TP : WEBCAM MOTORISEE

Nom des élèves du groupe :	Classe :	Date :

Au cours de ce TP, vous allez découvrir :

- Quels sont les éléments générateurs de mouvement de la Webcam ;
- Quel est le principe de fonctionnement d'un moteur pas à pas et comment le piloter ?
- Comment réaliser un interface graphique de pilotage de la Webcam sous Visual Basic.

Matériel à disposition :	1 platine Arduino 1 Shield Motor R3
Logiciel :	Arduino Visual Basic Express
Documents ressources :	ftp://172.16.123.58 Répertoire : /2ND/TPSystèmes/TP9.Webcam login : eleve / mot de passe : eleve

I. ETUDE PRELEMINAIRE

✎ Quelle est la fonction principale de la Webcam motorisée ?

✎ Dans quel contexte est-elle utilisée ? (loisir ? Professionnel ?)

✎ Quels en sont les utilisateurs principaux ?

✎ En vous aidant de la documentation technique et des indications données sur la boîte d'emballage de la Webcam, donner les caractéristiques principales. (Amplitude de mouvement, résolution d'image, consommation électrique,…)

II. ETUDE DES MOUVEMENTS DE LA WEBCAM

✎ Donnez la signification des termes AZIMUT et ELEVATION.

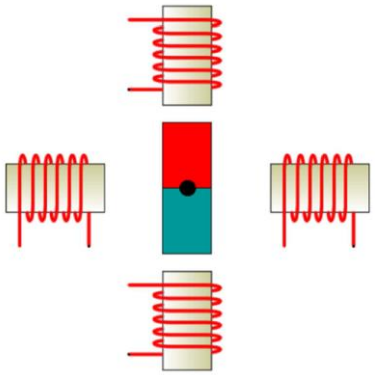
🖱 Vérifier les valeurs dans angles de rotation données dans la documentation technique en faisant des mesures expérimentales sur le mécanisme à l'aide d'un rapporteur. Les valeurs sont-elles identiques ?

III. ETUDE DU MOTEUR PAS A PAS

Un moteur pas à pas comprend un rotor aimanté placé au milieu de plusieurs électroaimants qui sont alimentés successivement.

Le pôle du rotor aimanté vient s'aligner avec le pôle magnétique créé par l'électroaimant ce qui provoque sa mise en rotation.

🖱 Visualisez la vidéo « Fonctionnement MPP.avi » disponible dans le dossier RESSOURCES du TP présentant une animation du fonctionnement d'un moteur pas à pas.



IV. PILOTAGE DE LA WEBCAM (AZIMUT) PAR PROGRAMMATION

Cette partie est consacrée à la réalisation d'une interface graphique permettant de piloter la base de la Webcam. Deux programmes sont nécessaires :

- L'un sur Arduino pilotera le moteur pas à pas ;
- Tandis que l'autre sous Visual Basic enverra à la platine Arduino les commandes de mise en rotation de la Webcam.

1. Programmation sous Arduino

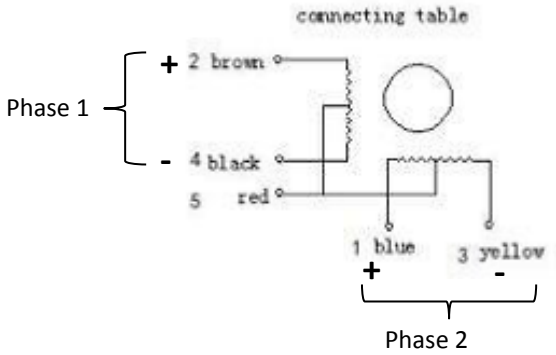
Programmation de la mise en rotation d'un moteur pas-à-pas dans les deux sens

On se propose de piloter la base de la Webcam à partir d'une platine Arduino qui sera programmée en langage C.

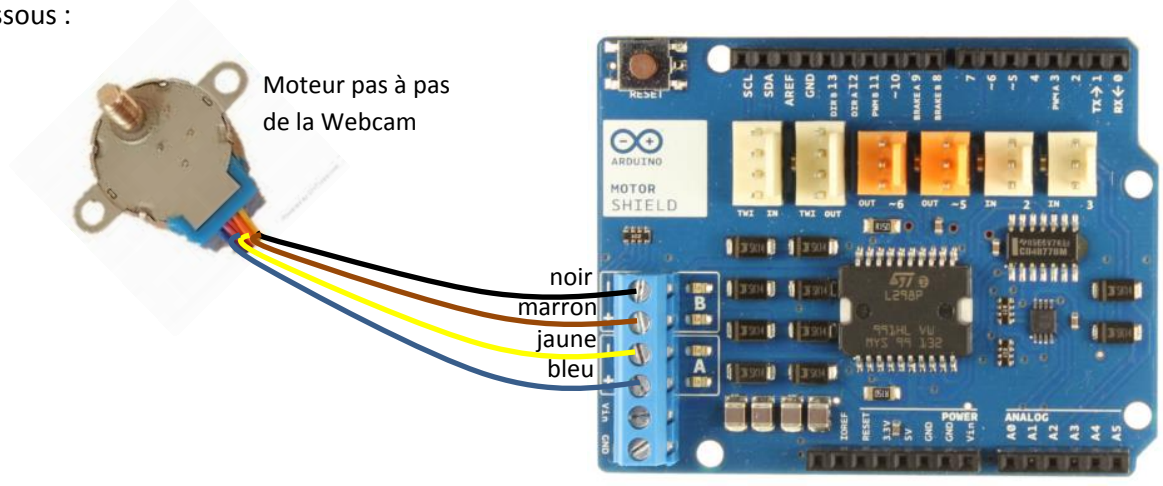
Afin de faciliter la mise en œuvre du programme, on utilisera une librairie <stepper.h> qui contient des macros pré-écrites de pilotage d'un moteur pas à pas.

Le moteur pas à pas de la caméra sera relié à un « Shield Arduino » dédié au pilotage des moteurs. Le moteur pas à pas de la base est un moteur 48 pas/tour.

Chaque phase du moteur pas à pas sera reliée de la manière suivante :

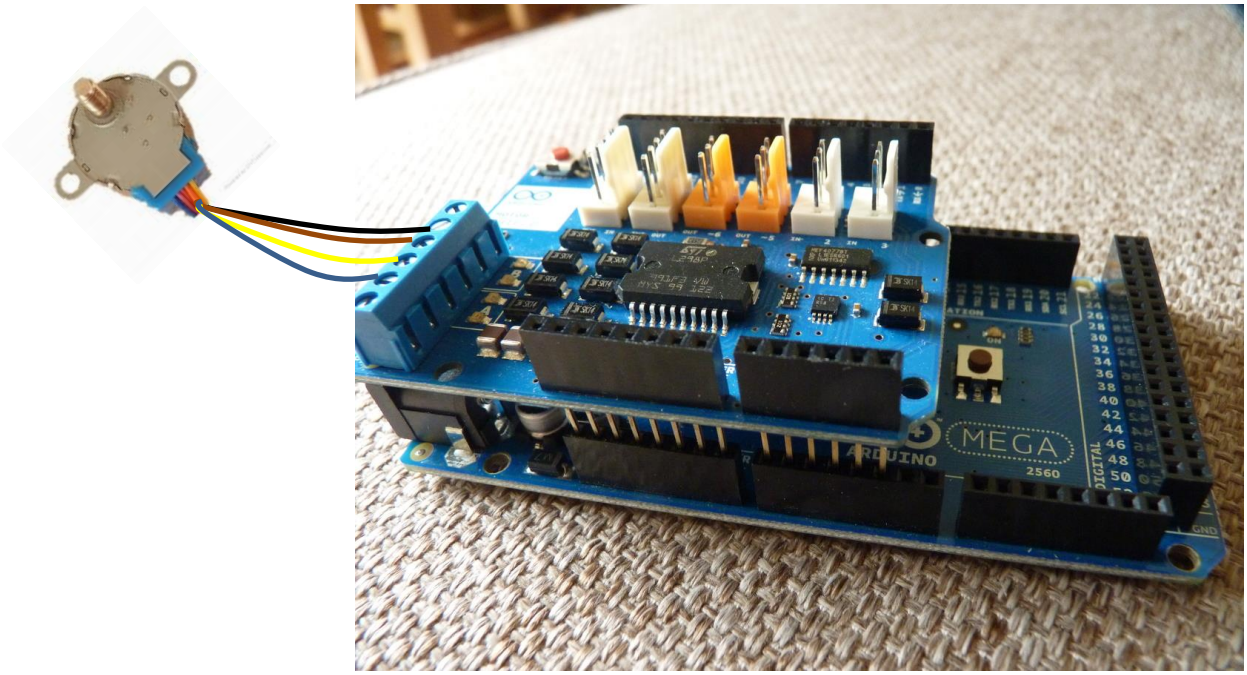


Pour cela, relier le moteur pas à pas de la caméra au shield Arduino « MOTOR SHIELD R3» comme indiqué ci-dessous :

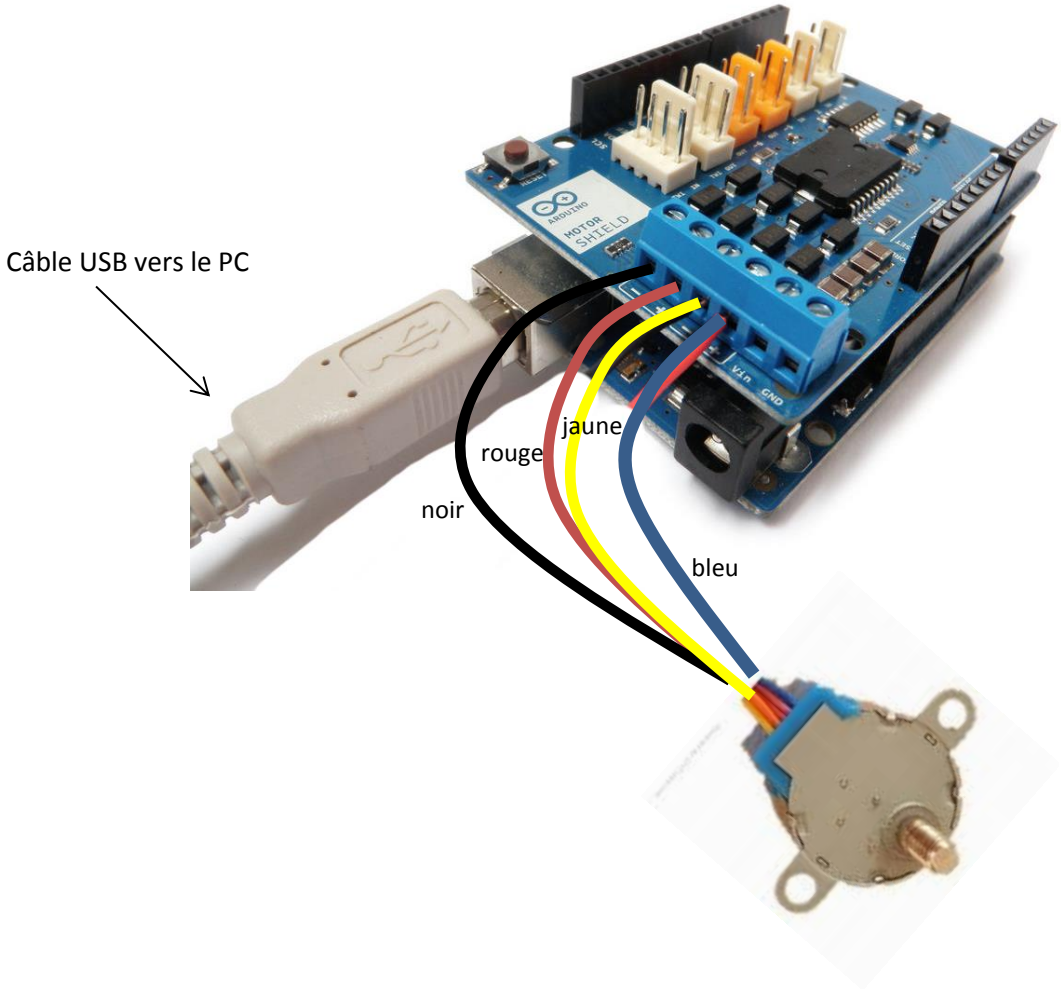


Shield Motor Arduino R3

Reliez maintenant le schield moteur à la plaquette Arduino MEGA comme ci-dessous :

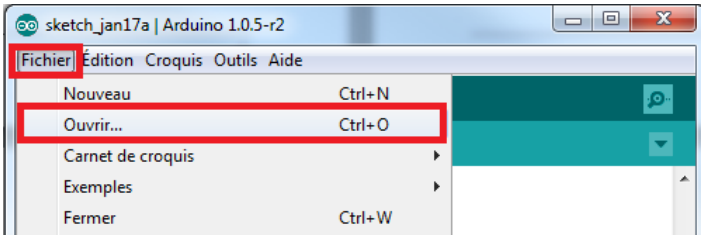


Reliez la platine Arduino MEGA au PC en utilisant le câble USB fourni et lancez le logiciel Arduino.

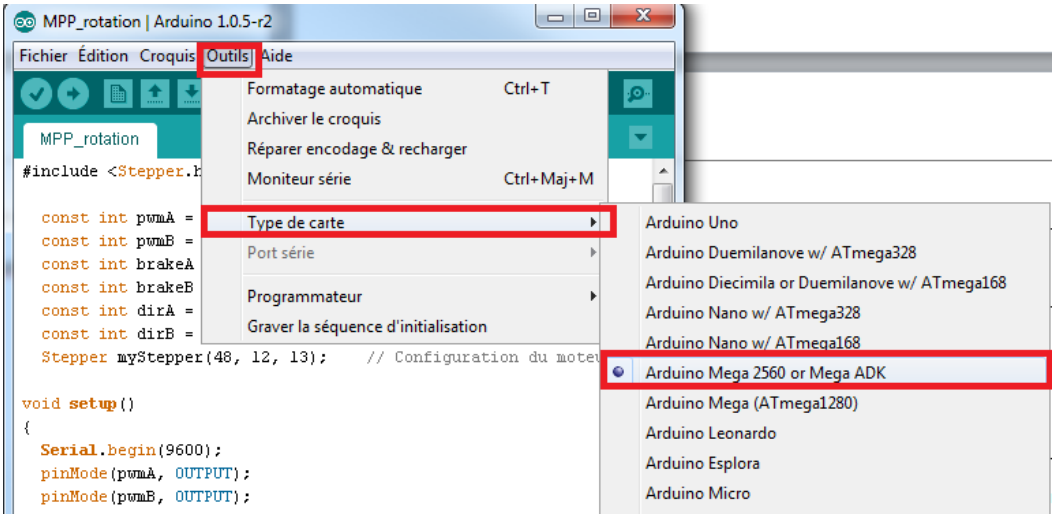


Lancer le programme « Arduino.exe » Arduino (disponible sur le bureau ou dans le menu démarrer)

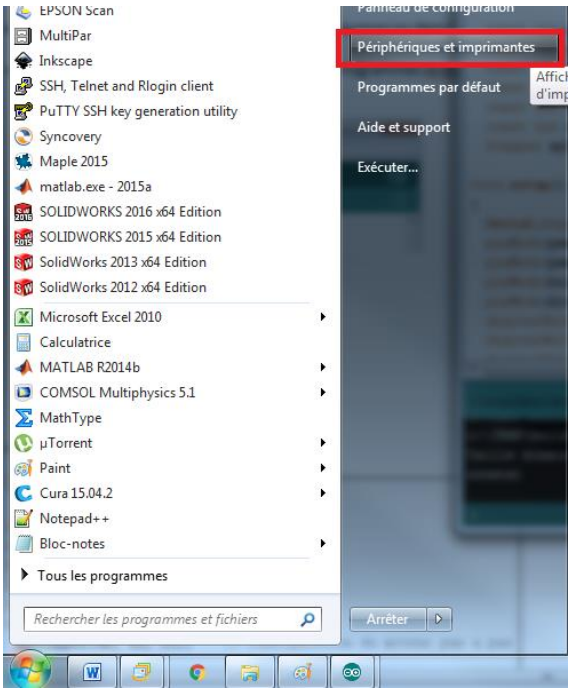
Ouvrir le programme « MPP_rotation » disponible dans le dossier « Programmes Arduino\MPP Rotation » des ressources :

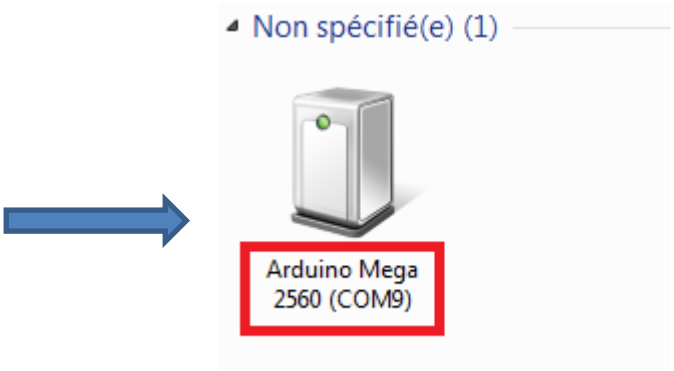


Configurer le programme pour utiliser des cartes Arduino de type « Arduino MEGA 2560 » :

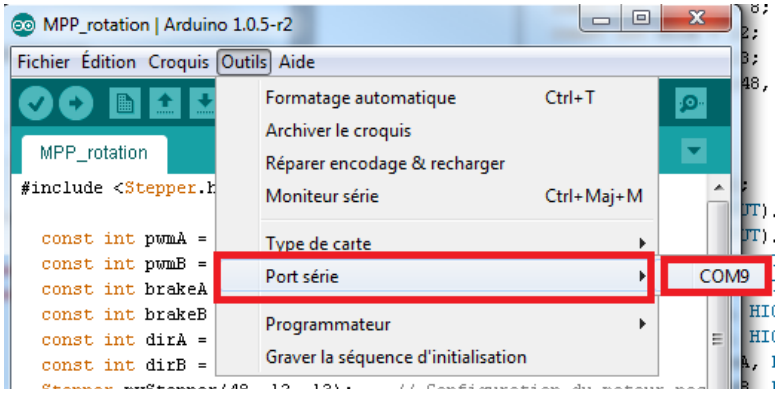


Aller dans le gestionnaire des périphériques afin d'identifier le numéro de port USB de l'ordinateur sur lequel est branché le kir Arduino :

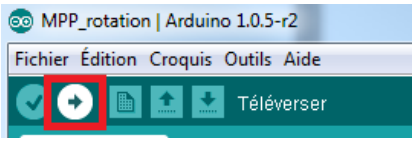




Sélectionner dans le programme le numéro du port série correspondant :



Exécuter le programme en appuyant sur l'icône « Téléverser »



⇨ La caméra devrait tourner sur elle-même (lentement).

```
#include <Stepper.h>

  const int pwmA = 3;
  const int pwmB = 11;
  const int brakeA = 9;
  const int brakeB = 8;
  const int dirA = 12;
  const int dirB = 13;
  Stepper myStepper(48, 12, 13);     // Configuration du moteur pas a pas

void setup()
{
  Serial.begin(9600);
  pinMode(pwmA, OUTPUT);
  pinMode(pwmB, OUTPUT);
  pinMode(brakeA, OUTPUT);
  pinMode(brakeB, OUTPUT);
  digitalWrite(pwmA, HIGH);
  digitalWrite(pwmB, HIGH);
  digitalWrite(brakeA, LOW);
  digitalWrite(brakeB, LOW);
}

void loop() {

  myStepper.step(1);        // rotation de 1 pas
  delay(100);               // pause de 100ms
}
```

Remplacer l'instruction myStepper.step(1) par myStepper.step(-1). Qu'observez-vous ?

Modifiez la valeur du delay : remplacer delay(100) par delay(800). Sur quoi cela agit-il ?

Choix du sens de rotation par réception d'un caractère sur le port série

Le programme « Recept_char_Portserie » permet la commande de l'allumage ou l'extinction de la LED connectée sur la broche 13 de la platine Arduino en fonction du caractère reçu sur le port série.

Port Série : Le microcontrôleur présent sur les platines Arduino UNO possède 2 broches de communication série Rx (Réception) et Tx (Transmission). Ces 2 broches, appelées « port série », permettent à l'Arduino de communiquer avec l'ordinateur lorsqu'il est relié par un câble USB à la platine.

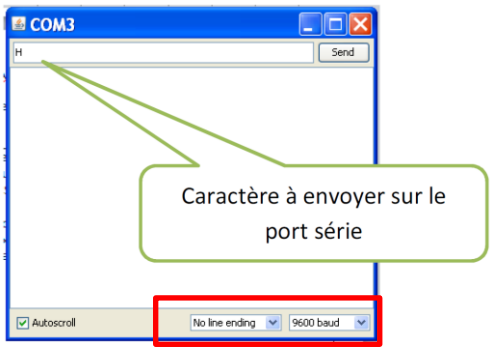
Ouvrir le programme « Recept_char_PortSerie » sous Arduino.

```
char val; // Data received from the serial port

void setup() {
pinMode(13, OUTPUT); // Set pin 13 as OUTPUT
Serial.begin(9600); // Start serial communication at 9600 bps
}

void loop() {
if (Serial.available()) { // If data is available to read,
val = Serial.read(); // read it and store it in val
}
if (val == 'H') { // If H was received
digitalWrite(13, HIGH); // turn the LED on
} else {
digitalWrite(13, LOW); // Otherwise turn it OFF
}
delay(100); // Wait 100 milliseconds for next reading
}
```

Allumage de la LED 13 si réception du caractère 'H'

Extinction sinon

Ouvrir le moniteur série en cliquant sur l'icône Moniteur série en haut à droite et testez le programme en envoyant alternativement les caractères « H » et « G ».



Le moniteur série permet de visualiser toutes les données qui transitent entre la platine Arduino et le PC via le câble USB.

Pour envoyer une donnée à l'Arduino, le caractère « H » par exemple, il suffit de tapez la lettre « H» dans la fenêtre du haut et de cliquer sur le bouton Send.

Attention : Il faudra régler le paramètre d'envoi de caractères sur « Pas de fin de ligne ».

En utilisant les 2 programmes précédents, compléter le programme ci-dessous permettant :

- La rotation horaire de 200 pas si réception du caractère H
- La rotation anti-horaire de 200 pas si réception du caractère G
- Aucune action si réception d'un caractère autre ;
- Une vitesse de rotation du moteur pas à pas réglée à 100 pas/sec.

Sauvegardez le programme sous « pilotage_webcam_portserie »

Programme à compléter et à tester :

```
#include <Stepper.h>

const int pwmA = 3;
const int pwmB = 11;
const int brakeA = 9;
const int brakeB = 8;
const int dirA = 12;
const int dirB = 13;
Stepper myStepper(48, 12, 13);    // Configuration du moteur pas a pas

char val; // Data received from the serial port

void setup()
{
  Serial.begin(9600); // Start serial communication at 9600 bps
  pinMode(pwmA, OUTPUT);
  pinMode(pwmB, OUTPUT);
  pinMode(brakeA, OUTPUT);
  pinMode(brakeB, OUTPUT);
  digitalWrite(pwmA, HIGH);
  digitalWrite(pwmB, HIGH);
  digitalWrite(brakeA, LOW);
  digitalWrite(brakeB, LOW);
  myStepper.setSpeed(100);
}

void loop()
{
  if (Serial.available()) { // If data is available to read,
     val = Serial.read(); // read it and store it in val
     }

  if (val == .....)
    {
        myStepper.step(....);      // rotation de 200 pas
        val=' ';
    }
else if (val== .....)
    {
        myStepper.step(....);      // rotation de -200 pas
        val=' ';
    }

delay(100); // Wait 100 milliseconds for next reading
}
```

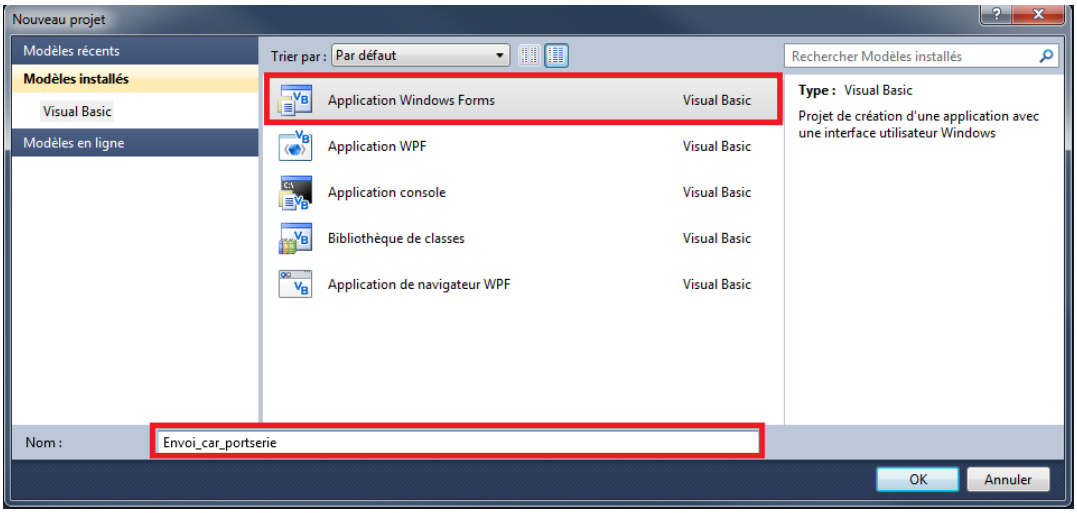
2. POUR ALLER PLUS LOIN - Programmation sous Visual basic 2010

On se propose dans cette partie de créer une interface graphique, capable de communiquer avec la platine Arduino afin de piloter la webcam motorisée.

Envoi d'un caractère sur le port série

Cette première interface devra être capable de se connecter à la platine Arduino via son port série et de lui envoyer le caractère 'H' lors de l'appui sur un bouton.

- Ouvrez le logiciel Visual Basic Express 2010 ;
- Cliquez sur « Fichier => Nouveau Projet » puis choisir « Application Windows Form » ;
- Nommer votre projet « Envoi_car_portserie » puis validez en cliquant sur OK

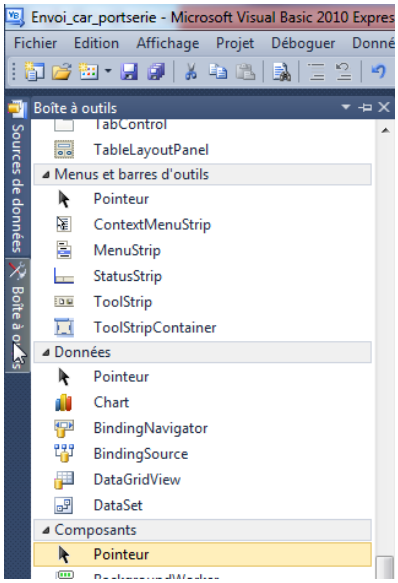


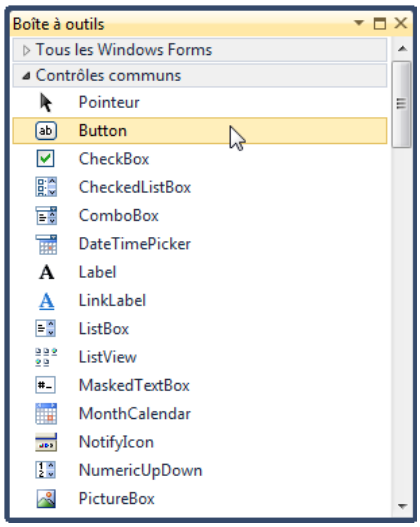
Les étapes nécessaires à la création d'un programme VB sont les suivantes :

- Placer les composants Windows nécessaires *(bouton, image, menu, case à cocher, timer…)*
- Paramétrer leur apparence dans la fenêtre de propriété à droite *(taille, couleur, visibilité ou non, position, texte…)*
- Associer à chaque composant un évènement en écrivant la fonction à réaliser lorsque l'évènement est produit.

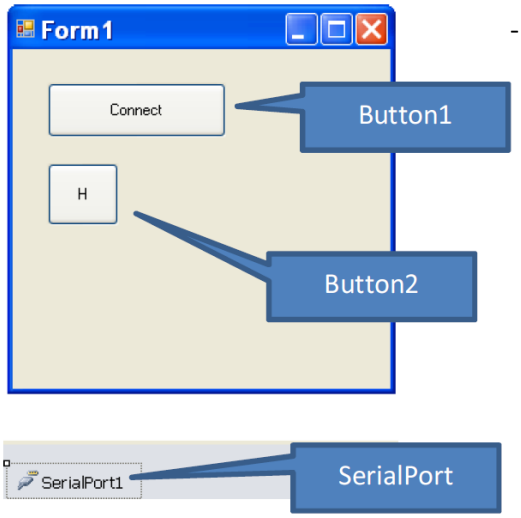
VB propose par défaut l'évènement le plus usité (ex : évènement Clic associé à un bouton) mais cela peut être modifié. Il faut double-cliquer sur le composant pour ouvrir la fenêtre de code

Pour placer les boutons, ouvrir la boite à outils à gauche de l'écran :

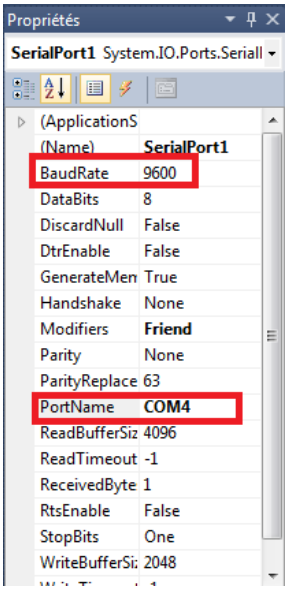




Réalisez l'interface ci-dessous :



- - Paramétrez le SerialPort1 en choisissant le même numéro de port que celui de l'Arduino.



Programmez les évènements liés aux boutons (en double-cliquant dessus) comme ci-dessous :

```
Imports System.IO.Ports

Public Class Form1

    Private Sub Button1_Click(ByVal sender As System.
        SerialPort1.Open()
        Button1.Enabled = False
    End Sub

    Private Sub Button2_Click(ByVal sender As System.
        SerialPort1.Write("H")
    End Sub
End Class
```

Import de la librairie System.IO.Port

Lors du **click** sur le bouton Connect :
Ouverture du Port série
Désactivation du bouton

Envoi du caractère « H » sur le Port série

Connecter la platine Arduino à l'ordinateur

Testez le programme en cliquant sur l'icône « Lecture » . **Attention :** *Appuyez d'abord sur « Connexion Port Série » avant d'appuyer sur le bouton « H ».*

Pilotage avancé de la Webcam

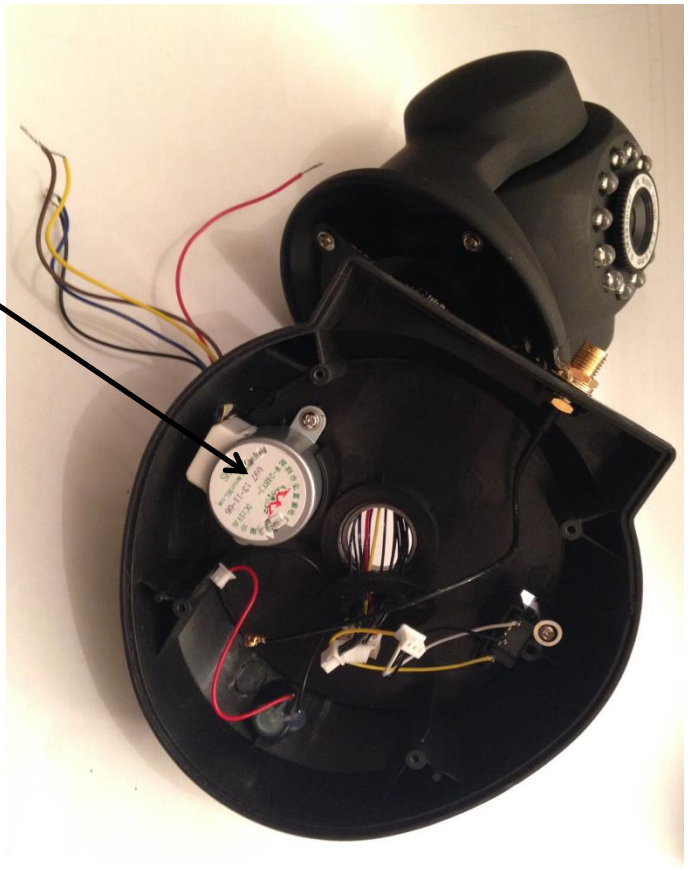
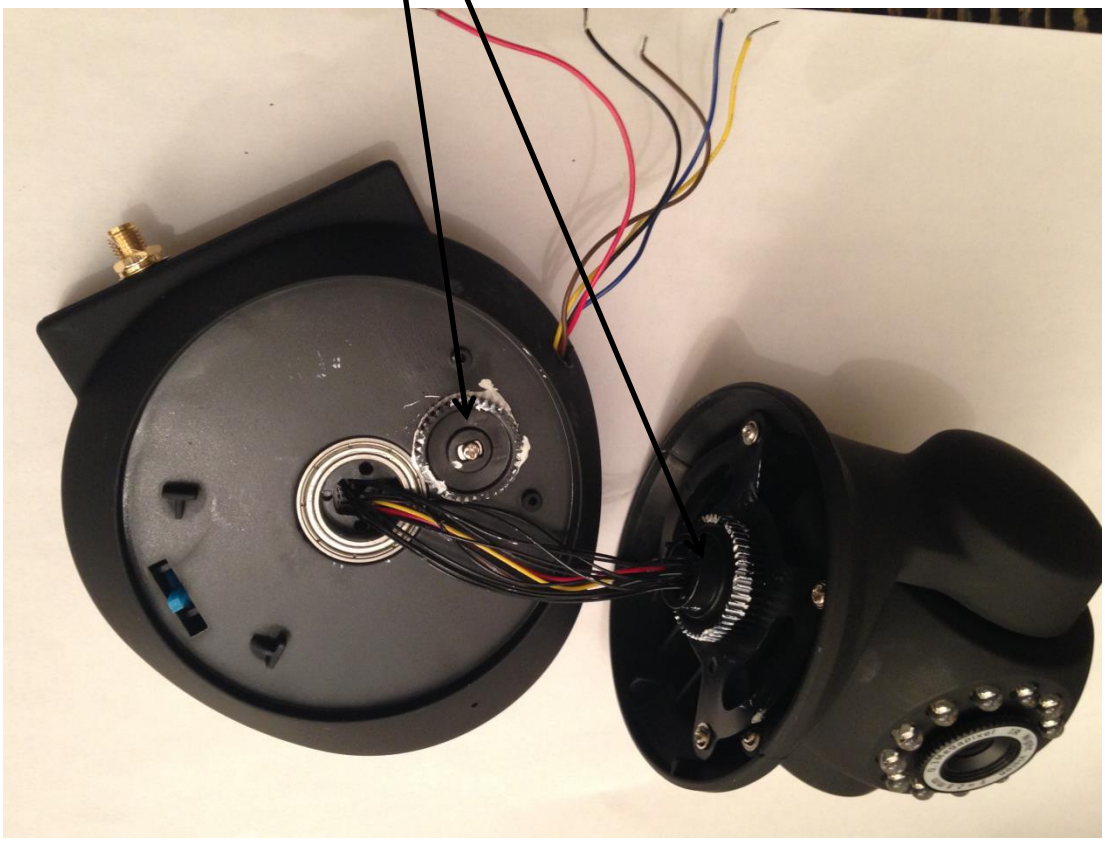
On souhaite réaliser un programme sous VB qui pilote la Webcam de la même façon que sur l'interface réelle, c'est-à-dire :

- Rotation continue dans le sens horaire lors de l'appui sur le bouton H ;
- Rotation continue dans le sens anti-horaire lors de l'appui sur le bouton G ;
- Arrêt lors de l'appui sur un bouton STOP.

Compléter l'interface graphique et le programme associé afin de pouvoir piloter la webcam comme demandé ci-dessus.

DOCUMENT RESSOURCE : VUE ECLATEE DE LA CAMERA

Moteur pas-à-pas de commande de l'AZIMUT



Engrenage commandé par le moteur pas-à-pas AZIMUT