



Source : © Zinesh / Adobe Stock

# boîte de conservation intelligente centrée sur l'être humain

**Maheshwaran Sadasivam et Vairamani Kanagavel (Inde)**

Dites adieu aux difficultés liées aux courses grâce à ce système d'inventaire pour la cuisine. Gardez une trace de vos courses sans effort, avec des mises à jour en temps réel des niveaux de stock. Ce système utilise des éléments peu coûteux, comme un ESP32 d'Espressif, un capteur de pesée et un amplificateur HX711 de SparkFun. Grâce au fonctionnement sur batterie et au chargement par USB, vous gardez toujours le contrôle. Éliminez les approximations, évitez les erreurs humaines et prenez des décisions d'achat précises !

L'objectif premier de toute innovation ou de tout développement est d'améliorer la qualité de la vie humaine, grâce à ce que nous appelons des systèmes centrés sur l'homme. La forte croissance des technologies domestiques intelligentes ces dernières années, est largement attribuée à l'application de l'internet des objets (IdO) dans la domotique. Notre projet sert de système de gestion des stocks de produits d'épicerie dans la cuisine pour l'utilisateur.

Notre système offre des données d'inventaire en temps réel sur les produits alimentaires dans la cuisine par le biais d'un navigateur web. Nous utilisons un contenant équipé de capteurs pour collecter des données sur les niveaux des produits. Ces données sont ensuite transférées et stockées sur une plateforme en cloud via Wifi. Les utilisateurs peuvent accéder à ces informations pour prendre des décisions avisées sur les quantités qu'ils doivent acheter ou réapprovisionner. Le système est alimenté par une batterie et est doté d'une capacité de chargement USB de type B.

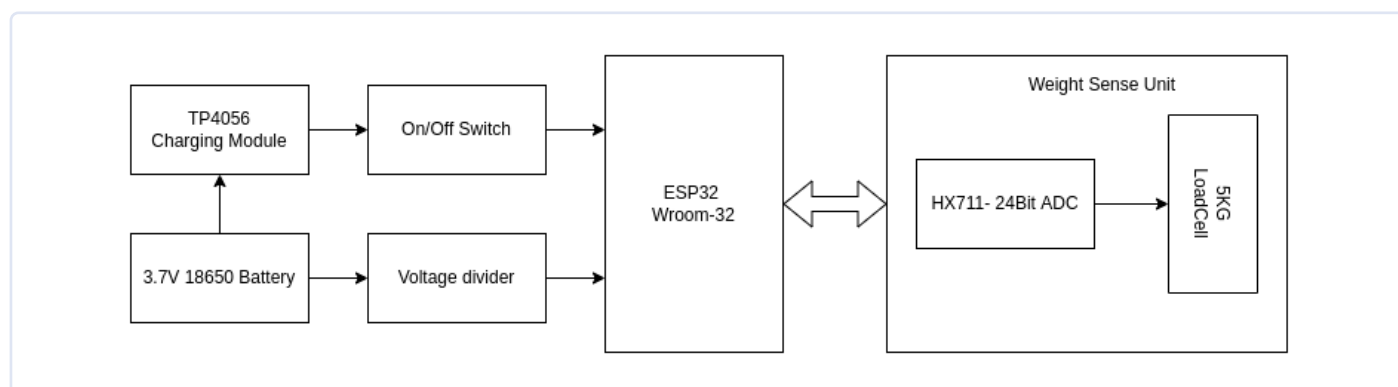


Figure 1. Schéma fonctionnel de la boîte à provisions centrée sur l'humain.



## Structure mécanique

La **figure 4a** montre les circuits à l'intérieur du prototype, tandis que la **figure 4b** montre la connexion du chargeur au prototype.

Le montage mécanique nécessaire pour fixer le capteur de pesage au récipient est le suivant :

1. Le capteur de pesage en aluminium doit comporter 4 trous taraudés et un marquage indiquant la direction de la force. Montez le côté sans le marquage sur la surface fixe et le côté avec le marquage sur la surface mobile. La flèche sur le côté marqué doit pointer vers le bas dans la direction dans laquelle la plate-forme se déplacera lorsqu'une charge est appliquée.
2. La plaque de montage et la plaque mobile doivent être aussi rigides que possible.
3. Veillez à placer des entretoises rigides entre les plaques de montage et le capteur de pesage, comme des entretoises ou des rondelles, qui conviennent parfaitement.

Nous avons utilisé un tube en PVC pour le cadre et une feuille de PVC de forme arrondie comme base pour placer des éléments dessus, en utilisant les écrous, boulons et rondelles nécessaires, car la cellule de charge se plie légèrement lorsqu'un poids est placé dessus. Le modèle CAO de l'installation mécanique et le dessin technique du prototype sont présentés à la **figure 5**.

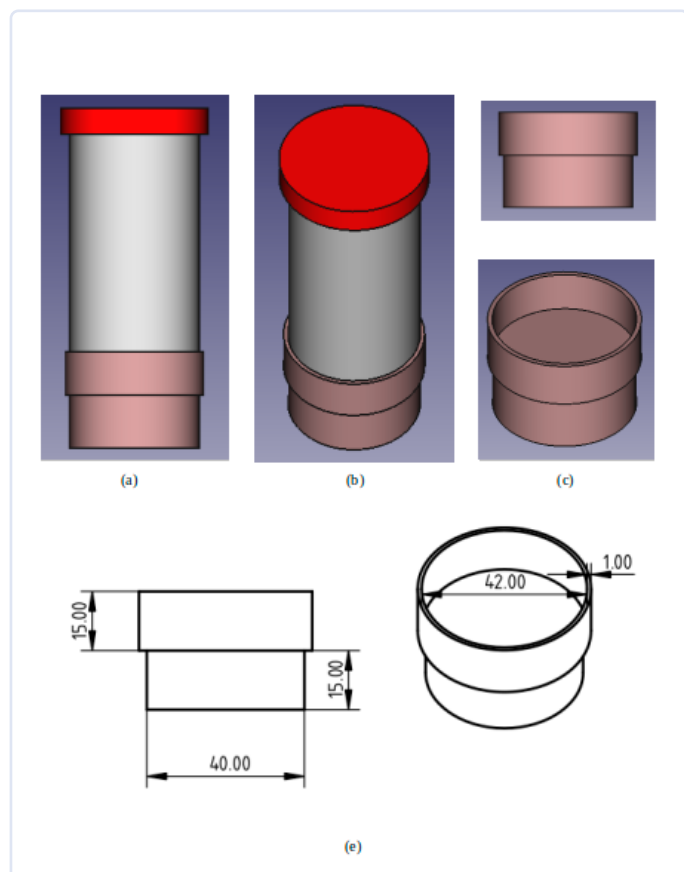


Figure 5. Modèle CAO du système et dessin technique.

## Description du micrologiciel

Le micrologiciel de ce système a été développé dans l'EDI Arduino en C embarqué. Les fichiers des bibliothèques *HX711*, *Adafruit IO*, et *Adafruit sensor* sont ajoutés dans l'EDI Arduino. Une fois le fichier de bibliothèque installé, connectez la carte de développement ESP32 à l'ordinateur avec un câble USB et téléchargez le code de ce système [5]. Remarque : remplacez votre SSID et votre mot de passe dans le code, puis compilez et téléversez le code sur l'ESP32.

Le flux global du micrologiciel est illustré à la **figure 6**. Le micrologiciel lance le Wifi pour obtenir une connexion Internet en fonction des informations d'identification. Ensuite, le CAN du module amplificateur HX711 est accessible via une interface à deux fils (horloge et données) pour mesurer le poids sur le capteur de pesage. La broche GPIO 34 du CAN de l'ESP32 est initialisée pour mesurer la tension de la batterie. Une fois l'initialisation terminée, l'ESP32 lit les données du capteur de pesage et la tension de la batterie. Les données de mesure sont transférées au tableau de bord Adafruit IO en utilisant le protocole MQTT à des intervalles définis, comme le montre la **figure 7**. Une fois le conteneur placé sur la cellule de charge, l'utilisateur doit appuyer sur le bouton « tare » sur le tableau de bord, ce qui réinitialisera la valeur de mesure à un point de référence 0. Le code exécute alors la fonction de tarage.

Des conseils sur la création d'un tableau de bord Adafruit IO peuvent être trouvés à l'adresse suivante [6].

## Conclusion

Le système mis au point ici sera une aubaine pour les gens qui pourront obtenir des informations de n'importe où et à n'importe quel moment sur les niveaux de stock. Il élimine l'erreur humaine et enregistre automatiquement les détails de l'inventaire. L'utilisateur peut planifier le réapprovisionnement et l'achat des produits d'épicerie, sur la base des détails de l'inventaire, à partir de n'importe quel navigateur. Ce système permet d'accéder aux données d'inventaire et d'en rendre compte en temps réel. À l'avenir, ce système pourrait être connecté à un magasin d'alimentation, obtenir les détails des commandes et permettre la livraison de produits alimentaires à la demande. L'utilisation du même système dans l'industrie élimine l'intervention humaine dans la gestion des stocks et fournit un support de surveillance 24h/24 et 7 j/7. ◀

VF : Laurent Raubet — 230092-01



## Liste des composants

### Résistances (résistance traversante ¼ Watt)

R1 = 10 kΩ

R2 = 4 kΩ

### Divers

SW1 = SW\_SPST (interrupteur pour l'alimentation)

BT1 = Élément de batterie (batterie rechargeable 18650 3.7 V lithium-ion)

U1 = TP4056 sur un circuit imprimé

U2 = ESP32 DEVKITC-32D (carte de développement ESP32)

U3 = carte d'interface HX711

U4 = capteur de pesage (5 kg)

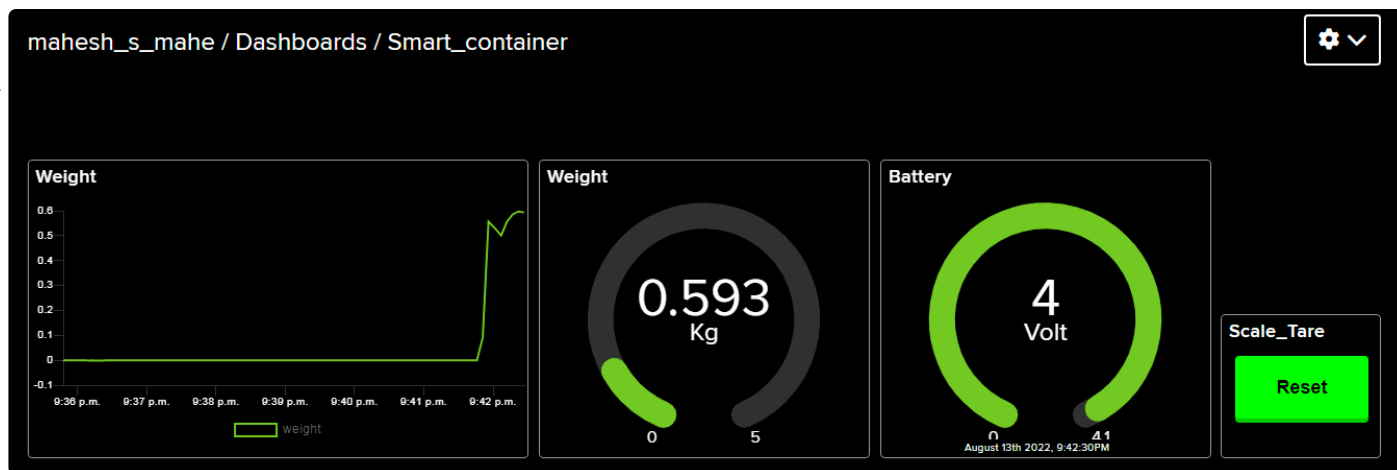


Figure 7. Mise à jour des valeurs de mesure dans le tableau de bord Adafruit IO.

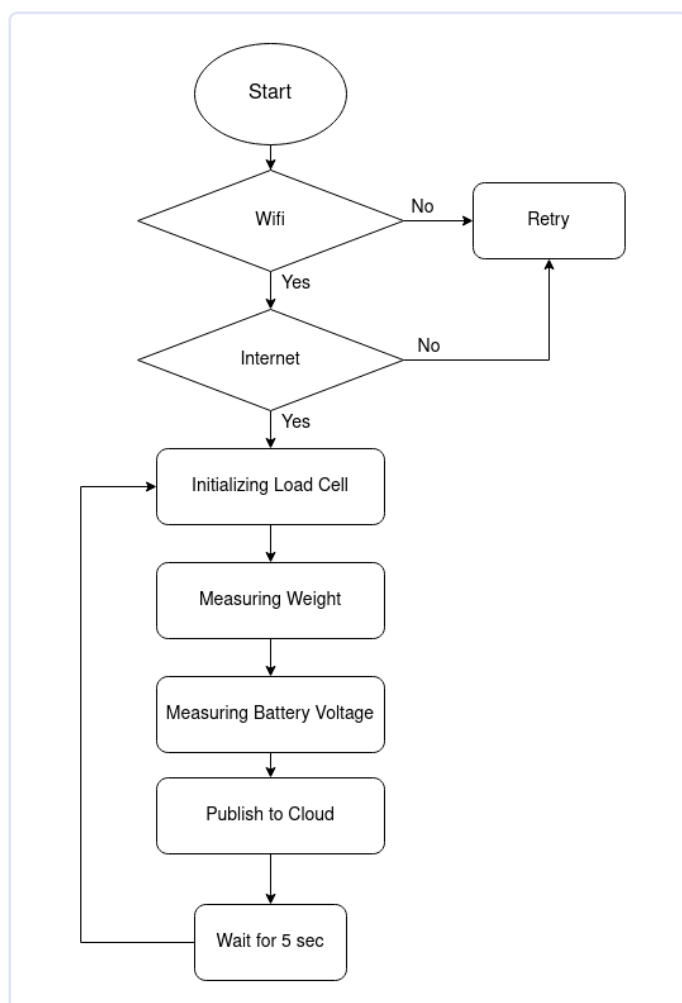


Figure 6. Organigramme du système.

### Questions ou commentaires ?

Contactez Elektor e([redaction@elektor.fr](mailto:redaction@elektor.fr)).



### À propos des auteurs

Maheshwaran Sadasivam est un chercheur en électronique ayant obtenu une maîtrise en électronique au collège Saint Joseph, à Tiruchirappalli, dans l'État du Tamil Nadu, en Inde. Il s'intéresse aux systèmes d'automatisation centrés sur l'homme, aux systèmes embarqués et à la robotique.

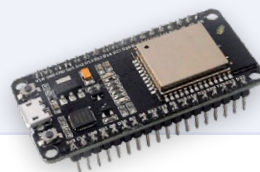


Vairamani Kanagavel est un chercheur en électronique ayant une maîtrise en électronique du collège Saint Joseph, à Tiruchirappalli, dans l'État du Tamil Nadu, en Inde, et titulaire d'un doctorat en instrumentation de l'université Madurai Kamaraj, à Madurai, Tamil Nadu, Inde. Ses domaines d'intérêt sont les réseaux de capteurs sans fil et les systèmes d'automatisation centrés sur l'homme.



### Produit

> **ESP32-DevKitC-32D**  
[www.elektor.fr/18701](http://www.elektor.fr/18701)



## LIENS

- [1] Premiers pas avec les capteurs de pesage : <https://tinyurl.com/loadcellstart>
- [2] Fiche technique du HX711 par SparkFun Electronics : <https://tinyurl.com/sfhx711>
- [3] Fiche technique de la série ESP32 : <https://tinyurl.com/esp32datasheet>
- [4] Chargeur linéaire de batterie TP4056 : <https://tinyurl.com/tp4056liion>
- [5] Téléchargement du logiciel : <https://elektormagazine.fr/230092-04>
- [6] Commencer avec Adafruit IO : <https://learn.adafruit.com/adafruit-io/getting-started>