Introduction
KISSBind provides a simple and intuitive method that allows a WirelessUSB™ Peripheral to bind to a nearby WirelessUSB Bridge without pressing a dedicated bind button on either device. This document assumes the reader to be familiar with the WirelessUSB technology. For more information on WirelessUSB please refer to WirelessUSB LS Theory of Operation, and WirelessUSB LS 2-Way HID Systems application notes.

Firmware

Peripheral
The KISSBind code is located in the “bind.c” source file for the WirelessUSB Mouse and Keyboard RDK. The #define KISS_BIND in the source file “ls_config.h” enables the peripheral KISSBind algorithm.

Bridge
The KISSBind code is located in the “bind-kiss.asm” source file for the Bridge in the CY4632 reference design kit. The #define KISS_BIND_SUPPORT in the source file “config.inc” enables the bridge KISSBind algorithm. The Bridge can support KISSBind and the previous bind algorithm in the same build (enabling KISS_BIND_SUPPORT does not disable the previous bind algorithm).

Theory of Operation

Peripheral
If the peripheral tries to send a report to a bridge and it fails, the peripheral tries to reconnect to the bridge. If the reconnect fails (the bridge is not active or the device has not been previously bound), the peripheral loops sending KISSBind Requests on all channels and PN codes. After a sending the KISSBind Request on all channels with a PN code, the peripheral sends Bind Requests on channel 77 using that PN code, with the PA set to zero. Only a bridge in close proximity will receive the Bind Request with the appropriate RSSI level. The device is bound when a valid Bind Response is received from a bridge. This process is repeated on all PN codes if a Bind Response is not received. Figure 1 shows a listener trace of the KISSBind Requests for a CY4632 reference design kit keyboard.

Bridge
If a bridge receives a KISSBind request, it changes to channel 77 and listens for a Bind Request. If a Bind Request with a minimum RSSI level is received an appropriate Bind Response is sent. The device will then be bound to the bridge. Figure 2 shows a listener trace of the Bind Requests and Bind Response for a CY4632 reference design kit keyboard. Note: The version of listener used for the traces displays the packet types for the KISSBind Request as “UNKNOWN”, the Bind Request as “BIND RSP” and the Bind Response as “BIND REQ”. Figure 3 illustrates the KISSBind sequence and the Peripheral and Bridge binding process.

References
- WirelessUSB LS Theory of Operation App Note
- WirelessUSB LS 2-Way HID Systems
- WirelessUSB LS Development Kit User’s Guide
- Device Class Definition for Human Interface Devices (HID)
- (http://www.usb.org/developers/hidpage)
- CYWUSB6934 Datasheet
Figure 1. Listener Trace KISSBind Requests
Figure 2. WirelessUSB Listener Trace Bind Request/Bind Response
Figure 3. KISSBind Sequence Diagram

[Diagram of KISSBind Sequence Diagram showing the interaction between Device and Bridge.]