

Textul si imaginile din acest document sunt licentiate

Attribution-NonCommercial-NoDerivs  
CC BY-NC-ND



Codul sursa din acest document este licentiat

Public-Domain

Esti liber sa distribui acest document prin orice mijloace consideri (email, publicare pe website / blog, printare, sau orice alt mijloc), atat timp cat nu aduci nici un fel de modificari acestuia. Codul sursa din acest document poate fi utilizat in orice fel de scop, de natura comerciala sau nu, fara nici un fel de limitari.

# Physical Web

Ce se înțelege prin Physical Web? Este o încercare de a crea o legătură între lumea reală (lumea fizică) și spațiul web prin intermediul unor jaloane (beacons) radio care transmit informații web relevante pentru locația fizică în care ne aflăm. Conceptul este prezent în mai multe tehnologii recente precum tehnologia iBeacon lansată de Apple în 2013 sau tehnologia URI Beacon lansată de Google în 2014. Chiar dacă aflat într-o etapă timpurie de standardizare conceptul de Physical Web pune bazele unei direcții inedite de interacțiune fizică cu mediul on-line. În cadrul lecției de față vom construi un jalon radio bazat pe tehnologia deschisă Eddystone, lansată de Google în 2015 – tehnologie succesoare tehnologiei URI Beacon. Mai multe informații despre Physical Web puteți găsi la adresele:



The physical web: A hidden world that will change things forever

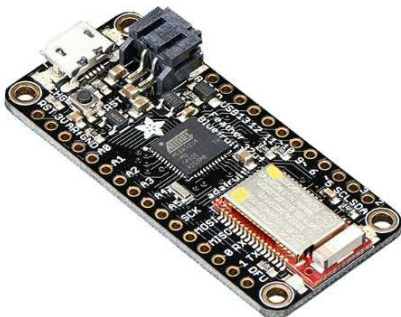
<http://www.techradar.com/news/world-of-tech/future-tech/the-physical-web-a-hidden-world-that-will-change-things-forever-1302664>

The Physical Web

<https://google.github.io/physical-web/>

Eddystone Protocol Specification

<https://github.com/google/eddystone/blob/master/protocol-specification.md>



Tehnologiile actuale de jalonare radio folosesc protocolul Bluetooth – mai exact Bluetooth 4.0 (BLE sau Bluetooth Smart). Pentru a crea un jalon este necesară utilizarea unui modul radio BLE sau a unei plăci de dezvoltare ce integrează un astfel de modul. Din acest motiv pentru implementare vom utiliza placa de dezvoltare Adafruit Feather 32U4 BLE. Această placă de dezvoltare combină microcontrolerul Atmel ATmega32U4 (prezent și pe Arduino Leonardo) cu modulul BLE Nordic nRF51822.

<https://www.robofun.ro/adafruit-feather-32u4-bluefruit-le>

<https://www.robofun.ro/forum/>

Pentru mai multe informații despre capacitățile și modul de utilizare a plăcii de dezvoltare Adafruit Feather 32U4 BLE se poate consulta materialul:

Adafruit Feather 32u4 Bluefruit LE

<https://learn.adafruit.com/adafruit-feather-32u4-bluefruit-le>

Placa de dezvoltare aleasă nu este singura variantă de implementare dar este o soluție simplă și compactă, compatibilă 100% cu mediul de dezvoltare Arduino IDE și, datorită posibilității de alimentare de la un acumulator LiPo, portabilă. Se pot dezvolta soluții similare utilizând și alte plăci de dezvoltare precum:

RedBearLab BLE Nano - nRF51822

<https://www.robofun.ro/wireless/wireless-bluetooth/redbearlab-ble-nano-nrf51822>

Arduino Uno + Bluefruit LE Shield

<https://www.robofun.ro/wireless/wireless-bluetooth/Bluefruit-LE-Shield-pentru-Arduino>

Simblee BLE Breakout - RFD77101

<https://www.robofun.ro/wireless/wireless-bluetooth/sparkfun-simblee-ble-breakout-rfd77101>

Exemple de proiecte similare realizate cu alte plăci de dezvoltare:

Eddy and his stones: a DIY beacon and some mobile apps

<https://www.simplicity.be/article/eddy-and-his-stones-diy-arduino-beacon-mobile-apps/>

DIY Arduino Beacons as an alternative to iBeacon

<https://evotothings.com/diy-arduino-beacons/>

Cheap Bluetooth 4.0 LE beacons, with your Arduino and a NRF24L01+ radio module

<http://makernews.info/featured/2015/09/BLE-nrf24l01-arduino.html>

Totuși. un aspect foarte important al plăcii de dezvoltare Feather 32U4 BLE este posibilitatea implementării unei varietăți foarte mare de proiecte diverse:

Tastură fără fir prin Bluetooth

<https://learn.adafruit.com/adafruit-feather-32u4-bluefruit-le/hidkeyboard>

Monitor cardiac fără fir

<https://learn.adafruit.com/adafruit-feather-32u4-bluefruit-le/heartratemonitor>

Sistem de preluare informații de la dispozitive inteligente mobile  
<https://learn.adafruit.com/adafruit-feather-32u4-bluefruit-le/controller>

Pentru a crea jalonul Eddystone cu ajutorul plăcii de dezvoltare Feather 32U4 BLE este suficient să încărcăm următorul program, în prealabil este necesar să instalăm sub mediul Arduino IDE biblioteca Adafruit BluefruitLE nRF51:

[https://github.com/adafruit/Adafruit\\_BluefruitLE\\_nRF51](https://github.com/adafruit/Adafruit_BluefruitLE_nRF51)

```
#include <SPI.h>
#include "Adafruit_BLE.h"
#include "Adafruit_BluefruitLE_SPI.h"

#define BLUEFRUIT_SPI_CS          8
#define BLUEFRUIT_SPI_IRQ        7
#define BLUEFRUIT_SPI_RST        4

#define BUFSIZE                    160
#define VERBOSE_MODE               true

Adafruit_BluefruitLE_SPI ble(BLUEFRUIT_SPI_CS,
                             BLUEFRUIT_SPI_IRQ, BLUEFRUIT_SPI_RST);

void error(const __FlashStringHelper*err) {
  Serial.println(err);
  while (1);
}
```

Esența programului se află în cadrul secțiunii *setup()* unde vom realiza inițializările necesare funcționării jalonului radio:

```
void setup() {
  while (!Serial);
  delay(500);
  Serial.begin(115200);
  Serial.println(F("Physical Web Beacon"));
  Serial.println(F("-----"));
  Serial.print(F("Initialising the Bluefruit LE module: "));
  if ( !ble.begin(VERBOSE_MODE) )
  {
    error(F("Couldn't find Bluefruit, make sure it's in CoMmanD mode &
                                                    check wiring?"));
  }
  Serial.println( F("OK!") );
  Serial.println(F("Performing a factory reset: "));
  if ( ! ble.factoryReset() ){
    error(F("Couldn't factory reset"));
  }
  Serial.println("Requesting Bluefruit info:");
  ble.info();
}
```

După inițializarea modului BLE vom transmite comenzile de configurare a jalonului radio (denumire, înscriere serviciu în tabela de funcționare, activare emitere informații, URL de transmis):

```
ble.println("AT+GAPDEVNAME=PhysWebBeac");
ble.waitForOK();
ble.println("AT+EDDYSTONESERVICEEN=1");
ble.waitForOK();
ble.println("AT+EDDYSTONEBROADCAST=1");
ble.waitForOK();
ble.println("ATZ");
ble.waitForOK();
delay(1000);
ble.println("AT+EDDYSTONEURL=http://robofun.ro/");
ble.waitForOK();
ble.println("ATZ");
ble.waitForOK();
delay(1000);
}
```

În cadrul secțiunii *loop()* nu se execută nimic:



```
Physical Web Beacon
-----
Initialising the Bluefruit LE module: OK!
Performing a factory reset:
AT+FACTORYRESET

<- OK
Requesting Bluefruit info:
-----
BLESPIFRIEND
nRF51822 QFACA00
258645F29D9803E5
0.7.0
0.7.0
Jun 28 2016
S110 8.0.0, 0.2
-----
AT+GAPDEVNAME=PhysWebBeac

<- OK
AT+EDDYSTONESERVICEEN=1

<- OK
AT+EDDYSTONEBROADCAST=1

<- OK
ATZ

<- OK
AT+EDDYSTONEURL=http://robofun.ro/

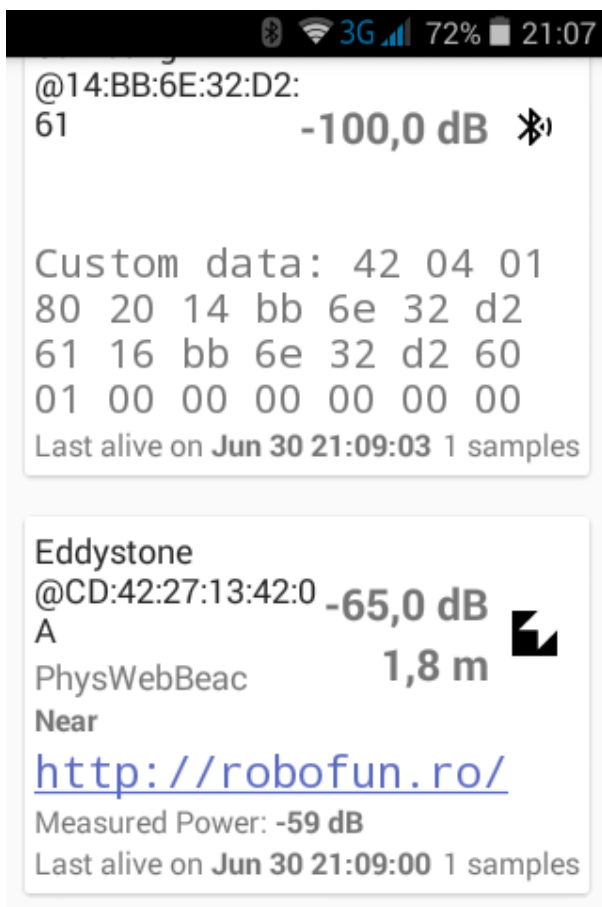
<- OK
ATZ

<- OK
```

```
void loop() {
    delay(1000);
}
```

Programul a fost realizat și testat cu Arduino IDE 1.6.9, extensia Adafruit AVR Boards 1.4.7 și biblioteca Adafruit BluefruitLE nRF51 1.9.0. Versiune firmware-ului modului nRF51822: 0.7.0.

Rularea programului va avea următoarea manifestare în consola serială Arduino IDE – imagine alăturată.



Pentru a verifica funcționarea jalonului este necesar un terminal mobil inteligent (mobil, tabletă) ce rulează sistemul de operare Android și posedă conectivitate Bluetooth 4.0. Există mai multe aplicații ce pot fi descărcate din Google Play și pot identifica și interacționa cu jalonul radio creat de noi, două exemple:

#### Beacon Toy

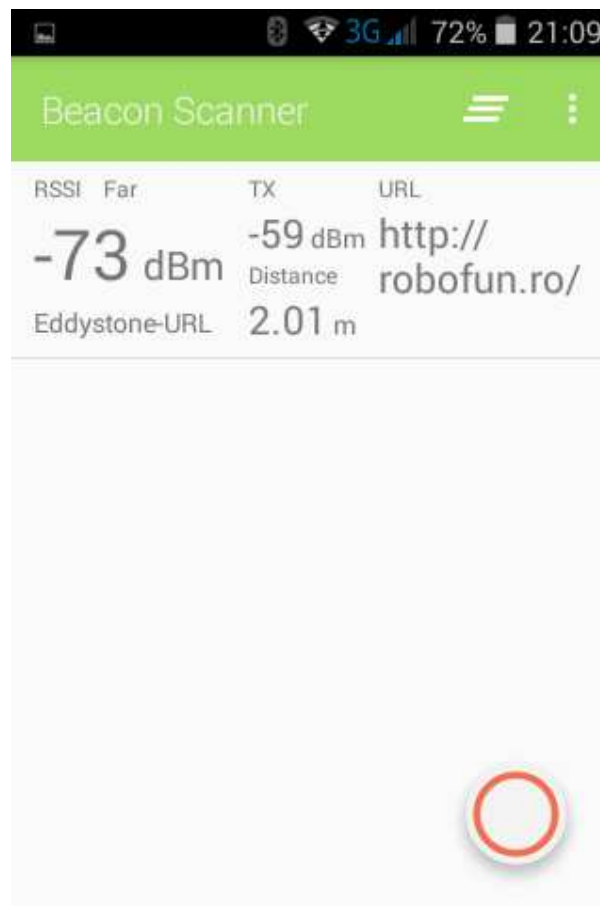
<https://play.google.com/store/apps/details?id=com.uriio>

(imagine stânga jalon PhysWebBeac)

#### iBeacon & Eddystone Scanner

<https://play.google.com/store/apps/details?id=de.flurp.beaconscanner.app>

(imagine dreapta jalon PhysWebBeac)



<https://www.robofun.ro/forum/>