FIT3142 Laboratory #4: Client-Server Applications Using Shared Memory Inter-Process Communications Part 2

Dr Carlo Kopp, SMIEEE, AFAIAA, FLSS, PEng
Faculty of IT, Clayton.
Email: Carlo.Kopp@monash.edu

August 11, 2015

Revision Status:

$Id: FIT3142-Laboratory-4.tex,v 1.3 2015/08/11 05:14:31 carlo Exp carlo $

Contents

1 Introduction

Inter-Process Communications (IPC) techniques are the foundation upon which all distributed computing schemes operate. The purpose of the this and the following two Laboratory exercises is to provide students with experience in coding and designing applications using client-server model using shared memory IPC, and then applying this using stream oriented socket IPC.

The aim of this lab will be to familiarise you with the behaviour of shared memory, using the Linux system and the C language. It extends on the work you did in Laboratory 3.

It is important that you do this work independently. Collaboration, copying and plagiarism will attract zero marks in this subject, without exception.

2 Lab Session 2

The second task for this two part lab involves modifying your server and client programs, produced in the previous lab, to permit the concurrent operation of four client programs using the single server program.

You will implement the following functions:

1. You will make use of a locking function when any client writes to the shared memory. You will use the `lock` variable in `SEG_DATA` for this purpose. If the lock is set the client must wait until it is released before writing.
2. Once a client is running it will advertise its presence to the server, and other clients, by setting a bit in the `present` variable. It will do this by reading the variable, and setting the next available `CLIENT_X` bit, defined in `segment-lock.h`.
3. The server will exit only after the last of the four clients has exited. Use a bitwise operator to set the `CLIENT_X` bit in the `exit` variable.
4. You must test the client server application using all four clients (Hint: use two clients first and once working reliably, test three and four clients).
5. You must edit `MyID-shm-client.c` to incorporate your own code.
6. You must edit `status-shm-server.c` to incorporate your own code.
7. You will submit by email the source file `MyID-shm-client-lock.c`.
8. You will submit by email the source file `MyID-shm-server-lock.c`.
9. If your program does not compile you will be awarded zero marks for this prac.
10. You must demonstrate your program to the demonstrator before the end of the second prac session. If your program does not work at all you will be awarded zero marks for this prac. If it has bugs, marks will be deducted accordingly.
11. Cheating, collaborating and plagiarism will attract zero marks.
12. **Save all of your program files as you will need them in future labs.**

3 Server and Client Function

```
deathstar[carlo]1223% status-server

STATUS DUMP
UP Status        = 0
Exit Status      = 0
RPM              = 3400
Crank Angle      = -1
Throttle Setting = 69
Fuel Flow        = 49
Engine Temp      = 79
Fan Speed        = 29
Oil Pressure     = 69
Waiting for client

STATUS DUMP
UP Status        = 0
Exit Status      = 0
RPM              = 3300
Crank Angle      = -2
Throttle Setting = 68
Fuel Flow        = 48
Engine Temp      = 78
Fan Speed        = 28
Oil Pressure     = 68
Waiting for client

STATUS DUMP
UP Status        = 0
Exit Status      = 0
RPM              = 3200
Crank Angle      = -3
Throttle Setting = 67
Fuel Flow        = 47
Engine Temp      = 77
Fan Speed        = 27
Oil Pressure     = 67
Waiting for client

UP Status        = 0
Exit Status      = 0
```

```
RPM              = 3100
Crank Angle      = -4
Throttle Setting = 66
Fuel Flow        = 46
Engine Temp      = 76
Fan Speed        = 26
Oil Pressure     = 66
Waiting for client

STATUS DUMP
UP Status        = 0
Exit Status      = 0
RPM              = 3000
Crank Angle      = -5
Throttle Setting = 65
Fuel Flow        = 45
Engine Temp      = 75
Fan Speed        = 25
Oil Pressure     = 65
Waiting for client

STATUS DUMP
UP Status        = 0
Exit Status      = 0
RPM              = 2900
Crank Angle      = -6
Throttle Setting = 64
Fuel Flow        = 44
Engine Temp      = 74
Fan Speed        = 24
Oil Pressure     = 64
Waiting for client

STATUS DUMP
UP Status        = 0
Exit Status      = 1
RPM              = 2800
Crank Angle      = -7
Throttle Setting = 63
Fuel Flow        = 43
Engine Temp      = 73
Fan Speed        = 23
Oil Pressure     = 63
```

```
Waiting for client
Task completed
deathstar[carlo]1224%


deathstar[carlo]1025% MyID-client

CLIENT STATUS DUMP
RPM              = 3200
Crank Angle      = -3
Throttle Setting = 67
Fuel Flow        = 47
Engine Temp      = 77
Fan Speed        = 27
Oil Pressure     = 67
Enter Command (1 to exit, 0 to continue): 0


CLIENT STATUS DUMP
RPM              = 3000
Crank Angle      = -5
Throttle Setting = 65
Fuel Flow        = 45
Engine Temp      = 75
Fan Speed        = 25
Oil Pressure     = 65
Enter Command (1 to exit, 0 to continue): 1

Task completed
deathstar[carlo]1026%
```