



المدرسة الوطنية للمهندسين بقرطاج  
Ecole Nationale d'Ingénieurs de Carthage

École Nationale d'Ingénieurs de  
Carthage

---


*Module*

# Technologies de l'Internet des Objets

---

*F. Rouissi, Technologies de l'IoT, 3<sup>ème</sup> année Génie Info, Septembre 2021*

1



المدرسة الوطنية للمهندسين بقرطاج  
Ecole Nationale d'Ingénieurs de Carthage

Structuration du cours


---

1. Introduction à l'IoT
2. Architecture des systèmes IoT
3. Technologies de l'IoT
4. Plateformes IoT

---

*F. Rouissi, Technologies de l'IoT, 3<sup>ème</sup> année Génie Info, Septembre 2021*

2



المدرسة الوطنية للمهندسين بقرطاج  
Ecole Nationale d'Ingénieurs de Carthage

# École Nationale d'Ingénieurs de Carthage

---


*Chapitre 2*

## Architectures des systèmes IoT

---

*F. Rouissi, Technologies de l'IoT, 3<sup>ème</sup> année Génie Info, Septembre 2021*

3



المدرسة الوطنية للمهندسين بقرطاج  
Ecole Nationale d'Ingénieurs de Carthage

# Introduction à l'Internet des Objets

---

- Composants IoT
- Connectivité entre les composants IoT
- Hardware, systèmes embarqués & gestion d'énergie
- Systèmes d'exploitation pour les dispositifs à ressources limitées

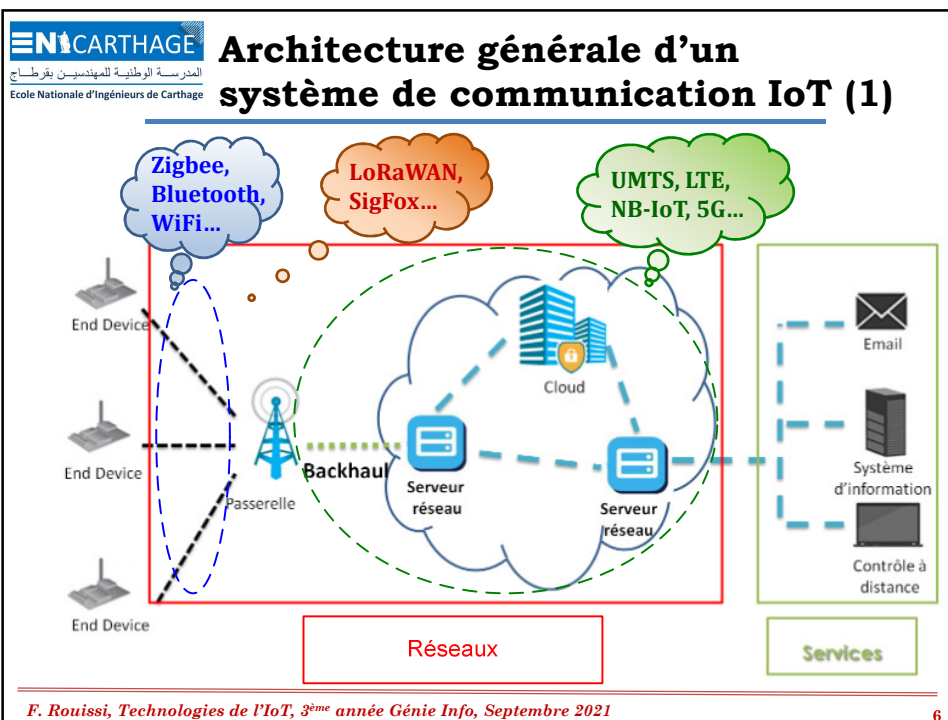
---

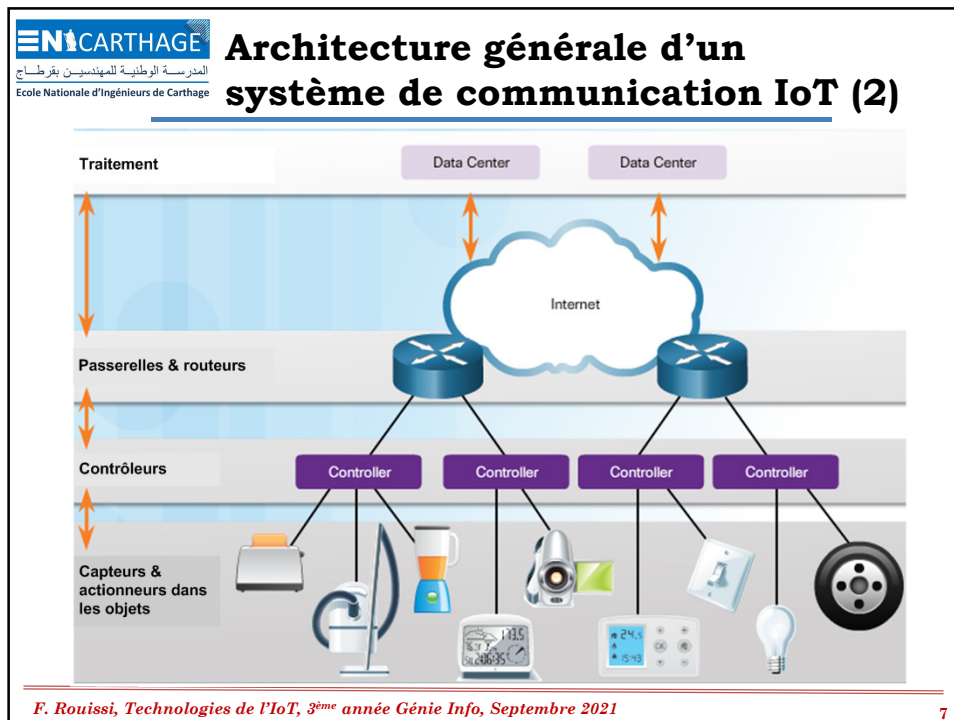
*F. Rouissi, Technologies de l'IoT, 3<sup>ème</sup> année Génie Info, Septembre 2021*

4

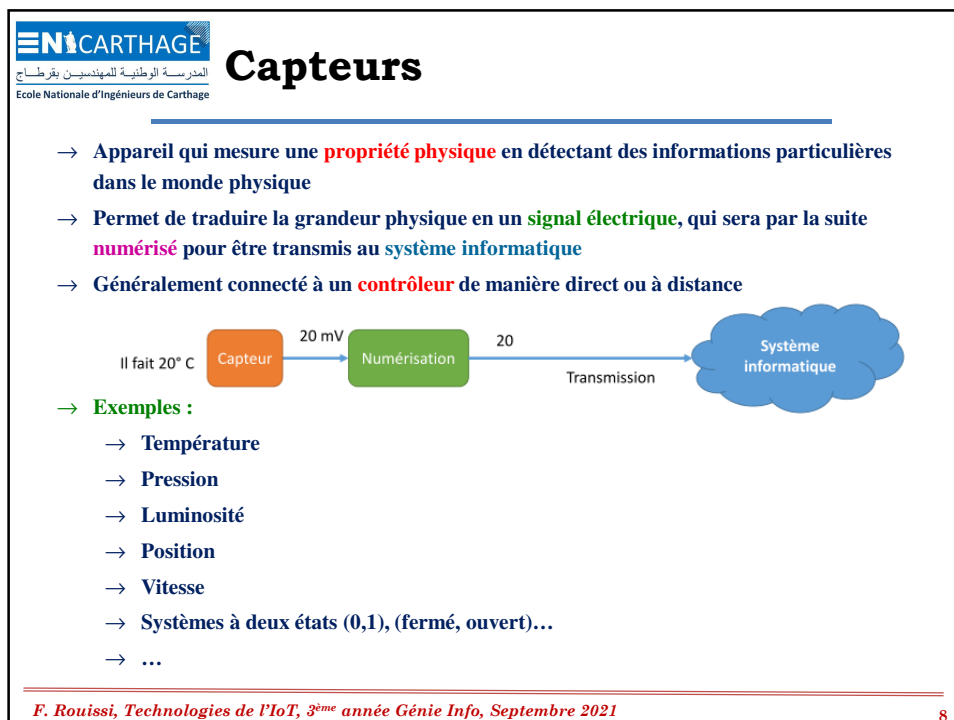
## Chapitre 2

### Composants IoT





7



8

## Actionneur & contrôleur

- **Actionneur** = **Moteur** de base qui peut être utilisé pour contrôler un système
  - Permet d'agir dans le monde physique et changer son état
  - Prend en charge de transformer un signal électrique en **résultat physique**
  - Peut être hydraulique, électrique ou pneumatique
  - **Exemple** : allumage d'un éclairage, déclenchement d'un avertisseur sonore, commande de robots....



- **Contrôleur** = équipement en charge de la **collecte des données** via les capteurs et de la **connectivité réseau**.

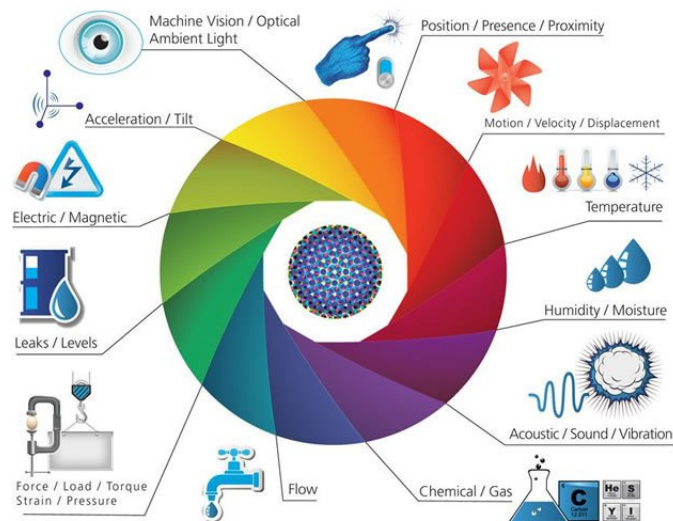
- peut avoir la capacité de **prendre des décisions** immédiates.
- peut également **envoyer des données** à distance à des ordinateurs plus puissants pour les analyser.



F. Rouissi, Technologies de l'IoT, 3<sup>ème</sup> année Génie Info, Septembre 2021

9

## Exemples de capteurs/ actionneurs



Source : IoT infographic Postcapes and Harbor Research – CC Attribution license

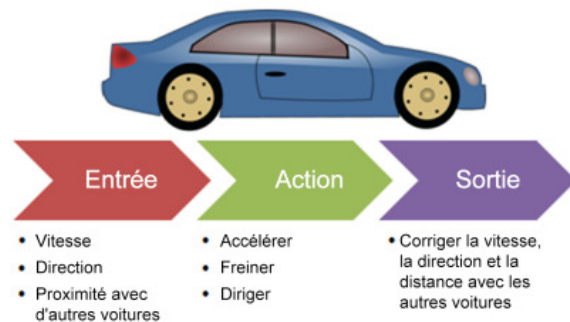
F. Rouissi, Technologies de l'IoT, 3<sup>ème</sup> année Génie Info, Septembre 2021

10

## Processus des systèmes de contrôle

→ **Processus** = une **série d'étapes ou d'actions prises** pour atteindre un résultat souhaité par le client du processus.

- Son rôle principal est de transférer les données dans des connexions : **M2M** (objet à objet), **M2P** (objet à personne), et/ou **P2P** (personne à personne ou collaboration)
- Les **boucles de rétroaction** (ou commentaires) correspondent au moment où le résultat d'un processus affecte l'entrée



F. Rouissi, Technologies de l'IoT, 3<sup>ème</sup> année Génie Info, Septembre 2021

11

## Système de contrôle

→ **Système de contrôle** = comprend un **contrôleur utilisant les entrées et les sorties** pour gérer et réguler le comportement du système et tenter d'atteindre l'état souhaité.

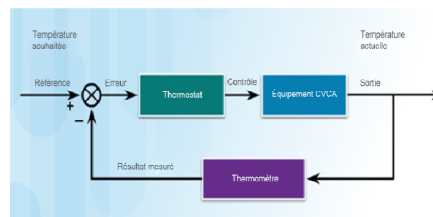
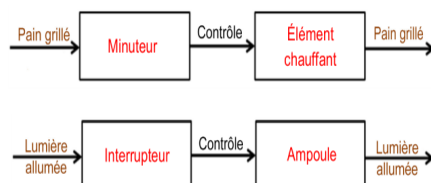
→ A pour rôle d'**appliquer les ajustements nécessaires** à une centrale afin d'atteindre le résultat souhaité

→ La partie contrôlée du système s'appelle souvent **l'équipement**.

→ Deux types de systèmes de contrôle :

→ En **boucle ouverte**

→ En **boucle fermée**



F. Rouissi, Technologies de l'IoT, 3<sup>ème</sup> année Génie Info, Septembre 2021

12

**ENICARTHAGE** المدرسة الوطنية للمهندسين بقرطاج  
Ecole Nationale d'Ingénieurs de Carthage

## Types de systèmes de contrôle (1)

### Système de contrôle en boucle ouverte :

- **N'utilisent pas** les boucles de rétroaction
- L'équipement effectue une action prédéfinie **sans vérification** du résultat
- sont souvent utilisés pour les **processus simples**

### Système de contrôle en boucle fermée :

- Tire parti des **boucles de rétroactions** pour savoir si le résultat obtenu correspond au résultat souhaité.
- Le résultat est **retransmis** à un contrôleur pour **ajuster l'équipement** pour la prochaine itération et le processus se répète

F. Rouissi, Technologies de l'IoT, 3<sup>ème</sup> année Génie Info, Septembre 2021

13

**ENICARTHAGE** المدرسة الوطنية للمهندسين بقرطاج  
Ecole Nationale d'Ingénieurs de Carthage

## Types de systèmes de contrôle (2)

### Système de contrôle en boucle fermée :

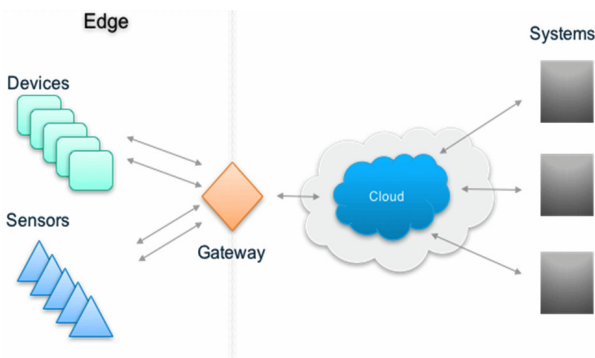
- Plusieurs types de systèmes de contrôle en boucle fermée :
  - **Contrôleurs proportionnels (P)** : basés sur l'**écart** entre le résultat mesuré et le résultat souhaité.
  - **Contrôleurs intégraux (PI)** : utilisent les **données de l'historique** pour mesurer l'écart avec le résultat souhaité
  - **Contrôleurs proportionnels, intégraux et dérivés (PID)** : incluent des données sur la rapidité à laquelle le système s'approche de la sortie souhaitée
- La plupart des systèmes intègrent des **composants interdépendants** qui contribuent au résultat ou l'influencent

F. Rouissi, Technologies de l'IoT, 3<sup>ème</sup> année Génie Info, Septembre 2021

14

## Passerelles IoT (1)

- Fait **le lien** entre l'objet et Internet
- Peut être un **simple smartphone** (BLE : Bluetooth Low Energy), un **routeur**, ou un **cœur de réseau**
- Généralement lié à un **opérateur** publique ou privé
- Joue un rôle crucial dans la liaison entre tout type de **dispositifs IoT**, **le réseau**, **le cloud** et/ou le centre de données
- Peut être un matériel séparé (hardware), ou un logiciel intégré (software), et très souvent une **combinaison des deux**, pour offrir un meilleur service et plus de **fonctionnalités** dans le réseau IoT

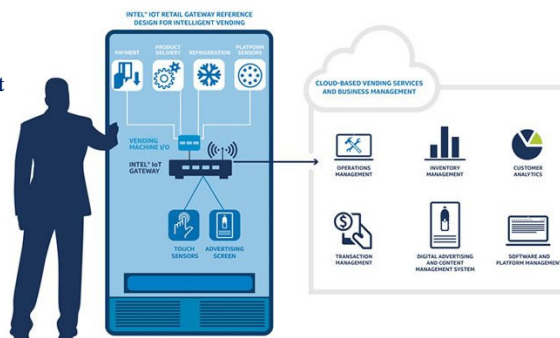


F. Rouissi, Technologies de l'IoT, 3<sup>ème</sup> année Génie Info, Septembre 2021

15

## Passerelles IoT (2)

- Appelé **nœud de bord (Edge node)** dans les réseaux de capteurs, et effectue un **traitement et stockage** local, avec en général la présence d'une interface utilisateur.
- Fonctionnalités de base :
  - **Connectivité** avec le cœur du réseau, **gestion de routage** des données
  - **Cryptage**, décryptage et **agrégation** des données
  - **Traduction** des différents protocoles existants dans l'environnement IoT ⇒ **interopérabilité**
  - **Traitement de bord (edge computing)**, pré-traitement de données....
  - **Contrôle et gestion à distance**



F. Rouissi, Technologies de l'IoT, 3<sup>ème</sup> année Génie Info, Septembre 2021

16



## Passerelles IoT (3)

### Types de passerelles dans un système IoT :

- Passerelles pour l'IoT en industrie
- Passerelles pour le traitement de bord « edge computing » ( réseaux de capteurs...)
- Passerelles pour « home automation »
- Routeurs dans les réseaux des opérateurs

### Améliorations du réseau IoT par présence de passerelles :

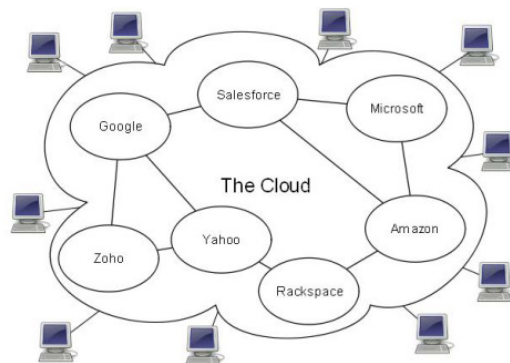
- **Haute évolutivité** ⇒ supporter tout type de données et traitement intelligent pour envoi/réception des centre de données au réseaux (dispositifs finaux)
- **Faible coût des dispositifs** ⇒ dispositifs ne demandent pas un traitement intelligent car c'est le rôle des passerelles
- **Réduction du coût global du réseau** ⇒ moins de communications M2M et taille réduit des réseaux WANs
- **Réduction des risques** ⇒ passerelles isolent les parties du réseau IoT global

## Composants cloud (1)

→ **Cloud Computing (CC)** = technologie permettant de **délocaliser les données et les applications** sur des **infrastructures dématérialisées** accessibles depuis Internet.

Source : Whatis.com

→ mode de structuration et **externalisation** des **composants du système d'information** de l'entreprise qui repose sur les **technologies de virtualisation et automatisation**.



**ENICARTHAGE**  
المدرسة الوطنية للمهندسين بقرطاج  
Ecole Nationale d'Ingénieurs de Carthage

## Composants cloud (2)

→ Le composant Cloud de l'architecture de l'IoT permet à l'utilisateur final d'utiliser des **applications disponibles en continu** et dont la **mise à jour est automatique** sans connaître l'emplacement exact des serveurs (applications & données)

→ A travers des **API** (interfaces de programmation)

→ Peut être mis à disposition **par un fournisseur (éditeur ou prestataire de services hébergés)** ou implémenté **de manière spécifique** pour répondre aux besoins de l'application.

**Panorama du marché des plateformes IoT**  
Source : Fusion Labs

IT Services	Capgemini	AtoS	accenture
Open source	KPN	DeviceHive	OpenIoT
Software & cloud	Microsoft Azure	amazon web services	OVH.com
Telecom	kpn	verizon	swisscom
Hardware & network	ARM	SIERRA WIRELESS	intel

*(Note: The 'Software & cloud' row is circled in the original image.)*

F. Rouissi, Technologies de l'IoT, 3<sup>ème</sup> année Génie Info, Septembre 2021

19

**ENICARTHAGE**  
المدرسة الوطنية للمهندسين بقرطاج  
Ecole Nationale d'Ingénieurs de Carthage

## Composants cloud (3)

→ Trois caractéristiques clés du CC :

- Service avec mise à jour **automatique et en continu**
- **Self-service & paiement à l'usage**, en fonction de la consommation
- **Allocation dynamique de capacité** (adaptation élastique aux pics de charge)

**Topologies du Cloud :**

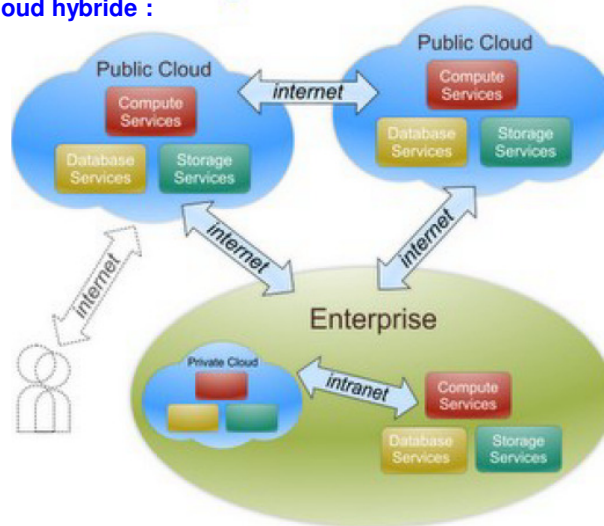
- **Cloud public** ⇒ externe à l'entreprise et partagé via l'accès à Internet, géré par un prestataire externe propriétaire des infrastructures.
- **Cloud privé** ⇒ structure interne à l'entreprise, dont l'accès est complètement dédié et sécurisé.
  - **Cloud virtuellement privé (virtual private)** ⇒ le cloud est externe de l'entreprise mais complètement dédié
  - **Cloud communautaire** ⇒ cas particulier ouvert aux partenaires de l'entreprise : clients, fournisseurs...
- **Cloud hybride** ⇒ conjonction des deux types (privé + public)

F. Rouissi, Technologies de l'IoT, 3<sup>ème</sup> année Génie Info, Septembre 2021

20

## Composants cloud (4)

Exemple de Cloud hybride :

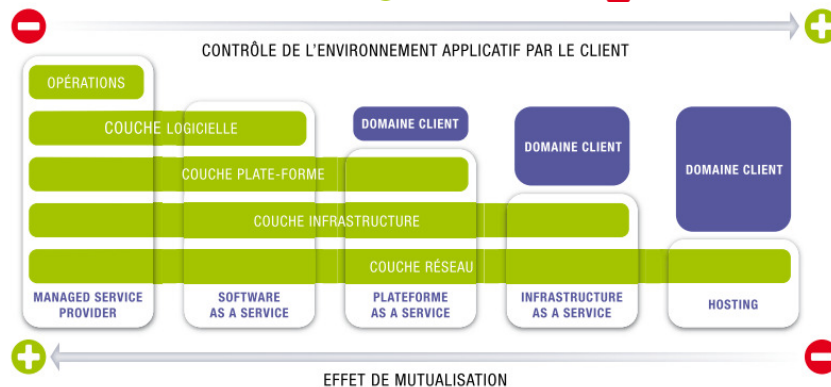


F. Rouissi, Technologies de l'IoT, 3<sup>ème</sup> année Génie Info, Septembre 2021

21

## Types de cloud (1)


- **IAAS** (Infrastructure As A Service)
- **PAAS** (Plateform As A Service)
- **SAAS** (Software As A Service)



Source : Syntec Numerique, "Cloud Computing : nouveaux modèles"

F. Rouissi, Technologies de l'IoT, 3<sup>ème</sup> année Génie Info, Septembre 2021


22



المدرسة الوطنية للمهندسين بقرطاج  
Ecole Nationale d'Ingénieurs de Carthage


## Cloud IAAS

- On dispose des **ressources matérielles** (serveurs virtualisé, stockage, réseau)
- L'accès à la ressource est **complet et sans restriction**
- ~ disposer d'une **infrastructure réelle**
- Louer un serveur dont les ressources peuvent évoluer à la demande.
- L'hébergeur mutualise le réseau (infrastructure + liaisons) et le stockage (SAN, NAS, Backup...)
- Possibilité de virtualisation des serveurs, & administration à distance
- **Les acteurs les plus importants** : **Amazon, Oracle Cloud Infrastructure (OCI), Gandi, OVH...**



23


F. Rouissi, Technologies de l'IoT, 3<sup>ème</sup> année Génie Info, Septembre 2021



المدرسة الوطنية للمهندسين بقرطاج  
Ecole Nationale d'Ingénieurs de Carthage

## Cloud PAAS

- On dispose d'une **plateforme capable** d'accueillir les applications de l'entreprises, ainsi que tous les outils de test et gestion
- La plateforme est spécifique à un langage et une base de donnée
- L'infrastructure hébergée est totalement **transparente**
- Les offres de PAAS incluent la notion de temps CPU, espace de stockage et bande passante ⇒ **payement** en fonction de ce qu'on consomme
- PAAS = IAAS + (logiciel au niveau serveur, base de donnée, composants SOA, runtimes )
- **Exemple de plateformes** : **AWS d'Amazon, Windows Azur, Google Cloud IoT (Google)** ...



24

F. Rouissi, Technologies de l'IoT, 3<sup>ème</sup> année Génie Info, Septembre 2021

**ENICARTHAGE**  
المدرسة الوطنية للمهندسين بقرطاج  
Ecole Nationale d'Ingénieurs de Carthage

## Cloud SAAS

- L'application est mise à la disposition de l'utilisateur final, sans besoin d'aucune connaissance technique
- Tout est chez le fournisseur de service SAAS
- On vend à l'entreprise un **service complet** de traitement de données ⇒ **opérateur de service** (alors que **PAAS** = éditeur de logiciel)
- L'utilisateur ouvre un compte et paie un abonnement, sans soucis de l'espace de stockage, ou puissance de la machine, ou de la mise à jour du logiciel.
- Se basent massivement sur les technologies **Web**, Coté client, seul un navigateur est disponible
- **Exemples de SAAS : Watson IoT Platform (IBM), Pelion (ARM), Live Objects (Orange)**

L'entreprise a le contrôle		
Domaine de responsabilité partagée identifié		
Le fournisseur de service a le contrôle		
IaaS public	PaaS public	SaaS public
Données	Données	Données
Applications	Applications	Applications
Machine virtuelle	Machine virtuelle	Machine virtuelle
Serveur	Serveur	Serveur
Stockage	Stockage	Stockage
Réseau	Réseau	Réseau

*F. Rouissi, Technologies de l'IoT, 3<sup>ème</sup> année Génie Info, Septembre 2021*

25

**ENICARTHAGE**  
المدرسة الوطنية للمهندسين بقرطاج  
Ecole Nationale d'Ingénieurs de Carthage

## Comparaison

Worldwide Public Cloud Services by Segment, 2017H2

Source : IDC 2018

→ Le SAAS est le plus vendu dans le monde et en France (~ 50 % du marché)

→ IAAS et PAAS en pleine croissance

8,5 MILLIARDS € Marché du cloud computing (SaaS, PaaS, IaaS) & des services associés en France pour 2017

+21% TCAM

Source : MARKESS

**Indicateurs clés du marché CC France en 2017-2020 :**

- 36% DU MARCHÉ IAAS 2017 EST DÉDIÉ À DES SOLUTIONS HÉBERGÉES DANS DES CLOUDS PUBLICS
- 11% DU MARCHÉ CLOUD 2017 EST RELATIF À DES SOLUTIONS PAAS & IPAAS
- 49% DU MARCHÉ CLOUD 2017 PORTENT SUR DES SERVICES COMPLÉMENTAIRES (conseil, paramétrage, intégration, formation, support, exploitation...)

Source : news informatique

*F. Rouissi, Technologies de l'IoT, 3<sup>ème</sup> année Génie Info, Septembre 2021*

26

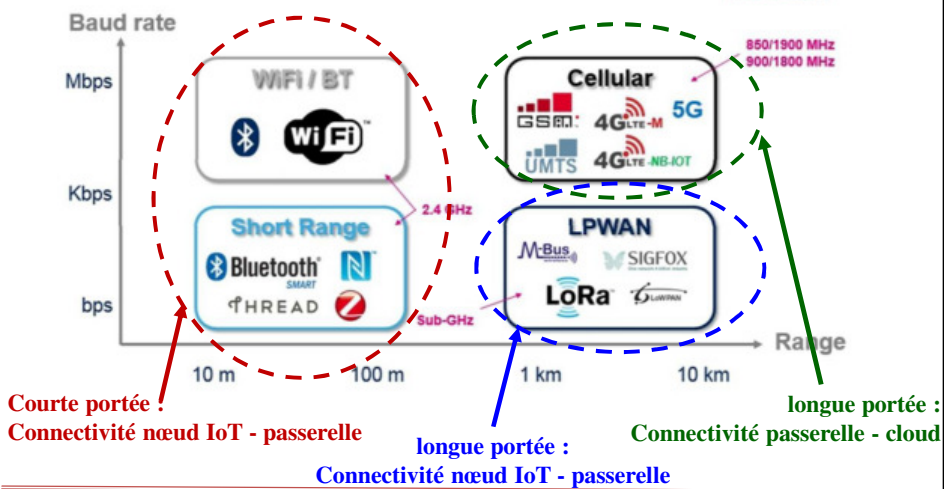
## Chapitre 2


### Connectivité entre les composants IoT

## Rappel des technologies réseaux en IoT

### Technologies de connectivité


(non exhaustive)



 <b>Technologies de communication courte portée</b>				
Technologie	Fréquence	Portée	Débit	Consommation d'énergie
Wifi	De 2,4 GHz à 6 GHz	300 m	Jusqu'à 54 Mbps	gourmande
Wifi Halow	900 MHz	De 60 à 80 m	Jusqu'à 18 Mbps	faible
LIFI (ou VLC)	480 THz & 670 THz	Qcq dizaines de mètres	Jusqu'à 96 Mbps	Moyenne
BLE	2,4 GHz	Qcq dizaines de mètres	1 Mbps	faible
ZigBee	2,4 GHz	100 m	Jusqu'à 250 kbps	faible
Z-Wave	868 MHz & 908 MHz	50 m	jusqu'à 40 Kbps	faible
RFID	900 MHz 13,56 MHz 125 KHz	De qcq cm à qcq mètres	En fonction de la fréquence	faible

*F. Rouissi, Technologies de l'IoT, 3<sup>ème</sup> année Génie Info, Septembre 2021*

29

 <b>Technologies de communication longue portée</b>				
Technologie	Fréquence	Portée	Débit	Consommation d'énergie
SigFox	900 MHz	Jusqu'à 50 km	1 kbps	Très faible
LoRaWAN	variable	Jusqu'à 15 km	≤ 50 kbps	faible
RPMA-Ingenu	2,4 GHz	Jusqu'à 65 km	Jusqu'à 100 kbytes	faible
Réseaux cellulaires classiques	EC-GSM-IoT	850 MHz 900 MHz	Jusqu'à 15 km	~ 10 kbps
	LTE-M	700 MHz- 900 MHz	Jusqu'à 11 km	Jusqu'à 1 Mbps
	NB-IoT	Même que LTE-M	Jusqu'à 11 km	~ 150 kbps

*F. Rouissi, Technologies de l'IoT, 3<sup>ème</sup> année Génie Info, Septembre 2021*

30

**ENICARTHAGE** المدرسة الوطنية للمهندسين بقرطاج  
Ecole Nationale d'Ingénieurs de Carthage

## Connectivité directe du nœud IoT au cloud

- Le dispositif est **directement connecté** à un service cloud sur Internet, via un **fournisseur de service**, pour **échanger les données** et **contrôler le trafic**
- Les données collectées par le capteur sont directement envoyées aux **plateformes de service cloud**
- **Approche simple** qui tire profit des technologies de communications existantes **sur le réseau IP**
- Approche utilisée par les appareils IoT : **Thermostat Nest**, les **TVs Smart de Samsung**

F. Rouissi, Technologies de l'IoT, 3<sup>ème</sup> année Génie Info, Septembre 2021

31

**ENICARTHAGE** المدرسة الوطنية للمهندسين بقرطاج  
Ecole Nationale d'Ingénieurs de Carthage

## Chapitre 2

### Hardware, système embarqué & gestion d'énergie

F. Rouissi, Technologies de l'IoT, 3<sup>ème</sup> année Génie Info, Septembre 2021

32

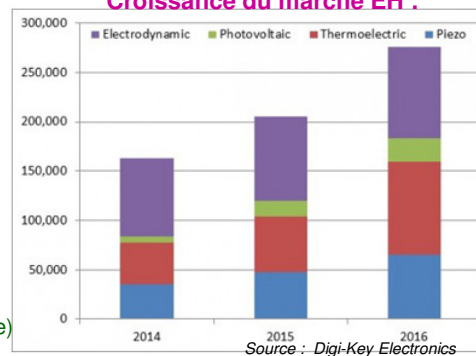


## Principe de gestion de l'énergie pour les objets connectés

- Exigence dans le marché de l'IoT en **dispositifs autonomes** et sans fil  $\Rightarrow$  dispositifs disposants de leur propre source d'alimentation
  - ↳ équipements à **batteries rechargeables** ou sans batterie
  - ↳ équipements communiquant par des **protocoles radio à faible consommation d'énergie** (**LPWAN** : SigFox, LoRaWAN...)
- **Pratiques d'optimisation** de la consommation d'énergie :
  - Maximiser la **réserve énergétique embarquée** (batterie), dans la limite des contraintes physiques
  - Limiter les opérations les plus consommatrices (telles que la transmission de données  $\Rightarrow$  effectuer le traitement en local)
  - Réduire la consommation lors des communications sur le réseau local ou distant
  - Avoir recours aux technologies de **récupération d'énergie** (« **energy harvesting** ») ou « **low tech** »






## Technologies de récupération d'énergie en IoT (1)

- Technologie reposant sur la **conversion** des différents types d'énergie issues de l'environnement ambiant ou des activités humaines en **énergie électrique**, à stocker par la suite pour être utilisée par les dispositifs autonomes
- Conception de piles qui ne se déchargent jamais avec une durée de vie > 20 ans
- Équivalent des **énergies renouvelables** classiques (solaire, chaleur, vent...) mais à échelle **plus petite** ou on exploite les plus infimes sources d'énergie  $\Rightarrow$  dispositifs de **micro-production d'énergie**
- Le « **energy harvesting** » fournit des quantités de courant **très limitées** pour des circuits électroniques à **très basse consommation**
- Les sources d'énergie issues de l'environnement proche :
  - Vibratoires
  - Mécaniques (force, pression)
  - Thermiques ou lumineuses (solaire)



**ENICARTHAGE** المدرسة الوطنية للمهندسين بقرطاج  
Ecole Nationale d'Ingénieurs de Carthage

## Technologies de récupération d'énergie en IoT (2)

	Piézoélectricité	Energie cinétique	Electromagnétisme	Thermoélectricité	Photovoltaïque	Autres
<b>Principe de fonctionnement</b>	Pression ou vibration sur un matériau	Récupération de l'énergie issue des mouvements d'un corps	Collecte de l'énergie véhiculée par les ondes électromagnétiques	Récupération de la chaleur fatale	Film placé sur une surface exposée aux rayons du soleil	Energie chimique, éolien, combinaison de plusieurs sources ?
<b>Exemples d'applications</b>	Trottoir ou semelle piézoélectrique, interrupteur autonome, capteur sur des rails, allume gaz mécanique	Électronique vestimentaire	Dynamo, chargeurs à manivelle	Chaleur des ordinateurs, de l'eau de douche, du corps (montres)	Calculatrice ou réveil solaire, film sur la paroi d'une fenêtre	À déterminer
<b>Illustrations</b>						
	Trottoir piézoélectrique	Le socket : ballon de football producteur d'électricité	Dynamo pour vélo	ecoRC : récupérateur de chaleur pour la douche	Film photovoltaïque	

Source : Siapartners energies & environment

F. Rouissi, Technologies de l'IoT, 3<sup>ème</sup> année Génie Info, Septembre 2021


35

**ENICARTHAGE** المدرسة الوطنية للمهندسين بقرطاج  
Ecole Nationale d'Ingénieurs de Carthage

## Technologies de récupération d'énergie en IoT (3)

→ **Solutions technologiques de récupération d'énergie pour les capteurs autonomes :**

Wireless Power



RF Energy Harvesting

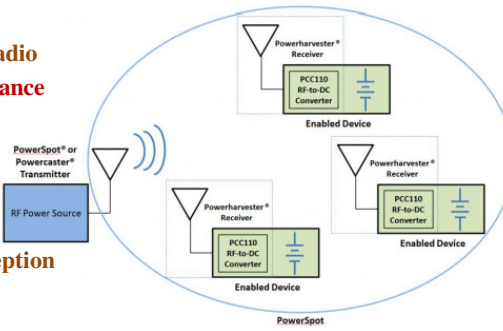
+ Autres solutions de gestion d'énergie  
(« Power Management »)....

**Diffusion intentionnelle de fréquences radio pour produire de l'énergie sans fil à distance**

⇒ Transmission de l'énergie

⇒ Travaux dans la conception de:

- Niveau de puissance
- Fréquence radio
- Gains d'antenne d'émission/réception
- Distance
- ...



F. Rouissi, Technologies de l'IoT, 3<sup>ème</sup> année Génie Info, Septembre 2021

36

**Technologies de récupération d'énergie en IoT (4)**

Wireless Power      RF Energy Harvesting

Utilisation de la **technologie RFID** pour récupérer l'énergie RF

Up to 10 meters

→ **UHF-RFID Reader** qui permet en plus de l'identification de **transmettre l'énergie RF** qui sera converti en énergie électrique à utiliser par le capteur

→ Capteur doté d'un **composant de conversion d'énergie** (« **powercast harvester** »)

Powercast provides >10X the power vs. traditional RFID

F. Rouissi, Technologies de l'IoT, 3<sup>ème</sup> année Génie Info, Septembre 2021

37

**Hardware, systèmes embarqués (1)**

→ Un objet connecté est typiquement composé de **plusieurs cartes électroniques** sur lesquelles sont montées des composants permettant la **connectivité**, le **calcul**, la **numérisation des données**...

↳ usage des technologies de **l'informatique embarquée** permettant l'optimisation de l'espace et du coût des composants

→ **Système embarqué** ⇒ **logiciel et matériel** sont conçus en même temps

→ Les objets IoT doivent être des systèmes embarqués connectés sur le réseau !!

**Architecture d'un système embarqué IoT :**

Equipement intelligent de traitement

capteurs → ADC → microcontrôleur → DAC → actionneurs

reçoit des infos du monde extérieur

Convertisseur analogique/numérique

Circuit intégré

Circuit logique programmable

informe le monde extérieur

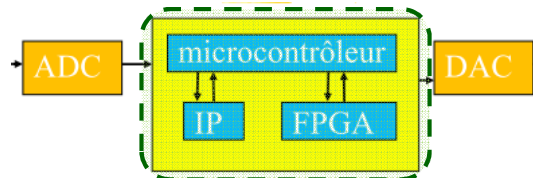
Convertisseur numérique/ analogique

Source : coursera Ian Harris – Introduction to the Internet of Things & embedded systems

F. Rouissi, Technologies de l'IoT, 3<sup>ème</sup> année Génie Info, Septembre 2021

38

## Hardware, systèmes embarqués (2)



- **Microcontrôleur** ≡ processeur + mémoire ROM + mémoire RAM + interfaces E/S... à faible consommation électrique et opérations ciblées
- **IP (Intellectual Property Core)** ≡ circuit intégré (≈ processeur) **dédié** à une fonctionnalité
  - **Exemple** : contrôleur réseau : ethernet, bus CAN dans les automobiles, codec audio...
- **FPGA (Field Programmable Gate Array)** ≡ circuit intégré logique qui peut être **reprogrammé** après sa fabrication
  - ≈ **PROM** pouvant charger de nombreuses applications
  - En fait **reconfigurable** plutôt que programmable

## Hardware, systèmes embarqués (3)

- Deux approches majeures des technologies des systèmes embarqués :

### 1. Systèmes construits autour d'un OS Embarqué :

- **Avantages** : ouverts, puissants, supporte des langages de programmation multiples
- **Inconvénients** : parfois complexes à mettre en œuvre, prise en main longue, coût relativement élevé, interfaçage plus complexe
- **Exemples** : Raspberry PI, Beaglebone...

### 2. Systèmes dédiés compacts à logiciel propriétaire :

- **Avantages** : très réactifs, faible coût, fonctionnement plus robuste (pas de couches logicielles), interfaçage aisé, prise en main rapide
- **Inconvénients** : moins puissants, langages de programmation plus limités, moins flexible sur le plan logiciel
- **Exemples** : Arduino, Genuino, Intel Galileo, ESP8266...



Arduino + shield



Raspberry Pi



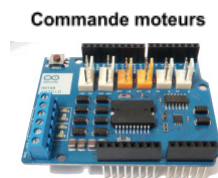
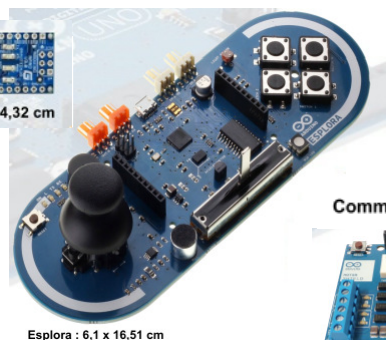
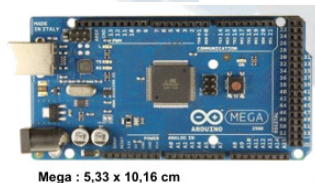
BeagleBone

## Microcontrôleur Arduino (1)

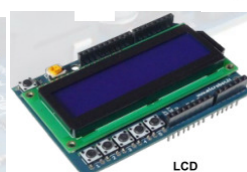
- **Carte Arduino** ≡ carte équipée du micro-contrôleur, qui permet, à partir d'évènements détectés par des capteurs, de programmer et commander des actionneurs
  - ↳ Interface programmable
- **Extension du projet Arduino** ≡ plateforme de développement et de prototypage (développement matériel & logiciel) **Open source**, composé de :
  - **Carte Arduino** ⇒ stocker un programme et le faire fonctionner
  - **Shields (cartes d'extension)** ⇒ fonctions diverses : relais, commande de moteurs, connectivité ethernet ou wifi..., afficheur LCD...
  - **IDE** ⇒ environnement de développement intégré **multi OS** pour l'édition et la compilation des programmes et leurs transferts dans la carte via le port USB
- Le microcontrôleur Arduino doté d'un **processeur 1 core à 16 MHz** généralement, et une **faible RAM** ⇒ faible consommation d'énergie et faible prix
- Carte Arduino la plus utilisée ≡ **Arduino Uno**

## Microcontrôleur Arduino (2)

### Divers Cartes Arduino :



### Divers Shields Arduino :



**Microcontrôleur Arduino (3)**

**Carte Arduino Uno :**

**Connexion USB (Universal Serial Bus) :**

- Alimenter la carte (5 Volts)
- Charger les programmes dans la carte

**Prise jack :**

- Brancher une alimentation (pile, batterie, adaptateur secteur)

**Entrées/ sorties numériques :**

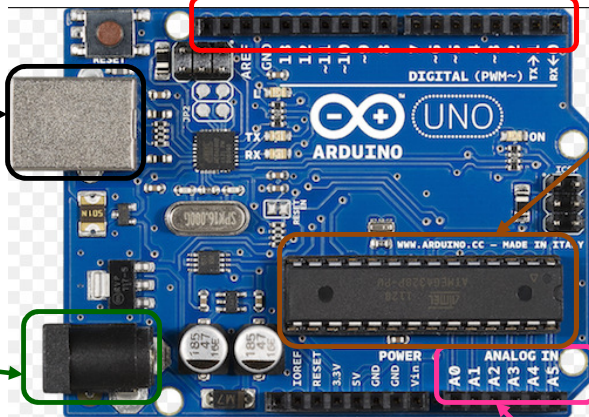
- Brancher les actionneurs et les détecteurs

**Microcontrôleur :**

- Stocker le programme et l'exécuter

**Entrées analogiques :**

- Brancher capteurs & détecteurs analogiques



The diagram shows an Arduino Uno board with several components highlighted by colored boxes and arrows. A red box highlights the digital pins (0-13) with labels for 'Entrées/ sorties numériques' and 'Brancher les actionneurs et les détecteurs'. A black box highlights the USB Type-B port with labels for 'Connexion USB (Universal Serial Bus)' and 'Alimenter la carte (5 Volts)' and 'Charger les programmes dans la carte'. A green box highlights the DC power jack with a label for 'Prise jack' and 'Brancher une alimentation (pile, batterie, adaptateur secteur)'. An orange box highlights the ATmega328P microcontroller chip with a label for 'Microcontrôleur' and 'Stocker le programme et l'exécuter'. A pink box highlights the analog pins (A0-A5) with a label for 'Entrées analogiques' and 'Brancher capteurs & détecteurs analogiques'.

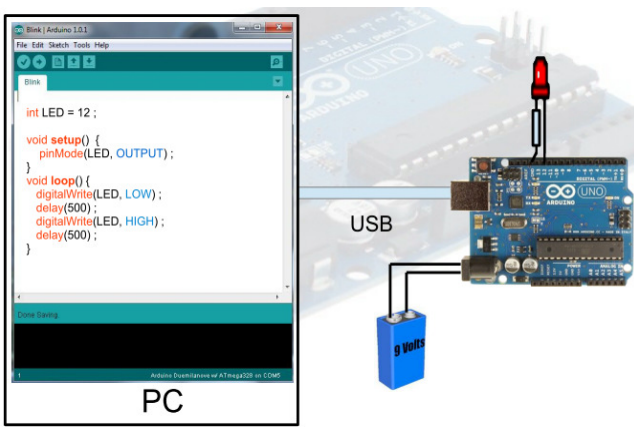
F. Rouissi, Technologies de l'IoT, 3<sup>ème</sup> année Génie Info, Septembre 2021

43

**Microcontrôleur Arduino (4)**

**IDE Arduino :**

- Environnement de développement intégré fonctionnant sur **plusieurs systèmes d'exploitation** (Windows, Gnu/Linux, Mac OS)
- Permet d'**éditer** le programme sur un ordinateur et de le **transférer** à la carte via le port USB
- Programmation très **proche au langage C**, avec **plusieurs bibliothèques logicielles disponibles**



The image shows the Arduino IDE interface on a PC screen. The code for a 'Blink' sketch is visible, showing the setup and loop functions. Below the code, there is a 'Compile' button and a 'Download' button. To the right of the IDE, there is a photograph of an Arduino Uno board connected to a PC via a USB cable. A 9V battery is also shown connected to the board's power pins.

PC

F. Rouissi, Technologies de l'IoT, 3<sup>ème</sup> année Génie Info, Septembre 2021

44

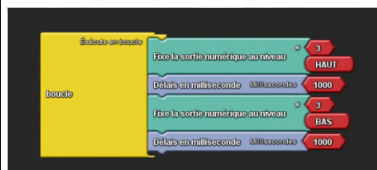


## Microcontrôleur Arduino (5)

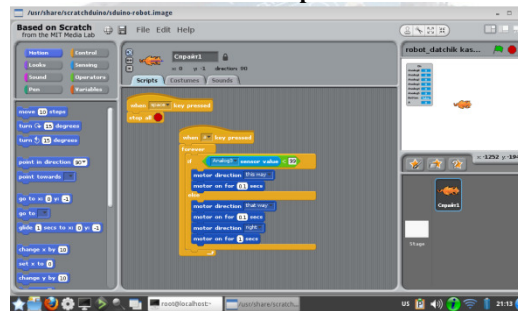
### IDE Arduino :

- Il existe d'autres outils facilitant la programmation :
  - ArduBlock** : outil qui se greffe au logiciel Arduino et permet la programmation en **mode graphique**
  - Scratch pour Arduino** : piloter Arduino à partir du code scratch, aussi en mode graphique

#### ArduBlock



#### Scratch pour Arduino



F. Rouissi, Technologies de l'IoT, 3<sup>ème</sup> année Génie Info, Septembre 2021

45

## Raspberry Pi (1)

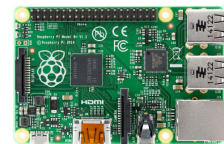
- **Raspberry Pi** ≡ nano-ordinateur ou ordinateur **compact** à taille réduite et **consommation faible** (économique)
  - Composé uniquement d'une petite carte mère, alimentée par un **câble USB**
  - De **nombreux ports USB** peuvent être utilisés pour connecter divers appareils, notamment des claviers, des souris, des disques externes et des caméras.
  - Plusieurs modèles : **A, A+, B, B+, B2, B3...**



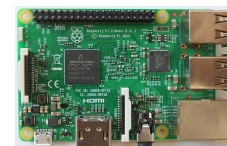
Modèle A



Modèle B




Modèle B+



Modèle B3 (Pi 3)

F. Rouissi, Technologies de l'IoT, 3<sup>ème</sup> année Génie Info, Septembre 2021

46



المدرسة الوطنية للمهندسين بقرطاج  
Ecole Nationale d'Ingénieurs de Carthage


# Raspberry Pi (2)

Modèle	A	A+	B1	B2 (Pi 2)	B3 (Pi 3)
CPU	700 MHz ARM1176JZF-S core (ARM11)			900 MHz quadricœur ARM Cortex-A7	1,2 GHz quadricœur ARM Cortex-A53
RAM	256 Mo		512 Mo	1 Go	
USB	1 port		2 ports	4 ports	
Audio/vidéo	Jack et composite/ HDMI et composite				
Connectivité réseau	0	10/100 Ethernet	0	10/100 Ethernet	10/100 Ethernet, Wifi 802.11n, Bluetooth 4.1
Stockage	SD	Micro SD	SD	Micro SD	

Source : [http://fr.Wikipedia.org/wiki/RaspBerry\\_Pi](http://fr.Wikipedia.org/wiki/RaspBerry_Pi)

F. Rouissi, Technologies de l'IoT, 3<sup>ème</sup> année Génie Info, Septembre 2021

47



المدرسة الوطنية للمهندسين بقرطاج  
Ecole Nationale d'Ingénieurs de Carthage

# Raspberry Pi (3)

Modèle	A	A+	B1	B2 (Pi 2)	B3 (Pi 3)
Entrées/sorties	GPIO 26 pts	GPIO 40 pts	GPIO 26 pts	GPIO 40 pts	
Poids	45 g	23 g	45 g		
Consommation	1,5 W	1 W	3,5 W	4 W	
OS	Raspbian OS, Debian GNU/Linux, Fedora, Arch Linux ARM, RISC OS, FreeBSD, Plan 9, Kali Linux			... + Snappy Ubuntu Core, SolydX RPI, <b>Windows 10 IoT</b>	

Source : [http://fr.Wikipedia.org/wiki/RaspBerry\\_Pi](http://fr.Wikipedia.org/wiki/RaspBerry_Pi)

→ L'OS officiel et recommandé à utiliser est **Raspbian** = **distribution GNU/Linux** (Linux-Dubian) **optimisée** pour le matériel du Raspberry Pi, embarquant **l'environnement de bureau LXDE** et **le navigateur web Midori**

→ Logiciel spécial lecteur multimédia léger pour le Raspberry Pi ⇒ **RaspyPlayer**

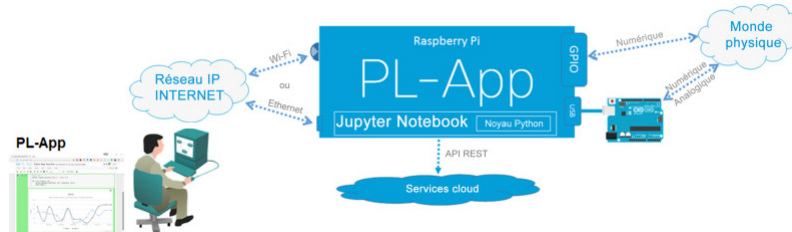
F. Rouissi, Technologies de l'IoT, 3<sup>ème</sup> année Génie Info, Septembre 2021

48



## Programmer sur Raspberry Pi (1)

- L'accès à RapBerry Pi est :
  - **En local** (utiliser un clavier et un écran branchés + OS dans la carte SD)
  - **À distance** : à l'aide de PL-App (plate-forme logicielle exécutée sur un circuit Raspberry Pi qui dispose d'une interface web basée sur un concept de **Notebooks**)



- Raspberry Pi prend en charge de nombreux langages de programmation :
  - **Langages visuels** : Blockly, Scratch
  - **Python, C, C++**
  - **Java, javascript, html5 ...**

## Programmer sur Raspberry Pi (2)

### Blockly :

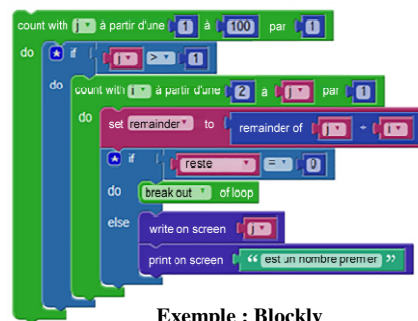
- permet de créer un programme en utilisant **des blocs de couleur**.
- Les blocs sont par la suite reliés les uns aux autres par glisser-déposer

### Python :


- **Le langage officiel du RaspBerry Pi**, installé par défaut dans Raspbian
- **Très puissant et facile** à apprendre

```
script.py
1 # This program adds two numbers
2
3 num1 = 1.5
4 num2 = 6.3
5
6 # Add two numbers
7 sum = float(num1) + float(num2)
8
9 # Display the sum
10 print('The sum of {0} and {1} is {2}'.format(num1, num2, sum))
```

Exemple de script Python



Exemple : Blockly



المدرسة الوطنية للمهندسين بقرطاج  
Ecole Nationale d'Ingénieurs de Carthage

## Comparaison Arduino-RaspBerry

- RaspBerry Pi est un outil puissant, adapté aux traitement complexe, mais :
  - N'inclut pas de broches **GPIO analogiques** (Arduino si !!!)
  - Ne fonctionne pas **en temps réel**
  - Pour certaines applications, il est possible que ses besoins en alimentation et sa taille soient **importantes !!!**

⇒ Possibilité d'utiliser **Arduino contrôlé par un RaspBerry Pi**

NOM DE LA CARTE	ARDUINO UNO	RASPBERRY PI (MODEL B)
Naissance	2005 (fabrication en Italie par Smart Projects)	2008 (accord de fabrication avec RS Components et Farnell/Element 14 en 2011)
Prix	30 \$	Moins de 40 \$
Taille	45,43 x 32,34 mm	85,60 x 53,98 mm
Processeur	ATmega328 8 bits d'Atmel à 16 MHz	BCM2835 de Broadcom basé sur un ARM11 à 700 MHz GPU intégrée (Video Core 4 de Broadcom)
Mémoires	2 Ko Ram, 1 Ko Eeprom	512 Mo Sdram
Mémoire Flash	32 Ko	Sur carte SD
Consommation	42 mA (0,5W)	700 mA (3,5W)
Ethernet	Non	10/100 Ethernet
USB	Non	2 USB 2.0
Sorties vidéo	Non	Composite et HDMI
Développement	Langage de programmation Arduino	Langages Scratch, <b>Python</b>

*F. Rouissi, Technologies de l'IoT, 3<sup>ème</sup> année Génie Info, Septembre 2021*

**51**



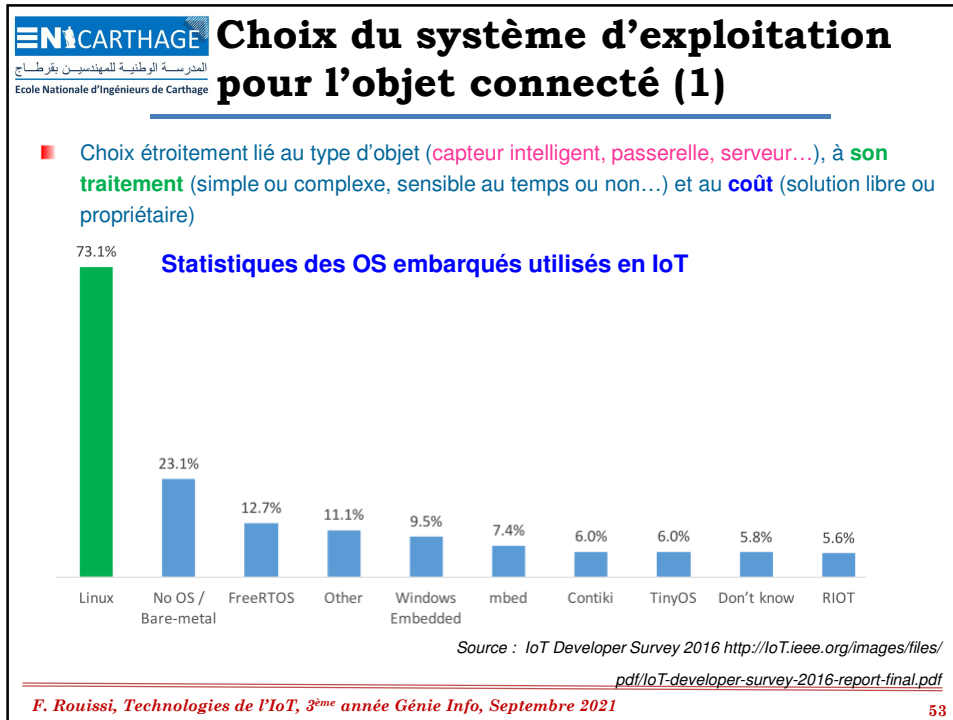
المدرسة الوطنية للمهندسين بقرطاج  
Ecole Nationale d'Ingénieurs de Carthage

## Chapitre 2

# Systèmes d'exploitation pour les dispositifs à ressources limités

*F. Rouissi, Technologies de l'IoT, 3<sup>ème</sup> année Génie Info, Septembre 2021*

**52**



**ENICARTHAGE** المدرسة الوطنية للمهندسين بقرطاج  
Ecole Nationale d'Ingénieurs de Carthage

## Choix du système d'exploitation pour l'objet connecté (2)

name	architecture	programming model	supported MCU families or vendors	programming languages	license
Contiki	monolithic	event-driven, Protothreads	AVR, MSP430, ARM7, ARM Cortex-M, PIC32, 6502	C <sup>*</sup>	BSD
RIOT	microkernel RTOS	multi-threading	AVR, MSP430, ARM7, ARM Cortex-M, x86	C, C++	LGPLv2
FreeRTOS	microkernel RTOS	multi-threading	AVR, MSP430, ARM, x86, 8052, Renesas <sup>*</sup>	C	modified GPL <sup>*</sup>
TinyOS	monolithic	event-driven	AVR, MSP430, px27ax	nesC	BSD
OpenWSN	monolithic	event-driven	MSP430, ARM Cortex-M	C	BSD
nuttX	monolithic or microkernel	multi-threading	AVR, MSP430, ARM7, ARM9, ARM Cortex-M, MIPS32, x86, 8052, Renesas	C	BSD
eCos	monolithic RTOS	multi-threading	ARM, IA-32, Motorola, MIPS ...	C	eCos License <sup>*</sup>
uClinux	monolithic	multi-threading	Motorola, ARM7, ARM Cortex-M, Atari	C	GPLv2
ChibiOS/RT	microkernel	multi-threading	AVR, MSP430, ARM Cortex-M	C	Triple License <sup>*</sup>
CoOS	microkernel RTOS	multi-threading	ARM Cortex-M	C	BSD
nanoRK	monolithic (resource kernel)	multi-threading	AVR, MSP430,	C	Dual License
Nut/OS	monolithic	multi-threading	AVR, ARM	C	BSD

Source :O.Hahm, E.Baccelli, H. Peterson, N. Tsiftes, « Operating Systems for Low-End Devices in the Internet of Things: a Survey », <https://hal.inria.fr/hal-01245551/file/IoT-OS-survey.pdf>

F. Rouissi, Technologies de l'IoT, 3<sup>ème</sup> année Génie Info, Septembre 2021

54