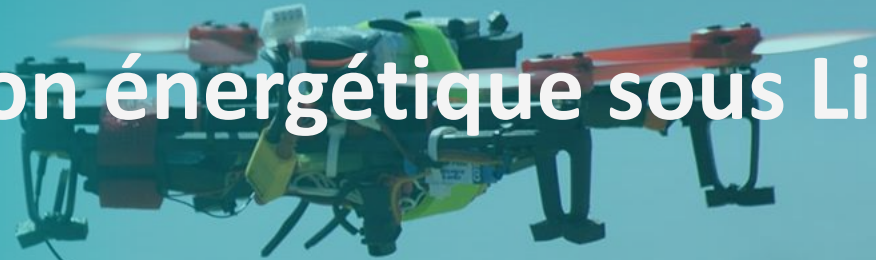


# Gestion de la consommation énergétique sous Linux

Cours 5 – Camélia Slimani – Jalil Boukhobza  
ENSTA-Bretagne – Lab-STICC

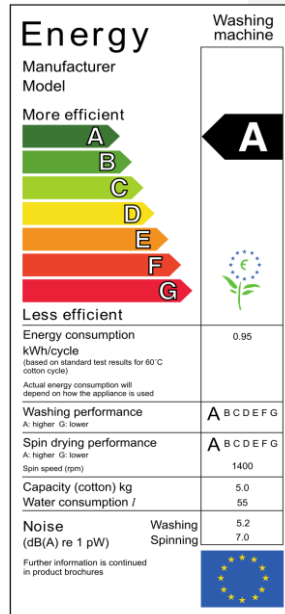


# Plan

- / Introduction à la problématique énergétique
- / Décomposition de la consommation énergétique sur un système embarqué
- / Energie dissipée sur les ressources de calcul d'un système embarqué
- / Mécanisme de gestion de la consommation énergétique sous Linux
- / Conclusion
- / Références

# Introduction à la problématique énergétique

- / Les équipements embarqués sont dans la majorité des cas **alimentés par batteries** : smartphones, capteurs, IoT.
- / L'usage de ce type d'équipements est en augmentation rapide. D'ici 2025, le nombre d'équipements IoT connectés devrait atteindre **2,5 milliards** [1].
- / L'**efficacité énergétique** désigne le rapport entre l'**énergie utile** (effectivement récupérée pour le travail utile), et l'**énergie fournie** (qui est dépensée) par un équipement.
- / Une faible efficacité énergétique d'un système embarqué affecte :
  - / Le temps de **disponibilité** de l'équipement ;
  - / Le **vieillessement** des équipements électroniques, à cause de leur **surchauffe** [2].
- / Dans ce cours :
  - / Composantes de la consommation énergétique sur système embarqué ;
  - / Les mécanismes employés par Linux pour réduire la consommation/augmenter l'efficacité énergétique.



# Introduction à la problématique énergétique

## Notion de consommation

### Puissance

Quantité d'énergie consommée par unité de temps, exprimée en Watt.

Minimisation imposée par les contraintes thermiques des circuits.

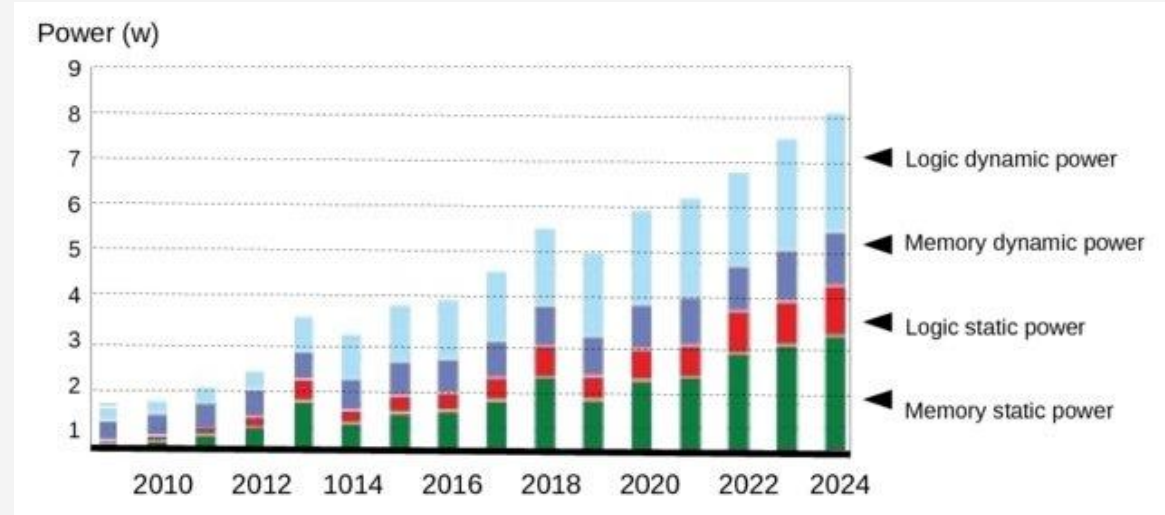
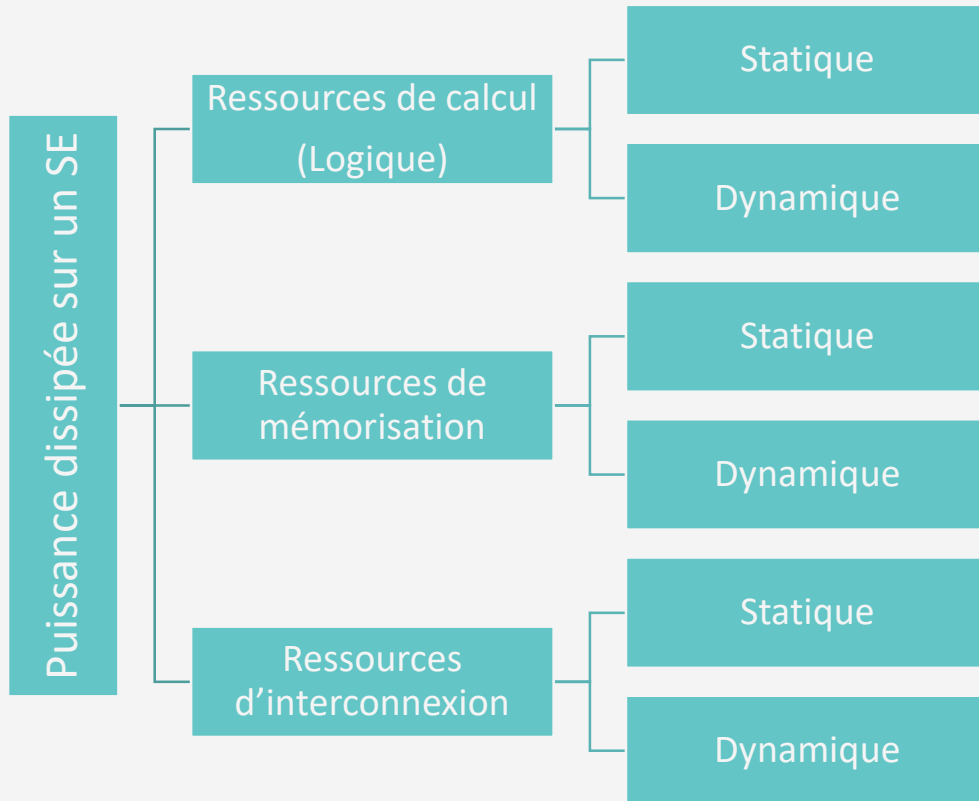
### Energie dissipée

Travail nécessaire pour le fonctionnement électrique d'un circuit pendant un temps considéré, exprimé en Joule.

Minimisation imposée par la technologie de batterie.

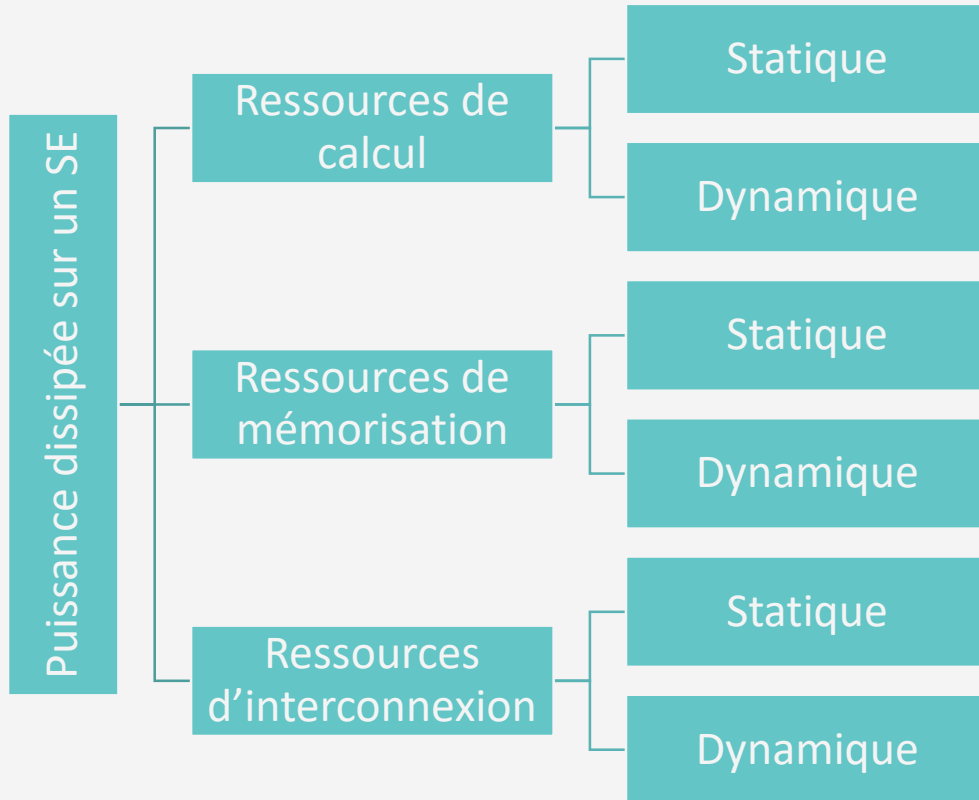
Ces grandeurs **peuvent être antagonistes**, ce qui fait de l'optimisation énergétique un **problème complexe** [3].

# Décomposition de la puissance sur un système embarqué



Tendances de consommation de puissance sur les systèmes embarqués [4]

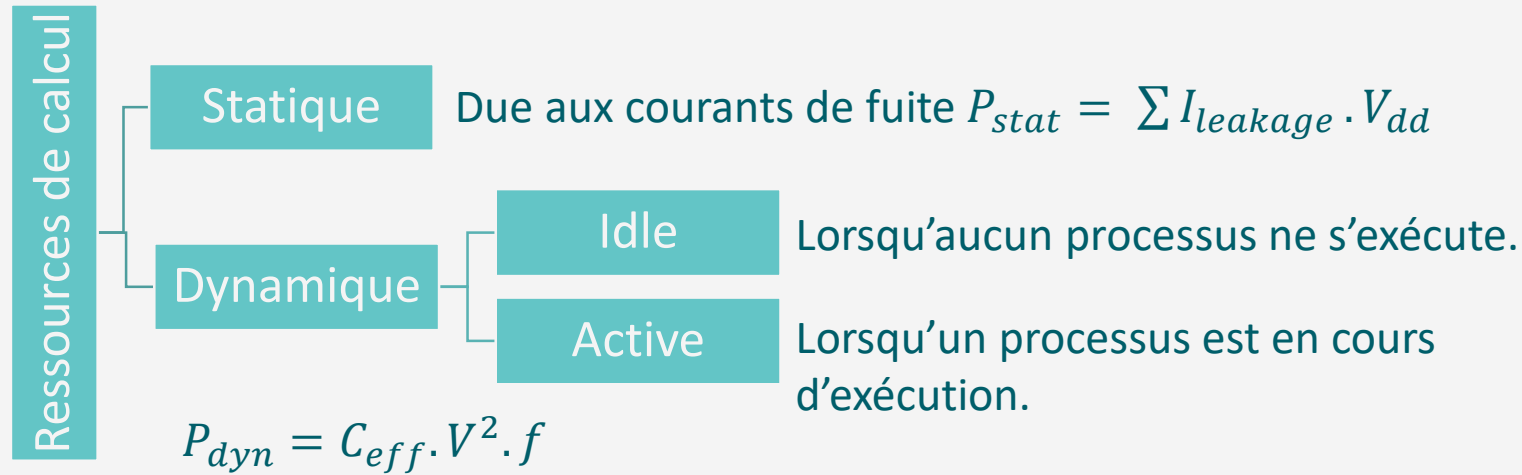
# Décomposition de la puissance sur un système embarqué



Puissance statique : consommation du circuit due aux courants de fuite.

Puissance dynamique : consommation due à l'activité du circuit.

# Décomposition de la puissance sur un système embarqué



Symbole	Signification
$I_{leakage}$	Courant de fuite
$V_{dd}$	Tension d'alimentation
$C_{eff}$	Capacitance active
$V$	Tension d'alimentation du processeur
$f$	Fréquence d'horloge di processeur

**Approximation admise :** la tension est proportionnelle à la fréquence.  $V = \alpha \cdot f \Rightarrow P_{dyn} = C_{eff} \cdot \alpha^2 \cdot f^3$

# Energie dissipée sur les ressources de calcul d'un système embarqué

Energie dissipée :

$$E = P \cdot t$$

$$E = E_{stat} + E_{dyn}$$

$$E = P_{stat} \cdot t + P_{dyn} \cdot t$$

Il est possible d'agir sur la fréquence pour réduire la consommation énergétique

Temps :

$$t = \frac{N}{f}$$

N étant le nombre de cycles d'horloge nécessaire à l'application (travail)

Energie dynamique :

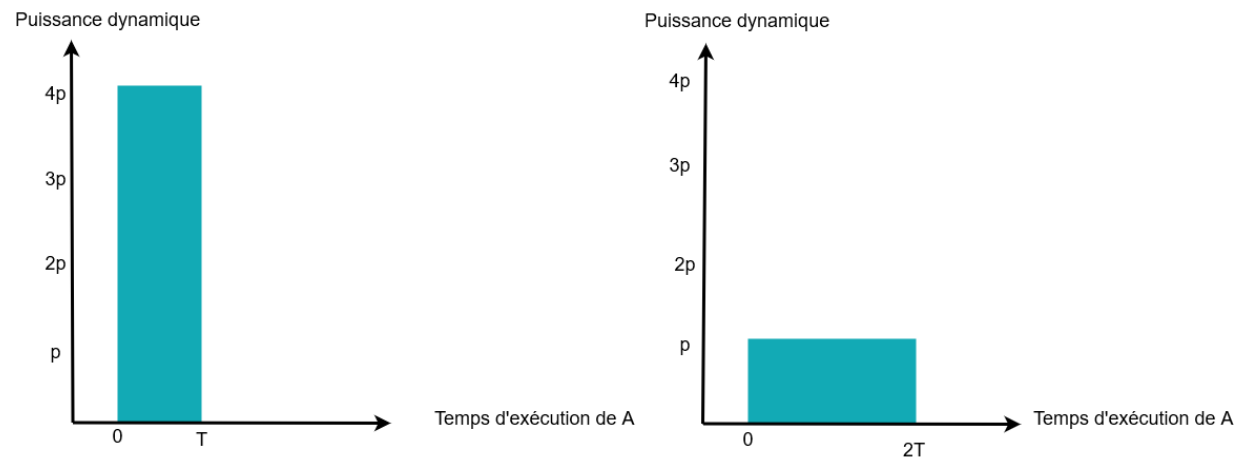
$$E_{dyn} = N \cdot C_{eff} \cdot \alpha^2 \cdot f^2$$



# La modulation de fréquence et de tension

/ La modulation de fréquence et de voltage processeur (*Dynamic Voltage and Frequency Scaling*) - DVFS : une technique qui consiste à **adapter de manière dynamique** la **fréquence** et le **tension** du processeur.

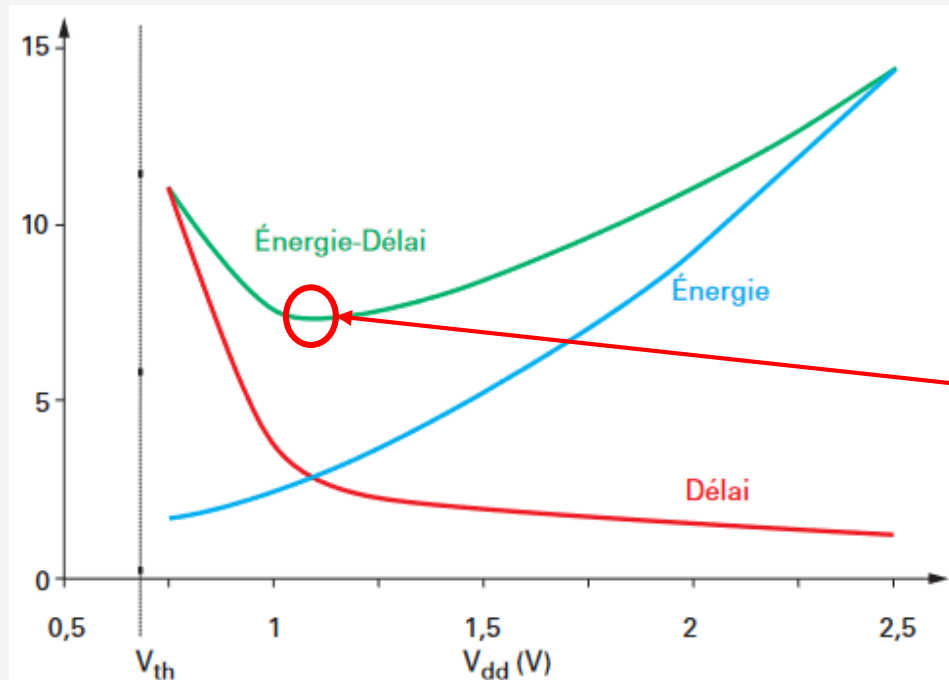
/ Exemple : Pour une tâche A, si on divise la fréquence du processeur par 2 :  $f = f_{max}/2$



+ La consommation énergétique est réduite de **75%**.

- Le temps d'exécution **double**.

# La modulation optimale de la fréquence et du voltage



Point de fonctionnement optimal en terme de compromis **énergie-délai**.

Effet de la variation de fréquence/voltage sur le délai, l'énergie et le compromis énergie-délai [3].

# Nature des tâches et opportunités de modulation de fréquence

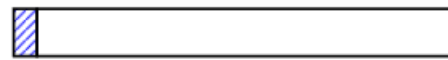
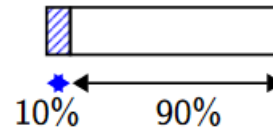
## Nature des tâches

Intensive en  
Calculs

Intensive en accès  
à la mémoire

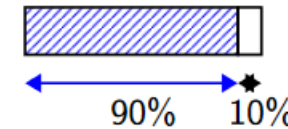
Intensive en  
entrées sorties

Cas d'une tâche  
intensive en calculs



+90%

Cas d'une tâche  
intensive en accès  
mémoire



+10%

Temps d'accès à la mémoire

Temps de calcul



Il est plus intéressant de réduire la fréquence/tension dans le cas d'une tâche intensive en accès mémoire qu'une tâche intensive en calculs  $\Rightarrow$  Modulation opportuniste

# Les mécanismes DVFS des systèmes d'exploitation

- / La majorité des équipements embarqués permettent la modulation de fréquence des cœurs du/des processeurs.
- / L'*Advanced Configure and Power Interface* (ACPI) est un standard qui permet au système d'exploitation de :
  - / Découvrir les *états de puissance* du processeur (*CPU Power states, C-state*). Les C-states optimisent la puissance durant les phases Inactives du processeur.
  - / Gérer les *états de performance* (*performance states, P-state*) du processeur. Les P-states optimise la puissance durant la phase active du processeur.
- / Ce standard a été originellement proposé par *Intel, Toshiba* et *Microsoft*. La terminologie d'ACPI a été reprise et adaptée pour les architectures ARM.

# C-states sur une Architecture ARM

## Run

- La hiérarchie du cœur (cœur + caches non partagés) est entrain d'exécuter un processus et est opérationnelle.

## Idle

- La hiérarchie du cœur n'est pas entrain d'exécuter un processus.
- Le cœur peut être réveillé grâce à une interruption système

## Sleep

- La hiérarchie du cœur n'est pas entrain d'exécuter un processus.
- Le contexte est perdu et doit être restauré
- Le cœur peut être réveillé grâce à une interruption système.

## Off

- La hiérarchie du cœur n'est pas entrain d'exécuter un processus.
- Le contexte est perdu et doit être restauré.
- Le réveil n'est possible que par une commande système explicite ou par un reset matériel.

# Les p-states ou points de fonctionnement des cœurs de processeurs

/ Les points de fonctionnement du processeur (*operating points – OPP*) sont l'ensemble des **tuples disponibilité, fréquence, tension** sur lesquels le processeur est capable de fonctionner.

```
SoC
|- device 1
|   |- opp 1 (availability, freq, voltage)
|   |- opp 2 ..
|   ...
|   ^- opp n ..
|- device 2
|   ...
^ - device m
```

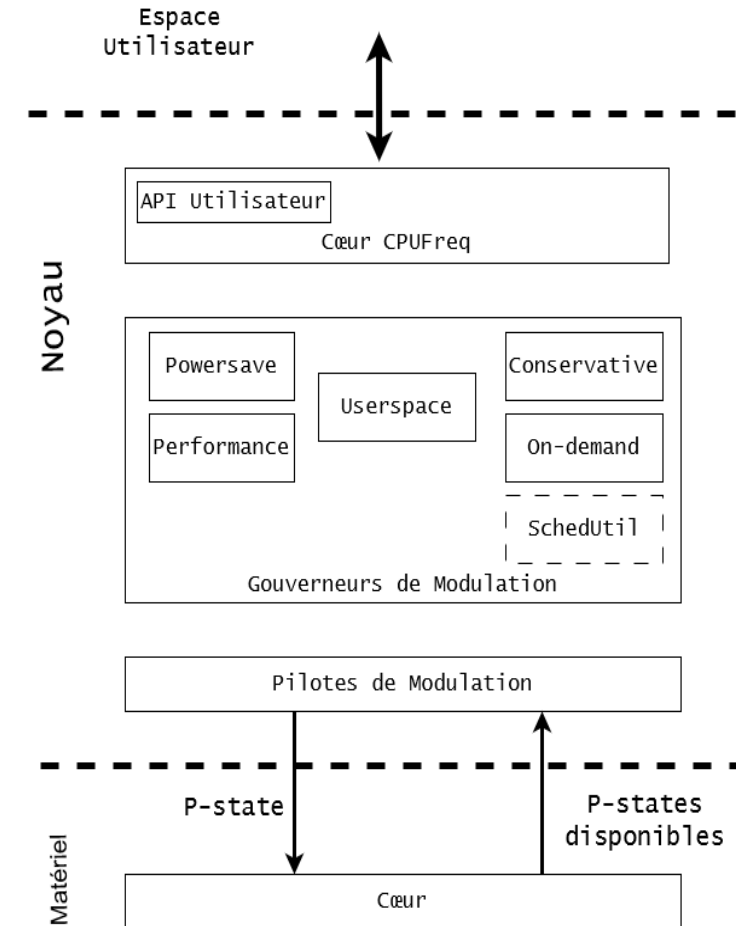
Apparition des OPP dans la structure *Device Tree* (fichier de description des composants matériels d'un système)

# Le mécanisme DVFS sous Linux

- / Le noyau Linux prend en charge la modulation de fréquence à travers le sous-système CPUFreq.
- / Sur la plupart des systèmes interactifs, la charge du processeur varie en permanence en états d'intense activité et périodes de repos [5].

# Le sous-système CPUFreq

- / Gouverneur de modulation : heuristique/algorithmme qui prend en charge la modulation de fréquence/tension.
- / Gouverneurs statiques :
  - / *Powersave* : utilise toujours la fréquence la plus basse pour limiter la consommation ;
  - / *Performance* : utilise toujours la fréquence la plus élevée pour minimiser le temps de traitement ;
- / Gouverneurs dynamiques :
  - / *Ondemand* : fait varier la fréquence **en fonction de la charge du processeur** ;
  - / *Conservative* : même principe que *Ondemand*, mais tente de minimiser les changements de fréquences ;
  - / *SchedUtil* : gouverneur dépendant de l'ordonnanceur. La fréquence choisie pour un processus donné dépend de sa classe d'ordonnancement. Par exemple, si le gouverneur est invoqué pour un processus temps réel, la fréquence est fixée à la valeur maximale.
- / Gouverneur *userspace* : le paramétrage de la fréquence s'effectue depuis l'espace utilisateur par un processus *root*.





# Systeme de fichiers CPUFreq

- / A l'initialisation du noyau, le Cœur CPUFreq crée un répertoire sysfs (kobject) appelé cpufreq sous : `/sys/devices/system/cpu/cpuX/`. X correspond au numéro du cœur.
- / Ce répertoire contient, entre autres, les fichiers suivants :
  - / `scaling_available_governors` : il contient la liste des gouverneurs utilisables.
  - / `scaling_governor` : il contient le gouverneur en cours d'utilisation.
  - / `scaling_available_frequencies` : il contient la liste des fréquences utilisable (points d'opérations).
  - / `scaling_cur_freq` : il contient la fréquence courante.
  - / `scaling_max_freq` : il contient la fréquence maximale utilisable.
  - / `scaling_min_freq` : il contient la fréquence minimale utilisable.

# Système de fichiers CPUFreq

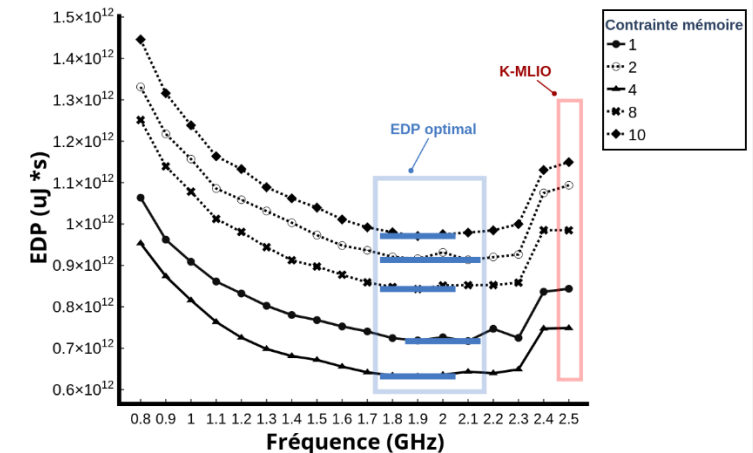
- / `ondemand/up_threshold` : seuil de différence d'utilisation CPU pour augmenter la fréquence.
- / `ondemand/sampling_down_factor`: il définit la fréquence à laquelle le noyau vérifie l'utilisation du CPU pour abaisser la fréquence.
- / `Conservative/freq_step` : il définit le pas d'augmentation de la fréquence.
- / `Conservative/down_threshold` : il contient le seuil de différence d'utilisation CPU à partir duquel la fréquence peut être utilisée.
- / `Conservative/sampling_down_factor` : il définit la fréquence à laquelle le noyau vérifie l'utilisation du CPU pour abaisser la fréquence.

# Que doit prendre en compte un ingénieur système embarqué ?

- / Choisir le matériel adapté au cas d'étude ;
- / Bien choisir le gouverneur selon les caractéristiques du système manipulée, et des tâches qui doivent être exécutées ;
- / Bien choisir le langage de programmation selon le besoin et les contraintes existantes sur la plateforme cible;
- / Limiter l'utilisation de bibliothèques ;
- / Veiller à séparer, autant que possible, les traitements calculatoires des accès mémoire, entrées/sorties.

# Exemple de travail effectué au sein du laboratoire sur l'amélioration de l'efficacité énergétique

- / EDP est une métrique qui traduit le compromis Energie-Temps. Elle est calculée comme le produit de l'énergie consommée à l'exécution d'une tâche et son temps d'exécution.
- / Tâche qui sollicite à la fois le processeur et la mémoire de manière intensive.
- / Une manière d'optimiser le compromis Energie-Temps est de fixer la fréquence à la valeur optimale durant l'exécution.
- / La tâche est de nature itérative. L'idée est alors de :
  - / Estimer durant la première itération le temps d'accès mémoire et de calcul ;
  - / Fixer la fréquence à une valeur proche de l'optimal pour les itérations suivantes.
- / La réduction de la consommation énergétique a été estimée à plus de 60%.



# Conclusion

- / La consommation énergétique due aux ressources de calcul peut-être décomposé en **énergie statique** et **dynamique**. L'énergie dynamique se décompose en énergie **idle** et **active**.
- / L'énergie dynamique dépend de la fréquence d'horloge du processeur. Réduire la fréquence **réduit la puissance dissipée** mais **augmente le temps d'exécution** des tâches.
- / La modulation de fréquence et de tension (**DVFS**) consiste à **adapter la fréquence** à la **charge de travail** du processeur.
- / DVFS est mis en œuvre sous Linux par le sous-système **CPUFreq**. Celui-ci implémente différentes heuristiques de modulation de fréquence et de tension.

# Références

- / [1] C. Guo, S. Ci, Y. Zhou and Y. Yang, "A Survey of Energy Consumption Measurement in Embedded Systems," in *IEEE Access*, vol. 9, pp. 60516-60530, 2021, doi: 10.1109/ACCESS.2021.3074070.
- / [2] *Efficacité énergétique (thermodynamique)* (2023) *Wikipedia*. Available at: [https://fr.wikipedia.org/wiki/Efficacit%C3%A9\\_%C3%A9nerg%C3%A9tique\\_\(thermodynamique\)](https://fr.wikipedia.org/wiki/Efficacit%C3%A9_%C3%A9nerg%C3%A9tique_(thermodynamique)) (Accessed: 24 October 2023).
- / [3] JULIEN, N. (2015) 'Énergie et puissance dans les systèmes embarqués', *Technologies logicielles Architectures des systèmes* [Preprint]. doi:10.51257/a-v1-h8260.
- / [4] Yahia Benmoussa. Performance and Energy Consumption Characterization and Modeling of Video Decoding on Multi-core Heterogenous SoC and their Applications. Multimedia [cs.MM]. Université de Bretagne Occidentale, 2015. English.
- / [5] Blaess, Christophe. Solutions temps réel sous Linux: Cas pratique : le Raspberry Pi 3. N.p., Eyrolles, 2019.
- / [6] J. Wysocki , R. (2017) *The linux kernel, CPU Performance Scaling - The Linux Kernel documentation*. Available at: <https://docs.kernel.org/admin-guide/pm/cpufreq.html> (Accessed: 25 October 2023).