

# System Verification

MUSTAFA ALI

# Tasks:

- Note: all tasks execution time is calculated from the actual implemented tasks using GPIOs and the logic analyzer

Task Name	Periodicity / Deadline (MS)	Execution Time (MS)
Button_1_Monitor	50	0.0012
Button_2_Monitor	50	0.0012
Periodic Transmitter	100	0.00547
UART Receiver	20	0.00327
Load_1	10	5
Load_2	100	12

# Methods of Verification:

- **1- Using Analytical Method:**

- 1.1. System Hyper period:

- It's the Least Common Multiple of all task periods
- $H = \text{LCM}(50, 50, 100, 20, 10, 100) = 100$

- 1.2. CPU Load

- $U = (E1 + E2 + E3 + E4 + E5 + E6) / H$
- where E is the Execution time and H is the Hyper period.
- $U = (0.0012*2 + 0.0012*2 + 0.00547 + 0.00327*5 + 5*10 + 12) / 100$   
 $= 0.6202662 = (62.0266\%)$

- 1.3. System stimulability check using URM and Time Demand Analysis Techniques:

- $\sum c_i / p_i \leq n(2^{1/n} - 1)$  ;sigma from i=1 to n
- $L. H. S = \sum c_i / p_i = 0.0012/50 + 0.0012/50 + 0.00547/100 + 1 \cdot 0.00327/20 + 5/10 + 12/100 = 0.62026$
- $R. H. S = n(2^{1/n} - 1) = 0.7348$
- $L. H. S \leq R. H. S$  so, the system is schedulable.

## 2- Time Demand Analysis:

a- Sort the tasks making the highest priority at the first:

Task Name	Periodicity / Deadline(MS)	Execution Time (MS)
1- Load 1	20	5
2-UART Receiver	20	0.00327
3-Button_1_Monitor	50	0.0012
4-Button_2_Monitor	50	0.0012
5- Periodic Transmitter	100	0.00547
6-Load 2 Simulation	100	12

b- Choose the critical instant 0 then :

$$W1(10) = 5 + 0 = 5 < \textit{deadline}$$

$$w2(20) = 0.00327 + 5 * 20/10 = 10.00327 < \textit{deadline}$$

$$w3(50) = 0.0012 + 0.00327 * 50/20 + 5 * 50/10 = 25.0093 < \textit{deadline}$$

$$\begin{aligned} W4(50) &= 0.0012 + 0.0012 * 50/50 + 0.00327 * 50/20 + 5 * 50/10 \\ &= 25.01 < \textit{deadline} \end{aligned}$$

$$\begin{aligned} w5(100) &= 0.00547 + 0.0012 * 100/50 + 0.0012 * 100/50 + 0.00327 * \\ &100/20 + 5 * 100/10 = 50.0266 < \textit{deadline} \end{aligned}$$

$$\begin{aligned} w6(100) &= 12 + 0.00547 * 100/100 + 0.0012 * 100/50 + 0.0012 * 100/50 \\ &+ 0.00327 * 100/20 + 5 * 100/10 = 62.02 < \textit{deadline} \end{aligned}$$






As all Tasks are less Than the deadline. So, the system is schedulable.

### 3- Using SIMSO offline simulator:

Used Scheduler: Fixed priority rate monotonic. Tasks Simulated:

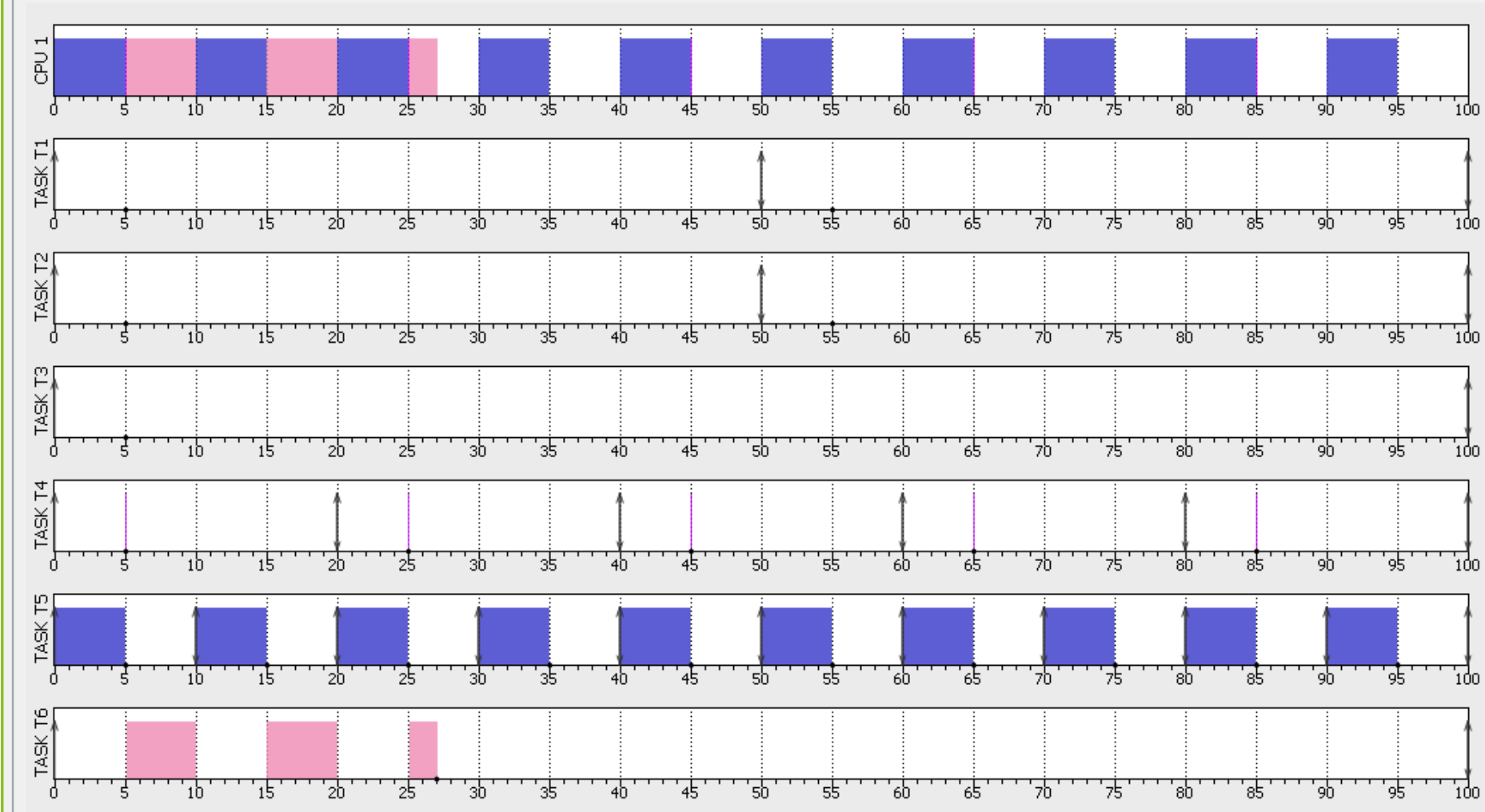
General		Scheduler	Processors	Tasks					
id	Name	Task type	Abort on miss	Act. Date (ms)	Period (ms)	List of Act. dates (ms)	Deadline (ms)	WCET (ms)	
1	TASK T1	Periodic ▼	<input type="checkbox"/> No	0	50	-	50	0.0012	
2	TASK T2	Periodic ▼	<input type="checkbox"/> No	0	50	-	50	0.0012	
3	TASK T3	Periodic ▼	<input type="checkbox"/> No	0	100	-	100	0.00547	
4	TASK T4	Periodic ▼	<input type="checkbox"/> No	0	20	-	20	0.00327	
5	TASK T5	Periodic ▼	<input type="checkbox"/> No	0	10	-	10	5	
6	TASK T6	Periodic ▼	<input type="checkbox"/> No	0	100	-	100	12	

- For The CPU load the is the same as the analytical mode

    Gantt Results			
* Unsaved 			
General	Logs	Tasks	Scheduler Processors
Observation Window:			
from 0.00 to 100.00 ms			
	Total load	Payload	System load
CPU 1	0.6203	0.6203	0.0000
Average	0.6203	0.6203	0.0000

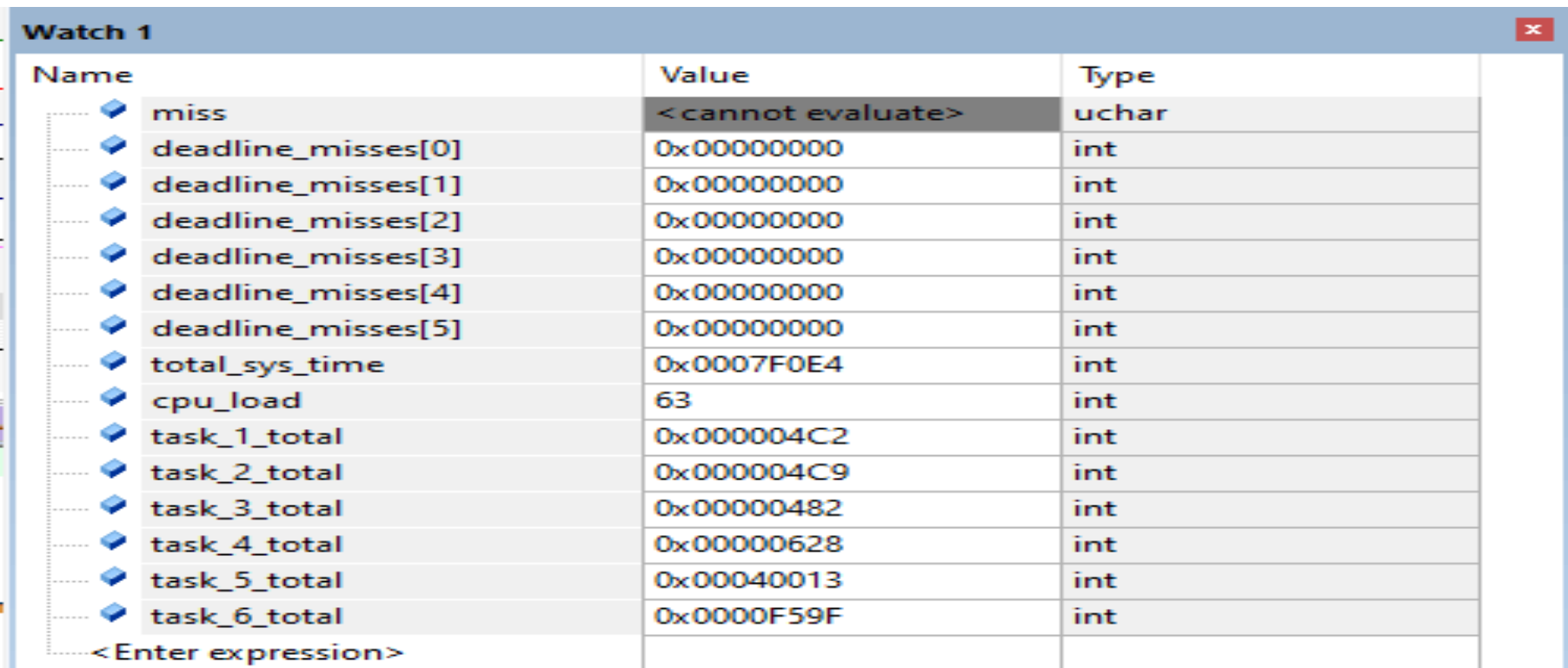


# Gantt chart over the Hyper period:



## 4- Using Keil Simulator at Runtime:

- 1- Calculate the CPU usage time using timer 1 and trace macros:



The screenshot shows the 'Watch 1' window in the Keil simulator. It contains a table with three columns: 'Name', 'Value', and 'Type'. The table lists various variables and their current values during simulation. The 'miss' variable is highlighted with a grey background and shows '<cannot evaluate>'. Other variables include 'deadline\_misses' (an array of 6 zeros), 'total\_sys\_time' (0x0007F0E4), 'cpu\_load' (63), and task-specific totals (task\_1\_total to task\_6\_total) with values ranging from 0x000004C2 to 0x0000F59F. A prompt '<Enter expression>' is visible at the bottom left of the table area.

Name	Value	Type
miss	<cannot evaluate>	uchar
deadline_misses[0]	0x00000000	int
deadline_misses[1]	0x00000000	int
deadline_misses[2]	0x00000000	int
deadline_misses[3]	0x00000000	int
deadline_misses[4]	0x00000000	int
deadline_misses[5]	0x00000000	int
total_sys_time	0x0007F0E4	int
cpu_load	63	int
task_1_total	0x000004C2	int
task_2_total	0x000004C9	int
task_3_total	0x00000482	int
task_4_total	0x00000628	int
task_5_total	0x00040013	int
task_6_total	0x0000F59F	int
<Enter expression>		

Note:

1- The CPU load is the same as the calculated analytically and the obtained using SIMSO offline simulator.

2- None of The Tasks Miss the Deadline

2. Using trace macros and GPIOs, plot the execution of all tasks, tick, and the idle task on the logic analyzer:

