

Photon RedBoard : de l'Arduino pour l'internet des objets

LOÏC JOSSE ^[1]

Les objets connectés font désormais partie de notre quotidien. C'est pourquoi l'utilisation des modules compatibles Arduino est devenue incontournable pour élaborer nos projets. Sparkfun nous propose son module Photon RedBoard, spécialement étudié pour connecter nos prototypes.

Encore une énième carte de type Arduino me direz-vous ! Alors entrons tout de suite dans le vif du sujet. Imaginons que vous envisagiez la réalisation d'un prototype commandé depuis internet *via* un smartphone, une tablette ou un ordinateur. Ce prototype devra être équipé d'un module wifi en liaison avec un serveur web. Les solutions envisagées sont souvent un ensemble de cartes « sorties du tiroir » ou l'utilisation d'un ordinateur PC plus onéreux...

Présentation de la carte

La Photon RedBoard ¹ est conçue à partir du module Photon développé par la société Particle. La plate-forme *cloud* de Particle possède une passerelle sécurisée vous permettant d'interagir entre votre application et votre module. La carte adopte un connecteur et utilise un code source compatible Arduino. Elle est équipée d'un processeur ARM Cortex M3 cadencé à 120 MHz, d'une

mots-clés

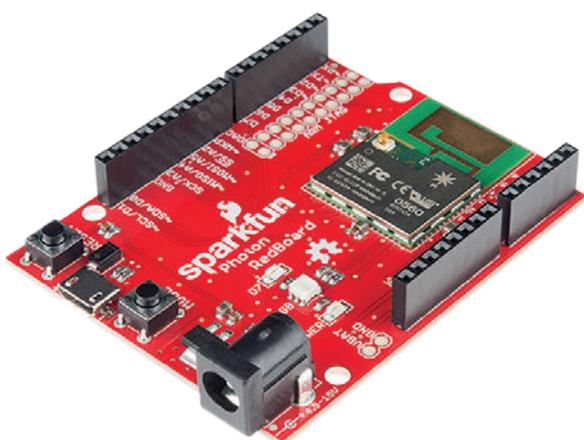
programmation,
logiciel,
projet

mémoire flash de 1 Mo et d'un module wifi. Sa particularité est de pouvoir développer très rapidement et très simplement des applications IoT (*Internet of Thing*) pouvant également être pilotées depuis un smartphone ou une tablette (*via* une application iOS™ ou Android™) ².

Le niveau de tension d'une sortie à l'état logique 1 est de 3,3 V ; en revanche, certaines entrées logiques tolèrent un niveau à 5 V. Les entrées analogiques doivent être comprises entre 0 et 3,3 V et ont une précision de 12 bits (0 à 4095). Les sorties PWM sont sur 8 bits, mais vous pouvez augmenter la précision en utilisant les sorties DAC sur 12 bits (tableau ³).

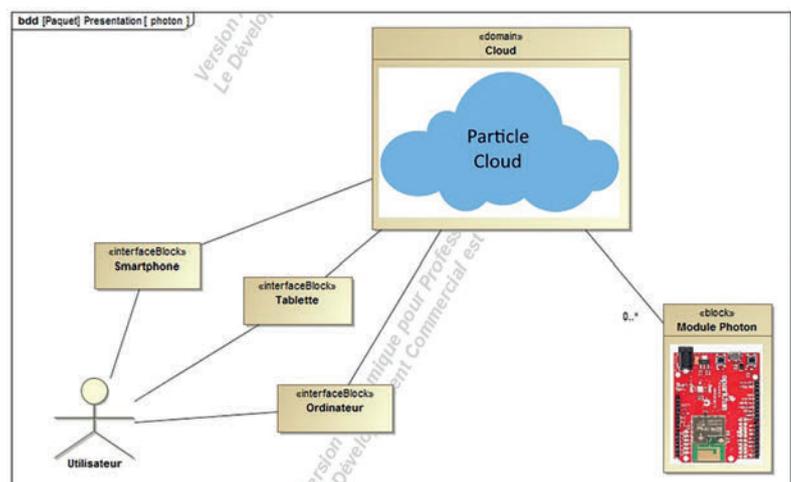
Première option

Nous vous proposons d'utiliser une application Particle téléchargée sur le site depuis votre smartphone ⁴. À partir de celle-ci, vous devez inscrire votre carte dans le *cloud* en lui donnant l'accès au wifi (voir En ligne, lien 1, section *Tinkering*). Une fois la carte enregistrée, il reste à flasher le code « Tinker » et cela se fait à distance, en appuyant sur une touche du smartphone. Vous pouvez désormais simplement commander les entrées/sorties du module depuis votre smartphone. Vous disposez des fonctions `digitalRead()`, `digitalWrite()`, `analogRead()` et `analogWrite()`. Déclencher un actionneur ou lire la valeur d'un capteur devient un jeu d'enfant.



¹ La carte Photon RedBoard

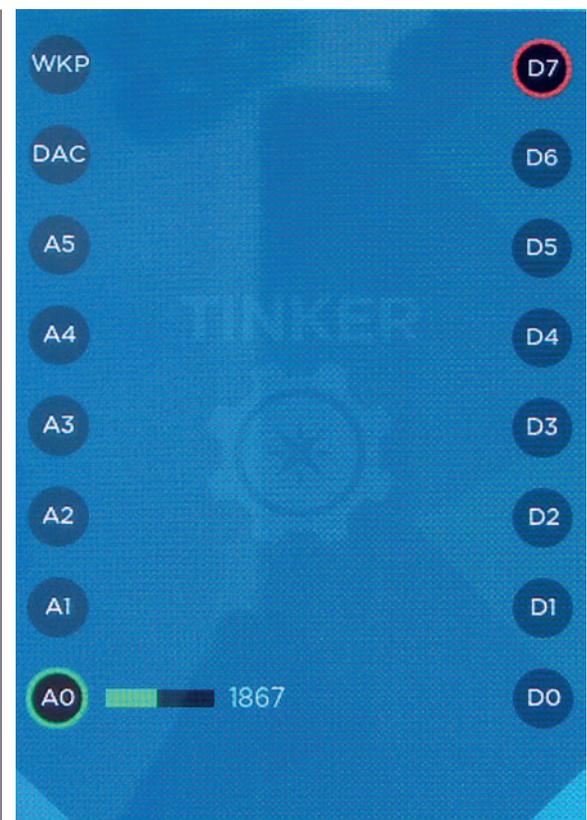
[1] Enseignant SII option information et numérique, lycée Louis-le-Grand, Paris



² Présentation du contexte de programmation

Arduino Uno R3	Photon RedBoard	Tolérance au 5 V
IOREF		
RESET	RESET	
3.3V	3.3V	
5V	V-USB	
GND	GND	
GND	GND	
VIN	VIN	
A0	A0	X
A1	A1	X
A2	A2 /SS1	X
A3	A3/SCK1/ DAC2	
A4 /SDA	A4 /MISO1	
A5 /SCL	A5 /MOSI1	
D0/RX	RX/ PWM	X
D1/TX	TX/ PWM	X
D2	D2/ PWM /MOSI3	X
D3/ PWM	D3/ PWM /MISO3	X
D4	D4/SCK3	X
D5/ PWM	D5/SS3	X
D6/ PWM	D6	X
D7	D7	X
D8	A6 / DAC	
D9/ PWM	A7/WKP/ PWM	X
D10/SS/ PWM	A2 /SS1	X
D11/MOSI/ PWM	A5/MOSI1/ PWM	X
D12/MISO	A4/MISO1/ PWM	X
D13/SCK	A3 /SCK1/ DAC2	
GND	GND	
AREF		
A4 /SDA	D0/SDA/ PWM	X
A5 /SCL	D1/SCL/ PWM	X

3 Comparaison des broches de la Photon RedBoard en regard de celles de l'Arduino Uno, version 3

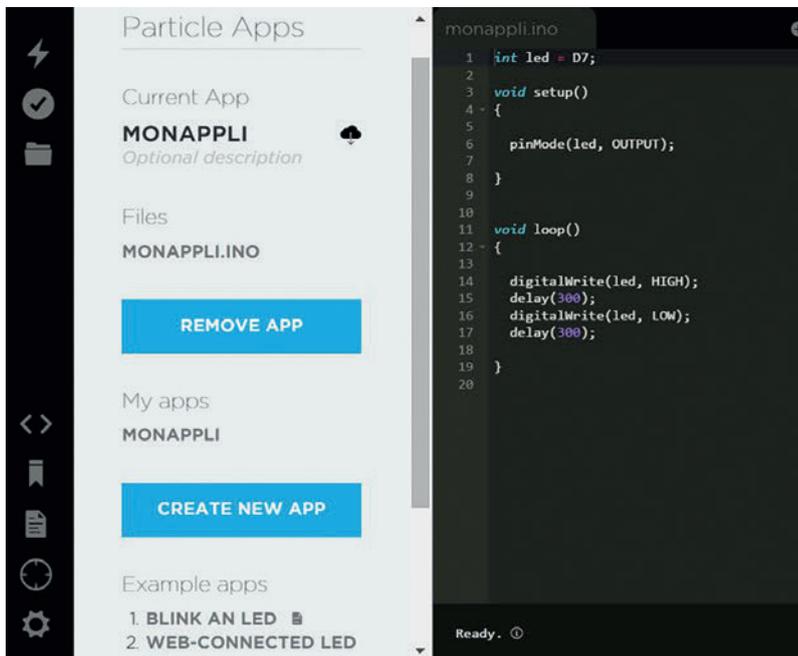


4 La plateforme Particle tinker

Deuxième option

Nous choisissons d'écrire une application et de la transférer dans la carte, toujours à distance ! Connectez-vous à internet pour accéder à l'environnement de développement intégré EDI (IDE en anglais) « Particle Apps » (voir En ligne, lien 2). Il vous faudra être inscrit sur le *cloud* de Particle tout comme votre carte. L'interface propose les fonctions principales d'un programme Arduino : **void setup()** et **void loop()** 5. Taper votre code, nommer/sauvegarder/vérifier et flasher-le, au travers du *cloud*, dans la carte. Constaté le bon fonctionnement.

À travers cette option, nous avons travaillé uniquement par internet, mais vous pouvez aussi télécharger l'application « Particle Dev » (voir En ligne, lien 3) pour saisir le code ; le transfert vers la carte se fera cependant *via* le *cloud*.



5 La partie programme : Particle build

Application

Nous allons réaliser une page web disposant de boutons radios pour commander une sortie logique et afficher la valeur d'une entrée analogique. Les codes HTML, CSS et Photon sont fournis ci-après.

HTML

```
<html lang="fr">
<head>
  <meta content="text/html; charset=UTF-8" http-equiv="content-type">
  <meta content="L. JOSSE" name="author">
  <title>Appli Photon</title>
  <link rel="stylesheet" href="photon.css">
</head >
<body>
<div contenu>
  <center>
    <form action="https://api.particle.io/v1/devices/N°carte
    /cmd?access_token=N°jeton" method="POST">
      <h1>Première application Photon</h1>
       <br><br>
      <table>
        <tr><td><input type="radio" name="args" value="on"> </td>
          <td>Allumer LED7</td></tr>
        <tr><td><input type="radio" name="args" value="off"> </td>
          <td>Éteindre LED7</td></tr>
        <tr><td><input type="radio" name="args" value="an0"> </td>
          <td>Lire entrée AN0</td></tr>
      </table><br><br>
      <input type="submit" value="Valider">
    </form>
  </center>
</div>
</body>
</html>
```

Photon

```
int led7pin = D7;
int an0pin = A0;
int an0value = 0;
void setup()
{
  pinMode(led7pin, OUTPUT);
  pinMode(an0pin, INPUT);
  Spark.function("cmd", photoncmd);
  digitalWrite(led7pin, LOW);
}
void loop()
{
}
int photoncmd(String command)
{
  if (command=="on")
  {
    digitalWrite(led7pin,HIGH);
    return 1;
  }
  else if (command=="off")
  {
    digitalWrite(led7pin,LOW);
    return 0;
  }
  else if (command=="an0")
  {
    an0value=analogRead(an0pin);
    return an0value;
  }
  else
  {
    return -1;
  }
}
```

CSS

```
body {
  margin: 10mm;
  background-color: #ccccff;
  border:1px solid black;
}
contenu {
  margin : 5mm;
}
tr {
  font-style:italic; font-weight:bold;
}
h1 {
  color:red;
}
```

Première application Photon



- Allumer LED7
- Éteindre LED7
- Lire entrée AN0

6 Un exemple de résultat dans une page web

Les codes ne présentent pas de difficultés majeures, mais veillez à bien repérer votre numéro de carte et celui du jeton délivré par Particle et à les remplacer dans le code HTML (en rouge). Vous trouverez ces numéros dans l'IDE Particle Build, onglets « Settings » et « Devices ».

En ligne

1. <https://learn.sparkfun.com/tutorials/photon-redboard-hookup-guide>
2. <https://build.particle.io/build>
3. <https://www.particle.io/dev>
4. <https://learn.sparkfun.com/tutorials/photon-development-guide/introduction>
5. <https://docs.particle.io/guide/getting-started/intro/photon/>

Retrouvez tous les liens sur <http://eduscol.education.fr/sti/revue-technologie>

Le code Photon doit être transféré dans la carte ; il suffit ensuite d'ouvrir la page HTML, de sélectionner l'une des radios et de valider.

Pour ceux qui désirent aller plus loin avec les outils du « Cloud Particle », sachez que vous pouvez aussi commander la carte en lignes de commandes ou encore intégrer l'environnement Particle à celui de Google Application.

Les cartes munies du module Photon sont distribuées chez Lextronic. La RedBoard coûte environ 43 euros ; c'est moins qu'une carte Arduino plus un *shield* wifi ou qu'une Arduino Yu. ■

[PUBLICITÉ]

technologie

s'affiche sur Éduscol...



Vous y trouverez :

- Le sommaire détaillé de chaque nouveau numéro
- Des liens vers de nombreux articles
- Un lien vers les archives de la revue

Vous pourrez y télécharger :

- Des articles d'archives de la revue
- L'éditorial et le Technomag de chaque numéro

... mettez-le dans vos favoris !

<http://eduscol.education.fr/sti/revue-technologie>

Rejoignez-nous sur **Twitter** : https://twitter.com/technologie_sti

et sur **Facebook** : <https://www.facebook.com/technologie.sti>

