

Arduino

pour la

domotique



Arduino

pour la

domotique

Marc-Olivier Schwartz
Blogueur et entrepreneur
dans le domaine du matériel libre

Traduit de l'américain
par Vincent Briois

DUNOD

Cet ouvrage en français est la traduction augmentée et remaniée
de l'ouvrage *Home Automation with Arduino*,
publié par Marc-Olivier Schwartz.

Maquette de couverture : Misteratomic

Toutes les marques citées dans cet ouvrage sont des
marques déposées par leurs propriétaires respectifs.

Le pictogramme qui figure ci-contre mérite une explication. Son objet est d'alerter le lecteur sur la menace que représente pour l'avenir de l'écrit, particulièrement dans le domaine de l'édition technique et universitaire, le développement massif du photocopillage.

Le Code de la propriété intellectuelle du 1^{er} juillet 1992 interdit en effet expressément la photocopie à usage collectif sans autorisation des ayants droit. Or, cette pratique s'est généralisée dans les établissements

d'enseignement supérieur, provoquant une baisse brutale des achats de livres et de revues, au point que la possibilité même pour

les auteurs de créer des œuvres nouvelles et de les faire éditer correctement est aujourd'hui menacée. Nous rappelons donc que toute reproduction, partielle ou totale, de la présente publication est interdite sans autorisation de l'auteur, de son éditeur ou du Centre français d'exploitation du droit de copie (CFC, 20, rue des Grands-Augustins, 75006 Paris).



© Dunod, 2015
5 rue Laromiguière, 75005 Paris
www.dunod.com

ISBN 9782100727117

Le Code de la propriété intellectuelle n'autorisant, aux termes de l'article L. 122-5, 2° et 3° a), d'une part, que les « copies ou reproductions strictement réservées à l'usage privé du copiste et non destinées à une utilisation collective » et, d'autre part, que les analyses et les courtes citations dans un but d'exemple et d'illustration, « toute représentation ou reproduction intégrale ou partielle faite sans le consentement de l'auteur ou de ses ayants droit ou ayants cause est illicite » (art. L. 122-4).

Cette représentation ou reproduction, par quelque procédé que ce soit, constituerait donc une contrefaçon sanctionnée par les articles L. 335-2 et suivants du Code de la propriété intellectuelle.

Table des matières

AVANT-PROPOS	XI
PARTIE I • INTRODUCTION À ARDUINO	1
CHAPITRE 1 • PREMIERS PAS	3
Une plateforme révolutionnaire	3
Ce qu'il faut savoir en matière d'électronique	4
CHAPITRE 2 • LIRE LES DONNÉES D'UN CAPTEUR	9
Matériel et logiciel requis	9
Configurer le matériel	10
Lire les données du capteur	11
CHAPITRE 3 • PILOTER UN RELAIS	15
Matériel et logiciel requis	15
Configurer le matériel	16
Piloter le relais	17
CHAPITRE 4 • UTILISER LA BIBLIOTHÈQUE AREST	19
Matériel et logiciel requis	19
Configurer le matériel	20
Contrôler le projet	22

PARTIE II • CONCEVOIR DES INSTALLATIONS AUTONOMES	27
CHAPITRE 5 • INSTALLER UN SYSTÈME D'ALARME SIMPLE	29
Matériel requis	29
Assembler le matériel	30
Configurer le matériel	32
Tester le système d'alarme	34
CHAPITRE 6 • MESURER TEMPÉRATURE, HUMIDITÉ ET LUMIÈRE	35
Matériel et logiciel requis	35
Configurer le matériel	36
Tester les capteurs	38
Afficher les données sur l'écran LCD	41
CHAPITRE 7 • CONSTRUIRE UNE LAMPE INTELLIGENTE	45
Matériel et logiciel requis	45
Configurer le matériel	47
Tester le relais	49
Mesurer la puissance et la commande automatique de l'éclairage	50
CHAPITRE 8 • INSTALLER DES DÉTECTEURS DE MOUVEMENT AVEC XBEE	57
Matériel et logiciel requis	57
Réaliser un détecteur de mouvement sans fil XBee	59
Tester le détecteur de mouvement	60
Utiliser le module XBee	62
Mettre au point l'interface centrale	67
CHAPITRE 9 • TRANSMETTRE DES MESURES EN BLUETOOTH	73
Matériel et logiciel requis	73
Monter une station Bluetooth	74
Jumeler un module Bluetooth	77
Mesurer la température à distance	78
Créer une interface	83
CHAPITRE 10 • COMMANDER UNE LAMPE EN WIFI	89
Matériel et logiciel requis	89
Assembler le projet	91
Tester le module WiFi	94
Commander la lampe à distance	97
Mettre en place une interface pour la lampe intelligente	103

CHAPITRE 11 • CONSTRUIRE UN TABLEAU DE BORD	111
Matériel et logiciel requis	111
Assembler le projet	112
Tester des modules	114
Créer l'interface centrale	119
PARTIE III • CONCEVOIR DES INSTALLATIONS CONNECTÉES	127
CHAPITRE 12 • CONCEVOIR UNE STATION DE MESURES SUR LE CLOUD	129
Matériel et logiciel requis	129
Connecter des capteurs à Arduino	130
Tester des capteurs	132
Transférer des données sur le cloud	134
Accéder aux données sur le cloud	141
CHAPITRE 13 • PILOTER UNE LAMPE DEPUIS LE WEB	145
Matériel et logiciel requis	145
Connecter un relais ou une lampe à Arduino	147
Tester le relais	148
Piloter votre projet depuis n'importe où	149
CHAPITRE 14 • PUBLIER DES RELEVÉS DE MESURES EN LIGNE	153
Matériel et logiciel requis	153
Configurer le matériel	154
Tester les capteurs	155
Mettre en place votre compte Temboo	159
Stocker des données dans Google Sheets	159
CHAPITRE 15 • INSTALLER UNE CAMÉRA DE SURVEILLANCE SANS FIL	167
Matériel et logiciel requis	167
Connecter une caméra USB à la carte Arduino Yun	169
Tester la caméra	171
Capturer des images à distance	172
CHAPITRE 16 • ORGANISER UN ARROSAGE AUTOMATIQUE EN FONCTION DE LA MÉTÉO	179
Matériel et logiciel requis	179
Configurer le matériel	180
Tester le capteur d'humidité et de température du sol	181
Configurer votre compte Carriots	183

Transférer des données vers le cloud	184
Déclencher une alerte e-mail automatique	185

PARTIE IV • CONCEVOIR DES CIRCUITS IMPRIMÉS POUR DES INSTALLATIONS PERSONNALISÉES 187

CHAPITRE 17 • CONSTRUIRE SON PROPRE SYSTÈME ARDUINO 189

Matériel et logiciel requis	189
Configurer le matériel	190
Tester le projet	194

CHAPITRE 18 • OPTIMISER ARDUINO POUR DES PROJETS À BASSE CONSOMMATION 195

Matériel et logiciel requis	195
Configurer le matériel	196
Tester le projet	198

CHAPITRE 19 • CONCEVOIR UNE CARTE D'EXTENSION ARDUINO 201

Matériel et logiciel requis	201
Concevoir la carte d'extension	202
Fabriquer la carte	206
Résultat final	207

CHAPITRE 20 • CONCEVOIR UNE CARTE ARDUINO PERSONNALISÉE AVEC EAGLE 209

Matériel et logiciel requis	209
Concevoir la carte	210
Fabriquer la carte	213
Résultat final	215

PARTIE V • CONSTRUIRE SES PROPRES BOÎTIERS EN IMPRESSION 3D 217

CHAPITRE 21 • IMPRIMER UN BOÎTIER SIMPLE POUR ARDUINO 219

S'approprier un modèle existant	219
Imprimer votre boîtier via un service en ligne	221
Tester le résultat avec un projet de domotique	223

CHAPITRE 22 • MODIFIER UN MODÈLE EXISTANT 225

Concevoir une carte Arduino et choisir un boîtier	225
Personnaliser le boîtier	227
Fabriquer le boîtier	229

CHAPITRE 23 • CONCEVOIR UN BOÎTIER POUR DES CAPTEURS	231
Choisir un boîtier adapté à un système Arduino	231
Concevoir le boîtier	232
Fabriquer et tester le boîtier	235
CONCLUSION	237
RESSOURCES	239
INDEX	241

Avant-propos

J'ai découvert l'univers fascinant de la domotique à l'occasion d'une visite chez l'un de mes amis. Tout semblait étonnement simple à mes yeux : éclairage qui s'active automatiquement à la tombée du jour, transmission du relevé des températures de chaque pièce de la maison vers un serveur central, configuration du statut de l'ensemble des détecteurs d'alarme à partir d'un téléphone portable...

La contrainte principale à cette époque était de taille : le système était spécialement conçu pour cette maison par des entreprises privées. Son acquisition était donc réservée à une clientèle aisée. D'ailleurs, nombreux sont ceux qui continuent à croire que la domotique reste financièrement inaccessible.

Ces systèmes « propriétaires » présentent selon moi un autre point faible : l'impossibilité de les contrôler soi-même. Vous devez suivre un processus bien défini par le fabricant à propos de l'unité centrale, des capteurs et de l'interface. Si par exemple un des capteurs de votre système est déficient, il devra être remplacé par un autre de la même marque. Je me souviens avoir essayé de manipuler un de ces systèmes vendus dans le commerce. Lorsque je tentais de rendre l'utilisation d'un capteur plus simple ou de corriger un bogue dans l'interface, c'était tout simplement impossible.

Bien sûr, créer sa propre installation domotique n'est pas un concept nouveau. Je me rappelle de mes premières manipulations sur un microcontrôleur en 2003. Cela me semblait assez simple pour quelqu'un du monde de l'ingénierie. Ces systèmes n'offraient que des applications limitées et chacun d'eux nécessitait l'apprentissage de connaissances spécifiques sur sa propre plateforme. De plus, les kits d'évaluation de ces microcontrôleurs étaient également assez onéreux.

Ces dernières années ont vu la montée en puissance d'un nouveau mouvement, celui du matériel libre (*open hardware*). Ce mouvement est semblable à celui du monde du logiciel et consiste à donner la possibilité à quiconque d'accéder aux plans de produits physiques et ainsi de les personnaliser. De ce mouvement est née une plateforme qui révolutionne le monde de l'électronique : la plateforme Arduino. Pourvue d'un environnement simple et agréable à utiliser, Arduino facilite la programmation de microcontrôleurs.

En ce qui me concerne, cela a changé complètement ma vision de la domotique. La réalisation d'installations domotiques est désormais accessible à la plupart de ceux qui

ont des notions en électronique et en programmation. Cet ouvrage est là pour vous montrer la démarche à suivre.

Depuis la parution initiale de ce livre au format numérique, des milliers d'utilisateurs ont mis en application les principes décrits afin de créer leurs propres installations domotiques. J'ai également reçu un certain nombre de commentaires pertinents qui m'ont aidé à améliorer l'ouvrage.

Pendant, beaucoup de ces retours regrettaient la difficulté relative des projets présentés. Ces commentaires évoquaient plus souvent le nombre de langages de programmation abordés que le contenu même des projets. Pour les projets impliquant par exemple la communication entre une carte Arduino et un ordinateur, j'avais utilisé à la fois les langages Python, PHP, HTML et JavaScript. Il était également nécessaire d'installer et de lancer un serveur web sur son ordinateur, ce qui rendait la tâche encore plus complexe.

C'est la raison pour laquelle j'ai effectué des modifications dans la nouvelle édition de ce livre en privilégiant la simplicité et la clarté.

En ce qui concerne la mise en place de l'interface sur un ordinateur, je n'ai conservé qu'un seul langage : **JavaScript**. J'ai supprimé l'ensemble des références aux autres langages tels que Python et PHP. Côté serveur, j'ai opté pour l'utilitaire de programmation **Node.js** dans JavaScript. Ainsi, il n'est plus nécessaire d'installer un serveur web sur votre ordinateur.

J'espère que ces changements vous paraîtront pertinents et que vous mettrez en application l'ensemble des connaissances acquises dans ce livre pour des projets encore plus élaborés avec Arduino.

À propos de l'auteur

Je suis ingénieur en électronique, entrepreneur et auteur, titulaire d'un diplôme d'ingénieur Supélec, la plus prestigieuse école de France dans ce domaine, et d'un master en microtechnique obtenu à l'École polytechnique fédérale de Lausanne (EPFL).

J'ai plus de cinq années d'expérience professionnelle en ingénierie électronique et je m'intéresse à tout ce qui touche l'électronique, la domotique, la plateforme Arduino, les projets à base de matériel libre et l'impression 3D.

Je travaille en tant qu'entrepreneur à plein-temps depuis 2011 et gère des sites Internet fournissant des informations sur le matériel libre. Je crée également en parallèle mon propre matériel libre.

Remerciements

J'adresse mes remerciements à tous mes amis qui m'ont apporté leur soutien tout au long de l'écriture de ce livre et de tous les projets sur lesquels j'ai travaillé.

Je remercie également mes parents pour leur soutien durant cette période mais aussi dans les différents projets que j'ai pu mener en général, lors des périodes difficiles et moins difficiles.

Enfin, je tiens à remercier Sylwia, ma compagne, pour son soutien permanent. C'est elle qui me donne l'inspiration nécessaire pour aborder chaque journée avec sérénité et la force de persévérer dans mes projets.

Merci à tous.

**RESSOURCES NUMÉRIQUES
ET SITE WEB COMPLÉMENTAIRE**

Cet ouvrage dispose d'un site web complémentaire en anglais, Open Home Automation, accessible à cette adresse :



<http://www.openhomeautomation.net>

Vous y trouverez davantage de projets et de ressources en matière de domotique et de matériel libre.

Vous pouvez retrouver l'ensemble des codes présents dans cet ouvrage à cette adresse :

<https://github.com/openhomeautomation/home-automation-arduino>

Ce dépôt **GitHub** créé spécialement pour le livre contient les codes les plus récents des projets abordés.

Partie I

Introduction à Arduino

1 Premiers pas	3
Une plateforme révolutionnaire	3
Ce qu'il faut savoir en matière d'électronique	4
2 Lire les données d'un capteur	9
Matériel et logiciel requis	9
Configurer le matériel	10
Lire les données du capteur	11
3 Piloter un relais	15
Matériel et logiciel requis	15
Configurer le matériel	16
Piloter le relais	17
4 Utiliser la bibliothèque aREST	19
Matériel et logiciel requis	19
Configurer le matériel	20
Contrôler le projet	22

1 Premiers pas

Avant de vous entraîner dans la réalisation de votre premier projet de domotique ouvert, je souhaiterais vous parler de la plateforme Arduino que nous allons utiliser tout au long de l'ouvrage.

UNE PLATEFORME RÉVOLUTIONNAIRE

Les débuts de Arduino remontent à 2005, lorsque ses créateurs, Massimo Banzi¹ et David Cuartielles, travaillaient sur la mise au point d'un appareil simple à programmer par un public de non-spécialistes afin que leurs étudiants en design puissent construire des projets impliquant des microcontrôleurs.

La plateforme Arduino n'a pas été conçue dans le seul but de fonctionner avec des cartes ou des microcontrôleurs, elle constituait également une solution complète matérielle et logicielle plus simple à utiliser par rapport aux autres microcontrôleurs.

Côté matériel, Arduino est un microcontrôleur monocarte, équipé en général d'un microcontrôleur 8 bits Atmel AVR, même si de nouveaux modèles tels que la carte Arduino Due disposent d'un processeur ARM 32 bits. Nous n'aurons pas besoin d'une carte aussi puissante pour nos projets ; de ce fait, nous utiliserons la carte Arduino la plus courante : le modèle **Arduino Uno**.

Les broches des cartes Arduino ont la particularité d'être disposées toujours de la même façon, ce qui rend ainsi les connexions aux cartes d'extension (*shield*) très aisées. Ces dernières fournissent des fonctionnalités complémentaires à la carte, comme la possibilité de commander un moteur à courant continu dans le domaine de la robotique, ou de se connecter à un téléphone portable en Bluetooth.

Selon moi, c'est bien la partie logicielle qui rend la plateforme Arduino si puissante.

Pour programmer une carte Arduino, vous pouvez vous servir du logiciel officiel (en téléchargement libre) et utiliser ensuite un langage proche de C++ pour écrire le code que vous désirez charger sur la carte.

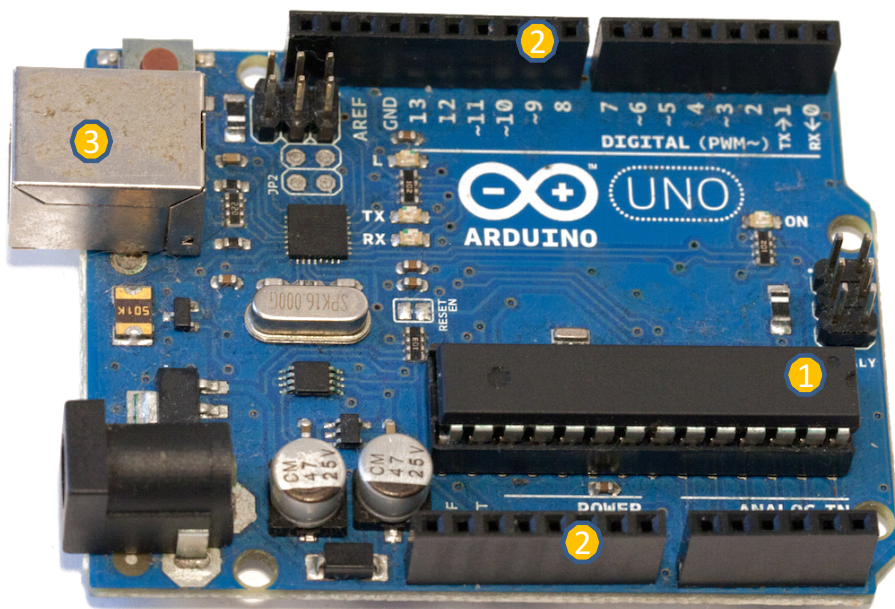
En comparaison avec d'autres microcontrôleurs, la programmation de la carte est très abordable ; vous pourrez très aisément lui faire réaliser ce que vous souhaitez. L'éclairage d'une LED ne nécessitera avec Arduino qu'une seule ligne de code. Avec d'autres microcontrôleurs, il vous faudrait en écrire plusieurs.

La plateforme Arduino dispose d'une très vaste communauté, ce qui est aussi un atout majeur. Il existe, grâce à cela, beaucoup de documentation sur le site officiel Arduino

1. Voir *Démarrez avec Arduino*, par M. Banzi et M. Shiloh (3^e édition, Tous Makers !, Dunod, 2015).

(www.arduino.cc) pour chaque fonction utilisée. Des tutoriels relatifs aux fonctions les plus courantes sont également à votre disposition.

Voici maintenant des informations complémentaires au sujet de la carte **Arduino Uno** avec un aperçu de la carte dont je me suis servi pour réaliser l'ensemble des projets décrits dans ce livre :



La carte en elle-même est de très petite taille.

Ce que vous voyez dans le coin inférieur droit de l'illustration est un **microcontrôleur Atmel** (❶), le « cerveau » de la carte. C'est lui qui reçoit le logiciel que nous allons développer dans le cadre de nos projets de domotique.

En haut et en bas de la carte, vous pouvez voir une série de **connecteurs** (❷). Ils serviront à connecter les signaux d'entrée et de sortie comme les entrées analogiques, les entrées et sorties numériques ainsi qu'à connecter des composants extérieurs aux tensions de référence (masse et 5 V).

Enfin, vous pouvez voir le **port USB** (❸) dans le coin supérieur gauche qui nous permettra de relier la carte à votre ordinateur.

CE QU'IL FAUT SAVOIR EN MATIÈRE D'ÉLECTRONIQUE

Ce livre n'aborde pas les principes généraux de l'électronique ; vous trouverez d'autres ouvrages plus adaptés à cet effet.

Il est question dans cet ouvrage de vous montrer comment réaliser une installation domotique. Vous y apprendrez à connecter différents composants, capteurs et autres appareils à la plateforme Arduino.

Cependant, afin de comprendre le mode de fonctionnement de ces composants, il vous faut connaître quelques principes fondamentaux. Vous trouverez dans cette partie un bref aperçu des principes évoqués dans les projets du livre.

Les principales variables utilisées en électronique

Il existe un bon nombre de variables pour définir un circuit. Nous allons uniquement aborder les plus importantes.

Un circuit électrique est comparable à un circuit hydraulique. Afin que l'eau puisse effectuer naturellement son circuit d'un point A à un point B, il est indispensable d'avoir un écart de hauteur entre ces deux points.

En électronique, cet écart est appelé **tension** (U), et cette grandeur est mesurée en volts.

Le débit d'eau entre un point A et un point B est comparable à un flux d'électrons au sein d'un circuit électrique. On appelle **courant électrique** ce flux d'électrons. Il est représenté par la lettre I.

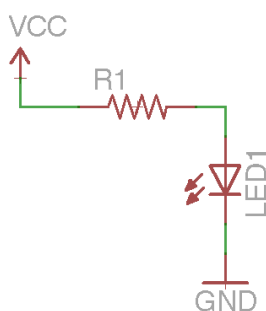
La **puissance** P dissipée par un équipement donné est obtenue en multipliant la tension par l'intensité. Elle est mesurée en watts (W) :

$$P = U * I$$

Représentation d'un circuit de base

Un circuit électrique peut être représenté à l'aide de symboles normalisés.

Voici par exemple un circuit simple comprenant une source de tension VCC (alimentation tension continue), une résistance R1, une LED appelée LED1 et une broche de masse (GND).



Par la suite, nous verrons plus en détail quelques-uns de ces composants mais nous allons pour l'instant nous contenter d'identifier les plus courants.

La première étape dans la lecture d'un schéma électrique consiste à repérer la source d'alimentation et les broches de masse.

Dans le cas présent, la source est représentée par la broche VCC. Dans la plupart des projets de ce livre, cette source sera d'une tension de 5 V.

La broche de masse est représentée ici par la broche GND.

Après avoir localisé ces deux broches, vous pourrez vous intéresser aux composants. Dans ce cas, il s'agit d'une résistance et d'une LED.

Sources d'alimentation

Dans le premier circuit de cette partie, la source d'alimentation était une broche nommée « VCC ». Théoriquement, VCC peut être n'importe quelle source de tension, mais elle est par convention une source d'alimentation positive à basse tension (de 3,3, 5 ou 12 V en général).

On utilisera en général le port USB de la carte Arduino pour alimenter les projets présentés dans cet ouvrage. Toutefois, il vous est possible d'alimenter votre carte Arduino à partir d'autres sources d'alimentation telles que des alimentations régulées directement branchées à une prise murale (veillez cependant à ne pas dépasser la tension maximale tolérée par votre carte Arduino) ou à des piles.

Résistance

La résistance est un composant clé au sein d'un circuit électrique. En reprenant l'exemple du circuit hydraulique, la résistance joue le rôle de limiteur de débit d'eau (ici d'électrons) à un endroit donné du circuit.



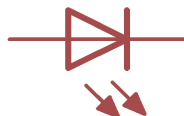
Afin de quantifier cette limitation, nous introduisons une nouvelle variable, R , qui s'exprime en ohms (Ω). On appelle **loi d'Ohm** la formule reliant tension (U), intensité du courant (I) et résistance (R) selon la relation :

$$U = R * I$$

LED

La LED (diode électroluminescente) est le composant le plus utilisé en tant qu'élément de signalisation et de test dans un circuit. Lorsque du courant électrique (d'une intensité de 20 mA en moyenne) traverse une LED, celle-ci produit une lumière rouge, bleue, verte ou encore blanche selon son type.

Sur une carte Arduino, elle sert par exemple à vérifier que la carte est bien alimentée ou à indiquer le bon fonctionnement de la communication série. Les LED peuvent également être utilisées pour effectuer des tests du logiciel (sur la broche n° 13).



Comme nous l'avons évoqué dans le cadre du premier circuit, on associe souvent les LED à des résistances afin de limiter le courant qui y circule. Gardez bien à l'esprit que les pattes des LED ne sont pas toutes identiques. Une alimentation électrique positive (VCC, par exemple) doit être reliée à la patte gauche (sur le schéma) de la LED, l'anode et l'autre patte, la cathode, doit quant à elle être connectée à la masse. La cathode est facilement repérable puisqu'elle dispose de la patte la plus courte.