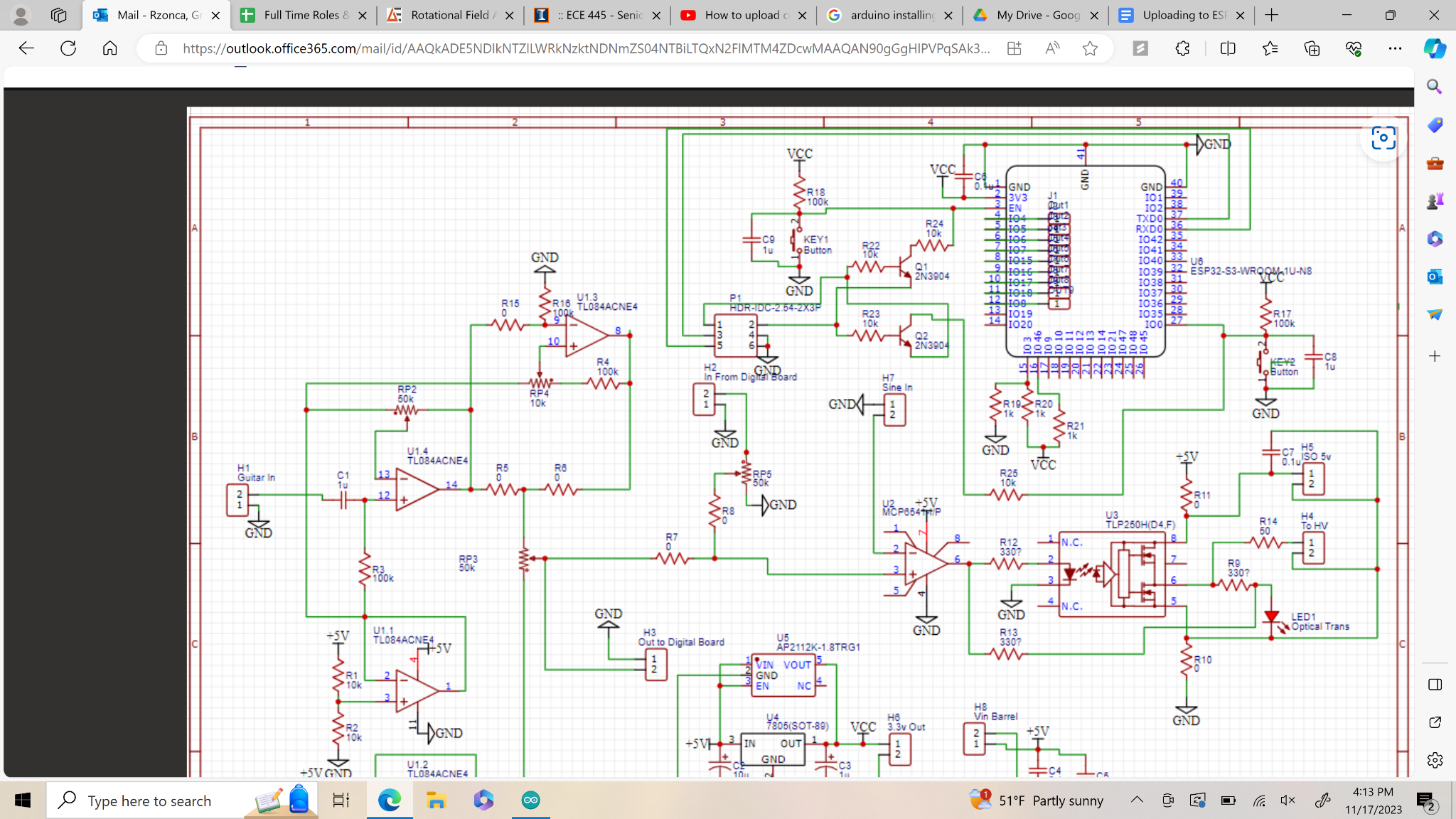
11/17/23

Griffin Rzonca

Uploading code to ESP32 on Board

* Got the code working last time using the dev kit, now need to upload using programming circuit onto our actual ESP32
* Code was tested and works, just work on uploading to our PCB today
* Using USB/UART bridge, which can upload from Arduino IDE with no problems



P1:

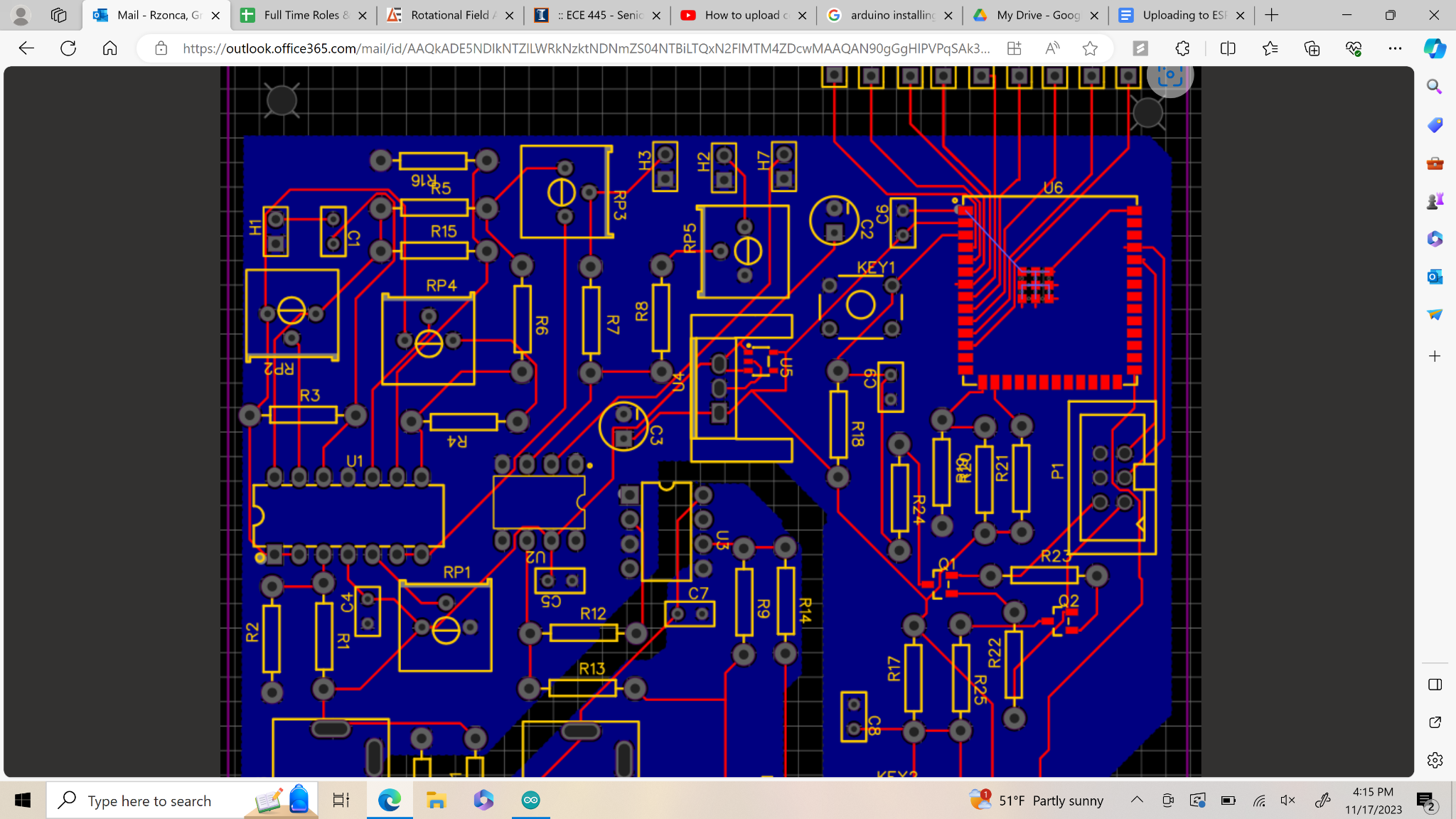
1: RTS

2: DTR

3: TX on ESP32

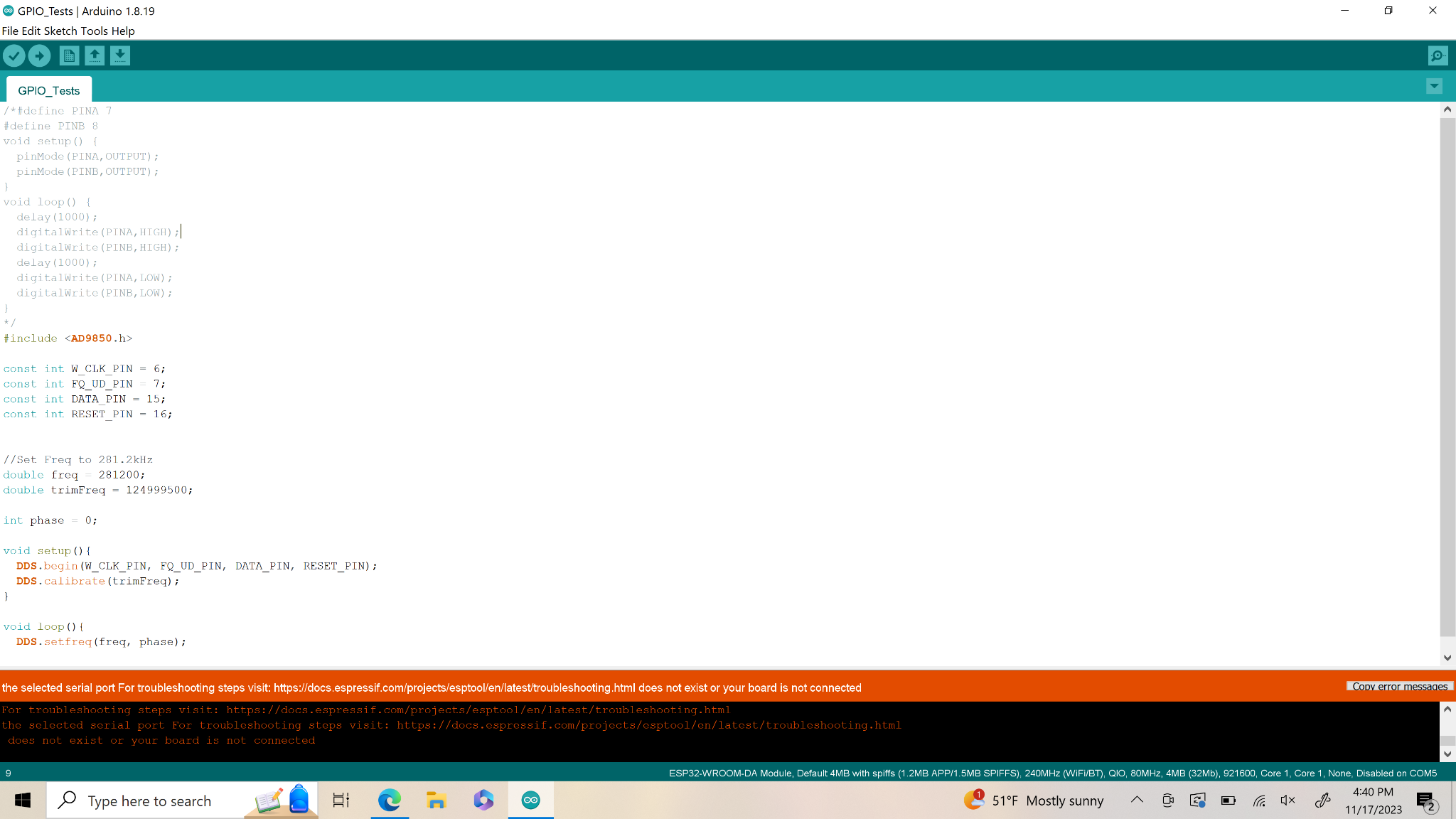
5: RX on ESP32

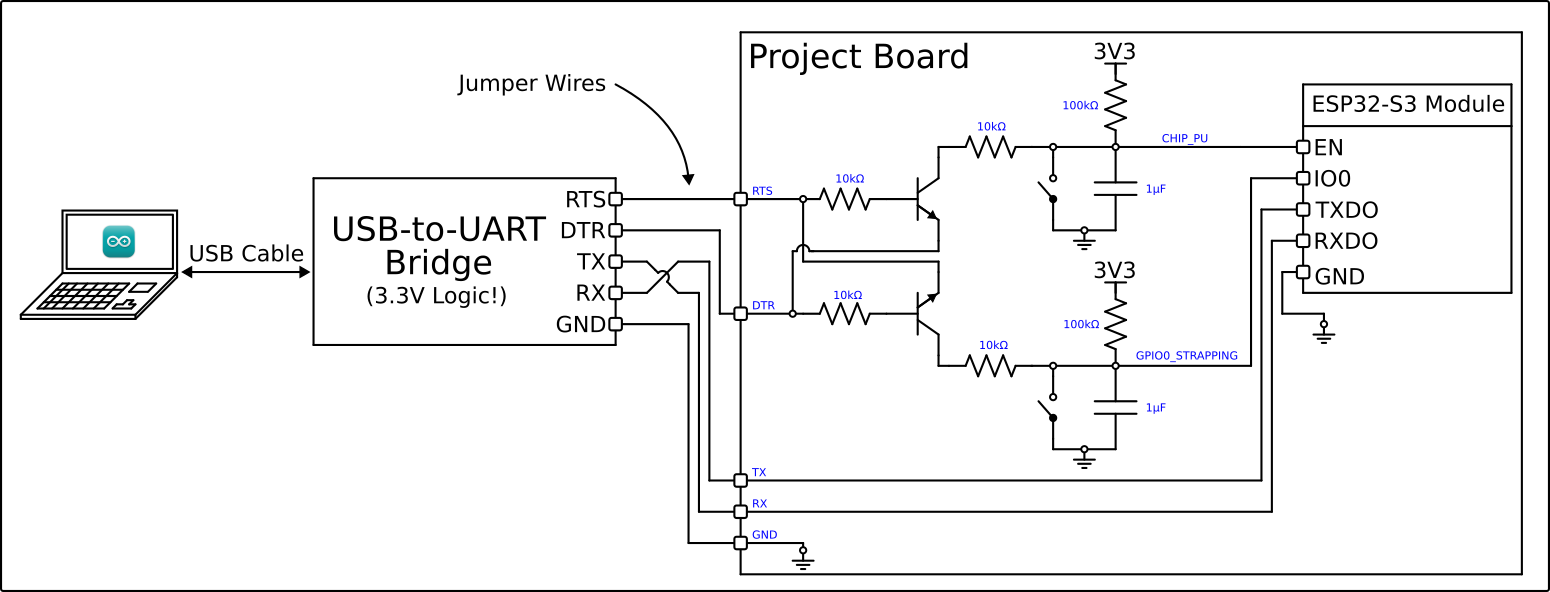
4,6: GND



Pinout:

| 6 | 5 - RX |
| --- | --- |
| 4 | 3 - TX |
| 2 - DTR | 1 - RTS |

* Wired up on breadboard and connected to PCB
* For some reason, the upload won’t work
* Certainly get outputs from USB/UART bridge, verified by oscilloscope
* Q1 and Q2 voltage values also change when uploading, also verified from scope
* Didn’t solder in the buttons, perhaps I needed to and it wasn’t in boot mode?
* Had the same error last time with the Dev Kit, even the same weird text formatting
* Says “Connecting…..” when I upload code, which is different from when I left the pins open and tried to upload
* Next, check pins on board itself receive data



Test Voltage Waveform:

EN: Yes

IO0: Yes

TXD0: No (Constant 3.3V)

RXD0: Yes

Gnd: Yes

* Is TXD0 supposed to be constant? I think it should, since the ESP32 probably doesn’t transmit any data when code is being uploaded to it, though I’m not sure
* Maybe the ESP32 needs to be reset or to be in a state where it can receive uploads
* Though, maybe it should be transmitting, so that the USB/UART bridge knows it’s uploading to something, and because it isn’t transmitting, that’s why it can’t connect! That’s probably it
* What does the ESP32 need to transmit? It has power already
* ESP32 is certainly receiving signals, it just won’t transmit any back to tell the USB/UART bridge any information
* Probed at ESP32 terminal, so it’s not like it’s getting lost along the way by a bad connection, it isn’t being sent from the ESP32 at all
* Been 5 or 6 hours, try again tomorrow
* Grounds may be different on USB/UART bridge vs PCB!
* For some reason, the ground on the ESP32 module is 2V higher than the ground on the bridge/the ground on pins 4 and 6
* According to the probes, the voltage from the ESP32’s ground to 3.3V input is still 3.3V, which should be okay
* And, the voltage from pinout ground to ESP32’s 3.3V is 3.3V
* But somehow the voltage between the two grounds is 2V
* How? Could this mean that even though the input waveforms looked okay
* Tried connecting grounds and still didn’t work, try again tomorrow, look at grounds mismatch
* Have some good info, call it for today since it’s been so long, try again tomorrow