



HID Wireless – state seguendo lo standard per sviluppare un altro prodotto “medio”?

Steve Kolokowsky, Cypress Semiconductor Corp.

Sommario

La customizzazione può aiutare a ridurre i consumi, a conseguire efficienza sui dati e a tagliare i tempi associati ai cicli di sviluppo. La standardizzazione permette invece di allentare la curva di apprendimento, di ridurre i rischi di progetto e di eliminare alcuni step nel processo di sviluppo. Quale dei due metodi è più indicato per la realizzazione di un sistema Wireless USB?

Introduzione

Quando si parla di standard Wireless USB esistono due opzioni: seguire lo standard wireless prevalente o sviluppare una soluzione proprietaria ottimizzata per la vostra applicazione.

Siete abbastanza audaci per ricorrere a una soluzione proprietaria con la promessa di un vantaggio concorrenziale sul mercato o la parola “proprietaria” associata al vostro prossimo progetto vi spaventa?

Gli standard sono ovunque. Essi consentono alle persone di comunicare e interagire tra loro. Alcuni standard sono universali: per esempio, la luce rossa dei semafori è sempre in alto, la verde sempre in basso. Altri standard sono arbitrari, per esempio il senso di circolazione del traffico. Per scrivere questo articolo è stata utilizzata una tastiera QWERTY, brevettata nel lontano 1868. Questo layout della tastiera non è più efficiente di altri ma nelle macchine per scrivere del diciannovesimo secolo permetteva di evitare il blocco dei martelletti. La tastiera QWERTY sarà mai sostituita con layout più efficienti? Probabilmente no. I costi associati all’addestramento degli operatori e al retrofitting di tutte le tastiere del mondo occidentale non lo giustificerebbero. Questo dimostra la potenza degli standard.

Come tutte le industrie, anche quella delle tecnologie prevede due tipi di standard: aperti e proprietari.

Uno standard aperto nasce in un’organizzazione di individui o in un gruppo che sviluppa consenso attorno a delle specifiche per l’implementazione di una serie di requisiti comuni. Gli standard aperti permettono a chiunque fosse interessato di utilizzare - per lo sviluppo del proprio prodotto - le relative “regole e norme”. Nel mondo wireless, standard di questo tipo sono 802.11, Bluetooth e Zigbee.

Gli standard proprietari normalmente evolvono da una linea di prodotto o da uno specifico sviluppo tecnologico di un fornitore. Lo standard proprietario è basato sulla tecnologia sviluppata dal singolo in forma proprietaria o esclusiva.

Nel mercato delle moderne reti Wireless PAN (Personal Area Network) esistono vari standard che permettono la comunicazione tra il PC e il dispositivo senza fili.

Protocol	Wireless interface	PC interface
WiFi	Standard (802.11)	Proprietary
WirelessUSB	WirelessUSB	Standard: USB HID
Bluetooth	Bluetooth	Standard: Bluetooth HCI

TABELLA 1 - Standard wireless e interfacce PC

Wi-Fi (802.11) offre una connessione largabanda e viene principalmente utilizzato per il networking wireless. Wireless USB prevede una connessione a banda ridotta per dispositivi HID e altre periferiche a bassa velocità. Bluetooth, oltre ad essere ampiamente sostenuto dalle applicazioni di connettività relative ai telefoni cellulari, viene sfruttato anche per mouse e tastiere wireless.

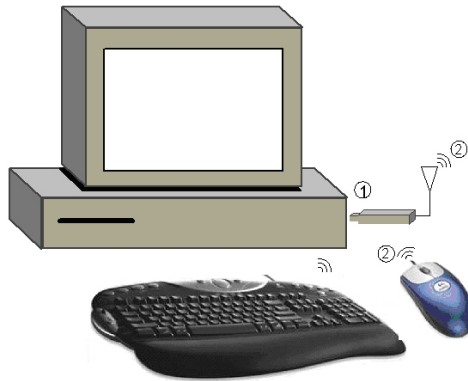


FIGURA 1: Mouse e tastiera wireless (tipici HID)

In questa applicazione sono due le aree dove è possibile standardizzare la connessione wireless al PC. In Figura 1, il numero “1” evidenzia la connessione PC, che normalmente è USB o PC-Card. Il numero “2” identifica la connessione wireless. Nell’ambito dell’applicazione, i progettisti devono preoccuparsi delle funzionalità hardware (chip mouse, USB, radio), del firmware (funzioni dei tasti del mouse) e del software (driver, programmi software del PC). La complessità della standardizzazione di ciascuno di questi elementi determina la complessità del sistema. Se la connessione PC è standardizzata, è possibile utilizzare dei driver di classe per controllare la scheda adapter wireless, denominata talvolta “dongle” wireless. Se la connessione wireless è standardizzata, dongle e dispositivi di produttori differenti possono interagire tra loro.

La capacità di lavorare in “plug and play” in qualsiasi ambiente è sicuramente un attributo positivo tipico dei dispositivi standardizzati. Questa interoperatività rappresenta un aspetto importante per i produttori e gli utilizzatori di dispositivi. I produttori possono infatti concentrarsi sui loro progetti e sulla loro capacità produttiva mentre i consumatori possono contare su una scelta decisamente maggiore.

Uno svantaggio significativo degli standard è dato dal fatto che possono gravare l’applicazione con una serie di funzionalità spesso non necessarie. Se lo standard è stato progettato all’insegna della modularità, è possibile rimuovere le funzioni non indispensabili. In molti casi però, anche il set minimo è troppo oneroso. Nelle applicazioni di networking - dove i prodotti di costruttori differenti devono comunicare tra loro - le soluzioni standard-based rappresentano normalmente la soluzione ideale. Ma nei prodotti che rivestono il ruolo di punti di connessione stand-alone, le soluzioni di rete complesse possono risultare sovradimensionate e controproducenti rispetto a uno sviluppo più semplice, in cui il progettista evita gli oneri legati a protocolli di rete troppo complicati.

Nelle applicazioni wireless HID, Bluetooth rappresenta un buon esempio di questo tipo di complessità. Esaminiamo un’implementazione Bluetooth rispetto a un’implementazione WirelessUSB di tipo custom. Le implementazioni Bluetooth richiedono da 15K a 64K di spazio ROM solo per ospitare le funzionalità base. Una implementazione WirelessUSB completa per un mouse richiede 5.5K, compreso il codice per la gestione dei tasti e dei sensori.

A questo punto vale la pena di aprire un capitolo su WirelessUSB:

WirelessUSB vs. Certified Wireless USB– Qual è la differenza?

C'è un po' di confusione tra WirelessUSB™ e Certified Wireless USB. Certified Wireless USB è il solo standard wireless USB sostenuto dall'USB Implementer's Forum, gruppo proprietario dello standard USB. Esso gira a 480MBit/secondo e copre distanze fino a tre metri dall'host. 480MBit/secondo è esattamente la stessa velocità di high-speed USB. Per il Certified Wireless USB sono richiesti driver speciali: Microsoft dispone già di prodotti specifici, anche se in versione alpha. I chip Certified Wireless USB sono attualmente in fase dimostrativa e per il momento non sono disponibili in volumi. Il mercato target per il Certified Wireless USB abbraccia tutti i dispositivi USB, ma il focus si concentra sulle stazioni docking virtuali, dove cioè un laptop è in grado di connettersi con più periferiche senza ricorrere ad alcun cavo.



WirelessUSB™ fa riferimento alla linea di prodotti di interfaccia USB a basso consumo da 1MBit/secondo di Cypress Semiconductor. Questi prodotti operano nella banda ISM dei 2.4 GHz ed offrono un range da 10 metri (massimo a 1 Mbit/s) a 50 metri (massimo a 62.5 kbit/s) ed utilizzano l'attuale infrastruttura di driver USB: non sono richiesti driver speciali. I prodotti WirelessUSB™ sono sul mercato da molti anni e sono proposti da fornitori quali Logitech e IBM. Essi sono utilizzati principalmente per i dispositivi HID (tastiere, mouse, controller per videogiochi) e per dispositivi audio quali i terminali VOIP. I dispositivi WirelessUSB™ sono molto più economici e consumano decisamente meno rispetto ai prodotti Certified Wireless USB.

Molti dei problemi associati allo spazio necessario per il codice sono originati dal fatto che Bluetooth è stato pensato come standard per reti dove i dispositivi possono "scoprirsi" a vicenda; la connettività dei prodotti Bluetooth è inoltre utilizzabile in qualsiasi industria e tutti i fornitori mondiali possono progettare prodotti compatibili.

Per onorare questi principi e per gestire le differenti specifiche, il protocollo Bluetooth è di conseguenza molto complesso.

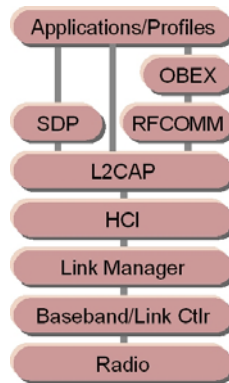


FIGURA 2: Stack di protocollo Bluetooth

Sfortunatamente, utilizzare uno stack di protocollo più complesso implica degli schemi di trasferimento dati più complessi. Nella figura 3, la differenza dei pacchetti dati tra WirelessUSB e Bluetooth è evidente. Ciò che illustra la figura è che – pur utilizzando lo stack Bluetooth nella sua forma più semplice - esso non è in grado di competere con la semplicità di una soluzione proprietaria appositamente progettata per il mercato HID.

A causa della natura del mercato Wireless HID, la differenza in termini di lunghezza dei dati è significativa. Ogni volta che i dati sono trasferiti, il dispositivo consuma energia ed è suscettibile ai disturbi introdotti dalle interferenze. Una scarsa durata delle batterie o la perdita di pacchetti dati possono avere un impatto negativo sulle aspettative degli utenti.

Data Byte	Sequence ID	Checksum
-----------	-------------	----------

A. Minimum Wireless USB packet

Address	Control	Length	FCS	Length (2 bytes)	Channel ID (2 bytes)	Data Byte
RFCOMM			L2CAP			Application

B. Minimum Bluetooth Packet

FIGURA 3: Lunghezza minima pacchetti WirelessUSB e Bluetooth

Un'importante ragione legata alla definizione di un protocollo custom riguarda la differenziazione. Benché gli standard assicurino un buon livello di certezza del progetto, essi normalmente non offrono molte possibilità di differenziazione. Nel mondo dei progetti HID, gli standard limitano l'innovazione e le possibilità di scelta. Alcune aree del mercato wireless HID dove la differenziazione è auspicabile sono:

- range e immunità alle interferenze
- Consumi
- sicurezza

L'immunità alle interferenze è un aspetto importante in questi giorni, specialmente nella banda libera dei 2.4GHz. Ormai sembrerebbe che in qualsiasi punto del territorio sia installato un hot spot WiFi. Sempre più dispositivi e telefoni sono equipaggiati con Bluetooth. I telefoni cordless e i forni microonde sono fonti imprevedibili di interferenze mentre Zigbee sta muovendo i primi passi. Affidandosi alle soluzioni standard, il livello di immunità alle interferenze è fisso. Nell'area dell'immunità, Bluetooth utilizza la tecnica FHSS (Frequency-Hopping Spread Spectrum) e suddivide la banda ISM a 2.4 GHz in 79 canali a 1 MHz. I dispositivi Bluetooth saltano tra i 79 canali 1600 volte al secondo in modalità pseudo-random. WirelessUSB utilizza un segnale radio simile a Bluetooth ma sfrutta una tecnica DSSS invece che FHSS. Ciascun canale WirelessUSB ha un'ampiezza di 1 MHz, consentendo di suddividere la banda ISM a 2.4 GHz in 79 canali da 1 MHz, come Bluetooth. I dispositivi WirelessUSB sono però più agili dal punto di vista della frequenza. In altre parole utilizzano un canale "fisso" ma cambiano dinamicamente canale se la qualità del link scende sotto una soglia ottimale a causa di interferenze o altri fattori. Questa scelta di canale "fisso" consente a WirelessUSB di allocare dinamicamente le aree di spettro più "calme" e utilizzarle in modo efficace. Poiché lo sviluppatore può controllare entrambe i lati del sistema per adattarli al proprio ambiente, questo canale "fisso" può essere adattato a esigenze applicative specifiche.

Oltre ad essere agile dal punto di vista della frequenza, WirelessUSB vanta un metodo unico per la correzione degli errori. WirelessUSB utilizza dei codici pseudo-noise (PN) per codificare i bit d'informazione in vari "chips". La maggior parte dei sistemi WirelessUSB utilizza due codici PN a 32-chip, consentendo la codifica di due bit d'informazione in un simbolo a 32-chip. Questo schema permette di correggere fino a tre errori di chip per simbolo e di rilevare fino a 10 errori di chip per simbolo. Questo "guadagno di codifica" consente a WirelessUSB non solo di funzionare con meno errori e perdite di pacchetti ma anche di coprire distanze più lunghe. Per i dispositivi a banda ridotta - quali i telecomandi wireless - è possibile coprire range ancora superiori utilizzando 64 chip per bit. Il codice PN WirelessUSB permette inoltre di trasmettere utilizzando meno energia.

Nel mondo HID, esistono pochi aspetti progettuali più importanti della vita delle batterie. Questo è il primo tema che la maggior parte dei progettisti deve affrontare. Gli utilizzatori non amano cambiare spesso le batterie. La vita delle batterie è influenzata dall'implementazione del protocollo, dai tempi associati al trasferimento dati e dal livello di ottimizzazione del firmware. I dispositivi Bluetooth - per esempio - devono sincronizzarsi con la rete su base regolare per "scoprire" la presenza di nuovi dispositivi. In applicazioni in cui i dispositivi entrano ed escono regolarmente dalla rete, la "scoperta" rappresenta una funzione vitale. Nei sistemi multipunto-punto a basso costo, lo svolgimento del protocollo può essere talvolta responsabile di un consumo superiore a quello necessario per trasmettere i dati!

Se un dispositivo è stato progettato per trasmettere i dati in modo efficiente, esso spenderà meno tempo per entrare e uscire dalle modalità trasmissione o ricezione, spenderà meno tempo per inviare nell'etere i dati correnti e non sarà troppo sensibile alla vita delle batterie. La scelta di un sistema proprietario ottimizzato per queste caratteristiche può garantire - a livello di consumi - dei vantaggi significativi.

I progetti HID wireless fanno riferimento a mercati unici. Dal punto di vista tecnologico, lo sviluppo a livello radio, di bandabase e di protocollo impone delle problematiche sofisticate. Energia, sicurezza e risultati d'uso sono aspetti altrettanto importanti. Su tutto, vita delle batterie e prezzo regnano sovrani. Per questo, sviluppare un sistema sofisticato ma semplice, complesso ma facile da usare, avanzato ma economico non è un compito semplice. Gli sviluppatori devono fronteggiare una serie di dure decisioni quali:

- 1) scegliere o meno uno standard aperto che da un lato garantisca una tecnologia collaudata e diffusa, dall'altro non gravi il progetto con oneri e complessità non necessarie per le applicazioni più semplici
- 2) scegliere o meno una soluzione proprietaria ottimizzata per una specifica applicazione, andando però incontro ai rischi tipici di una tecnologia single-vendor in un mondo in costante evoluzione.

I progettisti - infine - possono decidere se essere inseguitori o trascinatori, se essere uguali agli altri o differenziarsi, se essere prudenti o innovativi. Voi come siete? Il vostro prossimo progetto potrà rivelarvelo!



Reference

Avoiding Interference in the 2.4-GHz ISM Band, EETimes CommsDesign Feb 2005

An efficient 2.4-GHz wireless net for less, EETimes, October, 2003

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.