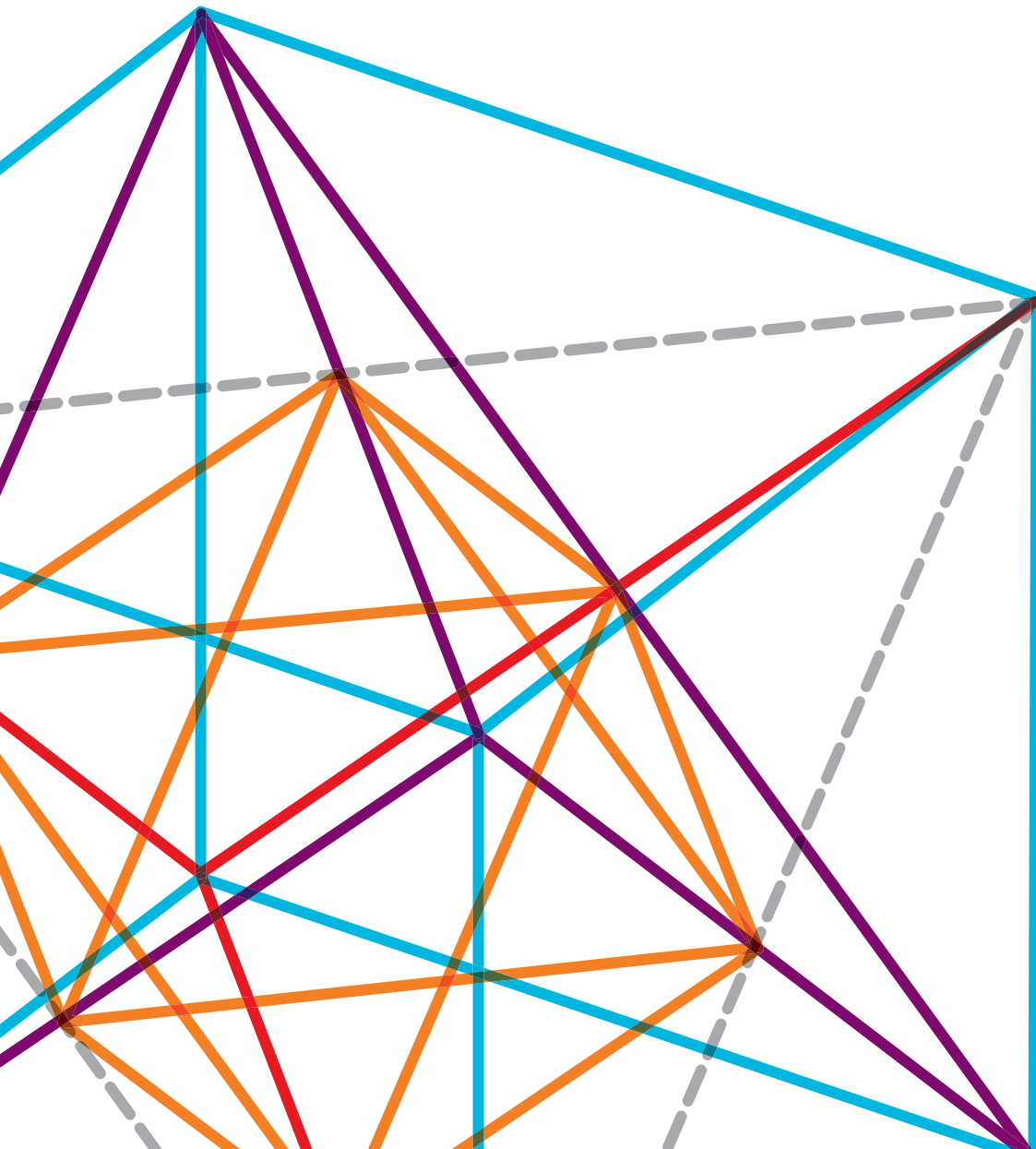


USB On The Go  
AppNote 2456935.971748  
October 5, 2014

---



SEE MORE. DO MORE.



# INTRODUCTION

---

USB On-The-Go (OTG) is a technology that allows a specially configured USB port to act as both a device port and a host port. A device port is what is normally thought of when users think of a USB device – one that connects to a computer or hub to enable communication with the printer. A USB host port is the port a user would typically find on a computer, and to which they would connect devices.

The USB OTG port, with the use of a special cable, enables the port to act either as a host or a device. This cable also provides a standard USB port to which a cable or hub can be connected.

All ZQ500™ models implement the USB OTG port on the printer itself.

## USB OTG SETGETDO (SGD) COMMANDS

---

### **usb.device.manufacturer\_string**

Purpose:           Retrieves the USB device manufacture's name  
Range:           set in the factory  
Default:         set in the factory  
Example:         ! U1 getvar "usb.device.manufacturer\_string"

**NOTE:** The command must be followed by a carriage return/line feed or a space character. This command is only used as a getvar command.

### **usb.otg.enable**

Purpose:           Enables/Disables USB OTG functionality  
Range:           otg, off, device, host  
Default:         otg  
Example:         ! U1 setvar "usb.otg.enable" "otg"

**NOTE:** SGD commands must be followed by a carriage return/line feed or a space character.

The command has four optional values:

- "off" - to completely disable the port
- "device" - to make the port act only as a device port
- "host" - to make the port act as a host port (with the proper cable connected)
- "otg" - to make the port switch automatically based on the cable connected

## usb.otg.status

Purpose: Returns the current mode of the OTG port  
Range: disconnected, device, host  
Default: returns the current mode of the OTG port  
Example: ! U1 getvar "usb.otg.status"

**NOTE:** SGD commands must be followed by a carriage return/line feed or a space character. The command will return:

- "disconnected" if nothing is connected to the USB port
- "device" if the port currently has a device connection
- "host" if the port currently has a host connection

## USB PORTS ON CRADLE

---

If using the optional cradle, there is a USB port that uses the same cable as the OTG port on the body of the printer. This port, however, does not support the OTG functionality, and the above SetGetDo commands

do not apply to it. There is a USB host port on the cradle as well that can only be used as a USB host port.

## USB HOST MIRROR FUNCTIONALITY AVAILABLE ON ZQ500 SERIES

---

USB Mirror provides an easy way to upgrade the application and configuration files of a printer using a USB hard drive or flash drive. FAT16 and FAT32 formatted volumes are supported. NTFS drives are supported for reading only.

At this time, only the USB Mirror functionality is available via the host interface on the ZQ500.

The functionality is identical to that of the ZT400 series of printers. Full documentation is available in the user manual for that printer.

The basic SetGetDo commands for USB Mirror operation are presented below. There are more SGDs for advanced operations available in the user manual.

## usb.mirror.enable

Purpose: Enables/Disables USB Mirror  
Range: on, off  
Default: on  
Example: ! U1 setvar "usb.mirror.enable" "on"

**NOTE:** SGD commands must be followed by a carriage return/line feed or a space character.

### **usb.mirror.auto**

Purpose: Configures automatic Mirroring  
Range: on, off, prompt  
Default: on  
Example: `! U1 setvar "usb.mirror.auto" "on"`

**NOTE:** SGD commands must be followed by a carriage return/line feed or a space character. The effect of the command choices are:

- "on" - Mirroring shall occur automatically when the USB stick is inserted and `usb.mirror.enable` is on
- "off" - Mirroring shall NOT occur automatically when USB stick is inserted
- "prompt" - an acknowledge alert with the message "Start USB Mirroring" and the options of "Start" and "Abort" shall be displayed on the printers LCD

### **usb.mirror.path**

Purpose: Configure path for Mirror files  
Range: settable by the user  
Default: Zebra  
Example: `! U1 setvar "usb.mirror.path" "/zebra"`

**NOTE:** SGD commands must be followed by a carriage return/line feed or a space character. This command is used to configure the location of the folder where the Mirror files are stored on the USB drive. If this folder does not exist, or does not contain all of the sub-folders necessary, the mirror operation will not occur or may partially occur.

By default this is "zebra" - meaning that the Mirror files are placed in the Zebra subdirectory off the root of the USB drive. In this folder, there should be a "files" folder, where there are files to be placed on to the E: drive of the printer, an "appl" folder, which contains the printer OS file to load on to the printer, and the "commands" folder, which contains content to be sent to the printer after the new printer OS has been downloaded, and the new files are in place on the E: drive.

### **usb.mirror.fetch**

Purpose: Start Mirror operation immediately  
Range: sending command causes Mirror event to immediately start  
Default: (does not apply to this command)  
Example: `! U1 do "usb.mirror.fetch"`

**NOTE:** SGD commands must be followed by a carriage return/line feed or a space character. This command is a `do` command, meaning it does not have a `setvar` or `getvar` implementation.

### **usb.mirror.error\_retry**

Purpose: How many times to retry on error (in seconds)  
Range: 0 - 65535  
Default: 0  
Example: `! U1 setvar "ip.mirror.error_retry" "5"`

**NOTE:** SGD commands must be followed by a carriage return/line feed or a space character. This command is used to set how many times the printer will attempt to retry the Mirror operation when an error occurs. Setting this to zero (0) will cause the printer to NOT retry when an error occurs.

### **usb.mirror.last\_error**

Purpose: Causes the printer to return data on the last Mirror error  
Range: getvar command only  
Default: "No Error" - if no Mirror operation has been attempted.  
Example: `! U1 getvar "usb.mirror.last_error"`

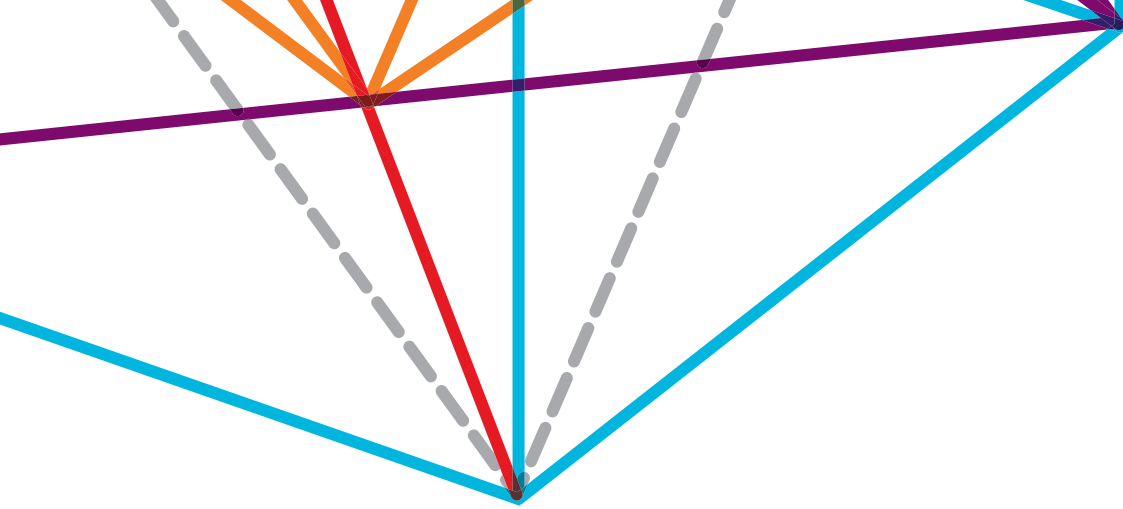
**NOTE:** SGD commands must be followed by a carriage return/line feed or a space character. This getvar only command returns the last error that occurred during a USB Mirror process. By default this command will return "No Error" if no Mirror operation has been attempted.

### **usb.mirror.success**

Purpose: Determines if the last USB Mirror event was successful  
Range: yes, no  
Default: If no USB Mirror operation has been attempted, the default response is "no"  
Example: `! U1 getvar "usb.mirror.success"`

**NOTE:** SGD commands must be followed by a carriage return/line feed or a space character. This read-only command returns "yes" if the last Mirror operation completed was successful; otherwise, it returns "no."

If no USB Mirror operation has been attempted, it returns "no" .



**Corporate Headquarters**

+1 800 423 0442  
inquiry4@zebra.com

**Asia-Pacific Headquarters**

+65 6858 0722  
apacchannelmarketing@zebra.com

**EMEA Headquarters**

+44 (0)1628 556000  
mseurope@zebra.com

**Latin America Headquarters**

+1 847 955 2283  
inquiry4@zebra.com

**Other Locations / USA:** California, Georgia, Illinois, Rhode Island, Texas, Wisconsin **Europe:** France, Germany, Italy, the Netherlands, Poland, Spain, Sweden, Turkey, United Kingdom **Asia Pacific:** Australia, China, Hong Kong, India, Indonesia, Japan, Malaysia, Philippines, Singapore, South Korea, Taiwan, Thailand, Vietnam **Latin America:** Argentina, Brazil, Colombia, Florida (LA Headquarters in USA), Mexico **Africa/Middle East:** Dubai, South Africa