Intel North Mux-Agent

Introduction

North Mux-Agent is a function of the Intel PMC firmware that is supported on most Intel based platforms that have the PMC microcontroller. It's used for configuring the various USB Multiplexer/DeMultiplexers on the system. The platforms that allow the mux-agent to be configured from the operating system have an ACPI device object (node) with HID "INTC105C" that represents it.

The North Mux-Agent (aka. Intel PMC Mux Control, or just mux-agent) driver communicates with the PMC microcontroller by using the PMC IPC method (drivers/platform/x86/intel_scu_ipc.c). The driver registers with the USB Type-C Mux Class which allows the USB Type-C Controller and Interface drivers to configure the cable plug orientation and mode (with Alternate Modes). The driver also registers with the USB Role Class in order to support both USB Host and Device modes. The driver is located here: drivers/usb/typec/mux/intel_pmc_mux.c.

Port nodes

General

For every USB Type-C connector under the mux-agent control on the system, there is a separate child node under the PMC mux-agent device node. Those nodes do not represent the actual connectors, but instead the "channels" in the mux-agent that are associated with the connectors:

```
Scope (_SB.PCI0.PMC.MUX)
{
    Device (CH0)
    {
        Name (_ADR, 0)
    }

    Device (CH1)
    {
        Name (_ADR, 1)
    }
}
```

_PLD (Physical Location of Device)

The optional _PLD object can be used with the port (the channel) nodes. If _PLD is supplied, it should match the connector node _PLD:

```
Scope (_SB.PCI0.PMC.MUX)
{
    Device (CH0)
    {
        Name (_ADR, 0)
        Method (_PLD, 0, NotSerialized)
        {
            /* Consider this as pseudocode. */
            Return (\_SB.USBC.CON0._PLD())
        }
    }
}
```

Mux-agent specific _DSD Device Properties

Port Numbers

In order to configure the muxes behind a USB Type-C connector, the PMC firmware needs to know the USB2 port and the USB3 port that is associated with the connector. The driver extracts the correct port numbers by reading specific _DSD device properties named "usb2-port-number" and "usb3-port-number". These properties have integer value that means the port index. The port index number is 1's based, and value 0 is illegal. The driver uses the numbers extracted from these device properties as-is when sending the mux-agent specific messages to the PMC:

```
Name (_DSD, Package () {
    ToUUID("daffd814-6eba-4d8c-8a91-bc9bbf4aa301"),
    Package() {
        Package () {"usb2-port-number", 6},
        Package () {"usb3-port-number", 3},
    },
})
```

Orientation

Depending on the platform, the data and SBU lines coming from the connector may be "fixed" from the mux-agent's point of view, which means the mux-agent driver should not configure them according to the cable plug orientation. This can happen for example if a retimer on the platform handles the cable plug orientation. The driver uses a specific device properties "sbu-orientation" (SBU) and "hsl-orientation" (data) to know if those lines are "fixed", and to which orientation. The value that these properties have is a string value, and it can be one that is defined for the USB Type-C connector orientation: "normal" or "reversed":

```
Name (_DSD, Package () {
    ToUUID("daffd814-6eba-4d8c-8a91-bc9bbf4aa301"),
    Package() {
        Package () {"sbu-orientation", "normal"},
        Package () {"hsl-orientation", "normal"},
    },
})
```

Example ASL

The following ASL is an example that shows the mux-agent node, and two connectors under its control:

```
Scope (_SB.PCI0.PMC)
{
    Device (MUX)
    {
        Name (_HID, "INTC105C")

        Device (CH0)
        {
            Name (_ADR, 0)

            Name (_DSD, Package () {
                ToUUID("daffd814-6eba-4d8c-8a91-bc9bbf4aa301"),
                Package() {
                    Package () {"usb2-port-number", 6},
                    Package () {"usb3-port-number", 3},
                    Package () {"sbu-orientation", "normal"},
                    Package () {"hsl-orientation", "normal"},
                },
            })
        }

        Device (CH1)
        {
            Name (_ADR, 1)

            Name (_DSD, Package () {
                ToUUID("daffd814-6eba-4d8c-8a91-bc9bbf4aa301"),
                Package() {
                    Package () {"usb2-port-number", 5},
                    Package () {"usb3-port-number", 2},
                    Package () {"sbu-orientation", "normal"},
                    Package () {"hsl-orientation", "normal"},
                },
            })
        }
    }
}
```