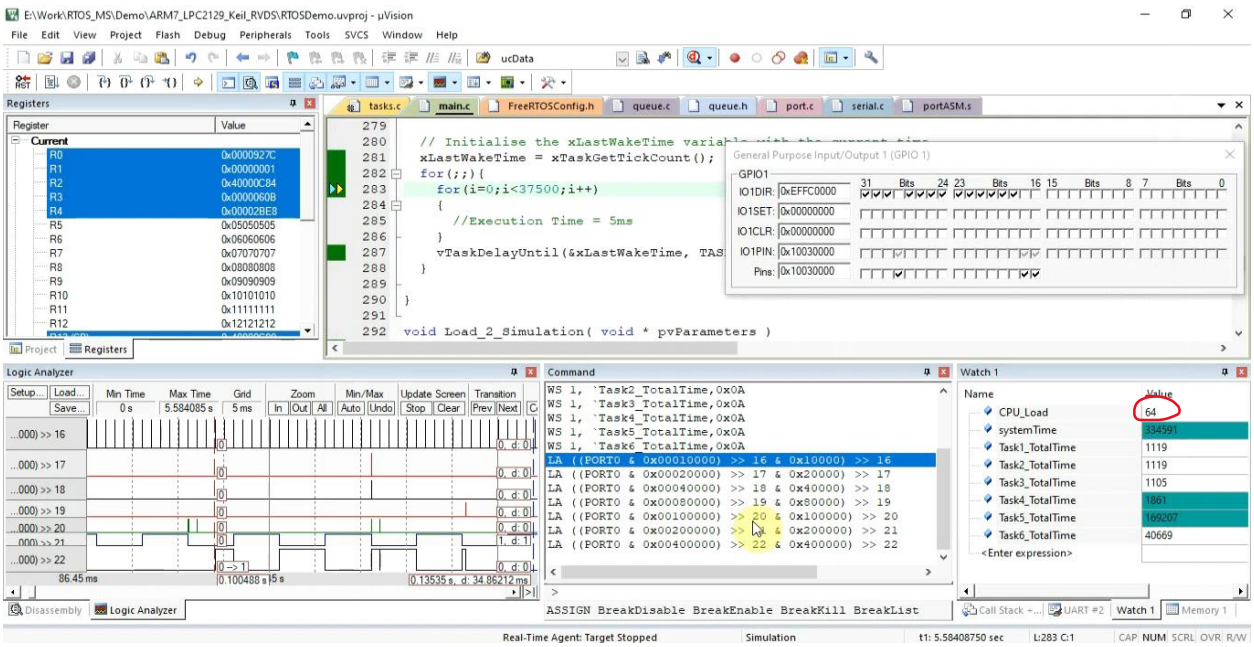
Report of Implementing FreeRTOS EDF Scheduler

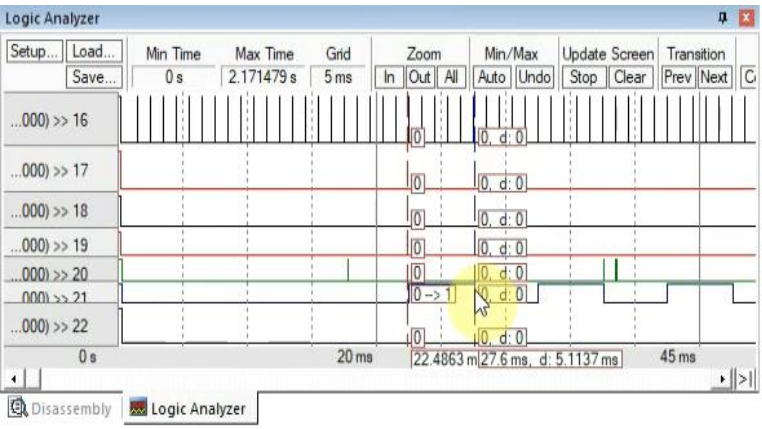
- **verification methods**

1- Keil uVision Logic Analyzer View

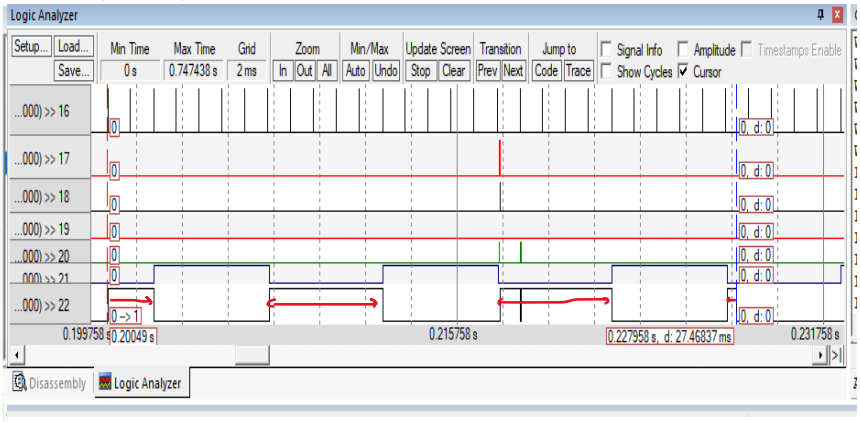


- Tasks Execution Time:

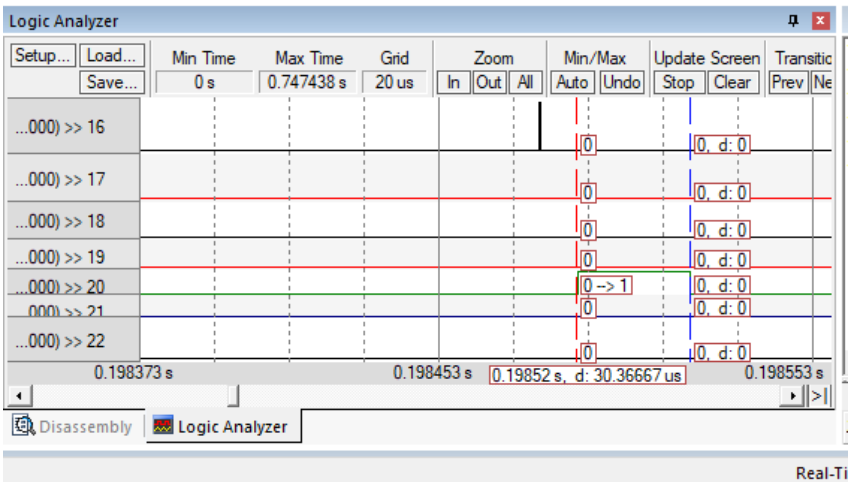
-Task 5 (Load 1) : Execution time = 5ms



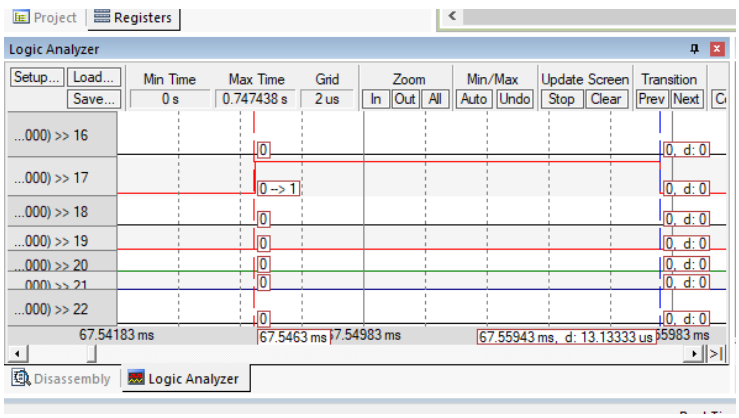
-Task 6 (load 2) : Execution time = 2+5+4.5+0.5 = 12ms

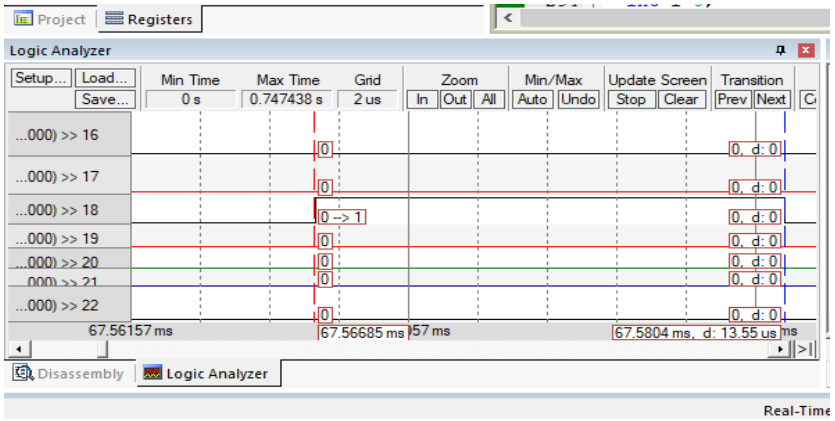


Task 4 (UART_Receiver): Execution time = 30 us

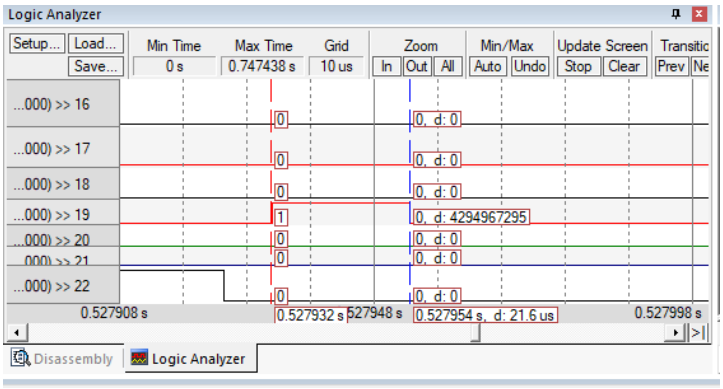


Task 1&2 (Button_Monitor): Execution time = 13 us

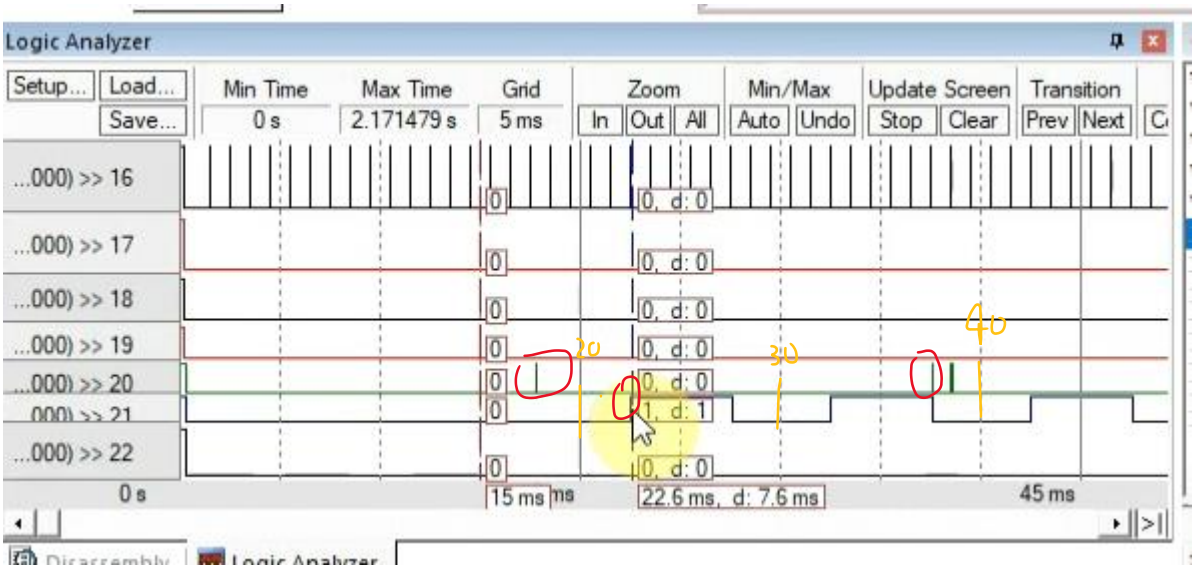




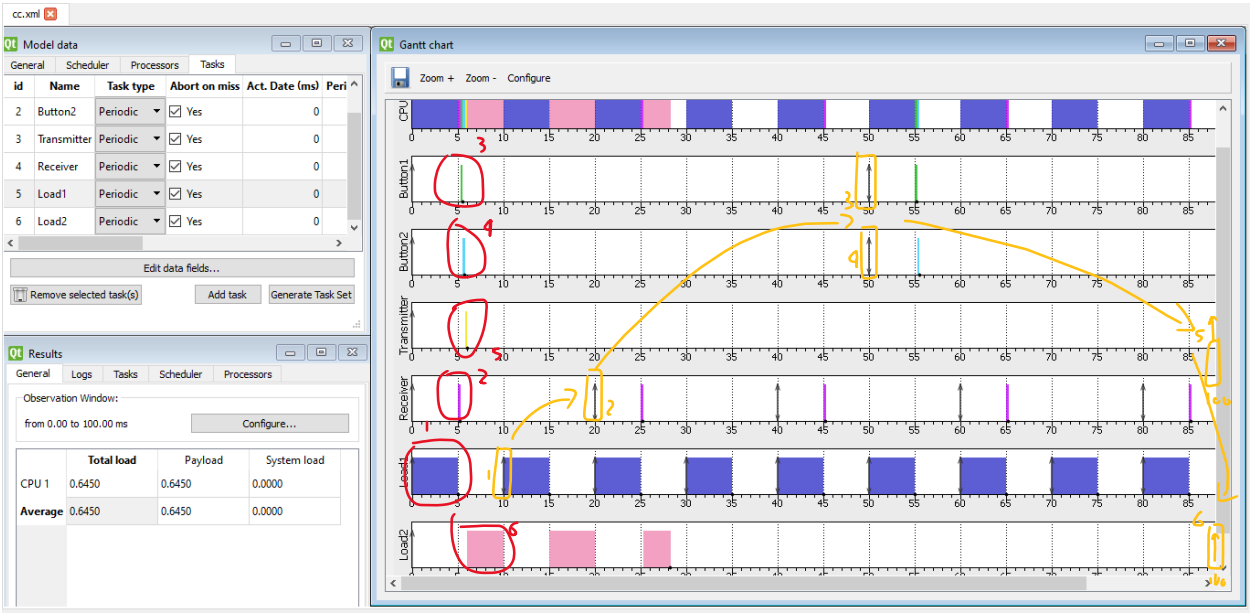
Task 3 (Transmitter): Execution time = 21 us



- How EDF Works well:
  - UART Task executed first because its deadline was first(20ms) until Load 1 scheduled so it has executed earlier since its deadline (10ms) , then UART Task executed .
  - All Tasks spends its execution Time before deadline so schedulability is achieved.



2- Using Simso offline simulator



CPU Load = 64% almost the same percentage as the real schedular performance of keil.

- from Gantt Chart we conclude the mentioned points in keil simulator,

the most important conclusion is: the earlier the deadline, the earlier execution of this task happens.

- System Calculations

1- Hyper period of a set of periodic tasks is **the least common multiple of periods of all the tasks in that set**.
   - H = lcm(p1, p2, p3, p4, p5, p6) = lcm(50, 50, 100, 20, 10, 100) = 100 ms

2- CPU usage is a measurement, in a percentage, of **how much time the CPU spends actively computing something**.

   - **Total CPU load = summation of(Task's Frequency * Task's worst case execution time)**.
   - U =((12+10*5+4*0.13+0.21+5*0.30)/100)*100= 64.23 %
     - ➢ almost the value of CPU Load simulated by Keil and Simso

3- System schedulability:
   Using time demand analysis

- Equation

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

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

Calculate time demand for Tasks:

- **Task 5: "Load_1_Simulation", {Periodicity: 10, Deadline: 10, ET: 5ms}**

  $W_1(10) = 5+(10/10)*0 = 5 < 10$ >> **Load 1 Simulation** is schedulable

- **Task 4: "Uart_Receiver", {Periodicity: 20, Deadline: 20, ET: 30us}**

  $W_2(20) = 0.30+(20/10)*5 = 10.3 < 20$ >> **Uart Receiver** is schedulable

- **Task 1: " Button_1_Monitor", {Periodicity: 50, Deadline: 50, ET: 13us}**

  $W_3(50) = 0.13+(50/20)*0.30 +(50/10)*5 = 25.88 < 50$ >> **Button 1 Monitor** is schedulable

- **Task 2: " Button_2_Monitor", {Periodicity: 50, Deadline: 50, ET: 13us}**

  $W_4(50) = 0.13+(50/20)*0.30 +(50/10)*5+0.13 = 26.01 < 50$ >> **Button 2 Monitor** is schedulable

- **Task 2: "Periodic_Transmitter", {Periodicity: 100, Deadline: 100, ET: 21us}**

  $W_5(100) = 0.21+(100/20)*0.30 +(100/10)*5+2*(100/50)0.13 = 56.03 < 100$

  >> **Periodic Transmitter** is schedulable

- **Task 6: "Load_2_Simulation", {Periodicity: 100, Deadline: 100, ET: 12ms}**

  $W_6(100) = 12+(100/20)*0.30 +(100/10)*5+2*(100/50)0.13+0.21 = 68.03 < 100$

  >> **Load 2 Simulation** is schedulable

>>>> All Tasks is schedulable -----> So, The system is schedulable

Using URM

- Equation:

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

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

$U_{rm} = 6*(2^{1/6} - 1) = 0.73$, $U = 64.32\% = 0.64$

$\therefore U < U_{rm}$ -----> so, The system is schedulable