RTOS :::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::

**Round Robin**

Technisch:  
- Schneidet die Task ab. Einstellen über OS\_Ticks.

- Wo auch immer Task ist, stoppt und der nächste fährt weiter.

Effekt:

Zwei Prozesse erscheinen Parallel. Beide laufen.

……………………………………………………………………………………………………………

**Prioritäten**

Technisch:

- Übersteuern Round Robin!

- Sobald ein Task höher ist, bricht der der Scheduler nach der Round-Robin-Zeit ab,   
 Doch da der Task mit höherer Priorität immer zu erst kommt,   
 kommt derselbe Task gleich nochmals dran!

Effekt:

- Nur Task mit hoher Priorität laufen

- Tiefe Prioritäten kommen nie dran!

……………………………………………………………………………………………………………

**Warum hat es Round Robin, wenn Prioritäten das Round Robin übersteuern?**

**Gibt es einen Weg, dass auch tiefere Prioritäten zum Zug kommen?**

- Eigentlich nicht

- Es braucht einen „Hack“: Man schickt die höheren Task zum Schlafen.

Wie lange sie schlafen, ist genau definiert.  
 Sobald sie aufwachen, stehen sie in der Schlange wieder zu forderst, und dominieren.

- Niedere Task kommen nur während der Schlafenszeit der höheren zum Zug.

**Grund für Round Robin bei Prioritäten**

- 2 (hohe oder auch niedere) Task haben die gleiche Priorität:

Durch Round Robin wechseln sich diese zwei Task Priorität ab.  
 Task laufen parallel. (Gilt sowohl für die hohen Task, die sich teilen, wie auch die tiefen Sammeltasks.)

……………………………………………………………………………………………………………

**Design**

- Ausschlaggebend ist nur der Task, der am zeitkritischsten ist. (Bei uns: Audio Task).  
 Bsp: Der Audiotask muss bevor sein Buffer leer ist, wieder den Buffer neu füllen.   
 Nur so läuft das Lied ununterbrochen.

Dieser Task hat die höchste Priorität.

- Alle anderen Task sind nicht wichtig und teilen sich die Ressourcen über Round Robin.  
- Alle nicht wichtigen Tasks kommen der Reihe nach dran.

……………………………………………………………………………………………………………

**Falle**

Pre-emptive (sobald fertig, weiter geben) ohne Round Robin.   
Ist der Task nie fertig, gibt es keinen pass() und keine Task werden mehr ausgeführt.