**Barrel Jack / USB PD Power Mux Notes**

# Contents

[Contents 1](#_Toc175764745)

[Power Muxing 1](#_Toc175764746)

[Alternate Power Muxing Design with Back-to-back MOSFETs 3](#_Toc175764747)

[USB-PD IC Suggestions 5](#_Toc175764748)

[Resistor-Controlled USB PD ICs 6](#_Toc175764749)

[How do PDOs Work 8](#_Toc175764750)

[MacBook Pro USB PD 3.0 Offers 9](#_Toc175764751)

# Power Muxing

To design a power path circuit that prioritizes the barrel jack input over the USB-C input, you can use a combination of P-channel MOSFETs and diodes. Here’s a simplified circuit design:

**Components:**

 **P-channel MOSFETs** (e.g., IRF9540N or similar)

 **Schottky Diodes** (e.g., SS34 or similar)

 **Resistors** (for gate pull-down and pull-up)

**Circuit Design:**

1. **Barrel Jack Input**:

 Connect the positive terminal of the barrel jack to the source of a P-channel MOSFET (Q1).

 Connect the drain of Q1 to the system power rail (Vout).

 Use a resistor (R1) to pull the gate of Q1 to ground (ensuring it turns on when the barrel jack is present).

1. **USB-C Input**:

 Connect the positive terminal of the USB-C input to the source of another P-channel MOSFET (Q2).

 Connect the drain of Q2 to the system power rail (Vout).

 Use a resistor (R2) to pull the gate of Q2 to ground (ensuring it turns on when the USB-C is present).

1. **Priority Control**:

 Connect a Schottky diode (D1) between the barrel jack positive terminal and the gate of Q2 (anode to barrel jack, cathode to gate of Q2).

 This ensures that when the barrel jack is present, the gate of Q2 is pulled high, turning off Q2 and preventing the USB-C from supplying power.

1. **Reverse Current Protection**:

 Place a Schottky diode (D2) between the USB-C positive terminal and the source of Q2 (anode to USB-C, cathode to source of Q2).

 This prevents reverse current flow back into the USB-C port.

**Schematic:**

Barrel Jack (+9V) ----+----+----->|----+----+---- Vout (System Power)

| | D1 | |

| | | |

| +--->|-----+ |

| | D2 |

| | |

+----+----+ |

| | | |

(R1) | (R2) |

| | | |

| | | |

(Q1) | (Q2) |

| | | |

+----+----+ |

| | | |

GND GND GND GND

USB-C (+9V) ----+----+----->|----+----+---- Vout (System Power)

| | D2 | |

| | | |

| +--->|-----+ |

| | D1 |

| | |

+----+----+ |

| | | |

(R2) | (R1) |

| | | |

| | | |

(Q2) | (Q1) |

| | | |

+----+----+ |

| | | |

GND GND GND GND

**Explanation:**

 **Q1**: P-channel MOSFET for the barrel jack input.

 **Q2**: P-channel MOSFET for the USB-C input.

 **D1**: Schottky diode to pull the gate of Q2 high when the barrel jack is present, turning off Q2.

 **D2**: Schottky diode to prevent reverse current flow into the USB-C port.

 **R1, R2**: Resistors to pull the gates of Q1 and Q2 to ground, ensuring they turn on when their respective inputs are present.

**Operation:**

 When the barrel jack is plugged in, Q1 is turned on, and the system is powered by the barrel jack.

 The presence of the barrel jack also pulls the gate of Q2 high via D1, turning off Q2 and preventing the USB-C from supplying power.

 If the barrel jack is removed, Q1 turns off, and Q2 turns on (if the USB-C is connected), seamlessly switching the power source to the USB-C input.

This design ensures that the system prioritizes the barrel jack input and only uses the USB-C input when the barrel jack is not available, with protection against reverse current flow into the USB-C port.

# Alternate Power Muxing Design with Back-to-back MOSFETs

The body diode in a P-channel MOSFET is an intrinsic diode that is always present between the source and drain. In the context of power path switching, the body diode can conduct current in the reverse direction (from drain to source), which can lead to unintended current flow and potential issues with power isolation.

**Effects of the Body Diode**

1. **Reverse Current Flow**: If the body diode is forward-biased, it can allow current to flow from the system power rail (Vout) back to the input source, which is undesirable. This can happen if the voltage on Vout is higher than the voltage on the input source.
2. **Leakage Current**: Even when the MOSFET is turned off, the body diode can still conduct a small amount of current, leading to leakage.

**Back-to-Back MOSFETs**

Using back-to-back MOSFETs (two MOSFETs with their sources or drains connected together) can effectively block the body diode and prevent reverse current flow. This configuration is often used in more complex power path designs to ensure complete isolation. However, it adds complexity and increases the component count.

**Simplified Design with Back-to-Back MOSFETs**

To address the body diode issue, you can use back-to-back P-channel MOSFETs for each power input. Here’s an updated design:

**Components:**

1. **P-channel MOSFETs** (e.g., IRF9540N or similar)

2. **Schottky Diodes** (e.g., SS34 or similar)

3. **Resistors** (for gate pull-down and pull-up)

**Circuit Design**

1. **Barrel Jack Input**:

 Connect the positive terminal of the barrel jack to the source of the first P-channel MOSFET (Q1).

 Connect the drain of Q1 to the source of the second P-channel MOSFET (Q2).

 Connect the drain of Q2 to the system power rail (Vout).

 Use a resistor (R1) to pull the gate of Q1 to ground (ensuring it turns on when the barrel jack is present).

 Use a resistor (R2) to pull the gate of Q2 to ground.

1. **USB-C Input**:

 Connect the positive terminal of the USB-C input to the source of the first P-channel MOSFET (Q3).

 Connect the drain of Q3 to the source of the second P-channel MOSFET (Q4).

 Connect the drain of Q4 to the system power rail (Vout).

 Use a resistor (R3) to pull the gate of Q3 to ground (ensuring it turns on when the USB-C is present).

 Use a resistor (R4) to pull the gate of Q4 to ground.

1. **Priority Control**:

 Connect a Schottky diode (D1) between the barrel jack positive terminal and the gate of Q3 (anode to barrel jack, cathode to gate of Q3).

 This ensures that when the barrel jack is present, the gate of Q3 is pulled high, turning off Q3 and preventing the USB-C from supplying power.

1. **Reverse Current Protection**:

 The back-to-back MOSFETs inherently provide reverse current protection by blocking the body diode.

Barrel Jack (+9V) ----+----+----+----+---- Vout (System Power)

                      |    |    |    |

                     (R1)  |   (R2)  |

                      |    |    |    |

                     (Q1)  |   (Q2)  |

                      |    |    |    |

                      +----+----+    |

                      |    |    |    |

                     GND  GND  GND  GND

USB-C (+9V) ----+----+----+----+---- Vout (System Power)

                |    |    |    |

               (R3)  |   (R4)  |

                |    |    |    |

               (Q3)  |   (Q4)  |

                |    |    |    |

                +----+----+    |

                |    |    |    |

               GND  GND  GND  GND

D1: Schottky diode between Barrel Jack (+9V) and gate of Q3 (anode to Barrel Jack, cathode to gate of Q3)

**Explanation:**

 **Q1, Q2**: Back-to-back P-channel MOSFETs for the barrel jack input.

 **Q3, Q4**: Back-to-back P-channel MOSFETs for the USB-C input.

 **D1**: Schottky diode to pull the gate of Q3 high when the barrel jack is present, turning off Q3.

 **R1, R2, R3, R4**: Resistors to pull the gates of the MOSFETs to ground, ensuring they turn on when their respective inputs are present.

**Operation:**

 When the barrel jack is plugged in, Q1 and Q2 are turned on, and the system is powered by the barrel jack.

 The presence of the barrel jack also pulls the gate of Q3 high via D1, turning off Q3 and preventing the USB-C from supplying power.

 If the barrel jack is removed, Q1 and Q2 turn off, and Q3 and Q4 turn on (if the USB-C is connected), seamlessly switching the power source to the USB-C input.

This design ensures that the system prioritizes the barrel jack input and only uses the USB-C input when the barrel jack is not available, with effective reverse current protection using back-to-back MOSFETs.

# USB-PD IC Suggestions

## Texas Instruments TPS65987D:

 USB Type-C and PD controller

 Supports PD 3.0 and various other charging protocols

 Available from Digikey, Mouser, and other major U.S. distributors

 STMicroelectronics STUSB4500:

 USB PD controller for sink applications

 Supports PD 3.0

 Available from Digikey, Mouser, and other U.S. distributors

 Microchip UPD301C:

 USB Type-C PD controller

 Supports PD 3.0 and other charging protocols

 Available from Digikey, Mouser, and other U.S. distributors

 Cypress (now part of Infineon) CYPD3177:

 USB Type-C PD controller

 Supports PD 3.0 and various other charging protocols

 Available from major U.S. distributors

 NXP PTN5110:

 USB Type-C PD controller

 Supports PD 3.0

 Available from U.S. distributors

These parts generally offer more features and flexibility compared to the CH224K, which might be beneficial for prototyping. They typically include integrated microcontrollers or more advanced configuration options, allowing for easier customization of PD behavior.

When selecting a part, consider factors like:

 Voltage and current requirements

 Specific PD features needed

 Ease of integration with your existing design

 Available documentation and development tools

Once you've narrowed down your options, check the stock availability and lead times on Digikey or Mouser to ensure you can get the parts quickly for your prototyping needs.

# Resistor-Controlled USB PD ICs

So, I actually want a simple part where the requested voltage can be configured with physical resistors instead of requiring a uC that communicates with the chip over SPI or I2C. I just don't need that level of complexity. I'm targeting 12V @ 5A (60 watts), which I will buck convert down to microprocessor-friendly voltage levels (3.3, 1.1, 0.9, etc).

I understand your preference for simplicity and the desire to configure the requested voltage using physical resistors without the need for a microcontroller. Given your requirements for 12V @ 5A (60W), here are some simpler alternatives that might better suit your needs:

## TPS25730

* Sink-only USB Type-C® and USB Power Delivery (PD) controller with no firmware development required
* Datasheet: <https://www.ti.com/lit/gpn/tps25730>

 Texas Instruments TPS25750

 USB PD sink controller

 Configurable using resistors

 Supports up to 100W

 Available from Digikey, Mouser

~~ Richtek RT1715:~~

~~ USB PD sink controller~~

~~ Configurable using resistors~~

~~ Supports up to 100W~~

~~ Available from Digikey, Mouser~~

~~ ON Semiconductor FUSB302B:~~

~~ USB Type-C PD controller~~

~~ Can be configured with resistors for basic operation~~

~~ Supports up to 100W~~

~~ Available from Digikey, Mouser~~

 Silergy SY6982:

 USB PD sink controller

 Configurable using resistors

 Supports up to 100W

 Available from Digikey

These parts are more similar to the CH224K in terms of simplicity and resistor-based configuration. They don't require microcontroller communication for basic operation, making them suitable for your needs.

When using these parts:

 You can typically set the desired voltage (12V in your case) using specific resistor values connected to configuration pins.

 They handle the PD negotiation automatically to request the desired voltage and current.

 You can then use the 12V output with your buck converters as planned.

Of these options, the TPS25750 from Texas Instruments might be the closest match to your requirements. It's simple to use, widely available, and well-documented. However, any of these options should work well for your 60W application with minimal complexity.

Remember to check the datasheets for specific resistor values needed to configure your desired 12V output, and verify the availability and lead times on Digikey or Mouser for your chosen part.

# How do PDOs Work

Please explain how PDOs actually work. I understand they are some kind of negotiation that occurs over the CC lines. Are the CC's diff pairs or duplicates to allow cable insertion either direction? What is the binary format of a PDO? Which side goes first - the host to advertise what it supports, or the device to make a request? What happens if they have nothing in common between the offered PDOs and the requested PDOs? Does VBUS just stay at 5V/1A?

Certainly! I'll explain Power Data Objects (PDOs) and the USB Power Delivery (PD) negotiation process in detail.

1. CC Lines:

 CC1 and CC2 are not differential pairs. They are duplicate lines to allow cable insertion in either direction.

 Only one CC line is active in a connection; the other is not used for communication.

1. PDO Format:

 A PDO is a 32-bit data structure.

 For a Fixed Supply PDO (most common), the format is:

 Bits 31-30: PDO type (10 for Fixed Supply)

 Bits 29-20: Voltage in 50mV units

 Bits 19-10: Maximum Current in 10mA units

 Bits 9-0: Various capabilities flags

1. Negotiation Process:

 The power source (e.g., charger) goes first, advertising its capabilities.

 The power sink (e.g., device) then requests based on the source's capabilities.

1. Detailed Steps:

a. Initial Connection:

 VBUS starts at 5V (default USB power)

 Devices detect connection via the CC lines

b. Capability Advertisement:

 Source sends "Source\_Capabilities" message containing its PDOs

 This lists all voltages/currents the source can provide

c. Request:

 Sink reviews the PDOs and selects one

 Sink sends "Request" message specifying the chosen PDO

d. Accept and PS\_RDY:

 Source sends "Accept" message if it can fulfill the request

 Source adjusts its output

 Source sends "PS\_RDY" when the new voltage is ready

e. Power Transfer:

 Sink can now draw power at the negotiated voltage/current

1. Mismatched Capabilities:

 If the sink can't use any of the source's PDOs, it typically continues using the default 5V/1A.

 Some devices might negotiate the closest available option.

1. Renegotiation:

 The sink can request a new PDO at any time by sending a new "Request" message.

 Example PDO Binary:

For 12V, 5A capability:

 10 (Fixed Supply) + 240 (12V in 50mV units) + 500 (5A in 10mA units) + flags

 Might look like: 10 1111 0000 0111 1101 0000 0000 0000

This process allows flexible power negotiation while maintaining compatibility with standard USB power when needed.

# MacBook Pro USB PD 3.0 Offers

MacBook Pro USB-C ports typically offer several Power Delivery Options (PDOs). While the exact PDOs can vary slightly between different MacBook Pro models and years, here's a general overview of what you can expect:

* 5V @ 3A (15W)

This is the standard USB power option, available on all USB-C ports.

* 9V @ 3A (27W)
* 15V @ 3A (45W)
* 20V @ 3A (60W)
* 20V @ 4.3A (86W) or 20V @ 5A (100W)

The highest power option varies depending on the specific MacBook Pro model. Newer, higher-end models often support up to 100W.

It's worth noting that:

1. Not all ports on a MacBook Pro may support the highest power output. Some models have certain ports limited to 60W while others can output the full 96W or 100W.

 The actual power delivery capability can also depend on the power adapter connected to the MacBook Pro. If it's running on battery or connected to a lower-power adapter, it may not offer all these PDOs.

3. Apple uses USB Power Delivery 3.0, which includes support for Programmable Power Supply (PPS). This allows for more granular voltage control between 3.3V and 21V, but this is typically used when the MacBook is receiving power, not supplying it.

4. When acting as a power source, the MacBook Pro will negotiate with the connected device to determine the appropriate voltage and current based on what the device requests and what the MacBook can supply.

To see the exact PDOs available from your specific MacBook Pro, you would need to use specialized USB-C analysis tools or software that can read the Power Delivery messages exchanged over the CC lines.