(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
4 February 2016 (04.02.2016)

WIPO | PCT

(10) International Publication Number
WO 2016/018753 A1

(51) International Patent Classification:
G06F 1/26 (2006.01)

(21) International Application Number:
PCT/US2015/042006

(22) International Filing Date:
24 July 2015 (24.07.2015)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
14/444,443 28 July 2014 (28.07.2014) US

(71) Applicant: **QUALCOMM INCORPORATED** [US/US]; ATTN: International IP Administration, 5775 Morehouse Drive, San Diego, California 92121-1714 (US).

(72) Inventors: **PATNAIK, Devdutt**; 5775 Morehouse Drive, San Diego, California 92121 (US). **CHOI, Jay, Yu Jae**; 5775 Morehouse Drive, San Diego, California 92121 (US). **REMPLE, Terrence, Brian**; 5775 Morehouse Drive, San Diego, California 92121 (US).

(74) Agent: **DAVENPORT, Taylor, M.**; Withrow & Terranova, PLLC, 2530 Meridian Parkway, Suite 300, Durham, North Carolina 27713 (US).

(81) **Designated States** *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JP, KE, KG, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) **Designated States** *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

Declarations under Rule 4.17:

— *as to applicant's entitlement to apply for and be granted a patent (Rule 4.17(ii))*

— *as to the applicant's entitlement to claim the priority of the earlier application (Rule 4.17(iii))*

Published:

— *with international search report (Art. 21(3))*

(54) **Title**: APPARATUSES, METHODS, AND SYSTEMS FOR ENABLING HIGHER CURRENT CHARGING OF UNIVERSAL SERIAL BUS (USB) SPECIFICATION REVISION 2.0 (USB 2.0) PORTABLE ELECTRONIC DEVICES FROM USB 3.X HOSTS

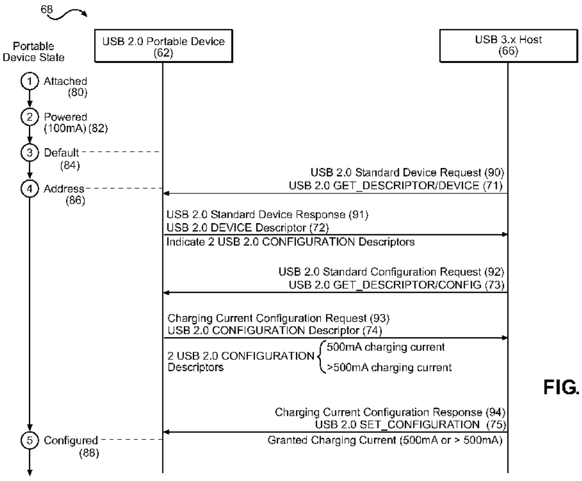
68
Portable Device State
USB 2.0 Portable Device (62)
USB 3.x Host (66)
1 Attached (80)
2 Powered (100mA) (82)
3 Default (84)
4 Address (86)
USB 2.0 Standard Device Request (90)
USB 2.0 GET_DESCRIPTOR/DEVICE (71)
USB 2.0 Standard Device Response (91)
USB 2.0 DEVICE Descriptor (72)
Indicate 2 USB 2.0 CONFIGURATION Descriptors
USB 2.0 Standard Configuration Request (92)
USB 2.0 GET_DESCRIPTOR/CONFIG (73)
Charging Current Configuration Request (93)
USB 2.0 CONFIGURATION Descriptor (74)
2 USB 2.0 CONFIGURATION Descriptors
500mA charging current
>500mA charging current
Charging Current Configuration Response (94)
USB 2.0 SET_CONFIGURATION (75)
Granted Charging Current (500mA or > 500mA)
5 Configured (88)

FIG. 3

(57) **Abstract**: Apparatuses, methods, and systems for enabling higher current charging of Universal Serial Bus (USB) Specification Revision 2.0 (USB 2.0) portable electronic devices from USB 3.x hosts are disclosed. In one aspect, a USB 2.0 controller is provided in a USB 2.0 portable device. A USB 3.x controller is provided in a USB 3.x host. The USB 2.0 controller is configured to draw a higher charging current than specified in USB 2.0 for the USB 2.0 portable device over a USB 2.0 cable. In order to draw the higher charging current without violating USB 2.0, the USB 2.0 controller is configured to use one or more reserved elements in an existing USB 2.0 descriptor(s) or bitmap(s) to indicate a higher charging current request from the USB 2.0 controller.

WO 2016/018753 A1

APPARATUSES, METHODS, AND SYSTEMS FOR ENABLING HIGHER CURRENT CHARGING OF UNIVERSAL SERIAL BUS (USB) SPECIFICATION REVISION 2.0 (USB 2.0) PORTABLE ELECTRONIC DEVICES FROM USB 3.X HOSTS

PRIORITY CLAIM

[0001] The present application claims priority to U.S. Patent Application Serial No. 14/444,443 filed on July 28, 2014 and entitled "APPARATUSES, METHODS, AND SYSTEMS FOR ENABLING HIGHER CURRENT CHARGING OF UNIVERSAL SERIAL BUS (USB) SPECIFICATION REVISION 2.0 (USB 2.0) PORTABLE ELECTRONIC DEVICES FROM USB 3.X HOSTS," the contents of which is incorporated herein by reference in its entirety.

BACKGROUND

I. Field of the Disclosure

[0002] The technology of the disclosure relates generally to charging a battery of portable electronic devices over a Universal Serial Bus (USB) cable connected to a USB host.

II. Background

[0003] Portable electronic devices, such as smartphones, tablets, laptop computers, and the like, can be powered by rechargeable batteries. These rechargeable batteries require periodic recharging. Universal Serial Bus (USB) is an industry standard that defines cables, connectors, and communications protocols for data and power transfers among personal computers and electronic devices. Ports can be used for both data transfer ports as well as charging ports to charge rechargeable batteries of portable electronic devices.

[0004] In this regard, Figure 1 illustrates a physical connectivity between a USB Specification Revision 2.0 (USB 2.0) controller 10 in a USB 2.0 portable electronic device 12 ("portable device 12") and a USB 3.x controller 14 in a USB 3.x host 16. The USB 2.0 controller 10 and the USB 3.x controller 14 are controlled by a USB 2.0 software driver 11 and a USB 3.x software driver 15, respectively. On one end, the USB 2.0 portable device 12 is embedded with a USB 2.0 micro-B receptacle 18. On the other end, the USB 3.x host 16 exposes a USB 3.x enhanced SuperSpeed standard-A receptacle 20. A detachable USB 2.0 cable 22, with a USB 2.0 micro-B plug 24 on one end and a USB 2.0 standard-A plug 26 on the other end, connects the USB 2.0 portable device 12 to the USB 3.x host 16. The USB 2.0 cable has a VBUS conductor, a GND conductor, a D- conductor, and a D+ conductor. The USB 2.0 micro-B receptacle 18 has a VBUS pin 28(1), a GND pin 30(1), a D- pin 32(1), and a D+ pin 34(1). The VBUS pin 28(1) and the GND pin 30(1) are used for charging, while the D- pin 32(1) and the D+ pin 34(1) are used for protocol handshakes and data transfer. The USB 3.x enhanced SuperSpeed standard-A receptacle 20 has a USB 2.0 interface 36 and a USB 3.x SuperSpeed interface 38. The USB 2.0 interface 36 has a VBUS pin 28(2), a GND pin 30(2), a D- pin 32(2), and a D+ pin 34(2). The USB 3.x SuperSpeed interface 38 has additional pins, namely a SSTX- pin 40, a SSTX+ pin 42, a SSRX- pin 46, and a SSRX+ pin 48. The SSTX- pin 40 and the SSTX+ pin 42 are used for SuperSpeed transmissions, while the SSRX- pin 46 and the SSRX+ pin 48 are used for SuperSpeed receptions.

[0005] The USB 2.0 portable device 12 in Figure 1 is allowed to draw up to 500mA charging current from the USB 3.x host 16 as a connected standard downstream port (SDP) according to Battery Charging Specification 1.2 (BC1.2). By means of hardware-based mechanism specified in USB 2.0, the USB 2.0 controller 10 presents itself as a USB 2.0 compatible device by pulling the D+ pin 34(1) HIGH. The USB 3.x controller 14 then detects the USB 2.0 portable device 12 as a USB 2.0 compatible device and selects the operating mode of the USB 2.0 portable device 12 according to USB 2.0. USB 2.0 provides for standard message descriptors to be exchanged between USB devices and hosts for charging purposes. Thus, the USB 3.x controller 14 will allow the USB 2.0 portable device 12 to draw up to 500mA according to BC1.2 even though the USB 2.0 cable 22 is capable of safely carrying more current. However, if the USB 2.0 portable device 12 was instead a USB 3.x compatible device connected to the USB 3.x host 16 with a USB 3.x cable (not shown), the USB 3.x controller 14 would allow the USB 2.0 portable device 12 to draw up to 900mA for faster charging time according to BC1.2.

SUMMARY OF THE DISCLOSURE

[0006] Aspects disclosed in the detailed description include apparatuses, methods, and systems for enabling higher current charging of Universal Serial Bus (USB) Specification Revision 2.0 (USB 2.0) portable electronic devices ("USB 2.0 portable device") from USB 3.x hosts. In aspects disclosed herein, it is desired to allow a USB 2.0 portable device connected to a USB 3.x host by a USB 2.0 cable, to draw more charging current than specified under USB 2.0 for the USB 2.0 portable device. A USB 2.0 cable is capable of safely carrying more current than the specified current limit in USB 2.0. However, it is also desired that USB 2.0 portable device remain compliant with USB 2.0 for compatibility and interoperability reasons. For the USB 2.0 portable device to remain compliant with USB 2.0, it is desired that no USB 2.0 data structure formats used to exchange information between the USB 2.0 portable device and the USB 3.x host be altered to enable a higher charging current for USB 2.0 portable electronic devices. Otherwise, such may result in the USB 2.0 portable device being incompatible with USB 2.0 and unable to pass USB 2.0 compliance tests required for USB 2.0 certification.

[0007] In this regard, in aspects disclosed herein, a USB 2.0 controller with a modified USB 2.0 software driver ("USB 2.0 controller") is provided in the USB 2.0 portable device. A USB 3.x controller with a modified USB 3.x software driver ("USB 3.x controller") is provided in the USB 3.x host. The USB 2.0 controller and the USB 3.x controller are compatible with USB 2.0 and USB 3.x, respectively. In certain aspects, to allow the USB 2.0 portable device to request a higher charging current from the USB 3.x host than specified in USB 2.0 over a USB 2.0 cable, without supporting a new data structure not provided in USB 2.0, the USB 2.0 controller is configured to be capable of requesting a higher charging current from the USB 3.x host in one or more reserved elements in an existing USB 2.0 data structure(s) (e.g., a fixed data structure like a descriptor) in response to a USB 2.0 Standard Device Request received from the USB 3.x controller. The USB 3.x controller is configured to receive and recognize a request for a higher charging current in a reserved element(s) in the USB 2.0 data structure(s) received from the USB 2.0 controller, and in turn provide the higher charging current to the USB 2.0 portable device.

[0008] In this regard in one aspect, a USB 2.0 controller is provided in a USB 2.0 portable device. The USB 2.0 controller is configured to detect a connected USB controller in a USB host over a USB 2.0 cable. The USB 2.0 controller is also configured to send a request in at least one reserved element of at least one USB 2.0 data structure to the USB host over the USB 2.0 cable to draw a higher charging current than a maximum charging current specified in USB 2.0 for a USB 2.0 portable device over the USB 2.0 cable. The USB 2.0 controller is also configured to receive a response from the USB host indicating whether the request to draw the higher charging current is permitted. If the request to draw the higher charging current is permitted, the USB 2.0 controller is configured to draw the higher charging current over the USB 2.0 cable from the host.

[0009] In another aspect, a method of a USB 2.0 portable device drawing a higher charging current from a USB 3.x host is provided. The method comprises detecting a connected USB controller in a USB host over a USB 2.0 cable. The method also comprises sending a request to the USB host over the USB 2.0 cable to draw a higher charging current than a maximum charging current specified in USB 2.0 for a USB 2.0 portable device over the USB 2.0 cable in at least one reserved element of at least one USB 2.0 data structure. The method further comprises receiving a response from the USB host indicating whether the request to draw the higher charging current is permitted and drawing the higher charging current over the USB 2.0 cable from the USB host if the request to draw the higher charging current is permitted.

[0010] In an additional aspect, a system for charging a USB 2.0 portable device from a USB 3.x host is provided. The system comprises a USB 2.0 portable device, a USB 3.x host, and a USB 2.0 cable. The USB 2.0 portable device comprises a USB 2.0 controller and a USB 2.0 receptacle. The USB 2.0 controller is configured to detect a connected USB controller in a host over a USB 2.0 cable. The USB 2.0 controller is also configured to send a request to the connected USB controller over the USB 2.0 cable to draw a higher charging current than a maximum charging current specified in USB 2.0 for the USB 2.0 portable device over the USB 2.0 cable in at least one reserved element of at least one USB 2.0 data structure. The USB 2.0 controller is further configured to receive a response from the connected USB controller indicating whether the request to draw the higher charging current is permitted and draw the higher charging current over the USB 2.0 cable from the USB host if the request to draw the higher charging current is permitted. The USB 2.0 receptacle is configured to engage a first USB 2.0 plug attached to the USB 2.0 cable for connecting to the USB host. The USB 3.x host comprises a USB 3.x controller. The USB 3.x controller is configured to detect the USB 2.0 controller in the USB 2.0 portable device over the USB 2.0 cable. The USB 3.x controller is also configured to receive the request from the USB 2.0 controller over the USB 2.0 cable to draw the higher charging current than the maximum charging current specified in USB 2.0 for the USB 2.0 portable device over the USB 2.0 cable in the at least one reserved element of the at least one USB 2.0 data structure. The USB 3.x controller is further configured to send a response to the USB 2.0 controller indicating whether the request to draw the higher charging current is permitted and provide the higher charging current over the USB 2.0 cable to the USB 2.0 portable device if a grant to draw the higher charging current is accepted. A USB 3.x receptacle is configured to engage a second USB 2.0 plug attached to the USB 2.0 cable for connecting to the USB 2.0 portable device. The USB 2.0 cable comprises the first USB 2.0 plug on one end of the USB 2.0 cable and the second USB 2.0 plug on another end of the USB 2.0 cable.

BRIEF DESCRIPTION OF THE FIGURES

[0011] Figure 1 is a schematic diagram of an exemplary Universal Serial Bus Specification Revision 2.0 (USB 2.0) controller in a USB 2.0 portable device connected to an exemplary USB 3.x controller in a USB 3.x host using a USB 2.0 cable;

[0012] Figure 2 is a schematic diagram of an exemplary USB 2.0 controller in a USB 2.0 portable device configured to request a higher charging current from a USB 3.x controller in one or more reserved elements of an existing USB 2.0 data structure(s);

[0013] Figure 3 is a flowchart illustrating an exemplary signaling flow of USB 2.0 control messages and USB 2.0 descriptors carried as payloads of USB 2.0 control messages of a USB 2.0 portable device involving requesting to draw a higher charging current than the maximum charging current specified in USB 2.0 from a connected USB 3.0 host;

[0014] Figure 4A is a table illustrating an exemplary format of a USB 2.0 DEVICE descriptor that can be used by the USB 2.0 controller in Figure 2 to request to draw a higher charging current from the USB 3.x controller in Figure 2;

[0015] Figure 4B is a table illustrating an exemplary format of a USB 2.0 CONFIGURATION descriptor that has one or more reserved elements that can be used by the USB 2.0 controller in Figure 2 to request to draw a higher charging current from the USB 3.x controller in Figure 2;

[0016] Figure 5 is a block diagram illustrating an exemplary structure and encoding of the USB 2.0 DEVICE descriptor in Figure 5A and the USB 2.0 CONFIGURATION descriptor in Figure 5B that have one or more reserved elements configured to be used by the USB 2.0 controller in Figure 2 to request to draw the higher charging current from the USB 3.x controller in Figure 2;

[0017] Figure 6 is a flowchart illustrating another exemplary signaling flow of USB 2.0 control messages and USB 2.0 data structures carried as payloads of USB 2.0 control messages of a USB 2.0 portable device to request to draw a higher charging current than a maximum charging current specified in USB 2.0 from a connected modified USB 3.x host;

[0018] Figure 7 is a block diagram illustrating an exemplary structure and encoding of a data structure returned by the USB 2.0 controller in Figure 2 in response to a USB 2.0 GET_STATUS Request sent by the USB 3.x controller in Figure 2;

[0019] Figure 8 is a block diagram illustrating an exemplary structure and encoding of a USB 2.0 SET_FEATURE control message that has one of more reserved elements configured to be used by the USB 3.x controller in Figure 2 to grant the higher charging current requested by the USB 2.0 controller in Figure 2; and

[0020] Figure 9 is a block diagram of an exemplary processor-based portable electronic device that can include the USB 2.0 portable device including the USB 2.0 controller in Figure 2.

DETAILED DESCRIPTION

[0021] With reference now to the drawing figures, several exemplary aspects of the present disclosure are described. The word "exemplary" is used herein to mean "serving as an example, instance, or illustration." Any aspect described herein as "exemplary" is not necessarily to be construed as preferred or advantageous over other aspects.

[0022] Aspects disclosed in the detailed description include apparatuses, methods, and systems for enabling higher current charging of Universal Serial Bus Specification Revision 2.0 (USB 2.0) portable electronic devices ("USB 2.0 portable device") from USB 3.x hosts. In aspects disclosed herein, it is desired to allow a USB 2.0 portable device connected to a USB 3.x host by a USB 2.0 cable, to draw more charging current than specified under USB 2.0 for the USB 2.0 portable device. A USB 2.0 cable is capable of safely carrying more current than the specified current limit in USB 2.0. However, it is also desired that USB 2.0 portable device remain compliant with USB 2.0 for compatibility and interoperability reasons. For the USB 2.0 portable device to remain compliant with USB 2.0, it is desired that no USB 2.0 data structure formats used to exchange information between the USB 2.0 portable device and the USB 3.x host be altered to enable a higher charging current for USB 2.0 portable electronic devices. Otherwise, such may result in the USB 2.0 portable device being incompatible with USB 2.0 and unable to pass USB 2.0 compliance tests required for USB 2.0 certification.

[0023] A data structure is a formatted data container so defined to enable data exchange between a sender and a receiver. For example, the data structure may include a descriptor and a bitmap in USB 2.0 and USB 3.x.

[0024] In this regard, Figure 2 illustrates an exemplary USB 2.0 controller 60 with a modified USB 2.0 software driver 61 ("USB 2.0 controller") in a USB 2.0 portable device 62 in this example. The modified USB 2.0 software driver 61 is programmed to control the USB 2.0 controller 60 to operate according to USB 2.0. In addition, the modified USB 2.0 software driver 61 is modified to allow the USB 2.0 portable device 62 to request and draw more charging current than specified under USB 2.0 for the USB 2.0 portable device 62. The USB 2.0 controller 60 is communicatively connected to a USB 3.x controller 64 with a modified USB 3.x software driver 65 in a USB 3.x host 66 through a USB 2.0 cable 22. Similarly, the modified USB 3.x software driver 65 is programmed to control the USB 3.x controller 64 to operate according to USB 3.x. In addition, the modified USB 3.x software driver 65 is modified to allow the USB 3.x host 66 to grant permission for the USB 2.0 portable device 62 to draw more charging current than specified under USB 2.0 for the USB 2.0 portable device 62. When the USB 2.0 portable device 62 is connected to the USB 3.x host 66 by the USB 2.0 cable 22, the USB 2.0 portable device 62 charges its rechargeable battery (not shown) by drawing a charging current from the USB 3.x host 66. USB 2.0 restricts the maximum charging current the USB 2.0 portable device 62 can draw from the USB 3.x host 66 over the USB 2.0 cable 22 (e.g., 500mA). However, USB 3.x allows the USB 3.x host 66 to provide a higher charging current (e.g. 900mA) to a connected USB 3.x portable device (not shown), but can only do so over a USB 3.x cable (not shown). The USB 2.0 cable 22 that connects the USB 2.0 portable device 62 to the USB 3.x host 66 is designed to safely carry a charging current higher than 500 mA. In a non-limiting example, the USB 2.0 cable 22 is able to safely carry up to 1500mA of charging current. Thus, it is desirable to allow the USB 2.0 portable device 62 to draw a higher charging current than specified in the USB 2.0 specification (USB 2.0) (e.g., greater than 500 mA) when connected to the USB 3.x host 66 by the USB 2.0 cable 22, if the USB 2.0 portable device 62 can remain compliant with USB 2.0 by supporting USB 2.0 charging current (e.g., 500mA).

[0025] In this regard, the USB 2.0 controller 60 is provided in the USB 2.0 portable device 62 instead of the USB 2.0 controller 10 in Figure 1. Also, the USB 3.x controller 64 is provided in the USB 3.x host 66 instead of the USB 3.x controller 14 in Figure 1. The USB 2.0 controller 60 is programmed by the modified USB 2.0 software driver 61, which is configured to be in compliance with USB 2.0, but includes added "modified" capabilities to request and draw the higher charging current higher than specified in USB 2.0 (e.g., greater than 500mA) from the USB 3.x controller 64 over the USB 2.0 cable 22. Likewise, the USB 3.x controller 64 is programmed by the modified USB 3.x software driver 65, which is designed to be in compliance with USB 3.x and can provide the higher charging current to the USB 2.0 portable device 62 over the USB 2.0 cable 22. In certain examples discussed in more detail below, the higher charging current can be provided in response to the higher charging current request received from the USB 2.0 controller 60. The USB 2.0 controller 60 is configured to request to draw the charging current higher than 500mA in response to a USB 2.0 Standard Device Request received from the USB 3.x controller 64. In the aspects discussed in more detail below, the modified USB 2.0 software driver 61 in the USB 2.0 controller 60 and the modified USB 3.x software driver 65 in the USB 3.x controller 64 are programmed to enable the higher charging current to be requested by the USB 2.0 controller 60 and to be drawn from the USB 3.x host 66. However, it is also possible to enable the USB 2.0 controller 60 to request and draw the higher charging current from the USB 3.x host 66 by means of hardware components or a combination of hardware components and software drivers in the USB 2.0 controller 60 and the USB 3.x controller 64, respectively.

[0026] To explain exemplary interactions between the USB 2.0 controller 60 and the USB 3.x controller 64 in Figure 2 to enable a higher charging current to be drawn by the USB 2.0 portable device 62 from the USB 3.x host 66, Figure 3 is provided. Figure 3 illustrates an exemplary signaling exchange sequence 68 of USB 2.0 control messages between the USB 2.0 controller 60 of the USB 2.0 portable device 62 and the USB 3.x host 66. The message flow of the exemplary signaling exchange sequence 68 illustrates a series of USB 2.0 control messages for the USB 2.0 controller 60 to request to draw a higher charging current than specified in USB 2.0. Elements of Figures 2 are referenced in connection with Figure 3 and will not be re-described here.

[0027] With reference to Figure 3, the USB 2.0 portable device 62 progresses through an attached state 80, a powered state 82, a default state 84, an address state 86, and a configured state 88 when attached to the USB 3.x host 66 by the USB 2.0 cable 22. The USB 2.0 portable device 62 progresses through these states before the USB 2.0 portable device 62 is ready for power charging and data transfer with the USB 3.x host 66. In the attached state 80, the USB 2.0 controller 60 of the USB 2.0 portable device 62 detects an attachment to the USB 2.0 cable 22. In the powered state 82, the USB 2.0 controller 60 is able to draw a charging current up to 100mA in this example from the USB 3.x host 66 over the USB 2.0 cable 22 in this example, such as specified in USB 2.0. In the default state 84, the USB 2.0 controller 60 waits for a reset from the USB 3.x host 66, which makes the USB 2.0 controller 60 addressable at the default address specified in USB 2.0. In the address state 86, the USB 2.0 controller 60 is assigned a unique address by the USB 3.x host 66 so that data exchange with the USB 3.x host 66 can occur. After receiving the unique address assigned by the USB 3.x host 66, the USB 2.0 controller 60 can request a charging current based on the exemplary signaling exchange sequence 68.

[0028] With reference to Figure 3, after assigning the unique address to the USB 2.0 controller 60 in the address state 86, the USB 3.x controller 64 transmits a USB 2.0 Standard Device Request control message 90 as part of the enumeration procedures defined in USB 2.0. As a non-limiting example, the USB 2.0 Standard Device Request control message 90 is a USB 2.0 GET_DESCRIPTOR/DEVICE control message 71. Upon receipt of the USB 2.0 Standard Device Request control message 90, the USB 2.0 controller 60 in the USB 2.0 portable device 62 sends a USB 2.0 Standard Device Response control message 91, which is a USB 2.0 Device Descriptor control message 72 as an example. The USB 2.0 Device Descriptor control message 72 contains a USB 2.0 DEVICE descriptor as payload. A USB 2.0 DEVICE descriptor format 98 is illustrated in Figure 4A, which is discussed in more detail below. The USB 3.x controller 64 continues with USB 2.0 enumeration by transmitting a USB 2.0 Standard Configuration Request control message 92, which is a USB 2.0 GET_DESCRIPTOR/CONFIG control message 73 as a non-limiting example. In response, the USB 2.0 controller 60 sends a charging current configuration request 93 to the USB 3.x host 66 over the USB 2.0 cable 22. According to the example in Figure 3, the charging current configuration request 93 is a USB 2.0 Configuration Descriptor control message 74, which contains a USB 2.0 CONFIGURATION descriptor as payload. A USB 2.0 CONFIGURATION descriptor format 99 is illustrated in Figure 4B, which is also discussed in more detail below. In the example of Figure 3, to allow the USB 2.0 controller 60 to request a higher charging current than specified in USB 2.0 from the USB 3.x host 66 over the USB 2.0 cable 22, the USB 2.0 controller 60 is configured to send the charging current configuration request 93 containing two (2) or more USB 2.0 CONFIGURATION descriptors to the USB 3.x host 66. The first USB 2.0 CONFIGURATION descriptor includes a charging current level request of 500mA specified in USB 2.0 in this example. The second USB 2.0 CONFIGURATION includes a charging current level request greater than 500mA, for example 900mA, in a reserved element of the USB 2.0 CONFIGURATION descriptor. A reserved element in a USB 2.0 descriptor is a field or value that is not used, but can be used without altering the descriptor format. Thus, in this example of USB 2.0 descriptors, the USB 2.0 controller 60 can be configured to request a higher charging current from the USB 3.x host 66 using a reserved element in a USB 2.0 descriptor, thus remaining compatible with USB 2.0. More exemplary detail on reserved elements in the USB 2.0 CONFIGURATION descriptor that can be used by the USB 2.0 controller 60 to request a higher charging current than specified in USB 2.0 is described in more detail below with regard to Figures 4A, 4B, and 5.

[0029] With reference back to Figure 3, at this point in the signaling exchange sequence 68, the USB 2.0 controller 60 does not have knowledge of whether the USB 3.x host 66 is configured to recognize and process the USB 2.0 CONFIGURATION descriptors encoded with reserved elements to request a higher charging current. In this regard, the USB 3.x controller 64 is configured to send a charging current configuration response 94 to indicate the permitted charging current for the USB 2.0 controller 60 to draw from the USB 3.x host 66. The USB 2.0 controller 60 receives the charging current configuration response 94 from the USB 3.x host 66. The charging current configuration response 94 is a USB 2.0 SET_CONFIGURATION control message 75 in this example. The charging current configuration response 94 contains a specified configuration value that identifies one of the USB 2.0 CONFIGURATION descriptors received from the USB 2.0 controller 60 in the charging current configuration request 93. With reference to the example discussed earlier, the USB 2.0 controller 60 may include two (2) USB 2.0 CONFIGURATION descriptors in the charging current configuration request 91 for the charging current level request of 500mA and a charging current level request of greater than 500mA (e.g., 900mA), respectively. Therefore, the specified configuration value in the charging current configuration response 94 indicates either the charging current level request of 500mA or the charging current level request of greater than 500mA (e.g., 900mA) is permitted. According to USB 2.0, the USB 2.0 controller 60 of the USB 2.0 portable device 62 can start to draw the charging current level identified by the specified configuration value in the charging current configuration response 94 after reaching the configured state 88.

[0030] To help describe specific encodings using reserved elements in existing USB 2.0 descriptors, Figures 4A and 4B illustrate a USB 2.0 DEVICE descriptor format 98 and a USB 2.0 CONFIGURATION descriptor format 99 as defined in USB 2.0, respectively. Elements of Figures 2 are referenced in connection with Figures 4A and 4B and will not be re-described herein. In the USB 2.0 DEVICE descriptor, a bcdUSB field defines a USB specification release number. For example, USB 2.0, USB 3.0, and the like. A bNumConfigurations field indicates the number of CONFIGURATION descriptors that are going to be sent in the USB 2.0 Configuration Descriptor control message 74 in Figure 3. In the USB 2.0 CONFIGURATION descriptor, a bMaxPower field indicates a desired charging current level in a unit of 2mA. For instance, a numeric value fifty (50) in the bMaxPower field indicates a desired charging current of 100mA (50 x 2mA). USB 2.0 defines a maximum value of 250 (0xFA) for the bMaxPower field, which corresponds to 500mA charging current. The numeric value between 251 (0xFB) (inclusive) and 255 (0xFF) (inclusive) are reserved elements therein. This means there can be up to five (5) USB 2.0 CONFIGURATION descriptors associated with different higher charging current levels in the USB 2.0 Configuration Descriptor control message 74.

[0031] As discussed above with regard to Figure 3, the USB 2.0 controller 60 is configured to request a higher charging current in one or more reserved elements in a USB 2.0 CONFIGURATION descriptor. The USB 3.x controller 64 is configured to send the charging current configuration response 94 to indicate the permitted charging current for the USB 2.0 controller 60 to draw from the USB 3.x host 66. Figure 5 illustrates an exemplary encoding method that can be employed by the USB 2.0 controller 60 to request a higher charging current from the USB 3.x host 66 and for the USB 3.x controller 64 to send the charging current configuration response 94 to indicate the permitted charging current.

[0032] In this regard, as illustrated in Figure 5, a USB 2.0 DEVICE descriptor 100, a first USB 2.0 CONFIGURATION descriptor 102, and a second USB 2.0 CONFIGURATION descriptor 104 are illustrated. The USB 2.0 controller 60 can be configured to provide the USB 2.0 DEVICE descriptor 100 having a bcdUSB field configured to indicate USB 2.0 and a bNumConfigurations field configured to indicate two (2) or more USB 2.0 CONFIGURATION descriptors to be used for requesting different higher charging current levels. The USB 2.0 controller 60 can be configured to provide the first USB 2.0 CONFIGURATION descriptor 102 to request the maximum charging current permitted by USB 2.0 (e.g., 500mA) by a numeric value 250 (0xFA) in a bMaxPower field. To allow the USB 2.0 controller 60 to request a higher charging current than specified in USB 2.0, in this example, the USB 2.0 controller 60 is also configured to provide the second USB 2.0 CONFIGURATION descriptor 104.

The second USB 2.0 CONFIGURATION descriptor 104 includes a request for higher charging current higher by setting the bMaxPower field to one of the reserved elements between 251 (0xFB) and 255 (0xFF) without adding any bits to the USB 2.0 CONFIGURATION descriptor 104 format. In this non-limiting example, the numeric value is 255 (0xFF) is designated by the USB 2.0 controller 60 to represent a charging current request of 900mA.

[0033] With continuing reference to Figure 5, the USB 3.x controller 64 is configured to receive both the first USB 2.0 CONFIGURATION descriptor 102 and the second USB 2.0 CONFIGURATION descriptor 104. The USB 3.x controller 64 can review the charging current requests in both the first USB 2.0 CONFIGURATION descriptor 102 and the second USB 2.0 CONFIGURATION descriptor 104. If the USB 3.x host 66 is configured to recognize the higher charging current request from the USB 2.0 controller 60 in a reserved element in the bMaxPower field in a USB 2.0 CONFIGURATION descriptor, the USB 3.x controller 64 will send the charging current configuration response 92 to the USB 2.0 controller 60 and explicitly identify the requested higher charging current to be drawn from the USB 3.x host 66. If the USB 3.x host 66 is not configured to recognize the higher charging current request from the USB 2.0 controller 60 in a reserved element in the bMaxPower field in a USB 2.0 CONFIGURATION descriptor, the USB 3.x host 66 will ignore the higher charging current request in the second USB 2.0 CONFIGURATION descriptor 104. The USB 3.x host 66 will recognize the charging current request by the USB 2.0 controller 60 in the first USB 2.0 CONFIGURATION descriptor 102 and send the charging current configuration response 92 to explicitly allow the lower charging current up to the specified charging current level in USB 2.0. Thus, by configuring the USB 2.0 controller 60 to request a higher charging current than specified in USB 2.0 using a reserved element in the USB 2.0 CONFIGURATION descriptor, the USB 2.0 controller 60 remains compatible with USB 2.0, but can also request a higher charging current in the event that the USB 3.x host 66 is configured with a USB 3.x controller 64 to recognize the higher charging current request in a reserved element.

[0034] It is to be noted that the USB 2.0 controller 60 can be configured to designate the other reserved elements in the bMaxPower field of a USB 2.0 CONFIGURATION descriptor for other higher charging current levels as desired. For example, the numeric value of 254 (0xFE) may be configured to represent a higher charging current of 1500mA. The USB 3.x controller 64 would have also be configured to understand the reserved elements as the same intended higher charging current as configured in the USB 2.0 controller 60.

[0035] Other USB 2.0 data structures beyond the USB 2.0 CONFIGURATION descriptor may also contain reserved elements that can be employed by the USB 2.0 controller 60 of the USB 2.0 portable device 62 to request higher charging current form the USB 3.x host 66 than specified in USB 2.0.

[0036] With reference back to Figure 3, the exemplary signaling exchange sequence 68 for requesting and granting the higher charging current begins when the USB 2.0 controller 60 is assigned the unique address in the address state 86, and concludes when the USB 2.0 controller 60 is configured in the configured state 88. In fact, it is also possible for the USB 2.0 controller 60 to request the higher charging current after it has been configured in the configured state 88. Figure 6 is provided to illustrate an alternative exemplary signaling exchange sequence 70 between the USB 2.0 controller 60 of the USB 2.0 portable device 62 and the USB 3.x host 66 after the USB 2.0 controller 60 is configured in the configured state 88. In this regard, the exemplary signaling exchange sequence 70 in Figure 6 may be conducted in conjunction with or independently from the exemplary signaling exchange sequence 68 in Figure 3. Elements of Figures 2 and 3 are referenced in connection with Figure 6 and will not be re-described here. If conducted independently from the signaling exchange sequence 68, the signaling exchange sequence 70 may be performed any number of times and at any time, thereby allowing the USB 3.x host 66 to dynamically control charging current.

[0037] With reference to Figure 6, after being configured in the configured state 88, the USB 2.0 controller 60 in the USB 2.0 portable device 62 draws a charging current from the USB 3.x host 66. According to the signaling exchange sequence 68 described in Figure 3, the charging current drawn by the USB 2.0 controller 60 may be a standard charging current (e.g., 500mA) or a higher charging current (e.g., 900mA) if the USB 2.0 controller 60 is configured to request for the higher charging current during USB 2.0 enumeration (e.g., the configuration process takes place between the address state 86 and the configured state 88 in Figure 3). According to the signaling exchange sequence 70 in Figure 6, the USB 3.x controller 64 transmits a USB 2.0 Standard Device Request control message 95 to obtain a current status of the USB 2.0 controller 60. As a non-limiting example, the USB 2.0 Standard Device Request control message 95 is a USB 2.0 GET_STATUS Request control message 76. Upon receipt of the USB 2.0 Standard Device Request control message 95, the USB 2.0 controller 60 in the USB 2.0 portable device 62 sends a charging current configuration request 96 to the USB 3.x host 66 over the USB 2.0 cable 22. According to the example in Figure 6, the charging current configuration request 96 is a USB 2.0 GET_STATUS Response control message 77. The USB 2.0 GET_STATUS Response control message 77 contains a USB 2.0 data structure known as the Information Returned by a GetStatus() Request to a Device (hereinafter referred to as "GET_STATUS bitmap" for the convenience of description) as payload. A bitmap is a type of data structure consisting of a plurality of bits. Each of the plurality of bits in the bitmap can have a numeric value of zero (0) or one (1), representing different predefined meanings, respectively. The USB 2.0 GET_STATUS bitmap format and encoding is illustrated in Figure 7 and is discussed in more detail below. In the example of Figure 6, to allow the USB 2.0 controller 60 to request a higher charging current than specified in USB 2.0 from the USB 3.x host 66 over the USB 2.0 cable 22, the USB 2.0 controller 60 is configured to send the charging current configuration request 96 containing the GET_STATUS bitmap, in which at least a reserved bit is encoded to request at least a higher than 500mA charging current, for example 900mA. A reserved bit in the bitmap is a bit that is not used, but can be used by redefinition without altering the bitmap format. Thus, in this example of the GET_STATUS bitmap, the USB 2.0 controller 60 can be configured to request a higher charging current from the USB 3.x host 66 using a reserved bit in the GET_STATUS bitmap, thus remaining compatible with USB 2.0.

[0038] With continuing reference to Figure 6, the USB 3.x controller 64 is configured to send a USB 2.0 Standard Feature Selector descriptor in a charging current configuration response 97 to grant a permission to the USB 2.0 controller 60 to draw a requested charging current higher than 500mA that the USB 3.x host 66 is able and willing to provide. According to the example in Figure 6, the charging current configuration response 97 is a USB 2.0 SET_FEATURE control message 78, which contains the USB 2.0 Standard Feature Selector descriptor in the payload. The USB 2.0 Feature Selector descriptor is another type of data structure. The format and encoding of the USB 2.0 Standard Feature Selector descriptor is illustrated in Figure 8 and described in detail later in this disclosure. The USB 3.x controller 64 can grant permission to allow the USB 2.0 controller 60 to draw the charging current higher than 500mA if the USB 2.0 controller 60 is not connected to the USB 3.x host 66 through a USB hub (not shown). Thus, if the USB 2.0 Standard Feature Selector descriptor has a reserved element, the USB 3.x controller 64 could be configured to provide an indication in such reserved element that a higher charging current is permitted to be drawn by the USB 2.0 controller 60. The USB 2.0 controller 60 could be configured to recognize the reserved element in the received USB 2.0 Standard Feature Selector to become aware that the USB 3.x host 66 will permit a higher charging current to be drawn. In response, the USB 2.0 controller 60 can accept the higher charging current granted by the USB 3.x host 66 if desired, or draws the original charging current as configured in the configured state 88 by rejecting the grant. To accept or reject the higher charging current granted by the USB 3.x host 66, the USB 2.0 controller 60 is configured to transmit a USB 2.0 ZLP control message 79 or a USB 2.0 STALL control message 79(1), respectively.

[0039] Figure 7 illustrates an exemplary USB 2.0 Information returned by a GetStatus() Request to a Device 105 ("GET_STATUS bitmap"). The exemplary GET_STATUS bitmap 105 has a total of sixteen (16) bits labeled as D0 to D15. Each of the sixteen (16) bits can be encoded with a numeric value zero (0) or one (1) to indicate a predefined meaning. Among the sixteen (16) bits in the exemplary GET_STATUS bitmap 105, bits D0 and D1 are already used by USB 2.0 to identify Self-Powered and Remote-Wakeup statuses, respectively. Bits D2 to D15 are currently reserved and any of the reserved bits can be redefined for requesting a higher than 500mA charging current. A plurality of reserved bits in the GET_STATUS bitmap 105 may be encoded to request for a plurality of charging current levels that are higher than 500mA. For example, bit D2 may be used to request for 900mA charging current, bit D3 may be used to request for 1500mA charging, and so on. By being configured to recognize the reserved bits in the GET_STATUS bitmap 105, the USB 3.x controller 64 can grant permission to the USB 2.0 controller 60 to draw a corresponding higher charging current using a reserved element in the USB 2.0 Standard Feature Selector descriptor.

[0040] In this regard, Figure 8 illustrates an exemplary encoding method for the USB 3.x controller 64 to encode a grant in a reserved element in the USB 2.0 SET_FEATURE control message 76 for the USB 2.0 controller 60 to draw the requested higher charging current that the USB 3.x host 66 is able and willing to provide. The USB 2.0 controller 60 and the USB 3.x controller 64 could both be configured to agree upon the meaning in terms of charging current level by a specified reserved value provided in the USB 2.0 SET_FEATURE control message 76.

[0041] As illustrated in Figure 8, the USB 2.0 SET_FEATURE control message 76 in Figure 6 contains a plurality of Standard Feature Selector descriptor 106. The Standard Feature Selector descriptor 106 defined in USB 2.0 include, as a possible feature selector 107, a DEVICE_REMOTE_WAKEUP identified by numeric value one (1) in a value field 108, an END_POINT_HALT feature selector identified by numeric value zero (0) in the value field 108, and a TEST_MODE feature selector identified by numeric value two (2) in the value field 108. Values above two (2) in the value field 108 of the associated feature selector 107 in the Standard Feature Selector descriptor 106 are reserved. Thus, the USB 3.x controller 64 can be configured to include another feature selector 107 representing a higher charging current than is permitted with a reserved value above two (2) in the value field 108. In this example, the USB 3.x controller 64 is configured to provide a HIGH_CURRENT_CHARGING_MODE_X, with X representing a numeric value (e.g., 1, 2, …), as a feature selector 107 in the Standard Feature Selector descriptor 106 with a reserved value of three (3) in the value field 108, as a non-limiting example, if the USB 3.x host 66 will permit the USB 2.0 controller 60 to draw a higher charging current. As a non-limiting example, HIGH_CURRENT_CHARGING_MODE_1 and HIGH_CURRENT_CHARGING_MODE_2 may be configured to represent 900mA and 1500mA charging currents, respectively.

[0042] If the USB 2.0 controller 60 is configured to recognize the permitted higher charging current in the HIGH_CURRENT_CHARGING_MODE feature selector 107 in the Standard Feature Selector descriptor 106 sent by the USB 3.x controller 64, the USB 2.0 controller 60 can choose to draw the permitted higher charging current from the USB 3.x host 66. Alternatively, the USB 2.0 controller 60 can be configured to ignore the indication of the permitted higher charging current included in the feature selector

107 in the Standard Feature Selector descriptor 106 sent by the USB 3.x controller 64. Or, if the USB 2.0 portable device 62 is not configured to recognize a reserved value in a feature selector 107 in the Standard Feature Selector descriptor 106, the USB 2.0 portable device 62 will not be aware of the permitted higher charging current indicated by the USB 3.x controller 64 as being permitted to be drawn from the USB 3.x host 66, which is equipped with the USB 3.x controller 64 to grant the higher charging current in a reserved element.

[0043] The apparatuses, methods, and systems for enabling higher current charging of USB 2.0 portable devices from USB 3.x hosts according to aspects disclosed herein may be applied to enable higher current charging of USB 3.x portable devices from USB 3.x hosts over a USB 2.0 cable or a USB 3.x cable. For USB 3.x portable devices, a similar, but not identical, use of reserved fields will apply.

[0044] The apparatuses, methods, and systems to enable higher current charging of USB 2.0 portable devices from USB 3.x hosts according to aspects disclosed herein may also be provided in or integrated into any processor-based device. Examples, without limitation, include a set top box, an entertainment unit, a navigation device, a communications device, a fixed location data unit, a mobile location data unit, a mobile phone, a cellular phone, a computer, a portable computer, a desktop computer, a personal digital assistant (PDA), a monitor, a computer monitor, a television, a tuner, a radio, a satellite radio, a music player, a digital music player, a portable music player, a digital video player, a video player, a digital video disc (DVD) player, and a portable digital video player.

[0045] In this regard, Figure 9 illustrates an example of a processor-based system 110 that can employ the USB 2.0 controller 60 illustrated in Figure 2. In this example, the processor-based system 110 includes one or more central processing units (CPUs) 112, each including one or more processors 114. The CPU(s) 112 may have cache memory 116 coupled to the processor(s) 114 for rapid access to temporarily stored data. The CPU(s) 112 is coupled to a system bus 118 and can intercouple master and slave devices included in the processor-based system 110. As is well known, the CPU(s) 112 communicates with these other devices by exchanging address, control, and data information over the system bus 118. For example, the CPU(s) 112 can communicate bus transaction requests to a memory controller 120 as an example of a slave device.

Although not illustrated in Figure 8, multiple system buses 118 could be provided, wherein each system bus 118 constitutes a different fabric.

[0046] Other master and slave devices can be connected to the system bus 118. As illustrated in Figure 9, these devices can include a memory system 122, one or more input devices 124, one or more output devices 126, one or more network interface devices 128, and one or more display controllers 130, as examples. The input device(s) 124 can include any type of input device, including but not limited to input keys, switches, voice processors, etc. The output device(s) 126 can include any type of output device, including but not limited to audio, video, other visual indicators, etc. The network interface device(s) 128 can be any devices configured to allow exchange of data to and from a network 132. The network 132 can be any type of network, including but not limited to a wired or wireless network, a private or public network, a local area network (LAN), a wide local area network (WLAN), and the Internet. The network interface device(s) 128 can be configured to support any type of communications protocol desired. The memory system 122 can include one or more memory units 134(0-N).

[0047] A USB 2.0 controller 60(1) is another slave device that can be connected to the system bus 118. The USB 2.0 controller 60(1) can be attached to a USB 2.0 cable 22(1), which has at least a VBUS pin 28(3), a GND pin 30(3), a D- pin 32(3), and a D+ pin 34(3). The CPU(s) 112 may be configured to access the USB 2.0 controller 60(1) over the system bus 118 to set or control the function of the USB 2.0 controller 60(1).

[0048] The CPU(s) 112 may also be configured to access the display controller(s) 130 over the system bus 118 to control information sent to one or more displays 136. The display controller(s) 130 sends information to the display(s) 136 to be displayed via one or more video processors 138, which process the information to be displayed into a format suitable for the display(s) 136. The display(s) 136 can include any type of display, including but not limited to a cathode ray tube (CRT), a liquid crystal display (LCD), a plasma display, etc.

[0049] Those of skill in the art will further appreciate that the various illustrative logical blocks, modules, circuits, and algorithms described in connection with the aspects disclosed herein may be implemented as electronic hardware, instructions stored in memory or in another computer-readable medium and executed by a processor or other processing device, or combinations of both. The master devices, and slave devices described herein may be employed in any circuit, hardware component, integrated circuit (IC), or IC chip, as examples. Memory disclosed herein may be any type and size of memory and may be configured to store any type of information desired. To clearly illustrate this interchangeability, various illustrative components, blocks, modules, circuits, and steps have been described above generally in terms of their functionality. How such functionality is implemented depends upon the particular application, design choices, and/or design constraints imposed on the overall system. Skilled artisans may implement the described functionality in varying ways for each particular application, but such implementation decisions should not be interpreted as causing a departure from the scope of the present disclosure.

[0050] The various illustrative logical blocks, modules, and circuits described in connection with the aspects disclosed herein may be implemented or performed with a processor, a Digital Signal Processor (DSP), an Application Specific Integrated Circuit (ASIC), a Field Programmable Gate Array (FPGA) or other programmable logic device, discrete gate or transistor logic, discrete hardware components, or any combination thereof designed to perform the functions described herein. A processor may be a microprocessor, but in the alternative, the processor may be any conventional processor, controller, microcontroller, or state machine. A processor may also be implemented as a combination of computing devices, e.g., a combination of a DSP and a microprocessor, a plurality of microprocessors, one or more microprocessors in conjunction with a DSP core, or any other such configuration.

[0051] The aspects disclosed herein may be embodied in hardware and in instructions that are stored in hardware, and may reside, for example, in Random Access Memory (RAM), flash memory, Read Only Memory (ROM), Electrically Programmable ROM (EPROM), Electrically Erasable Programmable ROM (EEPROM), registers, a hard disk, a removable disk, a CD-ROM, or any other form of computer readable medium known in the art. An exemplary storage medium is coupled to the processor such that the processor can read information from, and write information to, the storage medium. In the alternative, the storage medium may be integral to the processor. The processor and the storage medium may reside in an ASIC. The ASIC may reside in a remote station. In the alternative, the processor and the storage medium may reside as discrete components in a remote station, base station, or server.

[0052] It is also noted that the operational steps described in any of the exemplary aspects herein are described to provide examples and discussion. The operations described may be performed in numerous different sequences other than the illustrated sequences. Furthermore, operations described in a single operational step may actually be performed in a number of different steps. Additionally, one or more operational steps discussed in the exemplary aspects may be combined. It is to be understood that the operational steps illustrated in the flow chart diagrams may be subject to numerous different modifications as will be readily apparent to one of skill in the art. Those of skill in the art will also understand that information and signals may be represented using any of a variety of different technologies and techniques. For example, data, instructions, commands, information, signals, bits, symbols, and chips that may be referenced throughout the above description may be represented by voltages, currents, electromagnetic waves, magnetic fields or particles, optical fields or particles, or any combination thereof.

[0053] The previous description of the disclosure is provided to enable any person skilled in the art to make or use the disclosure. Various modifications to the disclosure will be readily apparent to those skilled in the art, and the generic principles defined herein may be applied to other variations without departing from the spirit or scope of the disclosure. Thus, the disclosure is not intended to be limited to the examples and designs described herein, but is to be accorded the widest scope consistent with the principles and novel features disclosed herein.

What is claimed is:

1. A Universal Serial Bus Specification Revision 2.0 (USB 2.0) controller in a USB 2.0 portable device configured to:

detect a connected USB controller in a USB host over a USB 2.0 cable;

send a request in at least one reserved element of at least one USB 2.0 data structure to the USB host over the USB 2.0 cable to draw a higher charging current than a maximum charging current specified in USB 2.0 for a USB 2.0 portable device over the USB 2.0 cable;

receive a response from the USB host indicating whether the request to draw the higher charging current is permitted; and

draw the higher charging current over the USB 2.0 cable from the USB host if the request to draw the higher charging current is permitted.

2. The USB 2.0 controller of claim 1 further configured to draw up to the maximum charging current over the USB 2.0 cable from the USB host if the request to draw the higher charging current is not permitted.

3. The USB 2.0 controller of claim 1 further configured to send the request to draw the higher charging current in a USB 2.0 Configuration Descriptor control message in response to receiving a USB 2.0 GET_DESCRIPTOR/CONFIG control message.

4. The USB 2.0 controller of claim 3 configured to send the request to draw the higher charging current by being configured to:

set a bNumConfigurations field in a USB 2.0 DEVICE Descriptor control message to indicate at least two USB 2.0 CONFIGURATION descriptors enclosed in the USB 2.0 Configuration Descriptor control message;

include a first USB 2.0 CONFIGURATION descriptor in the USB 2.0 Configuration Descriptor control message including a first charging current up to the maximum charging current specified in the USB 2.0 for the USB 2.0 portable device over the USB 2.0 cable; and

include at least one second USB 2.0 CONFIGURATION descriptor in the USB 2.0 Configuration Descriptor control message including at least one second charging current higher than the maximum charging current specified in the USB 2.0 for the USB 2.0 portable device over the USB 2.0 cable.

5. The USB 2.0 controller of claim 4 configured to:

include the first USB 2.0 CONFIGURATION descriptor in the USB 2.0 Configuration Descriptor control message by being further configured to include the first charging current in a bMaxPower field in the first USB 2.0 CONFIGURATION descriptor; and

include the at least one second USB 2.0 CONFIGURATION descriptor in the USB 2.0 Configuration Descriptor control message by being further configured to include the at least one second charging current in the bMaxPower field in the at least one second USB 2.0 CONFIGURATION descriptor.

6. The USB 2.0 controller of claim 5 further configured to include the first charging current indicative of 500 milliamps (mA) in the bMaxPower field in the first USB 2.0 CONFIGURATION descriptor.

7. The USB 2.0 controller of claim 5, wherein the at least one second charging current is comprised of at least one charging current above 500 milliamps (mA) and equal to or less than 1500 mA.

8. The USB 2.0 controller of claim 5 configured to include the at least one second charging current in the at least one bMaxPower field in the at least one second USB 2.0 CONFIGURATION descriptor by being further configured to set the bMaxPower field to at least one reserved value between 251 (0xFB) and 255 (0xFF) indicative of at least one predefined charging current above the maximum charging current.

9. The USB 2.0 controller of claim 8 further configured to set the at least one bMaxPower field to the at least one reserved value between 251 (0xFB) and 255 (0xFF) indicative of a predefined charging current comprised of at least one charging current above 500 milliamps (mA) and equal to or less than 1500 mA.

10. The USB 2.0 controller of claim 1 further configured to receive the response from the USB host by being configured to receive a USB 2.0 SET_CONFIGURATION control message from the USB host indicating whether the request to draw the higher charging current is permitted in a reserved value.

11. The USB 2.0 controller of claim 1 further configured to send the request to draw the higher charging current in a USB 2.0 GET_STATUS Response control message in response to receiving a USB 2.0 GET_STATUS Request control message.

12. The USB 2.0 controller of claim 11 further configured to send the request to draw the higher charging current by being configured to set at least one reserved bit in a USB 2.0 information returned by a GetStatus() request to a device data structure in the USB 2.0 GET_STATUS Response control message.

13. The USB 2.0 controller of claim 12, wherein the higher charging current is comprised of at least one charging current above 500 milliamps (mA) and equal to or less than 1500 mA.

14. The USB 2.0 controller of claim 12 further configured to set the at least one reserved bit between D2 and D15 in the USB 2.0 information returned by the GetStatus() request to the device data structure indicative of a predefined charging current comprised of at least one charging current above 500 milliamps (mA) and equal to or less than 1500 mA.

15. The USB 2.0 controller of claim 11 further configured to receive a grant in the at least one reserved element of the at least one USB 2.0 data structure in a USB 2.0 SET_FEATURE control message.

16. The USB 2.0 controller of claim 15 further configured to receive the grant in the at least one reserved element comprising a HIGH_CURRENT_CHARGING_MODE feature selector.

17. The USB 2.0 controller of claim 11 further configured to send a USB 2.0 control message to the USB host indicating whether a grant to draw the higher charging current is accepted.

18. The USB 2.0 controller of claim 17 configured to send a USB 2.0 ZLP control message to the USB host if the grant to draw the higher charging current is accepted.

19. The USB 2.0 controller of claim 17 configured to send a USB 2.0 STALL control message to the USB host if the grant to draw the higher charging current is not accepted.

20. The USB 2.0 controller of claim 1 integrated into a device selected from the group consisting of a set top box, an entertainment unit, a navigation device, a communications device, a fixed location data unit, a mobile location data unit, a mobile phone, a cellular phone, a computer, a portable computer, a desktop computer, a personal digital assistant (PDA), a monitor, a computer monitor, a television, a tuner, a radio, a satellite radio, a music player, a digital music player, a portable music player, a digital video player, a video player, a digital video disc (DVD) player, and a portable digital video player.

21. A method of a Universal Serial Bus Specification Revision 2.0 (USB 2.0) portable device drawing a higher charging current from a USB host, comprising:

detecting a connected USB controller in a USB host over a USB 2.0 cable;

sending a request to the USB host over the USB 2.0 cable to draw a higher charging current than a maximum charging current specified in USB 2.0 for a USB 2.0 portable device over the USB 2.0 cable in at least one reserved element of at least one USB 2.0 data structure;

receiving a response from the USB host indicating whether the request to draw the higher charging current is permitted; and

drawing the higher charging current over the USB 2.0 cable from the USB host if the request to draw the higher charging current is permitted.

22. The method of claim 21, further comprising drawing up to the maximum charging current over the USB 2.0 cable from the USB host if the request to draw the higher charging current is not permitted.

23. The method of claim 21 further comprising sending the request to draw the higher charging current in a USB 2.0 Configuration Descriptor control message.

24. The method of claim 23 wherein sending the request to draw the higher charging current further comprises:

setting a bNumConfigurations field in a USB 2.0 DEVICE descriptor in the USB 2.0 Configuration Descriptor control message to indicate at least two USB 2.0 CONFIGURATION descriptors enclosed in a USB 2.0 SET_CONFIGURATION control message;

including a first USB 2.0 CONFIGURATION descriptor in the USB 2.0 Configuration Descriptor control message including a first charging current up to the maximum charging current specified in the USB 2.0 for the USB 2.0 portable device over the USB 2.0 cable; and

including at least one second USB 2.0 CONFIGURATION descriptor in the USB 2.0 Configuration Descriptor control message including at least one second charging current higher than the maximum charging current specified in the USB 2.0 for the USB 2.0 portable device over the USB 2.0 cable.

25. The method of claim 24 wherein:

including the first USB 2.0 CONFIGURATION descriptor in the USB 2.0 Configuration Descriptor control message comprises including the first charging current in a bMaxPower field in the first USB 2.0 CONFIGURATION descriptor; and

including the at least one second USB 2.0 CONFIGURATION descriptor in the USB 2.0 Configuration Descriptor control message comprises including the at least one second charging current in at least one bMaxPower field in the at least one second USB 2.0 CONFIGURATION descriptor.

26. The method of claim 21, wherein receiving the response from the USB host further comprises receiving a USB 2.0 SET_CONFIGURATION control message from the USB host indicating whether the request to draw the higher charging current is permitted in a reserved value.

27. A system for charging a Universal Serial Bus Specification Revision 2.0 (USB 2.0) portable device from a USB host, comprising:

a USB 2.0 portable device, comprising:

a USB 2.0 controller configured to:

detect a connected USB controller in the USB host over a USB 2.0 cable;

send a request to the connected USB controller over the USB 2.0 cable to draw a higher charging current than a maximum charging current specified in USB 2.0 for the USB 2.0 portable device over the USB 2.0 cable in at least one reserved element of at least one USB 2.0 data structure;

receive a response from the connected USB controller indicating whether the request to draw the higher charging current is permitted; and

draw the higher charging current over the USB 2.0 cable from the USB host if the request to draw the higher charging current is permitted; and

a USB 2.0 receptacle configured to engage a first USB 2.0 plug attached to the USB 2.0 cable for connecting to the USB host; and

a USB 3.x host, comprising:

a USB 3.x controller configured to:

detect the USB 2.0 controller in the USB 2.0 portable device over the USB 2.0 cable;

receive the request from the USB 2.0 controller over the USB 2.0 cable to draw the higher charging current than the maximum charging current specified in the USB 2.0 for the USB 2.0 portable device over the USB 2.0 cable in the at least one reserved element of the at least one USB 2.0 data structure;

send a response to the USB 2.0 controller indicating whether the request to draw the higher charging current is permitted; and

provide the higher charging current over the USB 2.0 cable to the USB 2.0 portable device if a grant to draw the higher charging current is accepted; and

a USB 3.x receptacle configured to engage a second USB 2.0 plug attached to the USB 2.0 cable for connecting to the USB 2.0 portable device; and

the USB 2.0 cable comprising the first USB 2.0 plug on one end of the USB 2.0 cable and the second USB 2.0 plug on another end of the USB 2.0 cable.

28. The system of claim 27, wherein the USB 3.x controller is further configured to recognize a bMaxPower field in a USB 2.0 CONFIGURATION descriptor including the at least one reserved element between 251 (0xFB) and 255 (0xFF) indicating the requested higher charging current.

29. The system of claim 27, wherein the USB 3.x controller is further configured to recognize a reserved bit between D2 and D15 in information returned in response to a USB 2.0 GetStatus() Request indicating the requested higher charging current.

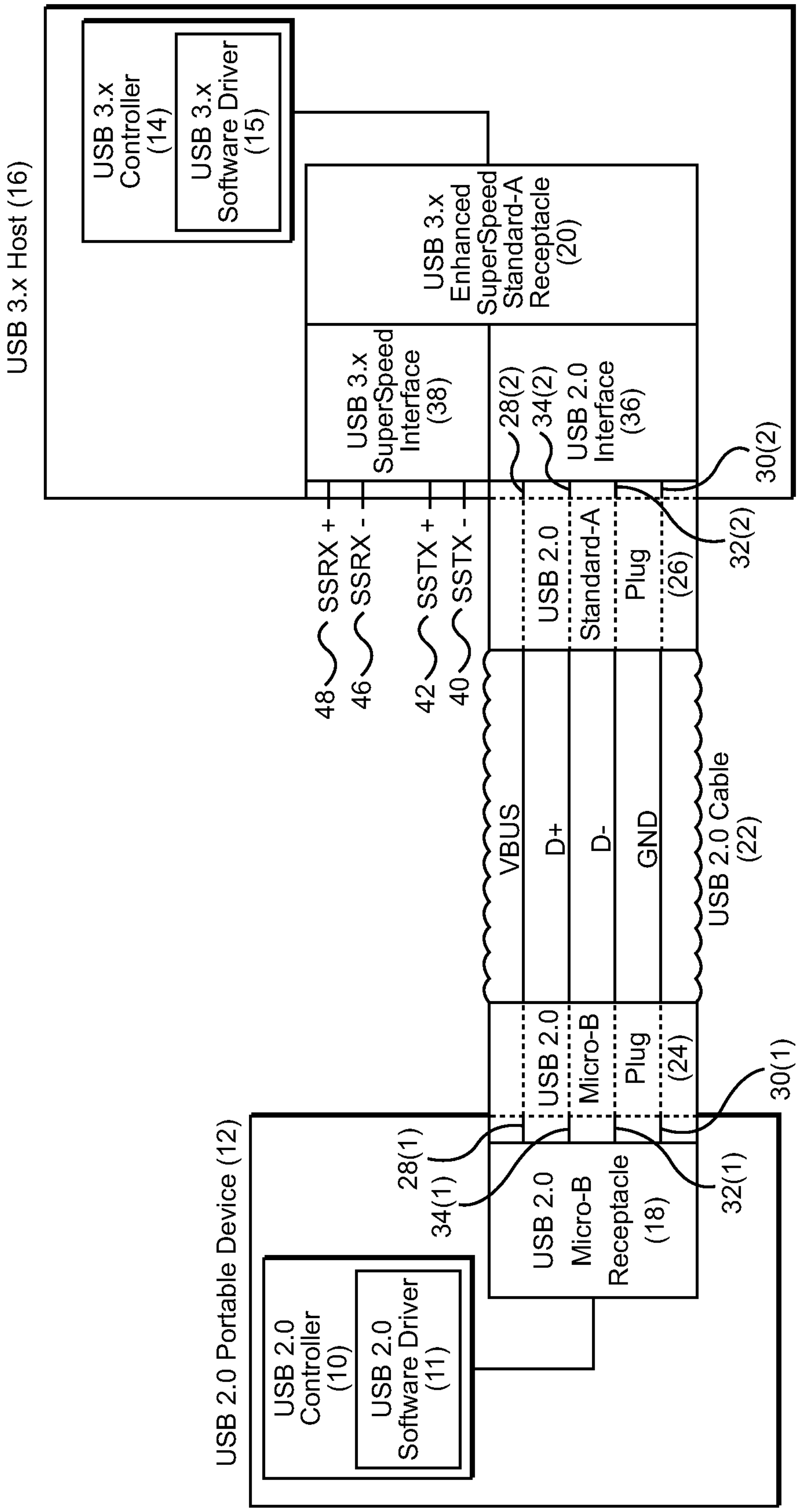
USB 3.x Host (16)
USB 3.x Controller (14)
USB 3.x Software Driver (15)
USB 3.x Enhanced SuperSpeed Standard-A Receptacle (20)
USB 3.x SuperSpeed Interface (38)
USB 2.0 Interface (36)
28(2)
34(2)
30(2)
32(2)
48
SSRX +
46
SSRX -
42
SSTX +
40
SSTX -
USB 2.0 Standard-A Plug (26)
VBUS
D+
D-
GND
USB 2.0 Cable (22)
USB 2.0 Micro-B Plug (24)
30(1)
28(1)
34(1)
32(1)
USB 2.0 Portable Device (12)
USB 2.0 Micro-B Receptacle (18)
USB 2.0 Controller (10)
USB 2.0 Software Driver (11)

FIG. 1
PRIOR ART

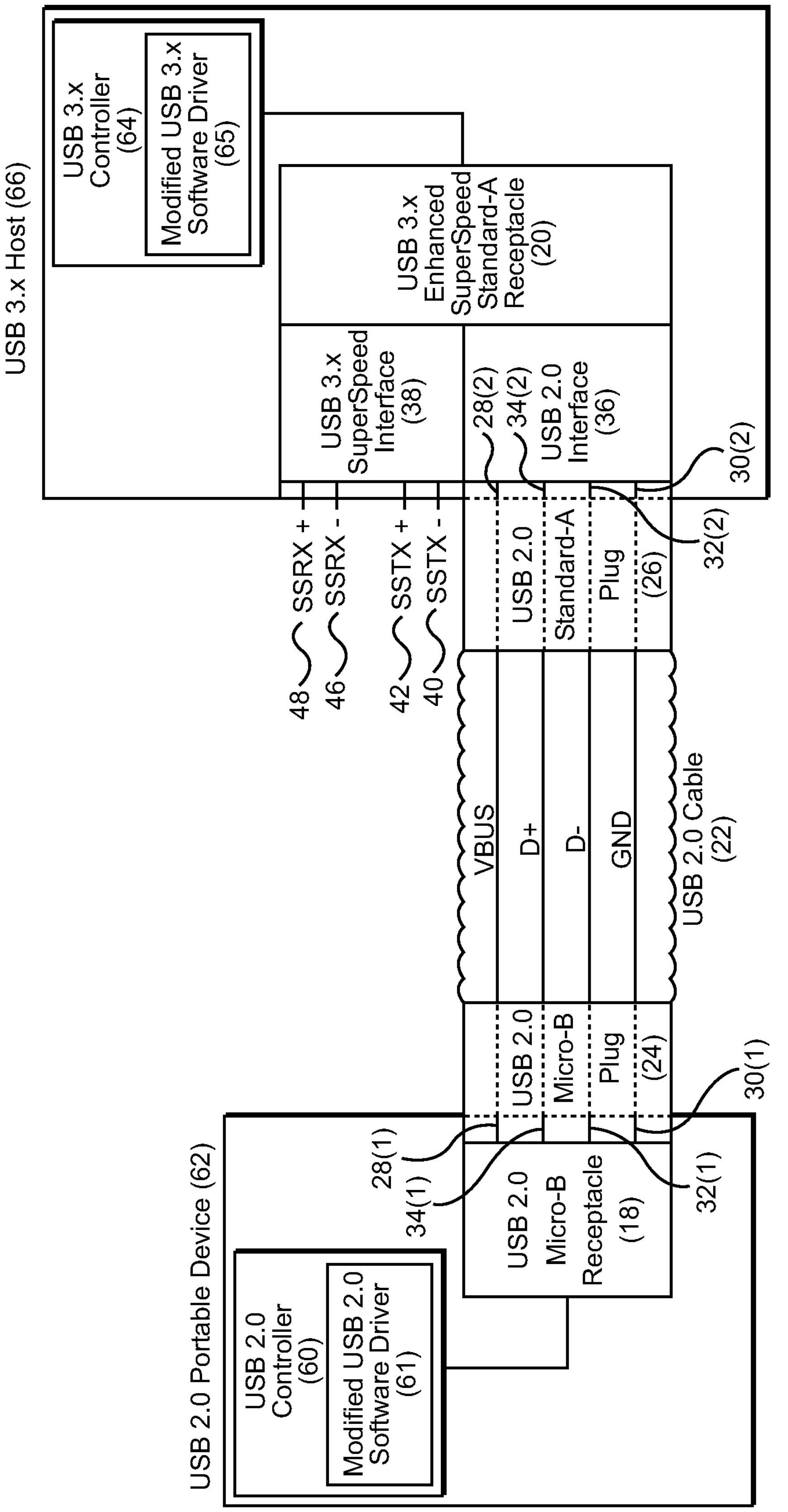
USB 3.x Host (66)
USB 3.x Controller (64)
Modified USB 3.x Software Driver (65)
USB 3.x Enhanced SuperSpeed Standard-A Receptacle (20)
USB 3.x SuperSpeed Interface (38)
USB 2.0 Interface (36)
28(2)
34(2)
30(2)
32(2)
48 SSRX +
46 SSRX -
42 SSTX +
40 SSTX -
USB 2.0 Standard-A Plug (26)
VBUS
D+
D-
GND
USB 2.0 Cable (22)
USB 2.0 Micro-B Plug (24)
30(1)
28(1)
34(1)
32(1)
USB 2.0 Micro-B Receptacle (18)
USB 2.0 Portable Device (62)
USB 2.0 Controller (60)
Modified USB 2.0 Software Driver (61)

FIG. 2

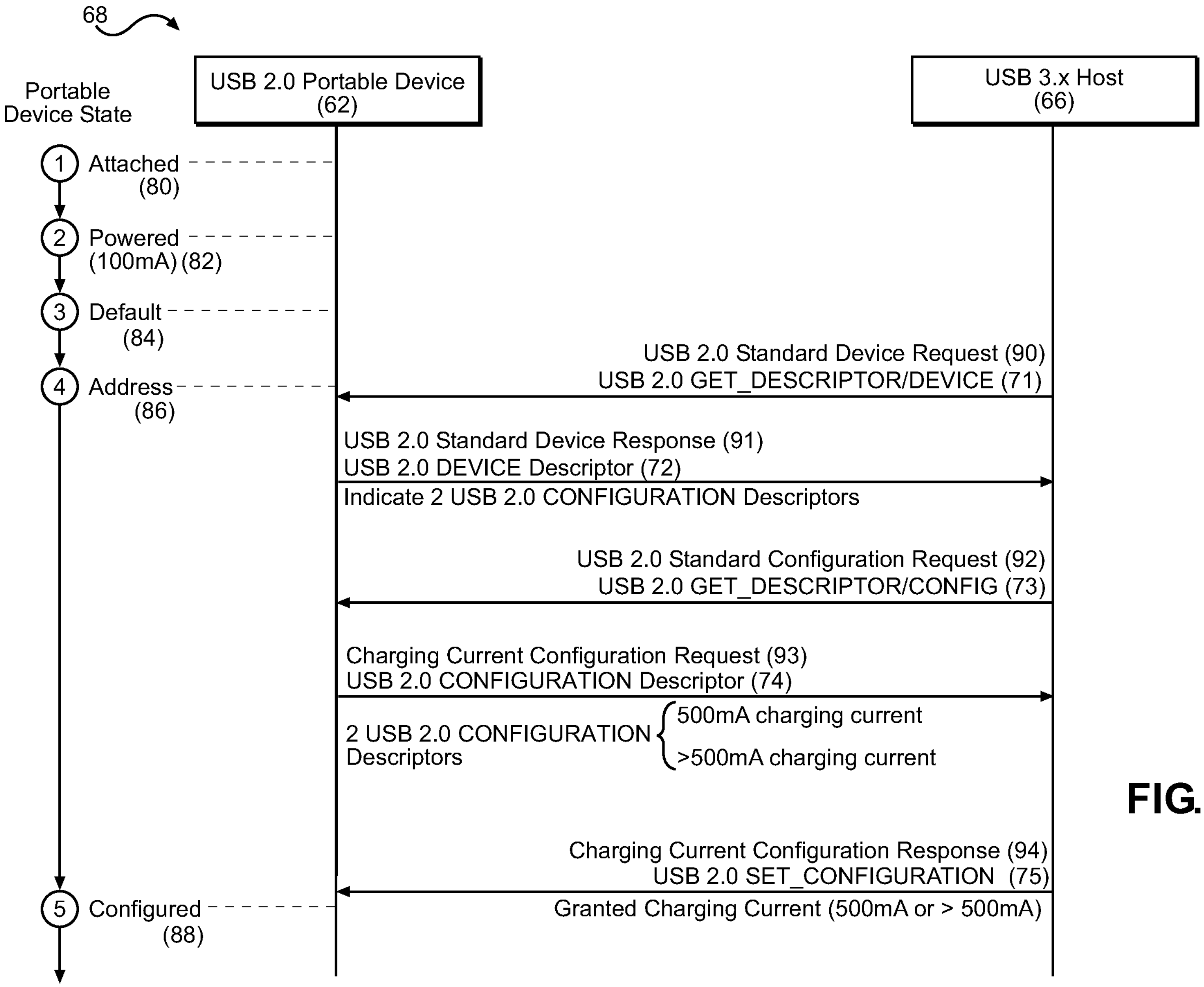
68
Portable
Device State
USB 2.0 Portable Device
(62)
USB 3.x Host
(66)
1 Attached
(80)
2 Powered
(100mA) (82)
3 Default
(84)
4 Address
(86)
5 Configured
(88)
USB 2.0 Standard Device Request (90)
USB 2.0 GET_DESCRIPTOR/DEVICE (71)
USB 2.0 Standard Device Response (91)
USB 2.0 DEVICE Descriptor (72)
Indicate 2 USB 2.0 CONFIGURATION Descriptors
USB 2.0 Standard Configuration Request (92)
USB 2.0 GET_DESCRIPTOR/CONFIG (73)
Charging Current Configuration Request (93)
USB 2.0 CONFIGURATION Descriptor (74)
2 USB 2.0 CONFIGURATION
Descriptors
500mA charging current
>500mA charging current
Charging Current Configuration Response (94)
USB 2.0 SET_CONFIGURATION (75)
Granted Charging Current (500mA or > 500mA)

FIG. 3

USB 2.0 DEVICE Descriptor Format (98)

Offset	Field	Size	Value	Discription
0	bLength	1	Number	Size of this descriptor in bytes
1	bDescriptorType	1	Constant	DEVICE descriptor type (= 1)
2	**bcdUSB**	2	BCD	USB Spec release number
4	bDeviceClass	1	Class	00h means each interface defines its own class FFh means vendor-defined class Any other value must be a class code
5	bDeviceSubClass	1	SubClass	SubClass Code assigned by USB-IF
6	bDeviceProtocol	1	Protocol	Protocol Code assigned by USB-IF
7	bMaxPacketSize0	1	Number	Max packet size for endpoint 0. Must be 8, 16, 32 or 64
8	idVendor	2	ID	Vendor ID - must be obtained from USB-IF
10	idProduct	2	ID	Product ID - assigned by the manufacturer
12	bcdDevice	2	BDC	Device release number in binary coded decimal
14	iManufacturer	1	Index	Index of string descriptor describing manufacturer - set to 0 if no string
15	iProduct	1	Index	Index of string descriptor describing product - set to 0 if no string
16	iSerialNumber	1	Index	Index of string descriptor describing device serial number - set to 0 if no string
17	**bNumConfigurations**	1	Number	Number of possible configurations

FIG. 4A

USB 2.0 CONFIGURATION Descriptor Format (99)

Offset	Field	Size	Value	Discription
0	bLength	1	Number	Size of this descriptor in bytes
1	bDescriptorType	1	Constant	CONFIGURATION descriptor type (= 2)
2	wTotalLength	2	Number	Total number of bytes in this descriptor and all the following descriptors.
3	bNumInterfaces	1	Number	Number of interfaces supported by this configuration
4	bConfigurationValue	1	Number	Value used by Set Configuration to select this configuration
5	iConfiguration	1	Index	Index of string descriptor describing configuration - set to 0 if no string
6	bmAttributes	1	Bitmap	D7: Must be set to 1 D6: Self-powered D5: Remote Wakeup D4...D0: Set to 0
7	**bMaxPower**	1	mA	Maximum current drawn by device in this configuration.

FIG. 4B

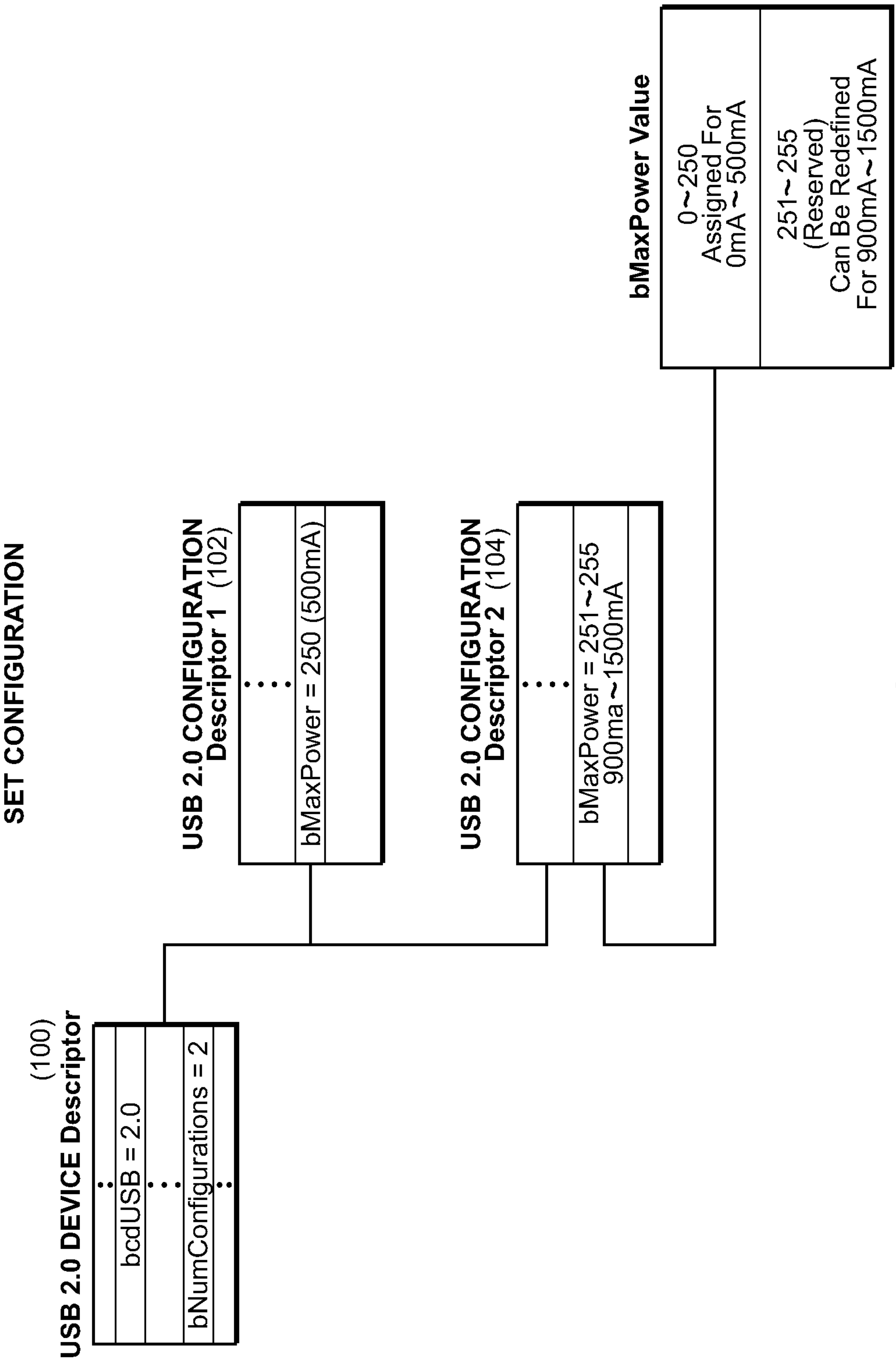
SET CONFIGURATION
USB 2.0 DEVICE Descriptor
(100)
bcdUSB = 2.0
bNumConfigurations = 2
USB 2.0 CONFIGURATION
Descriptor 1 (102)
bMaxPower = 250 (500mA)
USB 2.0 CONFIGURATION
Descriptor 2 (104)
bMaxPower = 251~255
900ma~1500mA
bMaxPower Value
0~250
Assigned For
0mA~500mA
251~255
(Reserved)
Can Be Redefined
For 900mA~1500mA

FIG. 5

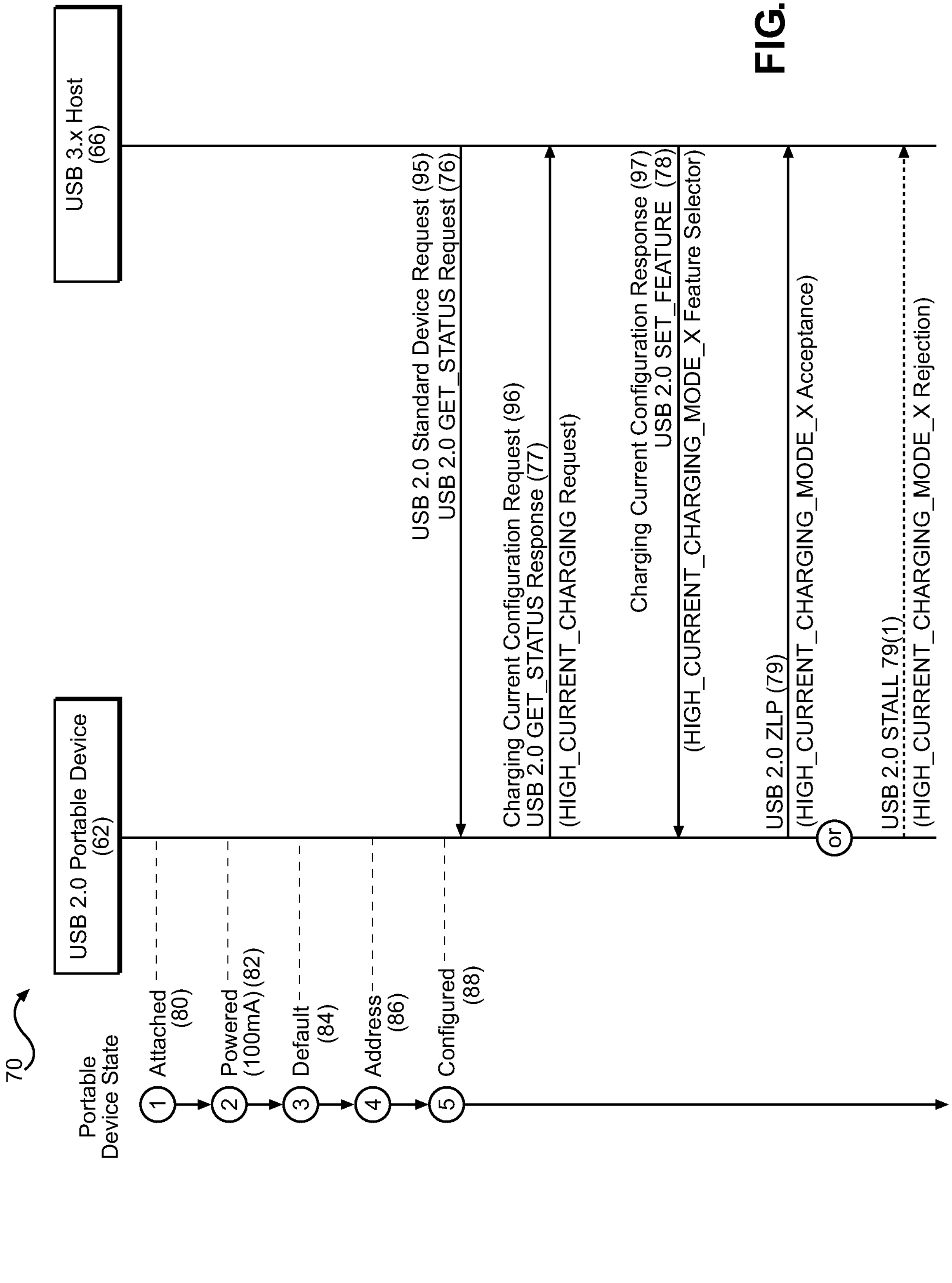
70
Portable Device State
1 Attached (80)
2 Powered (100mA) (82)
3 Default (84)
4 Address (86)
5 Configured (88)
USB 2.0 Portable Device (62)
USB 3.x Host (66)
USB 2.0 Standard Device Request (95)
USB 2.0 GET_STATUS Request (76)
Charging Current Configuration Request (96)
USB 2.0 GET_STATUS Response (77)
(HIGH_CURRENT_CHARGING Request)
Charging Current Configuration Response (97)
USB 2.0 SET_FEATURE (78)
(HIGH_CURRENT_CHARGING_MODE_X Feature Selector)
USB 2.0 ZLP (79)
(HIGH_CURRENT_CHARGING_MODE_X Acceptance)
or
USB 2.0 STALL 79(1)
(HIGH_CURRENT_CHARGING_MODE_X Rejection)

FIG. 6

Information Returned by a GetStatus() Request to a Device (105)

D7	D6	D5	D4	D3	D2	D1	D0
Reserved (Reset to Zero)				Request for 1500 mA Charge Current	Request for 900 mA Charge Current	Remote Wakeup	Self Powered
D15	D14	D13	D12	D11	D10	D9	D8
Reserved (Reset to Zero)							

FIG. 7

Standard Feature Selector Descriptor (106)

Feature Selector (107)	**Recipient**	**Value** (108)
DEVICE_REMOTE_WAKEUP	Device	1
END_POINT_HALT	Endpoint	0
TEST_MODE	Device	2
HIGH_CURRENT_CHARGING_MODE_X	Device	≥ 3

(X = 0, 1, 2...)

FIG. 8

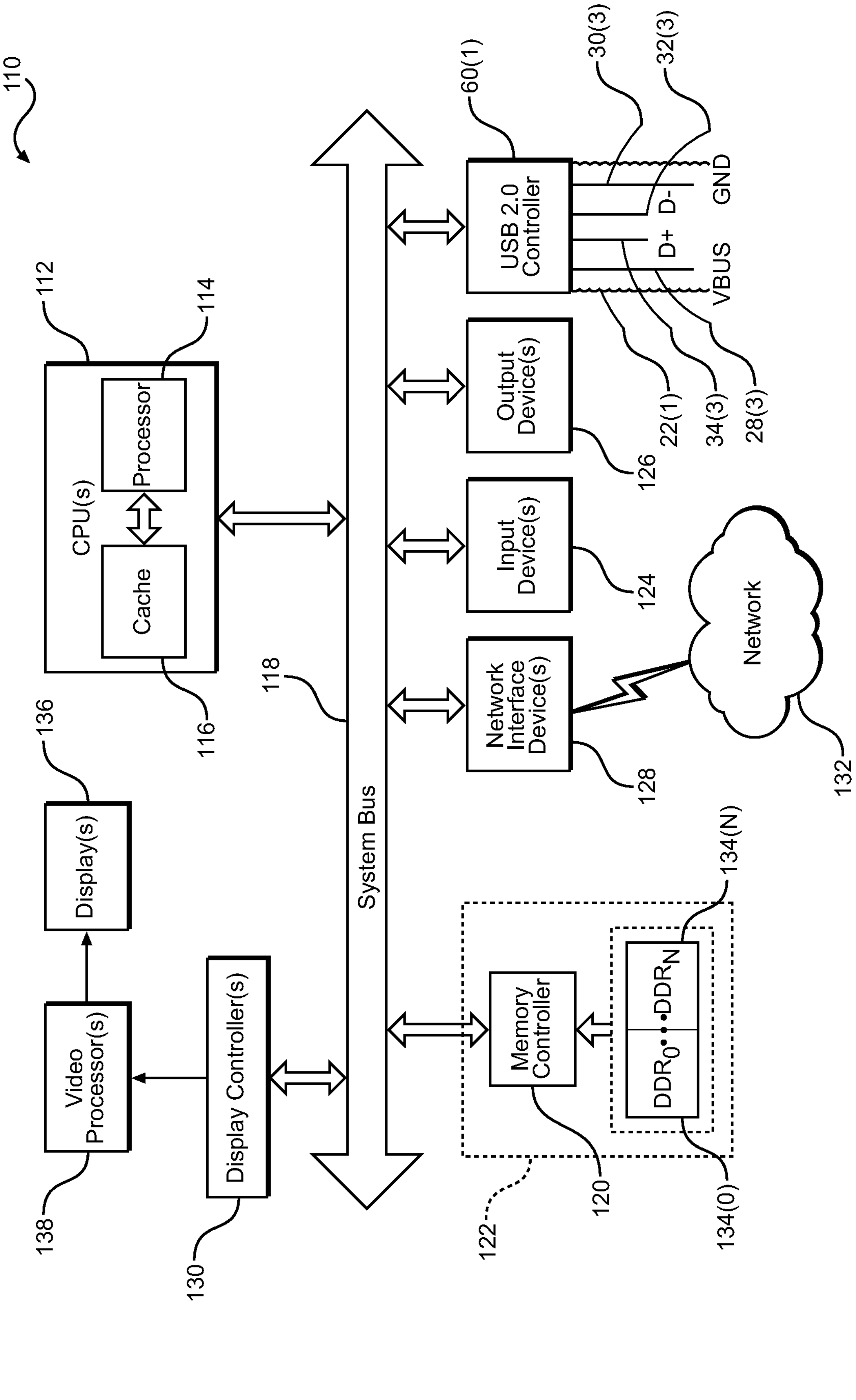
110
CPU(s)
112
Processor
114
Cache
116
Display(s)
136
Video Processor(s)
138
Display Controller(s)
130
System Bus
118
USB 2.0 Controller
60(1)
30(3)
32(3)
D+
D-
VBUS
GND
22(1)
34(3)
28(3)
Output Device(s)
126
Input Device(s)
124
Network Interface Device(s)
128
Network
132
Memory Controller
120
122
DDR$_0$ • • • DDR$_N$
134(0)
134(N)

FIG. 9

INTERNATIONAL SEARCH REPORT

International application No
PCT/US2015/042006

A. CLASSIFICATION OF SUBJECT MATTER
INV. G06F1/26
ADD.

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

EPO-Internal, WPI Data

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 2013/151731 A1 (LAI YI-LIN [TW] ET AL) 13 June 2013 (2013-06-13) the whole document -----	1-29
X	US 2011/271122 A1 (KING BENJAMIN JOHN [US]) 3 November 2011 (2011-11-03) the whole document -----	1-29
A	WO 01/57679 A2 (MICROSOFT CORP [US]) 9 August 2001 (2001-08-09) the whole document -----	1-29
	-/--	

☒ Further documents are listed in the continuation of Box C.

☒ See patent family annex.

* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search 9 September 2015	Date of mailing of the international search report 17/09/2015
Name and mailing address of the ISA/ European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Fax: (+31-70) 340-3016	Authorized officer Ghidini, Mario

1

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	USB IMPLEMENTERS FORUM: "Universal Serial Bus Specification Revision 2.0 - Chapters 1 - 5 and 9", INTERNET CITATION, 27 April 2000 (2000-04-27), XP002412161, Retrieved from the Internet: URL:http://www.usb.org/developers/docs/usb_20_05122006.zip [retrieved on 2006-12-19] the whole document -----	1-29

1

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No

PCT/US2015/042006

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2013151731 A1	13-06-2013	CN 103019355 A	03-04-2013
		EP 2602721 A2	12-06-2013
		TW 201325025 A	16-06-2013
		US 2013151731 A1	13-06-2013
US 2011271122 A1	03-11-2011	NONE	
WO 0157679 A2	09-08-2001	AU 3648001 A	14-08-2001
		US 6484219 B1	19-11-2002
		WO 0157679 A2	09-08-2001