

TRACE32: una storia di successo

Linux, a caccia di sprechi di tempo con il trace hardware

Saper analizzare il comportamento dinamico di un sistema target può essere molto importante durante il debug, ma spesso non se ne tiene conto. In un sistema real-time, il più delle volte una risposta in ritardo è dannosa tanto quanto una risposta sbagliata. Specialmente nel mondo Linux ci sono diversi strumenti software che aiutano a misurare le prestazioni di un sistema embedded, ma a volte servono solo ad aumentare il problema. Questo articolo mostra come Advanced Driver Information Technology GmbH (ADIT) a Hildesheim abbia usato TRACE32 di Lauterbach, uno strumento di trace hardware non intrusivo, per risolvere una situazione di questo tipo.

Una delle soluzioni offerte da ADIT consiste nel rilasciare Linux su processori sia Arm che Intel. ADIT utilizza SystemTap [1] per misurare le prestazioni complessive del sistema al fine di localizzare e rimuovere i colli di bottiglia. SystemTap utilizza a sua volta due validi servizi di Linux chiamati uprobe e kprobe, che permettono all'utente di creare un trace dinamico delle funzioni rispettivamente a livello utente e a livello kernel.

Durante i test effettuati in questa configurazione, per carichi di sistema lievi o moderati non furono osservati problemi. Si pensava quindi che uno strumento software come SystemTap avesse un impatto lieve sulle prestazioni in tempo reale del sistema nel suo complesso. Ma ciò che risultò davvero inaspettato fu che, nelle piattaforme basate su Arm, all'aumentare del carico elaborativo il sistema rallentava in modo molto più accentuato rispetto alle piattaforme Intel. Per confermare il problema venne creata una funzione di prova e si rilevarono misure su questa funzione con uprobe, constatando che una singola chiamata a uprobe impiegava il doppio del tempo su dispositivi Arm. Dato che uprobe internamente usa kprobe, inizialmente si pensò che il colpevole fosse kprobe. Ma non era così perché kprobe di fatto girava più velocemente su processori Arm rispetto a Intel: evidentemente il problema doveva essere nel codice di uprobe.

Dato che il problema era localizzato nel codice del trace software, questo stesso codice non poteva essere usato per localizzarlo.

“Non sapendo come procedere, anche perché non si può dire che il codice di uprobe sia semplice, ho deciso di usare TRACE32 per capire cosa stava succedendo. A volte una bella immagine può essere di grande aiuto. Basandomi sulla funzionalità "chart", ho potuto selezionare aree di codice da analizzare in modo più approfondito”, ha detto Frederic Berat, sviluppatore presso ADIT.

Fu così che ADIT decise di usare TRACE32 PowerTrace con le sue funzionalità di trace hardware. Il trace hardware non ha alcun impatto sulla temporizzazione del target e permette un'analisi molto approfondita su parti di codice anche molto piccole.

Sia i dispositivi Arm che Intel sono in grado di fornire informazioni non intrusive sul flusso di programma. In Arm questa funzionalità si chiama ETM (Embedded Trace Macrocell) mentre la corrispondente prestazione Intel si chiama IPT (Intel Processor Trace). L'informazione sul codice eseguito viene trasmessa su un certo numero di pin dedicati. Lo strumento TRACE32 si collega a questi pin per raccogliergli i dati e poi analizzarli, ricostruendo il flusso dell'applicazione e la tempistica dettagliata di ogni funzione.

Anche in ambienti complessi, TRACE32 riesce a registrare e analizzare il flusso completo di programma sia per le applicazioni dell'utente che per il codice del kernel. L'avanzamento funzionale dell'intero sistema viene ricostruito e mostrato, calcolandone anche le statistiche, sotto forma di grafico basato sul tempo o come gerarchia di funzioni. Mostrare l'esecuzione di un sistema Linux completo, compresi il kernel e i processi, produce un grafico molto grande, ma TRACE32 agevola l'analisi dei percorsi critici. È così che gli ingegneri di ADIT sono riusciti a focalizzare in dettaglio le parti di kernel relative a kprobe e uprobe.

Grazie alle avanzate capacità di analisi di TRACE32, divenne subito evidente che c'erano due colli di bottiglia (vedere figure 1 e 2). Il punto più importante era che uprobe su piattaforma Arm chiama quattro volte `preempt_disable()` e `preempt_enable()`, ciascuna delle quali provoca un controllo dello stack frame che impiega circa $0.6\mu\text{s}$, causando un ritardo complessivo di $2.4\mu\text{s}$. Questo non succede su processori Intel. Potrebbe sembrare che una singola differenza di $2.4\mu\text{s}$ non sia molto ma, se

uprobe viene chiamato molte volte al secondo, presto si arriva a un ritardo considerevole. Scavando più a fondo, un altro collo di bottiglia fu rilevato nelle operazioni sulle stringhe, che sono una componente indispensabile di uprobe. Purtroppo non è stato possibile velocizzare questa parte, che riguarda le differenze architetturali fra Arm e Intel.

Senza il trace real-time sarebbe stato quasi impossibile arrivare a questi risultati, ma con il trace real-time è stato facile tracciare il comportamento. Sapendo dove guardare, ADIT ha scoperto che il problema principale era nella configurazione del kernel. Quando è stato migrato da un'altra piattaforma, inavvertitamente un'impostazione temporanea di `CONFIG_PREEMPT_TRACE` è stata lasciata abilitata. Il trace ha mostrato che questo causava un'esplorazione dello stack in Arm mentre non comportava alcuna operazione su Intel. Questo ha provocato l'enorme differenza di prestazioni fra i due sistemi.

[1] <https://en.wikipedia.org/wiki/SystemTap>

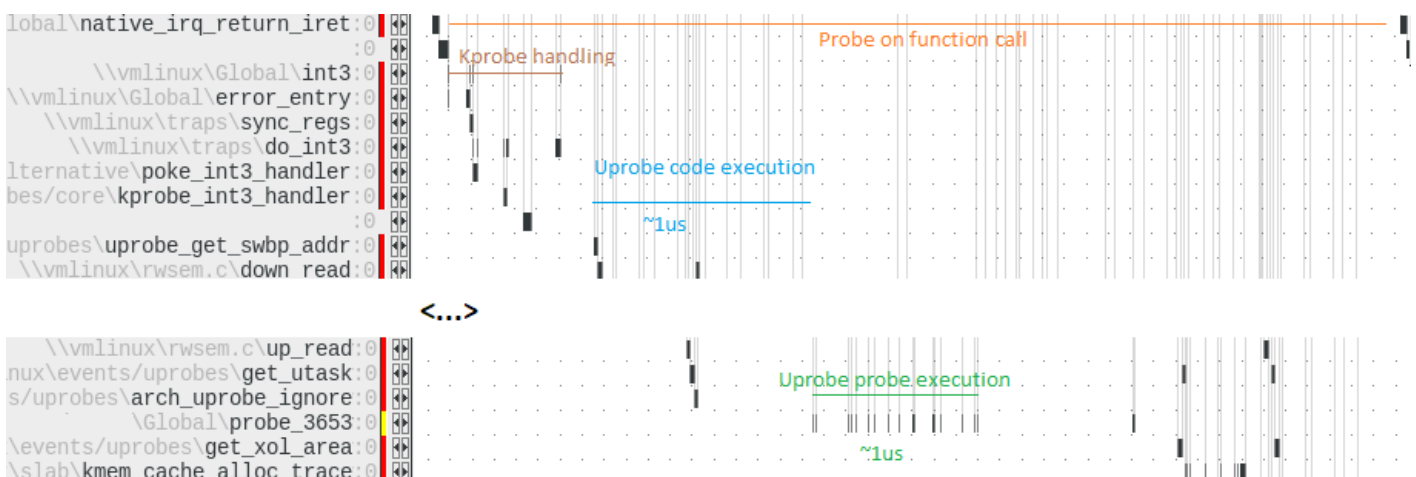


Figura 1: chiamata a uprobe su processori Intel

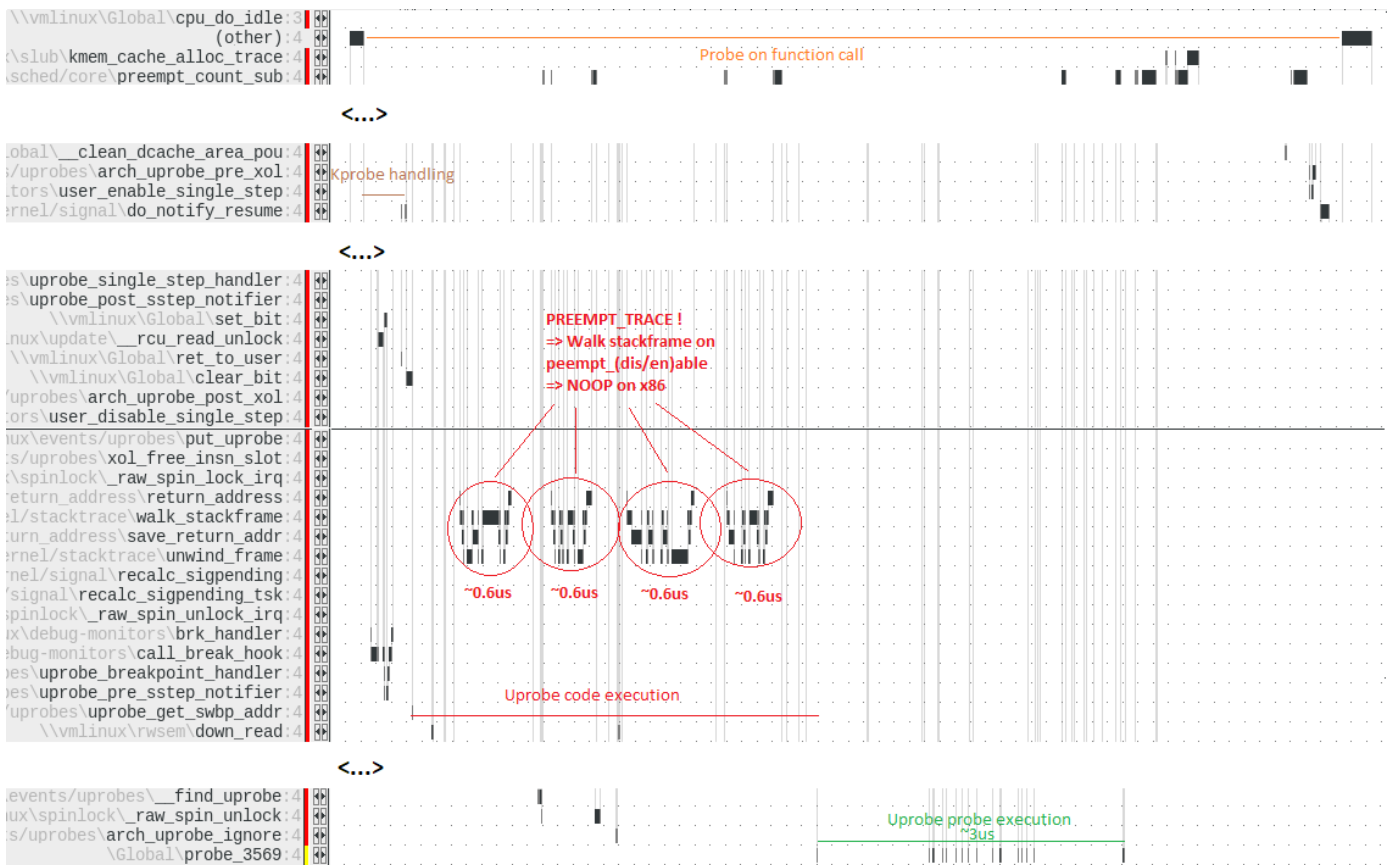


Figura 2: chiamata a uprobe su Arm

Riferimenti e documentazione

TRACE32 RTOS debugger per linux

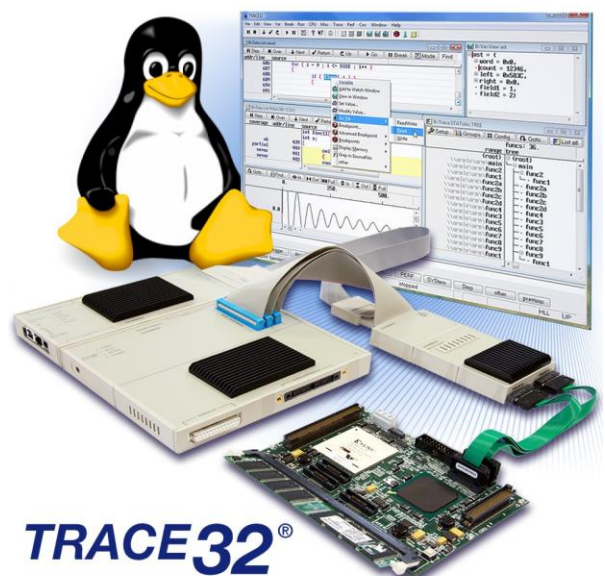
- web www.lauterbach.com/rtoslinux.html
- pdf www.lauterbach.com/doc/rtoslinux.pdf

TRACE32 Webinars & Tutorials (ITA)

- debug: www.lauterbach.com/tut-i_linux.html
- trace: www.lauterbach.com/tut-i_linux_trace.html

TRACE32 Linux debugging reference card

- pdf www.lauterbach.com/linux_card1_web.pdf



TRACE32[®]

I prodotti e servizi LAUTERBACH qui menzionati, come pure i loro rispettivi loghi sono marchi di fabbrica o marchi di fabbrica registrati di LAUTERBACH. Tutti gli altri nomi di prodotti e servizi menzionati sono marchi di fabbrica delle rispettive aziende.