NXP Semiconductors
User's Guide

Document Number: MBOOTSDPHOSTUG
Rev 2, 08/2018

sdphost User's Guide

Chapter 1
Introduction

This document describes the usage of Serial Download Procotol Host (sdphost), a PC host application. The sdphost tool provides a command line interface to send serial download protocol commands to NXP's i.MX RT devices enumerated on the host PC via USB-HID or UART device. ROM code is running on the device in serial download mode. The sdphost tool is a useful tool in the factory programming and manufacturing process. It can be used to test and develop the automation software and test setups. It can be invoked from other applications too.

This document introduces the serial download protocol, typical factory programming setup, and usage of the tool and description of its command line interface. It also provides a set of example usages of sdphost tool and its command line arguments with a device.

Chapter 2
Serial Download Protocol

Serial Download Protocol is a set of commands supported by NXP's i.MX RT devices in the Boot ROM application's serial download mode. See boot chapter in processor reference manual for protocol details. The sdphost tool provides the user with a simple and user-friendly command-line interface.

The purpose of serial download protocol is to provide means to download bootable images from a PC to the device's internal or external RAM memory. There are a set of commands to read and write to a memory/register unit, read status of the last command, download images to a given address in internal/external memory, and provide the address to jump and execute the downloaded image.

Chapter 3
Typical setup

The sdphost tool is used in the development phase of the device firmware application, manufacturing, and factory programming process. The sdphost tool is supported on Windows® OS, Linux® OS, and Mac® OS host environments. The release package consists of all three binaries in the respective Operating System folders.

Test setup would typically be the device connected to PC Host via USB or UART. The sdphost tool would run on the PC host, and the device would run in Boot ROM serial download mode.

The MCU has BOOT_MODE pins that can be used to boot the device in serial downloader mode. The device's reference manual provides the documentation on booting the device in serial downloader mode. See boot chapter in processor reference manual for setup details

Chapter 4
sdphost Command-Line usage

sdphost provides an easy-to-use command line interface. The syntax of the usage text is:

```
sdphost [-?|--help] [-v|--version] [-p|--port <name>[,<speed>]]
[-u|--usb [[[<vid>,]<pid>]]] [-V|--verbose] [-d|--debug]
[-j|--json] [-t|--timeout <ms>] -- command <args...>
```

The following table describes each command-line argument:

Table 1. sdphost command arguments

Argument	Description
-?/--help	Displays the usage syntax and description
-v/--version	Displays tool version
-p/--port <name>[,<speed>]	Connect to target over UART. Specify COM port and optionally baud rate (default=COM1,115200)
-u/--usb [[<vid>,]<pid>]	Connect to target over USB HID device denoted by vid/pid (default=0x15a2,0x0083). The device's reference manual provides the USB-HID device's VID and PID
-V/--verbose	Print extra detailed log information
-d/--debug	Print really detailed log information
-j/--json	Print output in JSON format to aid automation. The last -V/-d/-j takes precedence.
-t/--timeout <ms>	Set packet timeout in milliseconds (default=5000)
SDP Commands:	Following entries show the usage of sdphost to send the SDP commands to the device
read-register <addr> [<format> [<count> [<file>]]]	Read one or more registers at the given address. Data Format must be 8(byte), 16(half-word), or 32(word); default format is 32 bits. Count is number of bytes to read; default count is size of format (i.e. one register). Output file is binary; default is hex display on stdout.
write-register <addr> <format> <data>	Write one register at address. Data Format must be 8 (byte), 16(half-word), or 32(word). Data is data value to write.
write-file <addr> <file> [<count>]	Write file data at address. Count is the size of data to write in bytes; size of file will be used by default.

Table continues on the next page...

Table 1. sdphost command arguments (continued)

error-status	Read error status of last command.
jump-address <addr>	Jump to entry point of image with IVT at specified address.

Chapter 5
Usage example and SDP command description

This section of the user guide provides example usages of sdphost tool and command line arguments.

Here is a list of common arguments that can be used with the SDP command on the command line.

- **The -p argument**:

  This argument is required when the device communicates with the host in SDP mode via UART. The mandatory argument that goes with -p is the COM port number for the connected device. On Windows OS, it can be discovered in Device Manager application under *Ports*.

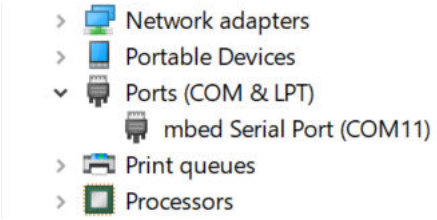

Figure 1. UART connection

  In Linux, in order to obtain a port, the following command can be used - $ dmesg | grep tty

  Connection over UART command - $ sudo ./sdphost -p /dev/ttyACM0 -- read-register 0x0000

  In MAC, in order to obtain a port, the following command can be used - $ ls /dev/{tty,cu}.*

  Connection over UART command - $ sudo ./sdphost -p /dev/cu.usbmodemFA132 -- read-register 0x0000

- **The -u argument**:

  This argument is required when the device communicates with the host in SDP mode via USB-HID. The device's vid and pid should be passed in as the arguments for sdphost to communicate with the correct device among the enumerated devices.

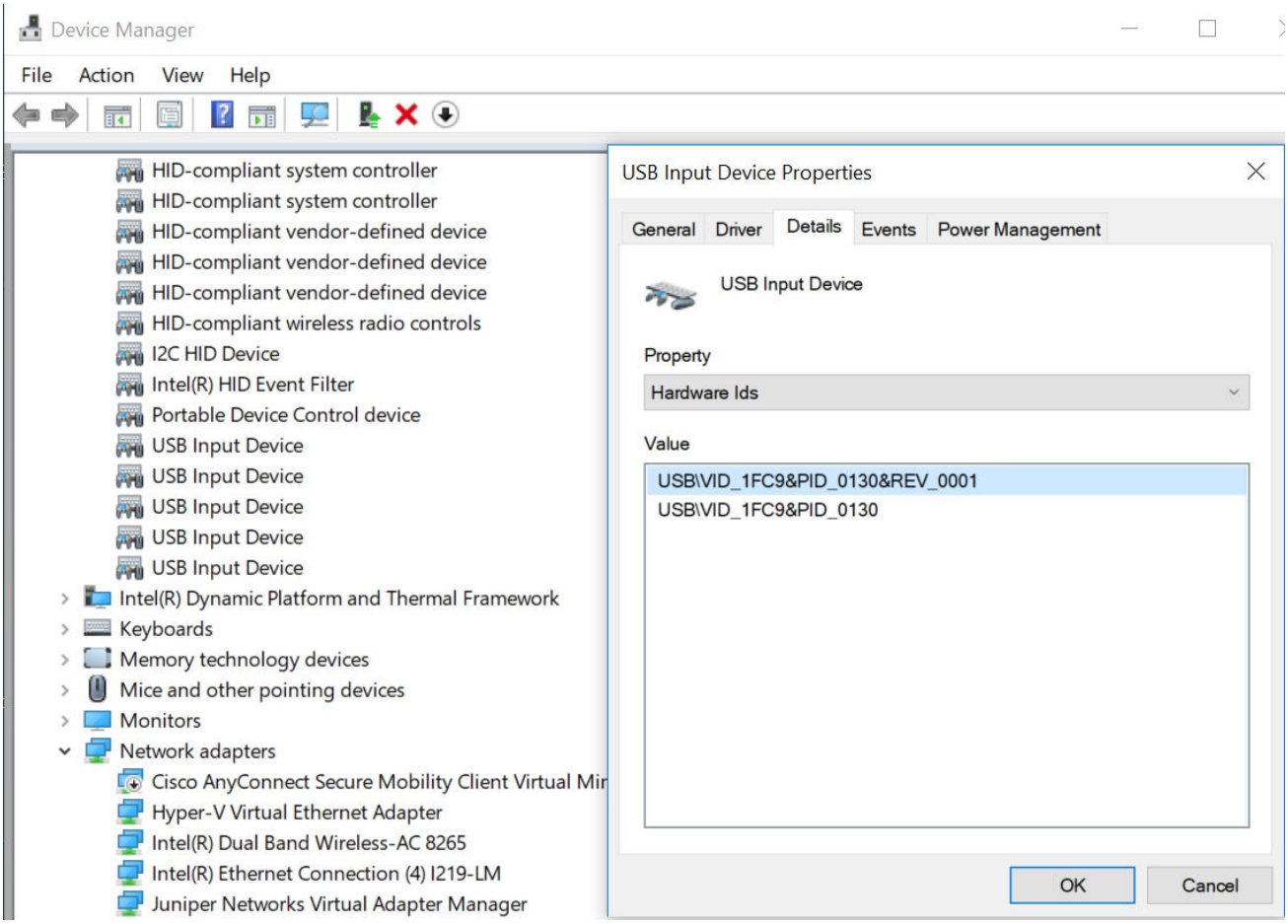

Figure 2. USB connection

On Linux, in order to obtain VID and PID, the following command can be used - $ lsusb

Connection over USB command - $ sudo ./sdphost -u 0x1fc9,0x0130 -- read-register 0x0000

In MAC, in order to obtain VID and PID -

Click Apple logo at top left > About This Mac > System Report... > USB

Connection over USB command - $ sudo ./sdphost -u 0x1fc9,0x0130 -- read-register 0x0000

- **The -j option**:

  This argument is required for a very useful option that outputs the results in JSON and can be used in several different ways to benefit the manufacturing process.

- **The -d option**:

  This argument is required to print extra detail that is useful in debugging. For USB-HID transfers, the option causes sdphost to print out the Report Id of the command, along with the data being sent and received.

The sdphost prints out the response from device with data and the response code. If there is an error or the device failed to connect, sdphost prints out a suitable error message, including the error code returned by the device, if any.

5.1 read-register command

Read-register command is used to read the contents of device memory location or register value. The address of the register or memory location should be passed in as the first argument. Optional arguments include the data format of the register value in number of bits and number of bytes to read.

Examples of sdphost read-register command.

- Example with device connected to COM port 11

```
C:\work\Tools>sdphost.exe -p COM11 -- read-register 0x20000000 32 10
97 36 08 38 a1 c2 f3 e3 bb 3e
Status (HAB mode) = 1450735702 (0x56787856) HAB disabled.
```

- Example with device connected in USB-HID mode
  - — Example with verbose or standard response

```
C:\work\Tools>sdphost.exe -u 0x1FC9,0x0130 -- read-register 0x20000000 32 10
97 36 08 38 a9 c2 f3 e3 bb 3e
Status (HAB mode) = 1450735702 (0x56787856) HAB disabled.
```

  - — Example with debug response

```
C:\work\Tools>sdphost.exe -d -u 0x1FC9,0x0130 -- read-register 0x20000000 32 10
[01 01 01 20 00 00 00 20 00 00 00 0a 00 00 00 00 00]
<03 56 78 78 56 a9 c2 f3 e3 bb 3e 8d 05 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00>
<04 97 36 08 38 a9 c2 f3 e3 bb 3e 8d 05 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00>
97 36 08 38 a9 c2 f3 e3 bb 3e
Status (HAB mode) = 1450735702 (0x56787856) HAB disabled.
```

  - — Example with JSON response

```
C:\work\Tools>sdphost.exe -j -u 0x1FC9,0x0130 -- read-register 0x20000000 32 10
{
   "command" : "read-register",
   "response" : [],
   "status" : {
      "description" : "1450735702 (0x56787856) HAB disabled.",
      "value" : 1450735702
   }
}
```

5.2 Write-register command

The write-register command is used to write a value to the device memory location or register address. The address of the register or memory location should be passed in as the first argument followed by data format of the register value in number of bits and the data to write.

```
C:\work\Tools>sdphost.exe -u 0x1FC9,0x0130 -- write-register 0x20000000 32 0xAABBCCDD
Status (HAB mode) = 1450735702 (0x56787856) HAB disabled.
Reponse Status = 311069202 (0x128a8a12) Write complete.

C:\work\Tools>sdphost.exe -u 0x1FC9,0x0130 -- read-register 0x20000000 32 10
dd cc bb aa a9 c2 f3 e3 bb 3e
Status (HAB mode) = 1450735702 (0x56787856) HAB disabled.
```

5.3 Write-file command

The write-file command is used to write the content of a binary file to the device's memory (internal RAM or SDRAM). The command requires the address of the location where the file contents will be written. The second parameter should specify the name of the binary file to be downloaded on the device. By default, sdphost writes the entire file length of bytes to the target memory. If fewer bytes are required, the optional third argument can be used to specify the size of data.

Typically, write-file is used to program the device with boot image and jump-address is used to start execution of boot image on the device.

```
C:\work\Tools>sdphost.exe -u 0x1FC9,0x0130 -- read-register 0x20000000 32 10
97 36 08 38 a1 c2 73 eb ba 2e
Status (HAB mode) = 1450735702 (0x56787856) HAB disabled.

C:\work\Tools>sdphost.exe -u 0x1FC9,0x0130 -- write-file 0x20000000 sample
Preparing to send 4 (0x4) bytes to the target.
(1/1)100% Completed!
Status (HAB mode) = 1450735702 (0x56787856) HAB disabled.
Reponse Status = 2290649224 (0x88888888) Write File complete.

C:\work\Tools>sdphost.exe -u 0x1FC9,0x0130 -- read-register 0x20000000 32 10
11 22 33 44 00 00 00 00 00 00
Status (HAB mode) = 1450735702 (0x56787856) HAB disabled.
```

5.4 Error-status command

Error-status command is used to read the error code from device for the last command.

Example of error-status command in JSON format

```
C:\work\Tools>sdphost.exe -j -u 0x1FC9,0x0130 -- error-status
{
   "command" : "error-status",
   "response" : [ 4042322160 ],
   "status" : {
      "description" : "1450735702 (0x56787856) HAB disabled.",
      "value" : 1450735702
   }
}
```

5.5 Jump-address command

Jump-address is typically used after successful write-file command where the boot image or any executable image is successfully downloaded on the device's memory for execution. Jump-address command requires the address of the image-vector-table (IVT). IVT can be part of the image or can be downloaded separately. It is a data structure used by ROM that provides information of the boot image entry point and other parameters used for authenticating the image for secure boot. IVT is described in more detail in device's reference manual. Jump-address command will result in execution of the image once rom process the IVT and on a successful authentication of the image.

```
C:\work\Tools>sdphost.exe -u 0x1FC9,0x0130 -- jump-address 0x20000000
Status (HAB mode) = 1450735702 (0x56787856) HAB disabled.
```

Chapter 6
Revision history

The following table contains a history of changes made to this user's guide.

Table 2. Revision history

Revision number	Date	Substantive changes
0	02/2016	Initial Release
1	05/2018	MCU Bootloader v2.5.0 release
2	08/2018	Updates for RT1060 and MCUBoot v2.6.0 release

How To Reach Us

Home Page:

nxp.com

Web Support:

nxp.com/support