EMBEDDED SYSTEMS

ADVANCED COURSE

UDACITY – SPRINTS - EGFWD

IMPLEMENTING EDF SCHEDUELER ON FREERTOS

BY AHMED SAMEH SAAD GOMAA

ahmedsamehsaad99@gmail.com

I. CALCULATING CPU LOAD

Hyper-period is 100 ms

id	Name	Task type	Abort on miss	Act. Date (ms)	Period (ms)	List of Act. dates (ms)	Deadline (ms)	WCET (ms)
1	Button_1	Periodic	No	0	50	-	50	0.016
2	Button_2	Periodic	No	0	50	-	50	0.016
3	Period_Tx	Periodic	No	0	100	-	100	0.016
4	UART_Rx	Periodic	No	0	20	-	20	0.016
5	Load1	Periodic	No	0	10	-	10	5
6	Load2	Periodic	No	0	100	-	100	12

$$\frac{(10*5)+(2*5)+(1*3)+(10*0.016)}{100}*100 = 63.16\ \%$$

II. CHECKING SCHEDULABILITY USING URM

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

$$\frac{0.016}{50}+\frac{0.016}{50}+\frac{0.016}{100}+\frac{0.016}{20}+\frac{5}{10}+\frac{12}{100} = 0.6216$$

$$6\left(2^{1/6} - 1\right) = 0.734$$

The design satisfies the condition and tasks are schedulable

III. CHECKING SCHEDULABILITY USING TIME DEMAND ANALYSIS

Arranging tasks by the least periodicity:

1. **Load 1**

   W(10) = 5 ms < 10ms

2. **UART Rx**

   $W(20) = 0.016 + \frac{20}{10} * 5 = 10.016ms < 20\ ms$

3. **Button 1**

   $W(50) = 0.016 + \frac{50}{20} * 0.016 + \frac{50}{10} * 5 = 25.056ms < 50\ ms$

4. **Button 2**

$$W(50) = 0.016 + \frac{50}{50} * 0.016 + \frac{50}{20} * 0.016 + \frac{50}{10} * 5 = 25.072ms < 50\ ms$$

5. **Periodic Tx**

$$W(100) = 0.016 + \frac{100}{50} * 0.016 + \frac{100}{50} * 0.016 + \frac{100}{20} * 0.016 + \frac{100}{10} * 5 =$$

$$50.128\ ms < 100\ ms$$

6. **Load 2**

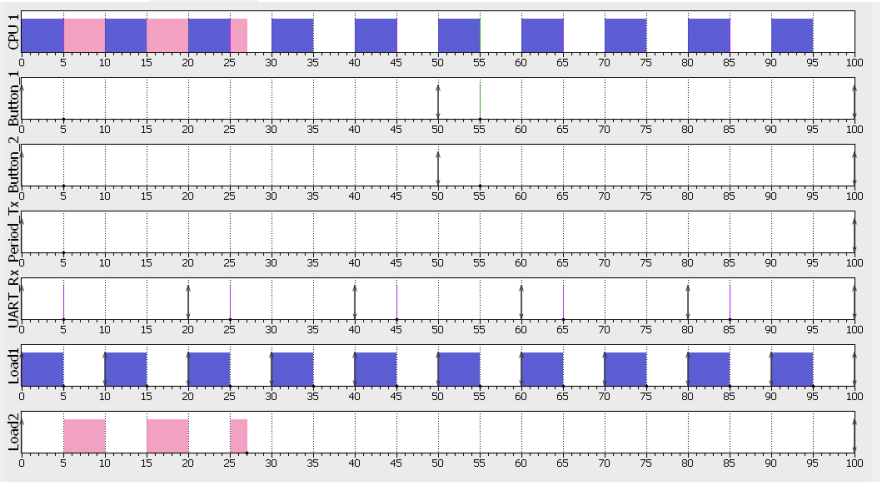
$$W(100) = 12 + \frac{100}{100} * 0.016 + \frac{100}{50} * 0.016 + \frac{100}{50} * 0.016 + \frac{100}{20} * 0.016 + \frac{100}{10} * 5 =$$

$$62.128\ ms < 100\ ms$$

We can notice that for all tasks – w(t) is always less than the period of the task. Therefore, all tasks are schedulable.

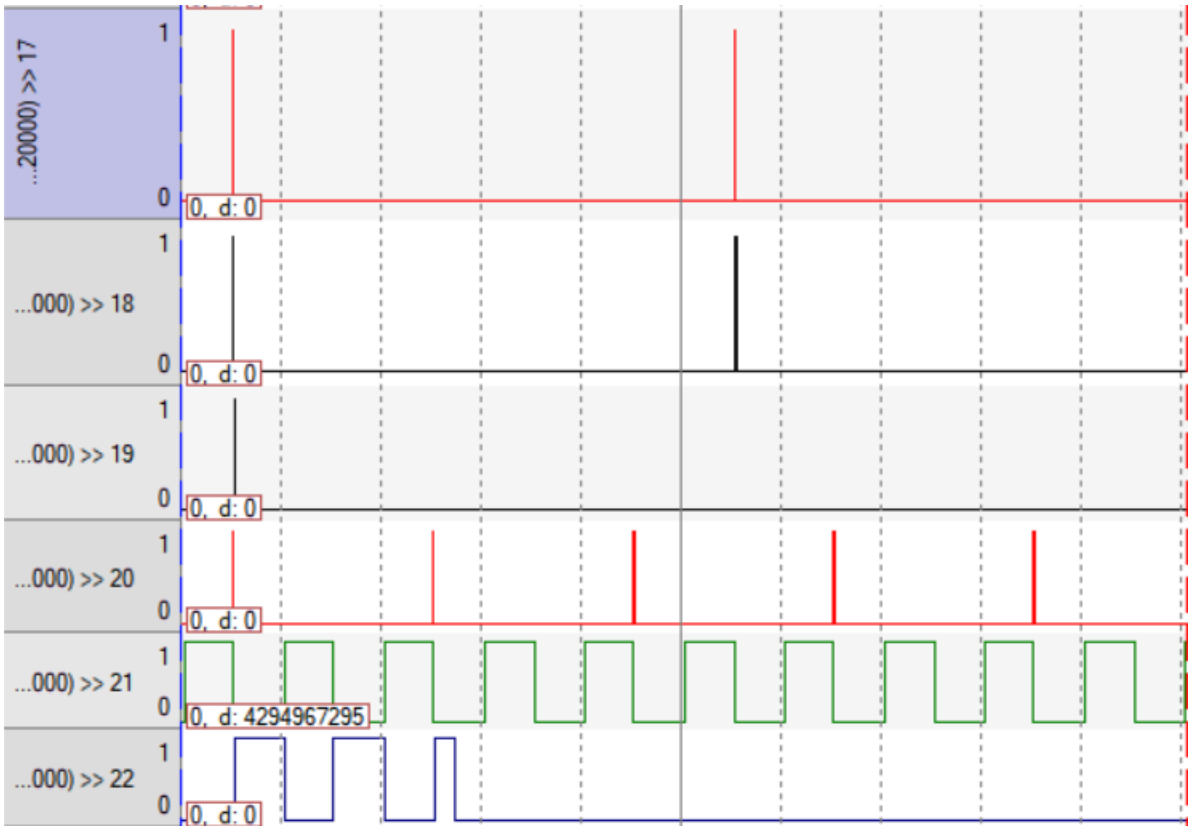
IV. USING SIMSO OFFLINE SIMULATOR

	Total load	Payload	System load
CPU 1	0.6216	0.6216	0.0000
Average	0.6216	0.6216	0.0000



V. KEIL CPU LOAD CALCULATIONS

Logic analyzer screenshot of a hyperperiod:


...20000) >> 17
1
0
0, d: 0
...000) >> 18
1
0
0, d: 0
...000) >> 19
1
0
0, d: 0
...000) >> 20
1
0
0, d: 0
...000) >> 21
1
0
0, d: 4294967295
...000) >> 22
1
0
0, d: 0