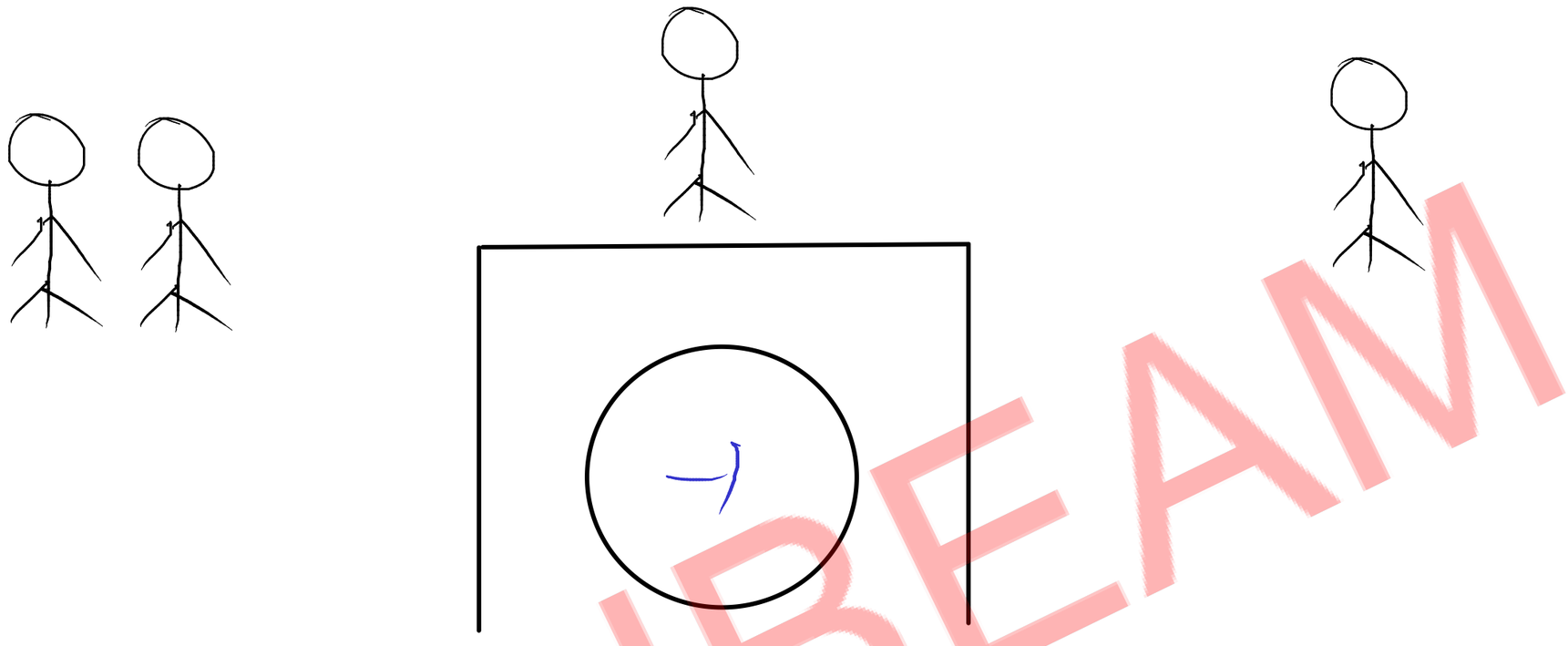
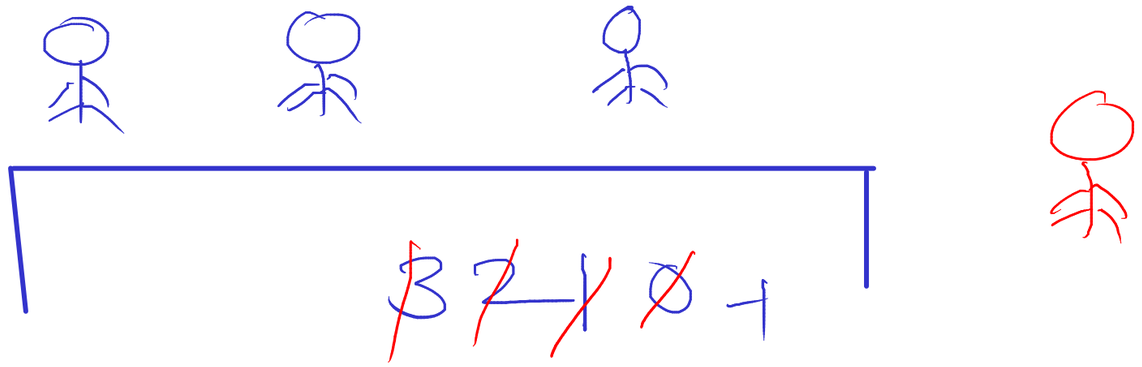


② Inc
→ cnt++
→ if someone is blocked on semaphore, then wakeup anyone

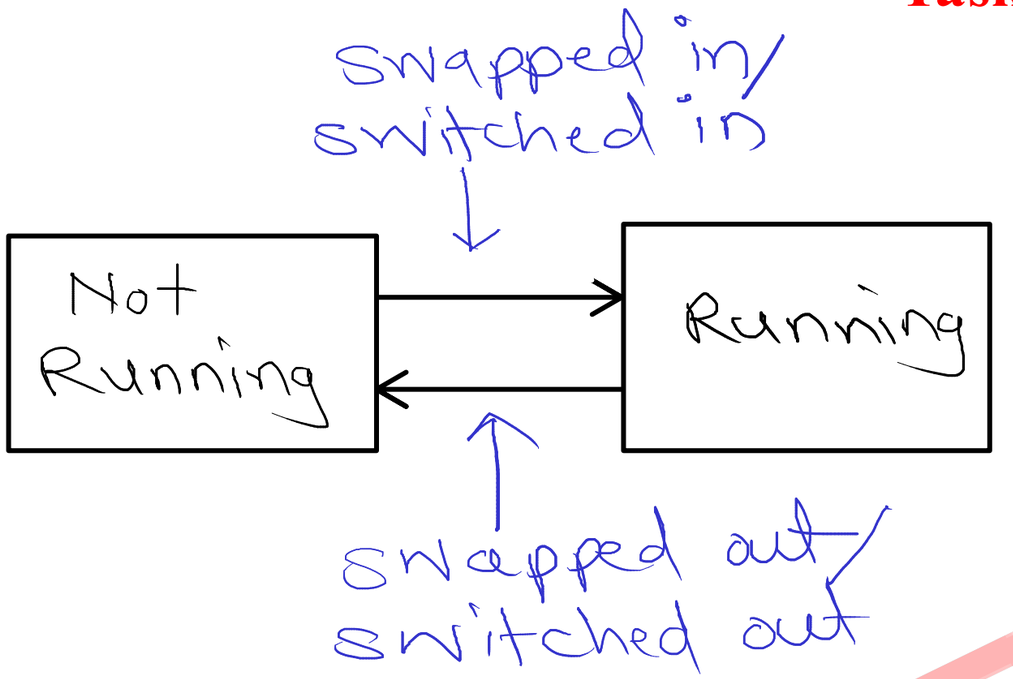
① Dec
→ cnt--
→ if(cnt < 0)
block current process

SUNBEAM

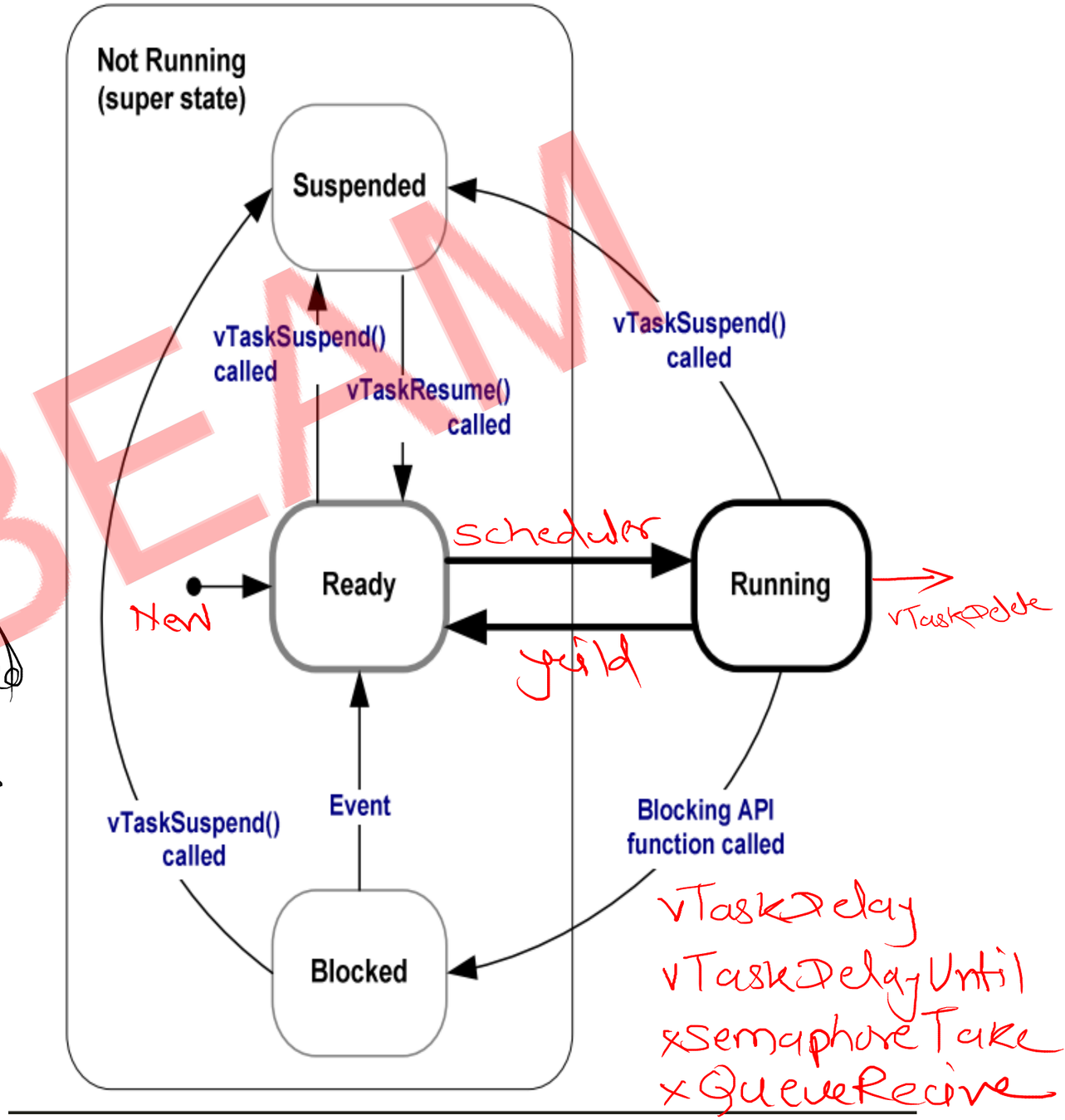


SUNBEAM

Task States



1) Suspended
   - infinite block
   - task will be susped only when vTaskSuspend is called
   - task will resume when vTaskResume will be called



SUNBEAM

2> Ready

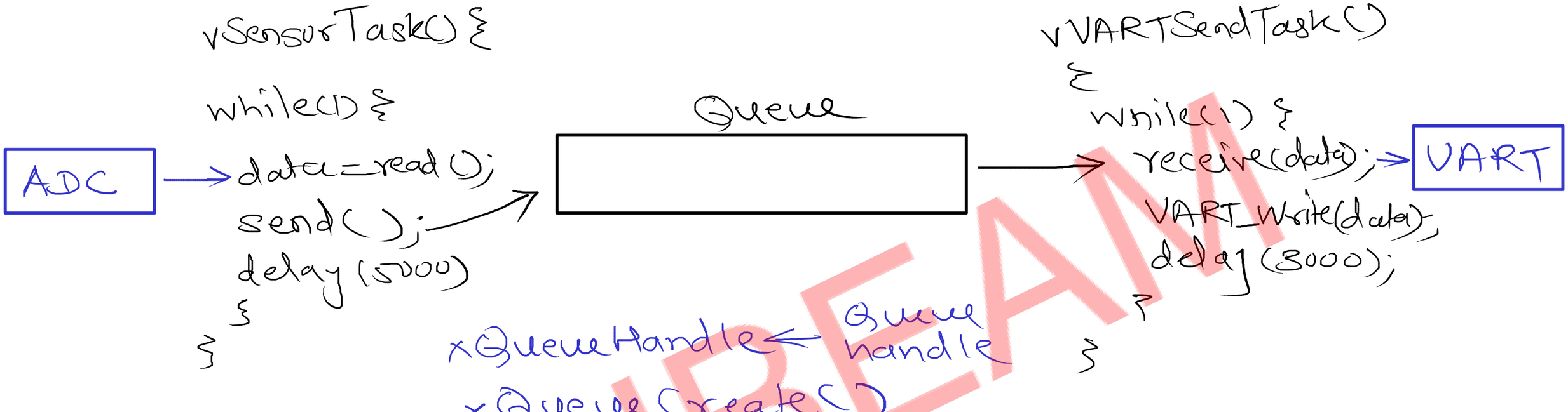
- if task is waiting to get CPU time (slice)

3> Blocked

- Task will be blocked due to any one of the reason below
  - I/O/event request
  - synchronisation
  - ITC - queue
  - blocking call (delay)

SUNBEAM

Queue



xQueueHandle ← Queue handle

xQueueCreate()

xQueueSendToback() / xQueueSendToFront()

xQueueReceive()

xQueuePeek()

xQueueMessagesWaiting()

SUNBEAM

Gate Keeper Task

```
vExtIntTask()
{
  while(1) {
    wait();
    send(cnt);
  }
}
```

GPIO →

```
vSensorTask() {
while() {
  data=read();
  send();
  delay(5000)
}
}
```

ADC →

```
vSensorTask() {
while() {
  data=read();
  send();
  delay(5000)
}
}
```

DAC →

Queue

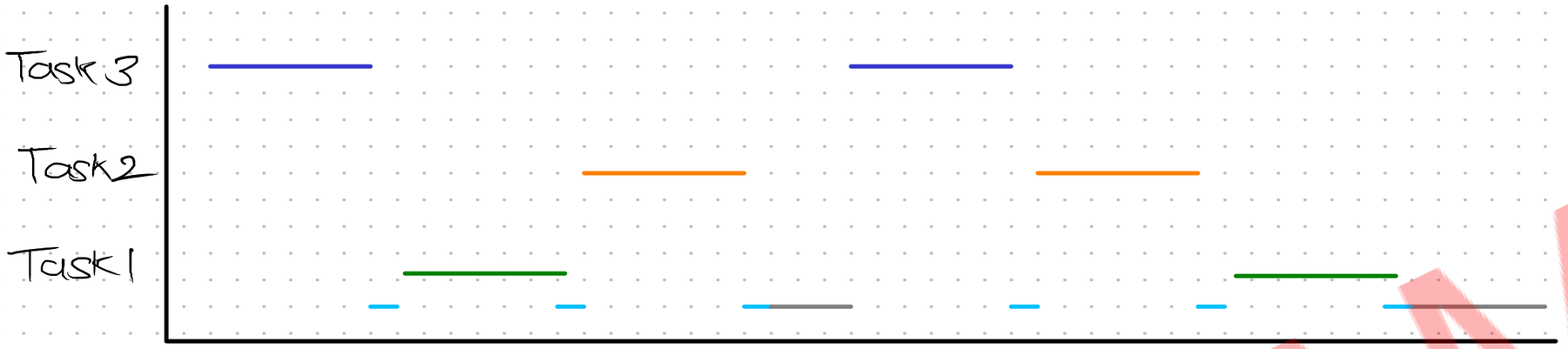
critical reading/ value can be pushed from front end

```
vVARTSendTask()
{
  while(1) {
    receive(data);
    VART_Write(data);
    delay(3000);
  }
}
```

→ VART

- only one task is performing all the operations of that peripheral is known as gate keeper task.
- one of the design pattern used in embedded.

SUNBEAM

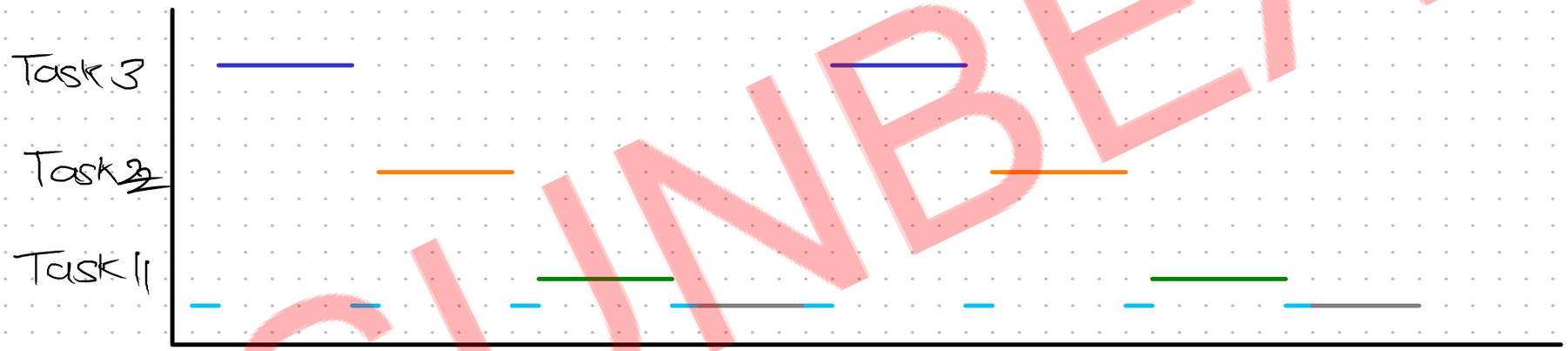


Priorities

Task1=1

Task2=1

Task3=1



Priorities

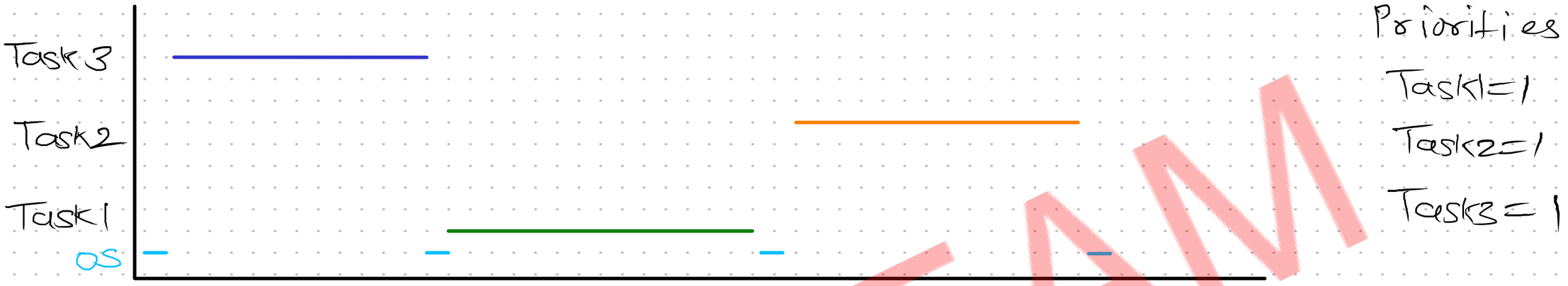
Task1=1 (L)

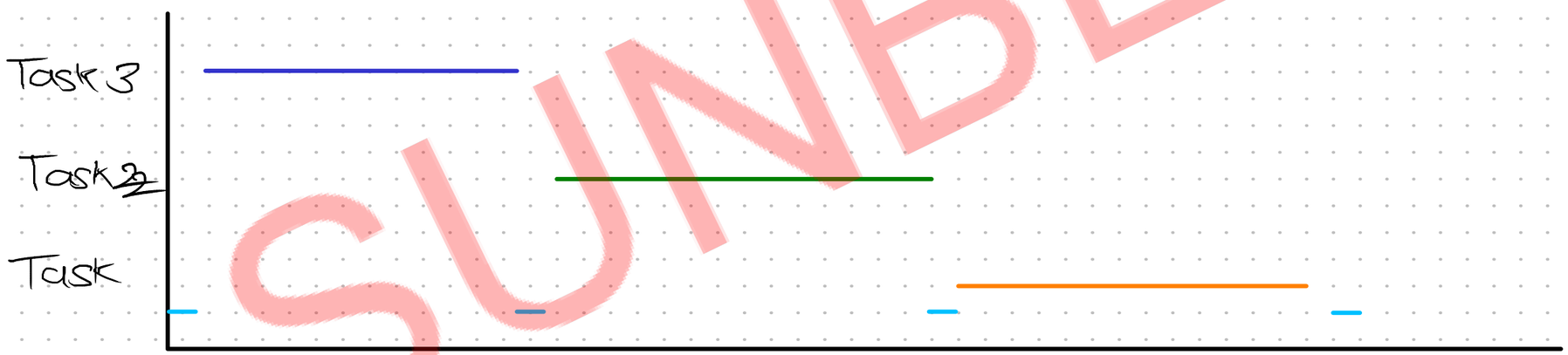
Task2=3 (M)

Task3=5 (H)

SUNBEAM

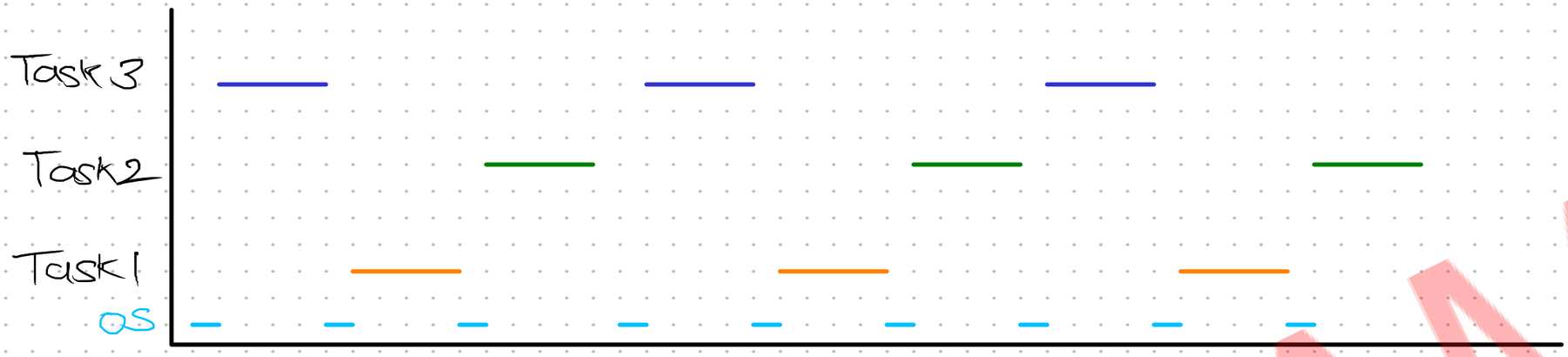
Co operative Scheduling





SUNBEAM

Co operative Scheduling

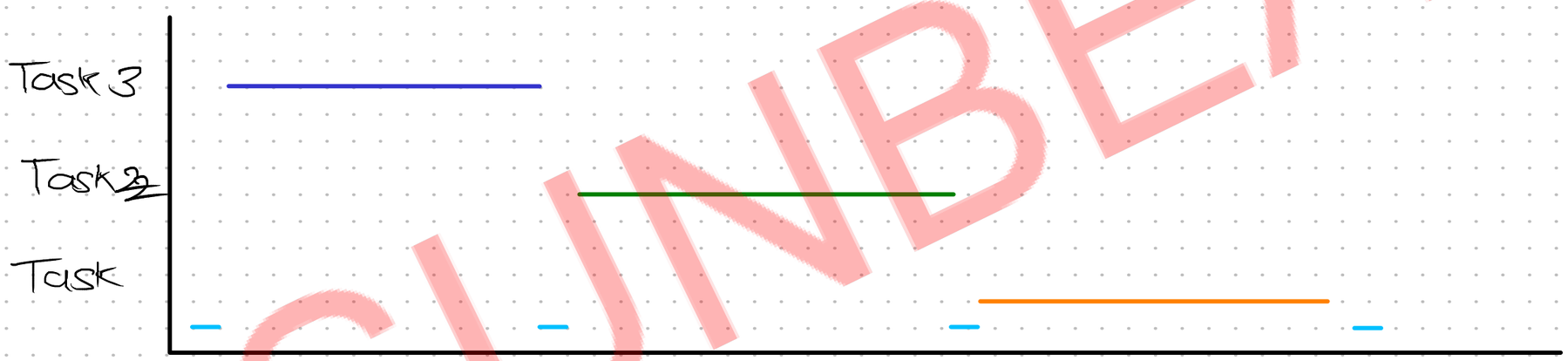


Priorities

Task1=1

Task2=1

Task3=1



Priorities

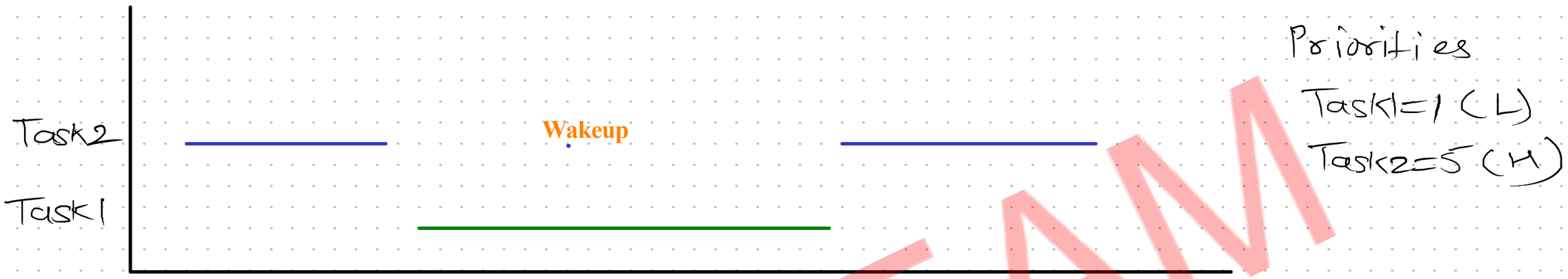
Task1=1

Task2=3

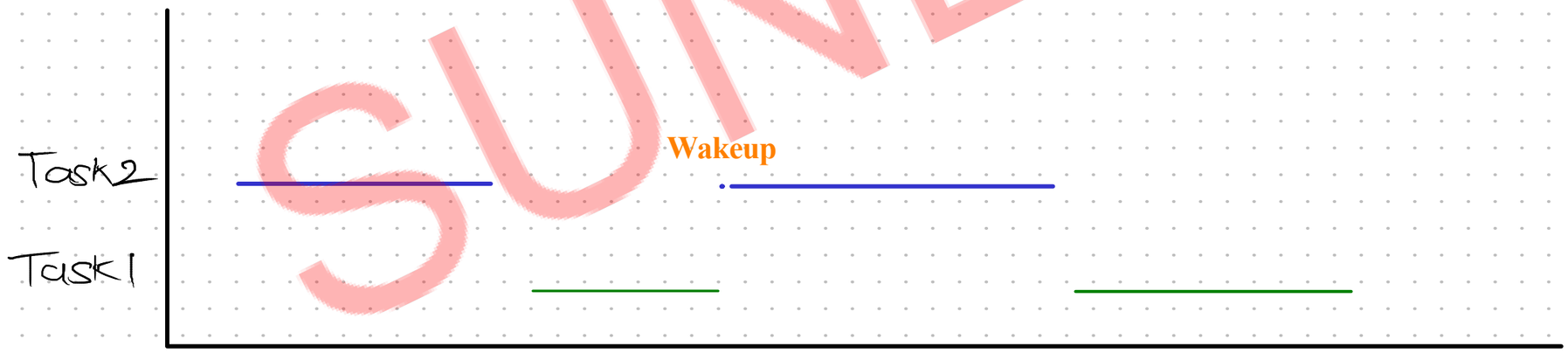
Task3=5

SUNBEAM

Co operative Scheduling



Preemptive Scheduling



SUNBEAM

RMA

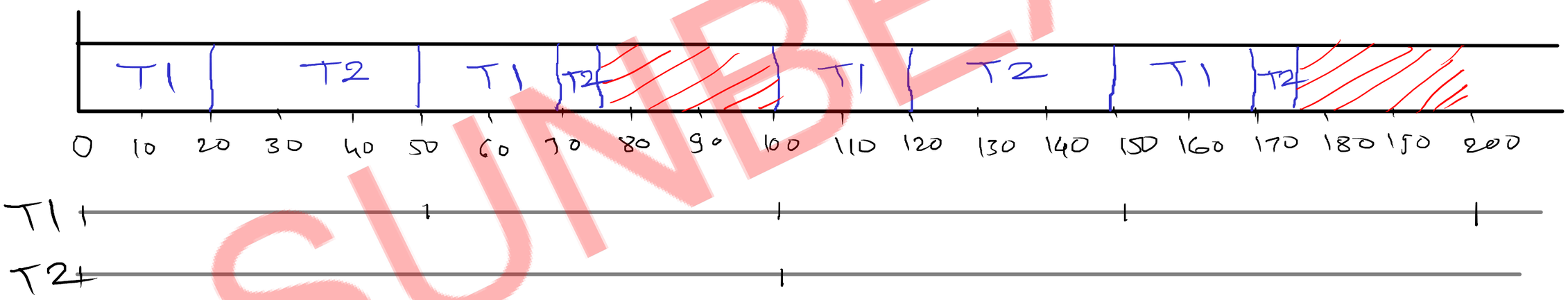
	CPU Burst	Deadline	Period
T1	20	50	50
T2	35	100	100

Hyper period = LCM (P1, P2 ....)
= LCM (50, 100)
= 100

Release point/ phase = 0

priority $\propto \frac{1}{\text{period}}$

T1 > T2 — Fixed/ static



$$\text{CPU utilization} = \frac{75}{100} = 0.75 = 75\%$$

$$\text{CPU Utilization} = \frac{T1}{P1} + \frac{T2}{P2} = \frac{20}{50} + \frac{35}{100}$$

$$= \frac{40+35}{100} = 0.75 = 75\%$$

SUNBEAM

RMA

CPU	Burst	Deadline	Period
P1	20	100	100
P2	30	150	150
P3	90	200	200

$$\text{CPU Utilization} = \frac{20}{100} + \frac{30}{150} + \frac{90}{200}$$

$$= \frac{120+120+270}{600}$$

$$= \frac{510}{600} = 0.85$$

$$\text{Max CPU utilization} = 3*(2^{1/3}-1)$$

$$= 0.78$$

CPU utilization > Max CPU utilization

Above task are not schedulable
According to this test

Liu & Layland test

$$\text{CPU Utilization} = \frac{T1}{P1} + \frac{T2}{P2} \ldots\ldots$$

$$\text{Max CPU utilization} = n*(2^{1/n}-1)$$

where n - no. of task

if CPU utilization > Max CPU utilization

then task are not schedulable

SUNBEAM