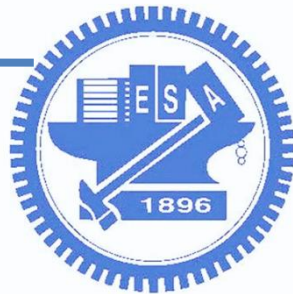


# 嵌入式系統設計概論與實作

曾煜棋、吳昆儒

**National Yang Ming Chiao Tung University**



# Last week

- 嵌入式應用: 網路攝影機
  - 影像辨識 (opencv)
    - 圖片旋轉, 裁切, 縮放
    - 人臉識別
    - 人臉輪廓識別

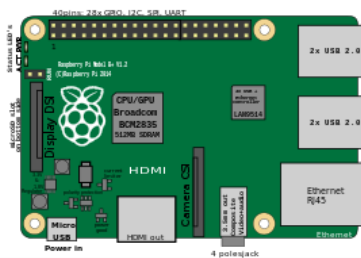


# This week

- 嵌入式應用: BLE beacon
  - Beacon applications
  - Eddystone, iBeacon protocol



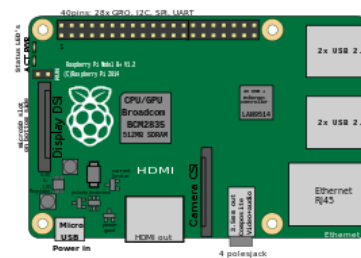
Receiver



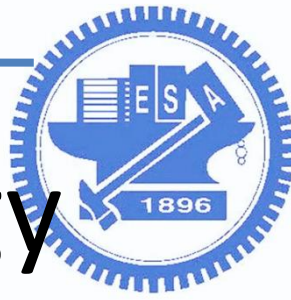
Beacon



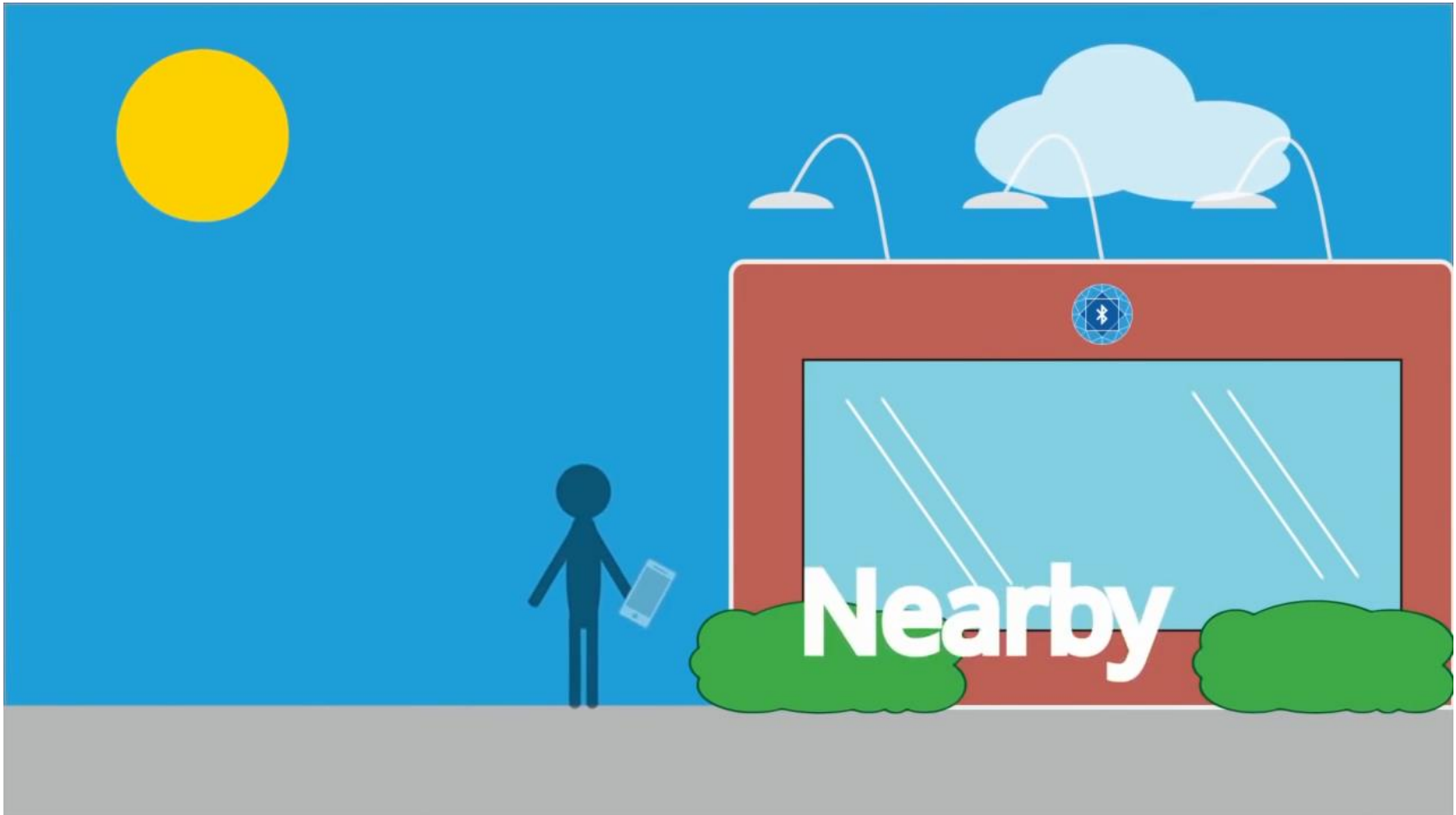
Beacon

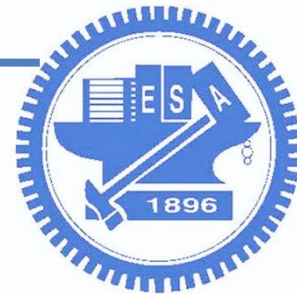


Receiver



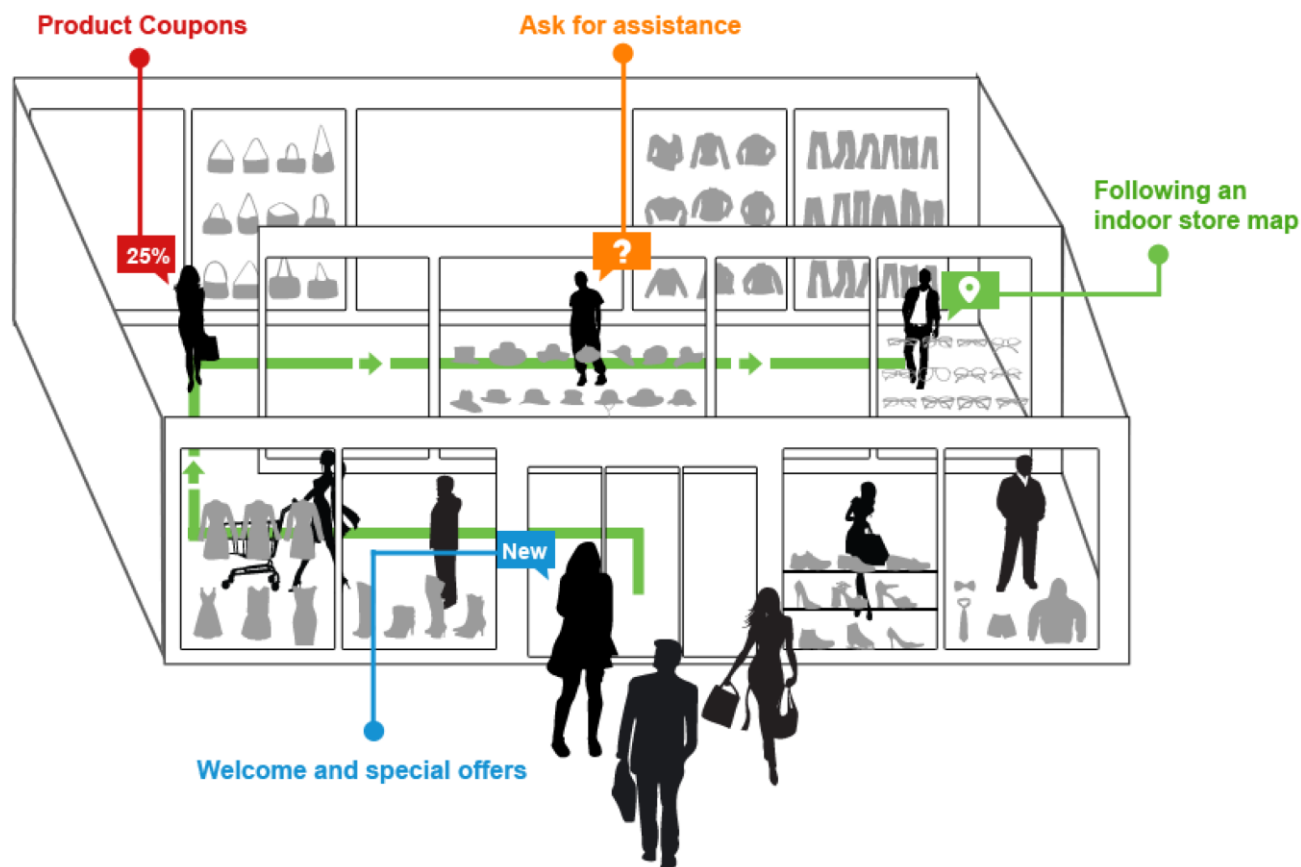
# What Is Beacon Technology

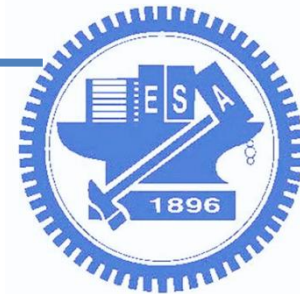




# BLE 的應用

- 微型定位服務
- 推播訊息





# AirTag



AirTag

## 它是如何運作的？

你的 AirTag 會發出安全的藍牙訊號，而同在「尋找」網絡中並且就在附近的裝置，都能偵測得到。這些裝置可將你 AirTag 的位置傳送到 iCloud，然後你就能打開「尋找」app 並在地圖上看到它。整個過程完全匿名並經過加密處理，保護你的隱私；而且它以高效運作，無須擔心電池續航力或數據用量的問題。



# Research

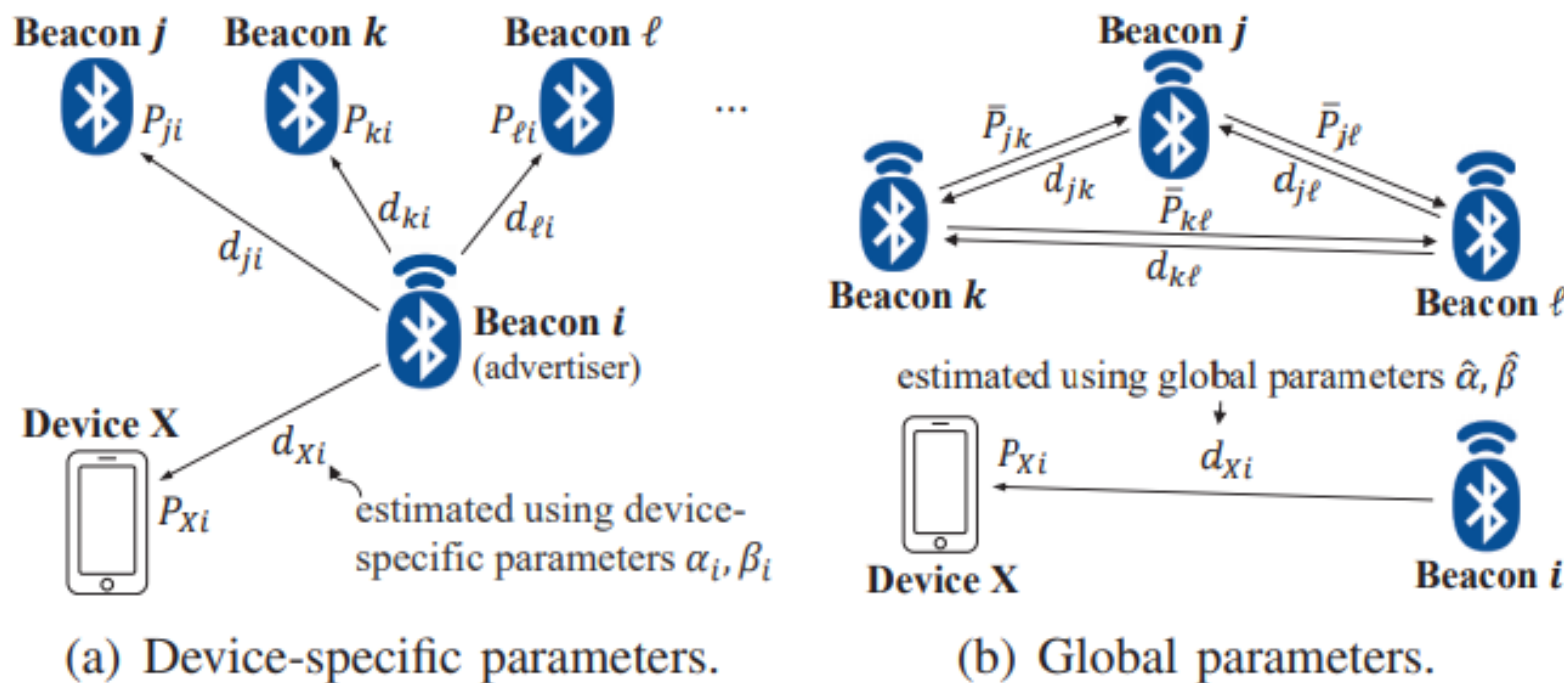
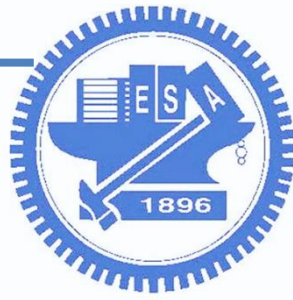


Fig. 2. Two types of regression parameters for adaptive ranging.

J.-W. Qiu, C.-P. Lin, and Y.-C. Tseng, "BLE-based Collaborative Indoor Localization with Adaptive Multi-lateration and Mobile Encountering", IEEE Wireless Communications and Networking Conference (WCNC), 2016.



# Bluetooth (藍牙)

- 目的
  - 為了解決電腦與電器設備之間的傳輸問題
- 特色
  - 短距離無線技術 (10 - 100m)
  - 使用 2.4 至 2.485 GHz 的 ISM 頻段
- Bluetooth Classic: 802.15
- Bluetooth 4.0 Low Energy (BLE): 802.15.1



<https://zh.wikipedia.org/zh-tw/%E8%97%8D%E7%89%99>

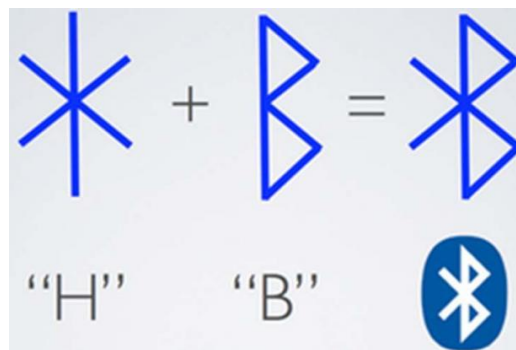


# 藍牙起源

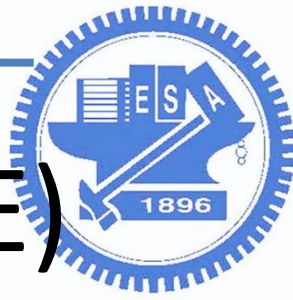


## □ 歷史

- 十世紀國王的名字 (Harald Blåtand)
  - 統一了因宗教戰爭和領土爭議而分裂的挪威與丹麥而聞名於世
  - 喜歡吃藍莓，因此牙齒都變成藍色 (Blue tooth)
  - 另一說，他的牙齒很差，看起來像藍色(blue, dark, black)
  - 他喜歡穿藍色的服飾，當時的藍色有昂貴、尊爵、不凡的意思
- 由 Ericsson 在 1994 年創製，希望為裝置間的通訊創造一組統一規則（標準化協定），以解決用戶間互不相容的移動電子裝置



不要寫成藍芽喔！



# Bluetooth Low Energy (BLE)

- 一種無線個人區域網路 (Wireless PAN) 的技術
- 出現目的：低成本，低耗電 (CR2032 電池可用 1 年)
- BT4 分 Classic(BR/EDR), High Speed(HS), Low Energy

	Classic	BLE
Throughput	2 ~ 3 Mbps	0.2 Mbps
Range	50 ~ 300 m	10 ~ 30 m
Power consumption	1 W	0.01 ~ 0.5 W
Connection time	5 s	0.1 s





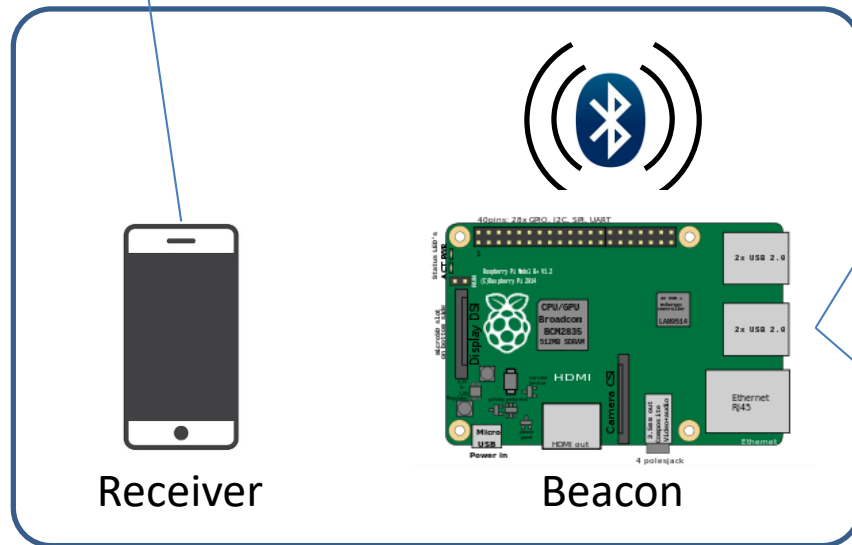
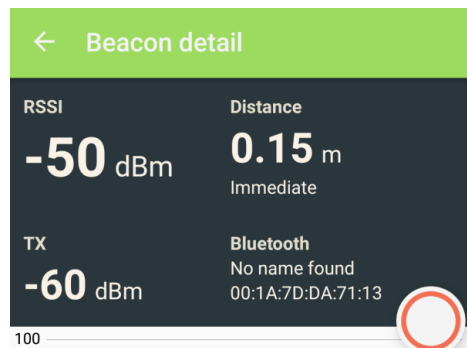
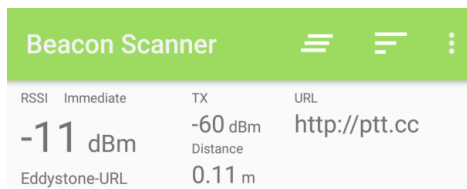
# Bluetooth on PI

- Does your BT device support BLE?
  - `hciconfig -a hci0 features` (“LE support”)

```
pi@raspberrypi:~$ hciconfig -a hci0 features
hci0:  Type: BR/EDR  Bus: USB
      BD Address: 00:1A:7D:DA:71:13  ACL MTU: 310:10  SCO MTU: 64:8
      Features page 0: 0xff 0xff 0x8f 0xfe 0xdb 0xff 0x5b 0x87
                        <3-slot packets> <5-slot packets> <encryption> <slot offset>
                        <timing accuracy> <role switch> <hold mode> <sniff mode>
                        <park state> <RSSI> <channel quality> <SCO link> <HV2 packets>
                        <HV3 packets> <u-law log> <A-law log> <CVSD> <paging scheme>
                        <power control> <transparent SCO> <broadcast encrypt>
                        <EDR ACL 2 Mbps> <EDR ACL 3 Mbps> <enhanced iscan>
                        <interlaced iscan> <interlaced pscan> <inquiry with RSSI>
                        <extended SCO> <EV4 packets> <EV5 packets> <AFH cap. slave>
                        <AFH class. slave> <LE support> <3-slot EDR ACL>
                        <5-slot EDR ACL> <sniff subrating> <pause encryption>
                        <AFH cap. master> <AFH class. master> <EDR eSCO 2 Mbps>
                        <EDR eSCO 3 Mbps> <3-slot EDR eSCO> <extended inquiry>
                        <LE and BR/EDR> <simple pairing> <encapsulated PDU>
                        <non-flush flag> <LSTO> <inquiry TX power> <EPC>
                        <extended features>
      Features page 1: 0x03 0x00 0x00 0x00 0x00 0x00 0x00 0x00
```



# BLE beacon



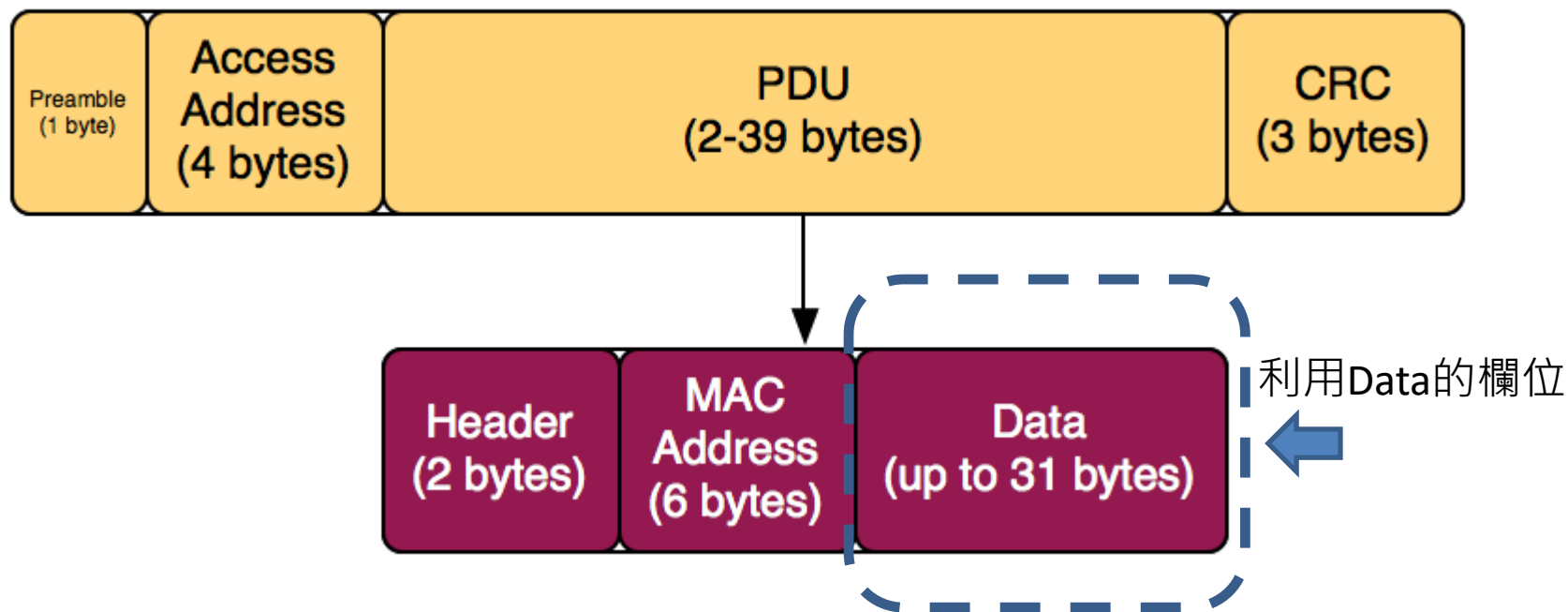
  
Eddystone

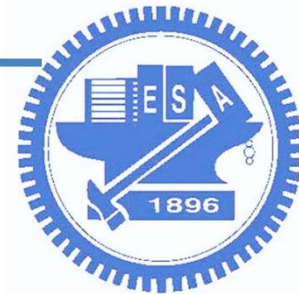
  
iBeacon



# BLE frame format

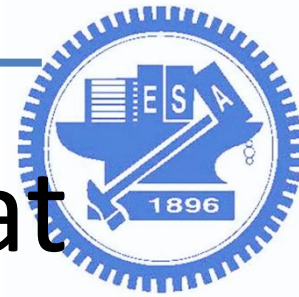
- 1 byte preamble
- 4 byte access address
- 2-39 bytes advertising channel PDU
- 3 bytes CRC



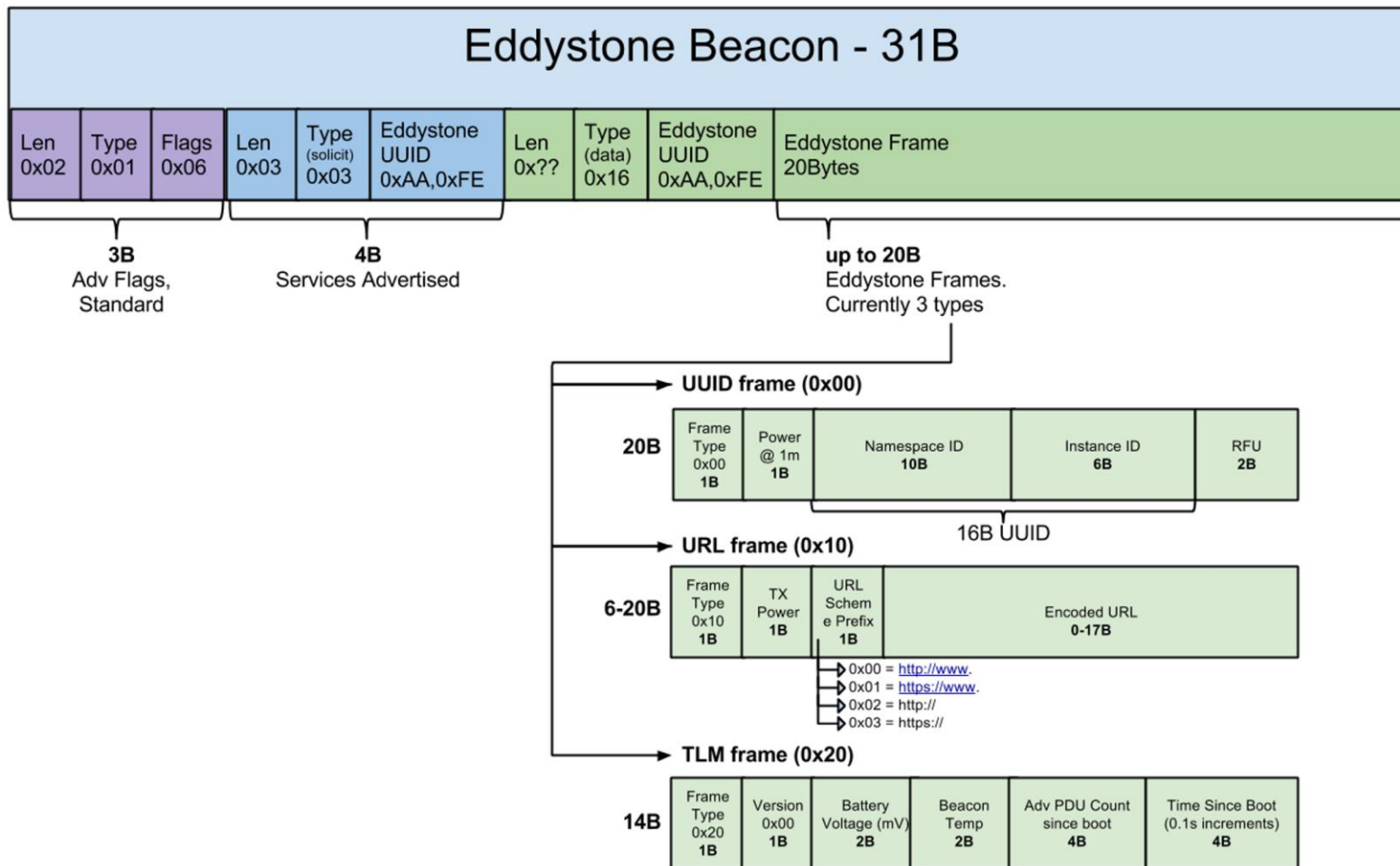


# A. Eddystone

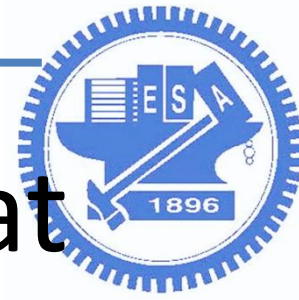
- Eddystone is a protocol specification that defines a Bluetooth low energy (BLE) message format for proximity beacon messages.
- Design Goals
  - Works well with Android and iOS Bluetooth developer APIs
  - Straightforward implementation on a wide range of existing BLE devices
  - Flexible architecture permitting development of new frame types
  - Fully compliant with the Bluetooth Core Specification



# A. Eddystone Frame format



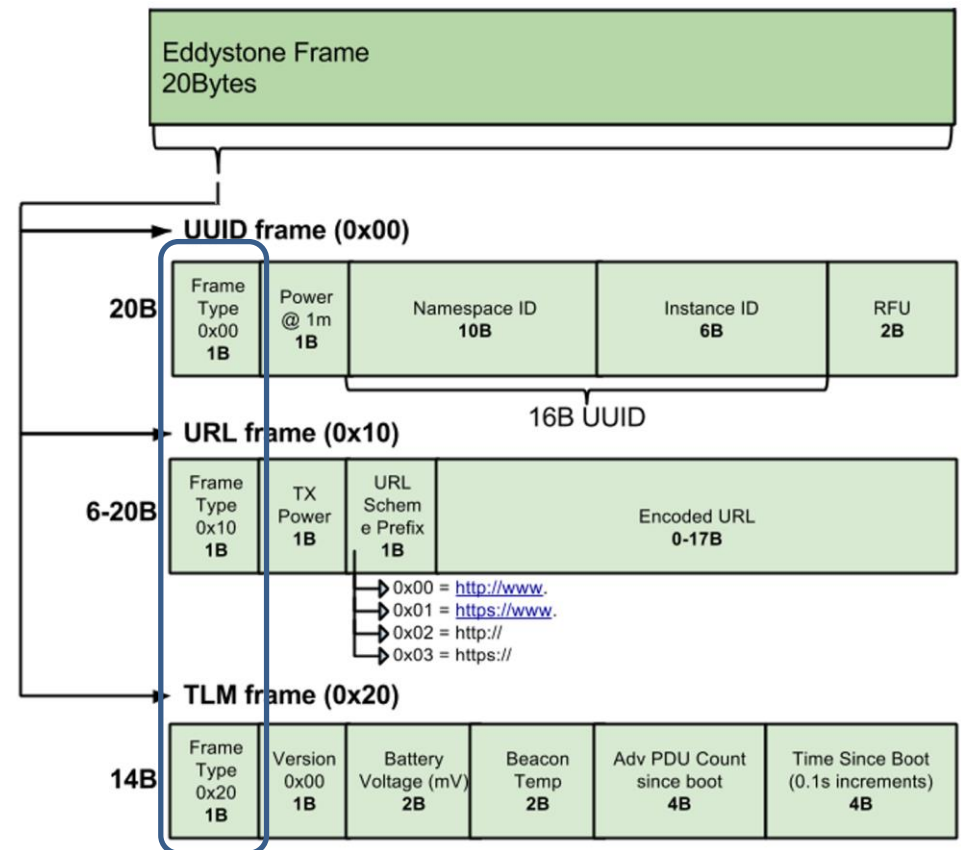
[https://developer.mbed.org/teams/Bluetooth-Low-Energy/code/BLE\\_EddystoneBeacon\\_Service/file/dfb7fb5a971b/Eddystone.h](https://developer.mbed.org/teams/Bluetooth-Low-Energy/code/BLE_EddystoneBeacon_Service/file/dfb7fb5a971b/Eddystone.h)



# A. Eddystone Frame format

## □ Eddystone Protocol Specification

Frame Type	High-Order 4 bits	Byte Value
UID	0000	0x00
URL	0001	0x10
TLM	0010	0x20
EID	0011	0x30
RESERVED	0100	0x40







# A. Eddystone by Python

- Source: <https://github.com/google/eddystone>
- Download tool
  - wget  
<https://raw.githubusercontent.com/google/eddystone/master/eddystone-url/implementations/linux/advertise-url>
  - chmod +x advertise-url
  - sudo ./advertise-url -u http://ptt.cc // adversary URL
  - sudo ./advertise-url -s // stop



# A. Eddystone

## □ 手機端可安裝app查看Eddystone訊息



iBeacon & Eddystone Scanner

flurp laboratories 工具

★★★★★ 132

這個應用程式與您的部分裝置相容。

已安裝

**Beacon Scanner**

RSSI	Immediate	TX	URL
-11 dBm	-60 dBm	Distance	http://ptt.cc
Eddystone-URL	0.11 m		

**Beacon detail**

<b>RSSI</b>	<b>Distance</b>
<b>-50 dBm</b>	<b>0.15 m</b>
	Immediate
<b>TX</b>	<b>Bluetooth</b>
<b>-60 dBm</b>	No name found
	00:1A:7D:DA:71:13

**Beacon Scanner**

RSSI	Immediate	TX	URL
-11 dBm	-60 dBm	Distance	http://ptt.cc
Eddystone-URL	0.11 m		

RSSI	Far	TX	UUID	
-87 dBm	-70 dBm	Distance	Major	Minor
iBeacon	2.91 m	10	231	
RSSI	Far	TX	UUID	
-90 dBm	-70 dBm	Distance	Major	Minor
iBeacon	3.03 m	10	236	
RSSI	Far	TX	UUID	
-88 dBm	-70 dBm	Distance	Major	Minor
iBeacon	3.16 m	10	237	
RSSI	Far	TX	UUID	
-92 dBm	-70 dBm	Distance	Major	Minor
iBeacon	3.16 m	10	232	
RSSI	Far	TX	UUID	
-94 dBm	-70 dBm	Distance	Major	Minor
iBeacon	3.16 m	10	232	

**Beacon detail**

<b>RSSI</b>	<b>Distance</b>
<b>-50 dBm</b>	<b>0.15 m</b>
	Immediate
<b>TX</b>	<b>Bluetooth</b>
<b>-60 dBm</b>	No name found
	00:1A:7D:DA:71:13

100  
80  
60  
40  
20  
0

URL  
http://ptt.cc



# A. Eddystone by Python

- `sudo ./advertise-url -u http://hk.3345678`

```
(COM8) [80x24]
連線(C) 編輯(E) 檢視(V) 視窗(W) 選項(O) 說明(H)
pi@raspberrypi:~$ sudo ./advertise-url -u http://hk.3345678
Advertising: http://hk.3345678
pi@raspberrypi:~$
```

Beacon Scanner		
RSSI 3分鐘前	TX	URL
-67 dBm	-60 dBm	http://hk
	Distance	.3345678
Eddystone-URL	1.15 m	

- `sudo ./advertise-url -s`

```
(COM8) [80x24]
連線(C) 編輯(E) 檢視(V) 視窗(W) 選項(O) 說明(H)
pi@raspberrypi:~$ sudo ./advertise-url -s
Stopping advertising
pi@raspberrypi:~$
```



# A. Eddystone

## □ Translate URL to Eddystone message

□ Ex: `http://ptt.cc`

□ ASCII table <https://zh.wikipedia.org/wiki/ASCII>

數值 (16進位)	網址
02	http://
70	p
74	t
74	t
2e	.
63	c
63	c

Decimal	Hex	Expansion
0	0x00	http://www.
1	0x01	https://www.
2	0x02	http://
3	0x03	https://



# A. Eddystone by hcitool

## □ 使用bluetooth工具傳送網址廣播

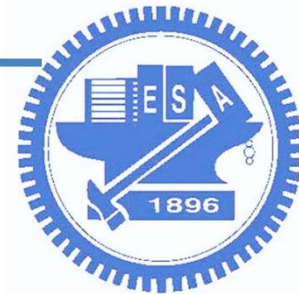
```
sudo hciconfig hci0 leadv 3 // 啟用藍牙的低耗能廣告(LE advertising)模式  
sudo hciconfig hci0 noscan // 並關閉掃描功能
```

```
sudo hcitool -i hci0 cmd 0x08 0x0008 14 02 01 1a 03 03 aa fe 0c 16 aa fe 10 ed  
02 70 74 74 2e 63 63 00 00 00 00 00 00 00 00 00 00 00 00 //傳送Eddystone frame
```

↑ <http://ptt.cc>

```
sudo hciconfig hci0 noleadv //停止廣告
```

數值 (16進位)	網址
02	http://
70	p
74	t
74	t
2e	.
63	c
63	c

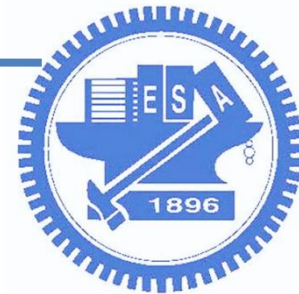


# A. Eddystone by hcitool

```
sudo hcitool -i hci0 cmd 0x08 0x0008 14 02 01 1a 03 03 aa fe 0c 16 aa fe 10 ed  
02 70 74 74 2e 63 63 00 00 00 00 00 00 00 00 00 00 00 00
```

- 0x08 0x0008: set the ad package
  - #OGF = Operation Group Field = Bluetooth Command Group = 0x08
  - #OCF = Operation Command Field = HCI\_LE\_Set\_Advertising\_Data = 0x0008
- 14: the ENTIRE following data packet in bytes (16進位的14 = 20 byte) 20 byte
- 02 01 1a: Eddystone Adv Flags
  - 0x06 - The device is BLE only. The full Bluetooth stack is not supported.
  - 0x1A - The device can be used as BLE as well as full Bluetooth Controller/Host simultaneously.
- 03 03 aa fe: Eddystone service adv
- 0c: length (12 byte)
- 16: type (data)
- aa fe: Eddystone UUID
- 10: URL frame type
- ed: TX power
- **02 70 74 74 2e 63 63**: <http://ptt.cc>, 共 7 byte
- 00 00 00 00 00 00 00 00 00 00 00 00: 共 10 byte

12 byte

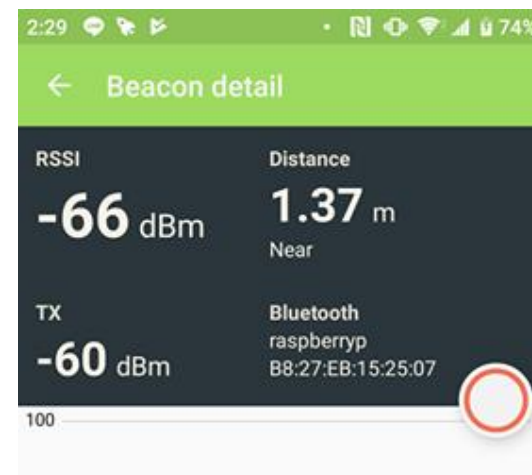
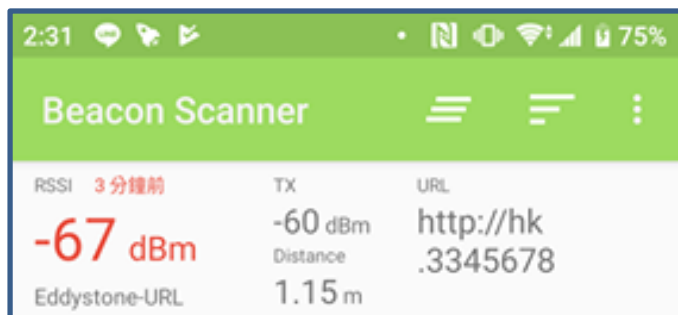


# Quiz 1

- Advertise an URL with your student ID.
  - Ex: nctu.5566

```
(COM8) [80x24]
連線(C) 編輯(E) 檢視(V) 視窗(W) 選項(O) 說明(H)
pi@raspberrypi:~$ sudo ./advertise-url -u http://hk.3345678
Advertising: http://hk.3345678
pi@raspberrypi:~$
```

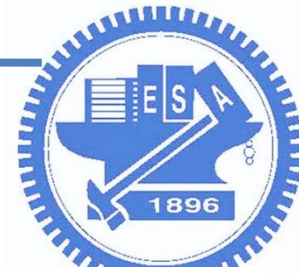
- Show/Capture your scanning result





- [illegible]





# iBeacon

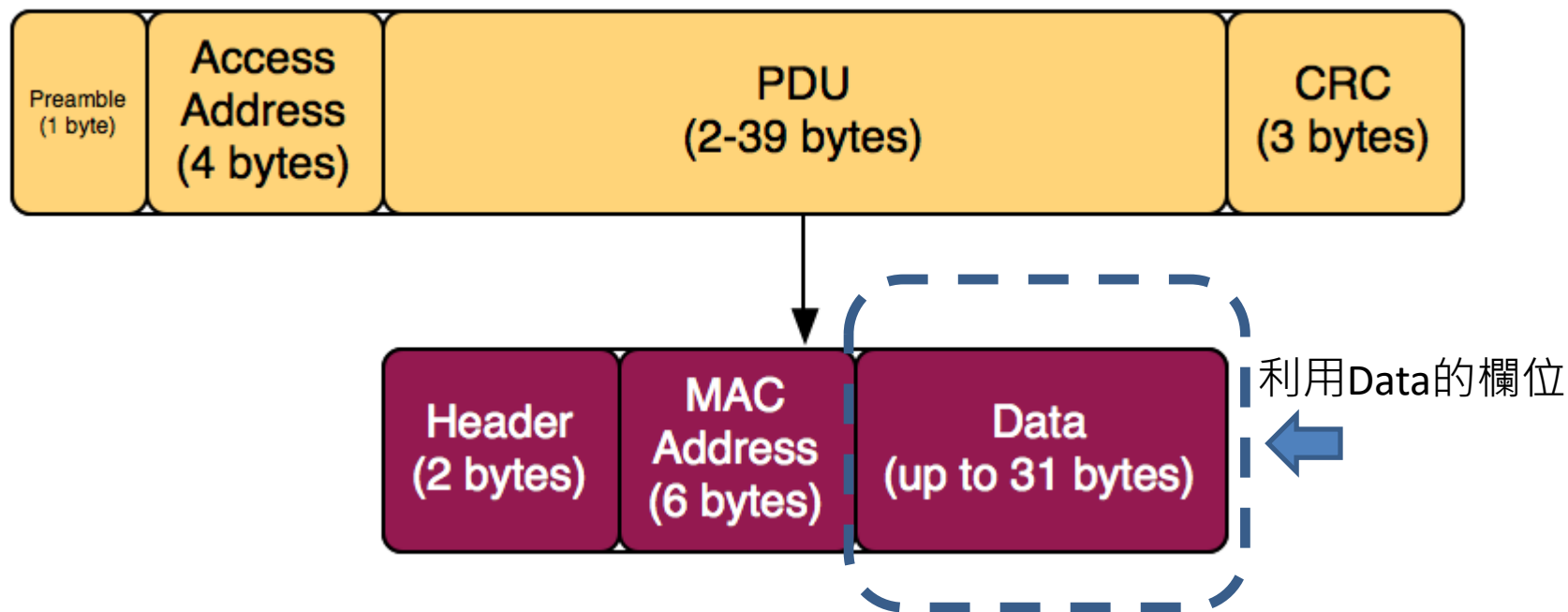
From welcoming people as they arrive at a sporting event to providing information about a nearby museum exhibit, iBeacon opens a new world of possibilities for location awareness, and countless opportunities for interactivity between iOS devices and iBeacon hardware.

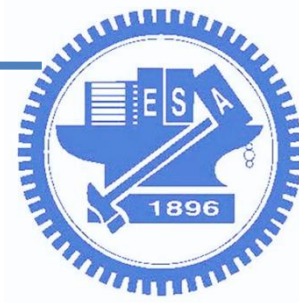
<https://developer.apple.com/ibeacon/>



# BLE frame format

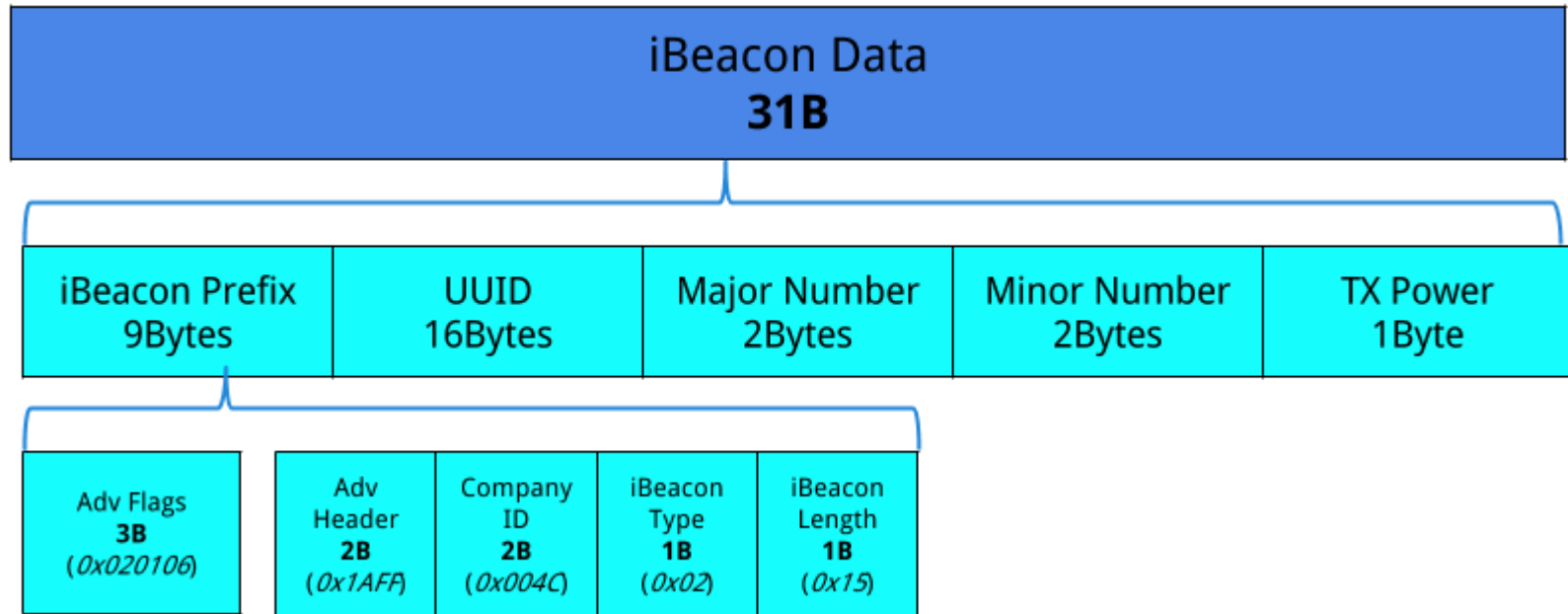
- 1 byte preamble
- 4 byte access address
- 2-39 bytes advertising channel PDU
- 3 bytes CRC





iBeacon

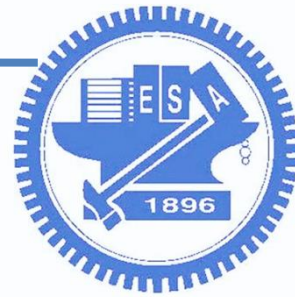
## B. iBeacon format





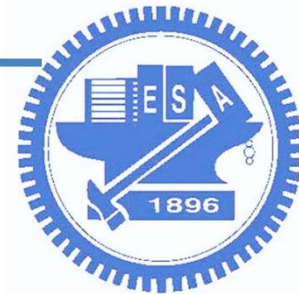
## B. iBeacon tool

- Dependencies
  - `sudo apt-get install bluetooth bluez-utils blueman`
- `git clone https://github.com/dburr/linux-ibeacon`
- `cd linux-ibeacon/`
- `chmod +x ibeacon`
- `sudo ./ibeacon -u your_uuid -M your_majorID -m your_minorID`
- `sudo ./ibeacon -z`



## B. iBeacon tool

- What is UUID?
  - UUID stands for Universally Unique Identifier. It contains 32 hexadecimal digits, split into 5 groups, separated by hyphens and should look something like this:  
**f7826da6-4fa2-4e98-8024-bc5b71e0893e**
- Each of the 5 groups must contain the following number of characters per section:
  - First section: 8
  - Second section: 4
  - Third section: 4
  - Fourth section: 4
  - Fifth section: 12



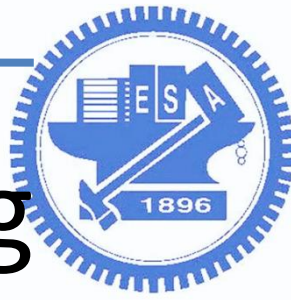
## B. Start advertising

- `sudo ./ibeacon -M 5566 -m 7788`
  - Major ID = 5566; Minor ID = 7788

```
(COM8) [80x24]
連線(C) 編輯(E) 檢視(V) 視窗(W) 選項(O) 說明(H)
pi@raspberrypi:~/linux-ibeacon$ sudo ./ibeacon -M 5566 -m 7788
Advertising on hci0 with:
    uuid: 0xE20A39F473F54BC4A12F17D1AD07A961
major/minor: 5566/7788 (0x15BE/0x1E6C)
    power: 200 (0xC8)
```

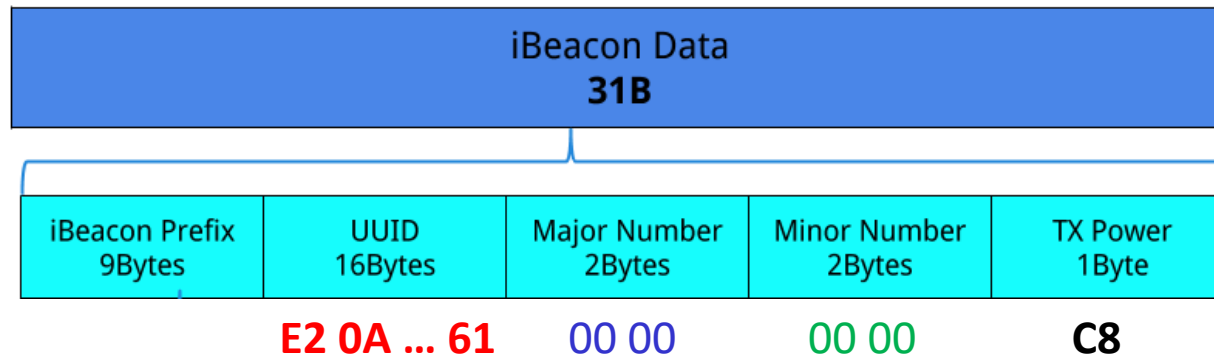
- Stop: `sudo ./ibeacon -z`

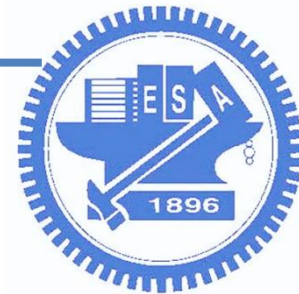
```
(COM8) [80x24]
連線(C) 編輯(E) 檢視(V) 視窗(W) 選項(O) 說明(H)
pi@raspberrypi:~/linux-ibeacon$ sudo ./ibeacon -z
Downing iBeacon on hci0
```



## B. Advertising by hciconfig

- ❑ `sudo hciconfig hci0 up`
- ❑ `sudo hciconfig hci0 leadv 3`
- ❑ `sudo hciconfig hci0 noscan`
- ❑ `sudo hcidtool -i hci0 cmd 0x08 0x0008 1E 02 01 06 1A FF 00 4C 02 15 E2 0A 39 F4 73 F5 4B C4 A1 2F 17 D1 AD 07 A9 61 00 00 00 00 C8 00`

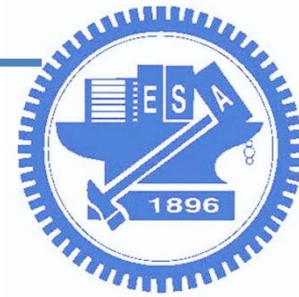




## B. iBeacon by hcitool

- ❑ `sudo hcitool -i hci0 cmd 0x08 0x0008 1E 02 01 06 1A FF 00 4C 02 15 E2 0A 39 F4 73 F5 4B C4 A1 2F 17 D1 AD 07 A9 61 00 00 00 00 C8`
- ❑ 0x08 0x0008: set the ad package
  - ❑ #OGF = Operation Group Field = Bluetooth Command Group = 0x08
  - ❑ #OCF = Operation Command Field = HCI\_LE\_Set\_Advertising\_Data = 0x0008
- ❑ 1E // the ENTIRE following data packet in bytes (31 byte)
- ❑ 02 01 06 // set the flags for General Discoverable and BR/EDR not supported
- ❑ 1A FF // the length of the Manufacturer specific data field will be 26 bytes
- ❑ 00 4C // Company ID
- ❑ 02 // iBeacon type, ID
- ❑ 15 // length of remaining data in bytes  
(16B UUID+ 2B major, 2B minor, 1B Txpower)
- ❑ **E2 0A 39 F4 73 F5 4B C4 A1 2F 17 D1 AD 07 A9 61** // UUID
- ❑ **00 00** // Major ID
- ❑ **00 00** // Minor ID
- ❑ **C8** // Tx power. C8 = 11001000 --(2's)--> -56dBm

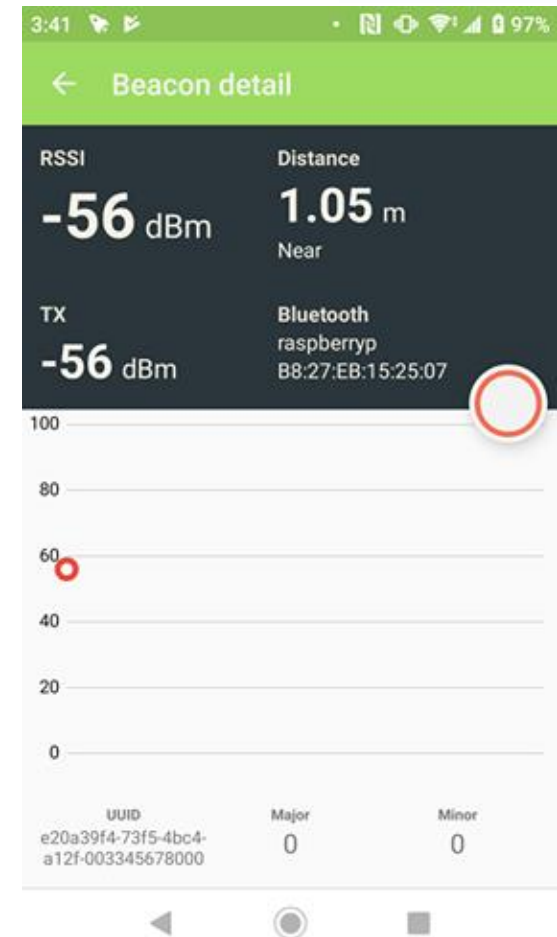


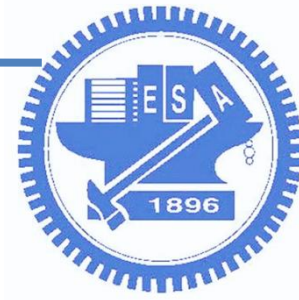


# Quiz 2

- Advertise with your student ID.
  - UUID should contain your student ID
  - Ex: **3345678**
- Show/Capture your scanning result

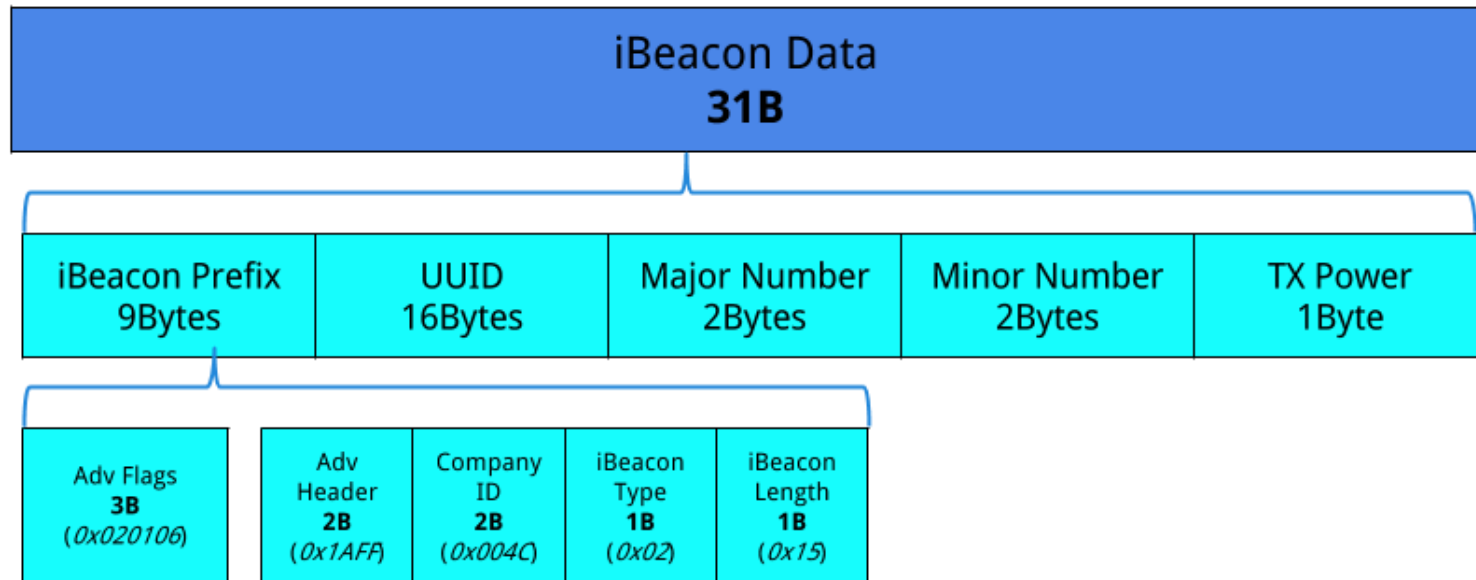
RSSI	Near		TX	UUID
-57 dBm			-56 dBm	003345678000
			Distance	Major Minor
iBeacon			1.16 m	0 0





# Discussion 2

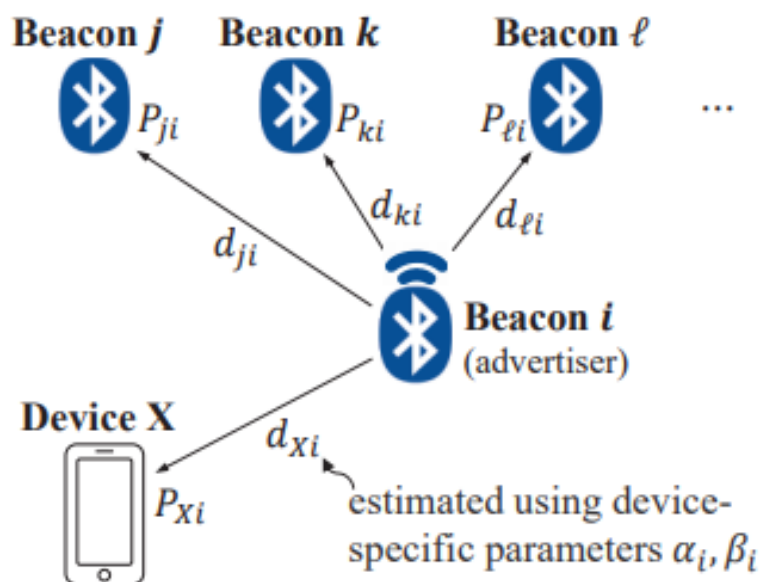
- What is the maximum value of major and minor ID?  
How to calculate this value?
  - You can refer to iBeacon data format



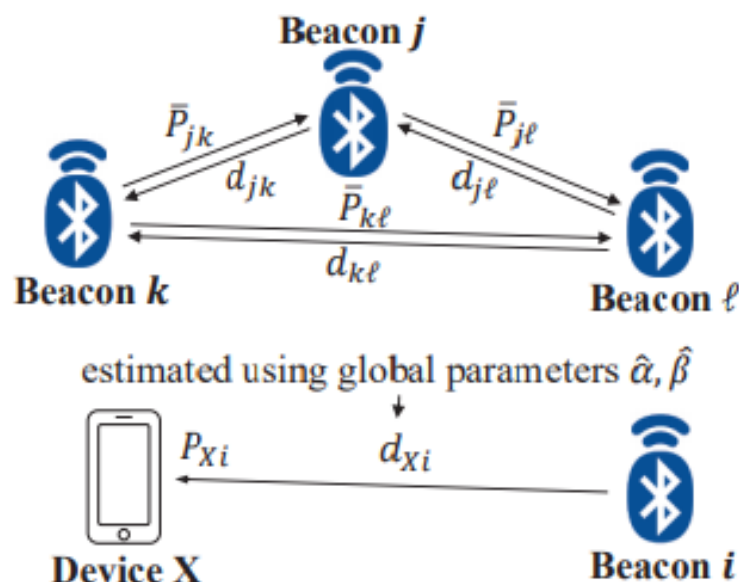


# Discussion 3

- BLE beacon can be used for indoor localization.
- If we put a lot of beacons, can we get a more precise result?



(a) Device-specific parameters.



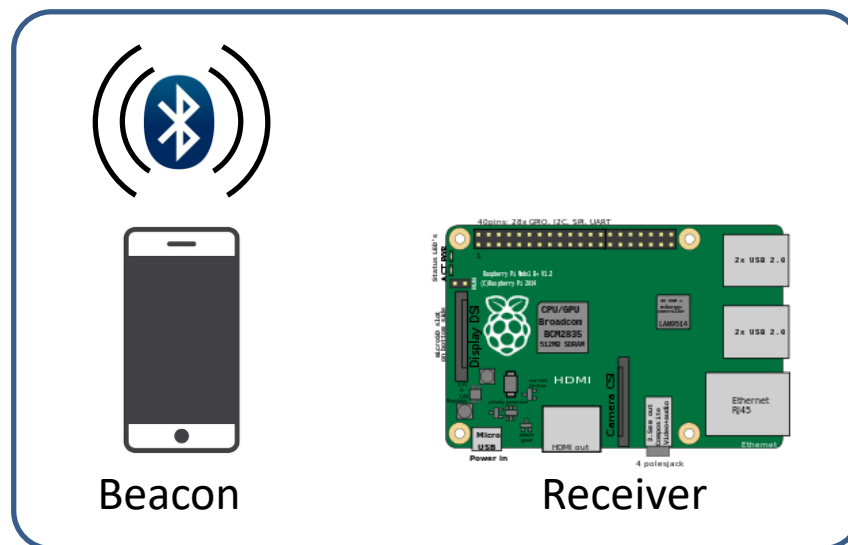
(b) Global parameters.

Fig. 2. Two types of regression parameters for adaptive ranging.



# Scan a nearby beacon

- How to calculate distance?
  - Based on **RSSI** (Received signal strength indication) and **TX power** in BLE frame
  - Use **propagation model (path loss)** to calculate
    - Ex: Free Space, Two-ray ground-reflection, Friis Transmission ... etc



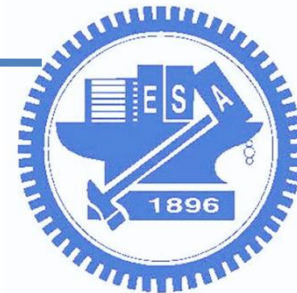
$$\text{Path Loss} = \text{TX\_power(dBm)} - \text{RX\_power(dBm)}$$



# Scan a nearby beacon

- Scan a specific ibeacon uuid:
  - Download and unzip sample code
  - Run: `sudo python ble_sample_code.py`
    - `uuid= 00000000111111110000000000556601`

```
pi@raspberrypi:~/9.blescan $ sudo python ble_sample_code.py
ble thread started
uuid: 00000000111111110000000000556601
major: 5566 , minor: 3388 , txpower: -80
rssi -64
-----
uuid: 00000000111111110000000000556601
major: 5566 , minor: 3388 , txpower: -80
rssi -62
-----
uuid: 00000000111111110000000000556601
major: 5566 , minor: 3388 , txpower: -80
rssi -65
-----
uuid: 00000000111111110000000000556601
major: 5566 , minor: 3388 , txpower: -80
rssi -62
-----
uuid: 00000000111111110000000000556601
major: 5566 , minor: 3388 , txpower: -80
rssi -62
```



# Scan result

fullpacket: 01 00 01 1d ec 5b b9 be 65 12 02 01 1a 02 0a 0c 0b ff 4c 00 10 06 4b 1e c2 bc fd 7d be None

Adstring= 65:be:b9:5b:ec:1d,be651202011a020a0c0bff4c0010064b,7874,48381,125,-66

fullpacket: 01 03 01 aa eb b1 8e 89 3e 1f 1e ff 06 00 01 09 20 02 1f 4a 06 42 b9 bf 03 04 2b 72 9e bf 0c 93 83 cc c6 d7 0f 17 a1 13 73 d5 None

Adstring= 3e:89:8e:b1:eb:aa,0642b9bf03042b729ebf0c9383ccc6d7,3863,41235,115,-43

fullpacket: 01 00 01 b1 a3 30 45 5c 7d 0e 02 01 06 0a ff 4c 00 10 05 01 1c c8 79 e7 a1 None

Adstring= 7d:5c:45:30:a3:b1,b1a330455c7d0e0201060aff4c001005,284,51321,-25,-95

fullpacket: 01 04 01 b1 a3 30 45 5c 7d 00 a1 None

Adstring= 7d:5c:45:30:a3:b1,010401b1a3,12357,23677,0,-95

fullpacket: 01 02 01 0c 5b f5 25 bf 55 1b 1a ff 4c 00 02 15 00 00 00 00 11 11 11 11 00 00 00 00 00 55 66 01 15 be 0d 3c b0 b7 None

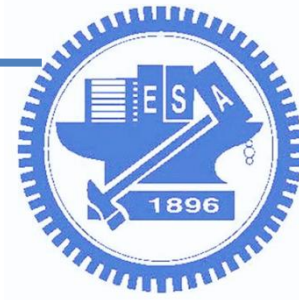
Adstring= 55:bf:25:f5:5b:0c,00000000111111110000000000556601,5566,3388,-80,-73

.....

uuid: 00000000111111110000000000556601

major: 5566 , minor: 3388 , txpower: -80

rsssi -73



# Solution by AltBeacon

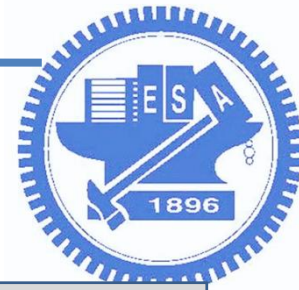
- AltBeacon: The Open and Interoperable Proximity Beacon specification

[CurveFittedDistanceCalculator.java]

```
public double calculateDistance(int txPower, double rssi) {
    if (rssi == 0) {
        return -1.0; // if we cannot determine accuracy, return -1.
    }
    LogManager.d(TAG, "calculating distance based on mRssi of %s and txPower of %s", rssi, txPower);

    double ratio = rssi*1.0/txPower;
    double distance;
    if (ratio < 1.0) {
        distance = Math.pow(ratio,10);
    }
    else {
        distance = (mCoefficient1)*Math.pow(ratio,mCoefficient2) + mCoefficient3;
    }
    LogManager.d(TAG, "avg mRssi: %s distance: %s", rssi, distance);
    return distance;
}
```

Source code: <https://github.com/AltBeacon/android-beacon-library/blob/b9876b45acd6c72c84b8d0325062259020eb89ba/lib/src/main/java/org/altbeacon/beacon/distance/CurveFittedDistanceCalculator.java>



# Calculate distance

```
ratio = rssi*1.0/txPower;  
if (ratio < 1.0) {  
    distance = Math.pow(ratio,10);  
}  
else {  
    distance = (Coefficient1)*Math.pow(ratio,Coefficient2) + Coefficient3;  
}  
return distance;
```

$$dist = \left( \frac{RSSI}{TXpower} \right)^{10}$$
$$dist = \alpha \left( \frac{RSSI}{TXpower} \right)^{\beta} + \gamma$$

```
{  
  "coefficient1": 0.42093,  
  "coefficient2": 6.9476,  
  "coefficient3": 0.54992,  
  "version":"4.4.2",  
  "build_number":"KOT49H",  
  "model":"Nexus 4",  
  "manufacturer":"LGE"  
},
```

```
{  
  "coefficient1": 0.1862616782,  
  "coefficient2": 8.235367435,  
  "coefficient3": -0.45324519,  
  "version":"6.0",  
  "build_number":"MPE24.49-18",  
  "model":"XT1092",  
  "manufacturer":"Motorola",  
}
```

Coefficient source:

[https://github.com/AltBeacon/android-beacon-](https://github.com/AltBeacon/android-beacon-library/blob/b9876b45acd6c72c84b8d0325062259020eb89ba/lib/src/main/resources/model-distance-calculations.json)

[library/blob/b9876b45acd6c72c84b8d0325062259020eb89ba/lib/src/main/resources/model-distance-calculations.json](https://github.com/AltBeacon/android-beacon-library/blob/b9876b45acd6c72c84b8d0325062259020eb89ba/lib/src/main/resources/model-distance-calculations.json)



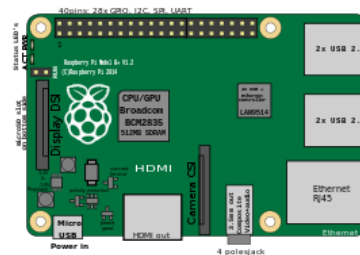


# Quiz 3

- Based on `ble_sample_code.py`, after receiving `txpower` and `rss`, **calculate the distance**
  - Req: Set your smartphone as an iBeacon. Put your uuid in python
- Refer to the formula by AltBeacon (use Nexus 4's coefficient)



Beacon

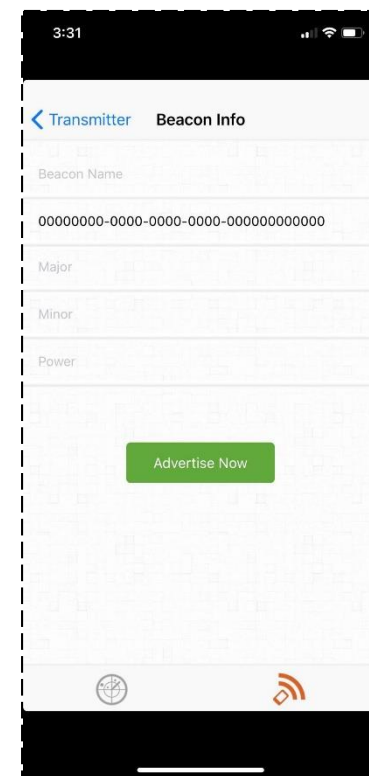
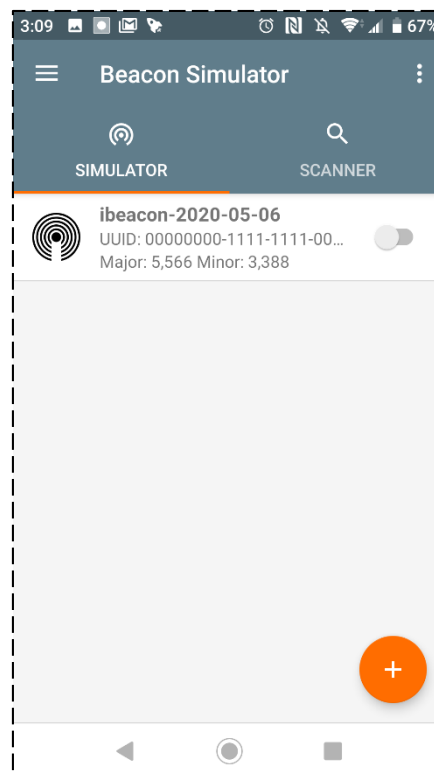
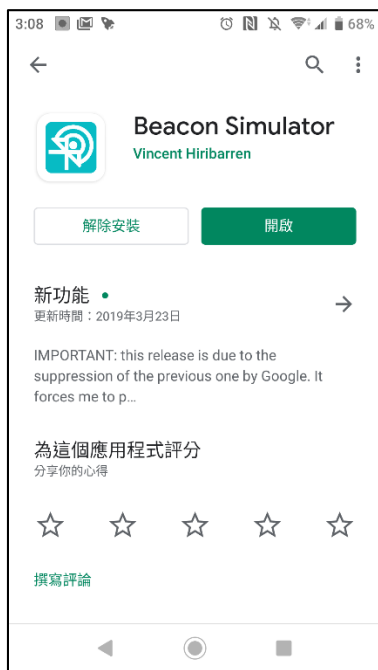


Receiver



# iBeacon generator app

- Android: Beacon Simulator
- iOS: Locate Beacon





# Summary

- Practice Lab (Eddystone and iBeacon)
- Write down the answer for discussion
  - **Discussion** (Deadline: Before 5/14, 12:00)
    - 1. translating URL to Eddystone message
    - 2. the maximum value of major and minor ID
    - 3. more beacons, more precise location?
- Demonstrate **Quiz 1 to Quiz 3** to TAs
  - **Quiz1: Eddystone + your student ID**
  - **Quiz2: iBeacon + your student ID**
  - **Quiz3: Calculate distance from iBeacon to your PI**
    - Deadline: Before 5/7, 15:10
    - Late Demo: Before 5/14, 15:10