

DRIVER

↳ modulo S.O. esterno al kernel dedicato alla gestione di una specifica periferica,  
che viene virtualizzata

→ interagisce con: job, psys, kernel;

→ periferiche viste come file speciali

→ 2 tipi di periferiche: 1 → caratteri, accesso seq. (1 byte o 1 blocco byte alla volta)

2 → a blocchi, accesso seq tram. indicizzazione blocchi

portiamo 2 tipi di driver: 1 → caratteri, interagiscono direttamente con il filesystem

2 → blocchi, interagiscono attraverso buffer con fsys e utilizziamo DMA; → direct memory access

→ quindi le periferiche sono 'trasformate' in file speciali:

- contenuti nella directory /dev;
- hanno nomi per specificare la periferica;

→ **i-mode associato:**

- indice tipo driver (c, b);
- major number, tipo di periferica;
- minor number, specifica tra i tipi di un tipo di perif;

scopro da

**ACCESSO** → 'nome file' → i-mode → major numb → minor numb → permette di accedere univoc al device del sistema  
è file spec.  
accesso al driver corretto (funz. richiesta)

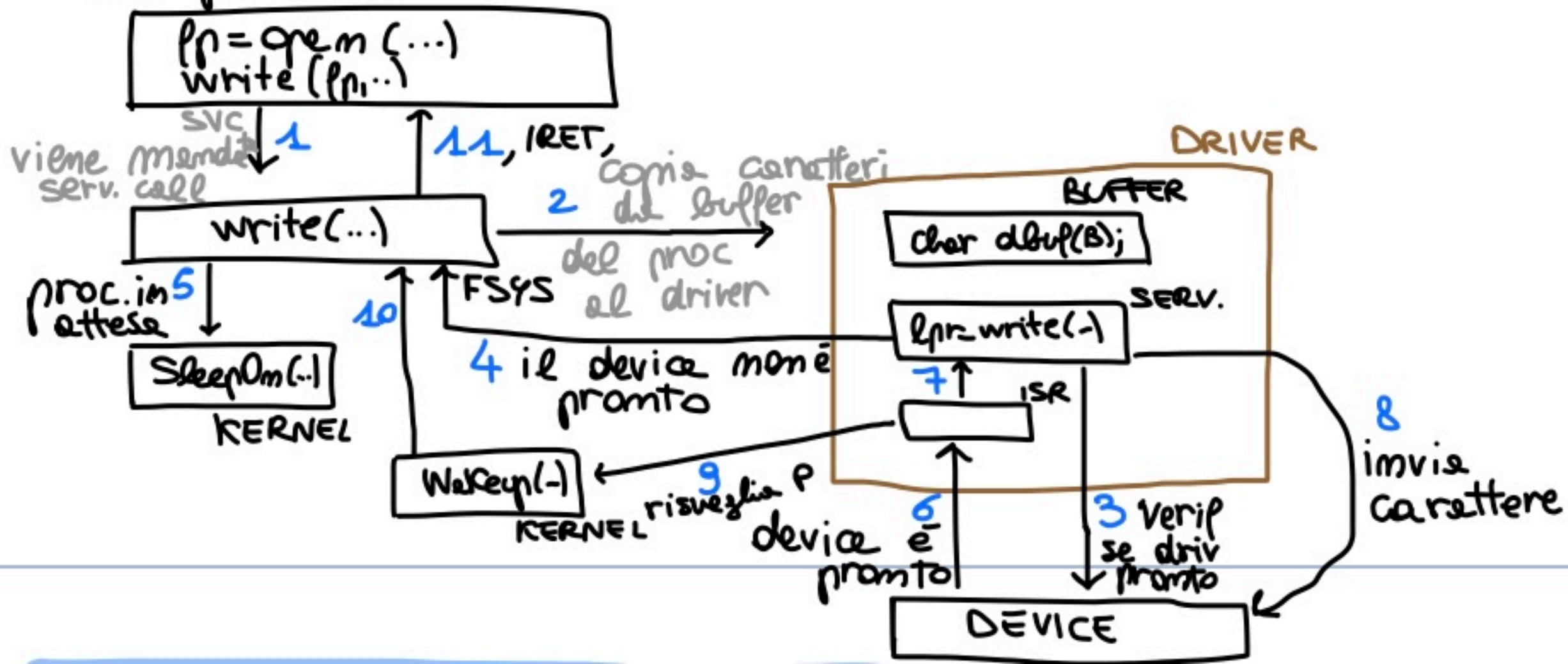
→ il **driver** può essere rappresentato da una **struttura** che raggruppa punt a funzioni che rappresentano i servizi;

↳ la prima funzione alla chiamata del driver inizializza le specifiche; copiate in una tabella

→ - device switch table: major number repr riga nella tab (device specifico);  
→ le colonne sono le funzioni possibili

## ACCESSO A PERIF A CARATTERE

nel processo





# ACCESSO PERIF A BLOCCHI

