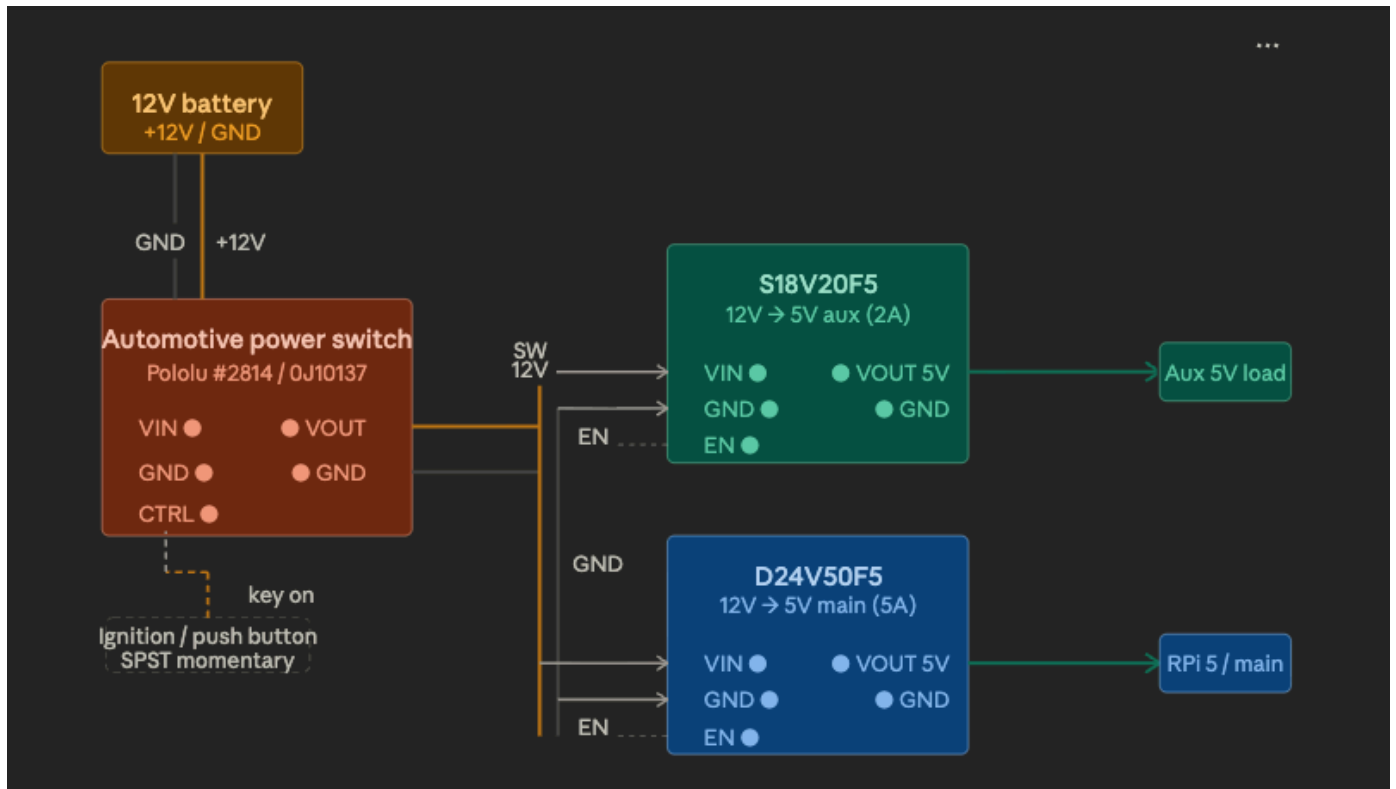


12V -> raspberry

The key facts from the datasheets:

- **Automotive Power Switch:** pins are VIN, VOUT, GND, plus a low-power CTRL input
- **S18V20F5:** pins are VIN, GND, VOUT, ENABLE (active-high, pulled up internally)
- **D24V50F5:** pins are VIN, GND, VOUT, ENABLE (same behaviour) Here's the complete wiring logic for all three boards:



Automotive Power Switch → regulators

The switch has VIN and GND on the input side (raw 12V from battery), and VOUT + GND on the output side (switched 12V). That VOUT becomes a common switched 12V bus that feeds both regulators in parallel:

- Switch VOUT → S18V20F5 VIN and → D24V50F5 VIN (same rail, tee it)
- Switch GND → S18V20F5 GND and → D24V50F5 GND (all grounds tied together back to battery)

CTRL / trigger pin on the switch

The switch has a CTRL input that activates it. In automotive use you wire this to your ignition sense line (or a panel-mounted momentary pushbutton between CTRL and GND). One press turns the switched rail on; another press turns it off. This means both regulators power up simultaneously when the vehicle key turns on.

ENABLE pins on both regulators

Both regulators have a 100 k Ω pull-up to VIN on the ENABLE pin, so they are on by default – you don't need to connect EN at all for them to run. However, because the RPi 5 is in your system you have an excellent option: wire the EN pins to RPi GPIO outputs (via a 10k Ω series resistor as protection). This lets the Pi sequentially disable peripherals before shutdown, or independently cut the S18V20F5's aux rail while keeping the main Pi 5V alive.

A practical note on wire gauges: The D24V50F5 can deliver 5A continuously. At 12V input that's ~2.5A draw on the input side – use at minimum 20 AWG for those traces. The S18V20F5 is 2A max output, so 24 AWG is fine on its side. The automotive switch's VOUT carries the sum of both, so size that trace for your total load plus margin (18 AWG minimum for this application).