System Overview

Task Number	Task Name	Task Periodicity (ms)	Task Deadline (ms)	Task Execution Time (ms)
Task_1	Button_1_Monitor	50	50	0.01255
Task_2	Button_2_Monitor	50	50	0.01255
Task_3	Periodic_Transmitter	100	100	0.025
Task_4	Uart_Receiver	20	20	0.0491
Task_5	Load_1_Simulation	10	10	5
Task_6	Load_2_Simulation	100	100	12

Calculating System's Hyper Period

System Hyper Period is Least common multiple of individual task periodicities.

Hyper Period = LCM (50, 50, 100, 20, 10, 100) = 100 MS

Calculating System's CPU Load

System CPU Load is the summation of all individual tasks execution time to periodicity's ratio.

$$= \frac{\left(\frac{100}{50}\right) * 0.01255 + \left(\frac{100}{50}\right) * 0.01255 + \left(\frac{100}{100}\right) * 0.025 + \left(\frac{100}{20}\right) * 0.0491 + \left(\frac{100}{10}\right) * 5 + \left(\frac{100}{100}\right) * 12}{100}$$

$$= 0.062307$$

$$= 62.3207\ \%$$

Calculating System's Schedulability

Using Utilization Rate Monotonic Approach

Assuming Rate-Monotonic Scheduler then the system is guaranteed to be scheduled if

$$CPU\ LOAD \leq (2^{\frac{1}{n}} - 1)$$

Since

CPU LOAD = 62.3207 %

And

n = number of tasks = 6

Then

URM (Utilization of rate monotonic) = $6(2^{\frac{1}{6}} - 1)$ = 0.73477 = 73.47%

Since

U < URM

Then

The System is guaranteed to be scheduled

Using Time Demand Analysis Approach

1. Finding the greatest common divisor of the system to calculate the step

GCD (50, 50, 100, 20, 10, 100) = 10 ms

2. Compute parameters

Beginning = Start of a Hyper period
Step = system GCD
Ending = End Hyper period

Beginning	T = 0
Step	10 ms
Ending	100 ms
Number Of Tasks	6

3. Compute The time demand Function $W_i(t)$

$$W_i(t) = E_i + \sum_{k=1}^{i-1} \lceil \frac{t}{P_K} \rceil * E_k$$

I: Task index

P: Task Period/Deadline

E: Execution time

- T1 and T2 checking
  T1: (P=50, C=0. 01255, D=50).
  T2: (P=50, C=0. 01255, D=50).
  As both tasks have the same deadline then we can't predict which one will execute first but as both of them has the same execution time so if the last executed one of them is schedulable whatever each one is the first and each is the last then both of them are schedulable.

- T4 checking
  T4: (P=20, C=0.0491, D=20).
  Time required =20.
  Time provided: $W4(20) = 0.0491 + (20/10) \times 5 = 10.0491 < 20$ (schedulable).

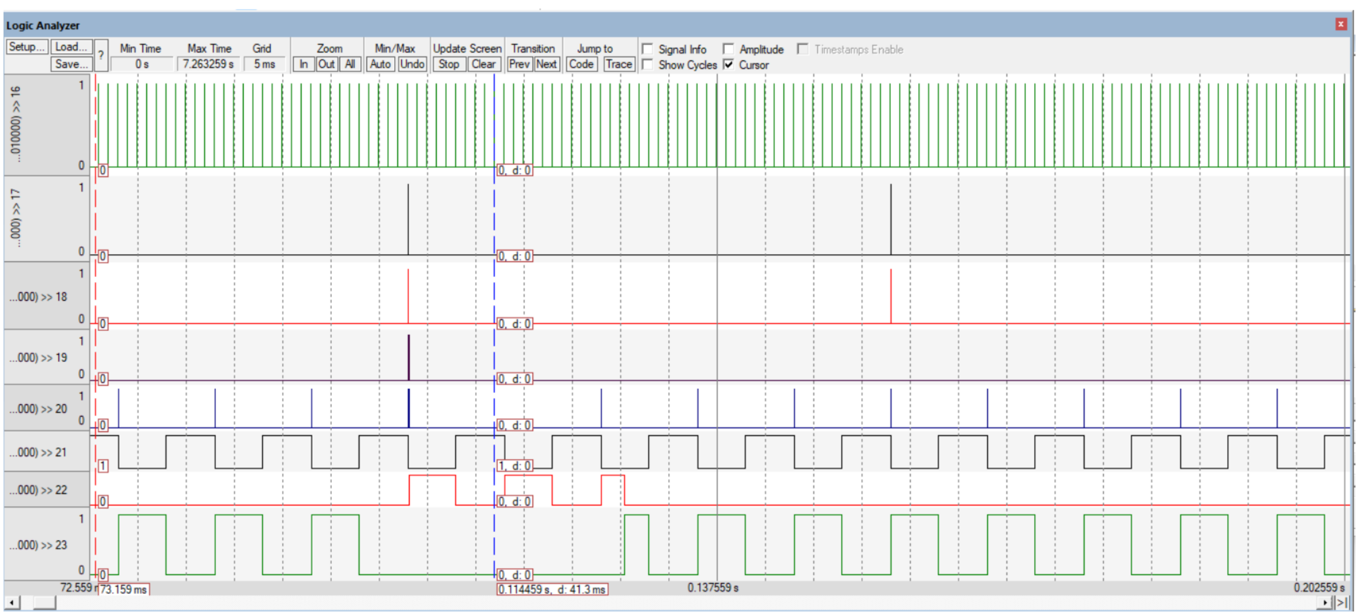
- T5 checking
  T5: (P=10, C=5, D=10).
  Time required = 10.
  Time provided: $W5(10) = 5 + 0 = 5 < 10$ (schedulable).

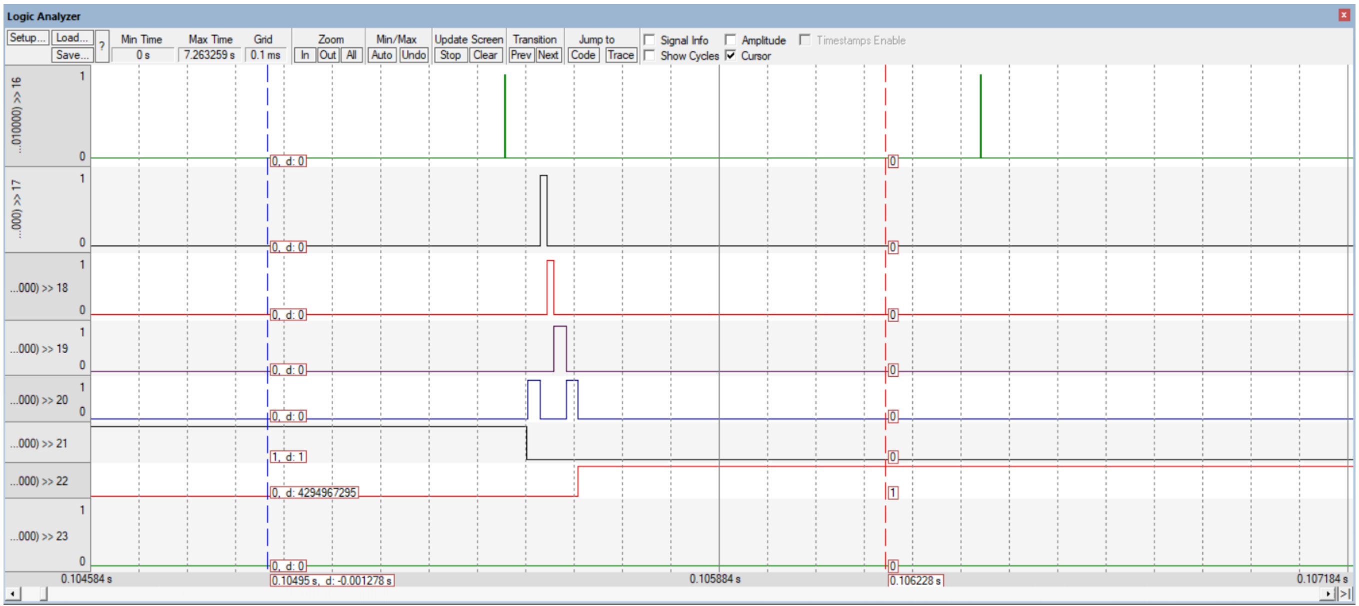
- T3 and T6 checking
  T3: (P=100, C=0.025, D=100).
  T6: (P=100, C=12, D=100).
  As both tasks have the same deadline then we can't predict which one will execute first and the two tasks have different execution times so every task will be treated to be the last executed one as a worst-case scenario.
  $W3(100) = 0.025 + (100/10) * 5 + (100 / 20) * 0.0491 + (100/50) * 0.01255 + (100/50) * 0.01255 + (100/100) * 12 = 62.3207 < 100$ (schedulable).
  $W6(100) = 12 + (100/10) * 5 + (100/20) * 0.0491 + (100/50) * 0.01255 + (100/50) * 0.01255 + (100/100) * 0.025 = 62.3207 < 100$ (schedulable)

Screenshots

SimSo Simulator

Keil uVision





Logic Analyzer
Setup...
Load...
Save...
Min Time
0 s
Max Time
7.263259 s
Grid
0.1 ms
Zoom
In
Out
All
Min/Max
Auto
Undo
Update Screen
Stop
Clear
Transition
Prev
Next
Jump to
Code
Trace
Signal Info
Show Cycles
Amplitude
Cursor
Timestamps Enable
...010000) >> 16
...000) >> 17
...000) >> 18
...000) >> 19
...000) >> 20
...000) >> 21
...000) >> 22
...000) >> 23
0, d: 0
1, d: 1
0, d: 4294967295
0.104584 s
0.10495 s, d: -0.001278 s
0.105884 s
0.106228 s
0.107184 s