Progress Presentation-I

e-Yantra Summer Intership-2016
FreeRTOS on LPC2148

K V S Sumakar
Kartikeyan V
Mentor:
Rutuja
Deepa

IIT Bombay

July 4, 2016

Overview of Project

- Project Name : FreeRTOS on LPC2148
- Objective : To create modules on the implementation of basic FreeRTOS concepts on LPC2148
- Deliverables : Documentation of each and every FreeRTOS concepts that has been implemented on LPC2148.

Overview of Task

Sr No.	TASKS	DEADLINES
1.	Basics of RTOS	4 days
2.	Multi-Tasking Examples	5 days
3.	Concepts of Semaphore and Mutex examples based on the concept	4 days
4.	Inter-Process communication-Mailbox and queues. Examples based on the concept	3 days
5.	Concept of Context Switching. Examples based on the concept	5 days
6.	A mini project that covers all the modules	3 days

Task Accomplished

- Learning basics of RTOS.
  - What is RTOS.
  - Its characteristics.
  - Difference between RTOS, GPOS.
- Implemented Multi-Tasking using FreeRTOS in Firebird V(LPC2148)
- Implemented Mutexes, Binary Semaphore and Counting Semaphore.
- Implemented Inter-Process communication.
  - Queues
  - Mailbox through Task notification.

```
Mutex
Back
Forward function access denied
Forward
Back Function access denied
Back
Forward function access denied
Forward
Back Function access denied
Back
Binary Semaphore
Semaphore given
Back
Semaphore given
Forward
Semaphore given
Back
Semaphore given
Forward
Semaphore given
Back
```

```
P3:Hungry
P5:Ate
P5:Thinking
P4:Left fork obtained Eating :)
P2:Right fork obtained
P4:Ate
P4:Thinking
P2:Left fork obtained Eating :)
P1:Right fork obtained
P3:Hungry
P5:Hungry
P2:Ate
P2:Thinking
P1:Left fork obtained Eating :)
P4:Right fork obtained
P1:Ate
P1:Thinking
P4:Left fork obtained Eating :)
P3:Right fork obtained
P5:Hungry
P2:Hungry
P4:Ate
P4:Thinking
```


MailBox using Task Notification
No Notice
No Notice
No Notice
N1 sent a Message
Received MSG from N1
N2 sent a Message
Received MSG from N2
N3 sent a MSG
Received MSG from N3
N4 sent a MSG
Received MSG from N4
N1 sent a Message
Received MSG from N1
No Notice
N2 sent a Message
Received MSG from N2
No Notice
N3 sent a MSG
ReceivN1 sent a Message
ed MSG from N3

Queue

```
Task 1 sent a message
Task 1 sent a message
Task 1 sent a message
Task 1 sent a message
Task 1 sent a message
Task 1 sent a message
Task 1 sent a message
Task 1 sent a message
Task 2 sent a message
Task 3 sent a message
Task 4 sent a message
Task 1 sent a message
Task 2 sent a message
Task 3 sent a message
Task 4 sent a message
Task 1 sent a message
Task 2 sent a message
Task 3 sent a message
Task 4 sent a message
Task 1 sent a message
Task 2 sent a message
```

Challenges Faced

- **Issue** : Porting RTOS into Firebird V and the configurations that needed changes.
- Solution : Replace the startup.s file and include various other libraries.
- Finding Implementation level difference between Binary Semaphore and Mutex.
- **Issue** : Loss of Data in Serial Communication.
- Solution :
  - Shortening the string size(temporary solution),
  - Tried creating a Mutex for accessing the Serial communication Functions.
  - Trying to solve Using Queues

Future Plans

- Implement Context switching.
- Create a mini project that can demonstrate all the learnt concepts together .

References

- http://www.freertos.org
- http://tinymicros.com/
- http://www.ocfreaks.com/cat/embedded/lpc2148-tutorials/
- http://www.rtos.be/2013/05/mutexes-and-semaphores-two-concepts-for-two-different-use-cases/

Thank You

THANK YOU !!!