

6

5

4

3

2

1

nes-bench

Original pad to a USB or Bluetooth keyboard, pad-ble v1

A

Original pad to a USB or Bluetooth keyboard, pad-ble v1

One original NES controller into an ESP32, presented to a phone or a laptop as a plain keyboard: the bridge's own pad poll with a host interface behind it. No Pi, no UNO, no console. Sheets 2 to 4 are the C6 design, kept as a record; that board died 2026-09-29.

TWO BUILDS, ONE CIRCUIT. USB on a Waveshare ESP32-P4-Module-DEV-KIT (sheets 5 to 10), proven 2026-09-27. BLE on a classic ESP32-WROOM-32 devkit (sheets 11 to 15, 2026-09-30) on D25, D26, D27, since GPIO6 is its flash and GPIO3 its console; it advertises and the bench head bonded with it. No pad wired to it yet; no phone tried on either.

Sheets

SHEET	TITLE	PAPER	HOW IT IS MADE
1	Cover and sheet index	letter	this sheet
2	pad-ble v1: one pad into the C6, as a BLE keyboard	ANSI B	schematic
3	Wiring diagram: every wire at right angles on the packages	ANSI B	schematic
4	Breadboard: which hole each part and jumper goes in	ANSI B	schematic
5	pad-ble v1 on the ESP32-P4: one pad into header P6	ANSI B	schematic
6	Wiring, ESP32-P4: every wire at right angles, P6's own pins	ANSI B	schematic
7	Breadboard, ESP32-P4: the board sits beside it, not on it	ANSI B	schematic
8	ESP32-P4 module header P6: where each wire goes	letter	schematic, lab log, bring-up tool
9	ESP32-P4 module header P6: where each wire goes (continued)	letter	schematic, lab log, bring-up tool
10	ESP32-P4 module header P6: where each wire goes (continued)	letter	schematic, lab log, bring-up tool
11	pad-ble v1 on a classic ESP32: one pad into the devkit	ANSI B	schematic
12	Wiring, classic ESP32: every wire at right angles	ANSI B	schematic
13	Breadboard, classic ESP32: on the board, rows 6 to 20	ANSI B	schematic
14	WROOM-32 header: where each wire goes	letter	schematic, lab log, bring-up tool
15	WROOM-32 header: where each wire goes (continued)	letter	schematic, lab log, bring-up tool
16	Wiring list, pad-ble v1	letter	derived from the schematic
17	Parts list	letter	derived from the schematic

A

B

B

C

C

D

REV	DATE	REVISION	tinymachines				nes-bench	
J	2026-09-26	the first pad was a replica; an original's lead ...	SHEET TITLE					Cover and sheet index
K	2026-09-26	J1 drawn as an original pad in its own colours o...	DRAWING NO.		REV	DATE	SHEET	
L	2026-09-28	measure-first item 4 closed and the host result ...	TM-NESB-003		M	2026-09-30	1 of 17	
M	2026-09-30	the BLE build moves to a classic ESP32-WROOM-32 ...	DRAWN BY		SOURCE		SCALE	
			bisenbek		git 5a52db9		NTS	

D

6

5

4

3

2

1

87654321

nes-bench

pad-ble v1: one pad into the C6, as a BLE keyboard

The buildable subset of pad-adapter.svg: USB powered, no cell, no charger, no mode switch. The board on the bench is an ESP32-C6-DevKitC-1 v1.2, read off its RGB@IO8 silkscreen 2026-09-21. Nothing built.

THE PAD, polled at 3V3 exactly as the bridge polls one

J1

original pad, on the bridge

1

GND

2

CLK

3

OUT0

4

D0

5

+5V

6

n/c

7

n/c

the pad half of the cut cable

The pad's 4021 is a CMOS part rated 3 to 18 V, so at 3V3 its D0 is 3V3 logic and needs no shifter. That is the datasheet, not this bench: MEASURE FIRST, item 4. If a pad will not follow its buttons at 3V3, add a 74LVC245 and feed the pad 5 V; the 245 is on hand.

ONE PAD, and that is the design rather than a first step. A BLE keyboard report carries six key slots and two pads can ask for ten, so a second pad could be polled and never sent. One was on this sheet until 2026-09-23 doing exactly that: wired, pulled up, given a pin, and thrown away. Two players wants gamepad mode with two report IDs, which is pad-adapter.svg's job.

Pin 5 is the supply and 6 and 7 carry nothing, as measured on this console's own cable (docs/wiring.md). RING THE CABLE OUT UNPLUGGED: a tone through a cable still in the console runs through its pull-ups, and pins that share nothing beep.

PAD1_D0

R1

10k

3V3

THE CONTROLLER, on USB power

U1

ESP32-C6-DevKitC-1 v1.2

PAD_LATCH

GPI02

PAD_CLK

GPI03

PAD1_D0

GPI06

X

GPI010

X

GPI011

3V3

3V3

GND

GND

BLE

radio

USB-C

VBUS

NimBLE HID keyboard; no USB device controller

3V3

C1

100nF

NOT GPIO4, GPIO5 or GPIO15: strapping pins on the C6. The three pad pins above are the map firmware/bridge/bridge.ino already polls a pad on, so firmware/pad-ble reuses poll_pad unedited.

GPIO10 (MODE_SW) and GPIO11 (LED) are NC here: the firmware reads one with a pullup and drives the other, and neither is wired. Declared rather than omitted, so the netlist sees a statement instead of silence.

C1 sits across the devkit's own 3V3 and GND pins. The devkit is regulated already; this is for the breadboard's inductance, which is what the bridge's own 100 nF was added for on 2026-09-15.

WHAT THIS BOARD CAN AND CANNOT DO, and why the answer is in silicon

BLE HID

Works, and reaches a phone with no app: a BLE HID-over-GATT keyboard pairs in Settings beside a Magic Keyboard. Browser emulators take keys. Needs four things, each in firmware/pad-ble/pad-ble.ino with its reason: bonding with encryption, the PnP ID, a battery service, and the HID service UUID plus the keyboard appearance IN THE ADVERTISEMENT, since a host decides what to offer as a pairable accessory from the advert alone.

USB HID

IMPOSSIBLE ON THIS PART, and not for want of firmware. The port marked USB is a USB-Serial-JTAG bridge, fixed in silicon. Its soc_caps.h defines SOC_USB_SERIAL_JTAG_SUPPORTED and does NOT define SOC_USB_OTG_SUPPORTED, and the core gates USBHIDKeyboard.h on the latter, so it compiles away. Read off the installed esp32 core 3.3.11 on 2026-09-21, not recalled. USB HID needs an S3, S2 or P4, and an S3 does BLE too, so it gets both modes.

Gamepad HID

Not built. iOS refuses a generic HID gamepad, taking only the MFi, Xbox, PlayStation and Switch Pro layouts, which is why keyboard mode came first.

Two players

Not on this board. A keyboard report carries six key slots and two pads can ask for ten, so a second pad could be polled and never sent. One was drawn and polled here until 2026-09-23 doing exactly that. Two players wants gamepad mode with two report IDs, on a part that can do it.

Bench mode

Not built, and the cheapest of them: the C6's USB already IS a serial port, so pad bytes over USB serial into a uinput shim need no new hardware.

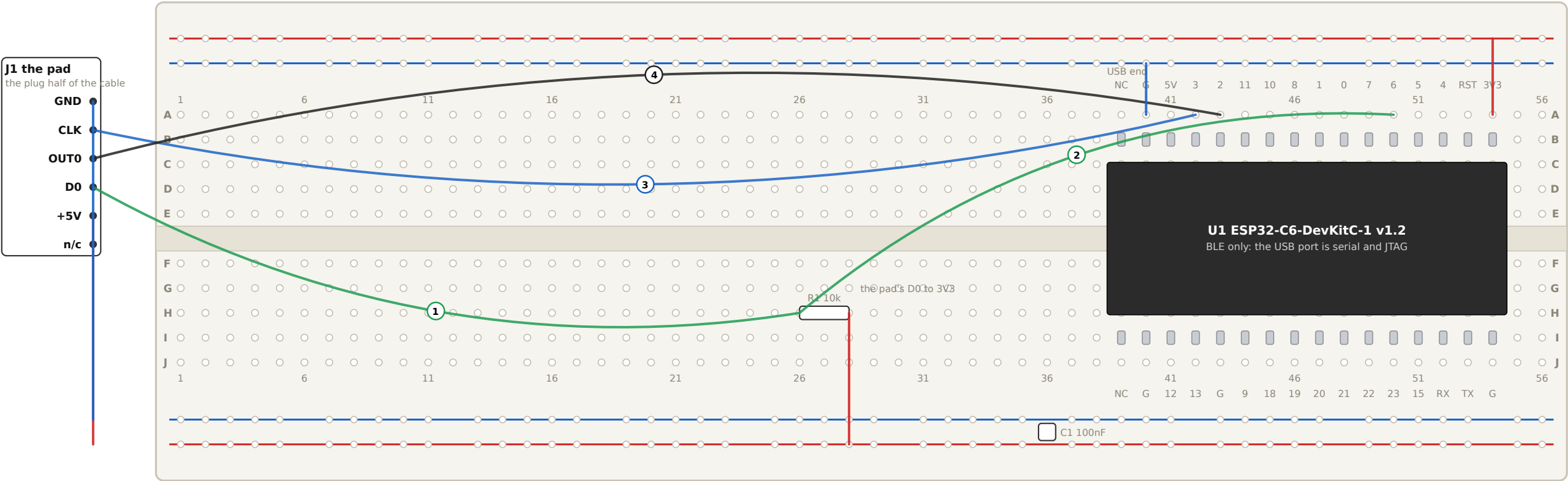
pad-ble.svg ansi-b placed at 76% drawn by tools/draw-schematics.py

REV	DATE	REVISION	tinymachines				nes-bench	
J	2026-09-26	the first pad was a replica; an original's lead ...	SHEET TITLE					
K	2026-09-26	J1 drawn as an original pad in its own colours o...	pad-ble v1: one pad into the C6, as a BLE keyboard					
L	2026-09-28	measure-first item 4 closed and the host result ...	DRAWING NO. TM-NESB-003		REV M	DATE 2026-09-30	SHEET 2 of 17	
M	2026-09-30	the BLE build moves to a classic ESP32-WROOM-32 ...	DRAWN BY bisenbek		SOURCE git 5a52db9		SCALE NTS	

87654321

pad-ble v1 on the breadboard: where everything goes

The devkit's pin labels are MEASURED off the board. It straddles the channel, so only rows A and J are reachable in its columns. The lead colours in the key are the bench's REPLICA cable; an original pad's differ.



4 numbered jumpers, and 5 short wires to the rails

Every one of these is read out of the schematic. Red stubs go to the nearest + rail, blue stubs to the nearest - rail; they are not numbered because they are all the same instruction.

- 1

PAD1_D0, the green lead: J1 D0 to R1 col 26 row H
- 2

PAD1_D0, the green lead: R1 col 26 row H to U1 pin marked 6 (col 50 row A)
- 3

PAD_CLK, the blue lead: J1 CLK to U1 pin marked 3 (col 42 row A)
- 4

PAD_LATCH, the black lead: J1 OUT0 to U1 pin marked 2 (col 43 row A)

breadboard-pad-ble.svg ansi-b placed at 77% drawn by tools/breadboard.py

REV	DATE	REVISION	tinymachines				nes-bench		
J	2026-09-26	the first pad was a replica; an original's lead ...	SHEET TITLE						
K	2026-09-26	J1 drawn as an original pad in its own colours o...	Breadboard: which hole each part and jumper goes in						
L	2026-09-28	measure-first item 4 closed and the host result ...	DRAWING NO. TM-NESB-003		REV M	DATE 2026-09-30		SHEET 4 of 17	
M	2026-09-30	the BLE build moves to a classic ESP32-WROOM-32 ...	DRAWN BY bisenbek		SOURCE git 5a52db9			SCALE NTS	

87654321

nes-bench

pad-ble v1 on the ESP32-P4: one pad into header P6

The build is USB now, not BLE. A Waveshare ESP32-P4-Module-DEV-KIT, header read from the vendor schematic 2026-09-24 and every hole confirmed with firmware/header-probe 2026-09-25. The wiring is the same either way.

THE PAD, polled at 3V3 exactly as the bridge polls one

J1

original NES pad (MN4021B)

1

GND

PAD_CLK

2

CLK

PAD_LATCH

3

OUT0

PAD1_D0

4

D0

3V3

5

+5V

6

n/c

7

n/c

1 brown 2 red 3 orange 4 yellow 5 3V3

Unchanged from pad-ble.svg, and that is the point: GPIO2, 3 and 6 exist and are free on BOTH parts, so poll_pad in firmware/pad-ble runs unedited on either. One sketch, one pad map, two targets.

The pad's 4021 is a CMOS part rated 3 to 18 V, so at 3V3 its D0 is 3V3 logic and needs no shifter. MEASURED 2026-09-27: an original's MN4021B follows its buttons at 3V3, 27 presses in 30 s under firmware/pad-diag. No 74LVC245 needed. A 10 ohm part fitted as the 10k pull-up hid it first: read the bands.

Pin 5 is the supply and 6 and 7 carry nothing, as measured on this console's own cable (docs/wiring.md). RING THE CABLE OUT UNPLUGGED: a tone through a cable still in the console runs through its pull-ups, and pins that share nothing beep.

J1 IS DRAWN AS AN ORIGINAL PAD, in Nintendo's colours, which are documented and NOT YET MEASURED: beep each wire to the MN4021B first, not the pin in the header sheet's wire table gives. The first pad on this build was a REPLICA, whose colours differ, and RED and YELLOW are different signals on each: wired by the replica's colours, an original puts its DATA on the GROUND rail.

PAD1_D0

R1

10k

3V3

THE CONTROLLER, on its own USB power, wired at header P6

U1

ESP32-P4-Module-DEV-KIT

PAD_LATCH

22

GPIO2

PAD_CLK

28

GPIO3

PAD1_D0

16

GPIO6

3V3

18

3V3

26

GND

header P6; pin numbers are P6's own

BLE

radio

USB-C

VBUS

PIN 1 IS NOT WHERE A RASPBERRY PI PUTS IT. The header is Pi-shaped and numbered the other way round: 5V on 1 and 3 where a Pi has 2 and 4, 3V3 on 2 and 18 where a Pi has 1 and 17, and every ground one pin from where a Pi user reaches. A HAT fits this board and does not work. Count from the schematic.

THE WHOLE CIRCUIT IS FIVE HOLES IN A ROW. Find the SECOND 3V3 on the even row; it reads

6 3V3 3 2 0 GND
p16 p18 p20 p22 p24 p26

and the build takes all of them EXCEPT 0. One landmark and one hole to skip beats five numbers: wired by number on 2026-09-25, two of the three signal wires went in wrong.

TEN PINS REACH P6 AND ARE STILL NOT YOURS. Worst is GPIO54 (pin 31), the onboard C6's RESET: take it and the radio dies. Sheets 8 to 10 list all ten with what each is wired to.

MODE_SW AND LED MOVED, and had to. On the C6 they were GPIO10 and GPIO11 and cost nothing because nothing was wired to them. Here those numbers are the ES8311 codec's I2S and are not on the header at all. They are GPIO21 and GPIO20 now, P6 pins 12 and 14, free and on the same even row. Declared, not wired.

C1 goes across the breadboard's own 3V3 and GND rails, not the board's pins: the module is regulated already and this is for the breadboard's inductance, as on the bridge since 2026-09-15.

WHAT THIS BOARD DOES, MEASURED, and why the direction changed on 2026-09-25

USB HID

THIS IS THE BUILD, PROVEN 2026-09-27: an original pad at 3V3, every button and chord, and a Linux host enumerating 303a:0002 NES Pad at full speed and receiving all eight keys. TinyUSB driven directly on the P4's FULL-speed OTG controller, PHY 0 (GPIO24/25) swapped from USB-Serial-JTAG to it: the Type-C socket marked USB, not PWR USB TO UART. The HOST/DEVICE jumper is the high-speed switch's and does not matter here. No radio, no co-processor, no pairing. Not yet tried: a phone, a browser page, latency. UART stays the log.

BLE HID

REACHES THE AIR BUT CRASHES THIS FIRMWARE, and not in our code. The P4 has no radio; the module's ESP32-C6 is it, over SDIO. The host's esp-hosted 2.12.11 asks that C6 for its firmware version, gets no answer, and the failure path reads a pointer nobody filled in: always 2881 ms after BLEDevice::init. Five explanations tested and eliminated (build options and PSRAM, the security block, a delay before init, a delay after init, our own GPIO setup). A smaller sketch survives the IDENTICAL timeout, so the garbage it reads is harmless there: luck, not correctness. The proper fix is the C6's slave firmware, through the C6 UART header. Not done.

Proven

The radio itself works: on 2026-09-24 the bench Pi's own receiver heard this board advertise at B0:A6:04:8A:D7:0E, which is NOT the P4's MAC, because the advertisement came off the C6. The BLE fault is the version RPC and its aftermath, not the link.

Flashing

First try, where the C6 devkit never once answered. On /dev/p4-uart through the board's CH343 at 1a86:55d3. The factory image was read off and kept before anything was written: 16 MB, sha256 ad22bf07..., so this board goes back if it has to.

Two players

Not on this sheet. A keyboard report carries six key slots and two pads can ask for ten, so a second pad could be polled and never sent.

pad-ble-p4.svg ansi-b placed at 76% drawn by tools/draw-schematics.py

87654321

REV

DATE

REVISION

tinymachines

nes-bench

SHEET TITLE

pad-ble v1 on the ESP32-P4: one pad into header P6

DRAWING NO.

TM-NESB-003

REV

M

DATE

2026-09-30

SHEET

5 of 17

DRAWN BY

bisenbek

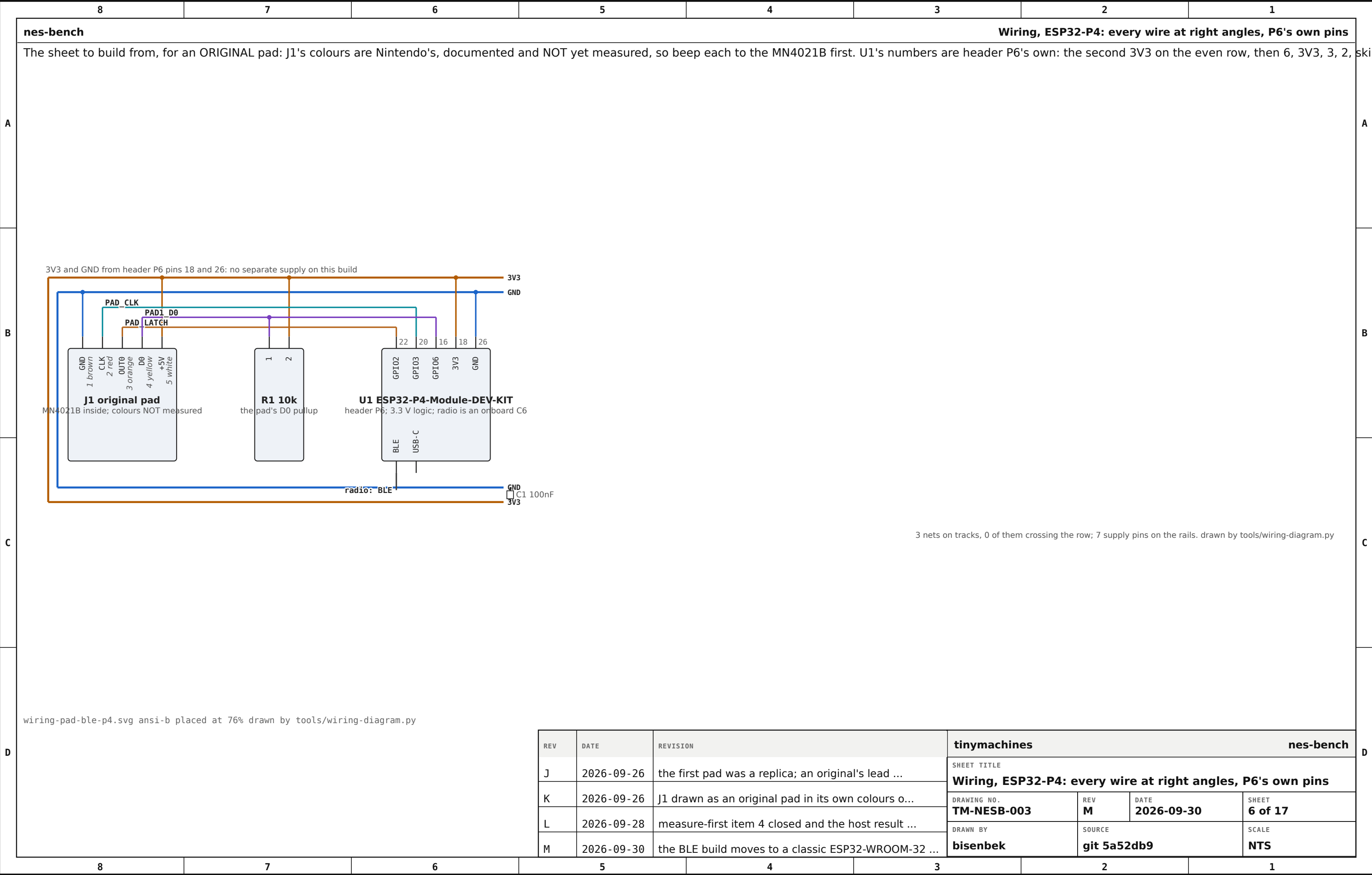
SOURCE

git 5a52db9

SCALE

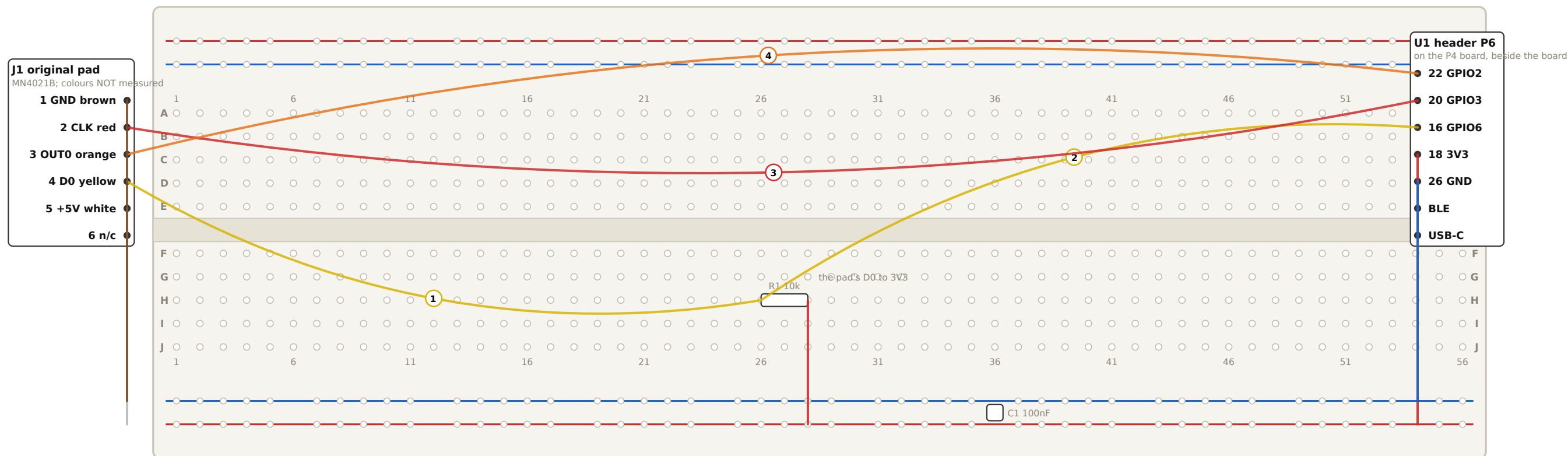
NTS

87654321



pad-ble v1 on the ESP32-P4: where everything goes

An ORIGINAL pad in Nintendo's colours, documented and NOT yet measured: beep each wire to its MN4021B first. The board sits beside the breadboard. A replica's colours differ; red and yellow swap meaning.



4 numbered jumpers, and 5 short wires to the rails

Every one of these is read out of the schematic. Red stubs go to the nearest + rail, blue stubs to the nearest - rail, except a stub from the pad, which is drawn in its own lead's colour; they are not numbered because they are all the same instruction.

- 1 PAD1_D0, the yellow lead: J1 D0 to R1 col 26 row H
- 2 PAD1_D0, the yellow lead: R1 col 26 row H to U1 GPIO6
- 3 PAD_CLK, the red lead: J1 CLK to U1 GPIO3
- 4 PAD_LATCH, the orange lead: J1 OUT0 to U1 GPIO2

breadboard-pad-ble-p4.svg ansi-b placed at 77% drawn by tools/breadboard.py

REV	DATE	REVISION	tinymachines				nes-bench	
J	2026-09-26	the first pad was a replica; an original's lead ...	SHEET TITLE					Breadboard, ESP32-P4: the board sits beside it, not on it
K	2026-09-26	J1 drawn as an original pad in its own colours o...	DRAWING NO.		REV	DATE	SHEET	
L	2026-09-28	measure-first item 4 closed and the host result ...	TM-NESB-003		M	2026-09-30	7 of 17	
M	2026-09-30	the BLE build moves to a classic ESP32-WROOM-32 ...	DRAWN BY		SOURCE		SCALE	
			bisenbek		git 5a52db9		NTS	

	6	5	4	3	2	1																																																																													
nes-bench							ESP32-P4 module header P6: where each wire goes																																																																												
A	The five wires, and which lead is which on each kind of pad																A																																																																		
	Every one is an EVEN pin on P6, and they are CONSECUTIVE: find the second 3V3 on that row and the circuit is 6, 3V3, 3, 2, skip 0, GND. THE LEAD COLOURS DEPEND ON THE PAD. The replica column is the bench's cable, rung out 2026-09-09. The original column is Nintendo's documented scheme, NOT YET MEASURED: beep each wire to the pad's own 4021 at the pin shown before powering it. DO NOT WIRE ONE BY THE OTHER'S COLOURS: RED is supply on the replica but clock on an original and YELLOW is ground on the replica but data on an original, so an original wired by replica colours puts its data output on the ground rail.																																																																																		
B	<table><tr><th>SIGNAL</th><th>P6 PIN</th><th>REPLICA LEAD</th><th>ORIGINAL LEAD</th><th>ORIGINAL: BEEP TO 4021</th></tr><tr><td>supply</td><td>pin 18</td><td>Red</td><td>White</td><td>pin 16 (VDD)</td></tr><tr><td>ground</td><td>pin 26</td><td>Yellow</td><td>Brown</td><td>pin 8 (VSS)</td></tr><tr><td>latch</td><td>pin 22</td><td>Black</td><td>Orange</td><td>pin 9 (P/S)</td></tr><tr><td>clock</td><td>pin 20</td><td>Blue</td><td>Red</td><td>pin 10 (CLOCK)</td></tr><tr><td>data</td><td>pin 16</td><td>Green</td><td>Yellow</td><td>pin 3 (Q8)</td></tr></table>																SIGNAL	P6 PIN	REPLICA LEAD	ORIGINAL LEAD	ORIGINAL: BEEP TO 4021	supply	pin 18	Red	White	pin 16 (VDD)	ground	pin 26	Yellow	Brown	pin 8 (VSS)	latch	pin 22	Black	Orange	pin 9 (P/S)	clock	pin 20	Blue	Red	pin 10 (CLOCK)	data	pin 16	Green	Yellow	pin 3 (Q8)	B																																				
	SIGNAL	P6 PIN	REPLICA LEAD	ORIGINAL LEAD	ORIGINAL: BEEP TO 4021																																																																														
	supply	pin 18	Red	White	pin 16 (VDD)																																																																														
	ground	pin 26	Yellow	Brown	pin 8 (VSS)																																																																														
	latch	pin 22	Black	Orange	pin 9 (P/S)																																																																														
	clock	pin 20	Blue	Red	pin 10 (CLOCK)																																																																														
data	pin 16	Green	Yellow	pin 3 (Q8)																																																																															
C																	C																																																																		
D	generated from tools/cheatsheet.py: the schematic, the lab log and tools/bringup.py's lead and ribbon tables																D																																																																		
	<table><tr><td>REV</td><td>DATE</td><td>REVISION</td><td colspan="13">tinymachines nes-bench</td></tr><tr><td>J</td><td>2026-09-26</td><td>the first pad was a replica; an original's lead ...</td><td colspan="13">SHEET TITLE ESP32-P4 module header P6: where each wire goes</td></tr><tr><td>K</td><td>2026-09-26</td><td>J1 drawn as an original pad in its own colours o...</td><td colspan="4">DRAWING NO. TM-NESB-003</td><td>REV M</td><td colspan="4">DATE 2026-09-30</td><td colspan="4">SHEET 8 of 17</td></tr><tr><td>L</td><td>2026-09-28</td><td>measure-first item 4 closed and the host result ...</td><td colspan="4" rowspan="2">DRAWN BY bisenbek</td><td colspan="4" rowspan="2">SOURCE git 5a52db9</td><td colspan="4" rowspan="2">SCALE NTS</td></tr><tr><td>M</td><td>2026-09-30</td><td>the BLE build moves to a classic ESP32-WROOM-32 ...</td></tr></table>																	REV	DATE	REVISION	tinymachines nes-bench													J	2026-09-26	the first pad was a replica; an original's lead ...	SHEET TITLE ESP32-P4 module header P6: where each wire goes													K	2026-09-26	J1 drawn as an original pad in its own colours o...	DRAWING NO. TM-NESB-003				REV M	DATE 2026-09-30				SHEET 8 of 17				L	2026-09-28	measure-first item 4 closed and the host result ...	DRAWN BY bisenbek				SOURCE git 5a52db9				SCALE NTS				M	2026-09-30	the BLE build moves to a classic ESP32-WROOM-32 ...
	REV	DATE	REVISION	tinymachines nes-bench																																																																															
	J	2026-09-26	the first pad was a replica; an original's lead ...	SHEET TITLE ESP32-P4 module header P6: where each wire goes																																																																															
	K	2026-09-26	J1 drawn as an original pad in its own colours o...	DRAWING NO. TM-NESB-003				REV M	DATE 2026-09-30				SHEET 8 of 17																																																																						
	L	2026-09-28	measure-first item 4 closed and the host result ...	DRAWN BY bisenbek				SOURCE git 5a52db9				SCALE NTS																																																																							
M	2026-09-30	the BLE build moves to a classic ESP32-WROOM-32 ...																																																																																	

	6	5	4	3	2	1																																																																																										
nes-bench																																																																																																
On the header and already spoken for			On the header and free																																																																																													
These reach the header and are still not yours. Driving one fights a part soldered to the board.			Nothing else on the board touches these. pad-ble takes GPIO2, 3 and 6 from here, which is the same map firmware/bridge/bridge.ino already polls a pad on, so poll_pad runs unedited.																																																																																													
<table><tr><th>GPIO</th><th>P6 PIN</th><th>ALREADY WIRED TO</th></tr><tr><td>GPI054</td><td>pin 31</td><td>the C6's RESET</td></tr><tr><td>GPI045</td><td>pin 39</td><td>MicroSD power switch</td></tr><tr><td>GPI053</td><td>pin 36</td><td>speaker amp CTRL</td></tr><tr><td>GPI036</td><td>pin 23</td><td>BOOT_MODE2 strap</td></tr><tr><td>GPI024</td><td>pin 28</td><td>USB1P1_N</td></tr><tr><td>GPI025</td><td>pin 27</td><td>USB1P1_P</td></tr><tr><td>GPI07</td><td>pin 4</td><td>codec SDA, 2.2k pullup</td></tr><tr><td>GPI08</td><td>pin 6</td><td>codec SCL, 2.2k pullup</td></tr><tr><td>GPI037</td><td>pin 7</td><td>console UART TXD</td></tr><tr><td>GPI038</td><td>pin 9</td><td>console UART RXD</td></tr></table>			GPIO	P6 PIN	ALREADY WIRED TO	GPI054	pin 31	the C6's RESET	GPI045	pin 39	MicroSD power switch	GPI053	pin 36	speaker amp CTRL	GPI036	pin 23	BOOT_MODE2 strap	GPI024	pin 28	USB1P1_N	GPI025	pin 27	USB1P1_P	GPI07	pin 4	codec SDA, 2.2k pullup	GPI08	pin 6	codec SCL, 2.2k pullup	GPI037	pin 7	console UART TXD	GPI038	pin 9	console UART RXD	<table><tr><th>GPIO</th><th>P6 PIN</th><th>NOTE</th></tr><tr><td>GPI00</td><td>pin 24</td><td></td></tr><tr><td>GPI01</td><td>pin 21</td><td></td></tr><tr><td>GPI02</td><td>pin 22</td><td>PAD_LATCH</td></tr><tr><td>GPI03</td><td>pin 20</td><td>PAD_CLK</td></tr><tr><td>GPI04</td><td>pin 17</td><td></td></tr><tr><td>GPI05</td><td>pin 15</td><td></td></tr><tr><td>GPI06</td><td>pin 16</td><td>PAD1_D0</td></tr><tr><td>GPI020</td><td>pin 14</td><td>LED, not wired</td></tr><tr><td>GPI021</td><td>pin 12</td><td>MODE_SW, not wired</td></tr><tr><td>GPI022</td><td>pin 11</td><td></td></tr><tr><td>GPI023</td><td>pin 8</td><td></td></tr><tr><td>GPI026</td><td>pin 32</td><td></td></tr><tr><td>GPI027</td><td>pin 37</td><td></td></tr><tr><td>GPI032</td><td>pin 25</td><td></td></tr><tr><td>GPI033</td><td>pin 30</td><td></td></tr><tr><td>GPI046</td><td>pin 35</td><td></td></tr><tr><td>GPI047</td><td>pin 38</td><td></td></tr><tr><td>GPI048</td><td>pin 34</td><td></td></tr></table>				GPIO	P6 PIN	NOTE	GPI00	pin 24		GPI01	pin 21		GPI02	pin 22	PAD_LATCH	GPI03	pin 20	PAD_CLK	GPI04	pin 17		GPI05	pin 15		GPI06	pin 16	PAD1_D0	GPI020	pin 14	LED, not wired	GPI021	pin 12	MODE_SW, not wired	GPI022	pin 11		GPI023	pin 8		GPI026	pin 32		GPI027	pin 37		GPI032	pin 25		GPI033	pin 30		GPI046	pin 35		GPI047	pin 38		GPI048	pin 34	
GPIO	P6 PIN	ALREADY WIRED TO																																																																																														
GPI054	pin 31	the C6's RESET																																																																																														
GPI045	pin 39	MicroSD power switch																																																																																														
GPI053	pin 36	speaker amp CTRL																																																																																														
GPI036	pin 23	BOOT_MODE2 strap																																																																																														
GPI024	pin 28	USB1P1_N																																																																																														
GPI025	pin 27	USB1P1_P																																																																																														
GPI07	pin 4	codec SDA, 2.2k pullup																																																																																														
GPI08	pin 6	codec SCL, 2.2k pullup																																																																																														
GPI037	pin 7	console UART TXD																																																																																														
GPI038	pin 9	console UART RXD																																																																																														
GPIO	P6 PIN	NOTE																																																																																														
GPI00	pin 24																																																																																															
GPI01	pin 21																																																																																															
GPI02	pin 22	PAD_LATCH																																																																																														
GPI03	pin 20	PAD_CLK																																																																																														
GPI04	pin 17																																																																																															
GPI05	pin 15																																																																																															
GPI06	pin 16	PAD1_D0																																																																																														
GPI020	pin 14	LED, not wired																																																																																														
GPI021	pin 12	MODE_SW, not wired																																																																																														
GPI022	pin 11																																																																																															
GPI023	pin 8																																																																																															
GPI026	pin 32																																																																																															
GPI027	pin 37																																																																																															
GPI032	pin 25																																																																																															
GPI033	pin 30																																																																																															
GPI046	pin 35																																																																																															
GPI047	pin 38																																																																																															
GPI048	pin 34																																																																																															
generated from tools/cheatsheet.py: the schematic, the lab log and tools/bringup.py's lead and ribbon tables																																																																																																
<table><tr><th>REV</th><th>DATE</th><th>REVISION</th></tr><tr><td>J</td><td>2026-09-26</td><td>the first pad was a replica; an original's lead ...</td></tr><tr><td>K</td><td>2026-09-26</td><td>J1 drawn as an original pad in its own colours o...</td></tr><tr><td>L</td><td>2026-09-28</td><td>measure-first item 4 closed and the host result ...</td></tr><tr><td>M</td><td>2026-09-30</td><td>the BLE build moves to a classic ESP32-WROOM-32 ...</td></tr></table>			REV	DATE	REVISION	J	2026-09-26	the first pad was a replica; an original's lead ...	K	2026-09-26	J1 drawn as an original pad in its own colours o...	L	2026-09-28	measure-first item 4 closed and the host result ...	M	2026-09-30	the BLE build moves to a classic ESP32-WROOM-32 ...	<table><tr><th colspan="4">tinymachines</th><th colspan="2">nes-bench</th></tr><tr><td colspan="6">SHEET TITLE</td></tr><tr><td colspan="6">ESP32-P4 module header P6: where each wire goes</td></tr><tr><td colspan="2">DRAWING NO.</td><td>REV</td><td>DATE</td><td colspan="2">SHEET</td></tr><tr><td colspan="2">TM-NESB-003</td><td>M</td><td>2026-09-30</td><td colspan="2">9 of 17</td></tr><tr><td colspan="2">DRAWN BY</td><td colspan="2">SOURCE</td><td colspan="2">SCALE</td></tr><tr><td colspan="2">bisenbek</td><td colspan="2">git 5a52db9</td><td colspan="2">NTS</td></tr></table>				tinymachines				nes-bench		SHEET TITLE						ESP32-P4 module header P6: where each wire goes						DRAWING NO.		REV	DATE	SHEET		TM-NESB-003		M	2026-09-30	9 of 17		DRAWN BY		SOURCE		SCALE		bisenbek		git 5a52db9		NTS																																		
REV	DATE	REVISION																																																																																														
J	2026-09-26	the first pad was a replica; an original's lead ...																																																																																														
K	2026-09-26	J1 drawn as an original pad in its own colours o...																																																																																														
L	2026-09-28	measure-first item 4 closed and the host result ...																																																																																														
M	2026-09-30	the BLE build moves to a classic ESP32-WROOM-32 ...																																																																																														
tinymachines				nes-bench																																																																																												
SHEET TITLE																																																																																																
ESP32-P4 module header P6: where each wire goes																																																																																																
DRAWING NO.		REV	DATE	SHEET																																																																																												
TM-NESB-003		M	2026-09-30	9 of 17																																																																																												
DRAWN BY		SOURCE		SCALE																																																																																												
bisenbek		git 5a52db9		NTS																																																																																												
	6	5	4	3	2	1																																																																																										

6

5

4

3

2

1

nes-bench

ESP32-P4 module header P6: where each wire goes (continued)

A

Not brought out to the header at all

Recorded so nobody goes looking for them. The SDIO group is the whole reason a part with no radio can advertise: it is the link to the ESP32-C6 sealed in the module.

GPIOs	WHAT HAS THEM	CONSEQUENCE
GPIO9 to GPIO13	I2S to the audio codec	out of reach
GPIO14 to GPIO19	SDIO to the onboard C6	out of reach
GPIO28 to GPIO31	Ethernet PHY	out of reach
GPIO34, GPIO35	Ethernet PHY; 35 is BOOT_MODE	out of reach
GPIO39 to GPIO44	MicroSD card	out of reach
GPIO49 to GPIO52	Ethernet PHY	out of reach

B

C

D

generated from tools/cheatsheet.py: the schematic, the lab log and tools/bringup.py's lead and ribbon tables

REV	DATE	REVISION	tinymachines				nes-bench			
J	2026-09-26	the first pad was a replica; an original's lead ...	SHEET TITLE							
K	2026-09-26	J1 drawn as an original pad in its own colours o...	ESP32-P4 module header P6: where each wire goes							
L	2026-09-28	measure-first item 4 closed and the host result ...	DRAWING NO.		REV	DATE	SHEET			
M	2026-09-30	the BLE build moves to a classic ESP32-WROOM-32 ...	TM-NESB-003		M	2026-09-30	10 of 17			
			DRAWN BY		SOURCE	SCALE				
			bisenbek		git 5a52db9	NTS				

6

5

4

3

2

1

87654321

nes-bench

pad-ble v1 on a classic ESP32: one pad into the devkit

The BLE build, on the part with its own radio. A 30-pin ESP32-WROOM-32 devkit, header read off the board 2026-09-30. NOT the C6 map: GPIO6 is this module's flash and GPIO3 its console, so the pad moves to D25, D26, D27.

THE PAD, polled at 3V3 exactly as the bridge polls one

J1

original NES pad (MN4021B)

1

GND

PAD_CLK

2

CLK

PAD_LATCH

3

OUT0

PAD1_D0

4

D0

3V3

5

+5V

n/c

6

n/c

7

1 brown 2 red 3 orange 4 yellow 5 white, NOT measured

The pad side is the P4 sheet's: the same plug, pullup and poll. Only the three pins it lands on move, because this part gives no choice (see the controller zone).

The pad's 4021 is a CMOS part rated 3 to 18 V, so at 3V3 its D0 is 3V3 logic and needs no shifter. MEASURED 2026-09-27: an original's MN4021B follows its buttons at 3V3 (pad-ble-p4.svg). Pin 5 is the supply; 6 and 7 carry nothing. RING THE CABLE OUT UNPLUGGED. A 10 ohm part fitted as the 10k hid it first.

J1 IS DRAWN AS AN ORIGINAL PAD, in Nintendo's colours, NOT YET MEASURED: beep each wire to the MN4021B first. A REPLICA's differ, and RED and YELLOW are different signals on each: wired by the replica's colours, an original puts its DATA on the GROUND rail. NO PAD IS WIRED TO THIS BOARD YET. With D27 floating the byte reads ff, all eight pressed, and the blue LED on D2 is on.

PAD1_D0

R1

10k

3V3

THE CONTROLLER, on its own USB power, labels as the board prints them

U1

ESP32-WROOM-32 devkit

PAD_LATCH

D25

PAD_CLK

D26

PAD1_D0

D27

n/c

D4

n/c

D2

3V3

3V3

GND

GND

30 pins; the labels are the board's own silkscreen; D<n> is GPIO<n>

3V3

C1

100nF

WHY NOT GPIO2, 3 AND 6. On an ESP32-WROOM-32 GPIO6 to GPIO11 are the module's own flash and never reach the header, and GPIO3 is U0RXD, the receive side of the CP2102 that is this board's only console. The C6 map would drive the flash the program runs from. So: D25 latch, D26 clock, D27 data, ADJACENT on the EN side, GND four pins below them on the same side, 3V3 the first pin of the other side. The EN side reads, up from the USB end:

VIN GND 13 12 14 27 26 25 33 32 35 34 VN VP EN

D4 (MODE_SW) and D2 (LED) are NC here: the firmware reads one with a pullup and drives the other. D2 is the devkit's own blue LED, so the pad-held light needs no wiring on this board.

ALSO NOT YOURS: TX0 and RX0 (the console), D12 and D15 (strapping pins the module samples at reset), and D34, D35, VP, VN, which are input only with no pullup. The header page lists them.

C1 goes across the breadboard's own 3V3 and GND rails, not the board's pins, as on every sheet since 2026-09-15.

WHAT THIS BOARD DOES, MEASURED 2026-09-30 with the bench head as the host

BLE HID

ADVERTISES AND BONDS. Boots straight into firmware/pad-ble built for esp32:esp32:esp32 (84% of the app partition), prints the key map and advertising as NES Pad, and the bench head's own radio lists it. BLEDevice::init does not crash here, unlike the P4. Paired from the head with an agent registered (a non-interactive bluetoothctl pair fails, no agent for the request), it bonds and the head grows a NES Pad keyboard on uhid. One connect prints one linked line, one disconnect one unlinked. NOT TESTED: a pad wired to it, a phone. A floating D27 reads ff, all pressed.

FORGET

WORKS ON THIS PART NOW, AND DID NOT. The sketch cleared bonds only under NimBLE; this core is Bluedroid, so FORGET printed rc -1 and cleared nothing while a reflash blanked the bond store anyway: the one-sided bond with no way out. It lists and removes the bonds one by one now, measured both ways: board blank and head bonded, the head connects and gets nothing (bluetoothd's unlikely error lines are the only tell); FORGET with one bond stored answers 1 bond(s) were stored here, rc 0, and the head's next connect drops until it removes its side and pairs again.

Flashing

Through the CP2102 at /dev/ttyUSB0 on the bench head, esptool's default reset, the merged image at 0x0 at 921600. The board arrived running an AT firmware (an at_customize partition, version line 2.4.0). Its ROM banner appears only when the reset is driven through RTS, not DTR alone.

Serial

Unlike the P4 over USB, this board's Serial IS the CP2102, so the B xx lines and STATUS, KEYS and FORGET are readable on the bench head at 115200.

Two players

Not on this sheet. A keyboard report carries six key slots and two pads can ask for ten, so a second pad could be polled and never sent.

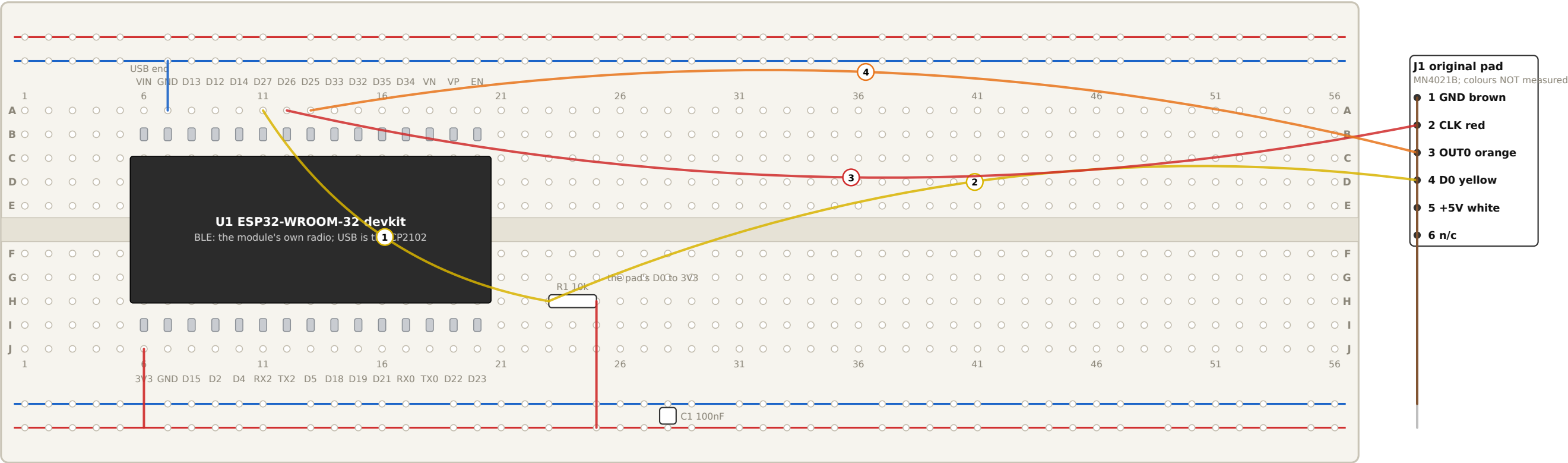
pad-ble-esp32.svg ansi-b placed at 76% drawn by tools/draw-schematics.py

REV	DATE	REVISION	tinymachines nes-bench			
J	2026-09-26	the first pad was a replica; an original's lead ...	SHEET TITLE pad-ble v1 on a classic ESP32: one pad into the devkit			
K	2026-09-26	J1 drawn as an original pad in its own colours o...	DRAWING NO. TM-NESB-003	REV M	DATE 2026-09-30	SHEET 11 of 17
L	2026-09-28	measure-first item 4 closed and the host result ...	DRAWN BY bisenbek		SOURCE git 5a52db9	SCALE NTS
M	2026-09-30	the BLE build moves to a classic ESP32-WROOM-32 ...				

87654321

pad-ble v1 on a classic ESP32: where everything goes

The devkit's labels are read off the owner's photograph of 2026-09-30, and the column numbers are that photograph's breadboard rows: USB at row 1, the pins in 6 to 20, the pad's three in 11 to 13 on the EN row. Only rows A and J are reachable under it. An ORIGINAL pad's colours; a repl



4 numbered jumpers, and 5 short wires to the rails

Every one of these is read out of the schematic. Red stubs go to the nearest + rail, blue stubs to the nearest - rail, except a stub from the pad, which is drawn in its own lead's colour; they are not numbered because they are all the same instruction.

- 1

PAD1_D0, the yellow lead: U1 pin marked D27 (col 11 row A) to R1 col 23 row H
- 2

PAD1_D0, the yellow lead: R1 col 23 row H to J1 D0
- 3

PAD_CLK, the red lead: U1 pin marked D26 (col 12 row A) to J1 CLK
- 4

PAD_LATCH, the orange lead: U1 pin marked D25 (col 13 row A) to J1 OUT0

breadboard-pad-ble-esp32.svg ansi-b placed at 77% drawn by tools/breadboard.py

REV	DATE	REVISION	tinymachines				nes-bench	
J	2026-09-26	the first pad was a replica; an original's lead ...	SHEET TITLE Breadboard, classic ESP32: on the board, rows 6 to 20					
K	2026-09-26	J1 drawn as an original pad in its own colours o...	DRAWING NO. TM-NESB-003		REV M	DATE 2026-09-30		SHEET 13 of 17
L	2026-09-28	measure-first item 4 closed and the host result ...	DRAWN BY		SOURCE		SCALE	
M	2026-09-30	the BLE build moves to a classic ESP32-WROOM-32 ...	bisenbek		git 5a52db9		NTS	

6

5

4

3

2

1

nes-bench

WROOM-32 header: where each wire goes

A

The five wires, and which lead is which on each kind of pad

USB at breadboard row 1, so the pins are rows 6 to 20 and a row number names a hole. The three signals are ADJACENT on the EN side (D14 D27 D26 D25 D33, up from the USB end), GND four rows below on the same side, 3V3 the first pin across. NOT the C6 map: GPIO6 is this module's flash, GPIO3 its console. THE LEAD COLOURS DEPEND ON THE PAD: replica rung out 2026-09-09; original NOT YET MEASURED, beep each to the 4021 first. RED is supply on the replica but clock on an original and YELLOW is ground on the replica but data on an original.

SIGNAL	BOARD PIN	REPLICA LEAD	ORIGINAL LEAD	ORIGINAL: BEEP TO 4021
supply	3V3, row 6	Red	White	pin 16 (VDD)
ground	GND, row 7	Yellow	Brown	pin 8 (VSS)
latch	D25, row 13	Black	Orange	pin 9 (P/S)
clock	D26, row 12	Blue	Red	pin 10 (CLOCK)
data	D27, row 11	Green	Yellow	pin 3 (Q8)

B

C

C

D

generated from tools/cheatsheet.py: the schematic, the lab log and tools/bringup.py's lead and ribbon tables

REV	DATE	REVISION	tinymachines nes-bench				
J	2026-09-26	the first pad was a replica; an original's lead ...	SHEET TITLE WROOM-32 header: where each wire goes				
K	2026-09-26	J1 drawn as an original pad in its own colours o...	DRAWING NO. TM-NESB-003		REV M	DATE 2026-09-30	SHEET 14 of 17
L	2026-09-28	measure-first item 4 closed and the host result ...	DRAWN BY bisenbek		SOURCE git 5a52db9		SCALE NTS
M	2026-09-30	the BLE build moves to a classic ESP32-WROOM-32 ...					

D

6

5

4

3

2

1

	6	5	4	3	2	1
	nes-bench			WROOM-32 header: where each wire goes (continued)		
A	Spoken for, or not there at all			On the header and free		
	Two are the console this board is flashed and read over, two are strapping pins sampled at reset. Below the line, what never reaches the header: the first row is why the C6 map cannot be used on this part.			pad-ble takes D25, D26, D27; D4 and D2 are the mode switch and the LED, declared and not wired. Not listed: D34, D35, VP, VN (GPIO34, 35, 36, 39), input only with no pullup, so not free for an output.		
B	GPIO	BOARD PIN	ALREADY WIRED TO	GPIO	BOARD PIN	NOTE
	GPI01	TX0, row 18	U0TXD, the console	GPI02	D2, row 9	LED, not wired; strap
	GPI03	RX0, row 17	U0RXD, the console	GPI04	D4, row 10	MODE_SW, not wired
	GPI012	D12, row 9	MTDI strap, 1.8 V flash	GPI05	D5, row 13	strapping pin
	GPI015	D15, row 8	MTDO strap, ROM log	GPI013	D13, row 8	
	GPI06-11	not on the header	the module's SPI flash	GPI014	D14, row 10	
C	GPI00	not on the header	the BOOT button	GPI016	RX2, row 11	
	9 more	not on the header	20,24,28-31,37,38 absent	GPI017	TX2, row 12	
				GPI018	D18, row 14	
				GPI019	D19, row 15	
				GPI021	D21, row 16	
				GPI022	D22, row 19	
D				GPI023	D23, row 20	
				GPI025	D25, row 13	PAD_LATCH
				GPI026	D26, row 12	PAD_CLK
				GPI027	D27, row 11	PAD1_D0
				GPI032	D32, row 15	
				GPI033	D33, row 14	
generated from tools/cheatsheet.py: the schematic, the lab log and tools/bringup.py's lead and ribbon tables						
D	REV	DATE	REVISION	tinymachines		
	J	2026-09-26	the first pad was a replica; an original's lead ...	nes-bench		
	K	2026-09-26	J1 drawn as an original pad in its own colours o...	WROOM-32 header: where each wire goes		
	L	2026-09-28	measure-first item 4 closed and the host result ...	DRAWING NO.	REV	DATE
	M	2026-09-30	the BLE build moves to a classic ESP32-WROOM-32 ...	TM-NESB-003	M	2026-09-30
				SHEET	15 of 17	
				DRAWN BY	SOURCE	SCALE
				bisenbek	git 5a52db9	NTS
	6	5	4	3	2	1

6

5

4

3

2

1

nes-bench

Wiring list, pad-ble v1

A

Every wire on the board, net by net

Connect everything on a row together. Read back out of the schematic, so it cannot disagree with it.

NET	ENDS	JOIN THESE
3V3	4	J1-5, R1-2, U1 3V3, C1-1
GND	3	J1-1, U1 GND, C1-2
VBUS	1	U1 USB-C
PAD1_D0	3	J1-4, R1-1, U1 GPIO6
PAD_CLK	2	J1-2, U1 GPIO3
PAD_LATCH	2	J1-3, U1 GPIO2
radio	1	U1 BLE

A

B

B

C

C

D

REV	DATE	REVISION	tinymachines nes-bench			
J	2026-09-26	the first pad was a replica; an original's lead ...	SHEET TITLE Wiring list, pad-ble v1			
K	2026-09-26	J1 drawn as an original pad in its own colours o...	DRAWING NO. TM-NESB-003	REV M	DATE 2026-09-30	SHEET 16 of 17
L	2026-09-28	measure-first item 4 closed and the host result ...	DRAWN BY bisenbek		SOURCE git 5a52db9	SCALE NTS
M	2026-09-30	the BLE build moves to a classic ESP32-WROOM-32 ...				

D

6

5

4

3

2

1

6		5		4		3		2		1					
nes-bench										Parts list					
Everything that has to be in a drawer															
Parts from every sheet, deduplicated. The status column is the only authored thing in this package's tables.															
A	PART		STATUS		NOTE							A			
	74HCT04		on hand		SN74HCT04N, the tube that arrived 2026-09-09										
	74HC165		on hand		the TI bag. v2b needs a second one										
	100nF		on hand		v2b runs seven										
	1k		on hand		R2, the clock pin's series element, placed 2026-09-15										
	100pF		on hand		C4, the clock pin's filter, placed 2026-09-15										
	Arduino UNO R3 (ATmega328P)		on hand		the pile										
	74HC595		on hand		the box of 30										
	100R		on hand		the trigger										
B	PC817 module		on hand		module, for the reset pads							B			
	relay module, 5 V coil		on hand		5 V coil with an opto input										
	LM1881N		TO ORDER		the one real order. An old National part, mostly resellers now: buy two										
	680k		check		LM1881 RSET. One resistor										
	74HCT165		on hand		the C6 sheets' register. HCT there because the C6 drives it at 3.3 V										
	74LVC245		on hand		C6 sheets only; no UNO sheet uses one any more										
	ESP32-C6-DevKitC-1 v1.2		on hand		DevKitC-1 v1.2, the alternative build										
	10k		on hand		pad adapter only; fitted on the P4 build 2026-09-27. The first part fitted there was 10 ohms: read										
	TP4056 + protection		to order		pad adapter only										
	MCP1700-3302 LDO		to order		pad adapter only										
C	slide		to order		pad adapter only, the power switch							C			
	power		to order		pad adapter only, the battery										
	+330R		check		pad adapter only, the LED										
	ESP32-WROOM-32 devkit		on hand		the 30-pin devkit on the breadboard, the BLE build since 2026-09-30										
D												D			
	REV		DATE		REVISION		tinymachines nes-bench								
	J		2026-09-26		the first pad was a replica; an original's lead ...		SHEET TITLE Parts list								
	K		2026-09-26		J1 drawn as an original pad in its own colours o...		DRAWING NO. TM-NESB-003		REV M		DATE 2026-09-30		SHEET 17 of 17		
	L		2026-09-28		measure-first item 4 closed and the host result ...		DRAWN BY bisenbek		SOURCE git 5a52db9			SCALE NTS			
	M		2026-09-30		the BLE build moves to a classic ESP32-WROOM-32 ...										
6		5		4		3		2		1					