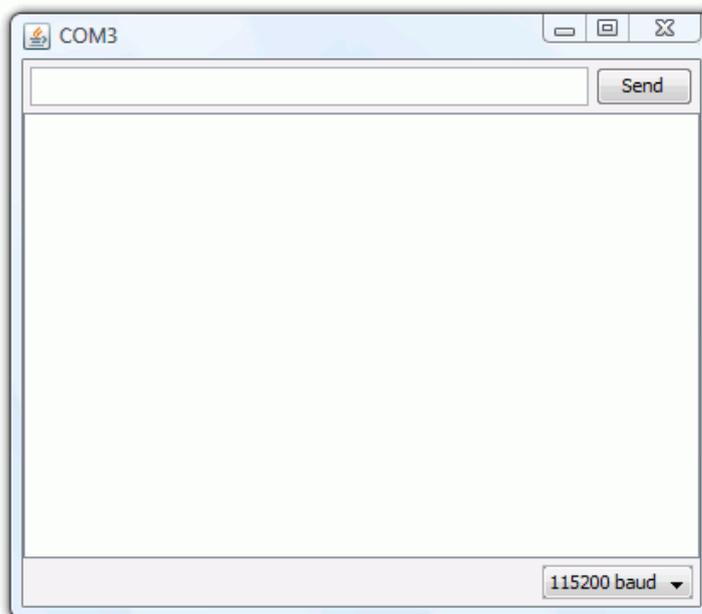
- Commencer par réaliser le montage indiqué sur plaque d'expérimentation
- Ensuite, programmer la carte Arduino avec ce programme (en bas de page) selon la procédure habituelle

### 8.2 Préparation du Terminal côté PC dans le logiciel Arduino

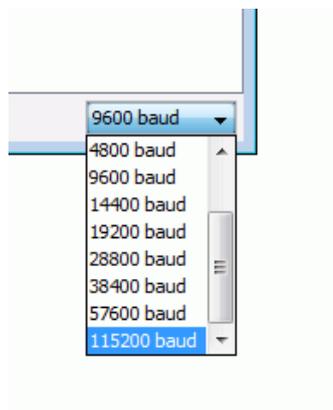
- Côté PC, il faut ouvrir la fenêtre terminal de l'IDE Arduino : pour ce faire, un simple clic sur le bouton « Sériel Monitor ».



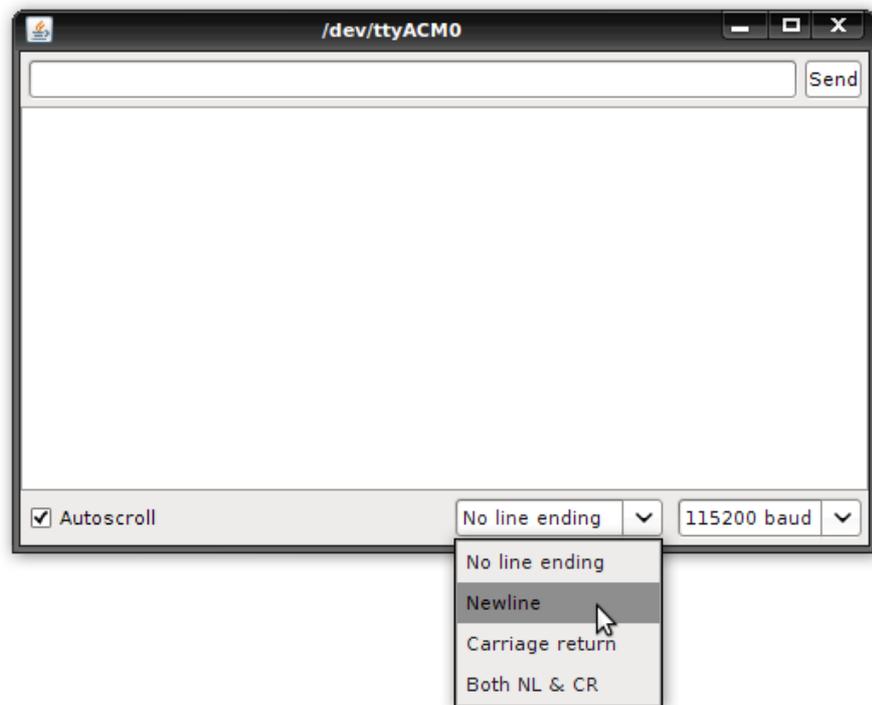
- La fenêtre « Terminal » s'ouvre alors :



- Il faut alors régler le débit de communication sur la même valeur que celle utilisée par le programme avec lequel nous allons programmer la carte Arduino :

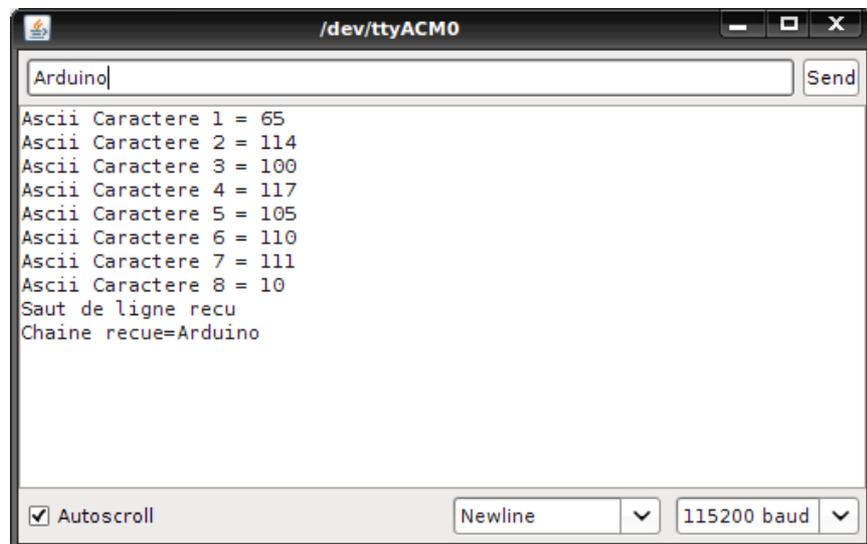


Point important, il faut également activer l'option "newline" qui active l'envoi du caractère "\n" (ascii 10) lorsque l'on saisi quelque chose dans le champ du Terminal suivi d'un <send>



### 8.3 Fonctionnement

- Saisir une chaîne dans le champ texte du Terminal suivi de envoi : la liste des codes ASCII des caractères de la chaîne recue s'affiche et la chaîne s'affiche également dans la fenêtre :



## 9. Le programme complet en langage Arduino

A copier/coller directement dans l'éditeur Arduino

```

// --- Programme Arduino ---
// Trame de code générée par le générateur de code Arduino
// du site www.mon-club-elec.fr

// Auteur du Programme : X. HINAULT - Tous droits réservés
// Programme écrit le : 29/1/2011.

// ----- Licence du code de ce programme -----
// This program is free software: you can redistribute it and/or modify
  
```

```
// it under the terms of the GNU General Public License as published by
// the Free Software Foundation, either version 3 of the License,
// or any later version.
// This program is distributed in the hope that it will be useful,
// but WITHOUT ANY WARRANTY; without even the implied warranty of
// MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
// GNU General Public License for more details.
// You should have received a copy of the GNU General Public License
// along with this program. If not, see <http://www.gnu.org/licenses/>.

// /////////////////////////////////// PRESENTATION DU PROGRAMME ///////////////////////////////////

// ----- Que fait ce programme ? -----
/* Le programme reçoit une chaîne de caractère se terminant par un saut de ligne
(depuis le Terminal série ou une interface Processing) sur le port série USB.
Cette chaîne reçue est affichée dans le Terminal. Ce programme utilise la
nouvelle classe String du langage Arduino. */

// --- Fonctionnalités utilisées ---

// Utilise la connexion série vers le PC

// ----- Circuit à réaliser -----

// La connexion série vers le PC utilise les broches 0 et 1 (via le câble USB)

// /////////////////////////////////// 1. Entête déclarative ///////////////////////////////////
// A ce niveau sont déclarées les bibliothèques incluses, les constantes, les
variables, les objets utiles...

// --- Déclaration des constantes ---

// --- Inclusion des bibliothèques ---

// --- Déclaration des constantes utiles ---

// --- Déclaration des constantes des broches E/S numériques ---

// --- Déclaration des constantes des broches analogiques ---

// --- Déclaration des variables globales ---

int octetReception=0; // variable de stockage des valeurs reçues sur le port
Série (ASCII)
char caractereRecu=0; // variable pour stockage caractère reçu
int compt=0; // variable comptage caractères reçus

String chaineReception=""; // déclare un objet String vide pour reception chaine

// --- Déclaration des objets utiles pour les fonctionnalités utilisées ---

// /////////////////////////////////// 2. FONCTION SETUP = Code d'initialisation
////////////////////////////////////
// La fonction setup() est exécutée en premier et 1 seule fois, au démarrage du
programme

void setup() { // debut de la fonction setup()

// --- ici instructions à exécuter 1 seule fois au démarrage du programme ---

// ----- Initialisation fonctionnalités utilisées -----
```

```

Serial.begin(115200); // initialise connexion série à 115200 bauds
// IMPORTANT : régler le terminal côté PC avec la même valeur de transmission

// ----- Broches en sorties numériques -----
// ----- Broches en entrées numériques -----

// ----- Activation si besoin du rappel au + (pullup) des broches en entrées
numériques -----

// ----- Initialisation des variables utilisées -----

} // fin de la fonction setup()
//
*****

////////////////////////////////////// 3. FONCTION LOOP = Boucle sans fin = coeur du
programme //////////////////////////////////
// la fonction loop() s'exécute sans fin en boucle aussi longtemps que l'Arduino
est sous tension

void loop(){ // debut de la fonction loop()

// --- ici instructions à exécuter par le programme principal ---

//---- code type réception valeur sur le port série ---

while (Serial.available()>0) { // tant qu'un octet en réception

    octetReception=Serial.read(); // Lit le 1er octet reçu et le met dans la
variable
    compt=compt+1;

    Serial.println("Ascii Caractere "+ String(compt) +" = "+
String(octetReception)); // affiche code ASCII Octet reçu

    if (octetReception==10) { // si Octet reçu est le saut de ligne

        Serial.println("Saut de ligne recu");
        Serial.println ("Chaine recue="+chaineReception); // affiche la chaine
recue

        chaineReception=""; //RAZ le String de réception
        compt=0; // RAZ compteur
        delay(100); // pause
        break; // sort de la boucle while

    }
    else { // si le caractère reçu n'est pas un saut de ligne

        caractereRecu=char(octetReception); // convertit l'octet reçu en
caractère
        chaineReception=chaineReception+caractereRecu; // ajoute le caratère
au String
    }

} // fin tant que octet réception

//---- une fois que le saut de ligne est reçu, on sort du While et on se
positionne ici

```

```
} // fin de la fonction loop() - le programme recommence au début de la fonction
loop sans fin
//
*****

// //////////////////////////////////////// FONCTIONS DE GESTION DES INTERRUPTIONS
//////////////////////////////////////

// //////////////////////////////////////// AUTRES FONCTIONS DU PROGRAMME ////////////////////////////////////////

// //////////////////////////////////////// Fin du programme ////////////////////////////////////////
```

[\[Obtenir le Code\]](#)

View Edit Attributes History Attach Print Search - Page last modified on April 06, 2011, at 10:42 AM