



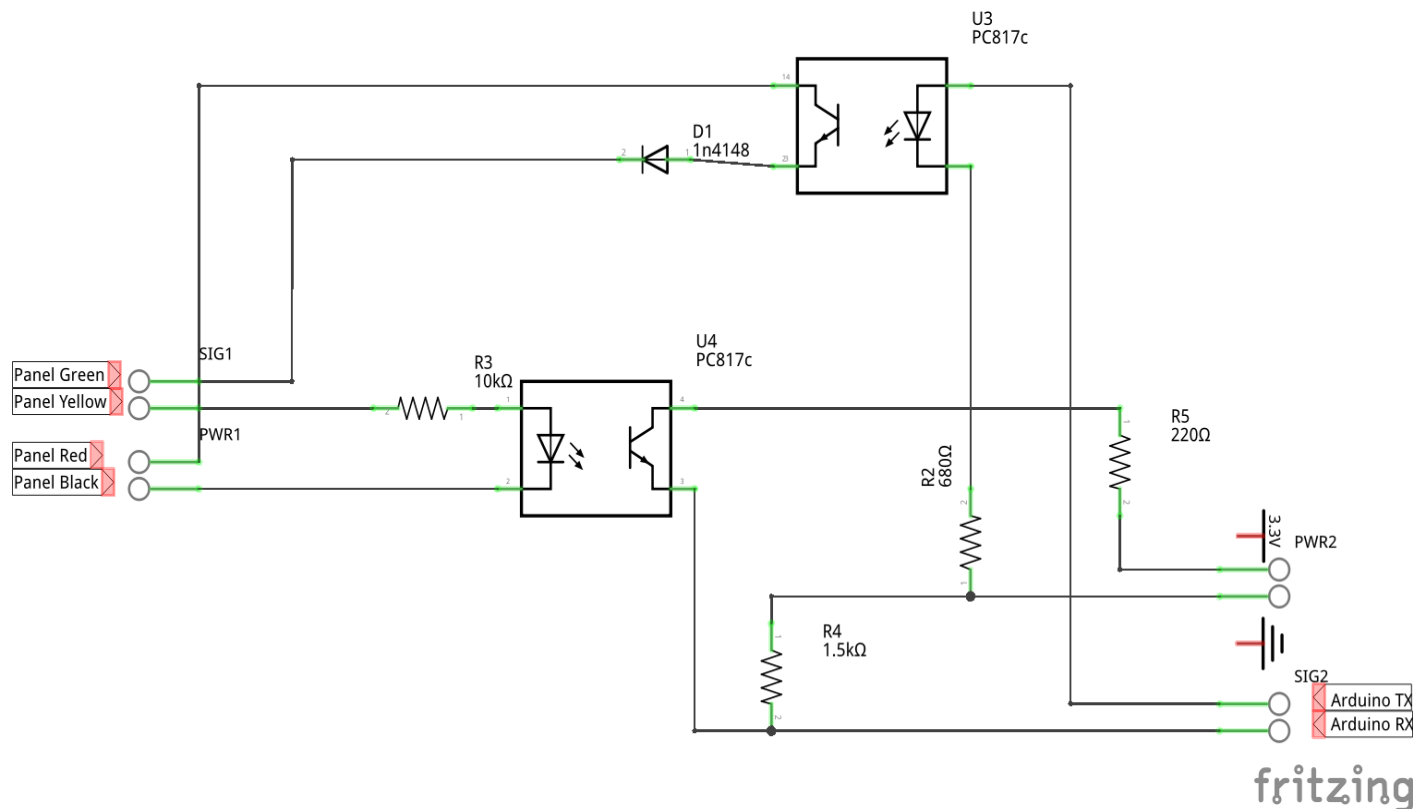
GET A TANC

# Hardware Documentation

A TANC is built on 3 components:

- Arduino Keypad Bus Circuit
- Raspberry Pi Notification Center
- Power Supply Switching Regulator

## Arduino Keypad Bus Circuit



The circuit is composed mainly of 2 4N25 optocouplers and a number of current limiting resistors. You can use male pin headers or screw terminals to make the interface needed at SIG1, PWR1, SIG2, and PWR2.

Note that the two optocouplers are *mirror images* of each other. Don't connect one backwards.

The Vista 20 security panel uses 13v lines for power and communication.

To supply a constant 3.3v signal to the collector of U4, you can use the 3.3v pin on the Arduino OR you can use another pin like A3 and set it to OUTPUT mode and digitalWrite HIGH in the setup() routine of the sketch.

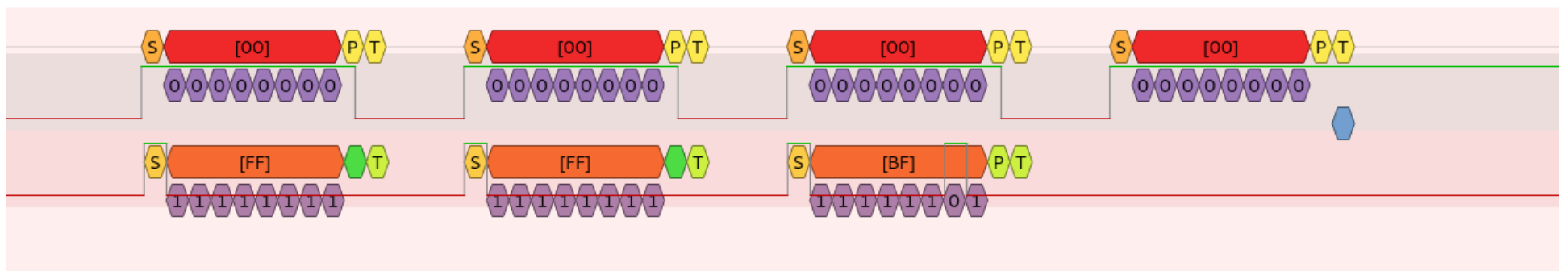
## 5v / 16Mhz Arduinos

This circuit should also work with 5v Arduino chips, but the Raspberry Pi's GPIO pins operate at 3.3v. You will need a level shifter if you want the Arduino to talk to the RPi's serial connection.

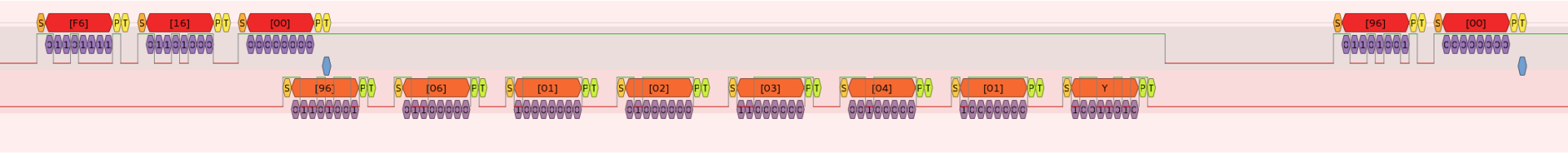
# Protocol Documentation

The ECP is a slightly modified UART format to fit on just 2 data wires. The packets are 8 bit, 1 stop, and even parity @4800 baud. Not all packets use parity, but the ones that don't use a parity bit will wait for one baud, essentially using 2 stop bits.

In order to send information, keypads must send their address to the server during regular polling intervals and only send data when the server has sent a CTS packet (0xF6). This server-led orchestration allows a star configuration with multiple keypads and a central panel all sharing 2 data lines.



The above shows a packet capture in PulseView of the panel polling all keypads for those that have data to send and the synchronised response from keypad 22.



The above shows a package capture in PulseView of the panel sending a CTS for keypad 22. Then keypad 22 sends 5 key presses and a checksum. The panel acknowledges that it got message 0x96.