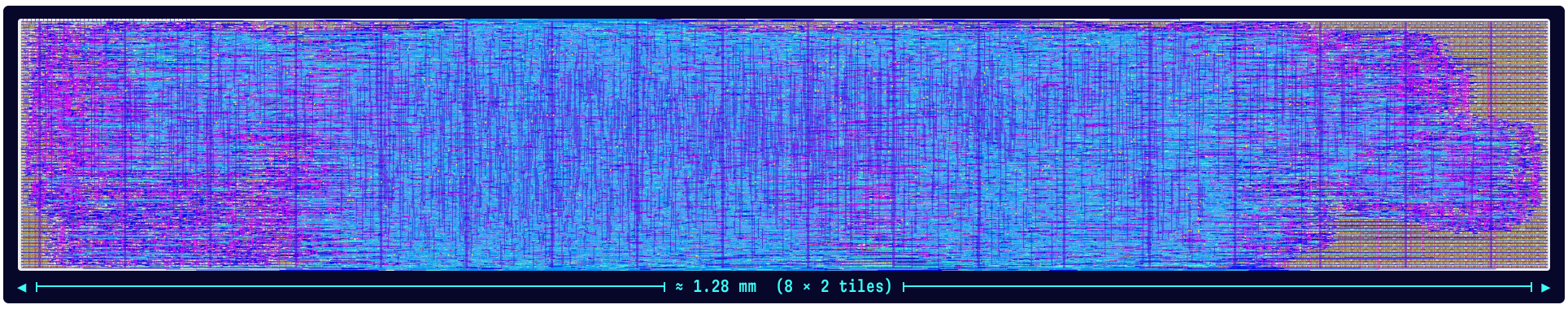
TINY TAPEOUT 06 · PROJECT 910 · SKYWATER 130 nm · tt_um_kianV_rv32ima_uLinux_SoC

THIS CHIP BOOTS LINUX.

The **KianV µLinux SoC** is a 32-bit RISC-V computer that fits in sixteen Tiny Tapeout tiles — about **1.28 mm × 0.2 mm** of silicon. It was designed by one person, fabricated on an open-source process, and **six independent testers have confirmed it working on real silicon**.



The actual silicon. Every line in this image is a real metal track on the die — magenta and cyan are different routing layers, placed by open-source tools. One Tiny Tapeout tile is about **160 × 100 µm**; this design occupies **8 × 2 of them**. The short edge is roughly two to three human hairs wide.

RV32IMA	30 MHz	16+8 MiB	16 tiles	130 nm
32-BIT RISC-V MUL/DIV + ATOMICS	SYSTEM CLOCK 34.5 MHz MAX TESTED	EXTERNAL SPI FLASH + PSRAM VIA QSPI PMOD	8 × 2 ARRANGEMENT ≈ 0.26 mm² TOTAL	SKYWATER OPEN PDK EFABLESS 2404C SHUTTLE

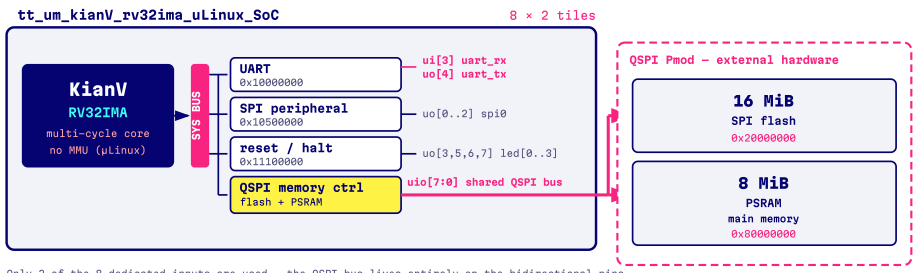
WHAT IT IS

Tiny Tapeout puts dozens of designs onto one shared chip, so anyone can have real silicon without buying a chip of their own. A tile holds **about a thousand logic gates**; most projects are counters or small state machines. **Hirosh Dabui submitted a computer.**

KianV is a multi-cycle RV32IMA core — a 32-bit RISC-V processor with hardware multiply, divide and atomic instructions — optimised for area rather than speed. There is no MMU — this is the no-MMU µLinux build — which is much of why it fits. Alongside the CPU sit a UART, an SPI peripheral, and a controller driving a **shared QSPI bus** out through the bidirectional pins to an external Pmod holding 16 MiB of flash and 8 MiB of PSRAM. That PSRAM is the machine's entire main memory.

On reset the CPU begins executing at `0x20100000` — one megabyte into the flash — where a bootloader hands off to a Linux kernel built for `riscv32`. About two minutes later, at 30 MHz, you get a shell.

BLOCK DIAGRAM



Only 2 of the 8 dedicated inputs are used — the QSPI bus lives entirely on the bidirectional pins. Instruction fetch and data traffic both cross that one external QSPI bus.

FLASH IMAGE LAYOUT → CPU RESETS TO 0x20100000

0x100000	bootloader.bin	Bootloader
0x180000	kianv.dtb	Device Tree Blob
0x200000	Image	Linux kernel + rootfs

MEMORY MAP

ADDRESS	SIZE	PURPOSE
0x10000000	0x14	UART peripheral
0x10500000	0x14	SPI peripheral
0x11100000	0x04	Reset / HALT control
0x20000000	16 MiB	SPI flash
0x80000000	8 MiB	PSRAM

CPU CONTROL REGISTER

ADDRESS	NAME	WRITE VALUE
0x11100000	CPU_RESET	0x7777 → reset CPU 0x5555 → halt CPU

PERIPHERAL REGISTERS

UART

ADDRESS	NAME	DESCRIPTION
0x10000000	UART_DATA	Write to transmit, read to receive
0x10000005	UART_LSR	Line status register
0x10000010	UART_DIV	Baud rate clock divider

SPI

ADDRESS	NAME	DESCRIPTION
0x10500000	SPI_CTRL0	Peripheral control
0x10500004	SPI_DATA0	Data
0x10500010	SPI_DIV	Clock divider

PINOUT — 8 IN / 8 OUT / 8 BIDIRECTIONAL

#	INPUT	OUTPUT	BIDIRECTIONAL
0	–	spi_cen0	ce0 flash
1	–	spi_sclk0	sio0
2	spi_sio1_so_miso0	spi_sio0_si_mosi0	sio1
3	uart_rx	led[0]	sck
4	–	uart_tx	sd2
5	–	led[1]	sd3
6	–	led[2]	cs1 psram
7	–	led[3]	always high

IT REALLY BOOTS — CONFIRMED INDEPENDENTLY, ON REAL SILICON

```
# uname -a
Linux (none) 6.8.0-rc1 #7 Wed Dec 25
08:58:21 UTC 2024 riscv32 GNU/Linux
- CarlFK, reporting from TT06 silicon
```

MATT VENN — TINY TAPEOUT
"I just booted uLinux on Tiny Tapeout 6. A very easy to use flashing tool by Uri Shaked made it easy to test Hirosh's!"

URISH
"Managed to boot Linux! … set the clock speed to 31.5 MHz, and got a UART console" — at 3,500,000 baud.

DDFL0417
"I managed to boot Linux and able to interact with the console via a USB to UART adapter. Then I test run the program /root/human_shader."

MOLE99
"Flashed the files via Tiny Tapeout Flasher and started the SoC at 30MHz, works without issues."

MITHRO
"Followed instructions … and was able to boot Linux!"

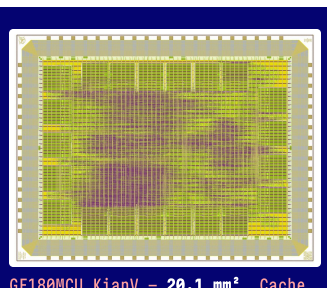
SPLINEDRIVE ♕ — THE DESIGNER
"First time right, time to market!"

WHAT HAPPENED NEXT

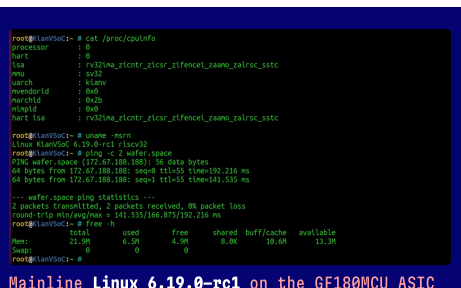
The TT06 tile proved the core worked in silicon. That was the beginning, not the end. The design was re-run on four further Tiny Tapeout shuttles — **TTSky25a, TTIHP25b, TTSky25b** and **TTIHP26a** — then scaled up and taped out as a **full-size ASIC through wafer.space** on the GlobalFoundries 180 nm PDK, using the open-source LibreLane flow: a **20.1 mm² die**, about 65% core utilisation, an estimated **13 mW**.

That chip is a far bigger machine: an **Sv32 MMU**, instruction and data caches, 4-way TLBs, an SDRAM controller feeding **32 MiB**, Ethernet and an SD card. It runs **mainline Linux**, µLinux and XV6. Booting it in gate-level simulation took **a solid month** of compute to reach a login prompt.

> "My KianV RISC-V ASIC is booting mainline Linux with SDRAM. Everything was developed from scratch."
> — HIROSH DABUI, AUGUST 2026 · BOARD DESIGN AND BRING-UP BY MACHDYNE



GF180MCU KianV — **20.1 mm²**. Cache SRAM blocks surround the central core, peripherals and IO around the rim.



Mainline **Linux 6.19.0-rc1** on the GF180MCU ASIC — note the Sv32 MMU, and a successful ping of wafer.space from the chip itself. **This is the follow-up chip, not the TT06 tile:** the TT06 design has no MMU.

KianV µLinux SoC designed by Hirosh Dabui (splinedrive). Tiny Tapeout: **Matt Venn**, **Uri Shaked** and **Sylvain Munaut**. TT06 was fabricated on the Efabless 2404C chipIgnite shuttle using the SkyWater 130 nm open-source PDK — launched 30 January 2024, submissions closed 19 April 2024, silicon received November 2024. Boot reports by urish, mithro, ddfl0417, CarlFK, splinedrive and mole99. The follow-up GF180MCU ASIC was taped out through wafer.space.

SOURCES: tinytapeout.com/chips/tt06 · TT06 datasheet · FOSSi Foundation El Correo Libre #82 & #99 · Crowd Supply wafer.space GF180MCU Run 2

TT06 PROJECT · KianV SOURCE · wafer.space