

QuickPi

Programmer des objets connectés en quelques minutes !

Le projet QuickPi propose un support facile à utiliser pour concevoir et tester des objets connectés. QuickPi peut être utilisé comme support motivant pour aborder la programmation et, notamment, dans le cadre de cours sur les objets connectés.

Facile à manipuler : QuickPi s'appuie sur un mini-ordinateur standard (appelé Raspberry Pi) auquel on peut connecter de nombreux capteurs (et moteurs) à l'aide de cables standardisés, ne nécessitant aucune soudure (technologie Grove).

Facile à programmer : avec QuickPi, les programmes sont écrits directement dans un navigateur (pas de logiciel spécifique à installer), en utilisant un langage standard (Scratch, Blockly, ou Python), et les codes sont transmis sur le mini-ordinateur par Wifi en une fraction de seconde.

Facile à tester : QuickPi offre un simulateur permettant de tester intégralement un programme même sans disposer de matériel. De plus, QuickPi permet de l'exécution pas-à-pas, non seulement dans le simulateur mais également avec le matériel final, ce qui permet d'étudier très facilement le comportement du programme et de comprendre ses erreurs.

Des possibilités infinies : QuickPi est basé sur un Raspberry Pi, qui est un véritable ordinateur, et peut se programmer en Python, un langage de programmation généraliste et très largement diffusé. Cette combinaison ouvre de très nombreuses possibilités, en pouvant s'appuyer sur une puissance de calcul importante ainsi que sur des bibliothèques de code existantes.

Tout public : même si QuickPi s'adresse en priorité aux élèves de collège et de lycée, il est suffisamment simple à utiliser pour des élèves de fin de primaire, et en même temps offre suffisamment de possibilités pour enseigner à des élèves post-bac.

Intégrable dans des plateformes d'enseignement : L'environnement QuickPi est conçu pour s'intégrer facilement dans des plateformes en ligne d'apprentissage, par exemple dans des MOOC sur les plateforme Fun, EdX, Moodle ou France-ioi. Selon les plateformes, les enseignants peuvent suivre l'activité de chaque élève, les conseiller à distance, organiser des examens en temps limité, ou gérer des projets par groupes.

Exercices corrigés automatiquement : Il est possible de créer des exercices évalués automatiquement. Le simulateur est alors utilisé pour tester, sur certains scénarios d'utilisation, si le programme se comporte comme requis en fonction des suites de valeurs fournies aux capteurs.

Prix abordable : le matériel de base coûte environ 50 euros; une version avec des capteurs et actuateurs un peu plus avancés coûte environ 100 euros, cependant notez qu'il n'est en général

pas nécessaire de disposer de tous les capteurs pour chaque Raspberry Pi, car il est rare de les utiliser tous en même temps. Surtout, les élèves peuvent tout préparer sur le simulateur avant de tester rapidement sur le matériel, donc un petit nombre de Raspberry Pi peut facilement être partagé entre les élèves de toutes la classe.

Vidéo de présentation : http://www.france-ioi.org/tmp/demo_quick_pi.mp4

QuickPi : un matériel facile à manipuler

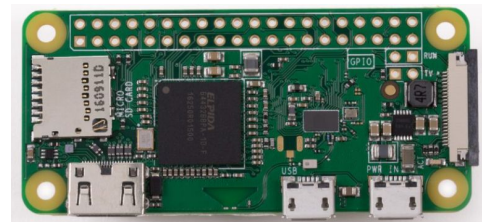
Raspberry Pi

Le Raspberry Pi a la puissance et l'architecture d'un véritable ordinateur (contrairement à des cartes basées sur des micro-contrôleurs comme l'Arduino ou Micro::bit). Cette puissance ouvre de très nombreuses possibilités, en particulier toutes les applications qui peuvent demander une puissance de calcul importante : traitements d'images, stockage d'une base de données, reconnaissance et synthèse vocale, etc. De très nombreuses bibliothèques Python ou autres peuvent être installées et rendues accessibles.

Deux versions sont utilisables avec QuickPi (avec module wifi intégré):



Raspberry Pi 3 Model A+



Raspberry pi Zero W

Le Raspberry Pi se connecte par Wifi à internet. Il s'agit donc réellement d'un objet connecté, avec tout ce que cela implique. En particulier, on peut écrire des programmes qui se connectent à des sites en ligne, pour récupérer diverses informations : prévisions météo / pollution, dernières nouvelles, messages postés sur twitter, etc.

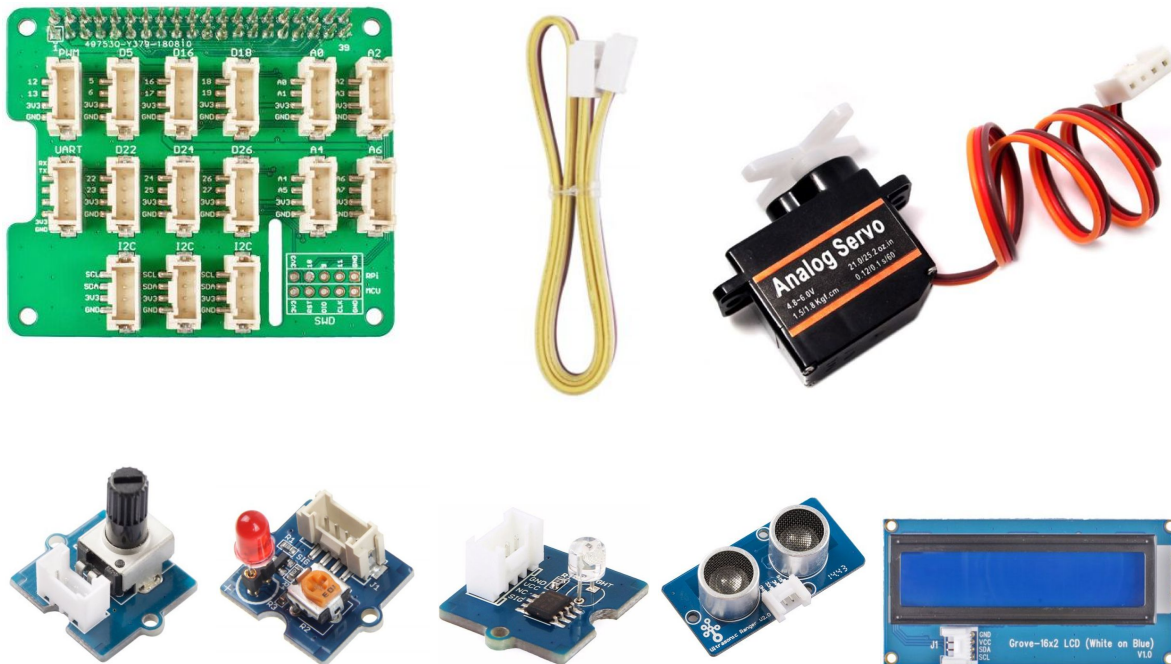
Inversement, le Raspberry Pi peut transmettre des données sur des serveurs : logs de la température, humidité, etc. signaler une intrusion, etc. tout est imaginable. Plus généralement, en exploitant la possibilité de recevoir et d'émettre des données, plusieurs Raspberry Pi, y compris situés à des kilomètres les uns des autres peuvent ainsi communiquer au travers du nuage, pour s'échanger des messages, ou collaborer. Les possibilités sont sans limites !

L'interface Grove

Le système Grove permet de connecter des composants, capteurs ou actuateurs, à l'aide de câbles standards, sans effectuer aucune soudure, et sans risquer de se tromper en branchant les câbles. Cette approche permet d'initier les élèves à l'aspect "lien avec le matériel" de l'informatique, sans pour autant perdre un temps considérable à réaliser des soudures et à vérifier des branchements.

Le système Grove est assez populaire, car il a été conçu à la fois sur Raspberry Pi et Arduino et a ainsi rapidement conquis un large public. Un très grand nombre de composants peuvent donc être achetés auprès de divers fournisseurs, à des prix compétitifs (voir la section sur les prix).

Concrètement, la carte Grove est une plaque à attacher au Raspberry Pi, exposant 15 connecteurs. À chacun d'eux, on peut attacher un composant compatible Grove en le reliant par un câble.



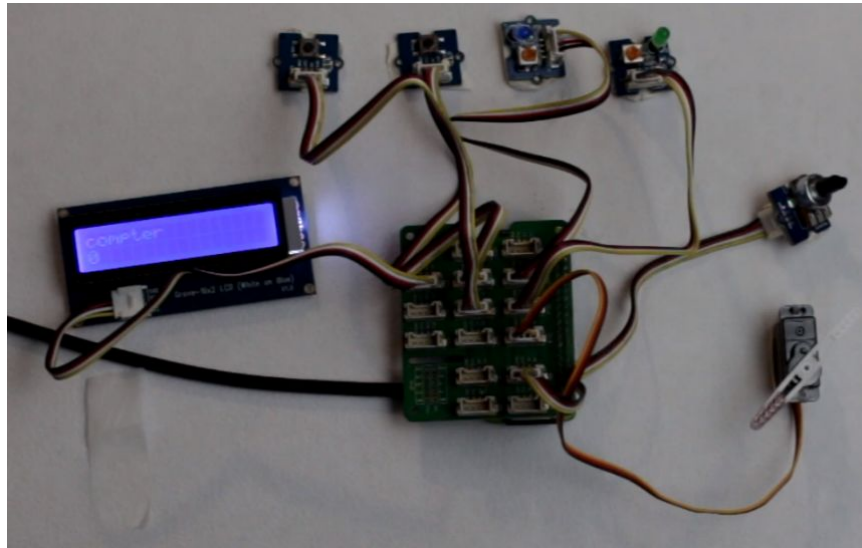
Exemples de composants

Voici une liste de capteurs populaires : bouton, interrupteur, potentiomètre, joystick, capteur de température, de luminosité, d'humidité, accéléromètre, compas, mesure distance (par ultrasons), détecteur de collisions, capteur de poussière, d'eau, détecteur de mouvements, mesure de la qualité de l'air, microphone, caméra, mesure de conductivité de la peau, détecteur de vibrations, capteur tactile, infrarouge, GPS, lecteur RFID, détecteur de gaz (hydrogène, méthane, alcool, butane, ...), détecteur de CO2, détecteur de ligne dessinée au sol, détecteur d'ultraviolets, interrupteur magnétique, détecteur de gestes, d'électricité, de battements du

coeur, baromètre, mesure de couleur, détecteur de flamme, capteur d'empreintes digitales, mini LIDAR, etc.

Voici une liste d'actuateurs populaires : LED, buzzer, servo-moteur, moteur, écrans LCD divers, vibreur, barres de LED, LED infrarouge, haut parleur, électro-aimant, atomiseur d'eau, ventilateur.

Les possibilités de projets en lien avec diverses disciplines sont illimitées !



raspberry Pi Zero G + carte Grove
+ 2 boutons, 2 LED, 1 potentiomètre, 1 servo moteur, 1 écran LCD.

De véritables objets connectés pour les élèves

Avec QuickPi, il ne s'agit pas de faire semblant de travailler sur des objets connectés : les élèves fabriquent eux-mêmes leurs objets en choisissant leurs capteurs et actuateurs, et ceux-ci peuvent réellement se connecter à internet. La même architecture est utilisée dans l'industrie, donc même si tout est fait pour que cela soit simple, l'élève n'aura au final pas l'impression de manipuler un jouet pour enfants.

Préparation du matériel en quelques minutes

Une fois le matériel livré, l'enseignant peut préparer chaque Raspberry Pi en quelques minutes:

- Chargement de notre version du système d'exploitation sur SD-Card
- Configuration du réseau wifi de l'établissement, ou branchement du pont wifi sur un câble ethernet et utilisation du réseau wifi ainsi créé.
- Affectation d'une adresse IP à chaque Raspberry Pi
- Insertion de la SD-Card dans le Raspberry

- Connexion de la carte Grove au Raspberry Pi
- Inscription de l'adresse IP sur une étiquette collée au Raspberry Pi

Tout cela est facilité grâce à notre guide d'installation.

QuickPi : un support facile à programmer

Accès immédiat à l'environnement de développement

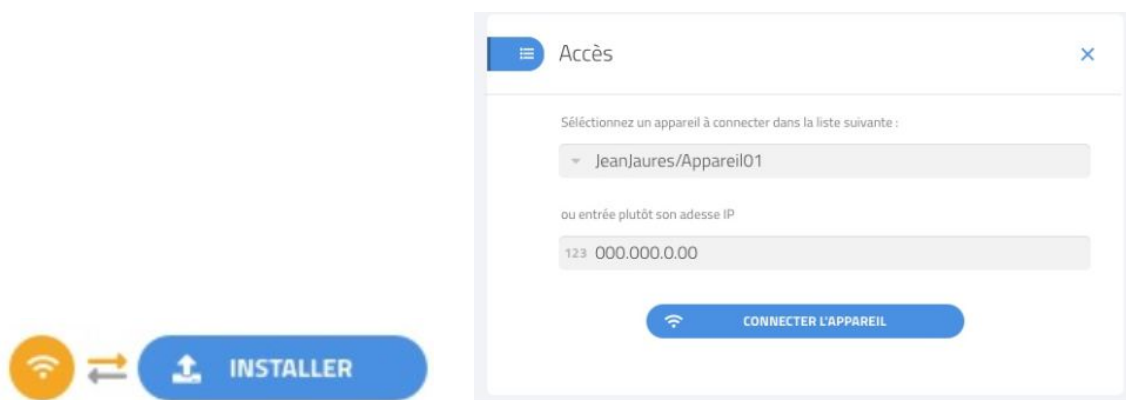
L'environnement de programmation est une application web, qui ne requiert aucune installation spécifique de logiciel ou de plugin. Il suffit d'ouvrir un navigateur et de se rendre sur le site de QuickPi, l'environnement de programmation est alors opérationnel immédiatement.

Le site est compatible avec tout navigateur moderne (Chrome, Firefox, Safari, Edge), et fonctionne sur ordinateur, tablette ou mobile (Android, iPad, Kindle Fire). Pour ceux qui le souhaitent, il est également possible de télécharger l'interface et de la faire fonctionner hors ligne.

L'environnement donne accès à une documentation en ligne de l'ensemble des instructions disponibles, incluant d'une part les constructions de base du langage (boucles, conditions, etc.), et d'autre part les constructions permettant d'interagir avec les composants matériel (bouton, LED, etc.).

Installation et test instantanés sur le Raspberry Pi

Connecter son environnement de programmation à un Raspberry Pi se fait en une étape : choisir le du Raspberry (inscrite sur l'appareil) dans une liste de ceux présents dans la classe, ou bien saisir l'adresse IP du Raspberry Pi dans la zone dédiée de son environnement.



Une fois l'environnement relié au Raspberry Pi, l'utilisateur peut à tout moment lancer l'exécution sur le Raspberry Pi, y compris en mode pas à pas (voir plus loin).

Le programme peut également être installé sur le Raspberry Pi, ce qui lui permet de fonctionner de manière autonome, c'est-à-dire même si l'on éteint l'ordinateur sur lequel on a écrit le programme. Cette opération d'installation se réalise en un simple clic.

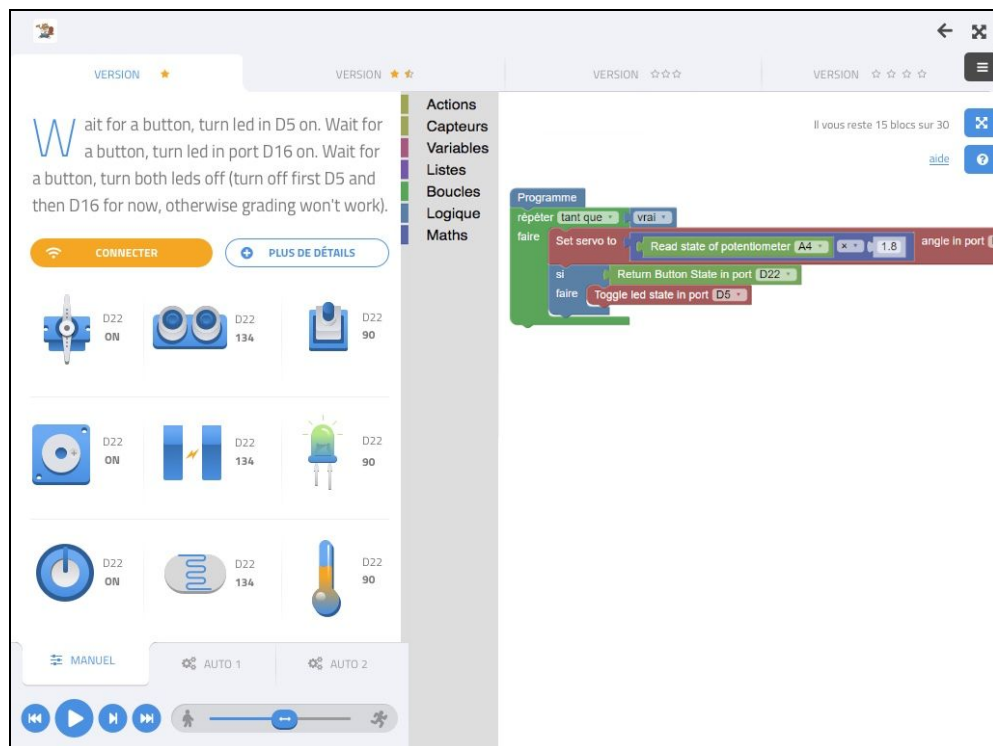
Langages Scratch, Blockly et Python

La même interface supporte plusieurs langages de programmation : les langages visuels Scratch et Blockly pour les plus jeunes ou débutants, et le langage textuel Python pour les plus âgés ou les plus expérimentés.

Toutes les fonctionnalités sont disponibles dans chacun des 3 langages. Les programmes écrits en Scratch ou en Blockly sont convertis automatiquement en code Python avant d'être transmis au Raspberry Pi.

Capture d'écran de l'environnement avec le langage Blockly montrée ci-dessous :

Note : dans la version finale, le sujet sera bien sûr entièrement en français.



Développement et tests sans Raspberry Pi ni composants

Les programmes peuvent non seulement être écrits mais également testés dans le navigateur, même en l'absence de Raspberry Pi, grâce au simulateur QuickPi.

Un simulateur ne suffit pas à tester tous les programmes "en condition réelles", et ne permet pas d'avoir la satisfaction d'interagir avec du matériel physique. Néanmoins, la mise à disposition d'un simulateur est très avantageuse, à plusieurs égards :

- Les élèves peuvent travailler à la maison sans avoir à disposer du matériel.
- Les élèves peuvent travailler en classe même s'il n'y a pas un kit matériel complet disponible pour chaque élève.
- Le simulateur permet une exécution pas à pas, qui permet de déboguer beaucoup plus facilement un programme.
- Le simulateur permet de régler la vitesse d'exécution, ce qui permet de gagner du temps pour effectuer certains tests.

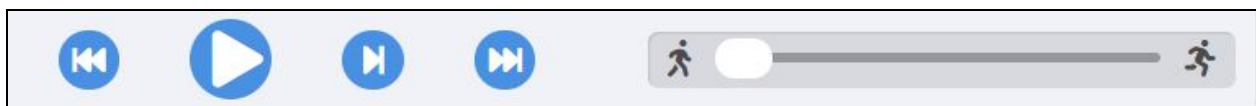
Une fois que le programme semble bien fonctionner sur le simulateur, les élèves peuvent alors exécuter leur programme "pour de vrai" sur un Raspberry Pi équipé des composants requis.

Compréhension facilitée grâce au pas à pas

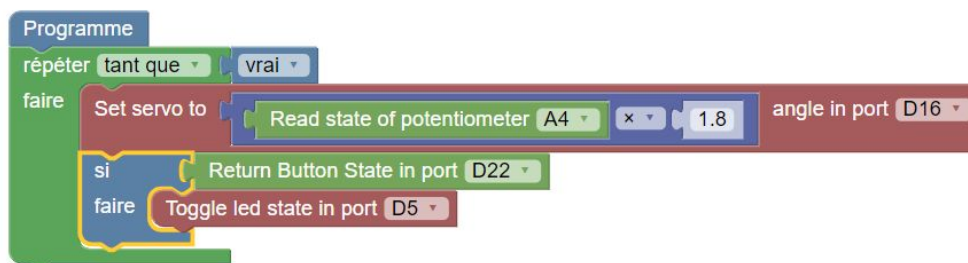
L'interface QuickPi permet d'exécuter un programme à la vitesse de son choix, mais aussi de l'exécuter pas à pas, c'est à dire instruction par instruction. À chaque étape, l'instruction en cours d'exécution est mise en avant dans l'éditeur, et les valeurs intermédiaires sont affichées.

La possibilité d'exécuter son programme en mode pas à pas est une fonctionnalité essentielle pour aider les jeunes dans leur apprentissage de la programmation.

La zone de contrôle de l'exécution est présentée ci-dessous : bouton de retour en arrière, lecture, exécution d'une seule instruction, exécution immédiate jusqu'au bout, et réglage de la vitesse d'exécution :



L'instruction courante est mise en évidence sur le navigateur. En Blockly ou Scratch, elle est surlignée en jaune comme illustré ci-dessous :



En Python, l'instruction en cours d'exécution est surlignée en bleu :

```
1 from quickpi import *
2
3 while 1:
4     waitForButton()
5     turnLedOn()
6     sleep(1)
7     turnLedOff()
8
```

Lors d'une exécution, l'état de tous les capteurs est visible sur l'interface, et le programme réagit immédiatement aux actions effectuées sur le matériel (bouton pressé, potentiomètre tourné, température modifiée, etc.) de la même manière qu'il le ferait si le programme était exécuté de manière autonome sur le Raspberry Pi.

Ce mode pas à pas fonctionne soit entièrement dans le navigateur, soit en visualisant l'exécution du programme dans le navigateur, mais en interagissant directement avec les capteurs et actionneurs du Raspberry Pi, dont l'état est visible dans le navigateur.

QuickPi : intégration dans un parcours pédagogique

Intégration dans une plateforme en ligne

QuickPi pourra être intégré dans un parcours pédagogique complet sur une plateforme d'enseignement, en commençant par des séquences d'exercices validés automatiquement qui font découvrir petit à petit les notions de programmation et les différents capteurs et actionneurs, avant de faire appel à sa créativité et concevoir ses propres projets.

L'environnement et les exercices sont conçus pour pouvoir être intégrés au sein de plateformes web d'apprentissage. Ainsi on peut les insérer dans un MOOC sur les plateformes Fun ou EdX, on peut les intégrer à un parcours sur Moodle ou sur la plateforme d'enseignement France-ioi.

Selon les fonctionnalités de la plateforme utilisée, les enseignants pourront suivre l'activité de chaque élève, les conseiller à distance, organiser des examens en temps limité, gérer des projets par groupes, voire créer leurs propres parcours d'exercices.

Support d'exercices validés automatiquement

Pour aider les élèves à découvrir les notions de programmation, à prendre en main le matériel, et à comprendre rapidement ce qui est possible, des exercices validés automatiquement seront disponibles dans cet environnement. Ces exercices permettent aux élèves de progresser très rapidement dans leur maîtrise de l'outil et du matériel.

Le fonctionnement est similaire à celui utilisé pour les exercices proposés lors du concours de programmation Algoréa (230 000 participants en 2019). La seule différence étant que les exercices font intervenir les capteurs et les actuators, et sont évalués à l'aide du simulateur.

En Scratch ou en Blockly, l'enseignant pourra pour chaque exercice limiter les blocs disponibles à ceux qui sont nécessaires à l'exercice, ce qui permet à l'élève de découvrir les notions petit à petit sans se retrouver submergé.

Les élèves pourront donc progresser parmi une séquence de petits exercices lors d'une première séance, avant de se lancer, beaucoup plus à l'aise, dans la réalisation de leur propre projet.

De plus, les enseignants pourront à terme créer leurs propres exercices. Un éditeur interactif d'exercices est prévu dans une prochaine version.

Une vie simplifiée pour les enseignants

Avec QuickPi, l'enseignant peut en très peu de temps, travailler sur des objets connectés avec ses élèves, avec un outil de développement spécialement conçu et adapté à un apprentissage rapide.

- **Au niveau du matériel** : un temps précieux est gagné grâce aux branchements simplifiés des câbles, et à la configuration rapide des connections sans fil.
- **Au niveau du logiciel** : un temps énorme est économisé grâce à l'exécution en pas à pas qui fonctionne quel que soit le langage choisi, et qui permet de détecter beaucoup plus rapidement les erreurs dans le code.
- **Au niveau des élèves** : le système d'évaluation automatique des exercices permet aux enseignants de laisser les élèves vérifier par eux-mêmes la validité de leur solution, et libère ainsi du temps pour que l'enseignant puisse se concentrer sur les élèves qui font face à des difficultés, au moment où ils y font face.

En savoir plus sur le projet

Diffusion de QuickPi

QuickPi est le nom provisoire du projet, il peut être amené à changer.

Une première version sera disponible en mars 2019.

Informations tarifaire

Le logiciel est gratuit (et open source).

L'intégration à une plateforme d'apprentissage est gratuite (en tous cas sur France-ioi).

Seul le matériel est à acheter.

Le détail du prix TTC des composants commandés en France, par exemple sur <https://www.gotronic.fr> (sauf chargeur et micro-SD) à l'unité est le suivant :

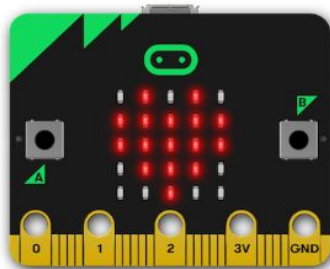
- Raspberry Pi zero WH : 20€
- Carte Grove : 11€.
- Chargeur 5V, 2A : 6€.
- Carte micro-SD : 4€
- Prix de chaque composant (cable de connexion inclus) :
 - LED : 2€
 - Buzzer : 2€
 - Bouton : 2€
 - Potentiometre : 3€
 - Interrupteur magnétique (détection d'ouverture de porte) : 3€
 - Capteur de distance à ultrasons : 5€
 - Capteur de lumière : 3€
 - Capteur de température : 3€
 - Écran LCD 2 lignes : 6.5€
 - Cable Grove pour 2 Servo-moteurs : 1.2€
 - Servo-moteurs : 3.80€

En option, un seul pour l'ensemble de la classe :

- Pont wifi à connecter au réseau local, si pas de réseau wifi utilisable : 30€
- Lecteur de carte micro-SD : 12€
- Cable USB vers micro-USB : 3€
- Adaptateur micro-SD vers SD : 3€ si acheté avec la carte micro-SD
- Pack de 10 aimants neodyme ronds : 2€

Comparison avec Micro:bit

Le Micro:bit est une carte basée sur un micro-contrôleur similaire à ceux d'Arduino, et équipée en série d'un certain nombre de capteurs et de LED. Un outil de développement permettant la programmation avec un langage visuel de Microsoft, ou bien le langage Python, est disponible.



Étant basé sur un micro-contrôleur, la carte est extrêmement limitée en termes de mémoire disponible et de puissance de calcul. QuickPi n'a pas ces limitations.

Micro:bit permet une simulation dans le navigateur lorsque l'on utilise le langage visuel de Microsoft, mais sans fonctionnalités d'exécution pas à pas, tandis que le langage microPython ne permet pas une exécution dans le navigateur, donc rend le débogage très difficile. Au contraire, QuickPi permet une exécution pas à pas pour tous les langages, à la fois en mode simulation complète, et dans un mode utilisant les capteurs et actionneurs directement sur le Raspberry Pi.

L'installation du programme sur Micro:bit nécessite plusieurs étapes manuelles, et passe par une traduction du code sur un serveur distant (donc nécessitant la disponibilité de la connexion internet). Au contraire, QuickPi permet une installation directe du programme en un clic depuis l'interface, sans délai.

Le Micro:bit n'est pas connecté à Internet donc n'est pas un véritable objet connecté. Au contraire, QuickPi dispose d'une connexion Wifi et peut acquérir des données, émettre des données, et ainsi interagir avec d'autres objets connectés sans limite de distance.

Le Micro:bit peut être étendu avec quelques composants supplémentaires, mais nécessitent le branchement de résistances etc. avec plusieurs câbles. Il faut un mode d'emploi pour chaque capteur ou actionneur à brancher. Au contraire, les composants Grove utilisés par QuickPi, se branchent directement dans des connecteurs, sans risque de se tromper.

Enfin, contrairement à Micro:bit QuickPi a été conçu dès le départ pour pouvoir s'intégrer facilement dans une plateforme d'apprentissage, ce qui permet aux enseignants de suivre le progrès des élèves.

Un projet réalisé par l'association France-ioi

QuickPi est un projet réalisé par l'association France-ioi, co-organisatrice du concours Castor (675 000 participants en France en 2018) et organisatrice du concours de programmation Algoréa (230 000 participants en 2019).

Les élèves et enseignants apprécieront le fait que cet outil est développé par une association française à but non lucratif, qu'ils sont très nombreux à connaître et apprécier, notamment au travers du concours Castor Informatique et du concours de programmation Algoréa.

En particulier, le fait que l'interface utilisée par QuickPi pour la programmation et pour la gestion des classes puisse être la même que pour le concours Algoréa facilitera la prise en main.

Informations complémentaires

Pour tout renseignement sur ce projet, merci de contacter info@france-ioi.org