

EDF Scheduler Implementation

This Report verifies system implementation with the EDF scheduler using: 1) Analytical methods: System hyper period, CPU load, URM method, Time demand analysis techniques. 2) Simso offline simulator. 3) Keil simulator in run-time. ** Author: Ahmad Akram **

Task	Periodicity(ms)	Deadline(ms)	Execution Time
Button_1_Monitor	50	50	135us
Button_2_Monitor	50	50	138us
Periodic_Transmitter	100	100	183us
Uart_Receiver	20	20	165us
Load_1_Simulation	10	10	5ms
Load_2_Simulation	100	100	12ms

Analytical Methods

- **System Hyper Period**

The smallest interval of time after which the periodic patterns of all the tasks is repeated:
Hyper Period = Least Common Multiple (10, 20, 50, 100) = 100 ms.

- **CPU Load**

(Total time of all tasks / system time = $T_i * P_i$)

CPU Load = $((0.00135 * 2) + (0.00138 * 2) + (0.00183 * 1) + (0.00165 * 5) + (5 * 10) + (12 * 1)) / 100 = 0.62$.

- **URM Method**

$URM = n (2^{(1/n)} - 1) = 6 (2^{(1/6)} - 1) = 0.73477$.

Since Total Utilization (U) \leq Rate-Monotonic utilization bound (URM),
 $(0.62) \leq (0.73477)$, Therefore the system is **schedulable**.

- **Time Demand Analysis**

1st deadline (p = 10): **L1 Task**

$$w(10) = 5,$$

$w(10) < 10$, **L1 Task is schedulable**

2nd deadline (p = 20): **Rx Task**

$$w(20) = 0.00165 + (20/10)5 = 10.00165, w(20) < 20, \text{ Rx Task is schedulable}$$

3rd deadline (p = 50): **B1 Task**

$$w(50) = 0.00135 + (50/20)0.00165 + (50/10)5 = 25.0055.$$

$w(50) < 50$, **B1 Task is schedulable**

3th deadline (p = 50): **B2 Task**

$$w(50) = 0.00138 + 0.00135 + (50/20)0.00165 + (50/10)5 = 25.0068,$$

$w(50) < 50$, **B2 Task is schedulable**

4th deadline (p = 100): **Tx Task**

$$w(100) = 0.00183 + (100/50)0.00138 + (100/50)0.00135 + (100/20)0.00165 + (100/10)5 = 50.0155$$

$w(100) < 100$, **Tx Task is schedulable**

4th deadline (p = 100): **L2 Task**

$$w(100)$$

$$= 12 + 0.00183 + (100/50)0.00138 + (100/50)0.00135 + (100/20)0.00165 + (100/10)5 = 62.0155,$$

$w(100) < 100$, **L2 Task is schedulable**

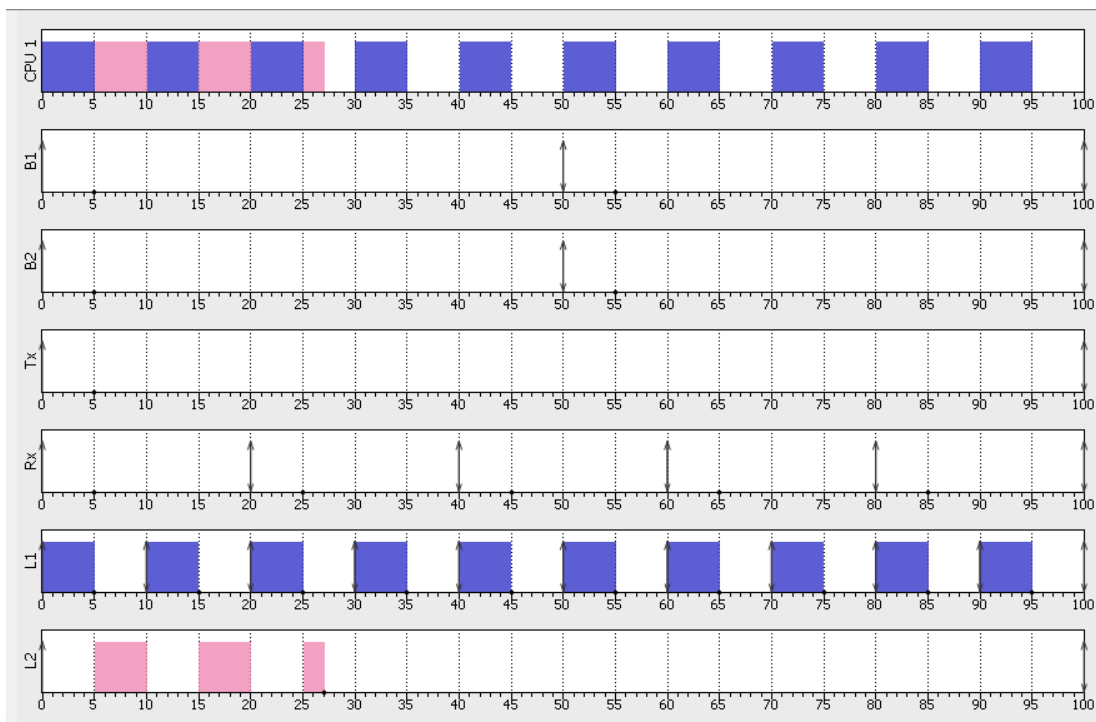
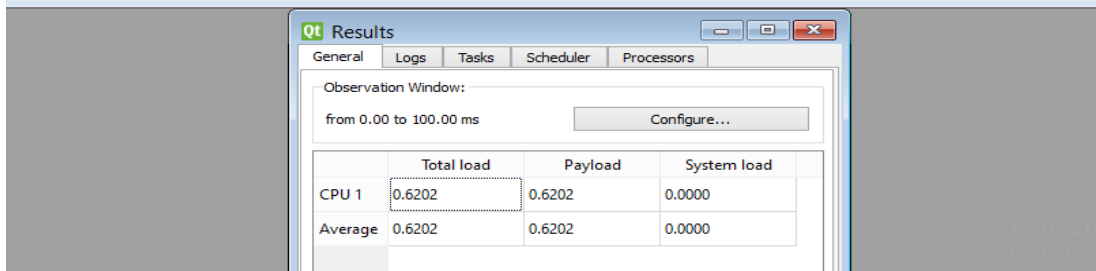
Hence, This system is schedulable.

Simso offline simulator

General Scheduler Processors Tasks										
id	Name	Task type	Abort on miss	Act. Date (ms)	Period (ms)	List of Act. dates (ms)	Deadline (ms)	WCET (ms)	Followed by	priority
1	B1	Periodic	<input type="checkbox"/> No	0.0	50.0	-	50.0	0.00135	▼	1
2	B2	Periodic	<input type="checkbox"/> No	0.0	50.0	-	50.0	0.00138	▼	1
3	Tx	Periodic	<input type="checkbox"/> No	0.0	100.0	-	100.0	0.00183	▼	1
4	Rx	Periodic	<input type="checkbox"/> No	0.0	20.0	-	20.0	0.00165	▼	1
5	L1	Periodic	<input type="checkbox"/> No	0.0	10.0	-	10.0	5.0	▼	1
6	L2	Periodic	<input type="checkbox"/> No	0.0	100.0	-	100.0	12.0	▼	1

Edit data fields...

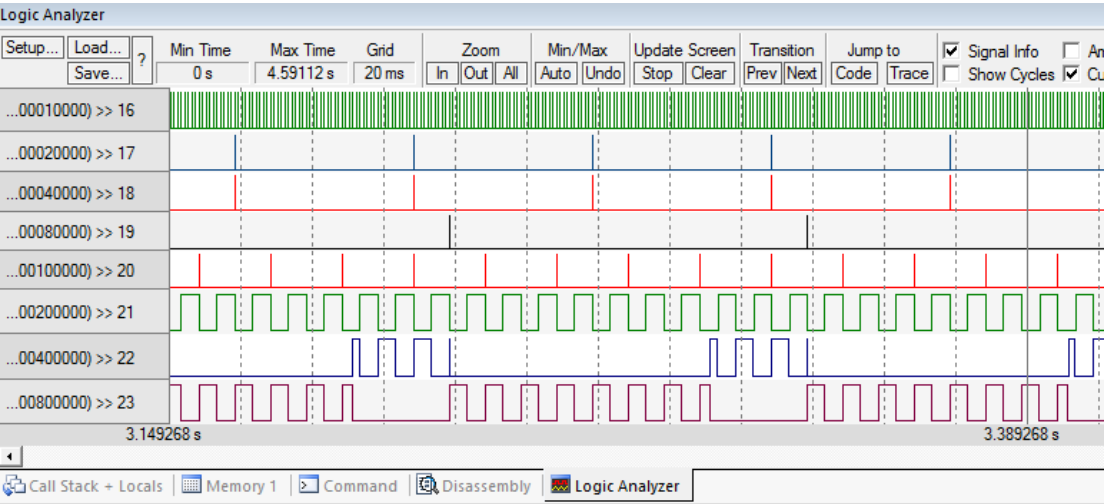
Remove selected task(s) Add task Generate Task Set



KEIL simulator

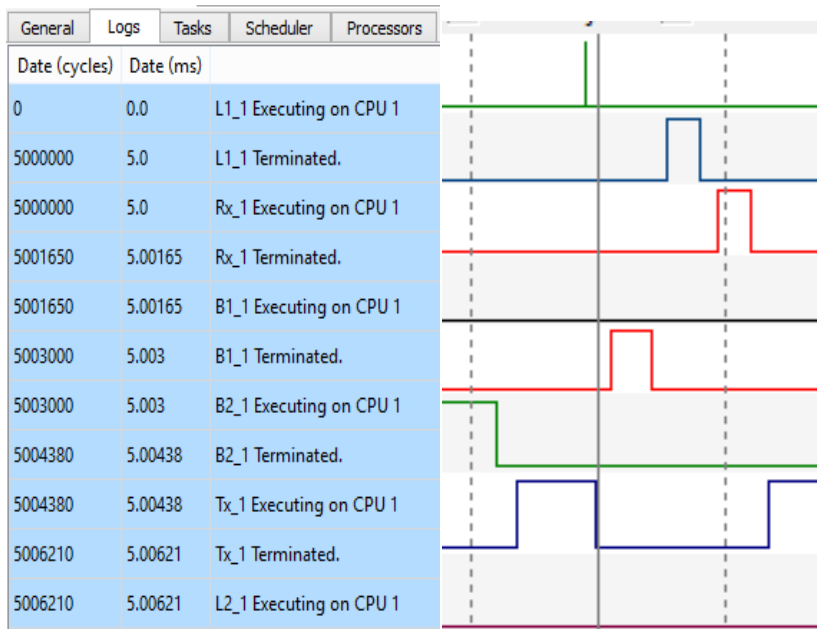
Pins Configurations	Task Tracing
PIN0(16)	Tick Hook
PIN1(17)	Button 1 Task
PIN2(18)	Button 2 Task
PIN3(19)	Periodic Transmitter Task
PIN4(20)	UART Receiver Task
PIN5(21)	Load 1 Simulation Task
PIN6(22)	Load 2 Simulation Task
PIN7(23)	Idle Task

Keil Simulation at runtime:



Name	Value	Type
System_Time	500040	int
CPU_Load	62	int
<Enter expression>		

In Conclusion



- I can clearly verify that the EDF has been successfully implemented, as each task executes based on its deadline and preempts currently executing tasks (ex., Rx, B1, and B2 preempt L2 when their deadlines come), and based on my results above, I have proven that the system is completely schedulable.