Producer Consumer Tasks FreeRTOS

Name : Tejas Madhukar Pidkalwar

1) higher priority for Consumer task:

Scenario:

- Consumer task
    - is given High Priority
    - is waiting on till queue receives valid element
- Producer task is given Low Priority
    - is given Low priority
    - sleeps for a 1 Second after sending data

Observation:

- ➔ Producer task send data to queue,
- ➔ Cooperative context switching happens to Consumer task, as high priority task waiting for valid element in queue
- ➔ Consumer task receives task and again context switching happens to go back to the Producer task to perform remaining functionality
- ➔ Consumer tasks sleeps for 1 second

```
tejas@tejas-Lenovo-G400s-Touch:~/Sem_2/CMPE-244/Embedded-Software-Assignments$ _build_x86_freertos/./x86_freertos.exe
Running as PID: 38444
Starting FreeRTOS Scheduler .....

Timer Resolution for Run TimeStats is 100 ticks per second.
Ping from Before sending the msg
The data 1 is received over the RTOS queue
Ping after sending 1 data over the RTOS queue
Ping from Before sending the msg
The data 1 is received over the RTOS queue
Ping after sending 1 data over the RTOS queue
Ping from Before sending the msg
The data 1 is received over the RTOS queue
Ping after sending 1 data over the RTOS queue
Ping from Before sending the msg
The data 1 is received over the RTOS queue
Ping after sending 1 data over the RTOS queue
Ping from Before sending the msg
The data 1 is received over the RTOS queue
Ping after sending 1 data over the RTOS queue
Ping from Before sending the msg
The data 1 is received over the RTOS queue
Ping after sending 1 data over the RTOS queue
Ping from Before sending the msg
The data 1 is received over the RTOS queue
Ping after sending 1 data over the RTOS queue
Ping from Before sending the msg
The data 1 is received over the RTOS queue
Ping after sending 1 data over the RTOS queue
Ping from Before sending the msg
The data 1 is received over the RTOS queue
Ping after sending 1 data over the RTOS queue
Ping from Before sending the msg
The data 1 is received over the RTOS queue
Ping after sending 1 data over the RTOS queue
Ping from Before sending the msg
```

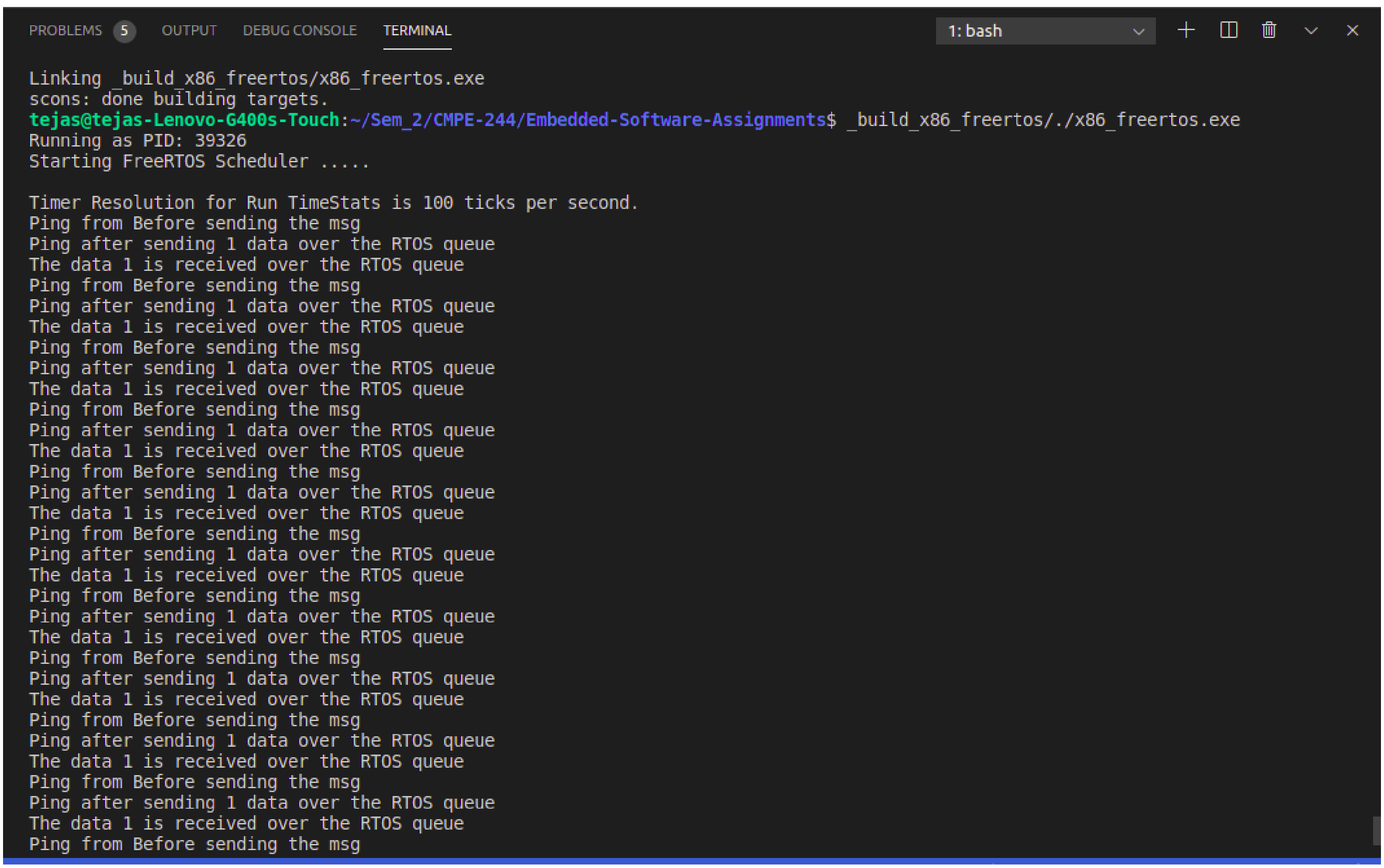
2) higher priority for consumer task

<u>Scenario:</u>

- Consumer task
  - is given Low Priority
  - is waiting on till queue receives valid element
- Producer task is given Low Priority
  - is given High priority
  - sleeps for a 1 Second after sending data

<u>Observation:</u>

- ➔ Producer task send data over the RTOS queue
- ➔ After sending data, Producer task goes to sleep
- ➔ Preemptive context switch happens to Consumer task
- ➔ Consumer task receives data from the queue
- ➔ Context switch back to Producer queue once it wake up from the sleep




```
Linking _build_x86_freertos/x86_freertos.exe
scons: done building targets.
tejas@tejas-Lenovo-G400s-Touch:~/Sem_2/CMPE-244/Embedded-Software-Assignments$ _build_x86_freertos/./x86_freertos.exe
Running as PID: 39326
Starting FreeRTOS Scheduler .....

Timer Resolution for Run TimeStats is 100 ticks per second.
Ping from Before sending the msg
Ping after sending 1 data over the RTOS queue
The data 1 is received over the RTOS queue
Ping from Before sending the msg
Ping after sending 1 data over the RTOS queue
The data 1 is received over the RTOS queue
Ping from Before sending the msg
Ping after sending 1 data over the RTOS queue
The data 1 is received over the RTOS queue
Ping from Before sending the msg
Ping after sending 1 data over the RTOS queue
The data 1 is received over the RTOS queue
Ping from Before sending the msg
Ping after sending 1 data over the RTOS queue
The data 1 is received over the RTOS queue
Ping from Before sending the msg
Ping after sending 1 data over the RTOS queue
The data 1 is received over the RTOS queue
Ping from Before sending the msg
Ping after sending 1 data over the RTOS queue
The data 1 is received over the RTOS queue
Ping from Before sending the msg
Ping after sending 1 data over the RTOS queue
The data 1 is received over the RTOS queue
Ping from Before sending the msg
Ping after sending 1 data over the RTOS queue
The data 1 is received over the RTOS queue
Ping from Before sending the msg
Ping after sending 1 data over the RTOS queue
The data 1 is received over the RTOS queue
Ping from Before sending the msg
```

3) same priority level for both tasks:

<u>Scenario:</u>

- Consumer task
  - is given Low Priority
  - is waiting on till queue receives valid element
- Producer task is given Low Priority
  - is given Low priority
  - sleeps for a 1 Second after sending data

Observation:

- ➔ Producer task sends data over RTOS queue
- ➔ Producer task goes to sleep
- ➔ Preemptive context switch happens to Consumer task
- ➔ Consumer task receives data over the queue
- ➔ Preemptive context switch happens back to Producer task once it wake up from sleep

```
PROBLEMS 5   OUTPUT   DEBUG CONSOLE   TERMINAL                    1: bash

tejas@tejas-Lenovo-G400s-Touch:~/Sem_2/CMPE-244/Embedded-Software-Assignments$ _build_x86_freertos/./x86_freertos.exe
Running as PID: 39572
Starting FreeRTOS Scheduler .....

Timer Resolution for Run TimeStats is 100 ticks per second.
Ping from Before sending the msg
Ping after sending 1 data over the RTOS queue
The data 1 is received over the RTOS queue
Ping from Before sending the msg
Ping after sending 1 data over the RTOS queue
The data 1 is received over the RTOS queue
Ping from Before sending the msg
Ping after sending 1 data over the RTOS queue
The data 1 is received over the RTOS queue
Ping from Before sending the msg
Ping after sending 1 data over the RTOS queue
The data 1 is received over the RTOS queue
Ping from Before sending the msg
Ping after sending 1 data over the RTOS queue
The data 1 is received over the RTOS queue
Ping from Before sending the msg
Ping after sending 1 data over the RTOS queue
The data 1 is received over the RTOS queue
Ping from Before sending the msg
Ping after sending 1 data over the RTOS queue
The data 1 is received over the RTOS queue
Ping from Before sending the msg
Ping after sending 1 data over the RTOS queue
The data 1 is received over the RTOS queue
Ping from Before sending the msg
Ping after sending 1 data over the RTOS queue
The data 1 is received over the RTOS queue
Ping from Before sending the msg
Ping after sending 1 data over the RTOS queue
The data 1 is received over the RTOS queue
Ping from Before sending the msg
Ping after sending 1 data over the RTOS queue
The data 1 is received over the RTOS queue
```

4) What is the purpose of the block time during xQueueReceive()?

=> Block time parameter allows the task, in which xQueueReceive() function is called, to sleep for mentioned amount of time. Every time period equal to block time, the task will look for valid element in FreeRTOS queue and will sleep in mean time. This way CPU will be free to perform other operations while the task is sleeping.

5) What if you use ZERO block time during xQueueReceive()?

=> If the block time is zero and there is not task delay added to the task, then the task will be continuously keep polling on the queue for the valid element during its active time period.

Extra Credit Assignment:

=> Added CLI command handler for command “taskcontrol” with suspend and resume functionality for mentioned tasks.
=> Below is the screenshot of suspending and resuming tasks “led0” and “led1”.