2/19/2025

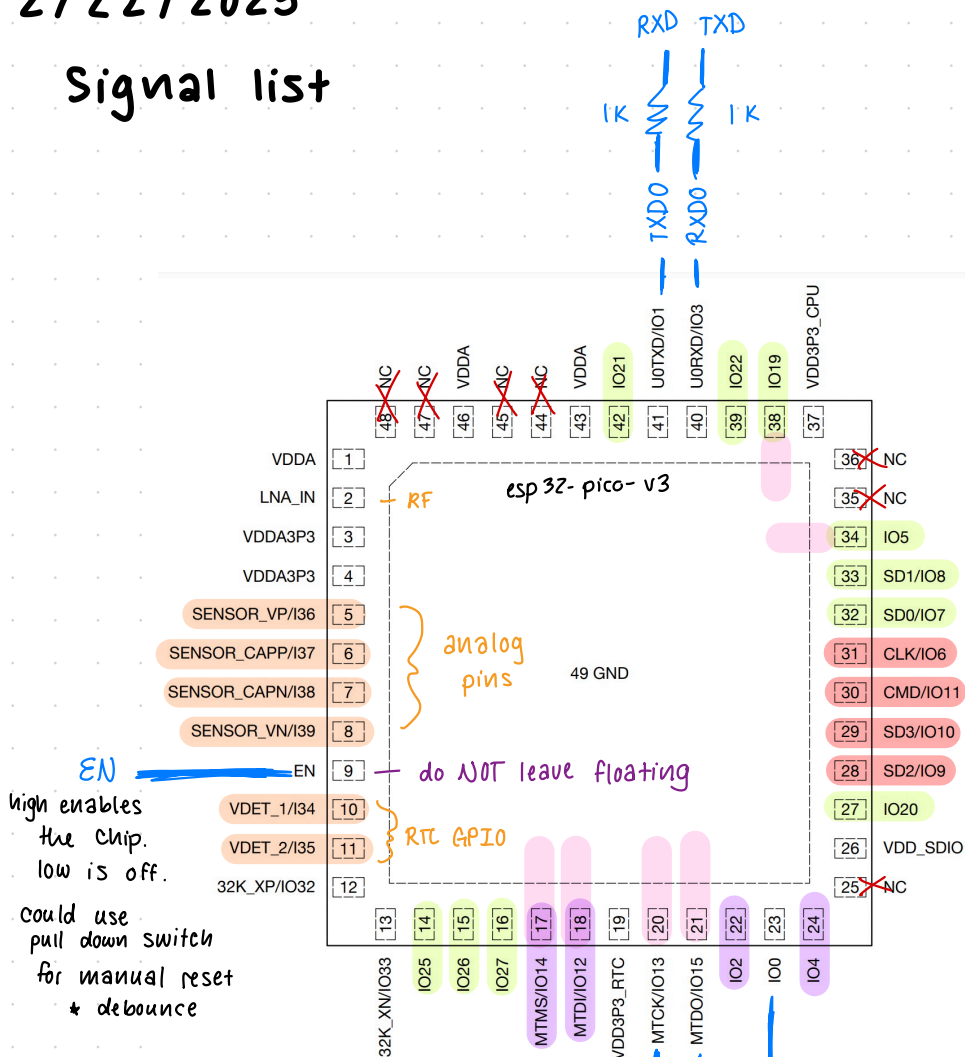## Programmer

VS Code w/ PlatformIO extension
  allows use of Arduino platform
  can use ESP-IDF w/o changing build platformer
  well documented

2/22/2025

Signal list

RXD TXD

1K   1K

TXD0 RXD0

esp 32- pico- v3

48 NC | 47 NC | 46 VDDA | 45 NC | 44 NC | 43 VDDA | 42 IO21 | 41 U0TXD/IO1 | 40 U0RXD/IO3 | 39 IO22 | 38 IO19 | 37 VDD3P3_CPU

1 VDDA
2 LNA_IN — RF
3 VDDA3P3
4 VDDA3P3
5 SENSOR_VP/I36
6 SENSOR_CAPP/I37 } analog pins
7 SENSOR_CAPN/I38
8 SENSOR_VN/I39
9 EN — do NOT leave floating
10 VDET_1/I34
11 VDET_2/I35 } RTC GPIO
12 32K_XP/IO32

49 GND

36 NC
35 NC
34 IO5
33 SD1/IO8
32 SD0/IO7
31 CLK/IO6
30 CMD/IO11
29 SD3/IO10
28 SD2/IO9
27 IO20
26 VDD_SDIO
25 NC

13 32K_XN/IO33 | 14 IO25 | 15 IO26 | 16 IO27 | 17 MTMS/IO14 | 18 MTDI/IO12 | 19 VDD3P3_RTC | 20 MTCK/IO13 | 21 MTDO/IO15 | 22 IO2 | 23 IO0 | 24 IO4

EN
high enables the chip.
low is off.

could use pull down switch for manual reset
* debounce

IO13 IO15

0Ω   0Ω

RTS CTS
↑   ↑
programmer can send   programmer can receive

IO0
↳ low enters program mode when en is high. Could also tie this to a button or RTS

* programs when DTR = 0, RTS = 1
=> EN = 1   IO0 = 0

input-only GPIO

JTAG: used to test PCB (shorts or opens)

ADC: analog to digital conversion channel
DAC: digital to analog conversion channel

SPI, HSPI, VSPI: signals of SPI modules

U0, U1, U2: signals of UART modules

NC: not connected

SD, HS1, HS2: SDIO

On programmer:
  CTS: clear to send
    telling chip it's ready to receive
  RTS: request to send
    asking chip to send data

Notes:
  rotary encoder outputs digital

DTR: data terminal ready
  tells chip its ready to send or receive data

For programming,

can use 4 pin header   or   6 pin header
          (excluding Vcc, Gnd)   (including Vcc, Gnd)

programmer	between	esp
TXD	1kΩ	RXD0 (40)
RXD	1kΩ	TXD0 (41)
RTS		IO13 (20)
CTS		IO15 (21)

programmer	board
3.3/5 V	ext. VCC
GND	ext. GND

use buttons to enable programming
    hold IO0, press EN button, then release boot