Réseaux de capteurs OpenSource

Nicolas Aguirre

18 novembre 2016

Smile
OPEN SOURCE SOLUTIONS

Plan

Moi

Ma grande passion les capteurs de températures !
(je peux en parler pendant des heures, et je suis libre le Mercredi soir)

Moi

J’aime bien aussi les petits machins :

- Quand ca tient dans la main
- Que ca s’alimente sur pile bouton pendants des années
- Qu’il y a pas beaucoup de flash et de ram

Moi

J’aime pas les fils.

Moi

Mais je ne suis pas Masochiste.

- Donc j'aime pas (trop) l'assembleur
- Et je préfére le C

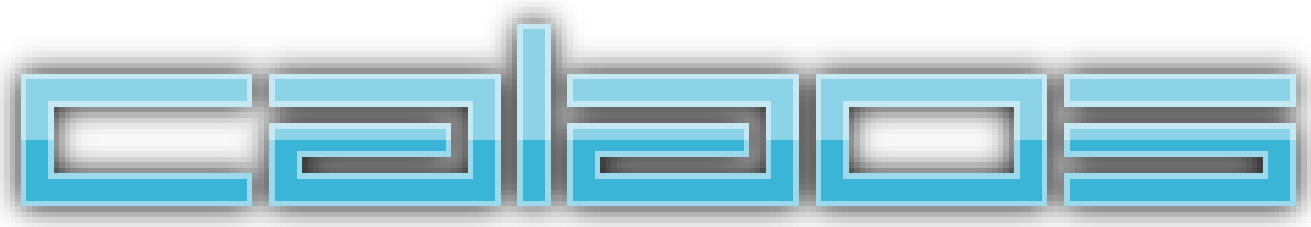
Moi

J’aime bien quand toutes les machines causent entre elles et que même ca s’apelle l’internet.

LE PROGRAMME QUI SUIT CONTIENT
DES PLACEMENTS COMMERCIAUX
DE PRODUITS, MARQUES OU SERVICES

Smile
OPEN SOURCE SOLUTIONS

Moi



J’aime la domotique et surtout le projet Calaos. (GPLv3 !)

Smile
OPEN SOURCE SOLUTIONS

Moi

J’aime beaucoup les normes et des RFC
(mais j’aime pas les lire)

Moi

J’aime l’IoT (celui des hipies pas celui des hipsters)

J’aime les objets communicants que l’on a commencé à développer dans les années 70 et qui, encore aujourd’hui continuent de communiquer ensemble.

Moi

Et ... Je suis Fan de logiciel libre !

Smile
OPEN SOURCE SOLUTIONS

Moi

Pour toutes ces raisons, j'ai commencé à m'intéresser aux capteurs de réseaux sans fil.

Mais quelles techno pour les réseaux sans fil ?

- Zigbee
- Zwave
- 433Mhz / 868Mhz
- NRF24L01
- EnOcean
- Threads
- 6LowPan

Plan

6LowPAN

- 6LowPAN = IPv6 LoW Power wireless Area Networks
- C'est le nom d'un groupe de travail de l'IETF
- basé sur le compression d'entête de paquets IPv6
- Par ce biais permet a ces paquets de transiter sur des réseaux sans fil de type 802.15.4
- 802.15.4 : payload de 128Bytes
- alors qu'IPv6 impose une MTU minimal de 1280 Bytes

RFC 6LowPAN

- la RFC4919 https ://tools.ietf.org/html/rfc4919 : IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs)
- ma RFC4944 https ://tools.ietf.org/html/rfc4944 : Transmission of IPv6 Packets over IEEE 802.15.4 Networks

Un mot sur Ipv6

On est en rupture d'adresses IPv4 (il se dit que ...)

- 2^{32} = 4294967296 adresses possibles
- 2^{128} = Brain overflow

Passons donc à IPv6 c'est trop stylé (dixit les stagiaires) et puis on a pas trop le choix, c'est les fondements de 6LowPAN.

Un mot sur Ipv6

Il faudrait 667 millions de millards d'appareils connectés par millimètres carrés pour épuiser le stock d'adresse IP.

Exercice

- Connaissant le nombre d'appareils connectés par millimètres carrés
- Qu'un appareil consomme en moyenne 100uA sous 3v
- Quelle est la puissance nécessaire pour alimenter les objets connectés sur la superficie de la France ?

Exercice

Quelqu'un ?

16 mots sur l'IPv6

- Les adresses sont l :oo :oo :oo :oo :oo :oo :oo :gues
- Et donc difficiles a retenir et en hexadécimal.
- Mais elles peuvent être affichées compressées ;
- La règle est : on peut supprimer une fois dans une adresse une série de 0 et remplacer ça par " :"
- Donc par exemple, l'adresse suivante :

```
FOOD:0000:0000:0000:0000:0000:DEAD:BEEF
```

- peut s'écrire :

```
FOOD::DEAD:BEEF
```

Abondance d'IPs ne nuit pas

Mais change la philosophie :

- Les interfaces réseaux peuvent avoir plusieurs adresses IP.
  - Une adresse lien-locale obligatoire (limité au lien L2, i.e Ethernet) et commencent par FE80
  - Une ou plusieurs autres adresses données par un DHCPv6 ou calculé a partir d'un Prefix et de l'adresse MAC
- L'implémentation du multicast est obligatoire
- Le broadcast est remplacé par le multicast sur le lien locale
- Le multicast est la base du protocole ND (remplacement d'ARP)
- Il n'y a plus de protocoles à cheval entre L2 et L3 (ARP, DHCP)

802.15.4

Protocole de communication IEEE pour les réseau sans fil Fait pour les LR WPAN : Low Rate Wireless Personnal Area Network)

- Faible consommation
- Faible débit
- Faible portée

Utilisé dans différentes implémentations :

- IP : 6LowPan
- Propriétaire : Zigbee

802.15.4

- Permet les réseau de type Etoile ou Mesh
- Adresses sur 16 ou 64 bits
- Méthode d'accès au media CSMA/CA (identique au wifi)
- Faible Consommation d'énergie
- Indication sur la qualité de liaison LQI
- Différentes Bande de fréquences possible :
  - 16 Canaux de 2.4 a 2.4835 GHz
  - 10 canaux de 902 a 928 MHz (US)
  - 1 Canal dans la bande 868 a 868.6 MHz (EU)

6LowPan

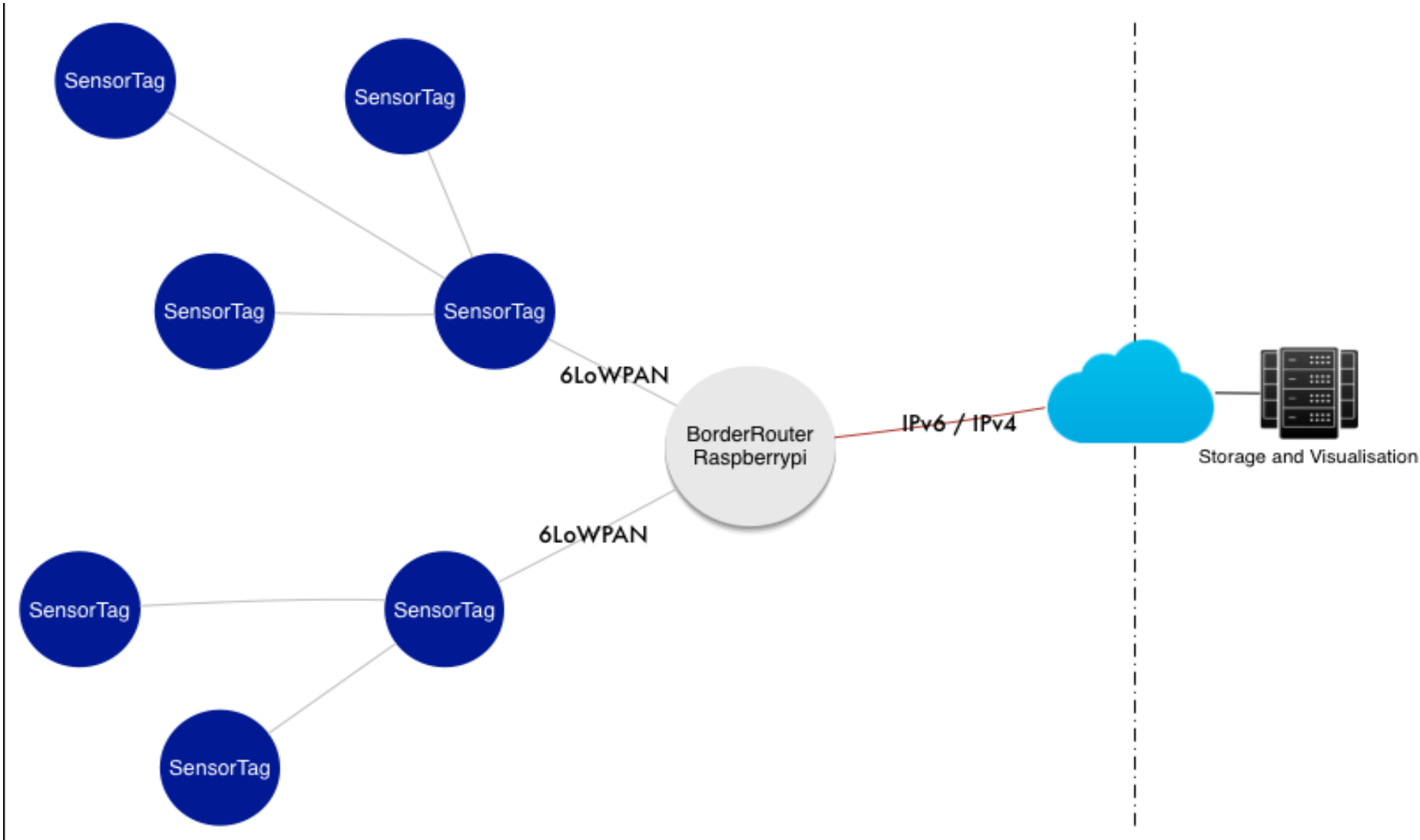
- LowPan est un réseau contraint en ressources (faible CPU, mémoire et batterie)
- Le débit du réseau est limité (jusqu'a 250Kb/s)
- La MTU d'un réseau 802.15.4 est de 128Bytes
- L'entête de la couche MAC est de 25Bytes
- => il ne reste que 102 bytes pour les couches supérieur
- La couche de sécurisation des données AES ajoute 21 Bytes supplémentaires

Mise en place

On a besoin de :

- une machine qui émet des données 6lowpan
- une machine qui reçoit des données IPv6
- une machine qui sert de passerelle entre le 6lowPAN et l'IPv6

Mise en place

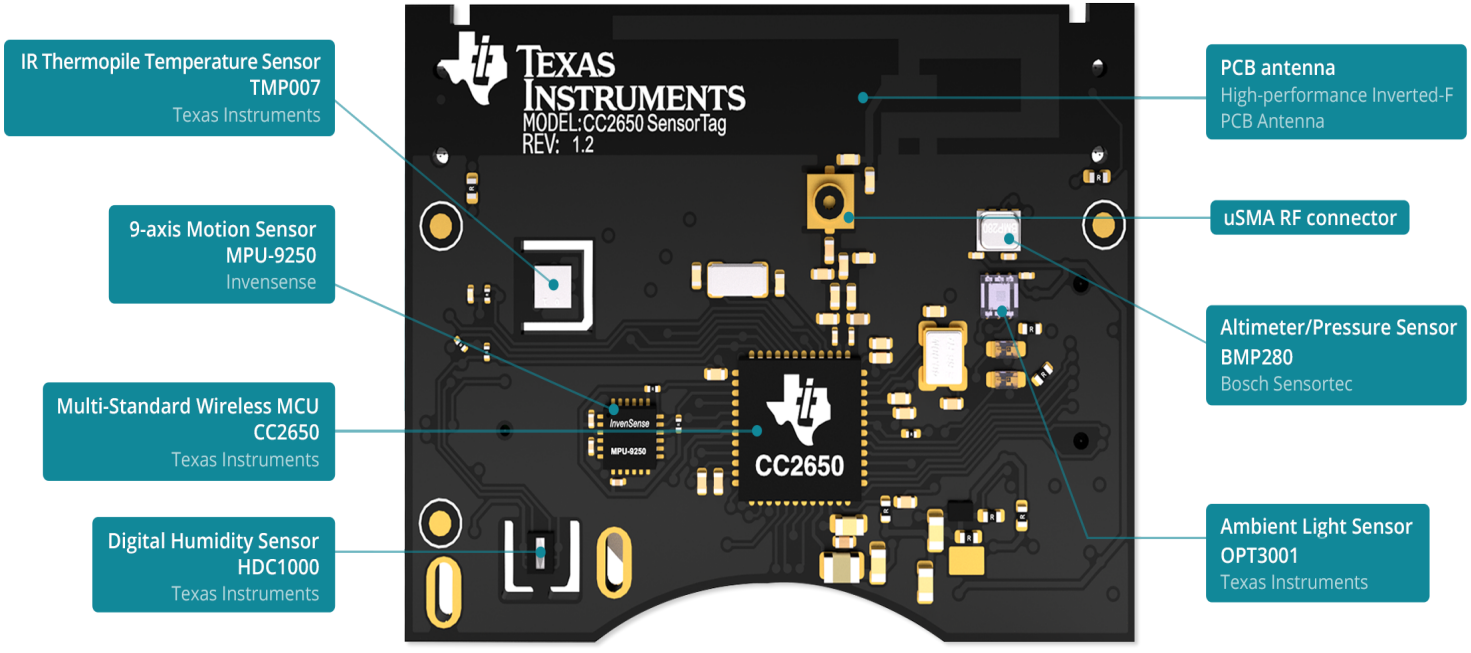


Smile
OPEN SOURCE SOLUTIONS

Machine émettrice

SensorTAG de Texas Instruments

Machine émettrice

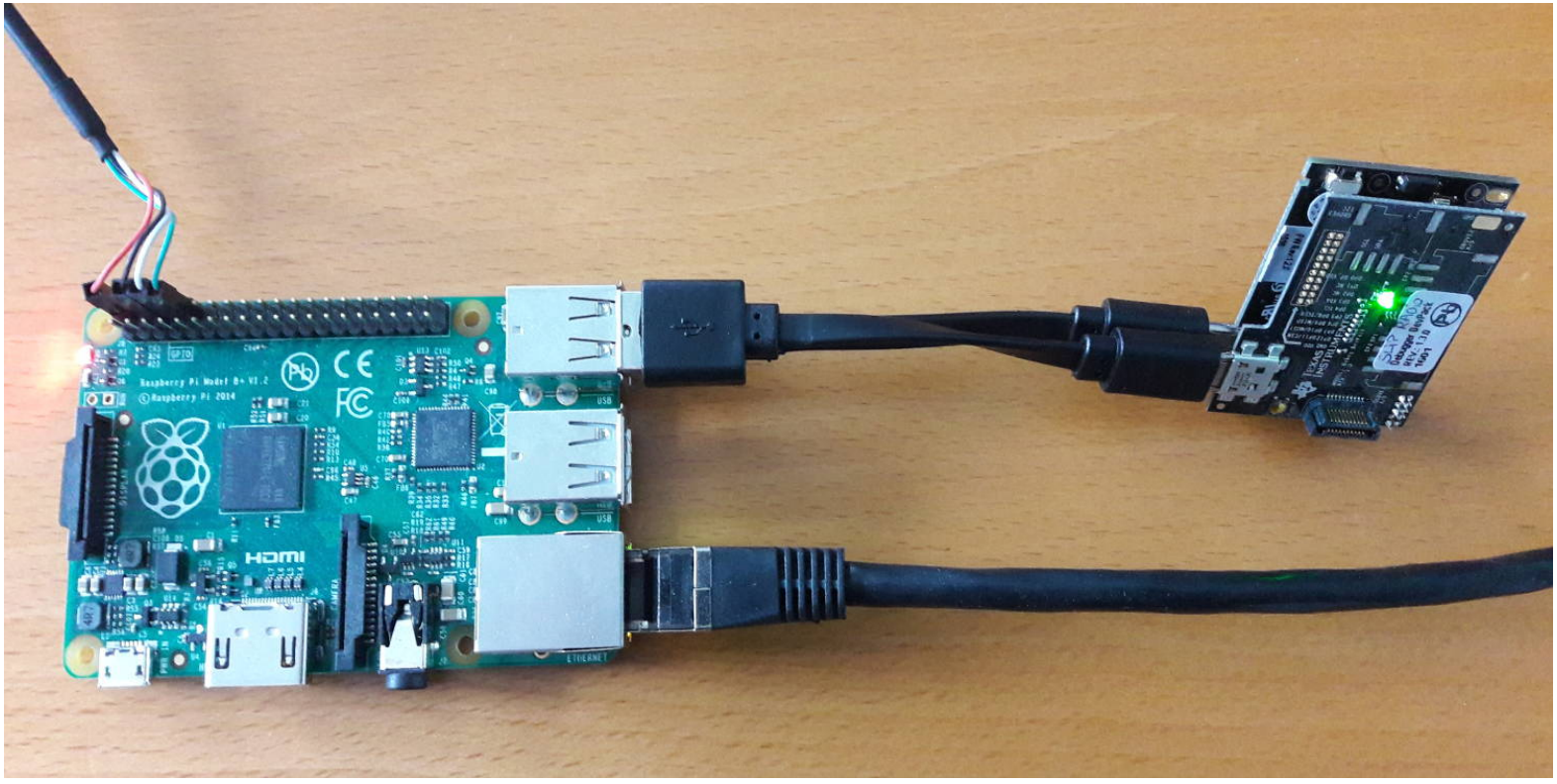


Machine réceptrice

Un PC avec un GNU/Linux ou tout autre solution compatible IPv6
Et même IPv4

Passerelle

Une RaspberryPI et un SensorTAG pour le support du 802.15.4



Coût Total

- Une RaspberryPI : 45 euros
- Un SensorTAG pour le BR : 30 euros
- Un pack debug (pour l'USB série) : 30 euros
- Un SensorTAG : 30 euros
- Un PC : on le compte ?

SensorTag

- Carte de dev "prête" à lemploi développée par Texas Instrument
- Cortex M3 48MHz 128KB de Flash 8KB de RAM
- Flash externe de 512KB (pour mise à jour OTA ou stockage)
- Consommation 10mA en émission, 100uA en veille
- Radio 802.15.4 ET Bluetooth Low Energy (BLE)
- 30 euros en quantités limitées sur le site de TI (max 3 par commandes :'(:'(:'( )
- Fonctionnement sur pile boutons

SensorTag

Les SensorTAG supportent l'OS Contiki.
Génial c'est Open Source (licence BSD)

Plan

Multitâche coopératif

Contiki est un système d'exploitation

- open source
- multi-tâches
- portable

Idéal pour

- les systèmes embarqués
- les réseaux de capteurs sans fil (WSN : Wireless sensor network)

Contiki est conçu spécifiquement pour des Micro-contrôleurs avec de petites quantités de mémoire. Une configuration typique :

- 2 Ko de RAM
- 40 Ko de ROM

Contiki

Contiki est écrit en C et est disponible sous licence BSD.
Contiki est utilisé principalement pour son support des réseau IP :

- IPv4
- IPv6

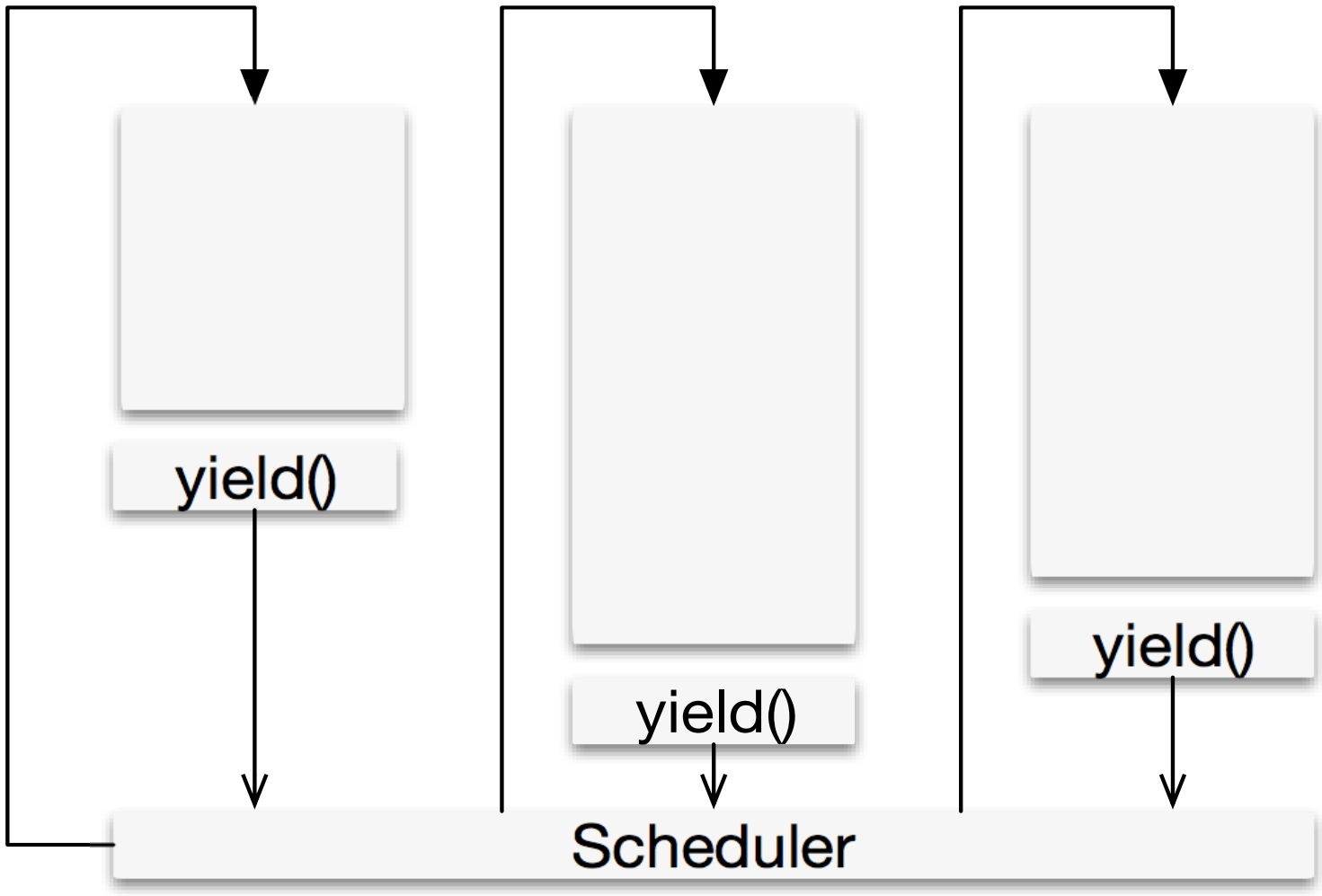
La pile uIPv6 est certifiée IPv6 Ready Phase 1 par CISCO
La pile Ip de contiki uIP a été initialement publié en 2011 et est aujourd'hui utilisée industriellement par différentes entreprises à travers le monde.

Contiki

Contiki est développé par une équipe de développeurs du domaine de l'industrie et du domaine universitaire. Le développeur principal :

- Adam Dunkels
- Swedish Institute of Computer Science.

Multitâche coopératif



Multitâche Coopératif

Proto-Threads

- Spécificité de contiki.
- Développés par Adamn Dunkels.
- Threads extrêmement légers et dépourvus de stack (stackless)
- Conçus spécifiquement pour les systèmes ultra contraints en mémoire.
- Fournissent une exécution de code linéaire pour les "event-driver".
- Sont implémentés en C.
- Peuvent être utilisés avec ou sans un système d'exploitation sous jacent.
- Permettent un contrôle de flux séquentiel sans avoir recours à des machine à états complexes.
- Permettent de se passer de la lourdeur d'un système multi threads complet.

Proto-Threads

exemple de code :

```
#include "pt.h"

static int counter;
static struct pt example_pt;

static
PT_THREAD(example(struct pt *pt))
{
  PT_BEGIN(pt);

  while(1) {
    PT_WAIT_UNTIL(pt, counter == 1000);
    printf("Seuil atteint\n");
    counter = 0;
  }

  PT_END(pt);
}
```

Proto-Threads

Et voici une version allégée de ce a quoi ressemble les MACROS :

```
struct pt
{
    unsigned short lc;
};

#define PT_THREAD(name_args)  char name_args

#define PT_BEGIN(pt)          switch(pt->lc) { case 0:

#define PT_WAIT_UNTIL(pt, c)  pt->lc = __LINE__; \
^^I                       case __LINE__: \
                          if(!(c)) return 0

#define PT_END(pt)            } pt->lc = 0; return 2

#define PT_INIT(pt)           pt->lc = 0
```

Proto-Threads

Et voici une version allégée de ce a quoi ressemble les MACROS :

```
static char example(struct pt *pt)
{
  switch(pt->lc) { case 0:

  while(1) {
    pt->lc = 12; case 12:
    if(!(counter == 1000)) return 0;
    printf("Seuil atteint\n");
    counter = 0;
  }

  } pt->lc = 0; return 2;
}
```

Proto-Threads

Vous aussi vous trouvez ca SALE ?

Proto-Threads

Oui mais :

- OverHead très faible : Seulement 2 octets par protothreads et pas de RAM dédié a la pile.
- Code portable : écrit en C et aucun code écrit en assembleur.
- Peut être utilisé avec ou sans OS
- Fourni des attentes bloquantes sans changement de contexte

Avec Contiki et un SensorTag on a la possibilité de communiquer en 6LowPan avec d'autres.
Mais on veux faire de l'IP !
Il nous faut une passerelle pour passer du 6LowPan à l'IPv6
C'est la que 6LBR entre en jeux !

Plan

6LBR

Projet initié par la société Belge Cetic : https ://www.cetic.be/
6lbr est un fork de Contiki
6LBR est une solution de Routeur de Bordure 6LowPan / RPL Il peut fonctionne en tant qu'application sur une carte embarqué ou sur un hôte Linux. Il est conçu pour être flexible. Il peut être configuré pour s'adapter a différente topologies réseau et interconnecter les réseaux de capteur sans fil au monde IP

6LBR

Le but de 6Lbr est d'interconnecter un réseau WSN basé sur le 802.15.4 et 6LowPan avec un réseau IPv6 basé sur l'Ethernet ou le Wifi.
Cette connexion peut être réalisé a différents niveaux du modèle OSI.

- Au niveau 2, link layer => on parle de pont réseau (bridge) ou de switch ;
- Au niveau 3, network layer => on parle de routeur (router)
- Au niveau 7, application layer => on parle de passerelle (gateway)

6LBR est une plateforme prête a l'emploi pour interconnecter des réseaux 6LowPAN et des réseaux IP. Il part du principe qu'il existe une interface du côté IP et une autre de type 802.15.4 du côté 6LowPAN.

6LBR

Pour intégrer 6LBR sur notre RPI :

- Soit on prend Raspbian, et on compile en prenant les sources sur https ://github.com/cetic/6lbr/
- Soit on prend OpenEmbedded et on peux ajouter le layer meta-6lbr ici :
  https ://github.com/Openwide-Ingenierie/meta-6lbr

Une fois que l'on a installé 6LBR sur la machine

Plan

Smile
OPEN SOURCE SOLUTIONS

IFTTT

IFTTT = If This Then That

- Webservice permettant de connecter différents services web entre eux via des actions primaires.
- C'est assez limité, mais assez intéressant pour la démo.
- Permet de faire des requêtes HTTP via un "Channel Maker" Pas possible de faire du IFTATTT (If This and This Then That)

IP64

- Un problème pas d'IPv6 pour IFTTT !
- Il faut donc passer par de l'IPv4.
- 6LBR gère le IP64
- On encapsule les adresses IPV4 dans des adresses IPV6.
- Dans l'autre sens c'est du NAT 4 to 6.

Show me the code

https ://github.com/Openwide-Ingenierie/contiki-boite-a-meuh