

Chapitre 15 – Electronique programmable et réseaux de trains miniatures

Qu'est-ce que l'électronique programmable ?

Faire de l'électronique programmable, c'est faire appel à des composants électroniques qui sont **capables d'exécuter un petit programme pour réaliser une tâche bien précise**. On parle donc de composants programmables, et ils sont de deux types : le microprocesseur et le microcontrôleur.

Comme expliqué dans le chapitre 14, le microprocesseur n'est pas adapté à notre hobby car il est trop complexe à mettre en œuvre. On va donc l'oublier pour se concentrer sur le **microcontrôleur** qui est une sorte de « microprocesseur autonome » comprenant dans son boîtier tous les composants qu'il faudrait rajouter autour d'un microprocesseur.

Nous allons voir durant cette deuxième partie de cours que tout ce qu'il est possible de réaliser en électronique câblée, le microcontrôleur sait le faire. Cela permet de réduire le nombre de composants électroniques dans un montage en remplaçant ceux-ci par un programme informatique. Ainsi par exemple, la fabrication d'un oscillateur nécessite au moins trois composants en électronique câblée (un CI, une résistance et un condensateur) alors qu'en électronique programmable, cela ne nécessite qu'un petit programme de 4 lignes.

L'électronique programmable permet donc de réduire le coût d'un montage et offre une souplesse que n'a pas l'électronique traditionnelle ; si on veut changer la fréquence de notre oscillateur, il suffit de changer la valeur d'un paramètre plutôt que dessouder des composants pour les changer.

L'électronique programmable est-elle réellement moderne et simplifiée ? Moderne, sans aucun doute car elle est de plus en plus utilisée (dans votre voiture, dans votre smartphone, dans votre électroménager, etc.) ! Simplifiée, oui et non : pour certains, programmer paraît plus simple que calculer un montage, pour d'autres, ce sera l'inverse. Mais dans certains cas, l'électronique programmable se révèle plus simple à mettre en œuvre, les cas où il faut décider d'une action à partir de plusieurs options par exemple.

Des modules à base de microcontrôleurs

Malgré toutes leurs qualités, les microcontrôleurs ont mis du temps avant de s'imposer chez les électroniciens amateurs pour deux raisons :

- Il fallait disposer d'un programmeur pour transférer le programme dans le microcontrôleur et ceux-ci étaient en général assez onéreux.
- Le programme devait être rédigé dans un langage symbolique appelé assembleur pas toujours facile à maîtriser et encore moins à déboguer (corriger les erreurs de programmation).

Depuis quelques années, des modules composés d'un microcontrôleur et de tout ce qui est nécessaire pour le faire fonctionner, sont proposés aux amateurs bricoleurs pour un prix modique ; il suffit de les relier à un ordinateur pour transférer le programme qui est maintenant écrit dans un langage sophistiqué, proche du langage anglais.

Ces modules existent chez plusieurs constructeurs : le plus populaire est certainement le module Arduino (ou Genuino, absolument identique pour le marché Européen) mais il en existe d'autres comme le Raspberry par exemple.

Les deux principales difficultés d'utilisation des microcontrôleurs étant résolues, il n'y a plus aucune raison pour ne pas les considérer comme des composants électroniques comme les autres. Le but de ce cours est de vous faire découvrir les microcontrôleurs en utilisant des modules Arduino pour en

simplifier l'approche. Les principes enseignés sont transposables à d'autres types de modules (Raspberry par exemple). Pour ceux qui veulent aller plus loin, nous ne manquerons pas de décrire les entrailles du microcontrôleur pour que chacun comprenne ce qui se passe à l'intérieur (tout en restant simple quand même).

La figure 15.1 montre un module Arduino Uno. Il comporte un microcontrôleur entouré en jaune sur la figure ; il s'agit d'un microcontrôleur de marque Atmel et de référence ATmega328P. On voit également une prise USB qui permet, grâce à un câble USB, de raccorder le module à l'ordinateur pour programmer le microcontrôleur : elle est entourée en vert. Enfin, on voit toute une série de connecteurs entourés en rouge sur lesquels on se raccorde pour exploiter les différents signaux que traite le microcontrôleur.

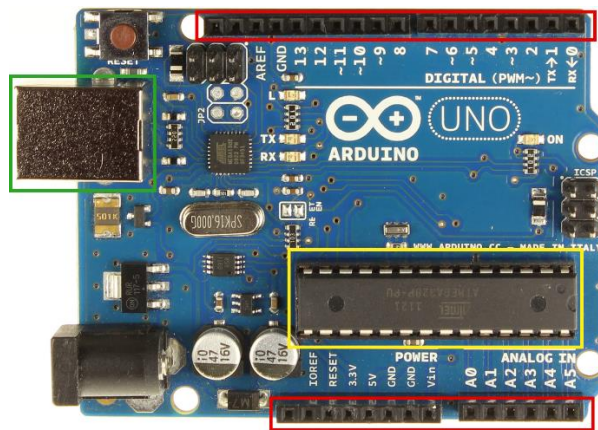


Figure 15. 1

Les connecteurs servent aussi à enficher des cartes appelées **shield** (bouclier en anglais car elles se mettent sur la carte module à la façon d'un bouclier protecteur) ; ces cartes offrent toute une panoplie d'applications pour ceux qui ne veulent pas faire d'électronique du tout, une sorte de solution toute faite. Leurs connecteurs mâles s'enfichent dans les connecteurs femelles de la carte Arduino, ce qui permet de récupérer tous les signaux sur la carte shield et de les exploiter. La figure 15.2 montre l'exemple de la carte shield LearnCbot vendue pour le cours de l'Ecole Polytechnique Fédérale de Lausanne, qu'on peut suivre sur internet (on appelle cela un **MOOC** Massive Open Online Course) ; la carte est composée entre autre de deux boutons poussoirs, et quatre LED avec résistances associées, et permet ainsi de simuler de nombreuses applications des microcontrôleurs.

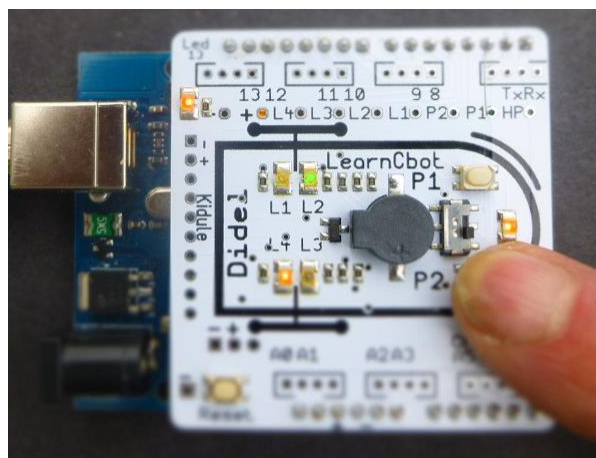


Figure 15. 2

Interfaçage d'un module Arduino avec un réseau de trains miniatures

Peut-on penser que les modules Arduino vont nous permettre de nous passer des composants électroniques traditionnels ? Pas tout à fait quand même, mais ceux-ci seront en nombre restreint puisqu'on remplace des fonctions électroniques complexes par des lignes de programme qui réalisent la même chose. Il faut tout de même savoir que le module Arduino ne délivre pas un courant très fort et que ses signaux doivent être amplifiés avant de pouvoir agir sur le réseau. De même, les signaux qui sont envoyés à l'Arduino pour être traités, doivent respecter certaines spécifications ; il est donc nécessaire de les adapter dans certains cas.

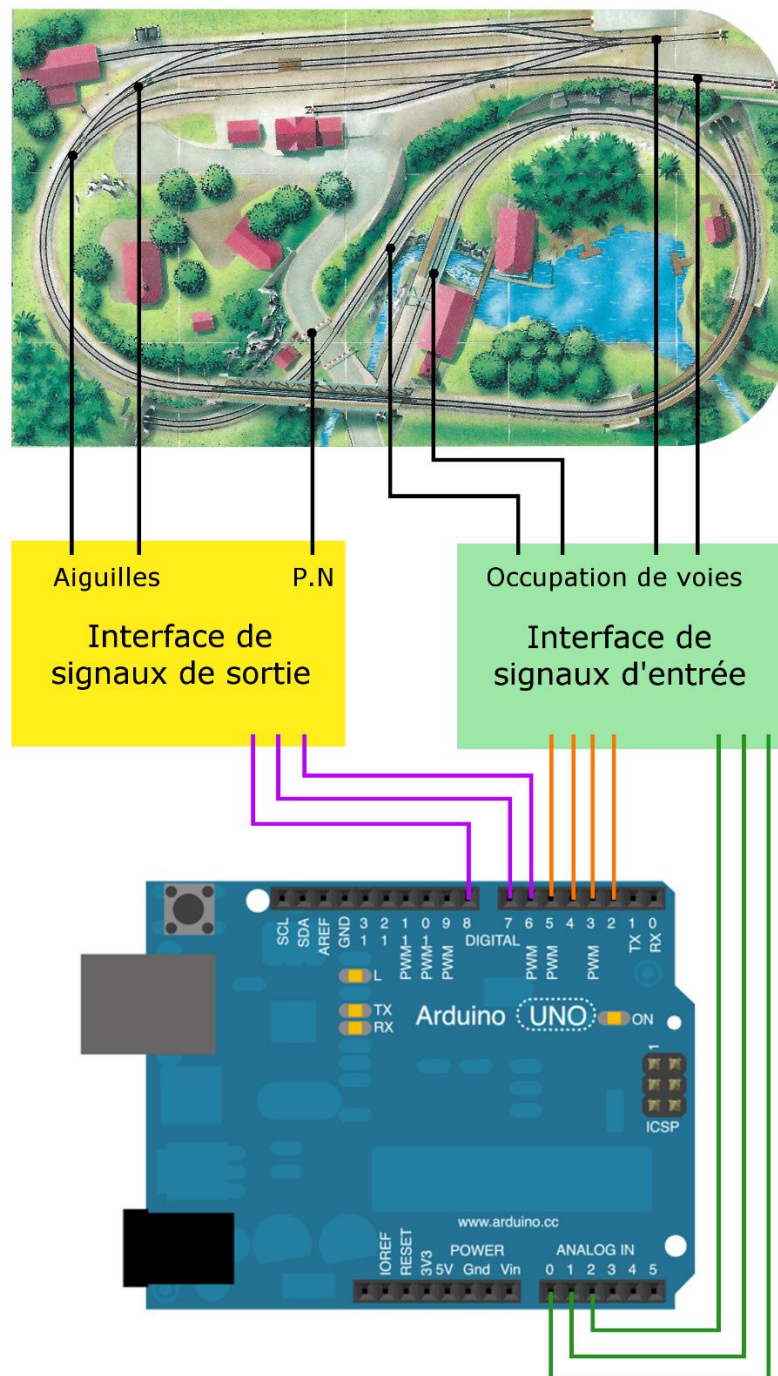


Figure 15. 3

La figure 15.3 montre comment un module Arduino peut être relié à un réseau miniature. On remarque deux genres d'interface entre réseau et module : **l'interface des signaux d'entrée** qui mettent en forme les signaux du réseau avant de les envoyer au module Arduino et **l'interface des signaux de sortie** qui amplifie les signaux du microcontrôleur avant de les envoyer au réseau pour produire une action (positionner une aiguille par exemple). Notons au passage que les signaux d'entrée sont de deux formes : analogiques (les signaux varient continûment) ou numériques (les signaux n'ont que deux valeurs possibles LOW ou HIGH). Sur la figure 15.3, les signaux d'entrée analogiques sont repérés en vert et les signaux d'entrée numériques en orange. Les signaux de sorties sont numériques et repérés en violet.

Vous l'avez maintenant compris ; cette partie du cours vous enseignera deux choses :

- L'interfaçage des signaux entre réseau et module Arduino (la partie électronique).
- La programmation du traitement des signaux pour en obtenir ce qu'on veut (la partie informatique).

Pour arriver à cela, nous resterons le plus simple possible, ce qui nous amènera peut-être dans certains cas à faire certains raccourcis afin de ne pas dévier vers un cours d'informatique de haut niveau. Le but est bien de mettre en œuvre des microcontrôleurs pour gérer, en partie du moins, un réseau de trains miniatures.

À retenir sur l'électronique programmable :

- L'électronique câblée fait appel à des composants traditionnels (résistances, condensateurs, diodes, transistors, circuits intégrés, etc.) qu'il faut assembler pour réaliser une certaine fonction.
- L'électronique programmable fait appel au microprocesseur ou au microcontrôleur qu'il suffit de programmer pour réaliser une certaine fonction.
- Le microprocesseur n'est pas adapté au modélisme ferroviaire alors que le microcontrôleur l'est.
- Les principales difficultés d'emploi du microcontrôleur sont l'écriture du programme et le transfert de celui-ci dans le circuit ; ces difficultés sont résolues par l'emploi de module Arduino (ou autre) très simples d'emploi.
- Pour que le module à base de microcontrôleur puisse être relié au réseau, il est nécessaire de concevoir des interfaces électroniques simplifiées pour adapter les signaux et les amplifier.

Liste des composants nécessaires pour continuer cette deuxième partie de cours :

Un module Arduino Uno

Un câble USB de type A-B (prise A, prise B)

Une platine d'essai

Des câbles de liaison pour platine d'essai

Du fil électrique monobrin

Une pile 9 V et sa prise

*Vous pouvez compléter cette liste **indispensable** par des composants électroniques tels que :*

Boutons poussoirs et ILS

Relais 12 V - 1 RT (au moins deux) (fonctionnent généralement sous 9 V)

Assortiment de DEL de toutes couleurs

Assortiment de résistances

Circuit intégré ULN 2803

Photocoupleurs 4N35 (au moins deux)

Quelques transistors

Quelques diodes

Et si vous désirez apprendre à programmer les microcontrôleurs ATtiny45 (voir chapitre 30) :

Plusieurs unités d'ATtiny45.