



Dominique MOLLARD

Version numérique

**OFFERTE !**

[www.editions-eni.fr](http://www.editions-eni.fr)

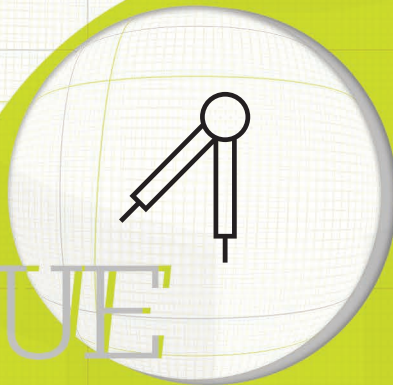
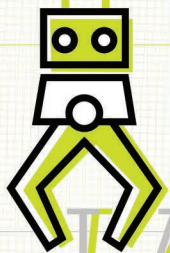
# Arduino

## Apprenez à coder avec mBlock

(projets robotiques, créatifs et scientifiques)



Fichiers  
complémentaires  
à télécharger



LA FABRIQUE

Les éléments à télécharger sont disponibles à l'adresse suivante :

<http://www.editions-eni.fr>

Saisissez la référence ENI de l'ouvrage **LFARDB** dans la zone de recherche et validez. Cliquez sur le titre du livre puis sur le bouton de téléchargement.



## Avant-propos

### Chapitre 1

## Développement et codage

1. Au commencement était le verbe : les langages informatiques .....	11
2. L'art de programmer un ordinateur (Donald E. Knuth) .....	13
2.1 En ce temps-là.....	13
2.2 Serez-vous calculer la date de Pâques du premier coup ? .....	15
3. La programmation textuelle : un frein à l'apprentissage ? .....	17
4. La programmation visuelle et les blocs .....	21
5. Aller plus vite et plus loin avec la programmation visuelle et la génération de code .....	25
6. Utiliser l'atelier mBlock pour développer des programmes pour l'Arduino .....	27
6.1 De Scratch à mBlock .....	28
6.2 De mBlock à Arduino .....	32
7. Les autres ateliers et outils de programmation visuelle .....	37
7.1 Ardublock .....	37
7.2 Blockly .....	37
8. Conserver ses données et choisir une solution de stockage local ou en « cloud » .....	38
8.1 Des fichiers locaux .....	38
8.2 Les structures de données pour le Web .....	41
8.3 Un stockage dans un nuage, petit ou grand .....	41
9. La loi du moindre effort : réutiliser et partager le code, utiliser des bibliothèques .....	42
10. Communiquer avec le reste du monde : les interfaces et protocoles .....	43
10.1 Multiplexage et protocoles courants (SPI, I2C) .....	43
10.2 Le protocole I2C .....	44

## 2 Titre du livre

10.3 Le protocole SPI.....	46
10.4 Comment choisir entre I2C et SPI ?.....	47
10.5 Les autres protocoles : 1-Wire, 3-Wire et SMBus.....	48
10.5.1 1-Wire.....	48
10.5.2 3-Wire et MicroWire.....	49
10.5.3 SMBus.....	49

### Chapitre 2

## Algorithmes et blocs

1. « Algorithme + structure de données = programme » (Nicklaus Wirth).....	51
1.1 Les structures de contrôle.....	52
1.2 Structures conditionnelles dans Scratch.....	54
1.3 Structures répétitives dans Scratch.....	55
1.4 Les structures de données.....	57
1.5 Les types de données utilisés par l'Arduino.....	57
1.5.1 Les nombres :.....	57
1.5.2 Les booléens.....	58
1.5.3 Les tableaux.....	58
1.5.4 Les chaînes de caractères.....	58
1.5.5 Les fonctions : void.....	58
1.5.6 Les constantes.....	58
1.6 Les structures de contrôle les plus utilisées.....	59
1.6.1 La structure conditionnelle.....	59
1.6.2 La structure répétitive.....	60
1.7 Les opérateurs.....	61
1.7.1 Les opérateurs de comparaison.....	61
1.7.2 Les opérateurs booléens.....	61
1.7.3 Les opérateurs composés.....	62
1.8 Les fonctions courantes.....	62
1.8.1 Entrées-sorties digitales.....	62
1.8.2 Entrées-sorties analogiques.....	62
1.9 Autres fonctions et bibliothèques.....	63
2. Qualités et défauts des algorithmes (complexité, rapidité, "gloutonnerie").....	63
2.1 Les critères de choix ou d'évaluation des algorithmes.....	64
2.2 De la gloutonnerie des algorithmes.....	64

3. Aller au-delà de Scratch : utiliser la bibliothèque de blocs de mBlock.....	66
--	----

## Chapitre 3

### Étendre les fonctionnalités de mBlock

1. Utiliser et gérer des extensions.....	67
1.1 Les extensions de base.....	69
1.2 Les extensions évoluées ou spécifiques.....	70
1.3 Les extensions cachées.....	85
1.4 Les extensions manquantes.....	86
2. Créer ses propres extensions.....	86
2.1 Entrer dans la logique des extensions de mBlock.....	87
2.2 L'arborescence d'une extension.....	88
2.3 La structure d'un fichier de définition d'extension.....	89
2.4 Quelques règles de syntaxe utiles à garder présentes à l'esprit.....	90
2.5 La description de l'extension.....	91
2.6 La définition du ou des blocs.....	92
2.7 Une précision concernant les bibliothèques.....	96
2.8 La définition des options des menus déroulants.....	97
2.9 Les valeurs des menus et les valeurs par défaut.....	97
2.10 La traduction.....	97
2.11 Le fichier JavaScript.....	98
2.12 Créer une extension avec un éditeur de code.....	98
2.13 Une démarche de création d'une extension et son application.....	98
2.14 Application à un exemple simple.....	102
2.15 Faire évoluer une extension.....	108
3. Assembler des scripts dans l'IDE de l'Arduino.....	113

## Chapitre 4

### Applications à quelques petits projets

1. L'Arduino comme base technique pour le prototypage.....	117
2. De la plaque d'expérimentation ("breadboard") au circuit imprimé.....	123
3. Déployer à grande échelle et miniaturiser : l'ESP8266.....	126

## 4 Arduino - Apprenez à coder avec mBlock

4. Quelques projets courants .....	128
4.1 Commander un moteur avec un transistor et un pont en H .....	128
4.2 Piloter un servomoteur classique .....	132
5. Gérer des capteurs : distance, lumière, son, température .....	134
5.1 Un cas pratique d'utilisation d'un capteur : la mesure de distance .....	136
5.2 Asservir un servomoteur à un capteur .....	138
6. Des projets moins courants .....	142
6.1 Utiliser un accéléromètre .....	142
6.1.1 Qu'est-ce qu'un accéléromètre ? .....	142
6.1.2 Le fonctionnement d'un accéléromètre et d'un gyroscope .....	144
6.1.3 Les bases : la lecture des données RAW .....	147
6.1.4 Un pas plus loin : le quaternion et les angles d'Euler .....	149
6.1.5 Utilisation dans les montages et projets .....	150
6.1.6 Les extensions mBlock pour mettre en œuvre un accéléromètre .....	151
6.2 Ajouter une base de temps .....	155
6.3 Stocker des données sur une carte SD .....	161

## Chapitre 5

### Conseils de Pro

1. Définir des fonctions et des procédures .....	165
1.1 Les fonctions .....	165
1.2 Les procédures .....	166
2. Adopter une règle de nommage des variables et des objets (et s'y tenir !) .....	172
3. Gérer les versions de code .....	172
3.1 Les outils de gestion de version .....	172
3.2 Une métaphore arboricole ou ferroviaire .....	173
4. L'assemblage et l'intégration .....	174
5. Tester et détecter les bugs avant l'exécution : une check-list avant décollage .....	175
6. Tracer et commenter les algorithmes et le code .....	177
7. Ne jamais supposer que le microcontrôleur ou l'ordinateur sait quelque chose... ..	181
8. Penser au pire, développer pour le meilleur .....	182
8.1 Penser au pire .....	182
8.2 Développer pour le meilleur .....	184

9. Autres proverbes utiles d'Henry Ledgard .....	185
10. Un exemple de problème : le rebond des boutons poussoirs et des détecteurs de contacts .....	186
10.1 Le problème .....	186
10.2 Les solutions .....	187
11. Cent fois sur le métier remettre son ouvrage .....	188
12. Gérer les bibliothèques .....	190
12.1 Chevrons et guillemets .....	190
12.2 Le cas particulier de mBlock .....	191

## Chapitre 6

### Un peu d'électronique et de mécanique bien utile

1. Courants forts et courants faibles, moyenne et basse tension .....	193
2. Les robots montrent leurs muscles : des courants forts et des tensions élevées .....	194
3. Des signaux faibles .....	196
4. Séparer les deux mondes .....	197
5. Juste assez d'électronique pour faire marcher ses montages .....	199
5.1 L'alimentation d'une LED .....	199
5.2 Potentiomètre et poussoirs .....	199
5.3 Commande de puissance .....	200
5.4 Pont en H .....	201
5.5 Commande de servomoteur .....	202
5.6 Régulation de tension .....	202
5.7 Utiliser des schémas courants et réutilisables .....	202
6. La mécanique pour les non-mécaniciens .....	204
6.1 La transformation du mouvement .....	204
6.2 Roues et pignons, poulies et courroies, bielles et vérins .....	205
6.3 Couple moteur et coefficient de réduction .....	206

## 6 Arduino - Apprenez à coder avec mBlock

### Chapitre 7

## Les objets connectés et l'Internet des objets (IoT)

1. L'Internet des objets (IoT, Internet of Things) .....	207
2. Sécuriser ses montages et contrôler les accès.....	209
2.1 Les objets connectés et leur environnement hostile .....	209
2.2 Sécuriser ses montages .....	210
2.3 Contrôler ses accès.....	211
2.4 Les IoT avec Arduino .....	212
3. Relayer la valeur d'un capteur par Internet, un affichage ou un son .....	212
3.1 Un premier montage basé sur un serveur web .....	214
3.1.1 Le capteur de température.....	215
3.1.2 Le son.....	215
3.1.3 L'affichage .....	216
3.1.4 La communication par Internet.....	216
3.1.5 Affectation des pins .....	217
3.1.6 Script mBlock .....	217
3.1.7 Composants nécessaires .....	219
3.1.8 Bibliothèques nécessaires.....	219
3.1.9 La gestion du capteur .....	219
3.1.10 La communication par Internet.....	221
3.1.11 L'affichage LCD 16 x 2 .....	224
3.2 Amélioration possible et évolution.....	226
4. Web-applications pour tablettes et portables .....	226
4.1 Blynk.....	226
4.1.1 Une extension mBlock pour Blynk.....	228
4.1.2 Application : éteindre et allumer une LED depuis Internet .....	230
4.2 ThingSpeak .....	233

### Chapitre 8

## Un beau projet pédagogique et créatif avec mBlock

1. Introduction .....	235
2. Une idée de projet pédagogique et créatif.....	235
3. Une variante ludique.....	238

4. Une touche artistique .....	238
5. Mettre en pratique : le pas-à-pas d'un projet créatif, ludique et artistique .....	240
5.1 Une première approche .....	240
5.2 La liste des composants nécessaires .....	240
6. Affectation des composants aux ports de l'Arduino .....	242
7. Déclaration des variables .....	244
8. L'algorithme de traitement .....	246
8.1 Initialisation des variables .....	248
8.2 Algorithme de traitement des données mesurées par les capteurs .....	248
9. Finalisation et mise en route .....	256

## Chapitre 9

### Les projets pédagogiques et scientifiques

1. Choisir un projet motivant, pédagogique et, si possible, « dans le courant » .....	259
2. Les stations de mesure .....	264
2.1 La mesure environnementale et climatique et le partage de ses données .....	265
2.2 Mesures et calculs .....	267
2.3 Horodatage (timestamp) ou estampillage des données .....	268
2.4 Stockage des données .....	270
2.5 Restitution des données .....	272

## Chapitre 10

### Une approche ludique par la robotique

1. Quelques robots célèbres .....	273
1.1 Robbie .....	274
1.2 Gort .....	274
1.3 R2-D2 et C-3PO .....	275
1.4 WALL-E .....	275
1.5 Marvin .....	276
1.6 NAO, Roméo et Pepper .....	276
1.7 TARS .....	277
2. Les kits de découverte du marché .....	278



## 8 Arduino - Apprenez à coder avec mBlock

3. L'approche « maker » et le recyclage d'objets .....	280
4. 1 pile + 1 Arduino + quelques composants + 2 servomoteurs = un robot suiveur de ligne ...	281
4.1 Les capteurs .....	281
4.2 La logique de commande.....	282
4.3 Les actionneurs .....	283
4.4 L'architecture générale du robot .....	283
4.5 Le script mBlock .....	284
5. Simplifier pour expérimenter .....	284
6. La documentation utile : les pas à pas .....	287
7. ERA : les trois séquences d'un atelier de robotique ou de programmation : Enseigner, Réfléchir, Appliquer.....	290
7.1 Enseigner .....	290
7.2 Réfléchir .....	292
7.3 Appliquer .....	293
7.4 Mettre en œuvre la démarche .....	293

## Chapitre 11

### Didactique de la programmation et apprentissage du numérique

1. À chacun son parcours, à chacun son approche .....	295
2. Didactique de la programmation et apprentissage du numérique .....	296
3. Le travail en atelier de « coding » : autonomie et collaboration .....	297
3.1 Le codage individuel.....	298
3.2 Le codage collectif.....	298
3.3 Le « hackaton » et le « game jam » ou « game creation event ».....	299
3.4 Un exemple d'apprentissage par le jeu avec Code Combat .....	299
4. Un autre triptyque : parcours pédagogique - documents d'appui - outils .....	300
5. Encourager pour aller plus loin et développer l'expertise .....	302

## Chapitre 12

**Utiliser ce livre en s'adaptant à son public :  
des élèves aux « makers »**

1. Dans une salle de classe.....	305
2. Lors d'une formation à la robotique.....	306
3. Dans un atelier de codage.....	306
4. Chez soi ou dans un « Lab ».....	308
5. Suivre le développement de la technique et améliorer ses montages.....	308
6. Miser sur les librairies et les composants intégrés.....	309

Index.....	311
------------	-----

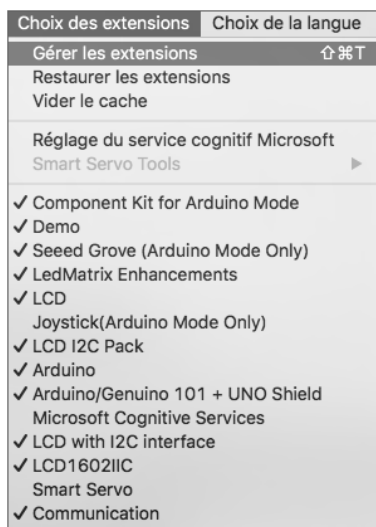
## Chapitre 3

# Étendre les fonctionnalités de mBlock △

### 1. Utiliser et gérer des extensions

Les extensions de mBlock viennent compléter les fonctionnalités livrées en standard et sont mises à disposition par des développeurs indépendants ou des fournisseurs de kits ou de matériel.

Le menu **Choix des extensions** de mBlock permet de sélectionner des extensions existantes ou d'en ajouter.



## 68 Arduino - Apprenez à coder avec mBlock

Par exemple, deux groupes d'extensions permettent de récupérer facilement la valeur d'un capteur de température très courant, le DHT11 de d-robotics puis de l'afficher avec un texte d'accompagnement sur un afficheur LCD de 2 lignes de 16 caractères.

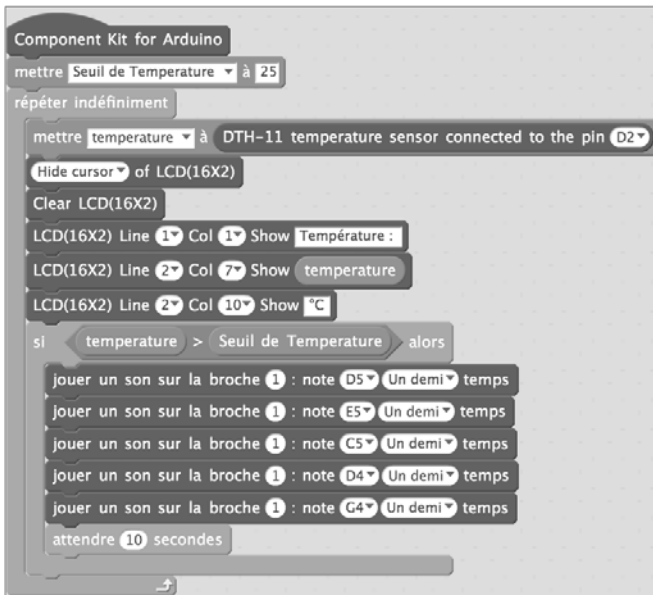
La première permet de récupérer directement la température lue par le capteur à partir du port digital indiqué. Il existe un autre bloc qui permet de réaliser la même action pour la température.

Les quatre blocs suivants sont dédiés à la gestion d'un afficheur LCD piloté avec le protocole I2C :

- Cacher ou montrer le curseur.
- Effacer l'écran.
- Afficher un texte à une position définie par une ligne et une colonne.
- Combiner les deux actions précédentes.
- Utiliser ou non le rétroéclairage.

Programmer une action particulière lors du dépassement d'un seuil de température devient alors très facile. Une boucle infinie permet de mesurer régulièrement et systématiquement la température à l'aide du capteur DHT11. Si cette température dépasse un seuil fixé défini en paramètre alors une courte mélodie se fait entendre. Pour laisser le temps au capteur de suivre les évolutions de la température, une temporisation est ajoutée.

Le programme résultant est de la forme :



```
Component Kit for Arduino
mettre Seuil de Temperature à 25
répéter indéfiniment
  mettre temperature à DTH-11 temperature sensor connected to the pin D2
  Hide cursor of LCD(16X2)
  Clear LCD(16X2)
  LCD(16X2) Line 1 Col 1 Show Température :
  LCD(16X2) Line 2 Col 7 Show temperature
  LCD(16X2) Line 2 Col 10 Show °C
  si temperature > Seuil de Temperature alors
    jouer un son sur la broche 1 : note D5 Un demi temps
    jouer un son sur la broche 1 : note E5 Un demi temps
    jouer un son sur la broche 1 : note C5 Un demi temps
    jouer un son sur la broche 1 : note D4 Un demi temps
    jouer un son sur la broche 1 : note G4 Un demi temps
    attendre 10 secondes
```

Un coup d'œil sur le code généré permet de repérer les bibliothèques utilisées :

```
#include <Arduino.h>
#include <Wire.h>
#include <SoftwareSerial.h>

#include "dht11.h"
#include <IRremote.h>
#include <dht11.h>
#include <LiquidCrystal_I2C.h>
#include <OneWire.h>
#include <Adafruit_GFX.h>
#include <Adafruit_TFTLCD.h>
```

On remarque qu'en plus des bibliothèques « standards » que sont Arduino.h, SoftwareSerial.h et Wire.h, se trouvent dht11.h et LiquidCrystal.h qui permettent de gérer le capteur DHT11 et l'écran LCD.

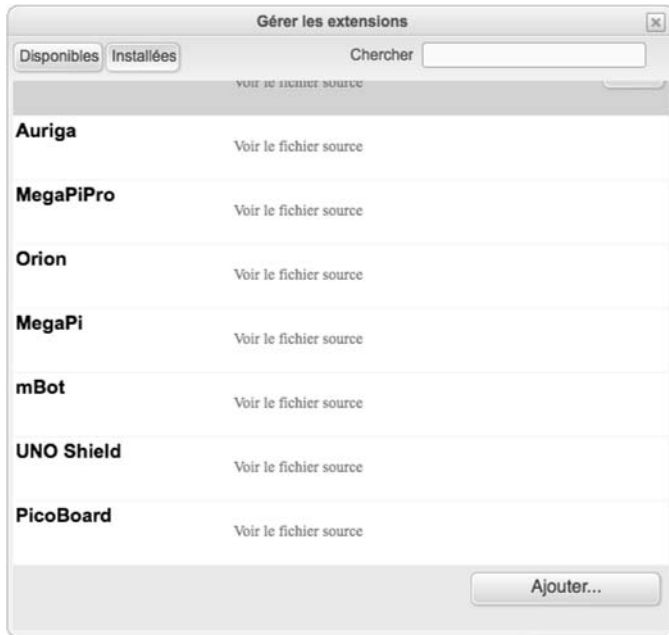
On remarque également d'autres bibliothèques qui ne sont pas utiles au projet, mais qui figurent parmi les extensions chargées.

### 1.1 Les extensions de base

Les extensions de base sont livrées « en standard » avec mBlock et comprennent les blocs nécessaires au pilotage des modules et des cartes supportées par cet environnement ainsi que des blocs basiques de pilotage de l'Arduino.

## 70 Arduino - Apprenez à coder avec mBlock

Les modules standards sont les suivants :



Le module Arduino est quant à lui optionnel et peut être enlevé de l'interface, mais il contient cependant tous les blocs de base.



Les blocs utilisables dans cette extension ont été présentés au chapitre Développement et codage.

### 1.2 Les extensions évoluées ou spécifiques

En complément des extensions de base, il en existe une cinquantaine d'autres qui permettent de coder des montages intéressants avec Arduino et qui sont téléchargeables depuis le menu « choix des extensions de mBlock ». Il est conseillé de n'ajouter ou d'activer que les extensions nécessaires, car le générateur de code ajoutera systématiquement les bibliothèques des extensions actives, ce qui alourdira le programme. L'environnement de mBlock est aussi ralenti lorsque le nombre d'extensions est trop important.

Un bon départ pour garder un environnement de développement lisible est de n'utiliser que les extensions « Arduino » et « Component kit for Arduino mode » qui contiennent l'essentiel. Ensuite, il est préférable de tester une par une les nouvelles extensions et de ne retenir que celles qui apportent réellement un plus, comme la gestion d'un écran TFT ou LCD, ou qui sont spécifiques à un matériel ou un kit de développement particulier.

Les extensions disponibles sont listées sur le site <http://www.mblock.cc/extensions/> et également sur GitHub. Cette liste est un sous-ensemble des extensions utilisées réellement par les développeurs, car elle ne contient que celles qui ont été partagées via le site GitHub. La possibilité d'ajouter des extensions directement à partir de son poste de travail facilite leur mise au point, car les différentes versions et corrections ne sont pas publiées inutilement. Seules les versions opérationnelles et dont le potentiel de réutilisation par d'autres développeurs est suffisant sont publiées sur le site communautaire.

À la date d'écriture de ce texte, les 49 extensions disponibles sur le site de MakeBlock se répartissent en 5 catégories, que l'on peut décomposer en sous-catégories :

Catégorie	Sous-catégorie
<b>Capteurs</b>	Capteur
	Lecteur RFID
<b>Actionneurs</b>	Afficheur à LED
	Afficheur LCD
	Afficheur TFT
	Contrôle de moteurs
<b>Communication</b>	Communication Bluetooth
	Communication Wi-Fi
	Internet des objets (IoT)
<b>Composants et modules</b>	Kit de composants
	Module spécifique
	Plate-forme Arduino
<b>Pédagogie</b>	Initiation
	Plate-forme robotique ou pédagogique
	Enseignement

Typologie des extensions disponibles dans mBlock

## 72 Arduino - Apprenez à coder avec mBlock

On distingue les extensions généralistes, qui apportent des blocs utilitaires pouvant assurer des fonctions courantes comme la gestion d'un afficheur ou d'un sonar, et les extensions spécifiques à un module ou à une plate-forme en particulier. Le fichier Excel accessible en téléchargement permet de retrouver facilement les extensions disponibles par catégorie.

La liste complète à la date de cet ouvrage dans l'ordre de sa présentation sur le site de MakeBlock est la suivante :

N°	Nom	Catégorie/ Sous-catégorie	Auteur	Date de publication	Description
1	Demo 1.1	Pédagogie/ initiation	Wangyu	22/06/2017	Une extension de démonstration pour apprendre à développer des extensions. Elle est détaillée dans le guide de développement d'extensions de MakeBlock.
2	Seed Grove (Arduino Mode Only) 1.0.1	Composants et modules	Wangyu	20/10/2016	Démonstration pour le starter kit Arduino de Seed : Grove Arduino Extension : Accéléromètre MMA7660, buzzer, capteur de température, de son Grove.
3	Component Kit for Arduino Mode 1.1.0	Composants et modules	CoderDojoFVG	06/11/2016	Blocs en appui du kit de composants pour Arduino utilisé par le CoderDojo de Friuli Venezia Guila (CoderDojoFVG).