EDF Scheduler based on FreeRTOS

Analytical Methods

System Hyperperiod

This project contains 6 tasks as follows :

Task	Period	Execution Time
Button_1_Monitor	50ms	13us
Button_2_Monitor	50ms	13.2us
Periodic_Transmitter	100ms	17.5us
Uart_Receiver	20ms	27.35us
Load_1_Simulation	10ms	5ms
Load_2_Simulation	100ms	12ms

Hyperperiod = 100ms

CPU Load

To calculate the CPU load, we need to calculate the execution time for each task multiplied by number of times these tasks came through one hyperperiod, then by summing these times and divide by the Hyperperiod we get the CPU load.

CPU load =[(13 ∗ 2) + (13.2 ∗ 2) + 17 · 5 + (27.35 ∗ 5)] ∗ 10−3 + (5 ∗ 10) + 12]/100
= 62.2 %

System Schedulability

- Rate-Monotonic utilization bound (only for RM Schedulers)
- Time demand analysis

Rate Monotonic Utilization Bound

$$U = \sum_{i=1}^{n} \frac{C_i}{P_i} \leq n(2^{\frac{1}{n}} - 1)$$

U = Total Utilization
C = Execution time
P = Periodicity
N = Number of tasks

The right-hand side :
(0.013/50) + (0.0132/50)+(0.0175/100)+(0.02735/20)+(5/10)+(12/100) = 0.622
The left-hand side (URM) :
6*(2^1/6 -1) = 0.73477

System is Schedulable **(0.622 ≤ 0.73477)**

Time Demand Analysis

This method measures the time required against the time providedfor each task

$$w_i(t) = e_i + \sum_{k=1}^{i-1} \left\lceil \frac{t}{p_k} \right\rceil e_k \quad for\ 0 < t \le p_i$$

W = Worst response time
E = Execution time
P = Periodicity
T = Time instance

tasks with higher periodicity (comes faster) take higherpriorities

Priority	Task	Period	Execution Time
0	Load_1_Simulation	10ms	5ms
1	Uart_Receiver	20ms	27.35us
2	Button_1_Monitor	50ms	13us
2	Button_2_Monitor	50ms	13.2us
3	Periodic_Transmitter	100ms	17.5us
3	Load_2_Simulation	100ms	12ms

Using the above equation, let's calculate response time for each task taking into consideration the effect of other tasks if they havehigher priority

Load_1_Simulation :
W(10) = 5 + 0 = 5ms ≤ 5ms **Schedulable**

Uart_Receiver :
W(20) = $27.35*10^{-3}$ + 5*(20/10) = 10.027ms ≤ 20ms **Schedulable**

Button_1_Monitor :
W(50) = $13*10^{-3}$ + 27.35*(50/20) + 5*(50/10) = 25.095ms <50ms **Schedulable**

Button_2_Monitor :
W(50) = $2 * 13 * 10^{-3}$+ $27.35*10^{-3} *$ + 5*(^{50}) = 25.108ms <50ms **Schedulable**

Periodic_Transmitter :
W(100) = ($17.5 * 10^{-3}$) + ($2 * 13 * 10^{-3}$
*100/50) ($27.35*10^{-3}$*100/20)+(5*100/10) =
50.188ms <100ms **Schedulable**

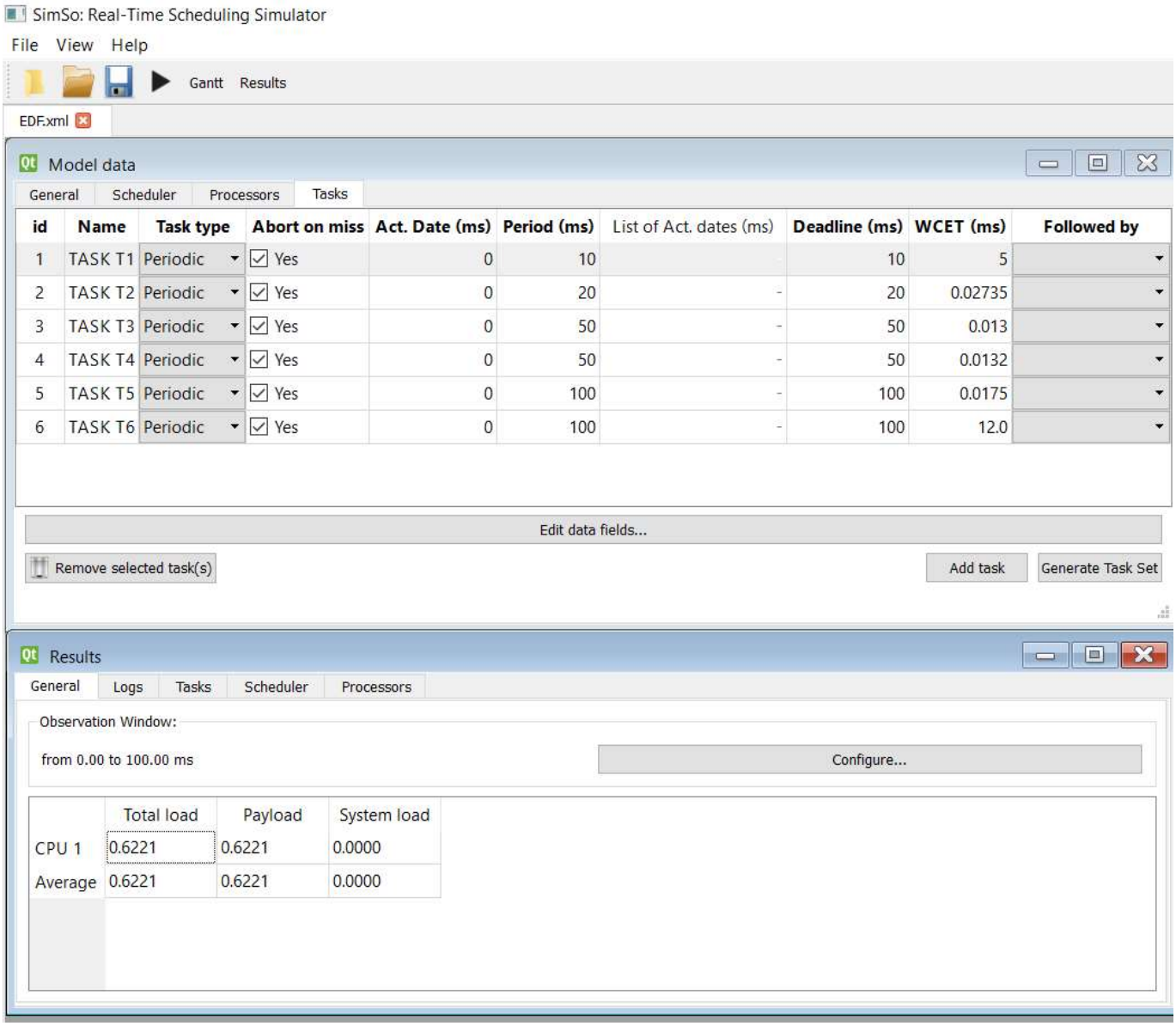
Load_2_Simulation :

$W(100) = 12 + (17.5 * 10^{-3}) + (2 * 13 * 10^{-3}$
$*100/50)\ \ (27.35*10^{-3}*100/20)+(5*100/10)$

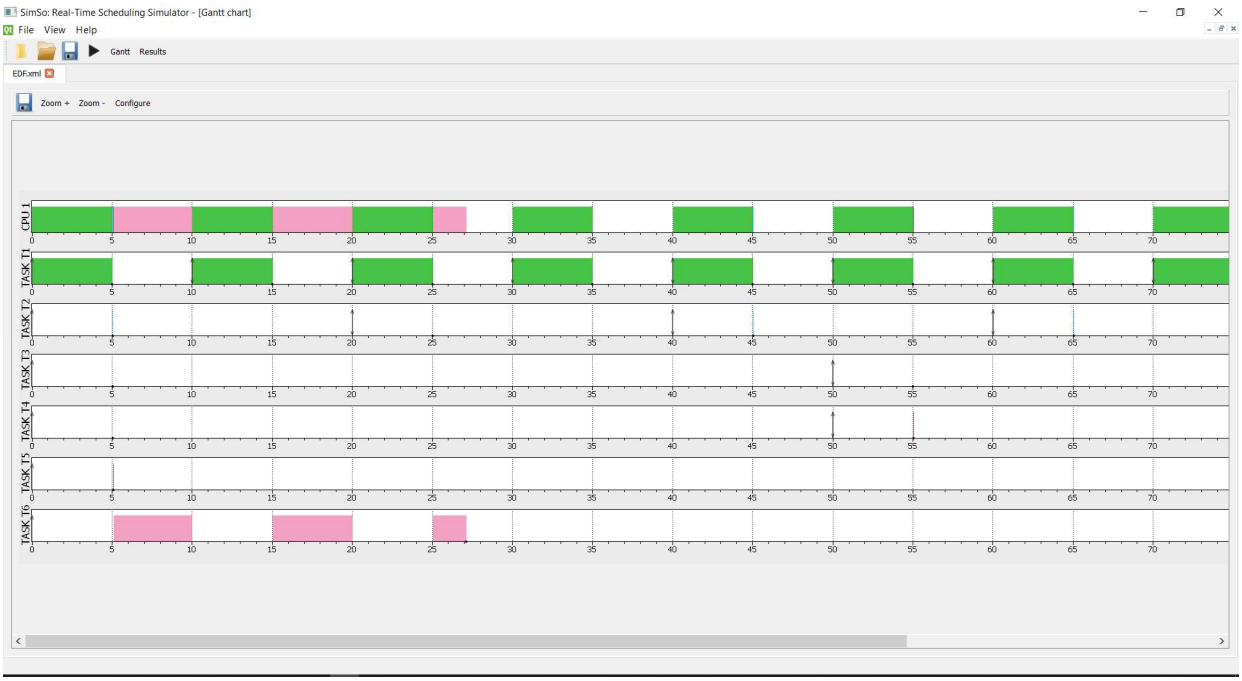
= 62.206ms < 100ms **Schedulable**

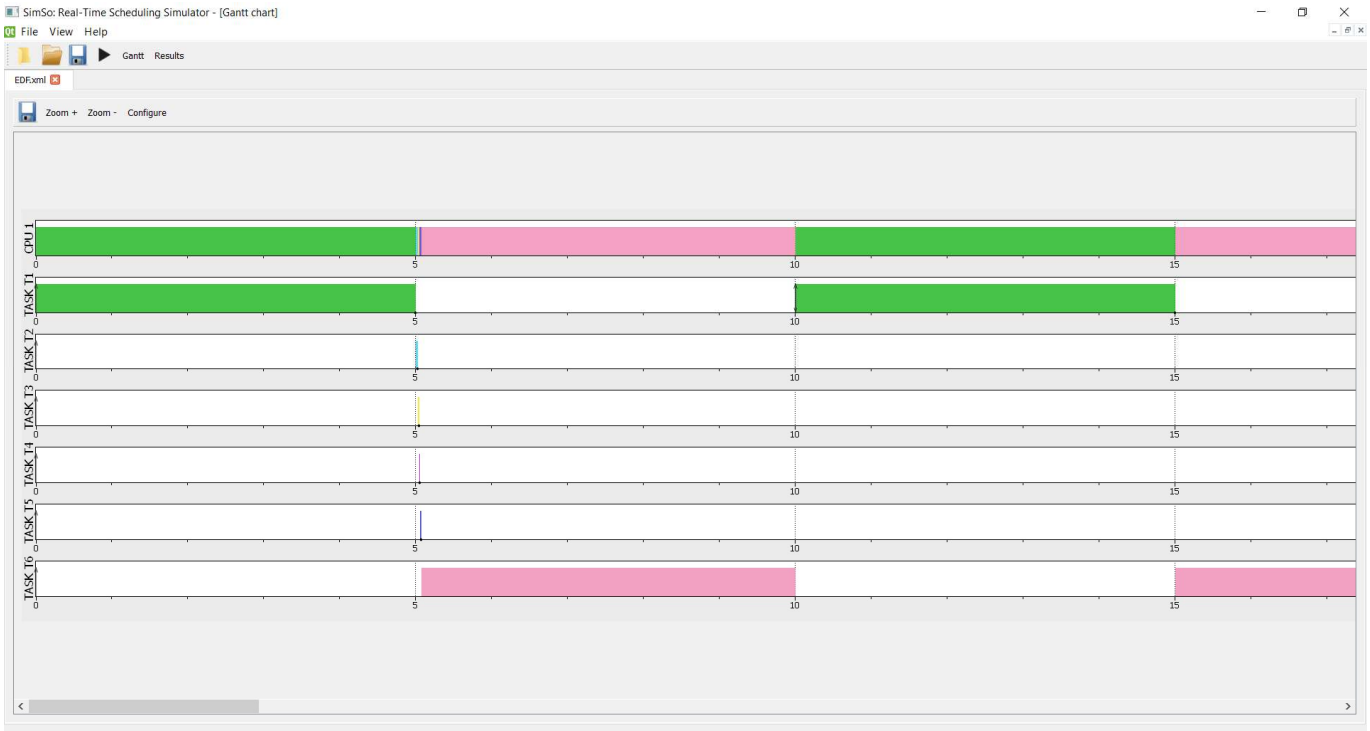
System is Schedulable

SIMSO Offline Results



As we see that CPU load = 62.2% which similar to analyticalapproach

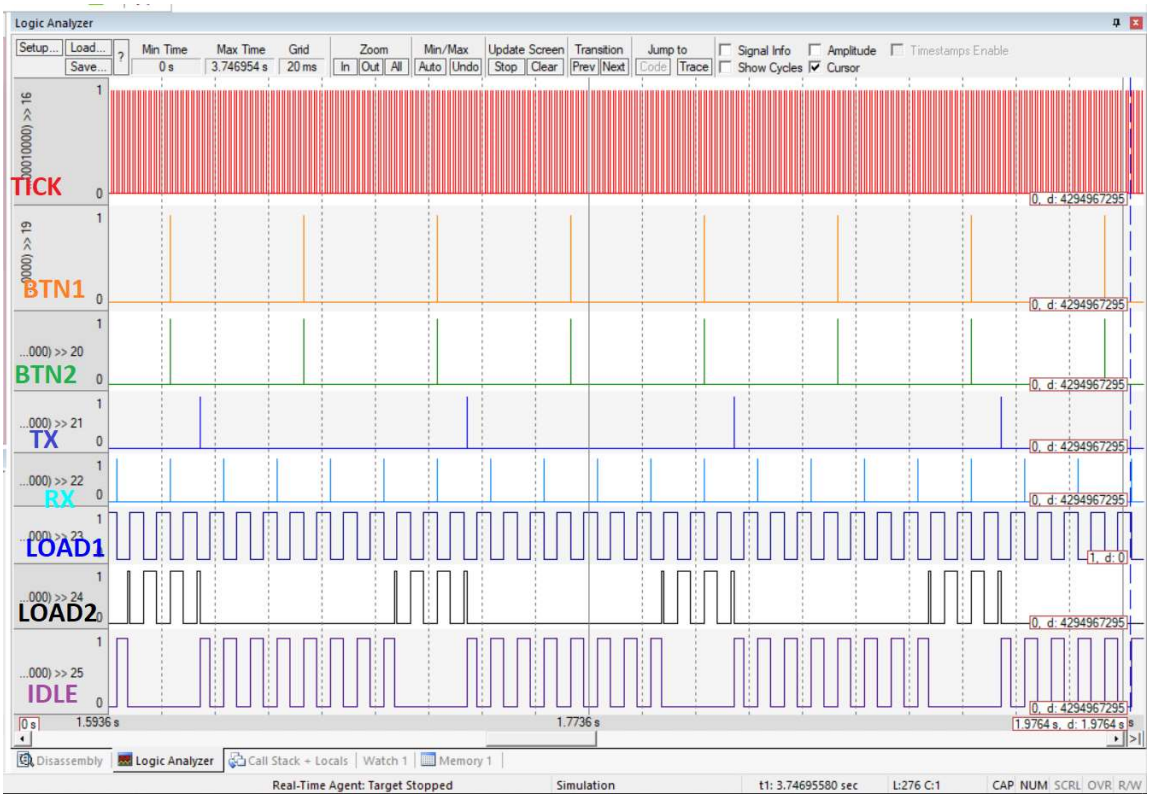


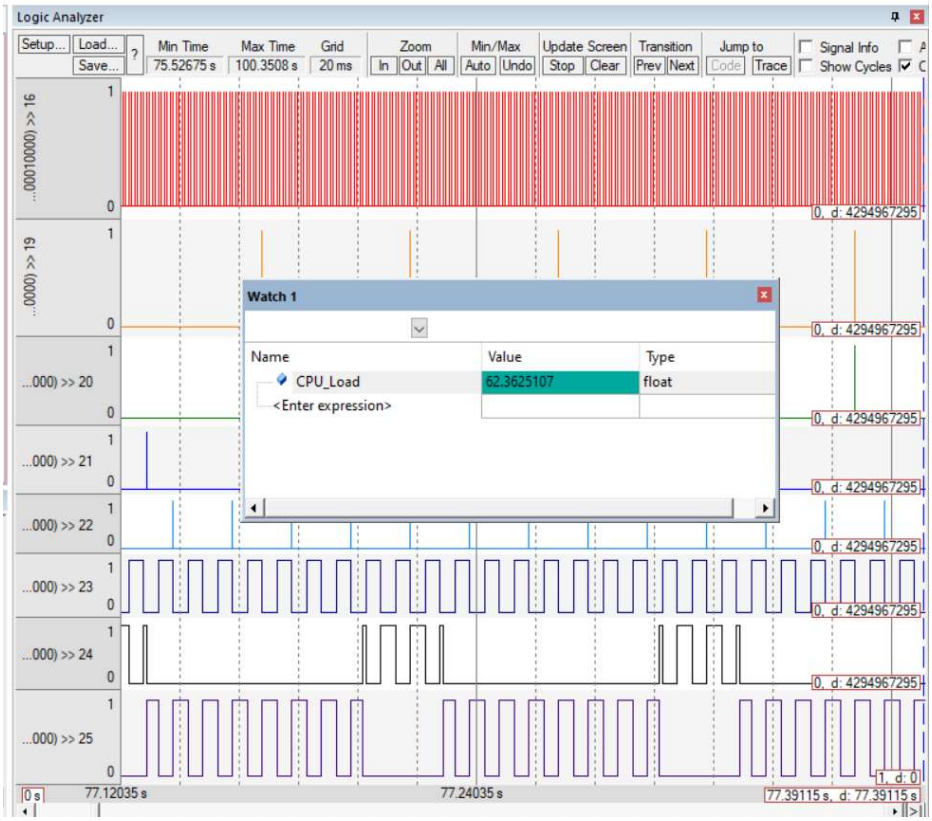


From above screenshots using simso, and the above results of analytical approach, we found that both approaches lead to same results and thereis a matching between them.

Keil Simulator Results

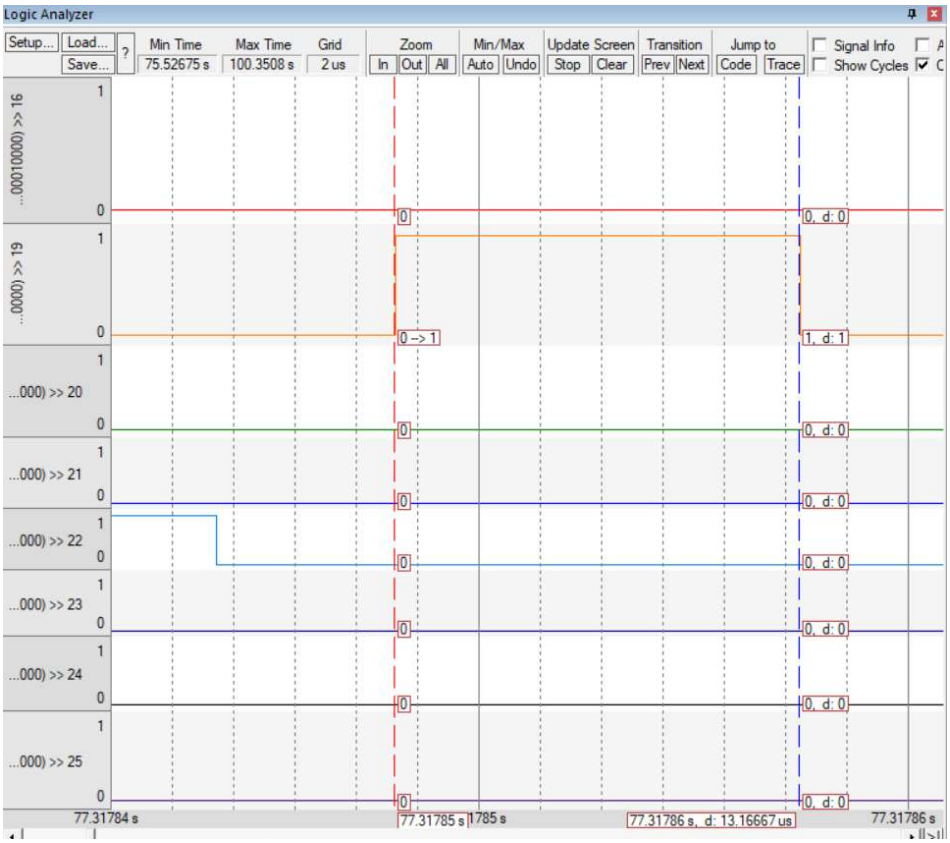
1.



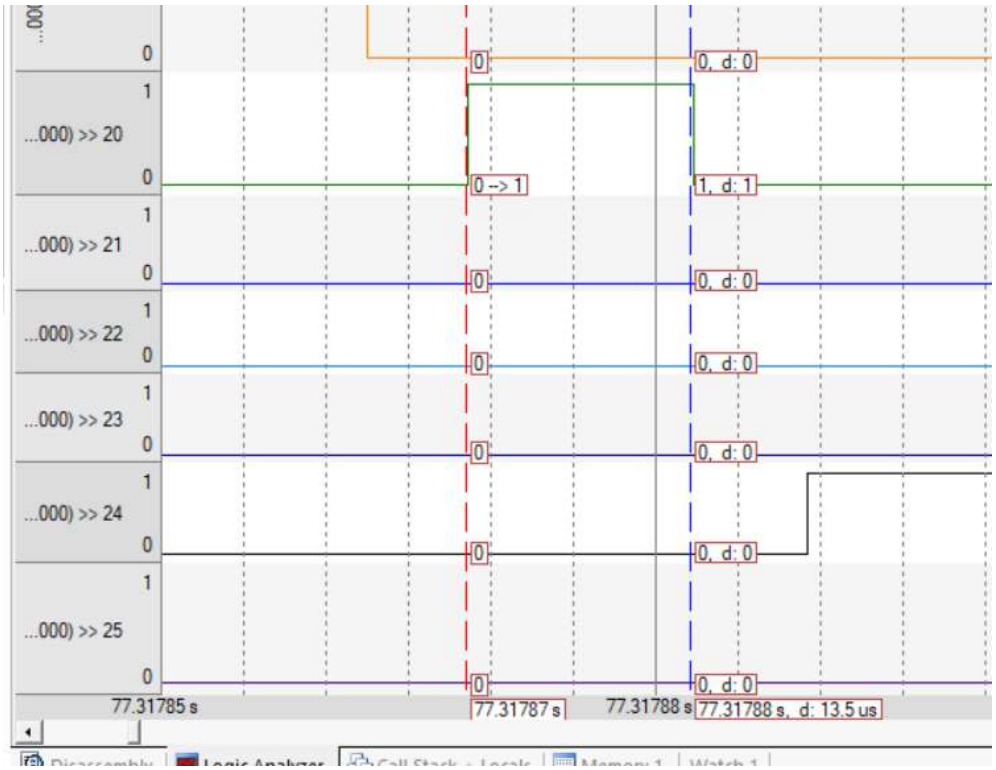


Here we can see that CPU_Load is about 62.36% This result matches our previous two approaches (analytical and simso simulator

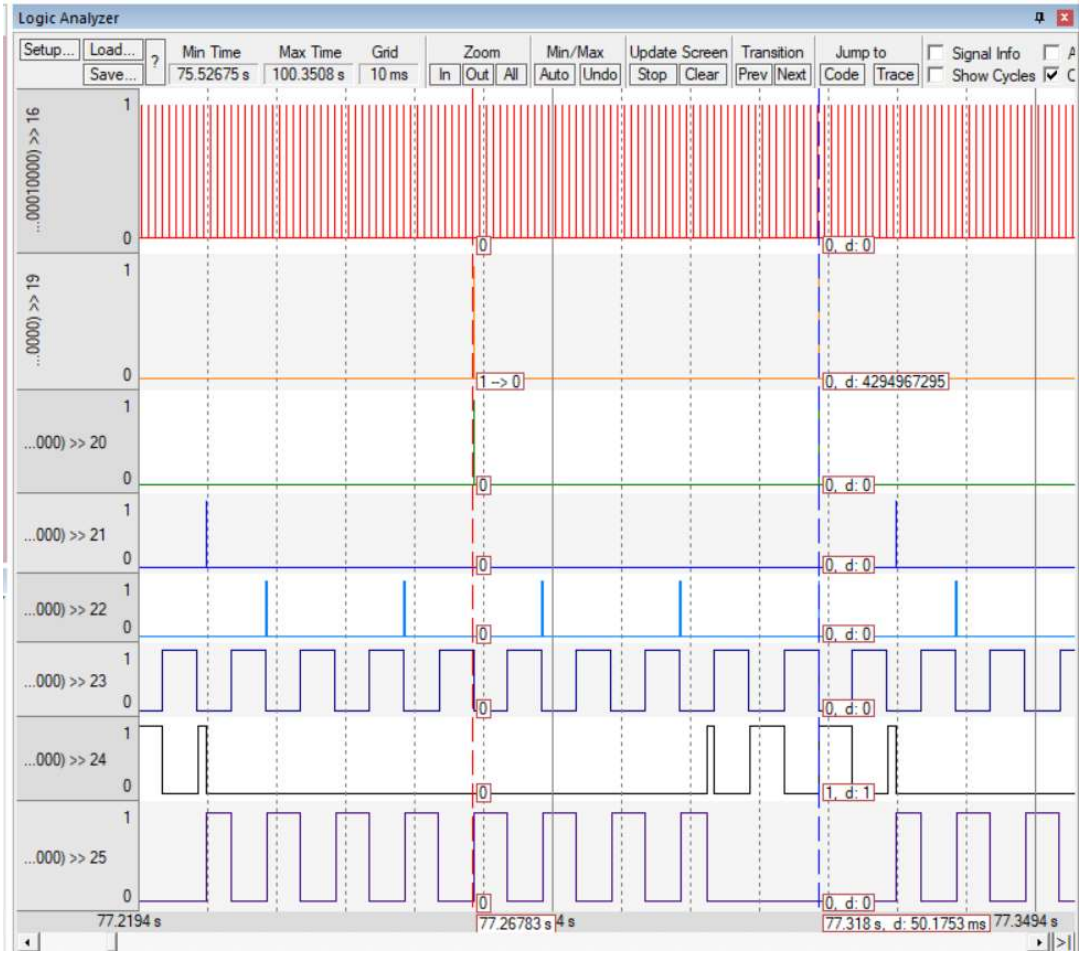
2. execution time of Button_1_Monitor task (13us)



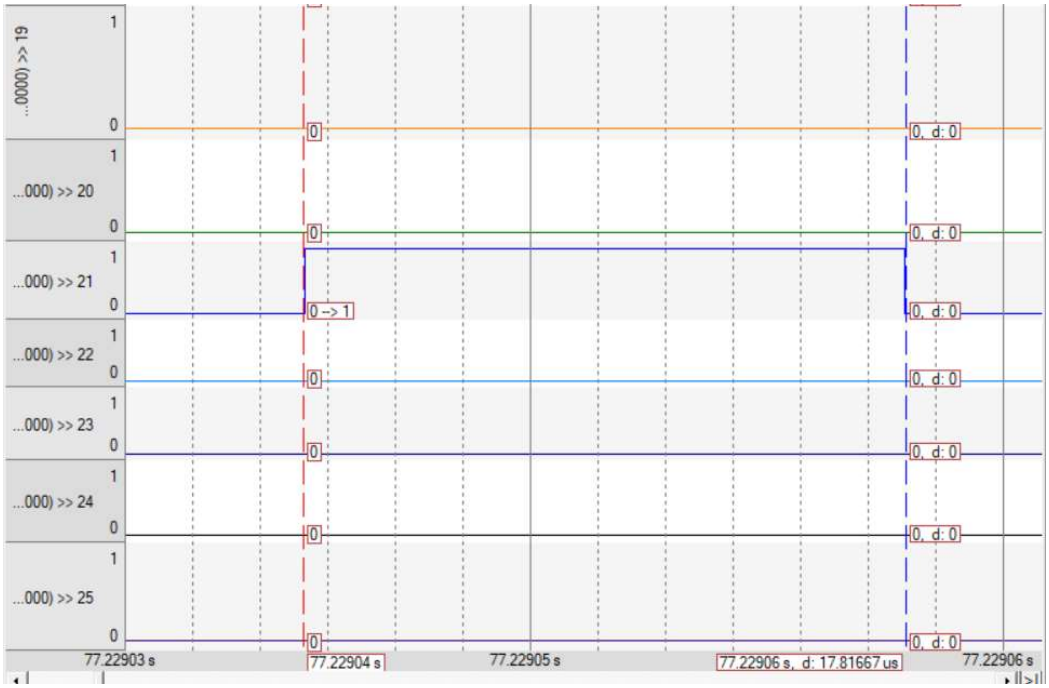
3. execution time of Button_2_Monitor task (13us).



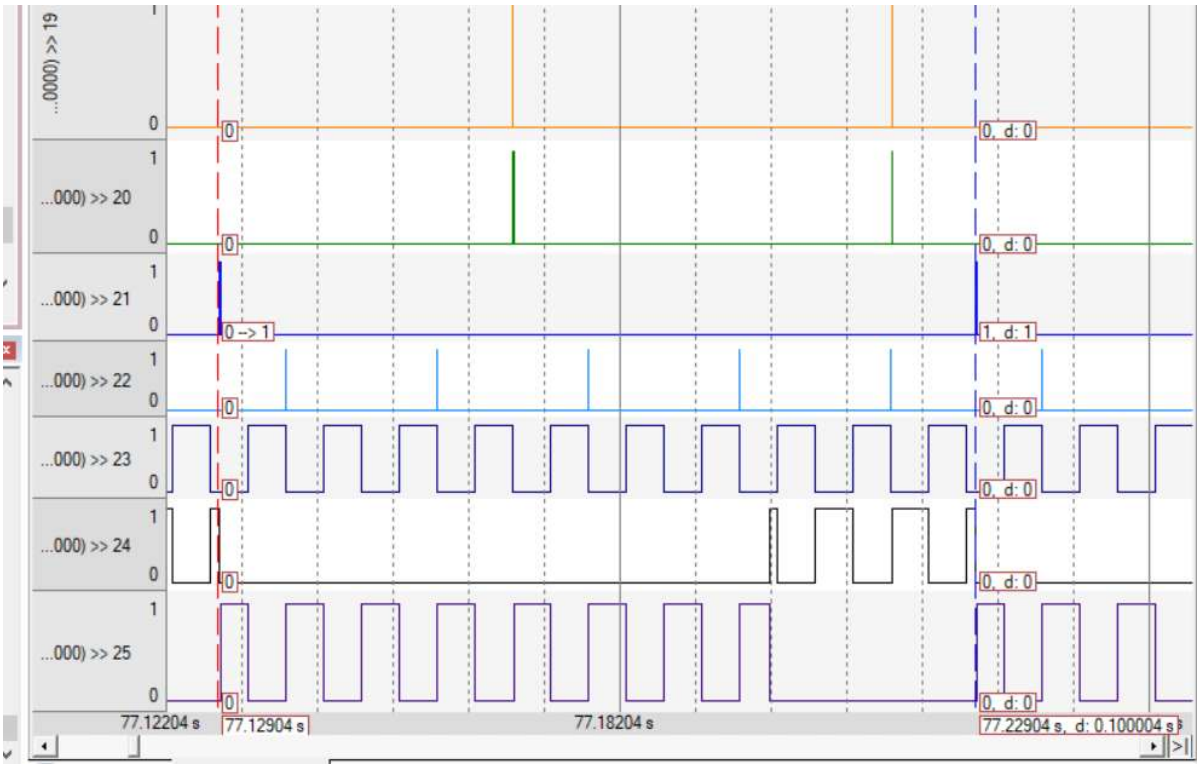
4.Button_1_Monitor and Button_2_Monitor Task periodicity = 50ms



5. execution time of Periodic_Transmitter task (17.5 us).

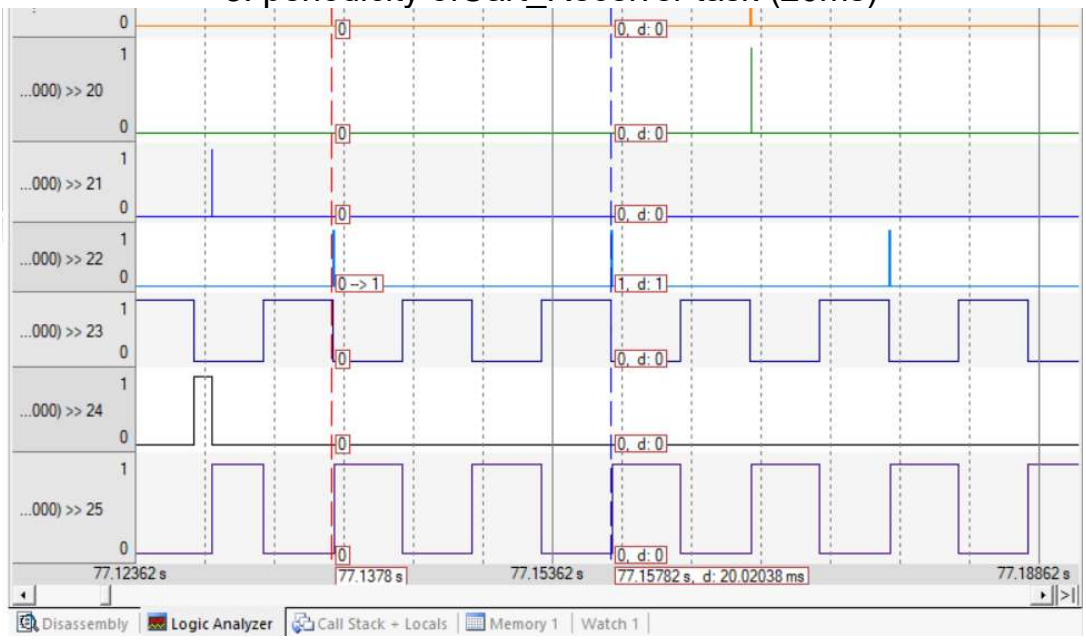


6. periodicity ofPeriodic_Transmitter task (100ms)



7. execution time ofUart_Receiver task (27.5us)

8. periodicity ofUart_Receiver task (20ms)



9. execution time ofLoad_1_Simulation task in run time (5ms).

10. periodicity ofLoad_1_Simulation task (10ms)

11.This case shows that Load_2_Simulation task is preempted by theabove task (Load_2_Simulation) three times, so total execution time for Load_2_Simulation task = 27 – (3*5) = 12ms

12. periodicity of Load_2_Simulation task (100ms) .