**Timer Lib**

..................................................................................................................................................................

Version: 1.0

Checked Kernel version: Redhat 3.10.0

Timer calls: POSIX

Author: Amit Huda

..................................................................................................................................................................

1. Why timer Lib?

* Timers create problems in multithreaded application as no of signal in Linux is limited. This library supports any number of timers per thread per process.

1. Where can be used?

* Library has been checked with Redhat kernel 3.10.0.
* As library used POSIX based timer calls, it can be used with any UNIX based system that supports POSIX calls with necessary modification.

1. Resolution support for timers.

* For 100ms timer error is 0.12ms.

1. How to use?

* Add libTimer.a library in your application.
* For user library provides three function calls, with them user can create timer, modify timer and delete timer as per application requirement.
* Function calls as below,

1. int MakeTimer( timer\_t \*timerID, int ExpireTimeInSecond, int ExpireTimeInNanoSecond,int IntervalTimeInSecond ,int IntervalTimeInNanoSecond, void (TimerHandler)(int));
2. int ResetTimerVal( timer\_t timerID, int ExpireTimeInSecond, int ExpireTimeInNanoSecond,int IntervalTimeInSecond ,int IntervalTimeInNanoSecond);
3. int DeleteTimer(timer\_t timerID);

* MakeTimer() creates new timer.
* ResetTimerVal() reconfigure timer.
* DeleteTimer() Deletes timer
* Each above mentioned call returns error no or 0(if no error presents).
* If error is there error number is as follow

1. TIMER\_ERROR\_NO\_ERROR 0
2. TIMER\_ERROR\_ID\_NOT\_ZER0 1
3. TIMER\_ERROR\_SIGACTION\_FAIL 2
4. TIMER\_ERROR\_TIMER\_CREATE\_FAIL 3
5. TIMER\_ERROR\_TIMER\_SET\_TIME\_FAIL 4
6. TIMER\_ERROR\_MALLOC\_FAIL 5
7. TIMER\_ERROR\_TIMER\_DELETE\_FAIL 6
8. TIMER\_ERROR\_TIMER\_NOT\_PRESENT 7

* Argument information as below,

1. timerID : Timer id of type timer\_t
2. ExpireTimeInSecond: Expire time in seconds.
3. ExpireTimeInNanoSecond: Expire time in nano seconds.
4. IntervalTimeInSecond: Interval time in seconds.
5. IntervalTimeInNanoSecond: Interval Time in nano seconds.
6. TimerHandler : Handler function that user want to call when timer fires.

* For further assistance find attached sample program