

# Commandez une loco analogique AVEC VOTRE SMARTPHONE

Dans notre dernier numéro, Christian Bézanger nous proposais de découvrir le monde des objets connectés grâce à un kit de découverte de l'ESP32. Aujourd'hui, il nous invite à passer à la pratique en construisant une manette WiFi pour commander une loco analogique.

Texte et illustrations: Christian Bézanger  
(christian.bezanger@gmail.com) (sauf mention contraire)

Une fois n'est pas coutume, cet article s'adresse plutôt à ceux qui ont déjà pratiqué Arduino et qui veulent aller plus loin, mais c'est peut-être aussi l'occasion de vous y mettre en reprenant l'initiation sur Locoduino; je ne détaillerai pas certaines manipulations que vous connaissez déjà, mais un fil sera ouvert sur le forum pour aider ceux qui en auront besoin. La **figure 1** montre ce que vous obtiendrez sur votre smartphone (ou votre tablette): la vitesse (valeur de PWM) est clairement affichée, un switch permet l'arrêt d'urgence et un potentiomètre horizontal (appelé slider) permet à la fois de déterminer la vitesse et le sens de marche de l'engin. Ce sens de marche est reporté par l'allumage d'une LED virtuelle verte ou rouge. Cet article insiste surtout sur la marche à suivre pour créer l'interface de commande via votre réseau WiFi. Je vais cependant donner quelques informations concernant les composants nécessaires et comment les mettre en œuvre.

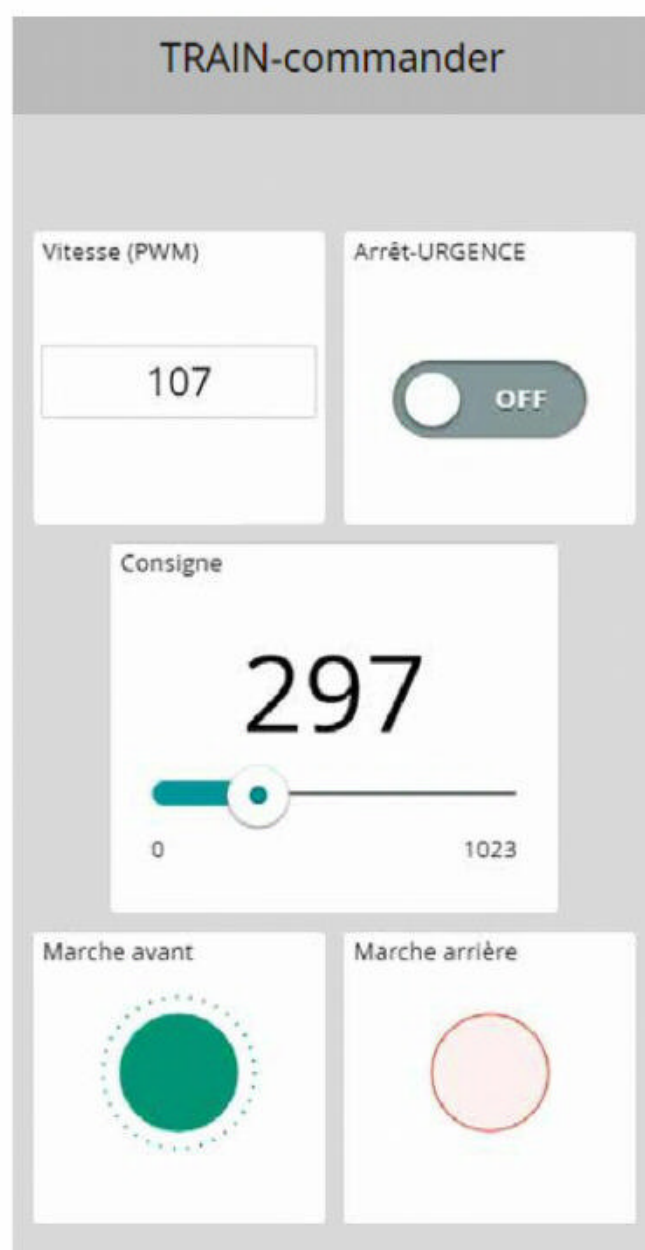
## Quel matériel est nécessaire ?

Il vous faut une carte Arduino prévue pour l'**Internet of Things (IoT)** et une carte Arduino Motor Shield qui permet de commander deux moteurs DC, c'est-à-dire deux réseaux analogiques indépendants. J'ai choisi d'utiliser la carte **Nano 33 IoT** qui est vendue 16 euros (en 2022) soit moins chère qu'une carte UNO. Vous devrez cependant souder vous-même les connecteurs (2 fois 15 broches) le long de la carte. Cette carte respecte les dimensions et le brochage des cartes Nano (voir **figure 2**), mais elle est équipée d'un processeur **SAMD21G** de 32

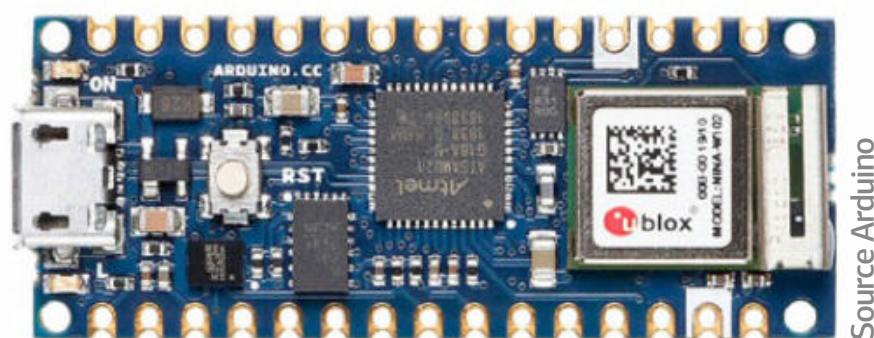
bits travaillant à 48 MHz. Elle peut se connecter en WiFi et bluetooth grâce à une puce **ublox NINA W102**. Cerise sur le gâteau, la carte dispose également d'un **IMU (Inertial Measurement Unit)**, le **LSM6DS3**. Le port USB est un port micro-USB; il faut donc avoir un câble adéquat permettant de transférer des données pour la programmer.

## Un montage économique

La carte Arduino Motor Shield (vendue 20 euros) permet de commander deux moteurs DC mais nous n'utiliserons que le canal A qui sera directement relié aux rails puisque le moteur d'une loco est en contact



**Figure 1.** Voici l'interface qui nous servira à commander notre machine en analogique.

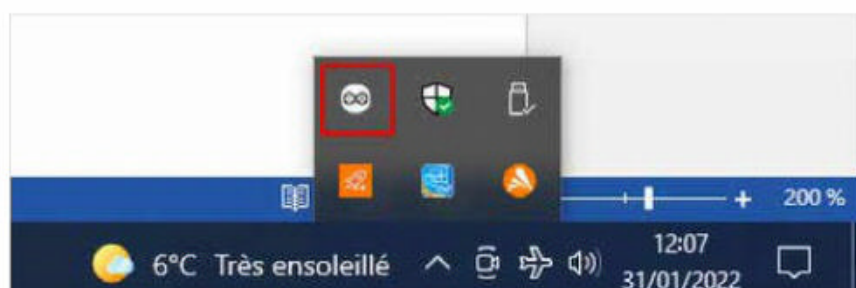
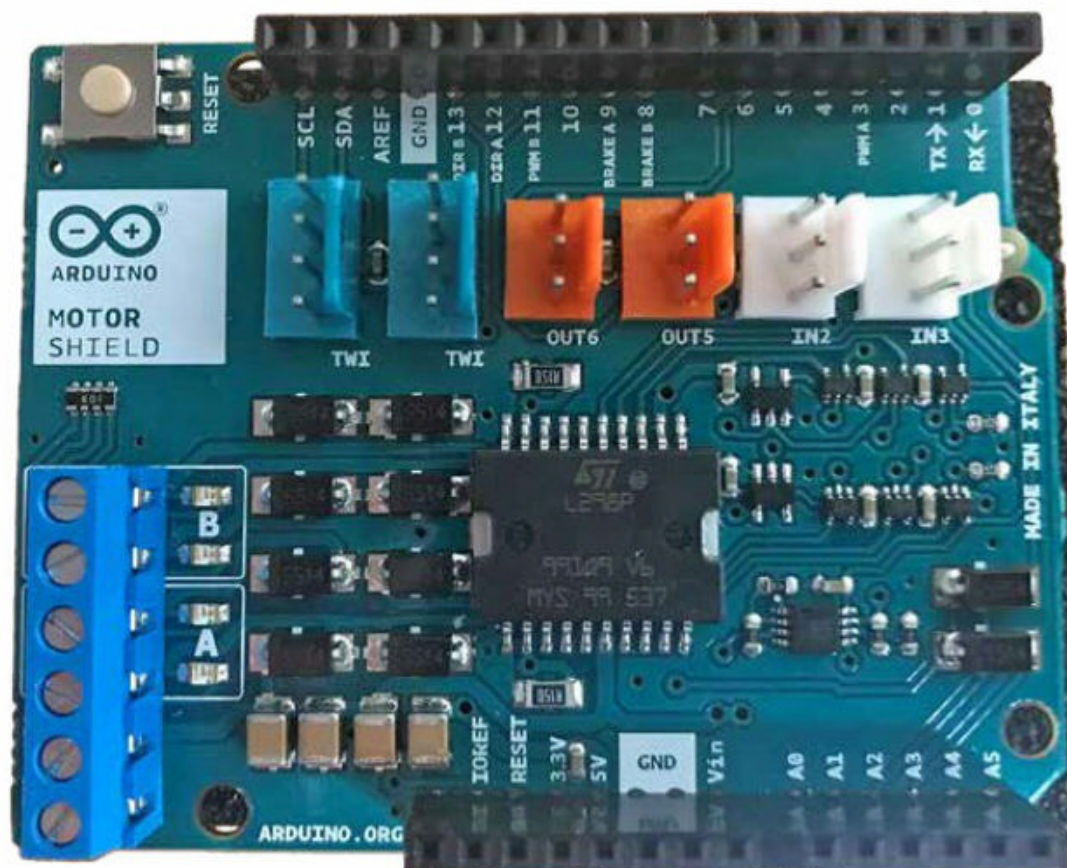


Source Arduino

**Figure 2.** Attention: cette carte travaille en 3,3 V et n'est pas résistante au 5 V!

# LE PRIX TOTAL DU MONTAGE EST DE 36 EUROS !

**Figure 3.**  
Exemple de carte  
Motor Shield.



**Figure 4.** L'icône encadré en rouge apparaît sur votre ordinateur.

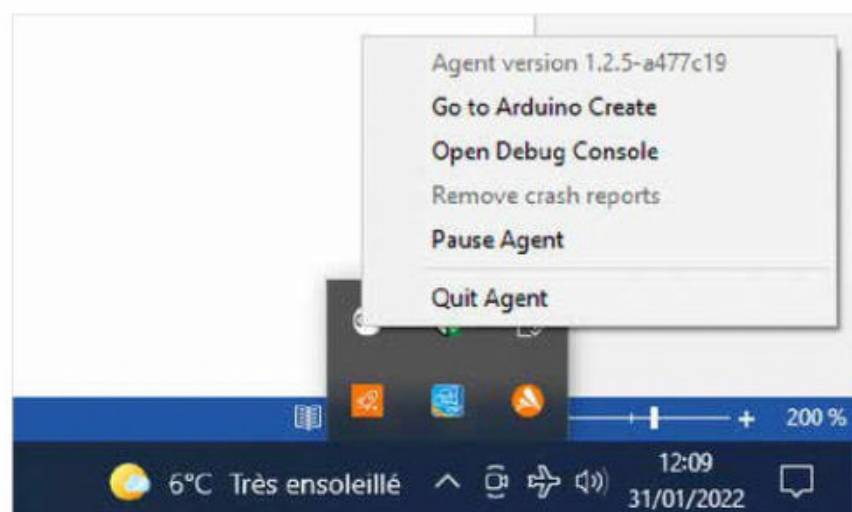
avec les rails. Pour diriger la loco, il faut définir un sens de marche et alimenter la carte avec un signal PWM (Pulse Width Modulation) généré par la carte Nano pour régler la vitesse. Cette carte shield est décrite à la page <https://store.arduino.cc/products/arduino-motor-shield-rev3> et un tuto permettant d'apprendre à l'utiliser peut être lu à la page <https://docs.arduino.cc/tutorials/motor-shield-rev3/msr3-controlling-dc-motor>. La **figure 3** montre la carte Motor Shield. Le prix de revient total de notre montage est de 36 euros, plus quelques câbles nécessaires pour relier les deux cartes. En effet, la carte Motor s'enfiche généralement sur une carte Uno, mais celle-ci est remplacée par la carte Nano 33 IoT. Les sorties de cette dernière doivent être reliées aux connecteurs de la carte Motor, tout en

se rappelant que la carte Nano fournit du 3,3 V alors que la carte Motor a besoin de 5 V; il est conseillé d'utiliser un convertisseur de signaux: cela vous coûtera quelques euros de plus, mais vous serez tranquille. Pour le canal A de la carte Motor, la sortie D12 commande la direction du mouvement de la loco et la sortie D3 fournit la PWM. La sortie D9 est utilisée pour l'arrêt d'urgence.

## Arduino IoT Cloud

L'utilisation d'Arduino IoT Cloud permet de s'affranchir de la difficulté de programmer une interface utilisable en WiFi. La première chose à faire est d'installer sur votre ordi-

**Figure 5.** En cliquant sur l'icône, le menu se déroule.



nateur un plug-in appelé **Arduino Create Agent** qui vous permettra de connecter votre carte à la plateforme Arduino (via le câble USB) et donc de la programmer. Tapez « Create agent » dans la barre de recherche du site Arduino pour voir comment l'installer. Une fois ce plug-in installé, vous devez obtenir une icône sur votre ordinateur illustrée par la **figure 4**. Vous passez par cette icône pour rejoindre la plateforme Arduino (Go to Arduino Create comme le montre la **figure 5**). Pour utiliser le Cloud d'Arduino, vous devez aussi créer un compte (une adresse mail, un pseudo et un mot de passe). Une fois votre compte créé, vous devez choisir un niveau ►►

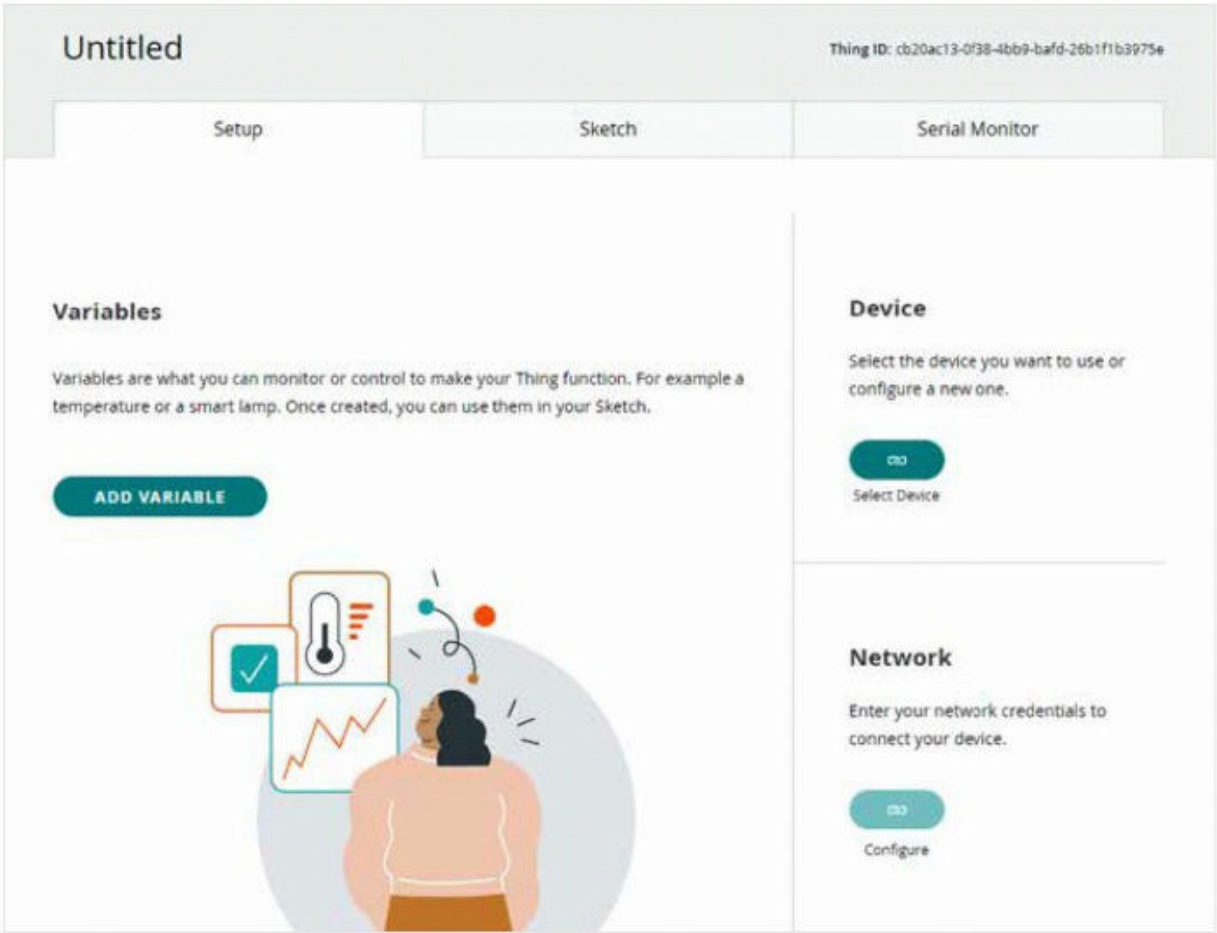


Figure 6. L'interface invite à définir les variables et le réseau WiFi.

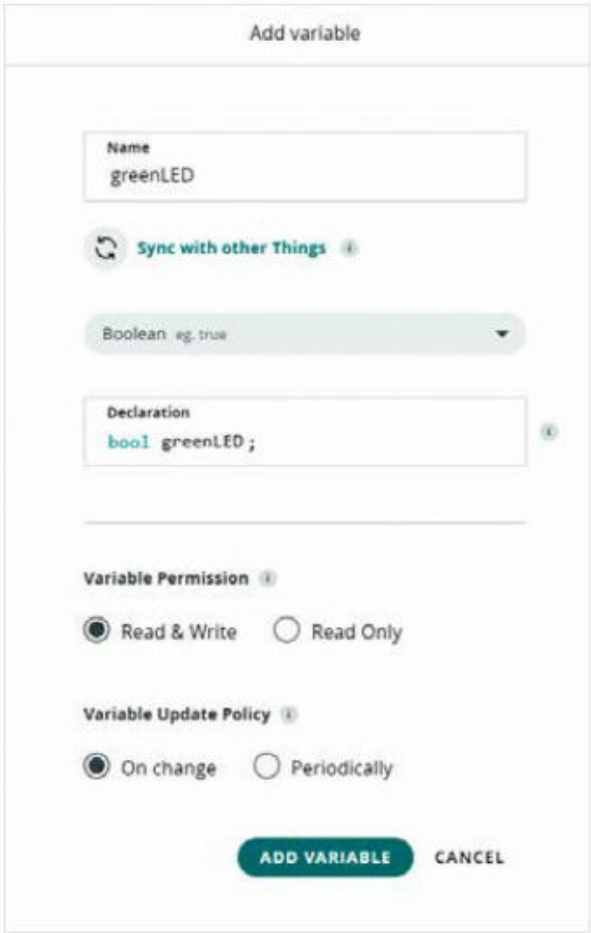


Figure 7. Interface de création d'une variable.

► d'abonnement (Cloud Plan); le premier niveau est gratuit et permet de découvrir le monde des objets connectés. Mais il ne permet pas de faire grand-chose; les niveaux suivants sont payants mais permettent plus de choses. Vous êtes maintenant prêt à utiliser Arduino IoT Cloud et à créer votre premier objet connecté.

### Comment fonctionne le Cloud ?

Le Cloud va vous permettre de **contrôler des variables**, et ces variables contrôleront des capteurs ou des actuateurs. Dans l'exemple de cet article, ces variables contrôleront la direction du mouvement et le rapport cyclique de la PWM en contrôlant les sorties de la carte Nano. En conséquence, pour travailler avec le Cloud, il est nécessaire de penser aux variables qui serviront à contrôler les périphériques. Et pour modifier ces variables (donc l'état des périphériques), il va falloir créer un tableau de bord comme celui de la **figure 1**. La plateforme Arduino génère automatiquement un sketch (programme) chaque fois que vous créez une variable (ou modifie le sketch en cours). Tout est déjà prévu pour vous connecter au Cloud et la seule chose à faire est de rajouter quelques lignes de programme dans ce sketch, en fonction de ce que vous voulez faire. Pour plus d'explications sur l'utilisation d'Arduino IoT Cloud, je vous conseille de lire cette page <https://docs.arduino.cc/cloud/iot-cloud/tutorials/iot-cloud-getting-started>.

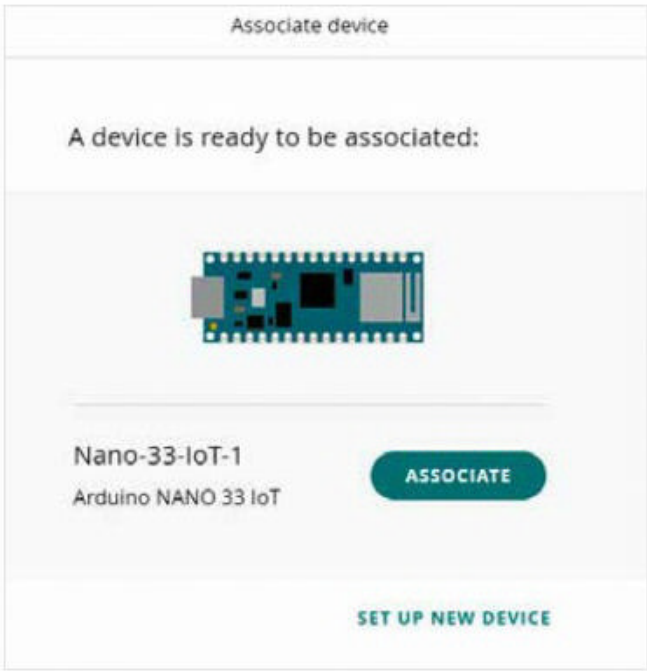


Figure 8. Ici, la carte connectée est déjà connue (appelée Nano-33-IoT-1).

### Création de notre objet connecté

Quand on veut créer un objet connecté, il faut définir des variables, associer à l'objet une carte connectable au WiFi et prévue pour l'IoT (Device), et définir le réseau WiFi à utiliser (Network). Tout ceci est montré par la **figure 6**. Remplacez « Untitled » par le nom de votre objet, par exemple « TRAIN-commander ». Choisissez les variables à ajouter au projet; ici, une variable de type int (entier sur 16 bits) pour la valeur du potentiomètre appelée « potValue », et deux variables de type boolean pour les deux LED virtuelles (si true, la LED est allumée, si false, elle est éteinte). On les appelle « greenLEDstate » et « redLEDstate ». Ces variables définiront également le sens de marche de la loco. Une

variable « vitesse » de type int sera affichée sur le tableau de bord (elle correspond à la valeur de PWM). Enfin, une variable booléenne « stopState » prendra en compte une demande d'arrêt d'urgence. Chaque variable créée implique une modification automatiquement générée du sketch. La **figure 7** montre comment créer une variable: on définit son nom, on choisit son type dans une liste proposée, et on sélectionne « Read & Write » ou bien « Read Only » suivant le cas (ici « Read & Write »). Sélectionnez aussi « On Change », puis « ADD VARIABLE ». Le choix de la carte (Device) est très simple s'il s'agit d'une carte Arduino. Ici, la carte connectée est déjà connue (appelée Nano-33-IoT-1) et proposée, mais on peut associer un nouveau composant (**figure 8**). On a ensuite la main

**ATTENTION**  
Pour se connecter au Cloud d'Arduino, le firmware de la puce WiFi NINA doit être supérieur à la version 1.4.4: une mise à jour peut être nécessaire et ceci est très simple à faire avec l'IDE. Les opérations sont expliquées sur le site d'Arduino.

```

103 /*
104  Since StopState is READ_WRITE variable, onStopStateChange() is
105  executed every time a new value is received from IoT Cloud.
106  */
107 void onStopStateChange() {
108  // Add your code here to act upon StopState change
109  if(stopState == true) {
110    vitesse = 0;
111    greenLEDstate = true;
112    redLEDstate = true;
113  }
114 }
115 }

```

```

52 /*
53  Since PotValue is READ_WRITE variable, onPotValueChange() is
54  executed every time a new value is received from IoT Cloud.
55  */
56 void onPotValueChange() {
57  // Add your code here to act upon PotValue change
58  vitesse = potValue - 512;
59  if(vitesse >= -50 && vitesse <= 50) {
60    greenLEDstate = false;
61    redLEDstate = false;
62  }
63  if(vitesse < -50) {
64    greenLEDstate = true;
65    redLEDstate = false;
66  }
67  if(vitesse > 50) {
68    greenLEDstate = false;
69    redLEDstate = true;
70  }
71  vitesse = abs(vitesse);
72  if(vitesse <= 50) {vitesse = 0;}
73  vitesse = vitesse / 2;
74  if(vitesse > 240) {vitesse = 240;}
75  Serial.println(vitesse);
76 }
77 }

```

Figure 9 & 10. Modifications à apporter au sketch.

pour configurer l'accès au réseau WiFi et ceci ne présente aucune difficulté puisqu'il suffit d'entrer le nom du réseau et le mot de passe pour y accéder.

## Le programme

Notre objet est créé et il est temps de s'intéresser au sketch (programme) généré automatiquement avec les variables choisies, et de le modifier pour qu'en bougeant le curseur du potentiomètre, on définisse vitesse et sens de circulation. Pour cela, on va rendre notre potentiomètre intelligent. Le point milieu du potentiomètre correspondra à l'arrêt du train (avec une certaine plage). Si on bouge le curseur vers la gauche, on définit le sens marche avant, vers la droite, marche arrière. Or la position du curseur correspond à un nombre compris entre 0 et 1023, et un petit traitement est nécessaire pour obtenir le rapport cyclique de la PWM réglant la vitesse (voir fiche pratique III.61 de février 2020). Le traitement est simple: vitesse est égale à potValue - 512 (elle est donc comprise entre -512 et 511). Si vitesse est négative, le sens est en avant, si elle est positive, le sens est en arrière. On prend alors la valeur absolue et on la divise par deux pour rester dans une

plage compatible pour une PWM (entre 0 et 255). Enfin, on limite la vitesse à 240; ainsi la valeur de PWM est toujours comprise entre 0 et 240 (vitesse maximale). La vitesse et le sens permet de commander la carte Nano qui commande à son tour la carte Motor. Les **figures 9 et 10** montrent ce qu'il faut ajouter au sketch généré automatiquement, dans les fonctions créées pour les variables. Je n'ai pas mis les commandes de la carte Motor, mais vous n'aurez aucun mal à les programmer en fonction des valeurs de vos variables. Inspirez-vous du tuto cité en début d'article.

## Création du tableau de bord

Ensuite, il faut définir un tableau de bord (appelé dashboard) qui servira à contrôler l'objet. Ceci est vraiment très simple puisqu'il suffit d'utiliser des widgets prédéfinis et de les relier aux variables. On commence par choisir un afficheur pour la donnée vitesse, puis un switch (interrupteur) qui, mis sur ON, arrête la loco en urgence. On choisit un slider (potentiomètre) qu'on dispose horizontalement pour plus de visibilité et on règle ses valeurs extrêmes (de 0 à 1023, valeurs choisies comme dans la fiche pratique citée

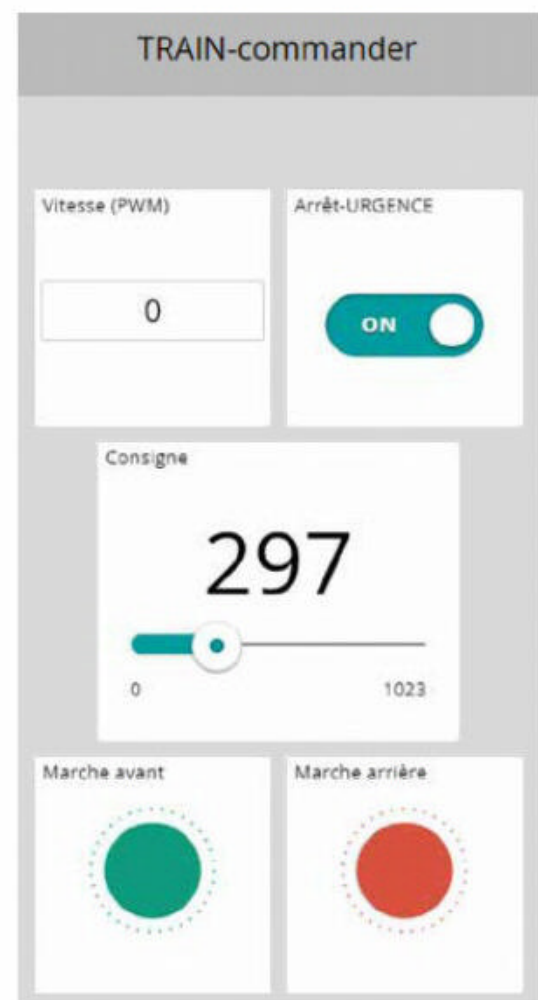


Figure 11. L'interface est intuitive et peut être organisée à votre guise.

plus haut). Enfin, deux LED virtuelles, une verte et une rouge, rappellent le sens de marche. Chaque widget doit avoir son nom et **être lié à une variable**. Le widget slider est appelé « Consigne » et est lié avec la variable « potValue », alors que les widgets LED sont appelés « Marche avant » et « Marche arrière » et sont liés avec les variables « greenLEDstate » et « redLEDstate ». La variable « greenLEDstate » permet de régler la sortie D12 définissant le sens de marche. L'afficheur est appelé « Vitesse (PWM) » et est lié à la variable « vitesse » qu'il affiche. Enfin, le switch est appelé « Arrêt-URGENCE » et est lié à la variable « stopState »; si le switch est sur ON, alors la vitesse est mise à 0 et les deux LED sont allumées (voir **figure 11**) (et la sortie D9 est commandée). Les différents widgets peuvent être organisés comme bon vous semble et vous pouvez aussi voir comment votre dashboard se présentera sur votre smartphone.

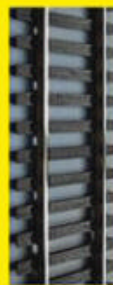
## En résumé : des avantages indéniables

L'intérêt d'Arduino IoT Cloud est qu'on a en fait très peu de code à écrire: on définit des variables et un tableau de bord pour contrôler l'objet. On devient en quelque sorte des designers et le sketch est généré automatiquement pour gérer la connexion WiFi. Il faut tout de même le compléter pour que l'objet fasse ce pour quoi on l'a conçu. Vous verrez, c'est amusant et cela marche assez bien. ■



# WWW. TRAIN-MODELISME.COM

*La passion des trains miniatures !*



Les rames voyageurs  
Les locomotives  
Les autorails  
Les voitures et wagons  
Les rames urbaines  
Les trains de travaux



Les trains à crémaillère  
La gamme économique  
Les rails et accessoires...  
Le digital et l'alimentation...  
Le décor, la caténaire, la signalisation...  
Les coffrets, l'entretien...



*Prix compétitifs, STOCK important avec gestion en temps réel, expéditions 24 H !*

*Commandes en ligne par CB ou E-card, par courrier et par fax 02-54-28-75-01 !*



Train Modélisme Rte d'Angles 36220 Néons/Creuse Tél 02-54-28-59-66 RCS 433 831 039 trainmodelisme @orange.fr

## BOURSE FERROVIAIRE DE L'ESPACE CHARENTON 75012 PARIS

**LE 19 JUIN 2022 DE 9H30 À 14H**

**DANS LE RESPECT DES MESURES SANITAIRES  
VOIR LES INFOS SUR WWW.PIERREDOMINIQUE.COM**

**LE SEULE MANIFESTATION  
QUI RESTE EN RÉGION PARISIENNE !**

ESPACE CHARENTON  
327 RUE DE CHARENTON 75012 PARIS  
Métro ligne 8 ou Tramway : Porte de Charenton

Entrée : 5,50 euros Buvette sur place  
Contact : SDMF tél. 06 25 43 93 77 sdmf@wanadoo.fr

## M. CITERNE

Trains Miniatures N - HO

Occasions - Dépôt-Vente

Digital • Roco • Lenz • Zimo • Trix

## VENTE PAR CORRESPONDANCE

*Ouvert du mardi au samedi - 9h-12h30 et 14h-19h*

**21 Bd du Temple - 75003 Paris**

**Tél./fax 01.42.78.00.16**

*Métro - République - Filles du Calvaire, Bus 20 - 65 - 96*

**PK**  
n°3  
Photographie ferroviaire

DIDIER DELATTE

## VAPEUR D'HIER DANS LE MONDE D'AUJOURD'HUI



Après les trains de Sylvain Assez vus de sa cabine de conduite (PK01), puis les paysages ferroviaires de Pierre Julien (PK02) ce nouveau numéro de PK vous propose cette fois de découvrir les vapeurs insolites de Didier Delatte, un "trainspotter" au sens noble du terme. Passionné de photographie et de trains, il a parcouru la planète à la recherche des dernières locomotives à vapeur encore en service. Cela donne des clichés étonnants, mêlant les époques et les paysages, invitation au voyage dans le temps et dans l'espace. 120 photos exceptionnelles de Didier en Asie du Sud-Est, en Chine, en Afrique Australe, en Amérique du Sud, dans les Balkans...

**RÉSERVER-LE DÈS MAINTENANT SUR TRAINS.LRPRESSE.COM AU PRIX DE 19,50 € (ABONNÉS 16 €). PARUTION FIN AVRIL.**

**PK**  
n°3  
Photographie ferroviaire



**VAPEUR D'HIER DANS  
LE MONDE D'AUJOURD'HUI**

DIDIER DELATTE