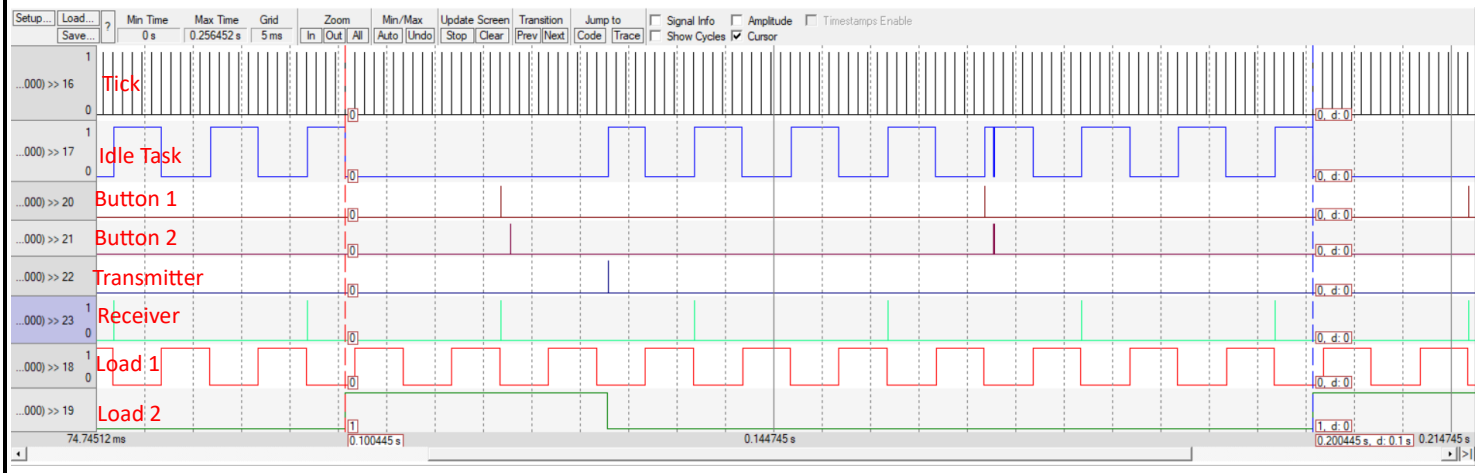
EDF Schedular Report

❖ **Given :**

- Button1_Task (P = 50 , D = 50)
- Button2_Task (P = 50 , D = 50)
- Transmitter_Task (P = 100 , D = 100)
- Receiver_Task (P = 20 , D = 20)
- Load1_Task (P = 10 , D = 10, ExT =5ms)
- Load2_Task (P = 100 , D = 100,ExT=12ms)



➢ **Verifying the system implementation :**

1- Calculate the system hyperperiod : Hp = 100 ms

2- Calculate the CPU load : 100% - (percentage of time the CPU spends in the Idle Task)

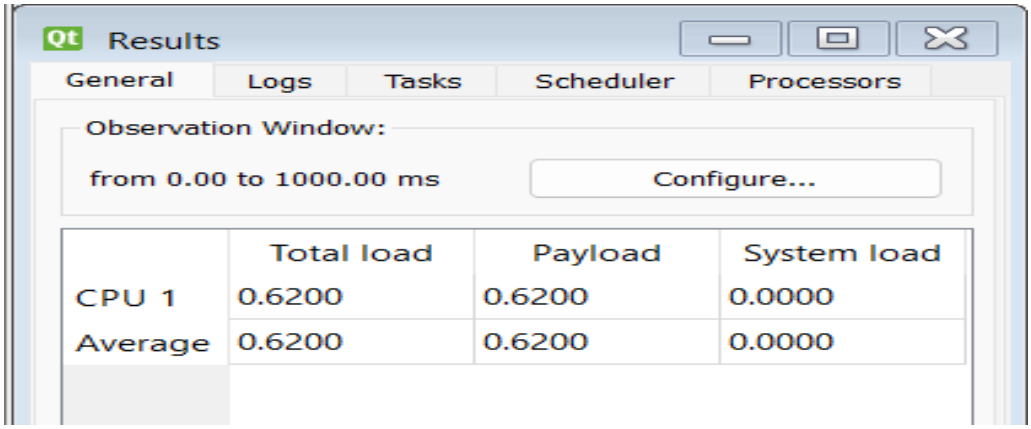
U = R/C

R : Busy time per hyperperiod

C = hyperperiod time

$U = ((10*5)_{Load\ 1} + (12)_{Load\ 2})/100 = 0.62 = 62\%$

- **SIMSO results :**



3- **Check system schedulability using URM and time demand analysis techniques (Assuming the given set of tasks are scheduled using a fixed priority rate -monotonic scheduler):**

URM = $\sum \frac{Ci}{Pi} \leq n(2^{\frac{1}{n}} - 1)$

URM : total CPU utilization

C: Execution time for each task

P: periodicity of each task

n: No of Tasks

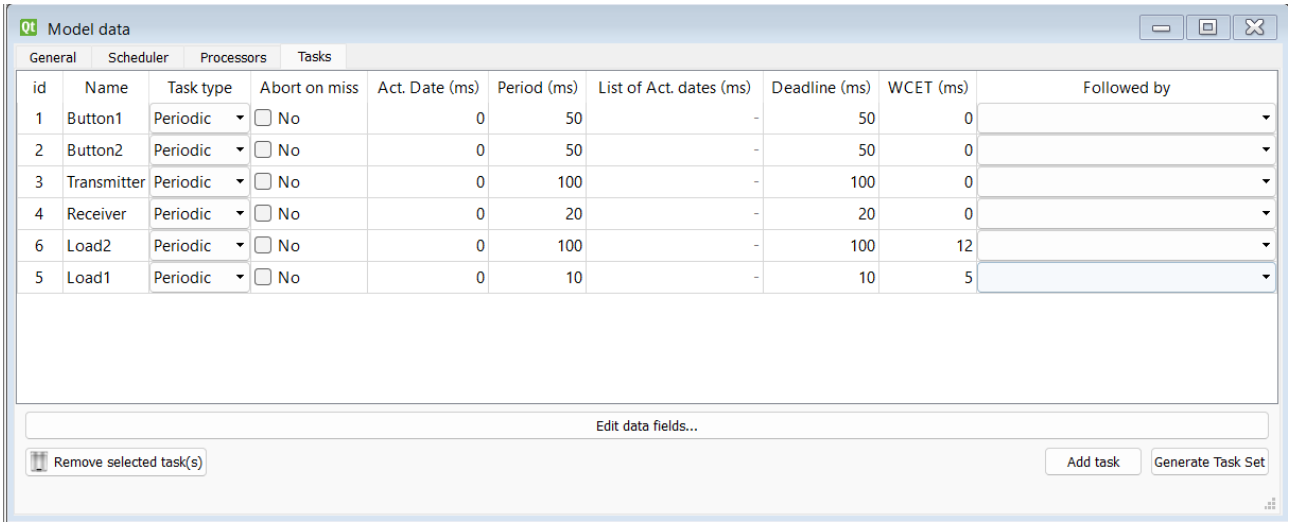
URM = ((5/10)+(12/100)) = 0.62

$n(2^{\frac{1}{n}} - 1)$ **= 6*(2^(1/6)-1) = 0.73**

URM < $n(2^{\frac{1}{n}} - 1)$ **------- > then the system is schedulable**

- **Using Simso offline simulator, simulate the given set of tasks :**

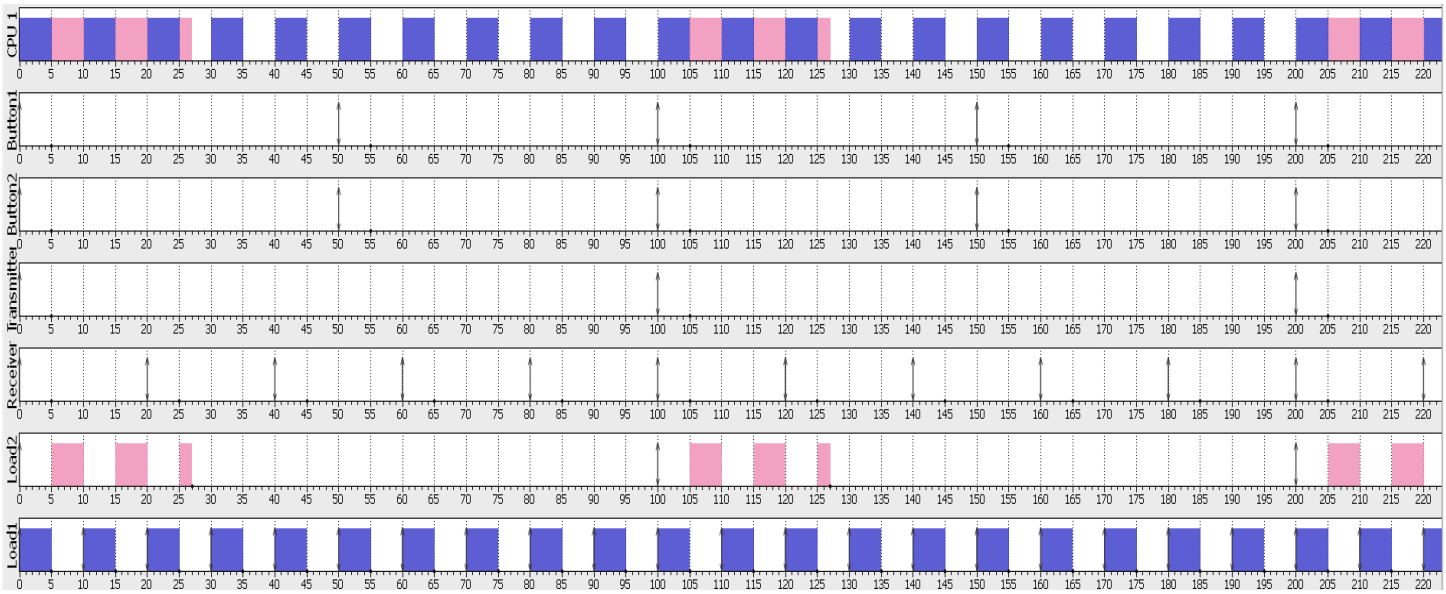
1- **Adding the tasks to Simso, Noting that the (Button1 , Button2 , transmitter , Receiver) tasks all have a zero execution time as they are by the micro seconds so it can be neglected**



id	Name	Task type	Abort on miss	Act. Date (ms)	Period (ms)	List of Act. dates (ms)	Deadline (ms)	WCET (ms)	Followed by
1	Button1	Periodic	No	0	50	-	50	0	
2	Button2	Periodic	No	0	50	-	50	0	
3	Transmitter	Periodic	No	0	100	-	100	0	
4	Receiver	Periodic	No	0	20	-	20	0	
6	Load2	Periodic	No	0	100	-	100	12	
5	Load1	Periodic	No	0	10	-	10	5	

2- Simso Tasks Plot :

- First line is the CPU utilization
- Second line is the Button1 task comes every 50 ms and never misses the deadline
- third line is the Button2 task comes every 50 ms and never misses the deadline
- fourth line is the Transmitter task comes every 100 ms and never misses the deadline
- fifth line is the Receiver task comes every 20 ms and never misses the deadline
- sixth line is the Load 2 task comes every 100 ms and executes within 12 ms and gets preempted two times by the Load 1 task
- seventh line is the Load 1 task comes every 10 ms executes within 5 ms



3- The CPU load is consistent with the one we calculated mathematically