Earlist Deadline First

Project

Calculating CPU Utilization for the following Tasks to Prove Schedulablility:

$Button\_1(P: 50, D: 50, E: 2)$

$Button\_2(P: 50, D: 50, E: 2)$

$Period\_Transmission(P: 100, D: 100, E: 2)$

$Uart\_Receive(P: 20, D: 20, E: 1.2)$

$$U = \sum_{i=1}^{n} \frac{C_i}{P_i} \leq n\left(2^{\frac{1}{n}} - 1\right) \qquad U = Total\ Utilization,$$

$$n = number\ of\ Tasks, C = Execution\ time,$$

$$P = hyper\ Period$$

$$U = \left(\frac{2 * 2}{100}\right) + \left(\frac{2 * 2}{100}\right) + \left(\frac{2}{100}\right) + \left(\frac{5 * 1.2}{100}\right) = 0.16$$

$$URM = \ 4 * \left(2^{\frac{1}{4}} - 1\right) = 0.76 \quad URM = Rate\ Monotonic\ Utilization$$

$\because U \leq URM$

$\therefore The\ System\ is\ Schedulable\ for\ Fixed\ Priority\ Task\ with\ Only\ 4\ Tasks$

Now to recalculate using Time Demand Analysis for The 4 main Tasks:
for (Task 4):

$w(1) = 1.2 + 0 = 1.2$
$w(2) = 1.2 + 0 = 1.2$
$w(3) = 1.2 + 0 = 1.2$
$w(4) = 1.2 + 0 = 1.2$
$w(5) = 1.2 + 0 = 1.2$
$w(6) = 1.2 + 0 = 1.2$
$w(7) = 1.2 + 0 = 1.2$
$w(8) = 1.2 + 0 = 1.2$
$w(9) = 1.2 + 0 = 1.2$
$w(10) = 1.2 + 0 = 1.2\ till\ w(20) = 1.2 + 0 = 1.2$
$w(20) < D = 1.2 < 20\ ,\qquad \therefore T4\ is\ schedulable$

now tasks (1) and (2) are going to enter:

$w(1) = 2 + 2 + (1.2 * 1) = 5.2$

$till\ w(20) = 2 + 2 + (1.2 * 1) = 5.2$

$from\ w(21) till\ w(40) = 2 + 2 + (1.2 * 2) = 6.4$

$from\ w(41) till\ w(50) = 2 + 2 + (1.2 * 3) = 7.6 < D = 20$

$\therefore T1\ and\ T2\ are\ schedulable$

$Now\ to\ add\ the\ last\ task\ (T3)$:

$w(1)\ till\ w(20) = 5.2 + 2 = 7.2$

$from\ w(21)\ till\ w(40) = 6.4 + 2 = 8.4$

$from\ w(41)\ till\ w(50) = 7.6 + 2 = 9.6$

$from\ w(51) till\ w(60) = 2 + (2 * 2) + (2 * 2) + (1.2 * 3) = 13.6$

$from\ w(61) till\ w(80) = 2 + (2 * 2) + (2 * 2) + (1.2 *\ 4) = 14.8$

$rom\ w(81) till\ w(100) = 2 + (2 * 2) + (2 * 2) + (1.2 *\ 5) = 16 < D$

$=\ 100\ , \therefore T3\ is\ Schedulable.$

Now to recalculate using Time Demand Analysis for The 6 Tasks with system load:

for (Task 5):

$w(1) = 5 + 0 = 5$

$w(2) = 5 + 0 = 5$

$w(3) = 5 + 0 = 5$

$w(4) = 5 + 0 = 5$

$w(5) = 5 + 0 = 5$

$w(6) = 5 + 0 = 5$

$w(7) = 5 + 0 = 5$

$w(8) = 5 + 0 = 5$

$w(9) = 5 + 0 = 5$

$w(10) = 5 + 0 = 5 < D = 10\ \ \therefore T5\ is\ Schedulable$

for (Task 4):

$w(1) = 1.2 + 5 = 6.2$

$w(2) = 1.2 + 5 = 6.2$

$w(3) = 1.2 + 5 = 6.2$

$w(4) = 1.2 + 5 = 6.2$

$w(5) = 1.2 + 5 = 6.2$

$w(6) = 1.2 + 5 = 6.2$

$w(7) = 1.2 + 5 = 6.2$
$w(8) = 1.2 + 5 = 6.2$
$w(9) = 1.2 + 5 = 6.2$
$w(10) = 1.2 + 5 = 6.2$
$from\ w(11) till\ w(20) = 1.2 + (5 * 2) = 11.2$
$w(20) < D = 11.2 < 20\ ,\qquad \therefore T4\ is\ schedulable$
now tasks (1) and (2) are going to enter:
$w(1) = 2 + 2 + (1.2 * 1) + (5 * 1) = 10.2$
$till\ w(10) = 2 + 2 + (1.2 * 1) + (5 * 1) = 10.2$
$from\ w(11) till\ w(20) = 2 + 2 + (1.2 * 1) + (5 * 2) = 15.2$
$from\ w(21) till\ w(30) = 2 + 2 + (1.2 * 2)\ + (5 * 3) = 21.4$

$from\ w(31) till\ w(40) = 2 + 2 + (1.2 * 2)\ + (5 * 4) = 26.4$
$from\ w(41) till\ w(50) = 2 + 2 + (1.2 * 3)\ + (5 * 5) = 32.6 < D = 50,$
$\therefore T1\ and\ T2\ are\ schedulable$
$Now\ to\ add\ the\ task\ (T3)\ \&\ (T6)$:
$from\ w(51) till\ w(60) = 2 + (2 * 2) + (2 * 2) + (1.2 * 3) + (5 * 6) + 12$
$= 55.6$
$from\ w(61) till\ w(70)$
$= 2 + (2 * 2) + (2 * 2) + (1.2 *\ 4) + (5 * 7) + 12 = 61.8$
$from\ w(71) till\ w(80)$
$= 2 + (2 * 2) + (2 * 2) + (1.2 *\ 4) + (5 * 8) + 12 = 66.8$
$from\ w(81) till\ w(90)$
$= 2 + (2 * 2) + (2 * 2) + (1.2 *\ 5) + (5 * 9) + 12 = 73$
$rom\ w(91) till\ w(100)$
$= 2 + (2 * 2) + (2 * 2) + (1.2 *\ 5) + (5 * 10) + 12 = 78$
$< D =\ 100\ , \therefore\ T3\ and\ T6\ \ are\ Schedulable.$
$\therefore Based\ on\ time\ Demand\ Analysis\ the\ System\ is\ fully\ Schedulable.$

Figure (1) shows the Schedulable System, while Figure (2) will show the System is going to be not Schedulable when it comes to Fixed Priority.

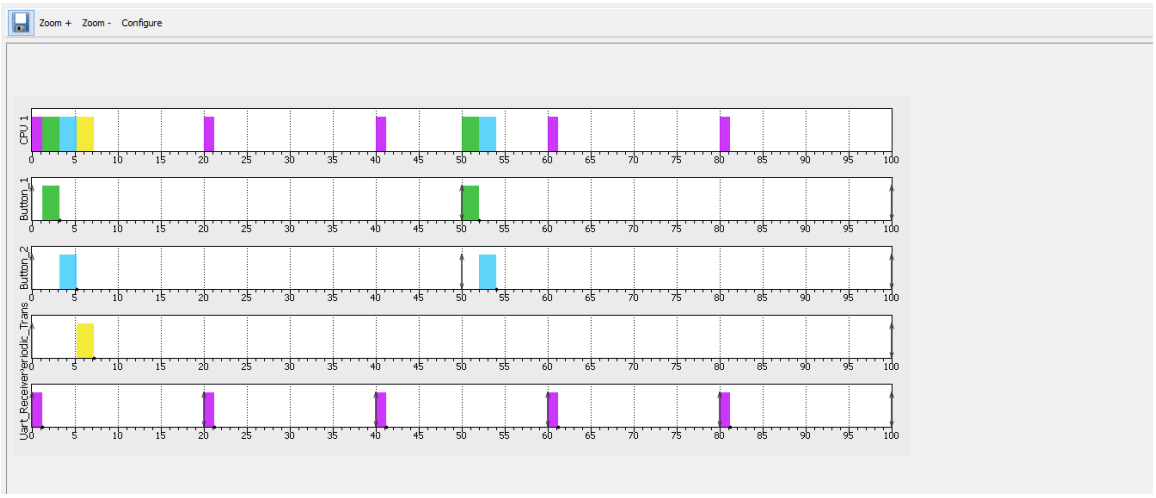

Figure (1) Rate Monotic Schedulablility for the 4 Main Tasks

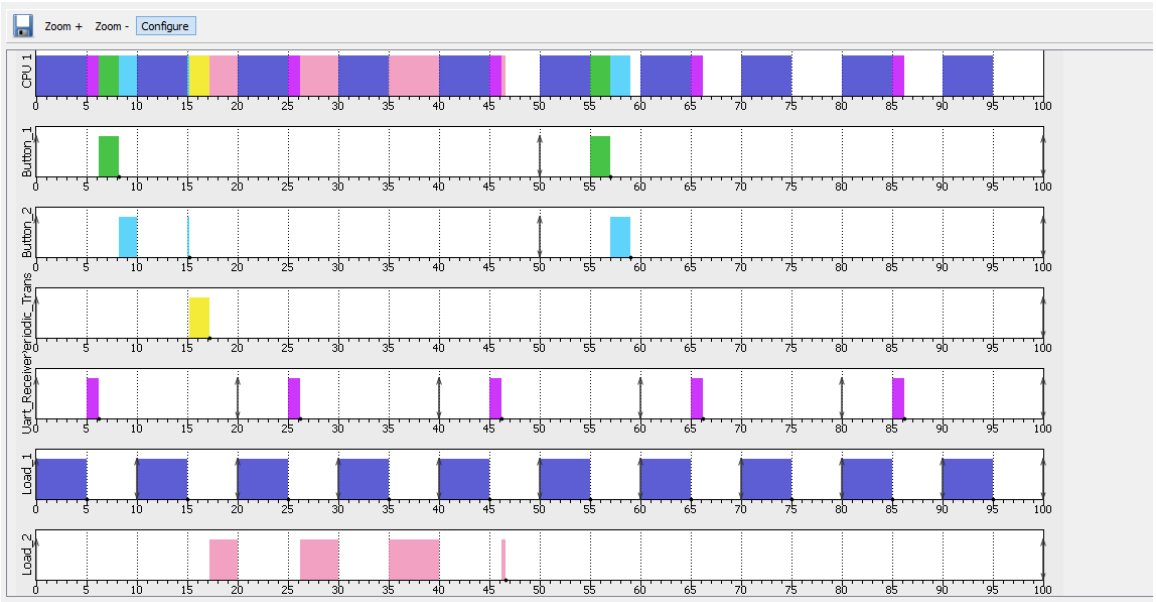

Figure (2) Rate-monotonic with all 6 Tasks added to CPU Load

The Following Figure is showing the Behavior of the system based on Events as follows:

Using Logic Analyzer:

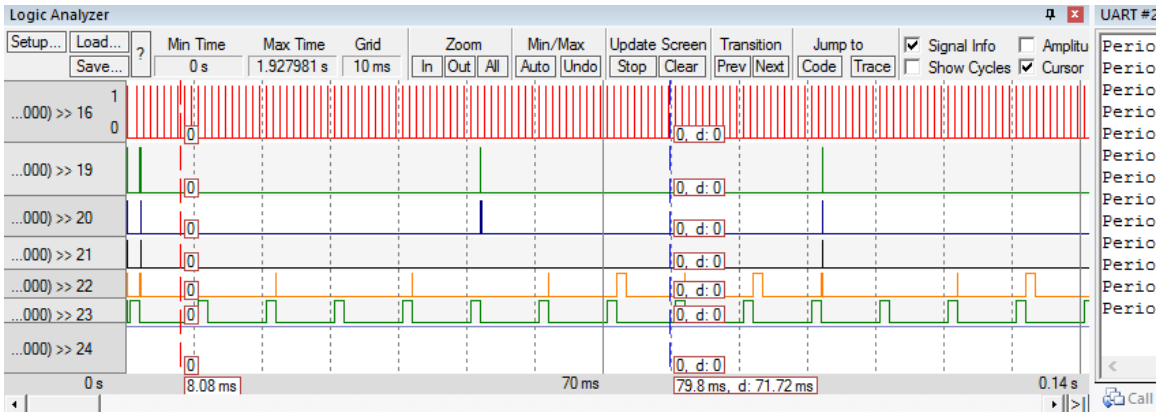

Figure (3) Logic Analyzer showing the behavior of the System with both load Functions sequentially