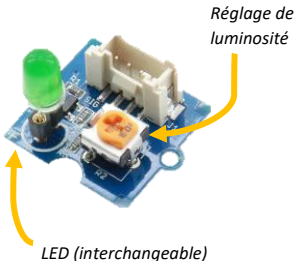
– Fiche pratique mBlock/Arduino–

Programmation d'une LED Grove

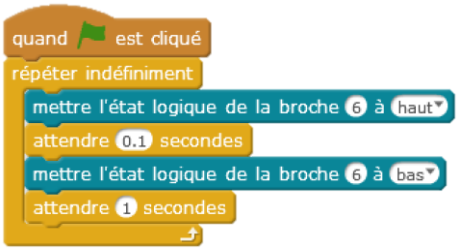
1- Constitution du module LED Grove

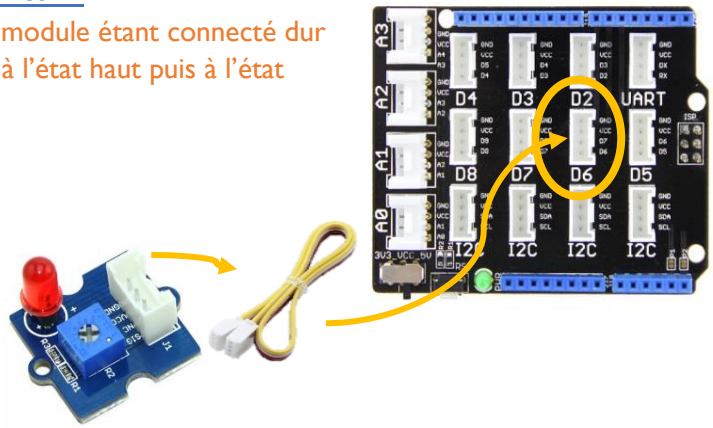


Les LED (Light Emiting Diode) ou DEL en français (Diode ElectroLuminescente) servent classiquement de voyant (marche/arrête, notification…). Son utilisation est classiquement faite en TOR (tout-ou-rien), on **l'actionne** en binaire. On reliera donc ce module sur l'une des sorties D2 à D8 de la carte Grove. A noter que ce module possède une résistance variable pour « polariser » plus ou moins la LED selon la couleur utilisée.

2- Montage et programme expérimental :

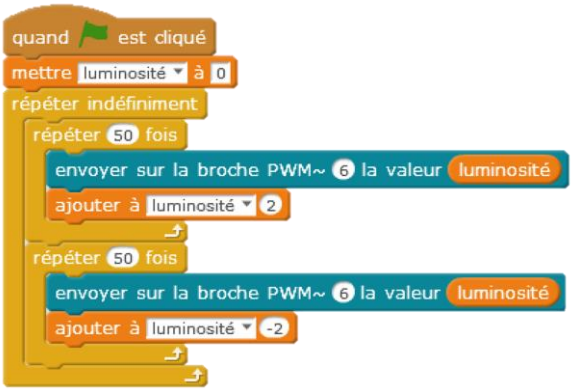
On peut difficilement imaginer plus simple… Le module étant connecté dur D6, on pilotera la sortie digitale 6 en la plaçant à l'état haut puis à l'état bas, de façon infinie, la diode va clignoter.



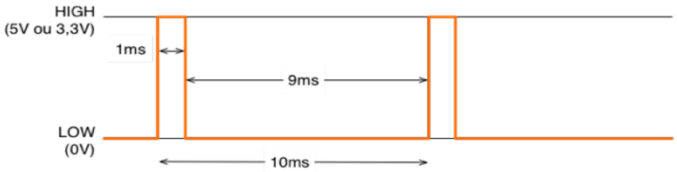


3- Pour aller plus loin :

Nous pouvons faire varier la luminosité de la LED en l'allumant puis l'éteignant puis en l'allumant…suffisamment vite pour ne pas la voir scintiller. Plus le temps d'allumage sera grand face à celui d'extinction, plus la LED semblera lumineuse. C'est ce que l'on appelle une commande par **Modulation de Largeur d'Impulsion** (MLI) ou PWM en anglais (Pulse Width Modulation). Une commande existe avec Arduino pour moduler les sorties numériques et les rendre ainsi « pseudo-analogiques ».



Grâce à ce programme l'intensité lumineuse de la LED va croître et décroitre alternativement : elle est modulée.



Exemple ici : la LED est allumée 1ms puis éteinte 9ms, sa luminosité sera faible. Le rapport cyclique vaut 1/10 soit 10%.
Avec la commande Arduino, 0 correspond à 0% de rapport cyclique (toujours à 0) et 255 à 100% de rapport cyclique (toujours à 1)

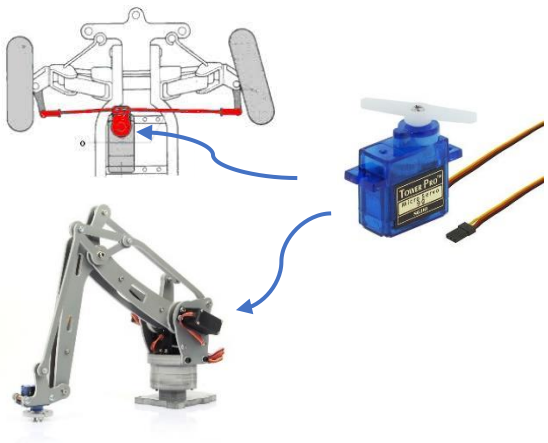
Cette commande « en largeur d'impulsion » MLI ou PWM est très utilisée pour faire varier la vitesse d'un moteur en robotique.

Attention : toutes les broches ne sont pas PWM, seules les broches 3, 5, 6, 9, 10 et 11 le permettent.

F. Prieur mai 2018

– Fiche pratique mBlock/Arduino–

Programmation d'un servomoteur



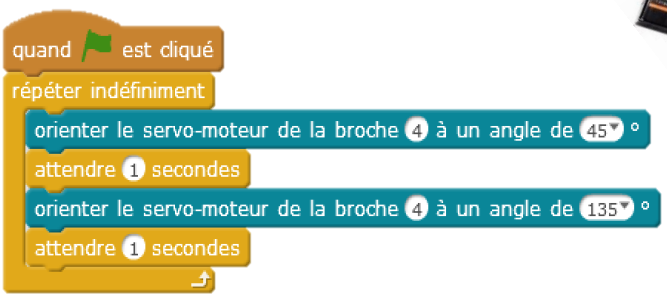
1- Qu'est-ce qu'un « Servomoteur »

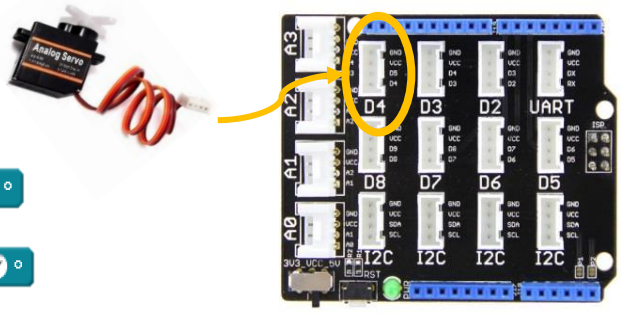
Un servo-moteur est un **actionneur** permettant de contrôler la position angulaire d'un dispositif. Il est très utilisé en modélisme et en robotique légère.

Remarque : selon les modèles choisis, la rotation peut se faire entre -180° et 180°, -45° et 45° ou bien encore sur plusieurs tours sans « buté »e » de fin de course. Le notre a un débattement de 0 à 180°.

2- Montage et programme expérimental :

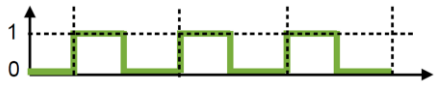
Ce programme de base positionne l'axe du servo à 45° une seconde puis à 135°une autre seconde, puis recommence sans fin. La base pour imaginer mille applications…



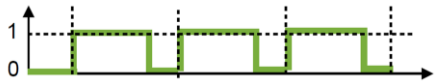


3- Pour aller plus loin :

Un servo-moteur est un élément à 3 broches : deux alimentations 5V/0V et un signal correspondant à la commande angulaire exprimée « en largeur d'impulsion » (MLI ou PWM « Pulse Width Modulation » en anglais).



-La commande est symétrique : le moteur est asservi à 0°



- La commande est asymétrique (ici ¾) : le moteur est asservi à ¾ de sa valeur angulaire maximale soit 135°

Ainsi, la commande du servo pourrait donc également se faire avec cette commande avec le même résultat :

Ce bloc pilote la sortie PWM (modulation de la largeur d'impulsion) de la sortie numérique de la broche D2 correspondant au connecteur D2 du shield Arduino™ Grove.

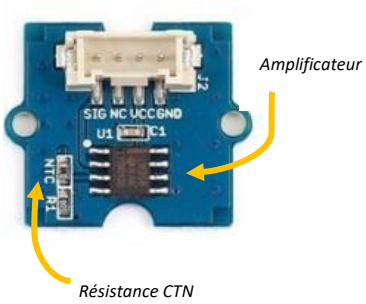
Pour nommer votre capteur, cliquer dans la zone de texte <Saisie libre> et taper le nom de votre actionneur.

Attention : toutes les broches ne sont pas PWM, seules les broches 3, 5, 6, 9, 10 et 11 le permettent.

F. Prieur mai 2018

– Fiche pratique mBlock/Arduino–

Utilisation du capteur de température Grove

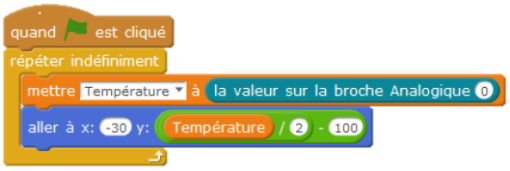


1- Qu'est-ce qu'un « capteur de température» ?

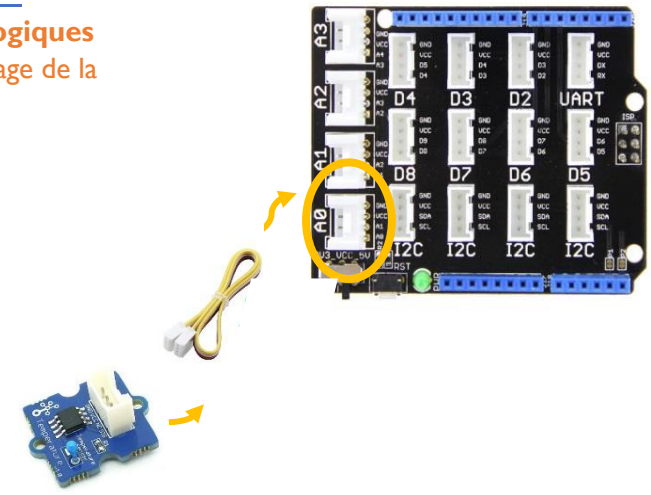
Ce module Grove est constitué d'une **résistance à coefficient de température négatif** (CTN) qui constitue le **capteur** de température et d'un amplificateur à circuit intégré LM358. Il suffit de l'alimenter et celui-ci fournit sur une broche une grandeur proportionnelle à la température mesurée. Sa plage de mesure s'étend de -40°C à +125°C avec une précision de 1,5°C.

2- Montage et programme expérimental :

Ce programme permet d'acquérir les **valeurs analogiques** présentes sur le connecteur A1du shield qui est l'image de la température de la CTN.



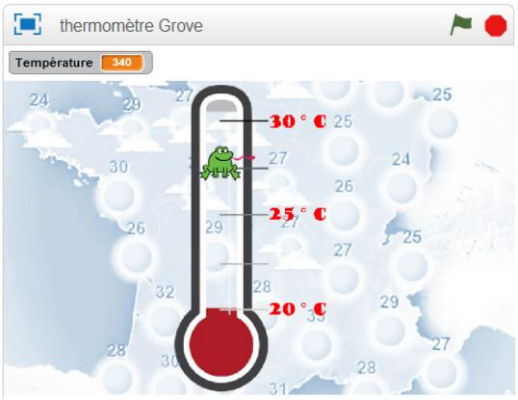
La température augmente ?... le lutin monte.



Remarque : la valeur renvoyée par le capteur (environ 340 à 380) n'est pas une mesure directe de la température mais l'image de celle-ci. Il faudrait pour l'exploiter trouver le décalage et le coefficient entre la valeur renvoyée et la température vraie : c'est l'**étalonnage**.

Remarque : Dans cet exemple « Température » est une variable créée pour le besoin grâce à la palette « Variables ». Conseil : pendant la mise au point, cochez la case correspondant pour afficher la mesure en temps réel dans la scène.

3- Pour aller plus loin :



1 -Télécharger le fond d'écran situé à https://bit.ly/2std84X .

2-Choisissez et redimensionnez un lutin « Grenouille ».

3- Trouvez le bon décalage horizontal (-30 dans l'exemple ci-dessus), vertical (-100 dans l'exemple ci-dessus) et le bon coefficient diviseur (2 dans l'exemple ci-dessus) pour que la grenouille évolue entre environ 25 et 30°C lorsque vous chauffez la CTN en la pinçant.

4- Affichez la scène en plein écran (icone bleue en haut à droite de la scène)

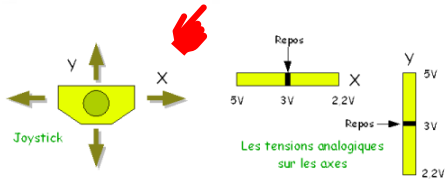
Vous venez de faire un thermomètre Arduino/mBlock !

F. Prieur mai 2018

– Fiche pratique mBlock/Arduino–

Utilisation du joystick Grove « 2 axes »

Remarque : le joystick a un « point de repos » (joystick « lâché » correspondant généralement à la demi-tension d'alimentation : il faudra probablement étalonner votre mesure…

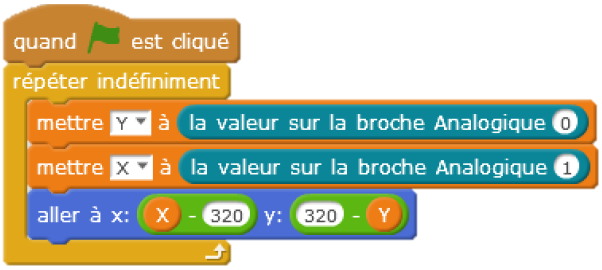


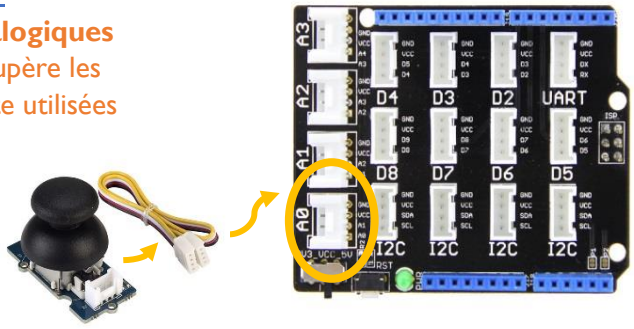
1- Qu'est-ce qu'un « joystick » ?

Un joystick est un **capteur** constitué de deux potentiomètres articulés sur une rotule. L'un est commandé par la poussée selon un axe (X), l'autre par la poussée sur l'axe orthogonal (Y). Il **mesure** donc la **déviation angulaire** selon deux axes. Il est classiquement utilisé dans les commandes de systèmes « deux axes »…ou pour bien dans les jeux vidéos 😉

2- Montage et programme expérimental :

Ce programme de permet d'acquérir les **valeurs analogiques** présentes sur le connecteur A1du shield. Celui-ci récupère les deux valeurs A0 (X) et A1(Y). Ces valeurs sont ensuite utilisées pour déplacer un lutin à l'écran…oui, c'est un jeu 😉

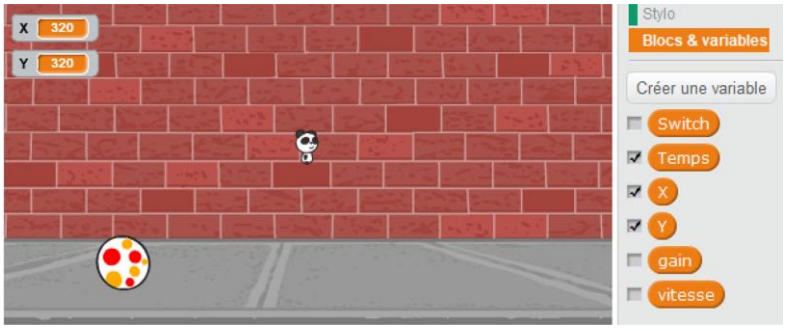




Remarque : Dans cet exemple X et Y sont deux variables créées pour le besoin (palette « Variables »)

3- Pour aller plus loin :

Le joystick est équipé de 4 fils : deux fils d'alimentation (5V et 0V) et deux fils pour les axes X et Y. Comment donc récupérer le « clic » sur le poussoir du joystick ? Il n'y a pas de fil dédié au bouton ! Etrange…



Cochez dans la palette « Variables » les deux variables utilisées X et Y, elles s'affichent alors dans la scène de Scratch.

Cliquez sur le joystick et observez les valeurs mesurées…comment utiliser cette particularité pour exploiter le « clic » ?

A vous de jouer…

F. Prieur mai 2018