

DMOSDK

Framework open-source pour objets radio à hyper étalement de spectre (UWB)

Benjamin Freeman

24 octobre 2019



Institut de Recherche
en Informatique de Toulouse

Sommaire

1 Présentation

2 Génèse

3 Open-source

4 Conclusion



Sommaire

- 1 Présentation
 - Intervenant
 - Laboratoire
- 2 Génèse
- 3 Open-source
- 4 Conclusion



Benjamin Freeman

- Ingénieur d'études IRIT depuis un an
- Développement d'outils logiciels et matériels pour les expériences et les démonstrateurs
- Contributeur sur Github : @Beetix



L'équipe et ses travaux

- Au sein de l'équipe Réseaux, Mobiles, Embarqués, Sans fil, Satellites (RMESS) à l'IUT de Blagnac
 - 5 enseignant-chercheurs
 - 1 doctorant
 - 1 postdoc
 - 1 ingénieur
- Autour de la thématique des réseaux d'objets connectés sans fil et IoT
 - Radio à hyper étalement de spectre (UWB)
 - LoRa



Testbed



Testbed LocURa4IoT



Institut de Recherche
en Informatique de Toulouse

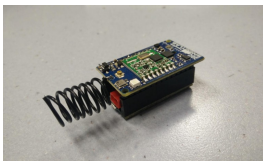
Nœuds



DecaWiNo



DWM1001-DEV



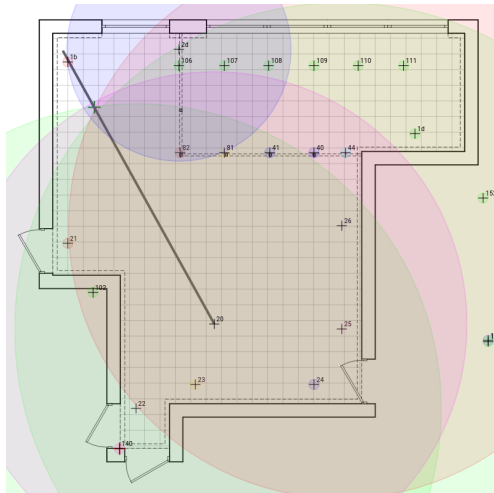
TeensyWiNo



WiNoLoRa



Exemple d'application



Localisation d'un nœud

Sommaire

1 Présentation

2 Génèse

- Bibliothèque Arduino
- Utilisation d'une nouvelle carte
- Développement d'un framework

3 Open-source

4 Conclusion



Écosystème Arduino

- Prototypage rapide
- Couche d'abstraction
- Exécution d'un même code sur des matériels variés
- IDE simpliste



The screenshot shows the Arduino IDE window titled "Blink | Arduino 1.8.5". The code editor contains the following text:

```
Blink 5  
  
This example code is in the public domain.  
  
http://www.arduino.cc/en/Tutorial/Blink  
*/  
  
// the setup function runs once when you press reset or power the board  
void setup() {  
  // initialize digital pin LED_BUILTIN as an output.  
  pinMode(LED_BUILTIN, OUTPUT);  
}  
  
// the loop function runs over and over again forever  
void loop() {  
  digitalWrite(LED_BUILTIN, HIGH); // turn the LED on (HIGH is the voltage level)  
  delay(1000); // wait for a second  
  digitalWrite(LED_BUILTIN, LOW); // turn the LED off by making the voltage LOW  
  delay(1000); // wait for a second  
}
```

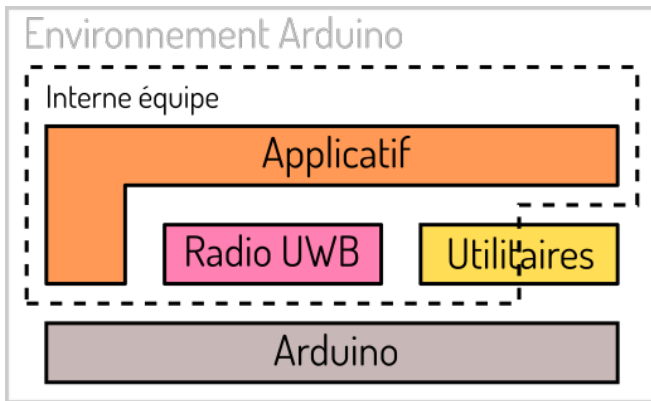
The status bar at the bottom indicates "32" and "Arduino/Genuino Uno on COM1".

IDE Arduino



Institut de Recherche
en Informatique de Toulouse

Architecture logicielle

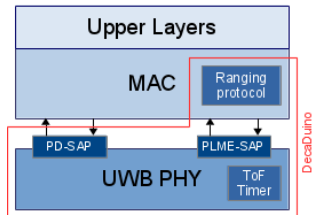


Architecture logicielle Arduino

DecaDuino

<https://github.com/irit-rmess/DecaDuino>

- Créée en 2014
- Ensemble de fonctions pour dialoguer avec la puce radio



Architecture

- Contributions de différents chercheurs



Institut de Recherche
en Informatique de Toulouse

Limitations

- Code difficile à industrialiser
- Travail conséquent pour supporter toutes les fonctionnalités de la puce



Problèmes de compatibilité

Besoin de retravailler le code

- Arduino
- DecaDuino



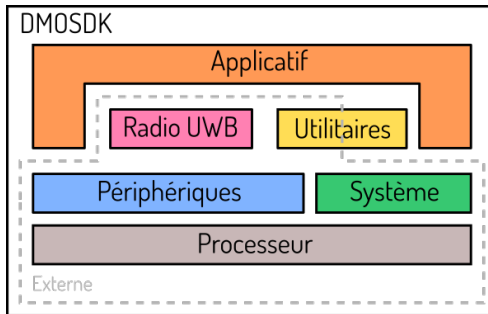
Code fabricant

- Apports
 - Exemple complet d'application
 - Bibliothèque exhaustive pour piloter la radio
- Limitations
 - Code externe pas à jour
 - Nécessité d'outils particuliers (et propriétaires)



Inversion de paradigme

- Développements initiés en novembre 2018 avec une première version stable en juillet 2019
- Identification et séparation des différentes briques logicielles
- Assemblage "minimaliste" des briques
- Découplage le plus possible des outils



Architecture DMOSDK



Sommaire

- 1 Présentation
- 2 Génèse
- 3 Open-source
 - Convictions
 - Application à DMOSDK
 - Plateforme collaborative
- 4 Conclusion



Rendre public des travaux financés par des fonds publics

- Tout le monde peut utiliser le code
- Tout le monde peut contribuer



Le bien commun doit rester commun

- Utilisation de licences ~~contaminantes~~ vaccinales
- Ou alors mise en place d'une licence payante



Moyen de communication

- La recherche publie des articles scientifiques que peu d'industriels lisent
- Possibilité de missions d'expertises



UWB pour tous

- Pure ingénierie
- Socle commun pour différents travaux



Extensions protocolaires réseaux

- Directement en lien avec nos recherches
- Ouvertes ou non
- Exemple PRAPEN (Protocoles de RAnging Précis et Économes en éNergie)

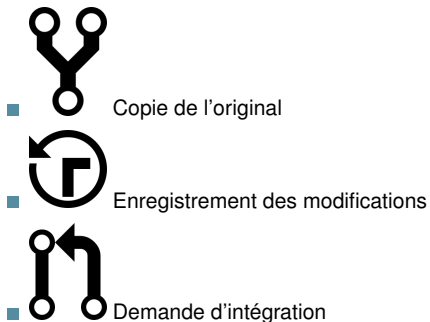


Fonctionnalités

- Intégration d'un gestionnaire de version
- Gestion
 - Utilisateurs
 - Groupes
 - Contributions
 - Bugs/Suggestions
- Hébergement de pages web
- Et bien d'autres...
- **Gratuit pour les projets open-source**



Contribution externe



Projet

<https://github.com/irit-rmess/DMOSDK>

The screenshot shows the GitHub repository page for **DMOSDK** by **irit-rmess**. The repository is described as "DecaWave Module Open Software Development Kit". It has 1 commit, 4 branches, 1 release, 1 environment, 1 contributor, and is licensed under GPL-3.0. The page includes navigation tabs for Code, Issues (1), Pull requests (0), Projects (0), Wiki, Security, Insights, and Settings. A table of recent commits is visible, showing three commits by user **Beetix** for files `boards/DWM1001-DEV`, `build_tools`, and `configs`, all published on July 1, 4 months ago.

Commit	Author	Date
boards/DWM1001-DEV	Beetix	Publish version 1.0, 4 months ago
build_tools	Beetix	Publish version 1.0, 4 months ago
configs	Beetix	Publish version 1.0, 4 months ago

Page projet



Institut de Recherche
en Informatique de Toulouse

Site web

<https://irit-rmess.github.io/DMOSDK/2019/07/02/genesis.html>

DMOSDK - DecaWave Module Open SDK

[About](#)

The genesis

Jul 2, 2019 · Benjamin Freeman

From Arduino to a custom framework

Abstract

This article relates RMESS research team's embedded framework switch. It was initiated by the upgrade of our DecaWave UWB (Ultra Wide Band) radio modules. Work on our Arduino based software was required. As we progressed with the port, more and more hacks were needed, making the usage of Arduino less and less relevant. Therefore we undertook a complete rework resulting in a clean custom solution with full control of the tools and software.

Context

The following work is carried out in the RMESS (Réseaux, Mobiles, Embarqués, Sans fil, Satellites) team

Page projet



Institut de Recherche
en Informatique de Toulouse

Sommaire

1 Présentation

2 Génèse

3 Open-source

4 Conclusion



Institut de Recherche
en Informatique de Toulouse

Conclusion

Rendons nos outils open-source

- Grande facilité à se lancer
- Afin de favoriser les synergies entre le plus d'acteurs



Merci de votre attention



Des questions ?