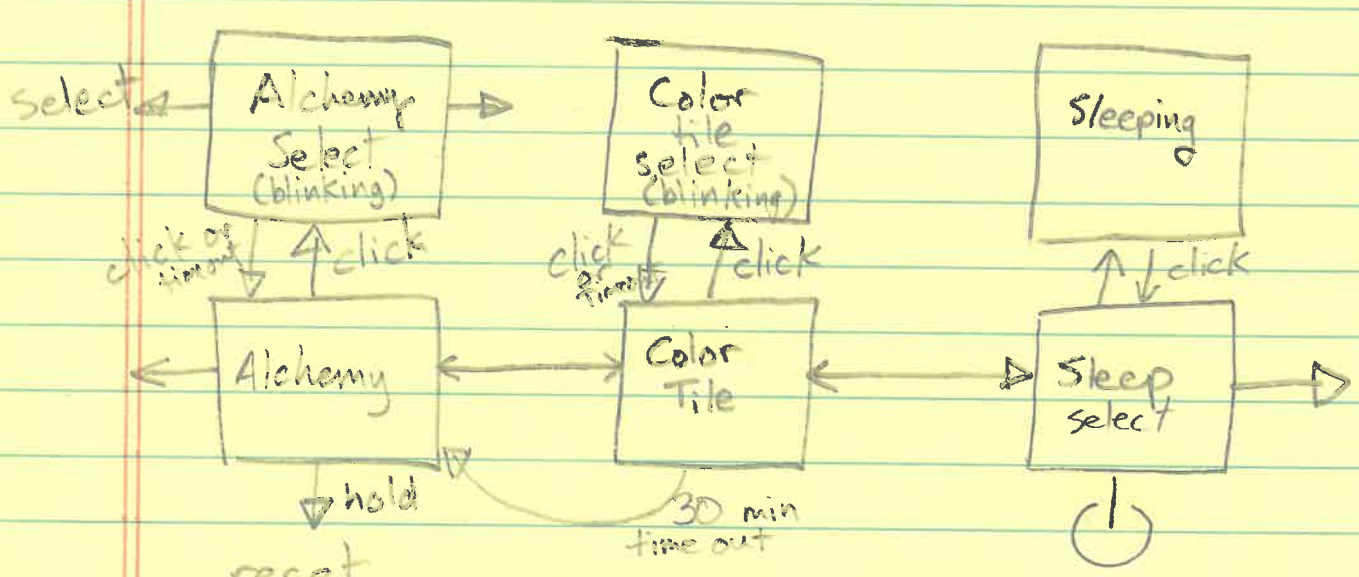
TIME VIRUS

Color tile is default mode during parties & Alchemy, otherwise.



Alchemy mode:

Same as "pairing mode" from spec

Color tile mode:

Solo: Displays the usual flag-based colors (detectable)

Connected: Users can build synchronized fabrics of colors

Sleep Mode:

Power save. LEDs are off or gently pulsing.

Basic Color Tile functionality:
Connections are always ordered

Selectable bling (with timeout)
spinning arms while disconnected

fabric_priority = lowest badge in fabric
fabric.root = starting badge in fabric

Bootstrapping a fabric:
(handshake)
2 connect & decide orientation & id.
root to leaf is horizontal +

New tile added to fabric:
(handshake first)
fabric to tile: root, priority, direction, frame_id
tile to fabric: ACK

Two fabrics connected:

if root_id same, do nothing.
Lower

Tile knows if it's at the "edge"?

SWIs

sw_clock_f (p=100)

hlc_spi_cb (x1 callback)

Tasks

ICallRemote (5)

GAPRole (3)

SimpleBLEBroadcaster (1)

led_brightness_task_fn (1)

HWIs

mp_tick

Handshake:

id time virus mode fabric_id fabric_pri seqnum?

Message types:

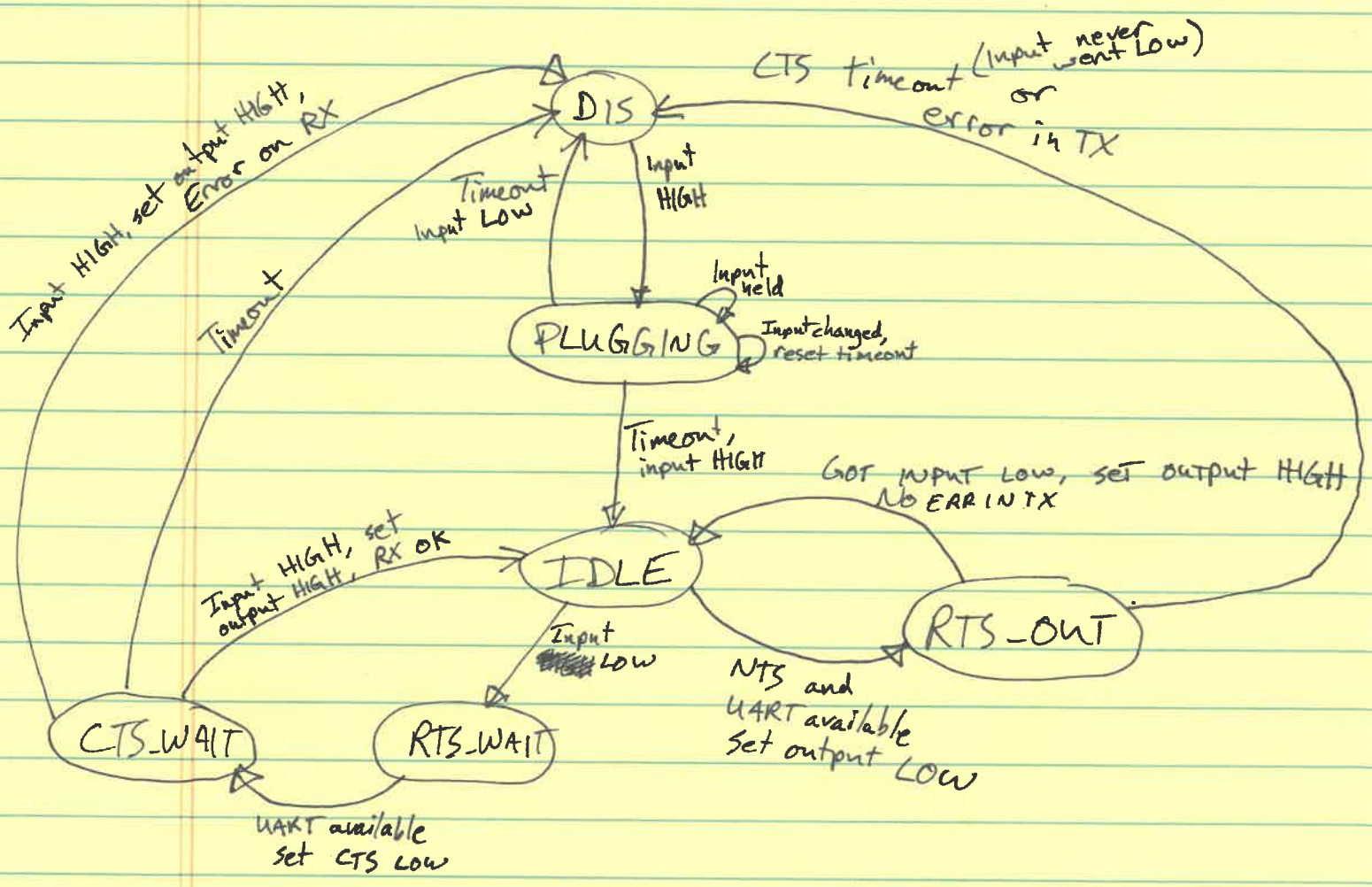
Handshake

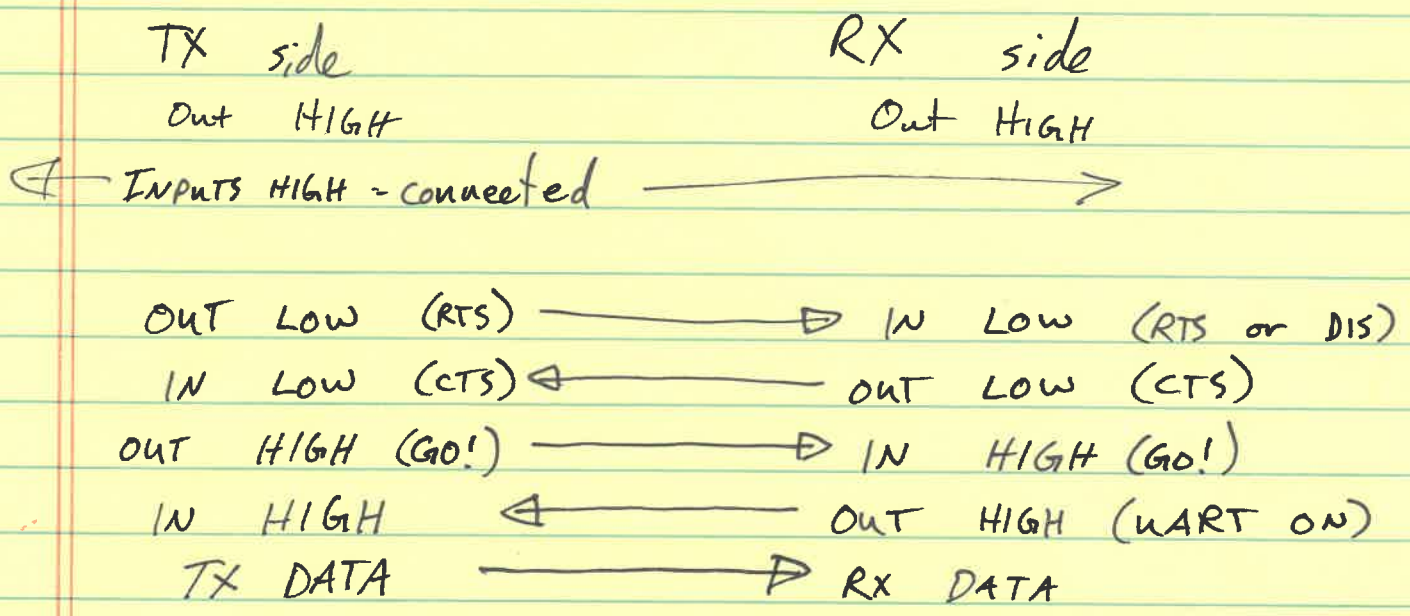
Topology update

EVERYTHING IS INVERTED!

THE UARTS ARE IDLE HIGH, SO THE FLOW CONTROL ALSO NEEDS TO BE IDLE HIGH.

SO: INPUTS NEED PULL DOWNS AND ALL LOGIC IS TO BE INVERTED:



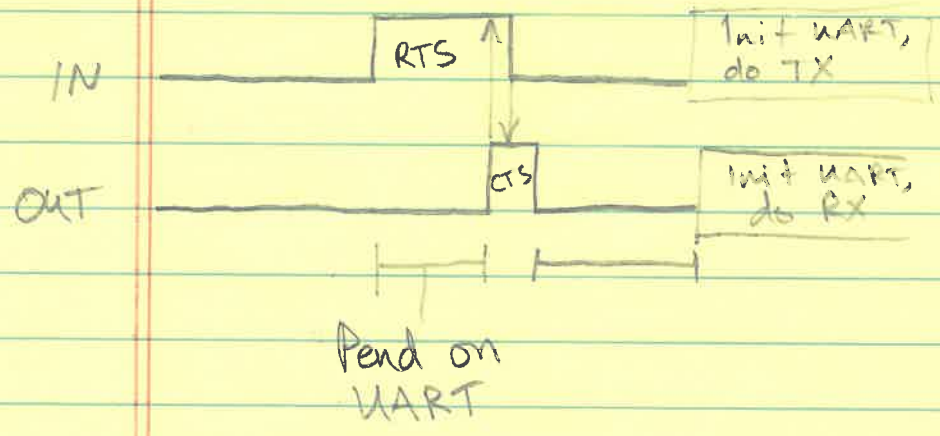

TX side
RX side
Out HIGH
Out HIGH
INPUTS HIGH - connected
OUT LOW (RTS)
IN LOW (RTS or DIS)
IN LOW (CTS)
OUT LOW (CTS)
OUT HIGH (GO!)
IN HIGH (GO!)
IN HIGH
OUT HIGH (UART ON)
TX DATA
RX DATA

Idle: IN: pull up
OUT: low

Connect: Detect IN pulled low

IN goes high: Either UNPLUGGED or RTS

OUT high for CTS. If IN goes low, HANDSHAKE
If IN stays high, DISCONNECT



→ Can tie HOLD to Vcc?

Flash chip:

Keep HOLD high
Keep WP low,
except when writing SR
(status register)

SCK	clock
SI	MOSI
SO	MISO
$\overline{CS}$	Enable
$\overline{WP}$	Write protect
$\overline{HOLD}$	

Architecture: 256 byte pages
4 kB sectors
64 kB blocks

Addresses are 24 bits.

To read: (not possible if WIP=1)
write 0x03 (READ)
write Addr, MSB first
Read desired number of bytes (addr will inc)

To write: (One page at a time, must be erased)
check WIP
write 0x06 (WREN)
check status to confirm WREN
write 0x02 (PP)
write address, MSB first
write desired number of bytes (addr will inc)
(order is MSbit first, lowest addr first)
wait for !WIP
WEL is automatically cleared.

Status Register (SR)

MSB							LSB
R7	R6	R5	R4	R3	R2	R1	R0
SRP	reserved	BP3	BP2	BP1	BP0	WEL	WIP

SRP: 0: $\overline{WP}$ pin does nothing | 1: $\overline{WP}$ pin works
BP: Block protect: 0b1111 = all; 0b0000 = none; see DS.
WEL: Read-only. 1 = write enabled; 0 = read-only
WIP: Read-only. "Write In Progress." 1 = busy; 0 = OK

Read SR:

write 0x05
read one byte (SR)

SR output ch	Scan line
1	13
2	8
3	5
4	15
5	16
6	6
7	7
8	14
2.1	10
.2	4
.3	12
.4	2
5	1
6	11
7	3
8	9