

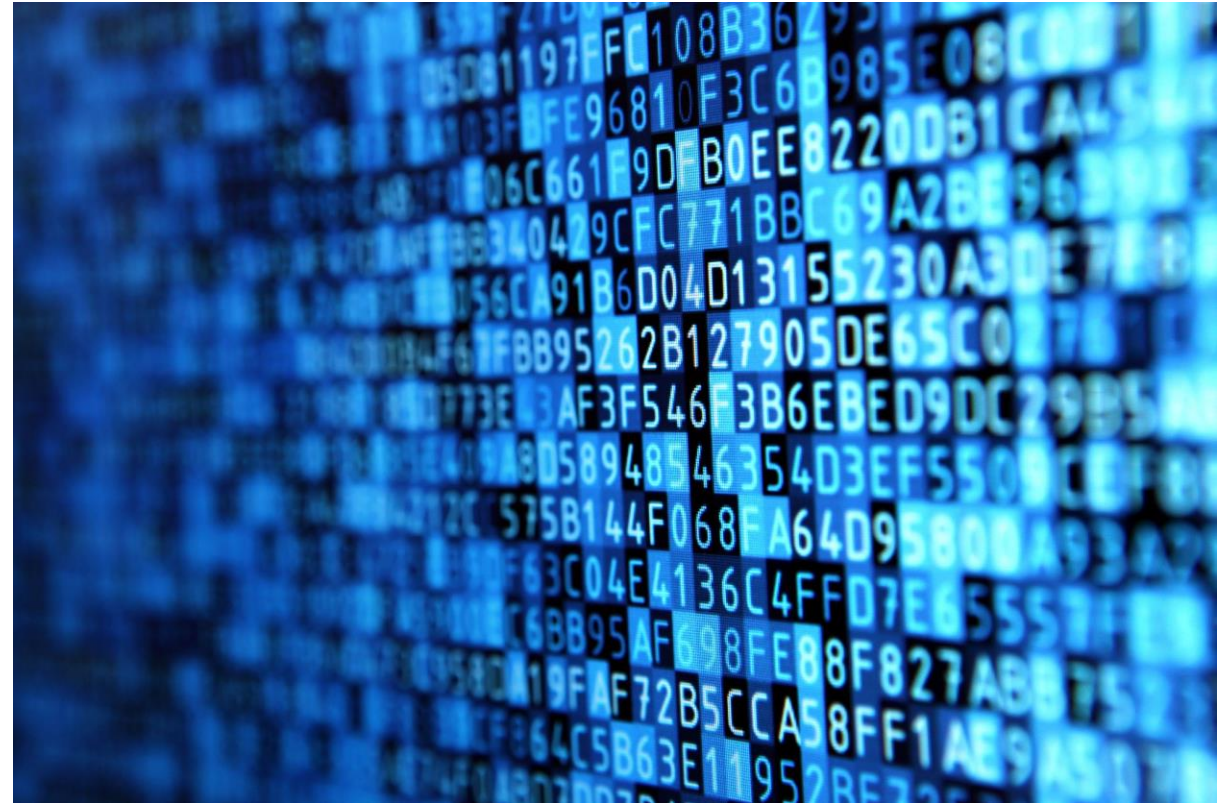
Mikrokontrollerit

Luento 5: Sulautetun sovelluksen rakenne ja kehittäminen

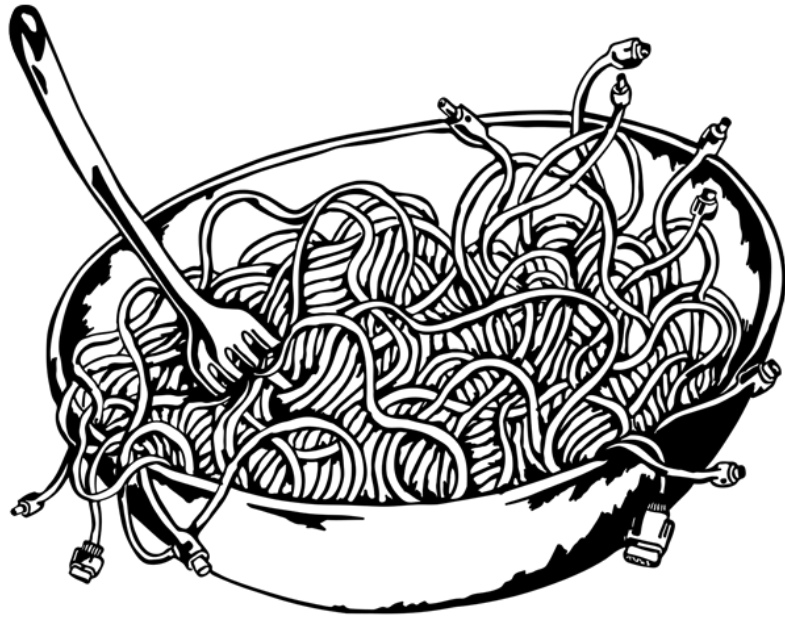
Olli Himanka, 10.11.2023

Luennon sisältö

- Sulautetun ohjelmiston rakenne: tyypillisiä ongelmia
- Yksinkertainen loop-rakenne: cyclic executive
- Event-loop –rakenne
- Moniajo
- Free RTOS



Ongelmallinen rakenne – huono esimerkki



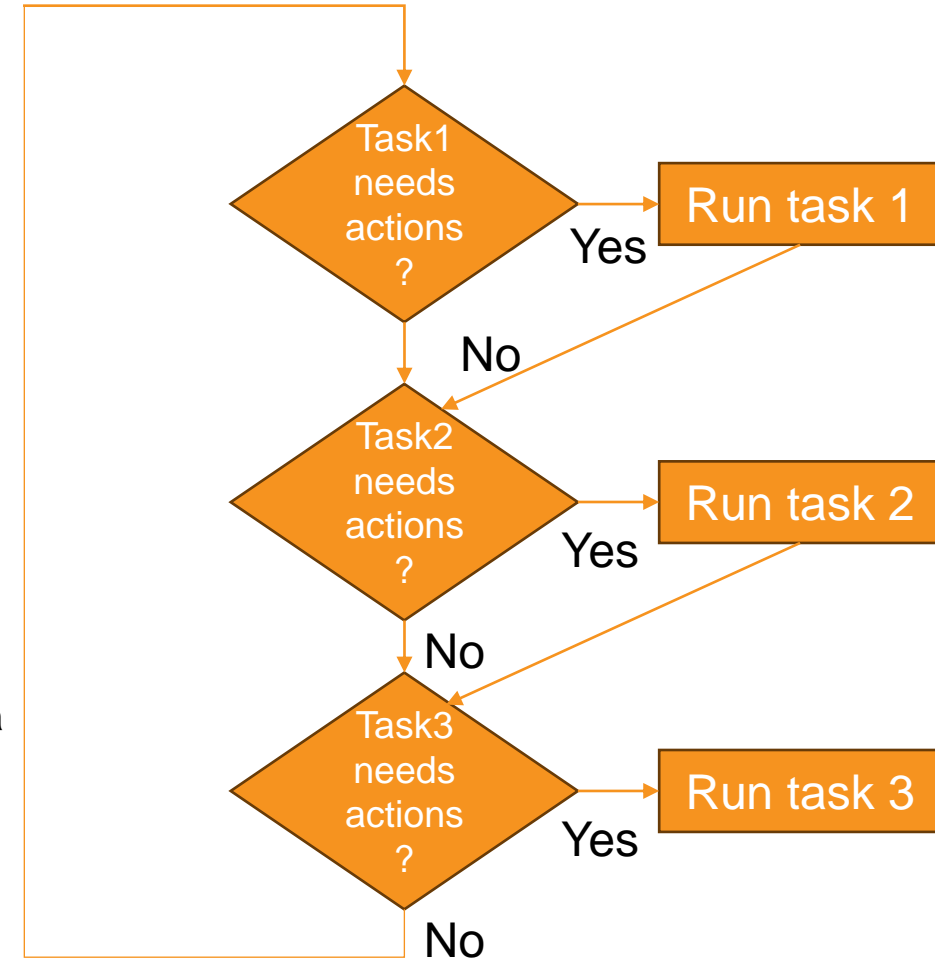
SPAGHETTI
CODE

Esimerkkikoodi 1: Mitä ongelmia huomaat?

- Ajastus tehty delay:n avulla
- -> Ohjelma ei vastaa syötteisiin reaaliajassa

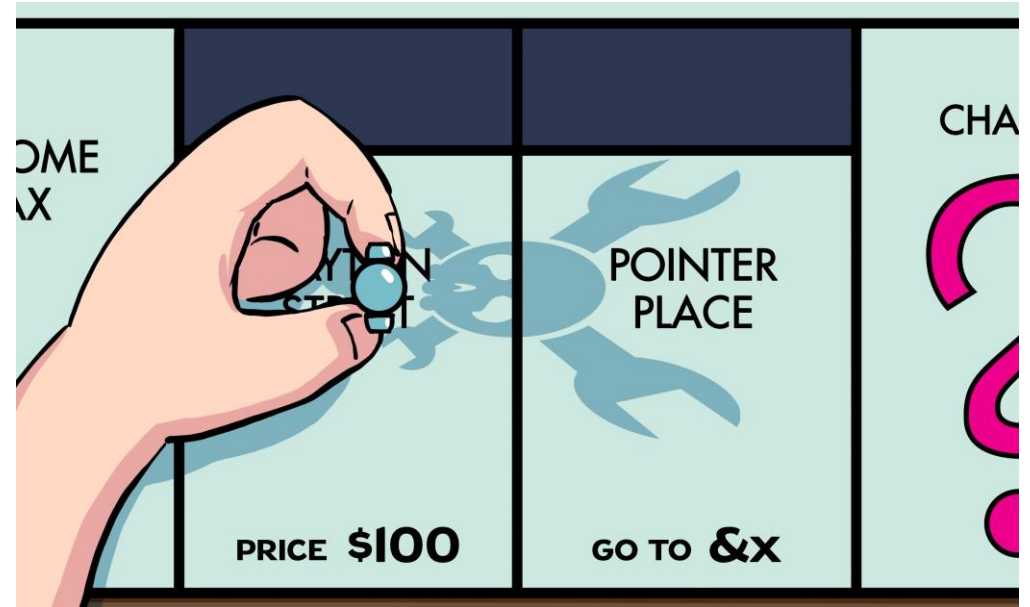
Loop -rakenne

- Arduinon perusrakenne:
 - Koodi pyörii pää-loopissa
- Esierkkikoodi 2:
 - Ehdollisia suorituksia
 - Ajoitus tehty timerin avulla
 - Koodi saa pyöriä koko ajan, ei pysäytetä delaylla
 - Hoidetaan taskit kun niiden suorittamisen ehdot täyttyy.
 - Vähänkään monimutkaisemmassa koodissa tarvitaan "tilakone"
 - Tilakone tarkoittaa että ohjelmalla on eri tiloja, jotka säilytetään tilamuuttujassa
- Event-loop :
 - Variaatio, jossa pää-loop lukee eventtejä jonosta.
 - Esim: Aikaa kulunut tietty määrä, nappia painettu.
 - Lähes kaikki koodi on periaatteessa vain eri eventtien käsittelijöitä



Cyclic executive

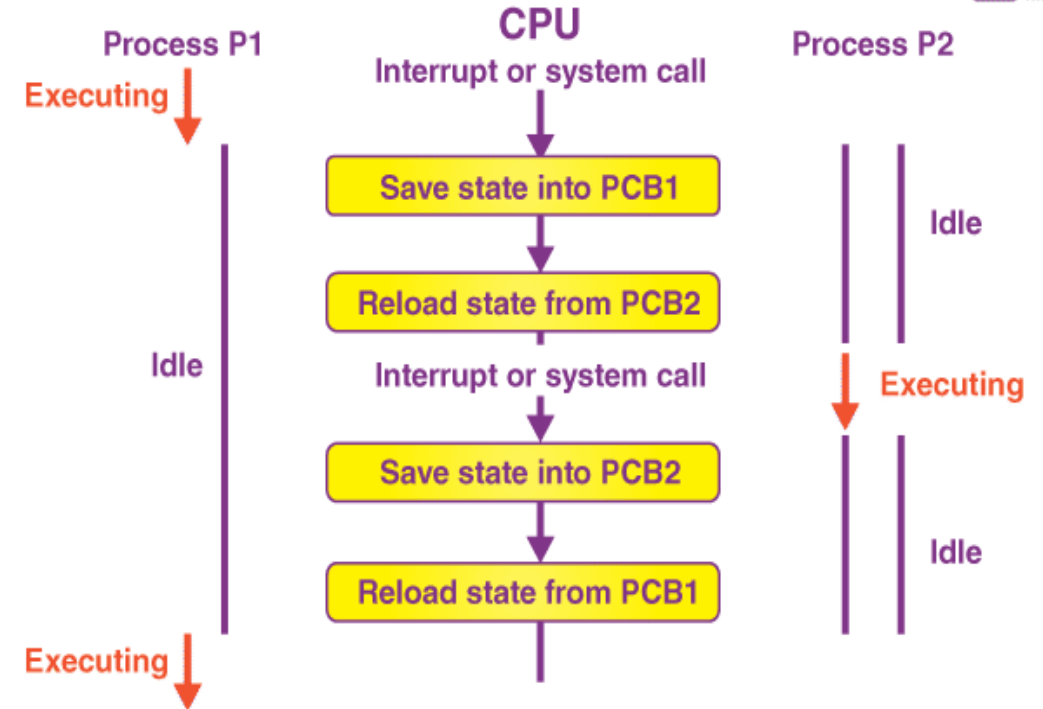
- Esimerkkikoodi 3
- Köyhän miehen moniajo
- Funktiopointterit
- Mahdollistaa suoritettavien taskien vaihtamisen, käynnistyksen ja sammutuksen ajon aikana.
- Taskien täytyy itse huolehtia, että vapauttavat suoritusvuoron muille, kun ovat valmiita.
- "Co-operative multitasking"



<https://hackaday.com/2018/05/02/directly-executing-chunks-of-memory-function-pointers-in-c/>

Pre-emptiivinen Moniajo

- Todellisessa moniajossa jokainen prosessi voi puuhailla omiaan.
- Käyttöjärjestelmän scheduler hoitaa prosessin keskeyttämisen, ja toisen prosessin jatkamisen.
- ”pre-emptiivisyys”, eli suoritus riistetään prosessilta halusi se tai ei.
- Jotta tämä voisi toimia, pitää tehdä ns. kontekstin vaihto. Kaikki rekisterit ja liput tallennetaan, ja palautetaan.



Kontekstin vaihto ja Schedulointi

- Kontekstin vaihdon liipaisee yleensä timer. ”On aika vaihtaa seuraava prosessi ajoon”
- Seuraavat pitää tallentaa ajossa olleesta prosessista, ja palauttaa ajoon laitettavaa prosessia varten:
 - Program counter
 - Pino osoitin
 - Liput
 - Rekisterit
- Kontekstin vaihto kuluttaa suoritusaikaa.
- Prosesseille voidaan allokoida aikaa eri periaattelilla, esim:
 - Round Robin: kaikki prosessit saavat saman verran suoritusaikaa
 - Hard priority: Prosesseilla on prioriteetti. Korkeamman prioriteetin prosessi saa aina aikaa ohi matalampien, niin paljon kunnes se on valmis luopumaan suorituksesta.

Free RTOS esimerkki

- FreeRTOS on ilmainen avoimen lähdekoodin moniajava reaaliaikakäyttöjärjestelmä
- Tukee Arduinoa
- Helppo käyttää
- Esimerkkikoodi 4

Kirjasto asennetaan Arduinon library managerista

```
#include <Arduino_FreeRTOS.h>
```

`xTaskCreate` käynnistää taskin



Moniajon sudenkupat

- Jaetut resurssit: Esim muuttujat/muistialueet, erilaiset oheislaitteet
 - Mutex (estetään yhtäaikainen käyttö)
 - "deadlock". Ohjelma jumittuu lukkoon, koska taskit varaavat resursseja "ristiin"

