



# Beágyazott információs rendszerek

Energiaviszonyok - tervezési szempontok

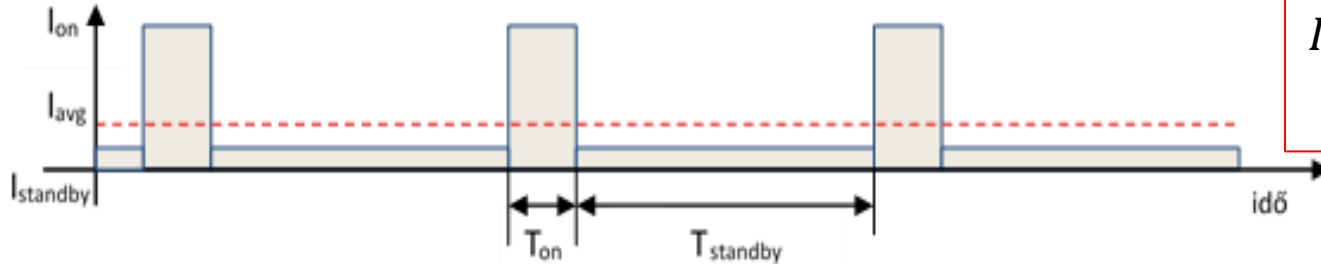
2020. október 29.

# Esettanulmányok:

## Energiaviszonyok – Tervezési szempontok

### Példa:

Két ceruzaelem „képesége”, szenzorhálózati alkalmazásban (mikrovezérlő+rádió+szenzorok):  
2 db AA elem (3000 mAh), üzemidő: min 1 év (8760 óra),  $P_{on}=150$  mW ( $I_{on}=50$ mA),  $I_{standby}=50\mu$ A.  
áramfelvételtel



$$I_{avg\ max} = \frac{3000mAh}{8760h} = 342\mu A$$

$$I_{avg} = \frac{I_{on}T_{on}}{T_{on}+T_{standby}} + \frac{I_{standby}T_{standby}}{T_{on}+T_{standby}} = I_{on}\lambda + I_{standby}(1 - \lambda),$$

$$\lambda = \frac{I_{avg\ max} - I_{standby}}{I_{on} - I_{standby}} = 0.0058 \approx 0.6\% \approx 8 \frac{perc}{nap}$$

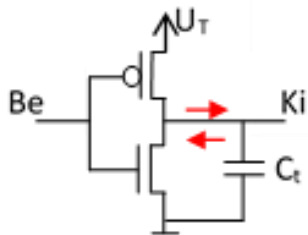
### Példa:

Ha **óránként** végzünk mérést, akkor minden órában **20 másodperc** üzemidő lehetséges.

Megjegyzés: Ha az üzemidő kb. 7 év, mint például a CO detektornál, ott  $I_{avg\ max} < 50\mu A$ !

$$7 * 60 * 8760 = 3\ 679\ 200 \text{ LED felvillanás!}$$

## CMOS áramkörök teljesítményviszonyai és energiaigénye:



A föld és a tápfeszültség között elhelyezkedő két tranzisztor felváltva zárt, és nyitott kapcsolóként üzemel. A szivárgást és az átkapcsoláskor rövid ideig tartó rövidzárási áramot leszámítva csak a kapacitív terhelést töltő/kisütő áram jelent terhelést a tápforrás számára.



A szivárgást elhanyagolva a fogyasztott teljesítmény:  $P \sim \alpha C_t U_T^2 f$  ahol  $U_T$  a tápfeszültség,  $\alpha$  az ún. kapcsolási aktivitás,  $C_t$  a terhelő kapacitás,  $f$  pedig az órajel frekvencia.

Az áramkör késleltetése:  $\tau \sim C_t \frac{U_T}{(U_T - U_k)^2}$  ahol  $U_k$  a küszöbfeszültség,  $U_k \ll U_T$ .  
Megállapítható, hogy:

- a tápfeszültség csökkentése a teljesítményt négyzetesen csökkenti;
- a késleltetés a tápfeszültség reciprokával nő,
- a maximális órajel-frekvencia csökkentés csak lineárisan csökkenti a teljesítményt.

### Az energiafogyasztás optimalizálásának lehetőségei (Dynamic Voltage Scaling: DVS):

$$P \sim \alpha C_t U_T^2 f, E \sim \alpha C_t U_T^2 f t = \alpha C_t U_T^2 (\text{ciklusok száma}).$$

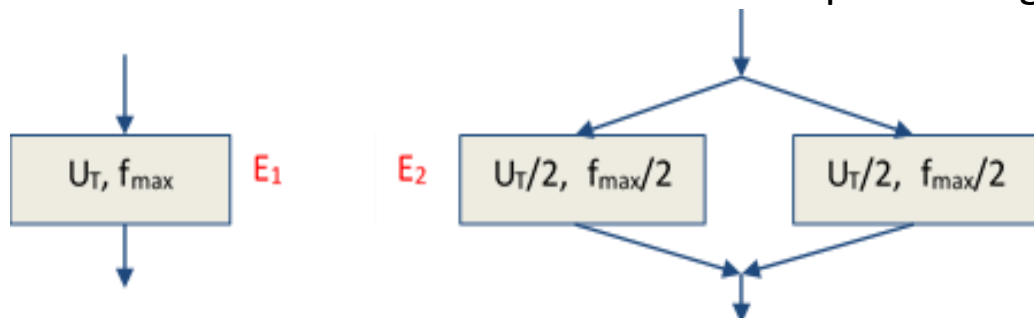
Egy adott taszk energiaigényének csökkentése érdekében:

- csökkenthetjük a tápfeszültséget;
- csökkenthetjük a kapcsolási aktivitást;
- csökkenthetjük a terhelő kapacitást;
- csökkenthetjük a ciklusok számát.

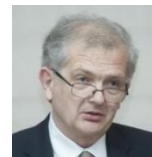
A szivárgás csökkentésének leghatékonyabb módja a tápfeszültség kikapcsolása azoknál az áramköröknél, amelyeket éppen nem használunk (Power Supply Gating).

### Párhuzamos kialakítás:

Kétszeresített hardver felére csökkentett tápfeszültséggel és órajel frekvenciával.

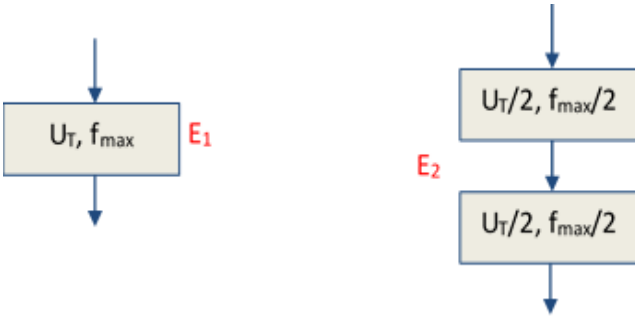


$$E_1 \sim U_T^2 (\text{ciklusok száma}), \quad E_2 = E_1/4.$$



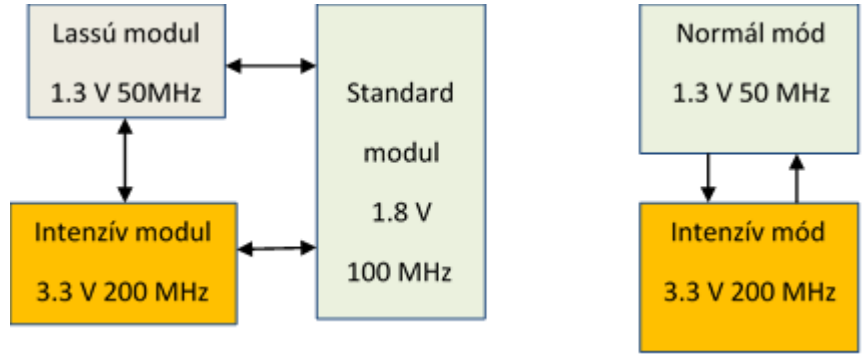
# Csővezeték kialakítás (Pipelining):

Kétszeresített hardver felére csökkentett tápfeszültséggel és órajel frekvenciával.

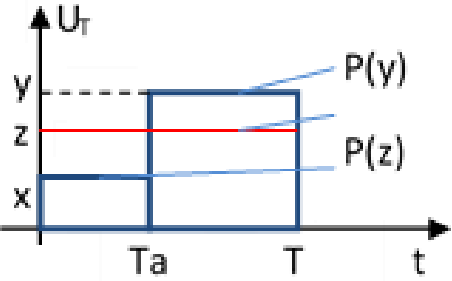


A műveletek száma változatlan, az energiaigény a negyedére csökkent.

**Nem minden komponens azonos sebesség igényű és az igény időben változhat:**



# Optimális stratégia (Dynamic Voltage Scaling):



**A eset:**  $Ta$  ideig  $x$  feszültséggel,  $(1 - a)T$  ideig  $y$  feszültséggel működtetjük az áramkört.

Az energiafogyasztás:  $T(aP(x) + (1 - a)P(y))$ .

**B eset:** végrehajtás  $z = ax + (1 - a)y$  feszültséggel  $T$  ideig.

Az energiafogyasztás:  $TP(z)$ .

Mivel a teljesítmény a tápfeszültség négyzetes függvénye, ezért,  $P(z) < aP(x) + (1 - a)P(y)$ , vagyis célszerű konstans feszültséggel működtetni az áramkört!

(A lineáris kombináció a parabola húrját adja meg, ami a parabola felett helyezkedik el.)



Példa:

$U_T$ [V]	5.0	4.0	2.5
Energia ciklusonként [nJ]	40	25	10
$f_{max}$ [MHz]	50	40	25
ciklusidő [ns]	20	25	40

Egy taszk végrehajtása  $10^9$  ciklus végrehajtását igényli.  
Ehhez 25 másodperc áll rendelkezésre.

a. Leggyorsabb végrehajtás:  $10^9$  ciklus @ 50 MHz. Ennek energiaigénye:

$$E_a = 10^9 * 40 * 10^{-9} = \mathbf{40 [J]}, \quad \text{idő igénye: } 10^9 * 20 * 10^{-9} = \mathbf{20s}.$$

b. Végrehajtás két feszültségen:  $0.75 * 10^9$  ciklus @ 50 MHz +  $0.25 * 10^9$  ciklus @ 25 MHz.

$$\text{Ennek energiaigénye: } E_b = 0.75 * 10^9 * 40 * 10^{-9} + 0.25 * 10^9 * 10 * 10^{-9} = \mathbf{32.5 [J]},$$

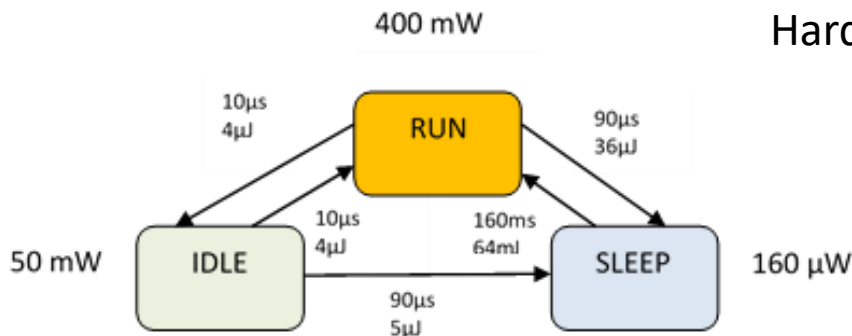
$$\text{idő igénye: } 0.75 * 10^9 * 20 * 10^{-9} + 0.25 * 10^9 * 40 * 10^{-9} = \mathbf{25s}.$$

c. Végrehajtás optimális feszültségen:  $10^9$  ciklus @ 40 MHz. Ennek energiaigénye:

$$E_c = 10^9 * 25 * 10^{-9} = \mathbf{25 [J]}, \quad \text{idő igénye: } 10^9 * 25 * 10^{-9} = \mathbf{25s}.$$

**Megjegyzés:** Értelemszerűen kell legyen idő-tartalék a taszk végrehajtásánál.

## Dinamikus teljesítmény menedzsment (Dynamic Power Management (DPM)):



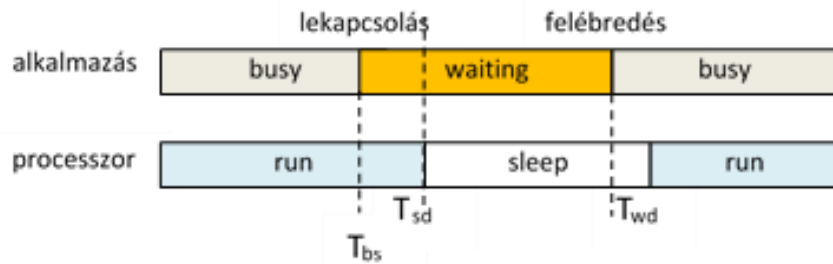
Hardver támogatást igényel. Példa: **StrongARM SA1100**

**IDLE:** egy szoftver rutin megállíthatja a processzort, ha nincs használatban, de IT-t fogad.

**SLEEP:** minden aktivitást a chipen leállít.



## A teljesítménymenedzsment időviszonyai



$T_{bs}$ : time before shutdown

$T_{sd}$ : shutdown delay

$T_{wd}$ : wakeup delay

A lekapcsolás csak hosszú használaton kívüli idők felmerülése esetén indokolt.

### Példa: Dinamikus teljesítmény menedzsment a működési frekvencia állításával

Tételezzük fel, hogy egy CMOS processzor  $P(f)$  teljesítményfelvétele  $f$  frekvencián:

$$P(f) = \left[ 10 \left( \frac{f}{100\text{MHz}} \right)^3 + 20 \right] \text{mWatt}$$

A teljesítményfelvétel csökkentését a végrehajtási frekvencia csökkentésével érjük el.

A maximum, ill. a minimum frekvencia értékek:

$f_{max} = 1000\text{MHz}$ ,  $f_{min} = 50\text{MHz}$ . A frekvencia-változtatás elhanyagolható ráfordítás igényű, a processzor  $50\text{MHz}$  és  $1000\text{MHz}$  között tetszőleges frekvencián működni tud.

A teljesítményfelvétel csökkentése érdekében a processzor **sleep** (alvó/kikapcsolt) állapotba is vezérelhető. Alvó állapotban a processzor fogyasztása elhanyagolható.

A processzor alvó állapotból futó állapotba vezérlése azonban energiát igényel, aminek értéke:

$3 \times 10^{-5} \text{Joule}$ . (Futó állapotból alvó állapotba vezérlés energiaigénye elhanyagolható.)

A ki/bekapcsolás időigénye ugyancsak elhanyagolható, pillanatszerűnek tekinthető.

A processzor három taszkot hajt végre:

	érkezési idő	határidő	ciklusok száma
$\tau_1$	0	2ms	100000
$\tau_2$	2ms	6ms	100000
$\tau_3$	6ms	7ms	80000

Itt most a végrehajtási idő a frekvencia függvénye!



A feladat szerint a processzor futó állapotban kell legyen a nulla és a 7ms időpillanatokban.

	érkezési idő	határidő	ciklusok száma
$\tau_1$	0	2ms	100000
$\tau_2$	2ms	6ms	100000
$\tau_3$	6ms	7ms	80000

### 1. feladat:

$C$  ciklus végrehajtásának energiaigénye  $\frac{CP(f)}{f}$ , hiszen  $C = tf$ , és  $E = Pt$ .

Ha túl alacsony a frekvencia, akkor a hosszú idejű végrehajtás, } növeli az energiaigényt.  
 ha túl nagy a frekvencia, akkor pedig a növekvő teljesítményfelvétel }

Létezik  $50MHz$  és  $1000MHz$  között egy olyan  $f_{krit}$  kritikus frekvencia, amely mellett tetszőleges számú  $C$  ciklus energia felvétele minimális!

### Mekkora ez a frekvencia az adott processzor esetében?

Megoldás: Keressük  $\frac{P(f)}{f}$  minimumát!

Vezessük be:  $f_n = \frac{f}{100MHz}$  normalizált frekvenciát!

$\frac{P(f_n)}{f_n}$  első deriváltja  $20f_n - \frac{20}{f_n^2}$ , ami akkor nulla, ha  $f_n = 1$ . Ezzel  $f_{krit} = 100MHz$

$$P(f) = \left[ 10 \left( \frac{f}{100MHz} \right)^3 + 20 \right] mWatt$$

**2. feladat:** A processzor *idle* (tétlen) állapotban  $f_{min}$  frekvenciával jár, és ekkor  $t$  másodperc alatt  $P(f_{min}) \times t$ .

**Megtérülési időnek** (*break-even time*) nevezzük annak az intervallumnak a hosszát, amelynek elteltével érdemes a processzort tétlen állapotból alvó állapotba kapcsolni.

### Mekkora a processzor megtérülési ideje?

Megoldás: Akkor érdemes alvó állapotba kapcsolni, ha az elérhető energia megtakarítás már fedezi a visszakapcsoláshoz szükséges  $3 \times 10^{-5}$  Joule többlet ráfordítást!





Energia(tétlen,  $f_{min}$ ) – Energia(alvó)  $\geq$  Energia(alvóból futó állapotba)

$$P(f_{min}) * t_{megtérülési} - 0 \geq 3 * 10^{-5} \text{Joule}$$

$$t_{megtérülési} \geq \frac{3 * 10^{-5} \text{Joule}}{10^{-3} * (10 * 0.5^3 + 20) \text{Watt}} = 1.412 \text{ms.}$$

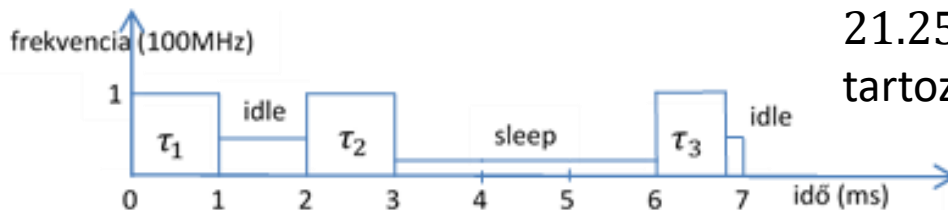
### 3. feladat:

Az ún. munkaterhelést megőrző ütemezést (*workload-conserving schedule*) úgy definiáljuk, mint egy olyan ütemezés, amely mindig végrehajt valamely taszkot hacsak a futásra kész taszkok listája nem üres.

Készítsünk ilyen ütemezést, amely minimalizálja az energiafogyasztást és egyidejűleg betartja a három taszkra vonatkozó ütemezési előírásokat!

Ehhez a taszk végrehajtások során használjuk az  $f_{krit}$  frekvenciát. **Mekkora az**

**energiafelhasználás ilyenkor?** Megoldás: Mivel  $P(f_{krit}) = 30 \text{mWatt}$ , és  $P(f_{min}) = 21.25 \text{mWatt}$ , így az ábra szerinti ütemezéshez tartozó energia-fogyasztás:



$$(30 \times 1 + 21.25 \times 1 + 30 \times 1 + 3 \times 10^1 + 30 \times 0.8 + 21.25 \times 0.2) \mu \text{Joule} = \mathbf{0.1395 \text{ mJoule}}$$

Vegyük észre, hogy az [1,2] intervallum rövidebb, mint a megtérülési idő, ezért nem merül fel az alvó állapotba kapcsolás.

### 4. feladat:

Lehet-e olyan munkaterhelést megőrző ütemezést készíteni, ami ennél is kisebb energiafogyasztással jár, miközben betartjuk a három taszkra vonatkozó ütemezési előírásokat?

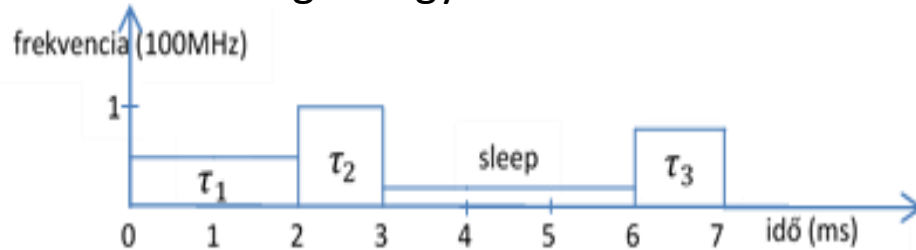




Megoldás:

Igen, mert kihasználható a teljesítményfelvétel konvex jellege: lassítható  $\tau_1$  és  $\tau_3$  végrehajtása oly mértékben, hogy ne kelljen *idle* állapotba kapcsolni!

Ezzel ugyanis annak ellenére, hogy a kritikus frekvenciánál alacsonyabbat alkalmazunk, mégis kevesebb energiát fogyasztunk.



A 3. feladatban: **0.1395 mJoule**

$$(21.25 \times 2 + 30 \times 1 + 3 \times 10^1 + (10 \times 0.8^3 + 20) \times 1) \mu\text{Joule} = \mathbf{0.12762 \text{ mJoule}}$$

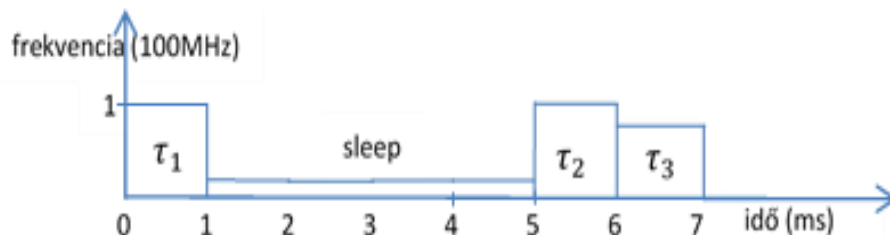
### 5. feladat:

Lehet-e olyan ütemezést készíteni, akár a munkaterhelést megőrző stratégia feladásával, amely még ennél is kisebb energiafogyasztással jár?

Megoldás:

Igen. A megoldás lényege, hogy egy blokkba gyűjtjük azokat az időszakokat, ahol a processzor nem fut, mert ezáltal érdemes lesz azonnal alvó állapotba küldeni, mielőst az lehetséges.

Ezen kívül a  $\tau_1$  task végrehajtásához a *kritikus frekvenciát* használjuk.



Ez a példa azt illusztrálja, hogy a munkaterhelést megőrző stratégiák nem feltétlenül a legjobbak.

$$(30 \times 1 + 3 \times 10^1 + 30 \times 1 + (10 \times 0.8^3 + 20) \times 1) \mu\text{Joule} = \mathbf{0.115120 \text{ mJoule}}$$

