AN2341

USB4715 FlexConnect Operation

Authors: *Andrew Rogers and Connor Chilton*
Microchip Technology Inc.

INTRODUCTION

The Microchip USB4715 smart hub allows any of the four FlexConnect-capable USB ports to assume the role of USB host at any time during hub operation. This host role exchange feature is called FlexConnect.

This functionality can be used in two primary ways:

- **Host Swapping:** This functionality can be achieved through a hub, where a host and a device can agree to swap the host or device relationship. The host becomes a device, and the device becomes a host.
- **Host Sharing:** A USB ecosystem can be shared between multiple hosts. Note that only one host may have access to the USB tree at a time.

FlexConnect can be enabled through any of the following methods:

- **Register Access:** An embedded SMBus master can control the state of the FlexConnect feature through basic write/read operations. This same register can be written to by a host during runtime to initiate FlexConnect.
- **USB Command:** FlexConnect can be initiated via a special USB command directed to the hub's internal Hub Feature Controller (HFC) device.
- **Direct Pin Control:** A select number of available GPIO pins on the hub in configuration 2 and 4 can be assigned the role of a FlexConnect control pin.

Sections

References

Consult the following documents for details on the specific parts referred to in this document:

- *USB4715 Data Sheet*
- *AN2651 Configuration of USB47xx and USB49xx*

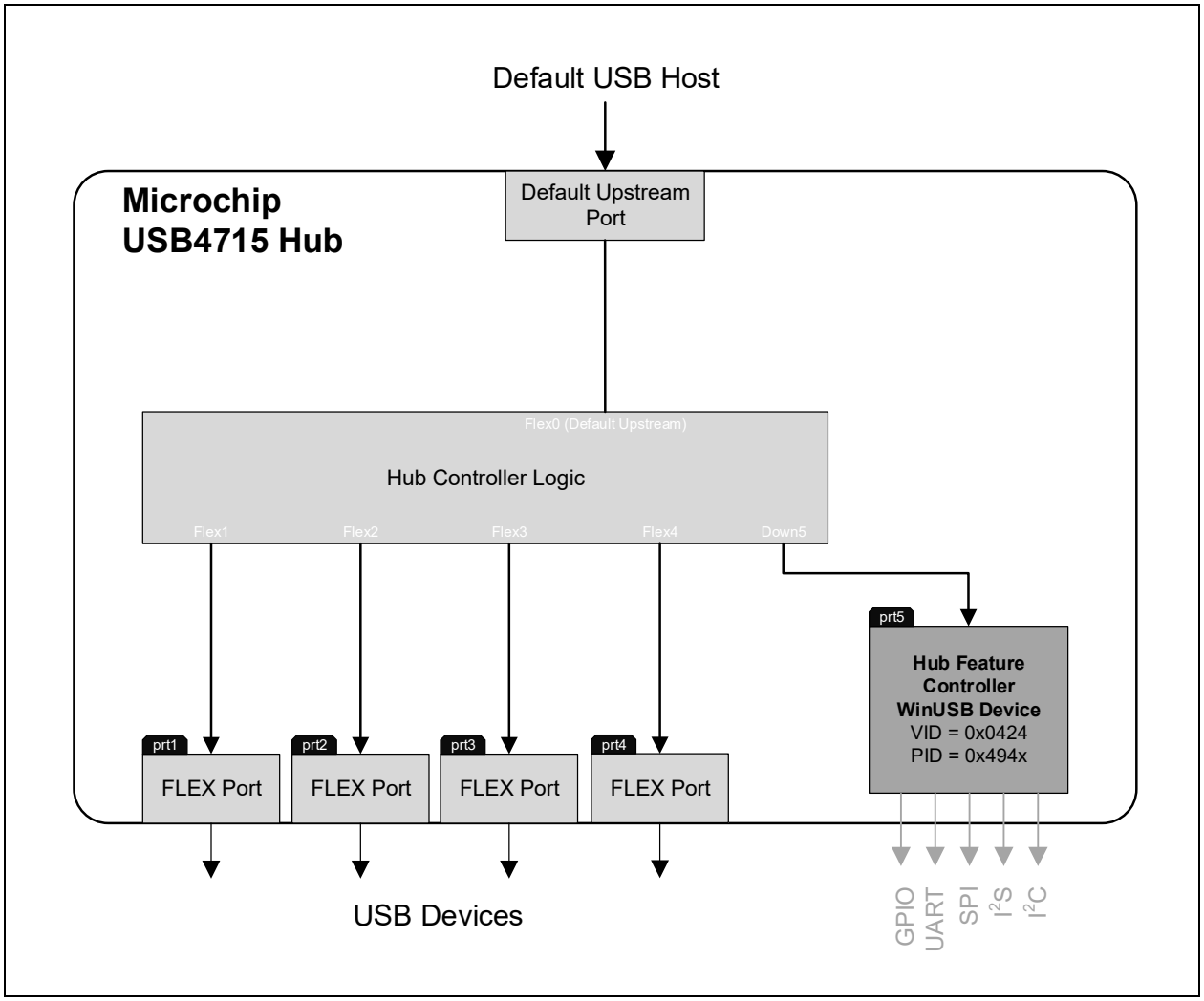
FUNCTIONAL OVERVIEW

The internal block diagram of Microchip USB4715 is shown in Figure 1.

The USB4715 has an internal USB device (or the HFC), which enables the advanced features of the hub. The HFC device is a standard WinUSB class device with a Product ID (PID) of 0x494x (where “x” depends on the configuration mode selected via the CFG_STRAP[1:0] pins).

The HFC is permanently connected to the internal port 5 on the USB4715.This port is configured as non-removable.

FIGURE 1: USB4715 INTERNAL BLOCK DIAGRAM



FlexConnect Initiation

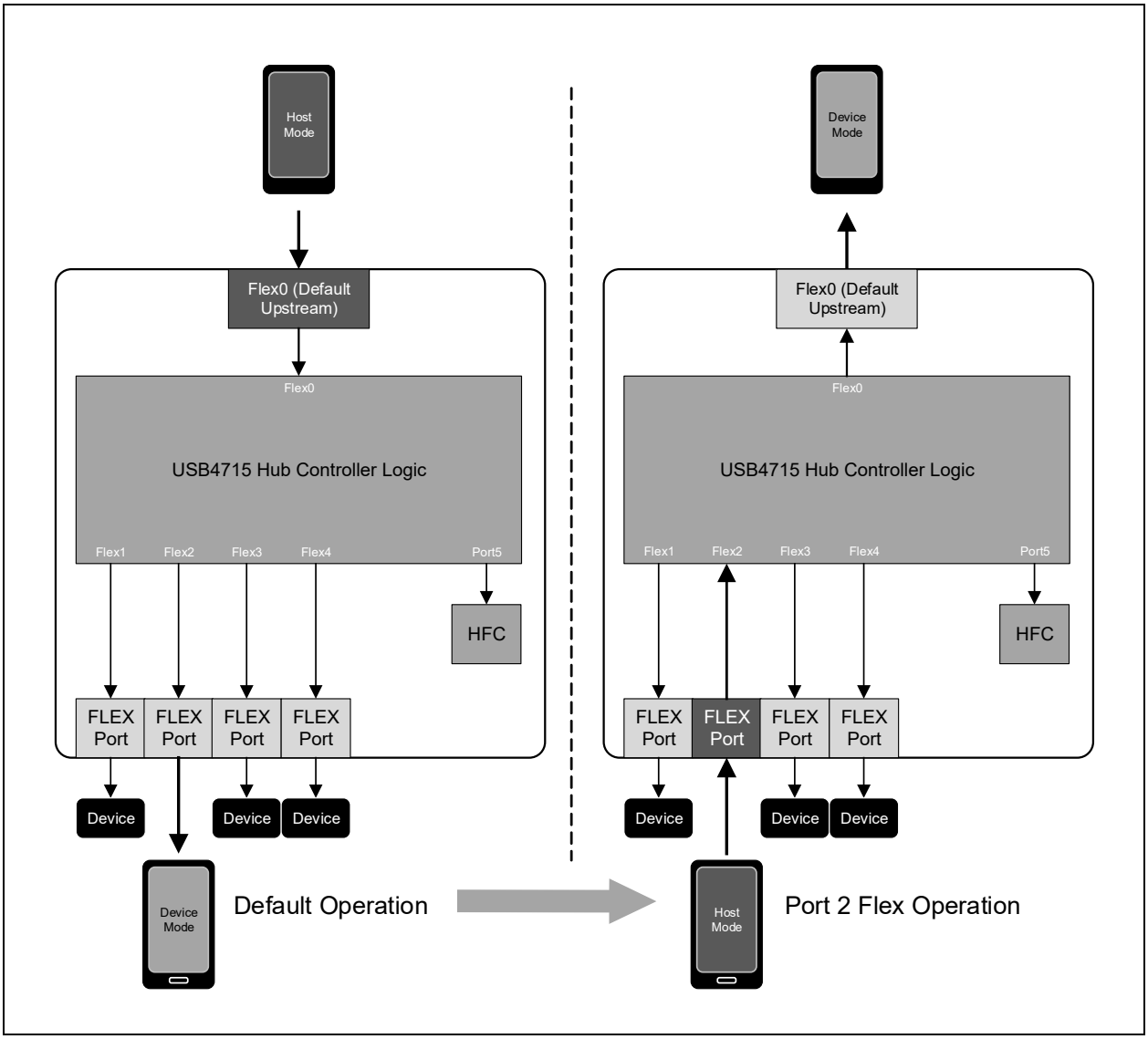
FlexConnect can be initiated via SMBus control or a USB command to the internal HFC device. The SMBus and runtime register details are located in FlexConnect Register Control on page 5. The details of this USB command are shown in FlexConnect USB Command Details on page 6, and an example USB protocol trace capture of the command is detailed in FlexConnect USB Command Example on page 7.

When FlexConnect is initiated, the current host port changes to a device port, and the designated port changes into a host port. This can be used for Host Swapping or Host Sharing type applications.

Host Swapping

In a Host Swapping application, the host role is exchanged between two dual-role-capable devices. This host mode switching can be initiated through ID pin control, protocol handshake, or some other proprietary method. An example is shown in Figure 2.

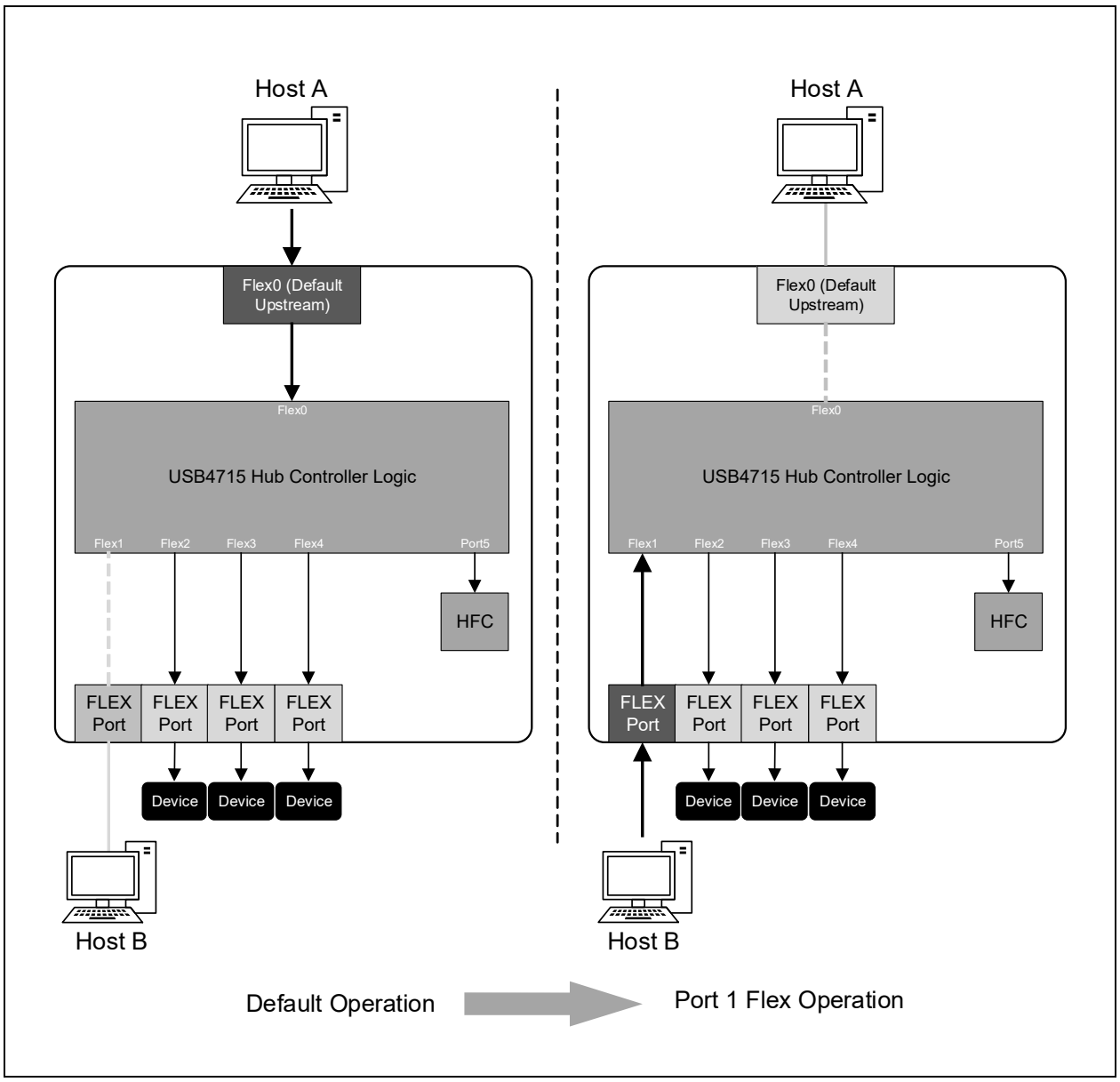
FIGURE 2: FLEXCONNECT "HOST SWAPPING" EXAMPLE



Host Sharing

In a Host Sharing application, multiple hosts are connected to the hub, and only one of these hosts has access to the USB device tree at a time. All other hosts are effectively ‘disconnected’ from the USB tree when they do not control the host port. An example of Host Sharing functionality is shown in Figure 3.

FIGURE 3: FLEXCONNECT “HOST SHARING” EXAMPLE



FLEXCONNECT REGISTER CONTROL

The FlexConnect feature can be controlled and configured via SMBus control. This method is useful for systems with an embedded microcontroller that can control the state of the FlexConnect feature via SMBus. In addition, this same register can be written to by a host during runtime to initiate FlexConnect.

After initiating the FlexConnect mode, the hub automatically performs ‘unflex’ and returns to the default Port 0 Host state if one of the following conditions is met:

- The new Flex host sends a USB command that commands the hub to return to the default Unflexed state.
- The new Flex host is disconnected any time after the Flex host has enumerated the hub.
- The hub has not yet been enumerated by the Flex host after the ENUM_TIMEOUT expires.
- The VBUS_DET on either the old upstream port or the new upstream port goes low.

To initiate FlexConnect, modify the registers defined in Table 1 accordingly.

TABLE 1: FLEXCONNECT FEATURE CONTROL REGISTER 1

FLEX_FEATURE (BFD2_3440h)			FlexConnect Feature Control Register 1
Bit	**Name**	**R/W**	**Description**
7:3	Reserved	R	Reserved (must always be ‘0’)
2:0	PORT	R/W	Physical downstream port to initiate or terminate the FlexConnect session 000 = No FlexConnect or Terminate FlexConnect session 001 = Downstream Port 1 is swapped with the upstream port. 010 = Downstream Port 2 is swapped with the upstream port. 011 = Downstream Port 3 is swapped with the upstream port. 100 = Downstream Port 4 is swapped with the upstream port. All others = ***Invalid***

FLEXCONNECT USB COMMAND DETAILS

A special USB command can be issued from the USB host to control the FlexConnect feature.

After sending the special USB command to initiate the FlexConnect mode, the hub automatically performs ‘unflex’ and returns to the default Port 0 Host state if one of the following conditions is met:

- The new Flex host sends a USB command that commands the hub to return to the default Unflexed state.
- The new Flex host is disconnected any time after the Flex host has enumerated the hub.
- The hub has not yet been enumerated by the Flex host after the ENUM_TIMEOUT expires.
- The VBUS_DET on either the old upstream port or the new upstream port goes low.

The USB command is a NO DATA control transfer that must be issued to Endpoint 0 of the internal HFC device. On USB4715, the HFC is the internal device located on port 6. The SETUP command format is shown in Table 2 and Table 3.

TABLE 2: MULTI-HOST BRIDGE SETUP PACKET

Setup Parameter	Value	Description
bmRequestType	0x41	Device-to-host, vendor class, targeted to interface
bRequest	0x90	SET_ROLE_SWITCH
wValue	0xYYYY	Bits detailed in Table 3
wIndex	0x0000	Reserved
wLength	00	No data

TABLE 3: WVALUE DETAIL OF FLEXCONNECT COMMAND SETUP PACKET

Bit	Name	Description
15:11	Reserved	Reserved (must always be ‘0’)
10:8	ENUM_TIMEOUT	These bits control a timer that is started after FlexConnect is initiated. If the Flex host does not enumerate the hub before the timer expires, the hub will revert to the Unflexed state. If the Flex host enumerates the hub within the time defined, then the hub will remain in the Flexed state. 000 = No timeout defined (hub will never automatically ‘unflex’) 001 = 10 ms 010 = 100 ms 011 = 500 ms 100 = 1 second 101 = 5 seconds 110 = 10 seconds 111 = 20 seconds
7	Reserved	Reserved (must always be ‘0’)
6:5	ROLE_SWITCH_TYPE	Must always write ‘01’ for FlexConnect
4	ROLE_SWITCH_STATE	1 = Transitions the port defined in PORT bits to the FlexConnect mode 0 = Exits the FlexConnect mode currently active on port defined in PORT bits
3:0	PORT	Physical downstream port to initiate or terminate Multi-Host Bridging session 0001 = Port 1 0010 = Port 2 0011 = Port 3 0100 = Port 4 All others = ***Invalid***

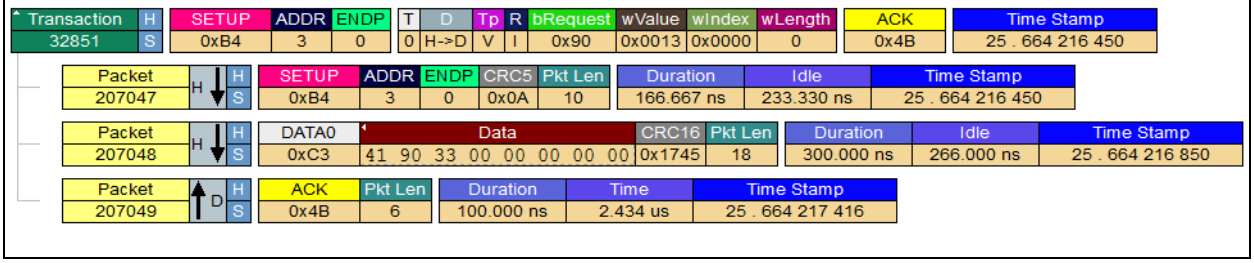
FlexConnect USB Command Example

An example of a FlexConnect initialization command for port 3 is shown in Table 4. This command is sent to Endpoint 0 of the HFC.

TABLE 4: FLEXCONNECT SETUP COMMAND EXAMPLE

Setup Parameter	Value	Note
bmRequestType	0x41	Device-to-host, vendor class, targeted to interface
bRequest	0x90	SET_ROLE_SWITCH
wValue	0x0033	Bits 15:11 = 0000b Bits 10:8 = 000b (no enum timer) Bit 7 = 0b Bits 6:5 = 01b Bit 4 = 1b (enter the FlexConnect mode) Bits 3:0 = 0011b (Port 3)
wIndex	0x0000	Reserved
wLength	00	No data

FIGURE 4: REGISTER READ SETUP TRANSACTION EXAMPLE



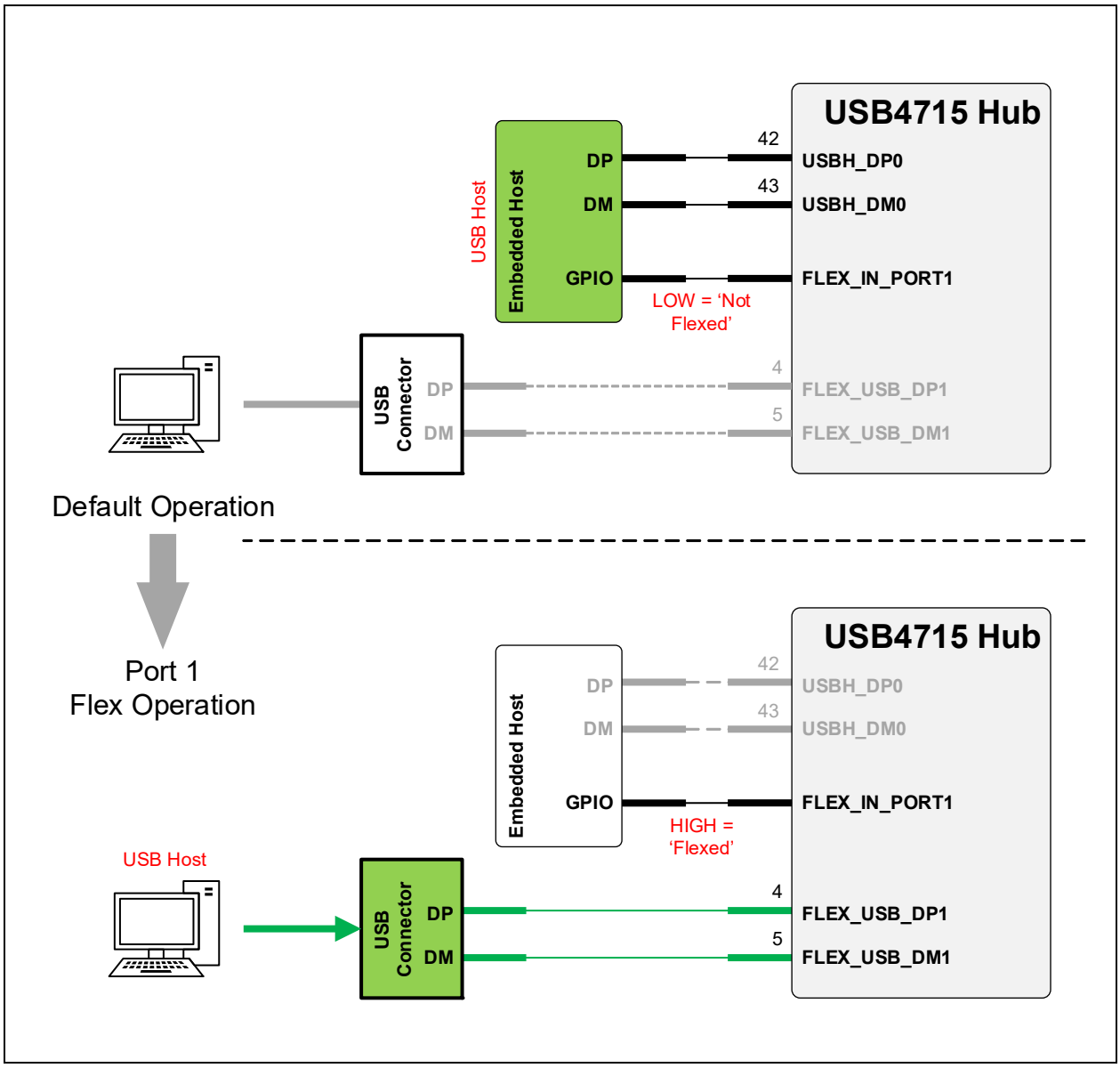
FLEXCONNECT PIN CONTROL

The FlexConnect feature can be initiated via direct pin control for systems that have an embedded MCU, which controls the state of FlexConnect. A select number of available PROG_FUNCx/GPIO pins in configuration modes 2 and 4 may be assigned the role of FLEX_IN_PORTx.

If using the direct pin control method, each port that utilizes the FlexConnect feature must also be assigned its own FLEX_IN_PORTx signal.

An example of how an embedded USB host may use the direct pin control method is shown in Figure 5..

FIGURE 5: FLEXCONNECT PIN CONTROL EXAMPLE



By default, FLEX_IN_PORT[1:4] pins are not assigned and hence not active. Only ports that are to be controlled via direct pin control must have a PROG_FUNC/GPIO pin assigned. Care must be taken to ensure that the assigned pin does not already have a critical pin function assigned to it.

Once one or more FLEX_IN_PORT[1:4] pins have been assigned, the pins behave as shown in Table 5. Note that only one FLEX_IN_PORT[1:4] should be asserted at once.

TABLE 5: EXAMPLE FLEXCONNECT STATE TRUTH TABLE

FLEX_IN_PORT4	FLEX_IN_PORT3	FLEX_IN_PORT2	FLEX_IN_PORT1	FlexConnect State
0	0	0	0	Port 0 = Host Port
0	0	0	**1**	Port 1 = Host Port
0	0	**1**	0	Port 2 = Host Port
0	**1**	0	0	Port 3 = Host Port
1	0	0	0	Port 4 = Host Port
All Others				INVALID STATE

Note: The SMBus and USB Command methods both have specific conditions where the hub will automatically perform ‘unflex’ and return to the default Port 0 Host state. When using the direct pin control method, the hub will always follow the state of the pins and will never attempt to automatically perform ‘unflex’.

The pin assignment registers are detailed in Table 6 to Table 9. These can be modified via the hub’s internal OTP memory, or through the SMBus interface during hub start-up configuration. These pins cannot be modified via runtime register writes.

TABLE 6: PORT 1 FLEXCONNECT TRIGGER PIO CONFIGURATION REGISTER

FLEX_IN_PORT1 (BFD2_3442h)			FlexConnect Trigger PIO Configuration Register
Bit	**Name**	**R/W**	**Description**
7	FLEX_IN_EN	R/W	1 - Enables FlexConnect trigger for port 1 through GPIO control. The specific GPIO pin is selected in bits [2:0] of this register. The FlexConnect state is entered upon the rising edge of the selected GPIO. The FlexConnect state is exited upon the falling edge of the selected GPIO. 0 - Disables FlexConnect pin trigger for port 1. All other bits in this register are ignored.
6:3	Reserved	R	Reserved
2:0	FLEX_IN_IO	R/W	Selects the GPIO to be used for the Port 1 FlexConnect trigger. 000 - Reserved 001 - Reserved 010 - PROG_FUNC4/GPIO6 (only in CONFIG2/4 modes) 011 - PROG_FUNC5/GPIO8 (only in CONFIG2/4 modes) 100 - PROG_FUNC6/GPIO10 (only in CONFIG2/4 modes) 101 - PROG_FUNC7/GPIO11 (only in CONFIG2/4 modes) 110 - Reserved 111 - Reserved

TABLE 7: PORT 2 FLEXCONNECT TRIGGER PIO CONFIGURATION REGISTER

FLEX_IN_PORT2 (BFD2_3443h)			FlexConnect Trigger PIO Configuration Register
Bit	**Name**	**R/W**	**Description**
7	FLEX_IN_EN	R/W	1 - Enables FlexConnect trigger for port 2 through GPIO control. The specific GPIO pin is selected in bits [2:0] of this register. The FlexConnect state is entered upon the rising edge of the selected GPIO. The FlexConnect state is exited upon the falling edge of the selected GPIO. 0 - Disables FlexConnect pin trigger for port 2. All other bits in this register are ignored.
6:3	Reserved	R	Reserved
2:0	FLEX_IN_IO	R/W	Selects the GPIO to be used for the Port 2 FlexConnect trigger. 000 - Reserved 001 - Reserved 010 - PROG_FUNC4/GPIO6 (only in CONFIG2/4 modes) 011 - PROG_FUNC5/GPIO8 (only in CONFIG2/4 modes) 100 - PROG_FUNC6/GPIO10 (only in CONFIG2/4 modes) 101 - PROG_FUNC7/GPIO11 (only in CONFIG2/4 modes) 110 - Reserved 111 - Reserved

TABLE 8: PORT 3 FLEXCONNECT TRIGGER PIO CONFIGURATION REGISTER

FLEX_IN_PORT3 (BFD2_3444h)			FlexConnect Trigger PIO Configuration Register
Bit	**Name**	**R/W**	**Description**
7	FLEX_IN_EN	R/W	1 - Enables FlexConnect trigger for port 3 through GPIO control. The specific GPIO pin is selected in bits [2:0] of this register. The FlexConnect state is entered upon the rising edge of the selected GPIO. The FlexConnect state is exited upon the falling edge of the selected GPIO. 0 - Disables FlexConnect pin trigger for port 3. All other bits in this register are ignored.
6:3	Reserved	R	Reserved
2:0	FLEX_IN_IO	R/W	Selects the GPIO to be used for the Port 3 FlexConnect trigger. 000 - Reserved 001 - Reserved 010 - PROG_FUNC4/GPIO6 (only in CONFIG2/4 modes) 011 - PROG_FUNC5/GPIO8 (only in CONFIG2/4 modes) 100 - PROG_FUNC6/GPIO10 (only in CONFIG2/4 modes) 101 - PROG_FUNC7/GPIO11 (only in CONFIG2/4 modes) 110 - Reserved 111 - Reserved

TABLE 9: PORT 4 FLEXCONNECT TRIGGER PIO CONFIGURATION REGISTER

FLEX_IN_PORT4 (BFD2_3445h)			FlexConnect Trigger PIO Configuration Register
Bit	**Name**	**R/W**	**Description**
7	FLEX_IN_EN	R/W	1 - Enables FlexConnect trigger for port 4 through GPIO control. The specific GPIO pin is selected in bits [2:0] of this register. The FlexConnect state is entered upon the rising edge of the selected GPIO. The FlexConnect state is exited upon the falling edge of the selected GPIO. 0 - Disables FlexConnect pin trigger for port 4. All other bits in this register are ignored.
6:3	Reserved	R	Reserved
2:0	FLEX_IN_IO	R/W	Selects the GPIO to be used for the Port 4 FlexConnect trigger. 000 - Reserved 001 - Reserved 010 - PROG_FUNC4/GPIO6 (only in CONFIG2/4 modes) 011 - PROG_FUNC5/GPIO8 (only in CONFIG2/4 modes) 100 - PROG_FUNC6/GPIO10 (only in CONFIG2/4 modes) 101 - PROG_FUNC7/GPIO11 (only in CONFIG2/4 modes) 110 - Reserved 111 - Reserved

FLEX_CONFIGURATION SPACE

The Flex_Configuration (FLEXCFG) space is a space within the hub that allows programming of registers. The features defined in these registers will only be implemented once the hub flexed. Example features include port enable/disable, battery charging configuration, and more. The FLEXCFG space follows a similar structural format as One-Time Programmable (OTP). Table 10, Table 11, and Table 12 show the registers associated with FLEXCFG.

TABLE 10: FLEX CONFIGURATION CONTROL REGISTER

FLEXCFG_CT (BFD2_345Bh)			Flex Configuration Control Register
Bit	**Name**	**R/W**	**Description**
7:1	Reserved	R	Reserved
0	APPLY_FLEXCFG	R/W	1 - FLEXCFG configuration of length specified in the FLEXCFG_LEN register will be applied during the Flex state transition. 0 - FLEXCFG configuration will not be applied on the next Flex state transition.

TABLE 11: FLEX CONFIGURATION LENGTH REGISTERS

FLEXCFG_LEN (BFD2_345Ch-BFD2_345D)			Flex Configuration Length Registers
Bit	**Name**	**R/W**	**Description**
15:10	Reserved	R	Reserved
9:0	FLEXCFG_LEN	R/W	This field indicates the length of the FLEXCFG data.

TABLE 12: FLEX CONFIGURATION DATA REGISTERS

FLEXCFG_DATA (BFD2_3800h-BFD2_39FF)			Flex Configuration Data Registers
Bit	**Name**	**R/W**	**Description**
7:0	FLEXCFG_DATA	R/W	This is where bytes from FLEXCFG configuration are programmed.

The recommended procedure for implementing FLEXCFG is as follows:

1. Create a configuration file with all the options that will be applied after flexing.
2. Program the bytes from the configuration file starting at the first register of FLEXCFG_DATA (BFD2_3800h).
3. Program the length of the data in the configuration file/FLEXCFG_DATA area into the FLEXCFG_LEG register.
4. Set the APPLY_FLEXCFG bit in FLEXCFG_CTL_REG.
5. Send the FlexConnect command.

At this point, the data from FLEXCFG_DATA registers of FLEXCFG_LEN length are applied to the hub and is visible to the new flexed host. The ports revert to their original state before flexing and after unflexing.

If the same configuration will be programmed when the next FlexConnect occurs, rewrite the length of the FLEXCFG patch in FLEXCFG_LEN and send the FlexConnect command. If a new data will be programmed, clear out the data in the FLEXCFG_DATA area then repeat and follow the recommended procedure in steps 1 to 5 above.

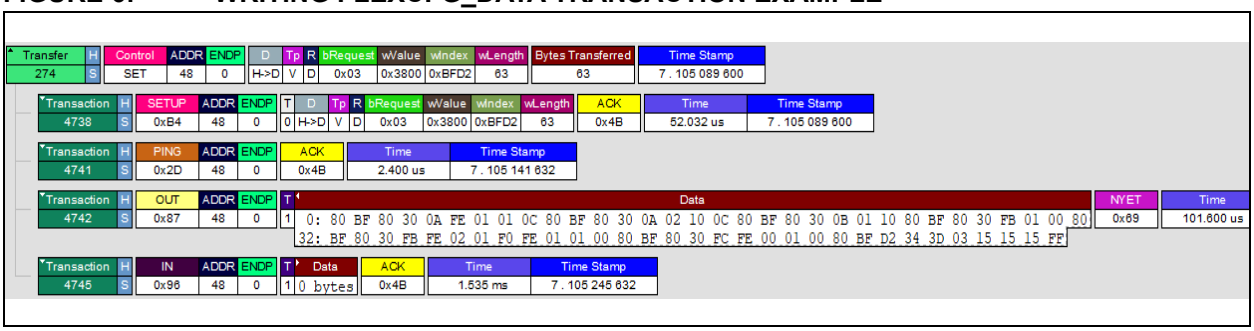
The following is an example configuration that might be programmed into the FLEXCFG space, as well as a USB trace of the process.

Step 1: Create a configuration file with all the options that will be applied after flexing.

In this example, FlexConnect is sent to port 1. This configuration file disables ports 2-4 and sets them to be dedicated charging ports (DCP). After flexing, the new host sees a 2-port hub with the other three ports disabled and set to DCP.

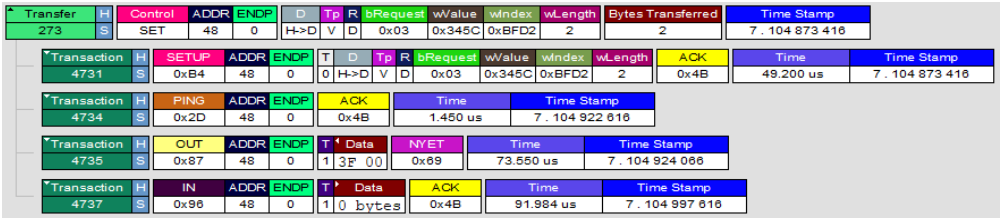
Step 2: Program the bytes from the configuration file starting at the first register of FLEXCFG_DATA (BFD2_3800h).

FIGURE 6: WRITING FLEXCFG_DATA TRANSACTION EXAMPLE



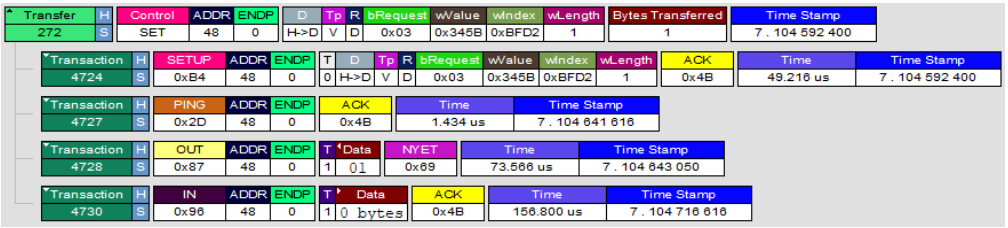
Step 3: Program the length of the data in the configuration file/FLEXCFG_DATA area into the FLEXCFG_LEG register.

FIGURE 7: WRITING FLEXCFG_LEN TRANSACTION EXAMPLE



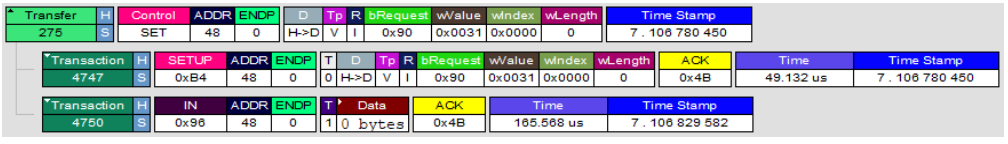
Step 4: Set the APPLY_FLEXCFG bit in FLEXCFG_CTL_REG.

FIGURE 8: WRITING FLEXCFG_CTL TRANSACTION EXAMPLE



Step 5: Send the FlexConnect command.

FIGURE 9: SENDING FLEXCONNECT COMMAND TO PORT 1 TRANSACTION EXAMPLE



Note: If battery charging will be enabled on a port via the FLEXCFG area, as in the example above, battery charging **must** also be enabled via OTP or strapping in the original unflexed configuration. For example, if port 2 will be configured as a battery charging port in the FLEXCFG area then battery charging must be enabled via the CFG_BC_EN strap or OTP write in the default Unflexed state.

ADDITIONAL HARDWARE CONSIDERATIONS

Schematics for FlexConnect applications are implementation-specific, and hence do not necessarily follow any common framework. The following sections describe the common considerations that must be made for each design.

VBUS_DET

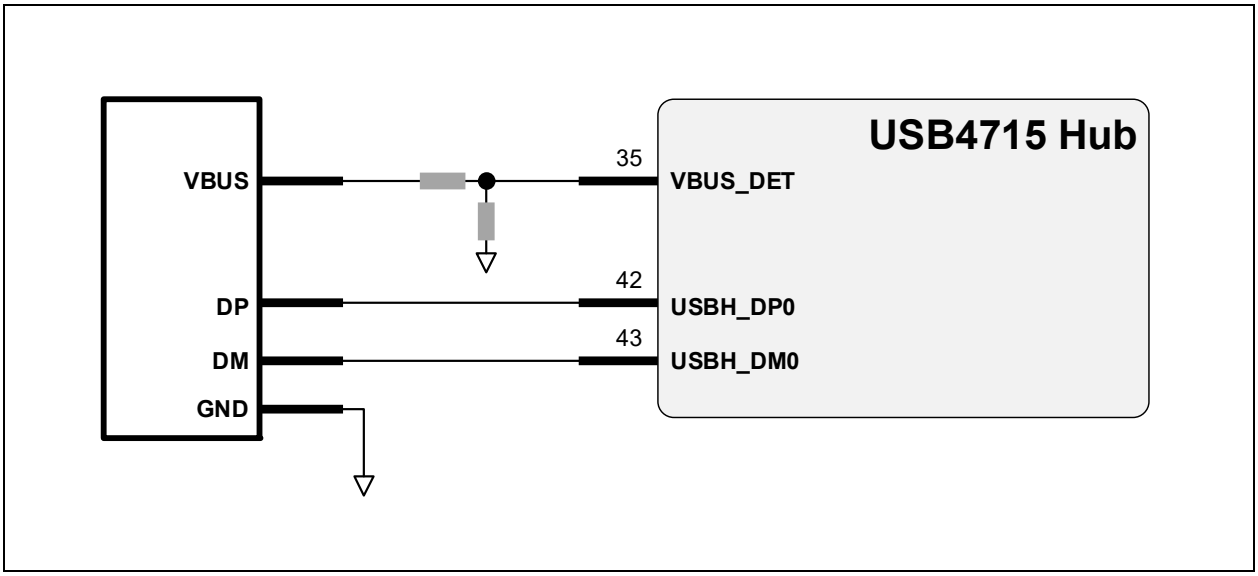
VBUS_DET is a signal used by the hub to determine when a USB host is connected. When VBUS_DET is not present, the hub enters a low-power state to conserve energy. This causes the hub to disconnect when enumerated either through the upstream port or through the new FlexConnect upstream port.

NO POWER ROLE SWAP VBUS_DET IMPLEMENTATION

If power roles do not change when initiating FlexConnect, it might be appropriate to connect VBUS_DET directly to the VBUS pin of the default USB host through a resistor divider. This is the standard VBUS_DET implementation, as shown in Figure 10. This is a popular implementation in the automotive application space as the head unit is always connected to the hub.

Note:	Ensure that the default USB host will not toggle its VBUS supply at any time when FlexConnect is initiated. Otherwise, the hub will be reset and revert to its default state.

FIGURE 10: NO POWER ROLE SWAP VBUS_DET CONFIGURATION



CONNECT VBUS_DET DIRECTLY TO A FIXED 3.3V

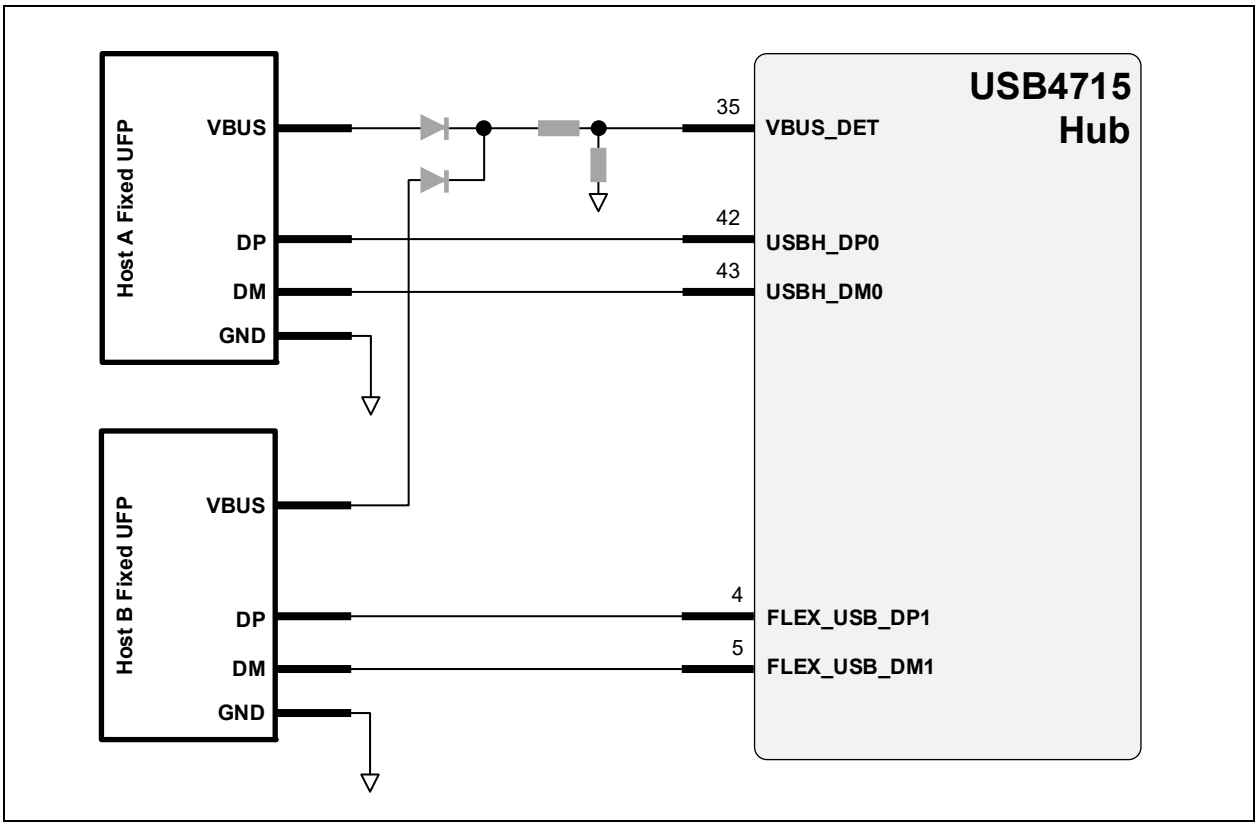
If low-power states are not required when no USB host is present, it is acceptable to connect VBUS_DET to a fixed 3.3V supply on the PCB. This method is only recommended for designs in which the hub is connected to an embedded host IC.

Note:	In some instances, a USB host might attempt to force a hard reset on a device by toggling VBUS. In this instance, the hub will not reset.

‘OR’ ALL HOST VBUS CONNECTIONS TO VBUS_DET

In Host Sharing type systems, ‘OR’ing all Host VBUS connections would be an appropriate solution, as shown in Figure 11. This can be used when the “one host at a time” implementation is used. The drawback with this type of solution is that there is no way to distinguish which host port is actually supplying VBUS.

FIGURE 11: DIODE-OR VBUS_DET SIGNALS



Note: The resistor divider values change depending on the characteristics of the diodes used.

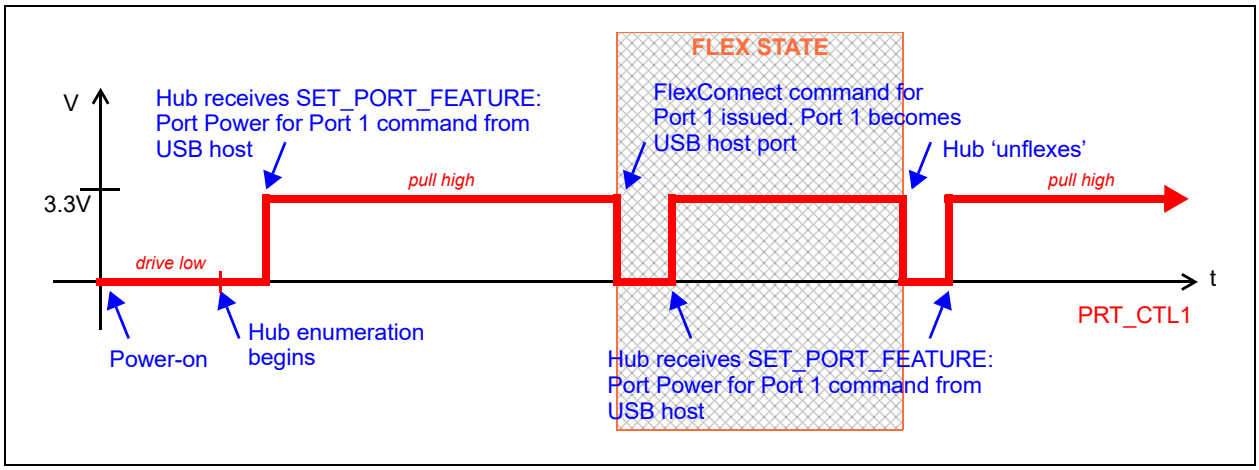
Port 1-4 Port Power Control

The required behavior of the PRT_CTLx pins should be considered before and after the FlexConnect mode is initiated during the system architecture design phase of the application.

By default, the PRT_CTLx pin continues to respond as the port control pin for its respective logical port in both the Unflexed and Flexed states.

Figure 12 describes how a PRT_CTLx pin operates by default when the FlexConnect mode is entered or exited. The example shown is for port 1 (PRT_CTL1), but the behavior is identical for all ports.

FIGURE 12: PRT_CTLX DURING FLEXCONNECT OPERATION WHEN FLEX_PRTCTL_EN = 0



FlexConnect State Indicator Output

GPIOs may also be assigned the role of FlexConnect status indicator outputs. The use of these outputs is completely optional. These outputs can be connected to external LEDs or an embedded microcontroller to communicate the current FlexConnect state.

The GPIO can be configured as drive-high or drive-low or open drain, and the polarity can be configured. This mode should not be used in combination with the FlexConnect GPIO input mode.

TABLE 13: PORT 1 FLEXCONNECT STATE INDICATOR GPIO CONFIGURATION REGISTER

FLEX_OUT_PORT1 (BFD2_3446h)			FlexConnect State Indicator GPIO Configuration
Bit	**Name**	**R/W**	**Description**
7	FLEX_OUT_EN	R/W	1 - Enables the FlexConnect state indicator for port 1 through GPIO. The specific GPIO is selected in bits [2:0] of this register. 0 - Disables the FlexConnect state indicator for port 1 through GPIO. All other bits in this register are ignored.
6	FLEX_OUT_ACTIVE_HIGH	R/W	1 - The selected GPIO is driven active-high while in the FlexConnect state; otherwise, it is driven low. 0 - The selected GPIO is driven active low while in the FlexConnect state; otherwise, it is driven high.
5	FLEX_OUT_OD	R/W	1 - The selected GPIO is in the open-drain output mode. 0 - The selected GPIO is in the drive-high or drive-low output mode.
4:3	Reserved	R	Reserved
2:0	FLEX_OUT_IO	R/W	Selects the GPIO to be used for the Port 1 FlexConnect state indicator. 000 - Reserved 001 - Reserved 010 - PROG_FUNC4/GPIO6 (Only in CONFIG2/4 modes) 011 - PROG_FUNC5/GPIO8 (Only in CONFIG2/4 modes) 100 - PROG_FUNC6/GPIO10 (Only in CONFIG2/4 modes) 101 - PROG_FUNC7/GPIO11 (Only in CONFIG2/4 modes) 110 - Reserved 111 - Reserved

TABLE 14: PORT 2 FLEXCONNECT STATE INDICATOR GPIO CONFIGURATION REGISTER

FLEX_OUT_PORT2 (BFD2_3447h)			FlexConnect State Indicator GPIO Configuration
Bit	**Name**	**R/W**	**Description**
7	FLEX_OUT_EN	R/W	1 - Enables the FlexConnect state indicator for port 2 through GPIO. The specific GPIO is selected in bits [2:0] of this register. 0 - Disables the FlexConnect state indicator for port 2 through GPIO. All other bits in this register are ignored.
6	FLEX_OUT_ACTIVE_HIGH	R/W	1 - The selected GPIO is driven active-high while in the FlexConnect state; otherwise, it is driven low. 0 - The selected GPIO is driven active-low while in the FlexConnect state; otherwise, it is driven high.
5	FLEX_OUT_OD	R/W	1 - The selected GPIO is in the open-drain output mode. 0 - The selected GPIO is in the drive-high or drive-low output mode.
4:3	Reserved	R	Reserved
2:0	FLEX_OUT_IO	R/W	Selects the GPIO to be used for the Port 2 FlexConnect state indicator. 000 - Reserved 001 - Reserved 010 - PROG_FUNC4/GPIO6 (Only in CONFIG2/4 modes) 011 - PROG_FUNC5/GPIO8 (Only in CONFIG2/4 modes) 100 - PROG_FUNC6/GPIO10 (Only in CONFIG2/4 modes) 101 - PROG_FUNC7/GPIO11 (Only in CONFIG2/4 modes) 110 - Reserved 111 - Reserved

TABLE 15: PORT 3 FLEXCONNECT STATE INDICATOR GPIO CONFIGURATION REGISTER

FLEX_OUT_PORT3 (BFD2_3448h)			FlexConnect State Indicator GPIO Configuration
Bit	**Name**	**R/W**	**Description**
7	FLEX_OUT_EN	R/W	1 - Enables the FlexConnect state indicator for port 3 through GPIO. The specific GPIO is selected in bits [2:0] of this register. 0 - Disables the FlexConnect state indicator for port 3 through GPIO. All other bits in this register are ignored.
6	FLEX_OUT_ACTIVE_HIGH	R/W	1 - The selected GPIO is driven active-high while in the FlexConnect state; otherwise, it is driven low. 0 - The selected GPIO is driven active-low while in the FlexConnect state; otherwise, it is driven high.
5	FLEX_OUT_OD	R/W	1 - The selected GPIO is in the open-drain output mode. 0 - The selected GPIO is in the drive-high or drive-low output mode.
4:3	Reserved	R	Reserved
2:0	FLEX_OUT_IO	R/W	Selects the GPIO to be used for the Port 3 FlexConnect state indicator. 000 - Reserved 001 - Reserved 010 - PROG_FUNC4/GPIO6 (Only in CONFIG2/4 modes) 011 - PROG_FUNC5/GPIO8 (Only in CONFIG2/4 modes) 100 - PROG_FUNC6/GPIO10 (Only in CONFIG2/4 modes) 101 - PROG_FUNC7/GPIO11 (Only in CONFIG2/4 modes) 110 - Reserved 111 - Reserved

TABLE 16: PORT 4 FLEXCONNECT STATE INDICATOR GPIO CONFIGURATION REGISTER

FLEX_OUT_PORT4 (BFD2_3449h)			FlexConnect State Indicator GPIO Configuration
Bit	**Name**	**R/W**	**Description**
7	FLEX_OUT_EN	R/W	1 - Enables the FlexConnect state indicator for port 4 through GPIO. The specific GPIO is selected in bits [2:0] of this register. 0 - Disables the FlexConnect state indicator for port 4 through GPIO. All other bits in this register are ignored.
6	FLEX_OUT_ACTIVE_HIGH	R/W	1 - The selected GPIO is driven active-high while in the FlexConnect state; otherwise, it is driven low. 0 - The selected GPIO is driven active-low while in the FlexConnect state; otherwise, it is driven high.
5	FLEX_OUT_OD	R/W	1 - The selected GPIO is in the open-drain output mode. 0 - The selected GPIO is in the drive-high or drive-low output mode.
4:3	Reserved	R	Reserved
2:0	FLEX_OUT_IO	R/W	Selects the GPIO to be used for the Port 4 FlexConnect state indicator. 000 - Reserved 001 - Reserved 010 - PROG_FUNC4/GPIO6 (Only in CONFIG2/4 modes) 011 - PROG_FUNC5/GPIO8 (Only in CONFIG2/4 modes) 100 - PROG_FUNC6/GPIO10 (Only in CONFIG2/4 modes) 101 - PROG_FUNC7/GPIO11 (Only in CONFIG2/4 modes) 110 - Reserved 111 - Reserved

APPENDIX A: APPLICATION NOTE REVISION HISTORY

TABLE A-1: REVISION HISTORY

Revision Level & Date	Section/Figure/Entry	Correction
DS00002341C (12-12-18)	All	Mentions of SMBus replaced with Register. Minor text changes throughout. Enhancement of graphics.
	FlexConnect Register Control on page 5	Updated this section. Updated Table 1 updated, and the removed “FlexConnect Feature Control Register 2” table.
	Table 6, Table 7, and Table 8	Updated the description of FLEX_IN_IO bit.
	Flex_Configuration Space on page 12	Added this section.
	FlexConnect USB Command Details on page 6, FlexConnect Pin Control on page 8, and Additional Hardware Considerations on page 14	Updated these sections. Under Additional Hardware Considerations: • Section title changed to No Power Role Swap VBUS_DET Implementation. • Updated Figure 11 and added a new note under this figure. • Removed “Assign a New VBUS_DET Pin to an Available GPIO” section. • Removed “PRT_CTLX/PRT_CTL0 DURING FLEXCONNECT OPERATION WHEN FLEX-_PRTCTL_EN = 1”. • Added FlexConnect State Indicator Output section.
	Table 13, Table 14, Table 15, and Table 16	Updated the description of FLEX_OUT_IO bit.
DS00002341B (08-22-17)	Table 6 to Table 16	Updated FLEX_IN_IO description.
	All	Corrected minor typos and pagination issues.
	Table 4	Updated wValue description.
	Figure 2	Corrected text on green Port 2 device.
DS00002341A (03-28-17)	All	Initial release.

THE MICROCHIP WEB SITE

Microchip provides online support via our WWW site at www.microchip.com. This web site is used as a means to make files and information easily available to customers. Accessible by using your favorite Internet browser, the web site contains the following information:

- **Product Support** – Data sheets and errata, application notes and sample programs, design resources, user's guides and hardware support documents, latest software releases and archived software
- **General Technical Support** – Frequently Asked Questions (FAQ), technical support requests, online discussion groups, Microchip consultant program member listing
- **Business of Microchip** – Product selector and ordering guides, latest Microchip press releases, listing of seminars and events, listings of Microchip sales offices, distributors and factory representatives

CUSTOMER CHANGE NOTIFICATION SERVICE

Microchip's customer notification service helps keep customers current on Microchip products. Subscribers will receive e-mail notification whenever there are changes, updates, revisions or errata related to a specified product family or development tool of interest.

To register, access the Microchip web site at www.microchip.com. Under "Support", click on "Customer Change Notification" and follow the registration instructions.

CUSTOMER SUPPORT

Users of Microchip products can receive assistance through several channels:

- Distributor or Representative
- Local Sales Office
- Field Application Engineer (FAE)
- Technical Support

Customers should contact their distributor, representative or Field Application Engineer (FAE) for support. Local sales offices are also available to help customers. A listing of sales offices and locations is included in the back of this document.

Technical support is available through the web site at: http://microchip.com/support

Note the following details of the code protection feature on Microchip devices:

- Microchip products meet the specification contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is one of the most secure families of its kind on the market today, when used in the intended manner and under normal conditions.
- There are dishonest and possibly illegal methods used to breach the code protection feature. All of these methods, to our knowledge, require using the Microchip products in a manner outside the operating specifications contained in Microchip's Data Sheets. Most likely, the person doing so is engaged in theft of intellectual property.
- Microchip is willing to work with the customer who is concerned about the integrity of their code.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of their code. Code protection does not mean that we are guaranteeing the product as "unbreakable."

Code protection is constantly evolving. We at Microchip are committed to continuously improving the code protection features of our products. Attempts to break Microchip's code protection feature may be a violation of the Digital Millennium Copyright Act. If such acts allow unauthorized access to your software or other copyrighted work, you may have a right to sue for relief under that Act.

ISBN: 978-1-5224-3954-7

QUALITY MANAGEMENT SYSTEM
CERTIFIED BY DNV
═ ISO/TS 16949 ═

Microchip received ISO/TS-16949:2009 certification for its worldwide headquarters, design and wafer fabrication facilities in Chandler and Tempe, Arizona; Gresham, Oregon and design centers in California and India. The Company's quality system processes and procedures are for its PIC® MCUs and dsPIC® DSCs, KEELOQ® code hopping devices, Serial EEPROMs, microperipherals, nonvolatile memory and analog products. In addition, Microchip's quality system for the design and manufacture of development systems is ISO 9001:2000 certified.

Worldwide Sales and Service

AMERICAS

Corporate Office
2355 West Chandler Blvd.
Chandler, AZ 85224-6199
Tel: 480-792-7200
Fax: 480-792-7277
Technical Support:
http://www.microchip.com/support
Web Address:
www.microchip.com

Atlanta
Duluth, GA
Tel: 678-957-9614
Fax: 678-957-1455

Austin, TX
Tel: 512-257-3370

Boston
Westborough, MA
Tel: 774-760-0087
Fax: 774-760-0088

Chicago
Itasca, IL
Tel: 630-285-0071
Fax: 630-285-0075

Dallas
Addison, TX
Tel: 972-818-7423
Fax: 972-818-2924

Detroit
Novi, MI
Tel: 248-848-4000

Houston, TX
Tel: 281-894-5983

Indianapolis
Noblesville, IN
Tel: 317-773-8323
Fax: 317-773-5453
Tel: 317-536-2380

Los Angeles
Mission Viejo, CA
Tel: 949-462-9523
Fax: 949-462-9608
Tel: 951-273-7800

Raleigh, NC
Tel: 919-844-7510

New York, NY
Tel: 631-435-6000

San Jose, CA
Tel: 408-735-9110
Tel: 408-436-4270

Canada - Toronto
Tel: 905-695-1980
Fax: 905-695-2078

ASIA/PACIFIC

Australia - Sydney
Tel: 61-2-9868-6733

China - Beijing
Tel: 86-10-8569-7000

China - Chengdu
Tel: 86-28-8665-5511

China - Chongqing
Tel: 86-23-8980-9588

China - Dongguan
Tel: 86-769-8702-9880

China - Guangzhou
Tel: 86-20-8755-8029

China - Hangzhou
Tel: 86-571-8792-8115

China - Hong Kong SAR
Tel: 852-2943-5100

China - Nanjing
Tel: 86-25-8473-2460

China - Qingdao
Tel: 86-532-8502-7355

China - Shanghai
Tel: 86-21-3326-8000

China - Shenyang
Tel: 86-24-2334-2829

China - Shenzhen
Tel: 86-755-8864-2200

China - Suzhou
Tel: 86-186-6233-1526

China - Wuhan
Tel: 86-27-5980-5300

China - Xian
Tel: 86-29-8833-7252

China - Xiamen
Tel: 86-592-2388138

China - Zhuhai
Tel: 86-756-3210040

ASIA/PACIFIC

India - Bangalore
Tel: 91-80-3090-4444

India - New Delhi
Tel: 91-11-4160-8631

India - Pune
Tel: 91-20-4121-0141

Japan - Osaka
Tel: 81-6-6152-7160

Japan - Tokyo
Tel: 81-3-6880- 3770

Korea - Daegu
Tel: 82-53-744-4301

Korea - Seoul
Tel: 82-2-554-7200

Malaysia - Kuala Lumpur
Tel: 60-3-7651-7906

Malaysia - Penang
Tel: 60-4-227-8870

Philippines - Manila
Tel: 63-2-634-9065

Singapore
Tel: 65-6334-8870

Taiwan - Hsin Chu
Tel: 886-3-577-8366

Taiwan - Kaohsiung
Tel: 886-7-213-7830

Taiwan - Taipei
Tel: 886-2-2508-8600

Thailand - Bangkok
Tel: 66-2-694-1351

Vietnam - Ho Chi Minh
Tel: 84-28-5448-2100

EUROPE

Austria - Wels
Tel: 43-7242-2244-39
Fax: 43-7242-2244-393

Denmark - Copenhagen
Tel: 45-4450-2828
Fax: 45-4485-2829

Finland - Espoo
Tel: 358-9-4520-820

France - Paris
Tel: 33-1-69-53-63-20
Fax: 33-1-69-30-90-79

Germany - Garching
Tel: 49-8931-9700

Germany - Haan
Tel: 49-2129-3766400

Germany - Heilbronn
Tel: 49-7131-67-3636

Germany - Karlsruhe
Tel: 49-721-625370

Germany - Munich
Tel: 49-89-627-144-0
Fax: 49-89-627-144-44

Germany - Rosenheim
Tel: 49-8031-354-560

Israel - Ra'anana
Tel: 972-9-744-7705

Italy - Milan
Tel: 39-0331-742611
Fax: 39-0331-466781

Italy - Padova
Tel: 39-049-7625286

Netherlands - Drunen
Tel: 31-416-690399
Fax: 31-416-690340

Norway - Trondheim
Tel: 47-7288-4388

Poland - Warsaw
Tel: 48-22-3325737

Romania - Bucharest
Tel: 40-21-407-87-50

Spain - Madrid
Tel: 34-91-708-08-90
Fax: 34-91-708-08-91

Sweden - Gothenberg
Tel: 46-31-704-60-40

Sweden - Stockholm
Tel: 46-8-5090-4654

UK - Wokingham
Tel: 44-118-921-5800
Fax: 44-118-921-5820