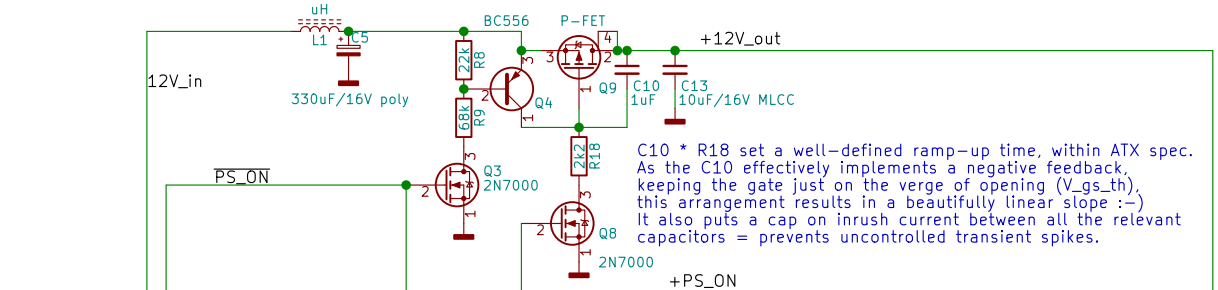
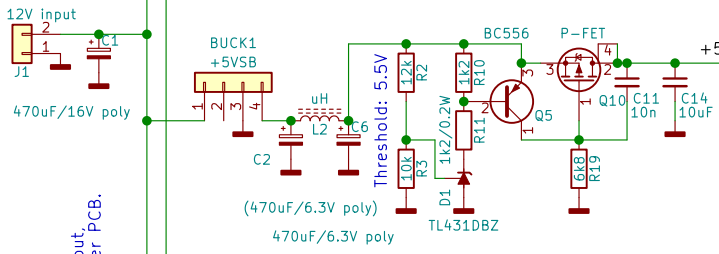
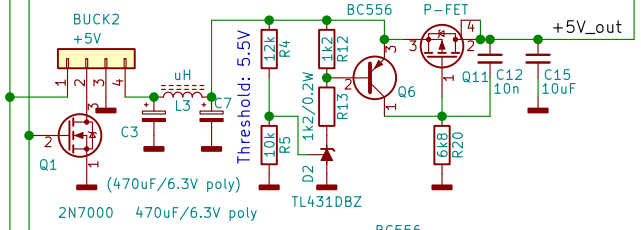
This is an ATX-style multiple buck converter, intended for use in an ATOM-based HTPC, powered by an internal or external 12V mains PSU.

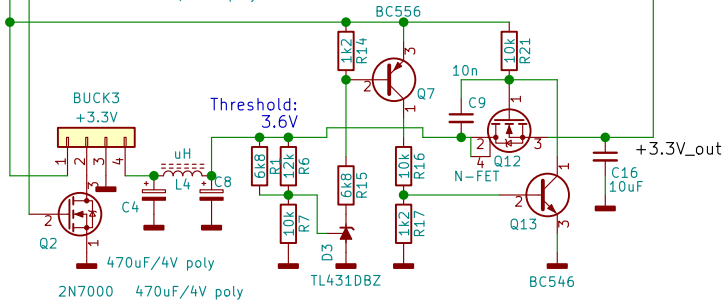
The 12V rail is just filtered and passed through. A power P-FET is used to switch the rail by PS_ON.



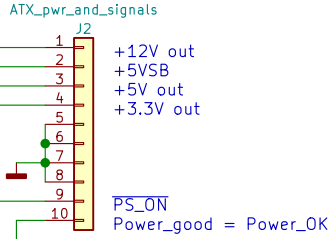
C10 * R18 set a well-defined ramp-up time, within ATX spec.
As the C10 effectively implements a negative feedback,
keeping the gate just on the verge of opening (V_gs_th),
this arrangement results in a beautifully linear slope :-)
It also puts a cap on inrush current between all the relevant
capacitors = prevents uncontrolled transient spikes.

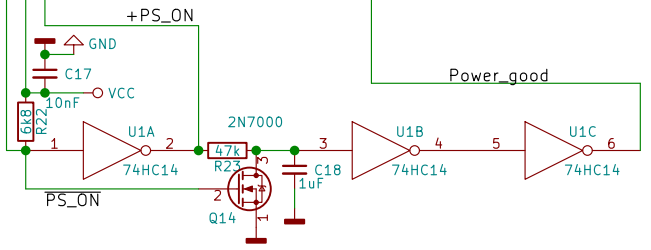






If the buck converter contains an elyt on the output,
there's no need to have another one on the carrier PCB.
See C2, C3, C4.





The Power_good output
is actually just timer-based.
There is no checking if the outputs
are really within limits,
by the time Power_OK goes high.
The fixed time constant is chosen such
that all the buck sections should be
safely up and running by then.
47k * 1uF = cca 50 ms

As I don't trust the buck converters very much, I have implemented "over-voltage disconnectors" on the buck-regulated rails.
In the 5V rails, both the TL431 and the P-FET can work with that rail voltage.
In the 3.3V rail, the TL431's minimum "Vce" doesn't leave much room for the forward voltage of the follow-up PNP BJT's base,
and also 3.3V is not enough to fully open a P-FET. Hence the more complex setup, making use of the 12V rail (and an N-FET).

Frank Rysanek DIY	
Sheet: /	
File: atx_3x_buck.sch	
Title: ATX triple buck	
Size: A4	Date:
KiCad E.D.A. kicad 4.0.7	Rev: Id: 1/1