

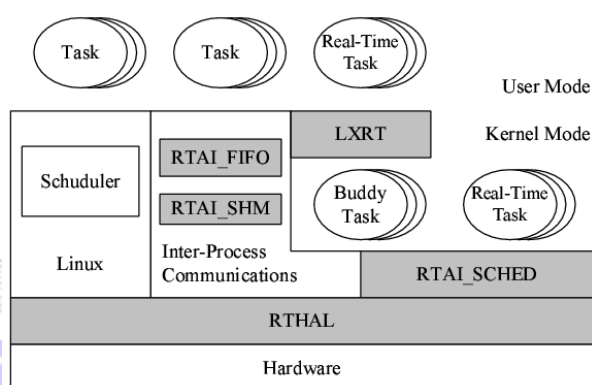
即時 Linux 系統資源管理

Budget-Based Resource Reservation for Real-Time Linux

指導教授：郭大維博士

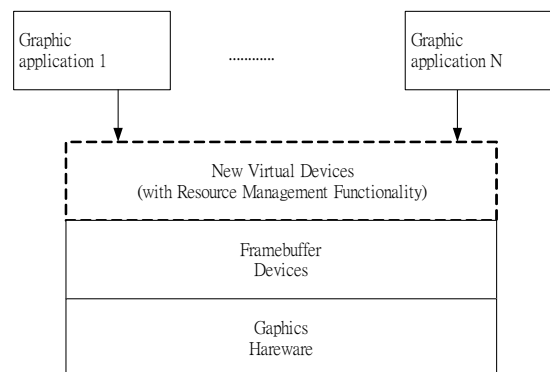
研究生：彭念劬、謝仁偉、劉進雙、魏仲佑

台大資工所 即時暨嵌入式系統實驗室 <http://rtlab.csie.ntu.edu.tw/>



圖(一) RTAI 及 Linux 整合系統架構圖。

RTAI (Real-Time Application Interface) 提供一個輕量 (light weight) 且高效能 (high performance) 的介面，藉由此介面，使用者可以在 Linux 的平台下，能夠實作硬即時 (hard real-time) 及軟即時 (soft real-time) 的應用程式。本研究首先設計一個以「執行額度」為基礎資源管理 (budget-based resource reservation) 的 RTAI 實作，延續 RTAI 的既有功能，允許使用者為他們撰寫的每個即時工作 (real-time task) 設定一個執行額度。在此執行額度的時間內，即時工作可以獲得 CPU 的執行使用權，且不會被具有高優先權的工作佔先 (preempt)。不同於其他先前的研究，我們更實作執行額度於使用者空間 (user space) 的 LXRT 軟即時工作，這比單純的 RTAI 實作更為複雜。本研究修改了中斷處理、RTAI 排程器，以及 RTAI 函式 `rt_task_wait_period()`，並沒有更改任何 Linux 核心原始碼；同時提供具有使用者介面之即時資源監控器。主要的應用平台建置於 Linux 核心 2.4.0-test10、RTAI 24.1.2，在 PII 和 PIII 的平台上。



圖(二) 資源保留於 Framebuffer 之架構圖.

Framebuffer 裝置是一種將顯示硬體抽象化的裝置。應用程式使用 Framebuffer 裝置的方法是將 Framebuffer 映射 (map) 至它本身的使用者空間以達成直接使用 Framebuffer 裝置的目的。為了確保應用程式能即時使用，我們亦建置了一套應用函式庫以保留 Framebuffer 裝置資源給應用程式使用。一個新增的「虛擬裝置」用以維護 Framebuffer 裝置資源管理的內部資料結構，透過此種方式，Framebuffer 裝置的原始碼 (source code) 就無需變更。一旦具有此資源保留機制，系統負擔量增高的時候，應用程式可獲得較為平順的顯示服務。