



Debugging delle applicazioni wireless

By Troy Gentry, Principal Applications Engineer for the Human Interface Devices group per Cypress Semiconductor

Sommario

Le sessioni di debugging delle applicazioni wireless spesso eccedono le capacità del buffer di cattura degli analizzatori e sono soggette alle difficoltà di decodifica dei protocolli seriali. Questo lascia gli ingegneri alle prese con tutte le problematiche associate all'esecuzione di task avide di tempo ma necessarie per cucire insieme e decodificare i dati catturati al fine di analizzare le attività più interne di un singolo evento d'utente. Questo articolo illustra in dettaglio la facilità con cui è possibile eseguire il debugging e la documentazione delle applicazioni wireless utilizzando degli strumenti che operano in ambiente PC.

Introduzione

I processi di acquisizione per il debugging delle interazioni delle applicazioni wireless a livello di sistema - anche di quelle apparentemente più elementari, per esempio un mouse wireless per desktop - spesso eccedono le capacità del buffer disponibile negli apparati di test tradizionali. Anche per eseguire il debugging dei processi più semplici - per esempio l'accoppiamento di due dispositivi wireless attivato da un utente attraverso la pressione di un tasto di uno dei due dispositivi seguita dalla pressione di un altro tasto sull'altro dispositivo - gli analizzatori tradizionali possono risultare totalmente inadeguati. Similmente accade in qualsiasi altra sessione di debugging wireless interactions-based in cui gli eventi dove è necessaria l'analisi si trovano agli estremi di una lunga sequenza di dati catturati, con vuoti apparenti nel mezzo.

Le dimensioni inadeguate del buffer di cattura dell'analizzatore costringono gli ingegneri a onerose operazioni di cucitura e decodifica dei dati. Gli analizzatori PC-based, progettati con le prestazioni necessarie per trasferire su base continua il buffer di cattura nella memoria del computer, permettono non solo di semplificare le attività di cattura dei dati delle sessioni più lunghe ma anche di ridurre il tempo speso nel post-processing e nell'analisi.

Chi analizza per la prima volta lunghe sessioni di dati catturati spesso constata con sorpresa la presenza di eventi a livello di sistema di cui non si era accorto lavorando con gli apparati di test tradizionali.

Decifrare gli impervi protocolli seriali bit-per-bit e byte-per-byte nello stesso modo in cui opererebbe la mente umana è un'operazione soggetta ad errori e richiede tempi eccessivamente lunghi. Le tradizionali strategie di debugging delle applicazioni wireless - ad esempio riempire di comandi di debug il codice applicativo - può alleviare alcuni problemi ma a spese di dannosi effetti collaterali quali la lievitazione delle dimensioni del codice, le inconsistenze di esecuzione, la confusione e l'uso di risorse hardware aggiuntive. Gli analizzatori PC-based equipaggiati con flessibili dotazioni di decodifica bus-level permettono di ridurre notevolmente il tempo di debugging cycle-to-cycle decodificando e visualizzando accuratamente le grosse quantità di dati catturati nel buffer al semplice click di un mouse.

Una decodifica bus-level rapida ed efficiente consente di rendere più efficiente il lavoro degli ingegneri trasformando quella che sarebbe un'onerosa e pesante attività di decodifica bit-level in fluide sessioni di debugging results-driven. Una volta che vrete utilizzato la decodifica automatica bus-level per risolvere velocemente i bug delle interazioni complesse a livello di sistema delle applicazioni wireless vi sentirete come se vi avessero tolto un peso dalle spalle. Questo - perché no? - vi renderà anche più propensi a documentare i risultati in un memo. A questo punto infatti i dettagli sono veramente a portata di mano.

Gli apparati di test tradizionali offrono agli ingegneri una succinta dotazione di opzioni di "Salva come" per l'analisi e il post-processing dei dati catturati. A livello routinario, questi analizzatori salvano il buffer dei dati catturati in formati proprietari: spesso questo comporta la perdita di informazioni di debug vitali. Abituamente, gli ingegneri sono costretti a sprecare ore e ore a convertire in formati ASCII e a importarli nel software PC-based per le analisi e il post-processing finale interi buffer di dati catturati su antiquati floppy disk. Gli analizzatori PC-based memorizzano nativamente nell'hard drive del computer i dati catturati nel buffer direttamente, facilitando l'importazione delle informazioni in spreadsheet quali Microsoft Excel per il post-processing e il trattamento.

Tali sistemi garantiscono contemporaneamente la possibilità di riprodurre nel loro formato originario i dati catturati utilizzando strumentazione virtuale PC-based.

Oltre a questo, le applicazioni wireless sono materia per esperti e interessano quindi varie discipline ingegneristiche. La riduzione del tempo di post-processing dei dati catturati rappresenta un aspetto importante nel debugging interaction-based delle applicazioni wireless in quanto permette di incrementare la quantità di tempo speso in queste attività dagli ingegneri specializzati nelle varie discipline.



La possibilità di riprodurre dei dati noti come corretti e di confrontarli con quelli nel buffer del sistema di debugging mette a disposizione uno strumento inestimabile. La riproduzione può anche essere utilizzata come tool di formazione per comprendere meglio e documentare le interazioni e le dipendenze delle applicazioni wireless che possono manifestarsi sul campo.

Il debugging delle applicazioni wireless sul campo rappresenta spesso una sfida anche per gli ingegneri più brillanti. Portarsi in valigia pesanti apparati di test durante i viaggi da un aeroporto all'altro è un incubo per molti tecnici. Parallelamente, il rituale legato alla riorganizzazione del test bench per "l'ultimo debug della settimana" talvolta richiede più tempo rispetto a quello necessario per catturare i dati. Gli analizzatori PC-based progettati per sfruttare i vantaggi delle risorse offerte dal vostro PC sono spesso abbastanza piccoli per poter essere utilizzati sul tavolino del sedile dell'aereo. Questa affermazione merita una ripetizione. Gli analizzatori PC-based sono spesso abbastanza piccoli per poter essere utilizzati sul tavolino del sedile dell'aereo. Suppongo che questo possa solo significare che il cielo non è un limite.

Cypress Semiconductor
198 Champion Court
San Jose, CA 95134-1709
Phone: 408-943-2600
Fax: 408-943-4730
<http://www.cypress.com>

© Cypress Semiconductor Corporation, 2007. The information contained herein is subject to change without notice. Cypress Semiconductor Corporation assumes no responsibility for the use of any circuitry other than circuitry embodied in a Cypress product. Nor does it convey or imply any license under patent or other rights. Cypress products are not warranted nor intended to be used for medical, life support, life saving, critical control or safety applications, unless pursuant to an express written agreement with Cypress. Furthermore, Cypress does not authorize its products for use as critical components in life-support systems where a malfunction or failure may reasonably be expected to result in significant injury to the user. The inclusion of Cypress products in life-support systems application implies that the manufacturer assumes all risk of such use and in doing so indemnifies Cypress against all charges.

PSoC Designer™, Programmable System-on-Chip™, and PSoC Express™ are trademarks and PSoC® is a registered trademark of Cypress Semiconductor Corp. All other trademarks or registered trademarks referenced herein are property of the respective corporations.

This Source Code (software and/or firmware) is owned by Cypress Semiconductor Corporation (Cypress) and is protected by and subject to worldwide patent protection (United States and foreign), United States copyright laws and international treaty provisions. Cypress hereby grants to licensee a personal, non-exclusive, non-transferable license to copy, use, modify, create derivative works of, and compile the Cypress Source Code and derivative works for the sole purpose of creating custom software and or firmware in support of licensee product to be used only in conjunction with a Cypress integrated circuit as specified in the applicable agreement. Any reproduction, modification, translation, compilation, or representation of this Source Code except as specified above is prohibited without the express written permission of Cypress.

Disclaimer: CYPRESS MAKES NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH REGARD TO THIS MATERIAL, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. Cypress reserves the right to make changes without further notice to the materials described herein. Cypress does not assume any liability arising out of the application or use of any product or circuit described herein. Cypress does not authorize its products for use as critical components in life-support systems where a malfunction or failure may reasonably be expected to result in significant injury to the user. The inclusion of Cypress' product in a life-support systems application implies that the manufacturer assumes all risk of such use and in doing so indemnifies Cypress against all charges.

Use may be limited by and subject to the applicable Cypress software license agreement.