

**USB'S DAYS AS A HOST-CENTRIC BUS ARE NUMBERED. USB ON-THE-GO TECHNOLOGY WILL SOON ENABLE USB PERIPHERALS TO COMMUNICATE WITHOUT A HOST PC. THE NEW CAPABILITIES WILL EXTEND TO A RAFT OF CONSUMER PRODUCTS WHAT IS ALREADY BECOMING THE PC WORLD'S MOST POPULAR EXTERNAL BUS.**

# Understanding USB On-The-Go

**T**HE USB SPECIFICATION INTRODUCED a simple and inexpensive infrastructure for easily connecting multiple external peripherals to a PC. That was several years ago; today, more than 500 million peripheral devices are designed with USB ports, making USB the market's dominant I/O-connectivity standard. The widespread availability of USB peripherals is now driving non-PC applications, such as mobile, handheld, and embedded post-PC applications, to adopt USB for direct-I/O connections. In addition, many devices that have traditionally functioned as peripherals now require direct connections to other devices. USB's greatest limitation—its lack of support for point-to-point communication between devices—has deterred its use in consumer-electronic devices, such as mobile phones and PDAs. However, these devices are gaining popularity and intelligence, increasing the need for direct connections among them. The answer to this requirement is in a developing standard called USB OTG (USB On-The-Go), a supplement to the USB specification that eliminates the requirement for a PC to act as host in exchanges of data among connected devices.

There has been much speculation—and, unfortunately, some misinformation—regarding USB OTG. The OTG specification's goal is to enhance certain USB peripherals to enable them to also act as hosts for a selected set of peripherals. OTG introduces point-to-point communication between these enhanced USB peripherals. OTG's complexity comes from the requirement that these dual-role devices be able to directly exchange data with each other. Moreover, because the specification targets portable devices with significant power constraints, the initial OTG host must supply only a small amount of power.

Two connected devices take turns functioning as the bus host and the peripheral, thus maintaining the current USB host/peripheral architecture model. The OTG host always initiates communication with a normal bus-enumeration process (bus reset, acquisition of USB descriptors, and peripheral-device configuration). After these steps, the device serv-

ing as OTG host may transfer data to and from an OTG device performing as a peripheral. The OTG specification defines a mechanism for exchanging the roles of OTG host and peripheral. The initial role of each device is defined by which mini plug a user inserts into its receptacle.

There are two types of OTG devices: dual-role devices and peripheral-only devices. A dual-role device can function as either a USB peripheral or an OTG host. Dual-role devices must be able to supply at least 8 mA on  $V_{BUS}$ . Though a peripheral-only device has no host capabilities, it must be able to request a dual-role device to communicate with it. Dual-role devices must minimally operate at full speed. Operation at high speed is optional. Peripheral-only OTG devices may operate at any of USB's three signaling rates.

OTG dual-role devices support only a subset of all USB devices (potentially, only one). Supported devices may be other instances of the same OTG dual-role device, other dual-role devices, or self-powered USB peripherals that the dual-role device can power. The dual-role device must provide drivers for the listed supported devices. For example, an OTG still-image camera may have a driver for a specific printer or for a set of printers. In such a case, a user will be able to print from that camera to only the specific printer or the set of printers; it is unlikely that he or she will be able to print to every USB printer. The dual-role device's vendor must provide drivers for all of the supported devices. An OTG-device vendor that claims to support a class of devices, such as keyboards, that meet the power requirements for OTG peripherals must supply the corresponding class driver.

## CABLES AND CONNECTORS

The USB 2.0 specification defines three types of connector pairs (plugs and receptacles): the standard-A, the standard-B, and the mini-B. The mini-B connector was developed for smaller peripherals, such as mobile phones. Because work on OTG be-

gan before completion of this mini connector, a new pin, the ID pin, was added to the plug in anticipation of OTG's needs and was left unconnected in the mini-B plug.

The OTG specification introduces a fourth plug, the mini-A, and two receptacles, mini-A and mini-AB. The differences between the mini-A and mini-B connectors are as follows. First, each plug is keyed so that the mini-A plug does not fit into a mini-B receptacle, and a mini-B plug does not fit into a mini-A receptacle. However, the mini-AB receptacle accepts either a mini-A or a mini-B plug. Second, the mini-A plug's ID pin is shorted to ground. The overmoldings of the two plugs are different; the mini-A plug has a more oval shape, whereas the mini-B plug is more square. Last, the plastic inside the plugs and receptacles is color-coded. Inside the mini-A plug and receptacle, the plastic is white; inside the mini-B plug and receptacle, the plastic is black; and inside the mini-AB receptacle, the plastic is gray.

The USB 2.0 specification defines two cables: the standard-A to standard-B and the standard-A to mini-B. OTG defines two more cables: the mini-A to standard-B and the mini-A to mini-B. The end-to-end delay of the mini-A-to-mini-B cables has been reduced to allow the use of adapters at the A end of the cable.

Adapters are required to allow for connecting different combinations of devices. For example, connecting a USB device with a captive cable and a standard-A plug to an OTG dual-role device requires a standard-A-receptacle-to-mini-A-plug adapter. Conversely, connecting an OTG device with a captive cable and a mini-A plug to a standard-A port requires a mini-A-receptacle-to-standard-A-plug adapter. The only permitted usage of the mini-A receptacle is on an adapter.

There are no usability issues when users are connecting two dual-role devices. However, when a user is connecting a peripheral-only device to an OTG dual-role device, he or she might plug the mini-B end of the cable into the dual-role device and then try plugging the mini-A end of the cable into the peripheral-only device's mini-B receptacle. Because the mini-B receptacle is keyed to prevent its accepting mini-A plugs, users should let the differences in the cable ends guide

## **BECAUSE THE MINI-B RECEPTACLE IS KEYED TO PREVENT ITS ACCEPTING MINI-A PLUGS, USERS SHOULD LET THE DIFFERENCES IN THE CABLE ENDS GUIDE THEM TO REVERSE THE CABLE CONNECTIONS.**

them to reverse the cable connections. The color-coding inside the plugs and receptacles plus the overmolding requirements for the plugs can also be a guide.

### **DUAL-ROLE DEVICES**

Because each OTG dual-role device incorporates a mini-AB receptacle, a mini-A-to-mini-B cable can directly interconnect the devices. Users will perceive no difference in the devices based on the cable connection. That is, users will not know which device is the initial OTG host and which is initially the peripheral.

A dual-role device's initial role is defined by which end of the cable a user inserts into the device's mini-AB receptacle. The dual-role device with the mini-A plug is the initial OTG host, also known as the A-device. Conversely, the dual-role device with the mini-B plug is the initial peripheral, also called the B-device. The dual-role device determines which end of the cable to insert by whether or not the ID pin is shorted to the GND pin.

The A-device must supply the voltage on  $V_{BUS}$  when a communication session is in progress. This key difference between the A- and the B-devices causes the two connected devices to be unequal and prevents a peer-to-peer connection. Because the A-device supplies the voltage on  $V_{BUS}$  and, therefore, controls when a communication session occurs, the B-device requires a mechanism for requesting a communication session. This mechanism is the SRP (Session Request Protocol).

### **PERIPHERAL-ONLY OTG DEVICES**

Peripheral-only OTG devices are normal USB peripherals, but they also sup-

port the SRP. These devices must have an OTG capability descriptor, which indicates that the device supports the SRP but is not a dual-role device. Last, peripheral-only OTG devices may use only the B receptacles or must have a captive cable with a mini-A plug attached. Peripheral-only OTG devices may not use the mini-AB receptacle.

The SRP is the mechanism for a peripheral-only device and a dual-role device, configured as a B-device, to request a communication session with a dual-role device configured as an A-device. This protocol consists of two signaling methods generated by the B-device: data-line pulsing and  $V_{BUS}$  pulsing.

The A-device must detect one of these two methods and respond by starting a communication session. Two protocols are defined because having them simplifies implementing A-devices, which need to be designed to respond to only one of the two protocols. (There are small advantages and disadvantages in responding to either method.) There is virtually no added cost, other than some extra firmware, in having B-devices generate both methods. Dual-role and OTG peripheral-only devices must be capable of generating the SRP.

First, the B-device executes data-line pulsing; the B-device must pulse its pullup resistor for 5 to 10 msec. The B-device must then drive  $V_{BUS}$  ( $V_{BUS}$  pulsing) for a period long enough to charge a capacitance on  $V_{BUS}$ . The capacitance is smaller than 13  $\mu\text{F}$  to at least 2.1V. OTG devices have a capacitance of 6.5  $\mu\text{F}$  or less but do not charge above 2V a  $V_{BUS}$  capacitance of 96  $\mu\text{F}$  or more. (Standard hosts have a capacitance of 96  $\mu\text{F}$  or more.) This limitation prevents the  $V_{BUS}$  current from the B-device from damaging standard host ports.

### **EXCHANGING DEVICE ROLES**

The OTG specification defines a negotiation protocol, the HNP (Host Negotiation Protocol). This protocol provides a means by which the A- and B-devices can exchange the OTG host and peripheral roles.

Because the A-device is by default the bus host, it supplies the voltage on  $V_{BUS}$ . When the A-device drives, the voltage rises above a valid level, and the B-device asserts its pullup resistor. When detecting a connection, the A-device resets and

enumerates the B-device. The A-device then uses the functions that the B-device provides. Once the A-device finishes using the B-device, it may determine that the B-device is a dual-role device by querying for the B-device's OTG-capability descriptor. If the B-device responds with a valid OTG-capability descriptor that indicates it supports HNP, the A-device generates a set-feature command (HNP\_Enable), which informs the B-device that it may assume the host role once the bus is suspended. (The A-device can also determine that the B-device is a dual-role device by successfully completing this set-feature command.) The A-device then suspends the bus.

The B-device signals that it wishes to assume the host role by signaling a disconnect (deasserting its pullup resistor if operating at full speed or deasserting its pulldown resistors if operating at high speed). In response to the B-device's disconnect, the A-device asserts its pullup resistor and operates as a peripheral. After the B-device resets and enumerates the A-device, the B-device may use the functions that the A-device provides. Once the B-device finishes with the A-device's functions, the B-device suspends the bus and asserts its pullup resistor, returning to operation as a peripheral. Detecting the bus suspend, the A-device signals a disconnect and resumes the host role. If the B-device is a dual-role device, and the A-device doesn't wish to use the B-device's functions, the A-device reissues the same set-feature command and then suspends the bus. If the HNP-enabled B-device fails to signal its request to assume the host role within a specified time frame, the A-device can end the session by dropping  $V_{BUS}$ .

If the B-device initiates a session, the A-device may skip enumeration after reset by issuing the set-feature (HNP\_Enable) command while the B-device is still in the default state. If the command succeeds—that is, if the B-device is a dual-role device—the A-device suspends the bus. If the command fails—that is, if the B-device operates only as a peripheral—the A-device enumerates the B-device and, if possible, services the reason for the session request.

#### DIFFERENCES WITH USB 2.0

Because portable devices are typically battery-powered, forcing such devices to

source a relatively large amount of current would result in very short battery life. The current-driving capability is often unnecessary anyway, because the device connected to the portable device is often self-powered. Instead of providing 100 or 500 mA at its port, a dual-role device must provide at least 8 mA. Dual-role devices are allowed to provide more current to accommodate additional peripherals but are not required to do so.

Another key difference is the  $V_{BUS}$  capacitance at the connector. OTG dual-role devices have a capacitance of 1 to 6.5  $\mu\text{F}$ . Standard hosts have a capacitance of 96  $\mu\text{F}$  or more. This difference is significant during the SRP. When an OTG device performs  $V_{BUS}$  pulsing, it drives a charge onto the  $V_{BUS}$  line through either a constant current source or a resistor. If the OTG device is connected to a stan-

### THE USB OTG SPECIFICATION INTRODUCES NEW MECHANISMS TO ENABLE NEW APPLICATIONS FOR USB PERIPHERALS.

dard host, the voltage will not be driven above 2V, thereby avoiding any possible damage to the host. However, because OTG devices have much smaller capacitance than standard hosts, another OTG device will have its  $V_{BUS}$  line driven above 2.1V.

Section 7.1.2.2 of the USB 2.0 specification defines four test points corresponding to the connectivity from host silicon (TP1) to host connector (TP2) to peripheral connector (TP3) to peripheral silicon (TP4). The maximum allowable delay is 30 nsec. Section 7.1.16 of the USB 2.0 specification limits the cable delay (from TP2 to TP3) to 26 nsec. Because of the possible use of adapters, which are allowed a 1-nsec delay, any cable with a mini-A plug is limited to a cable delay of 25 nsec. Using a 5m standard-B-to-standard-A cable with a standard-A-receptacle-to-mini-A-plug adapter violates, by 1 nsec, the 26-nsec limitation. However, when operating as peripherals, OTG devices have an only 1-nsec delay between TP3 and TP4, and, when operating as OTG hosts, a 1-nsec delay between TP1

and TP2. Therefore, the 30-nsec delay between host and peripheral silicon is preserved.

#### COMPLIANCE PROGRAM

A compliance-specification parallel to the OTG specification is under development. The document will define a compliance program that includes OTG-device role-exchange-support testing, OTG-host signal-quality testing, session-start-protocol-support testing, peripheral-only-device power-constraint testing and end-user-experience testing. The USB-IF's (USB Interface Forum's) current USB 2.0 peripheral-compliance program is a precursor to the future compliance program. Because OTG devices are USB peripherals with the additional ability to directly communicate with other OTG devices, they must pass both the USB-IF-compliance program and the OTG-compliance program to be considered OTG-compliant.

The USB OTG specification introduces new mechanisms to enable new applications for USB peripherals. Only the drivers that the OTG dual-role device contains limit the number of combinations of OTG devices that users can connect. The introduction of OTG will not result in every OTG dual-role device communicating with all other OTG devices. Initially, it will support only a small number of devices. However, the list of supported OTG dual-role devices will quickly grow. In the future, vendors of OTG dual-role devices will likely include a mechanism for downloading drivers to their devices to expand the number of supported devices. □

#### ACKNOWLEDGMENT

*The author acknowledges the work and contributions of the following people, who helped to draft and edit the USB OTG specification: David Murray of TransDimension; David Wooten of Cypress Semiconductor; Zong Liang Wu, PhD, of TransDimension; and Mark Yi of Opti Inc.*

#### AUTHOR'S BIOGRAPHY

*Kosta Koeman is a staff software engineer with Cypress Semiconductor (Woodinville, WA), where he has worked for more than a year. He holds a BSEE from the University of Washington (Seattle) and an MSEE from the Oregon Graduate Institute (Beaverton, OR).*