

„Ha beérjük annyival, hogy elátkozzuk vagy dicsőítjük a technikát, akkor sohasem jutunk el lényegének a megragadásához.”

Martin Heidegger

MIKROVEZÉRLŐS RENDSZERFEJLESZTÉS

ChibiOS/RT

Threads: Gyakorlat

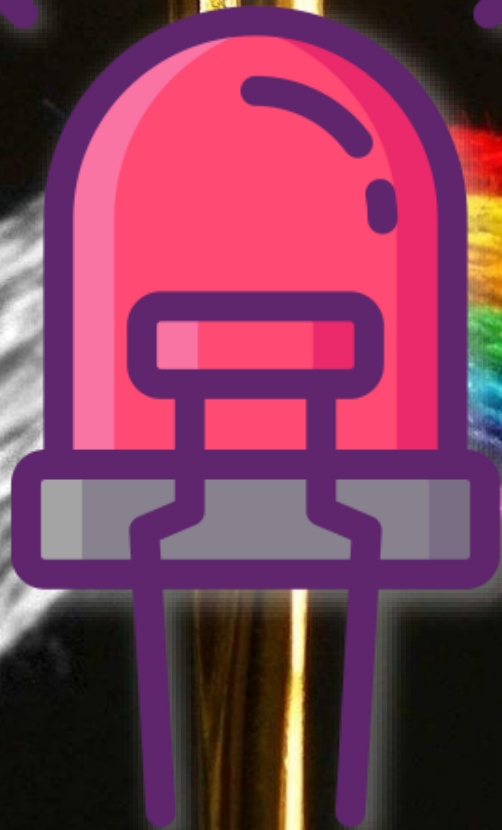
Zsupányi Krisztián

Threads: Gyakorlat



STARTING STARTING STARTING STARTING

**FELADAT:
LED SZÁL
SHELL-BŐL**



LED SZÁL, ELINDÍTÁSSAL ÉS LEÁLLÍTÁSSAL

Készíts egy ChibiOS alapú programot, amelyben **egy külön szál (thread) felel egy LED villogtatásáért**, és ezt **a szálát parancssoron (shellen) keresztül lehet elindítani és leállítani!**

Stream lekérése Chp nélkül:

`smShellGetHandler(SHELL_INTERFACE_CDC)`

Feladat követelményei:

1. Indítás és leállítás:

- A „**blink**” nevű shell-paranccsal lehessen, elindítani a LED villogtató szálát, ha az még nem fut vagy leállítani azt, ha már fut.

2. Villogtató szál működése:

- A szál neve legyen **"blink,,** , a szál a **GPIO port LED3** lábát **100 ms-onként váltogassa**.
- **Leállításkor** a LED-et kapcsolja ki, majd **írja ki** a shellre, hogy a **„Blink thread vége”**.

3. Szálkezelés:

- A leállítást **chThdTerminate()** és **chThdWait()** segítségével oldd meg.
- A szálban a kilépést **chThdShouldTerminateX()** figyelésével valósítsd meg.

```
elif operation == "MIRROR_X":
    mirror_mod.use_x = False
    mirror_mod.use_y = True
    mirror_mod.use_z = False
elif operation == "MIRROR_Z":
    mirror_mod.use_x = False
    mirror_mod.use_y = False
    mirror_mod.use_z = True

#selection at the end -add back the deselected mirror modifier object
mirror_ob.select= 1
modifier_ob.select=1
bpy.context.scene.objects.active = modifier_ob
print("Selected" + str(modifier_ob)) # modifier ob is the active ob
mirror_ob.select = 0
time = bpy.context.selected_objects[0]
bpy.data.objects[time.name].select = 1
```

LEDEK – VILLOGÁS SEBESSÉG SHELL-BŐL

Threads: Gyakorlat

LEDEK – VILLOGÁS SEBESSÉG SHELL-BŐL

Írj ChibiOS programot, ami 4 statikus szálon futó LED-et kezel; a LED-eket shell-parancsokkal lehet elindítani/leállítani és a villogás periódusát beállítani.

Feladat követelményei:

- Négy LED (index 1..4), érdemes `led_pads[]` tömbböt készíteni.
- Minden LED-hez külön, statikus szál (`chThdCreateStatic`) fut, lehet generikus `LedThread` függvénnnyel; a szálnak kapnia kell az LED indexét (`arg`).



1. Szál viselkedése:

- **ha** `led_running[idx] == true` \Rightarrow a `led_period_ms[idx]` alapján villog (toggle, sleep period/2),
- **ha** `false` \Rightarrow LED kikapcsolva, rövid (100 ms) várakozás.
- **ha** `period = 0`, használjon alapértelmezett 500 ms-ot.

2. Shell-parancs: `led <start|stop|speed> <id> [ms]`

- **led start <id>** — elindítja a megadott LED-et (1–4), majd villogtatja adott periódusidővel.
- **led stop <id>** — leállítja és kikapcsolja a LED-et, `led_running=false`.
- **led speed <id> <ms>** — beállítja a periódust (ms) a megadott LED-re.

Például: **led start 3**

Például: **led stop 3**

Például: **led speed 3 200**

3. Hibakezelés: érvénytelen id esetén hibaüzenet („Ismeretlen id”), speed-nél hiányzó ms esetén hibaüzenet.

LEDEK – VILLOGÁS SEBESSÉG SHELL-BŐL

```
static THD_WORKING_AREA(waLed1, LED_THREAD_STACK); #define LED_THREAD_STACK 256
static THD_WORKING_AREA(waLed2, LED_THREAD_STACK);
static THD_WORKING_AREA(waLed3, LED_THREAD_STACK);
static THD_WORKING_AREA(waLed4, LED_THREAD_STACK);
```



Periódus: **be** + **ki** állapot ideje

1x

```
static THD_FUNCTION(LedThread, arg) {
    int idx = (int)(intptr_t)arg;
    chRegSetThreadName("led");
    while (true) {
        ...
    }
}
```

String összehasonlítás:

strcmp(<str1>, <str2>) == 0

```
chThdCreateStatic(waLed1, sizeof(waLed1), LED_THREAD_PRIO, LedThread, (void*)0);
chThdCreateStatic(waLed2, sizeof(waLed2), LED_THREAD_PRIO, LedThread, (void*)1);
chThdCreateStatic(waLed3, sizeof(waLed3), LED_THREAD_PRIO, LedThread, (void*)2);
chThdCreateStatic(waLed4, sizeof(waLed4), LED_THREAD_PRIO, LedThread, (void*)3);
```


„Ha beérjük annyival, hogy elátkozzuk vagy dicsőítjük a technikát, akkor sohasem jutunk el lényegének a megragadásához.”

Martin Heidegger

KÖSZÖNÖM A FIGYELMET!

Zsupányi Krisztián

Threads: Többszálú alkalmazás



ENDING ENDING ENDING ENDING E