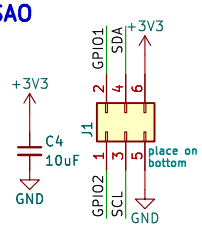
Blinkencap

The world's most difficult to manufacture blinkenlight

This simple add-on for the Hackaday Supercon badge uses the massive overkill that is an ASIC (a custom integrated circuit) with over 150 individual projects, to control a single RGB LED, and comes in a snazzy ASIC cap form factor. For us plebes without ASICs, LEDs are also controllable through header and autoblink.

SAO



Simple Add-on (SAO) header.

Traditionally uses a through-hole, 2x3 100mil male header to mate with a polarized (keyed) female on badges.

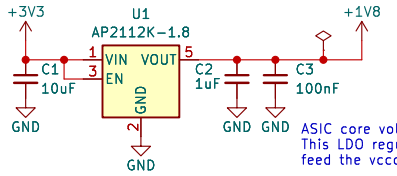
In order to avoid putting holes in the cap, an SMD component is used here. Hopefully I didn't mess up the connector geometry, yet again.

Specs: Simple Add-on v2 (link)

GPIO1	NT5	led_red	GPIO1	NT1	p_clk
GPIO2	NT6	led_green	GPIO2	NT2	ctrl_rst
SCL	NT7	led_blue	SCL	NT3	ctrl_inc
			SDA	NT4	ctrl_ena

To avoid spilling these net names everywhere and have suitably labelled connections, net ties are used here.
KiCad net tie footprints are butt ugly, I should take 5 minutes and do some cleaner implementation. It's on the stack.

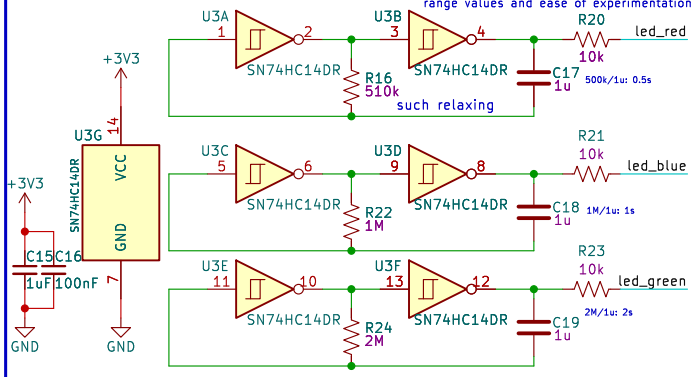
1.8V LDO



ASIC core voltage is 1v8. This LDO regulates down to feed the vccd* on the ASIC.

Auto-blink Relaxation Oscillators

Used bigger components for passives here to allow for wider selection in high range values and ease of experimentation



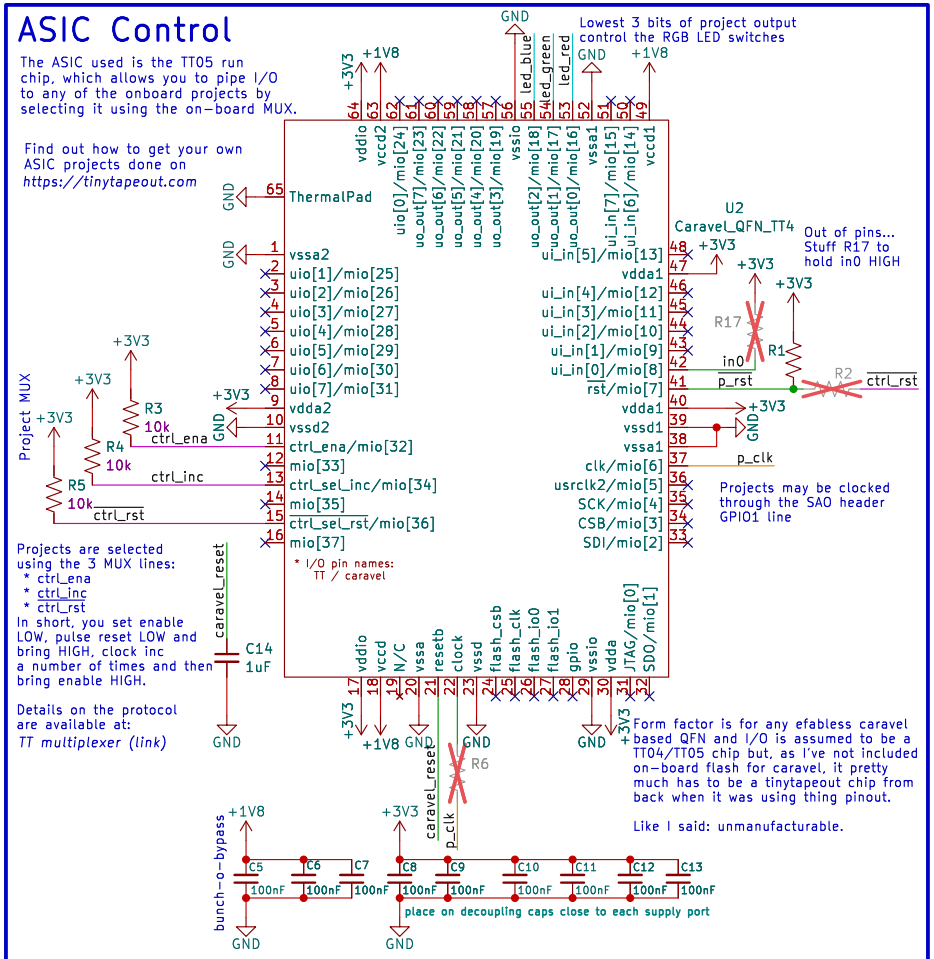
The schmitt-triggered hex inverter is setup as 3 independent relaxation oscillators, which will transition between LOW and HIGH at a rate determined by the R and C in the feedback loop.
Each of the three then controls the switch for one of the LED colours. This is a weak signal thanks to the series resistance on the output, so you can override it from the ASIC or the SAO header.
Selecting *different* resistances for each oscillator will allow for distinct colour patterns, as various combinations are illuminated simultaneously.
When using a 1u cap, the time constant (tau) for the combination is simply the number of R megs seconds, easypeasy.

ASIC Control

The ASIC used is the TT05 run chip, which allows you to pipe I/O to any of the onboard projects by selecting it using the on-board MUX.

Find out how to get your own ASIC projects done on *https://tinytapeout.com*

Lowest 3 bits of project output control the RGB LED switches



Out of pins... Stuff R17 to hold in0 HIGH

Projects may be clocked through the SAO header GPIO1 line

Projects are selected using the 3 MUX lines:
* ctrl_ena
* ctrl_inc
* ctrl_rst

In short, you set enable LOW, pulse reset LOW and bring HIGH, clock inc a number of times and then bring enable HIGH.

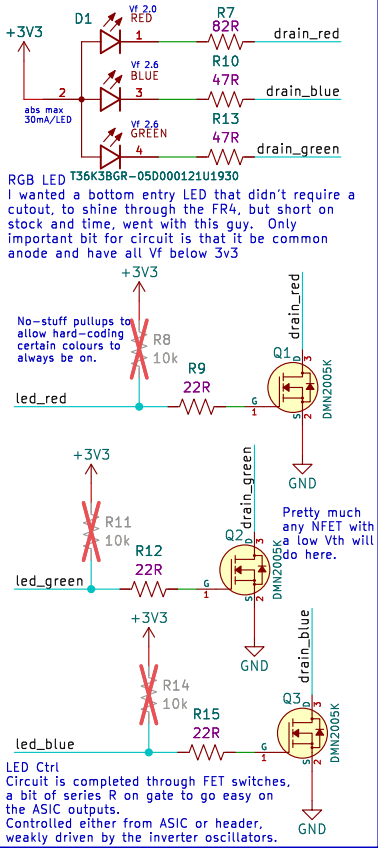
Details on the protocol are available at:
TT multiplexer (link)

* I/O pin names: TT / caravel

Form factor is for any efabless caravel based QFN and I/O is assumed to be a TT04/TT05 chip but, as I've not included on-board flash for caravel, it pretty much has to be a tinytapeout chip from back when it was using thing pinout.

Like I said: unmanufacturable.

The Blinkenlight



RGB LED

I wanted a bottom entry LED that didn't require a cutout, to shine through the FR4, but short on stock and time, went with this guy. Only important bit for circuit is that it be common anode and have all Vf below 3v3

Pretty much any NFET with a low Vth will do here.

LED Ctrl
Circuit is completed through FET switches, a bit of series R on gate to go easy on the ASIC outputs.
Controlled either from ASIC or header, weakly driven by the inverter oscillators.

My inspiration:
Matt's famous cap

FID1 Fiducial
FID2 Fiducial
FID3 Fiducial

Psychogenic Technologies

Sheet: /
File: blinkencap.kicad_sch

Title: Blinkencap Simple Add-On

Size: A4	Date: 2024-10-09	**Rev: 1.0**
KiCad E.D.A. 8.0.5		Id: 1/1