Tiny Tapeout: A Shared Silicon Tapeout Platform Accessible To Everyone

Matt Venn
Tiny Tapeout, YosysHQ
Email: matt@tinytapeout.com

***Index Terms*—ASIC, Multi Project Chip, Open Source Silicon, Tiny Tapeout.**

I. Introduction

TINY TAPEOUT is a multi project chip platform that makes it easier and cheaper to get application specific integrated circuit (ASIC) designs manufactured.

Open source tools and process design kits (PDK [1]) are used so no licenses or non disclosure agreement (NDAs) are required. As the tools run on remote cloud servers no software needs to be installed locally on the user's machine. As long as the template structure is followed, however, Tiny Tapeout does support the use of proprietary tools.

Each Tiny Tapeout ASIC production run sees around 400 open source designs multiplexed to 24 general purpose input/output (GPIO) pins. After manufacture the resulting chip is mounted to a demonstration board for ease of testing. Each chip contains a copy of every design, which can be selected and tested in turn.

At the same time each participant submits documentation for their design, which used to create a printable datasheet [2] along with an online project index at TinyTapeout.com/runs/ [3]. The datasheet helps participants explore other designs on the chip in addition to their own.

By separating the cost of area on a silicon wafer and the finished physical chip, the Tiny Tapeout participant group is able to share the cost of chip packaging and circuit board manufacture while still being able to test and measure all the designs on the chip. For use in educational settings it is possible for multiple students to submit individual designs while sharing the finished chips and circuit boards, reducing the cost still further.

Each Tiny Tapeout tile (Fig. 1) is approximately $160 \times 100\,\mu m^2$. This provides enough room for around 1000 logic gates when built upon the SkyWater 130nm open source PDK. Multiple tiles can be interconnected to enable larger designs, while analog and mixed signal support is on the roadmap for the next shuttle.

Community engagement in Tiny Tapeout has been strong, with 756 designs submitted over the first five shuttles. A curated selection of projects is provided in section IX. An online chat server for participants has 1000 members with 1600 subscribers to the project's mailing list. Individuals submitting designs to Tiny Tapeout tend to self identify as hobbyists, students, and teachers, as shown in Fig. 2.

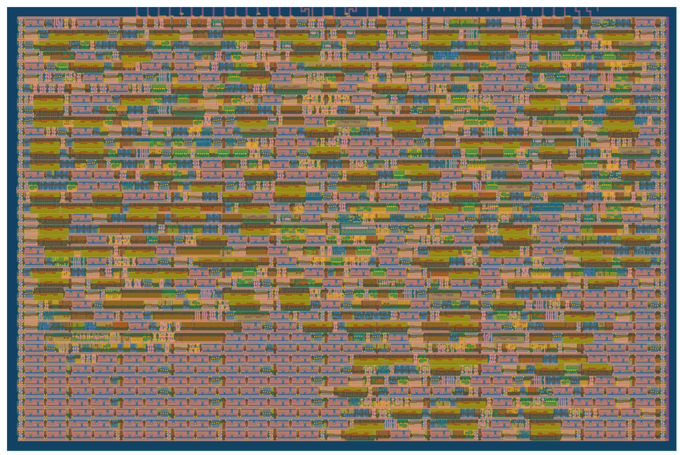

Fig. 1. A 2-D render of a single Tiny Tapeout tile.

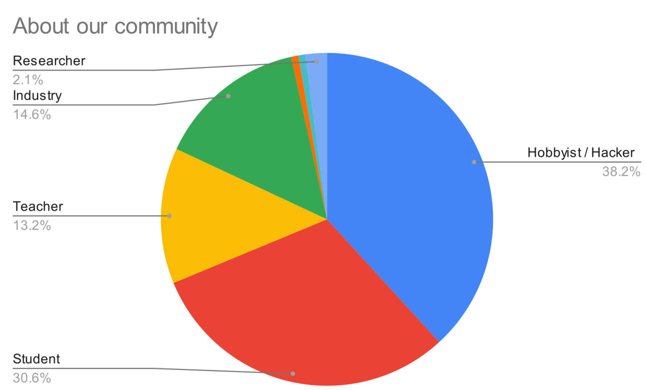

Fig. 2. Tiny Tapeout 04 participant self identification.

The first [4] Tiny Tapeout production run, which was provided as a free and experimental effort with a total of 152 designs, was submitted to the seventh Google-sponsored [5] lottery based multi project wafer (MPW) shuttle in September 2022. The next four shuttles combined a total of 582 designs, all sponsored by and manufactured through the Efabless [6] chipIgnite MPW service. Table I shows a summary of all Tiny Tapeout shuttle runs to date.

The rest of this paper will detail the Tiny Tapeout design flow, multiplexer evolution, circuit board design, the results of post production silicon testing, and the project's next steps.

TABLE I
TINY TAPEOUT SHUTTLE SUMMARY

Run	Launched	Shuttle	Designs	Delivery date	Architecture	Number of IOs	IO bandwidth	Analog support
TT01	2022-08-17	MPW7	152	n/a	Scan chain	16	5 kHz	no
TT02	2022-11-09	2211Q	165	2024-01-30	Scan chain	16	5 kHz	no
TT03	2023-03-01	2304C	249	2024-02-28	Scan chain inverted clock	16	10 kHz	no
TT04	2023-07-01	2309	143	2024-04-15	Mux	26	50 MHz	no
TT05	2023-09-11	2311	174	2024-05-12	Split Mux	26	50 MHz	no
TT06	2024-02-01	2404	TBD	2024-11-30	Split Mux	38	50 MHz	yes

II. DESIGN FLOW

Tiny Tapeout designs are primarily developed in the Verilog hardware description language (HDL) or Wokwi [7]. Wokwi is a web based visual schematic editor for hardware description, designed as an easier way for individuals with no prior HDL experience to get started. The Tiny Tapeout website [8] includes a basic Wokwi getting started guide, demonstrating how to use the tool to draw circuits, which is made available in English and Spanish.

The design flow has the participant create a GitHub [9] source code repository based on provided templates then add their ASIC design. This triggers automated tests and the generation of binary layout files in GDSII [10]. If all tests pass and the binary layout files are correctly generated, the design is then submitted to a quarterly shuttle for production in silicon.

The Tiny Tapeout GitHub templates [11] make use of GitHub Actions [12]—an automatic continuous integration system triggered every time the repository is updated. This reduces duplicated effort and makes it possible for Tiny Tapeout to support large numbers of participants without excessive technical overhead.

There are four main jobs in the continuous integration system:

1) GDS: installs OpenLane [13] and the SkyWater Sky130 [14] PDK, builds the binary layout files, and generates a summary of the design (Fig. 3). The summary includes utilization, standard cells used, a 2-D render (Fig. 1) and an interactive 3-D viewer (Fig. 4). This job can also optionally run a gate-level verification of the design.
2) Verification: installs the YosysHQ open source computer-aided design (CAD) suite, which includes many common electronic design automation (EDA) tools; uses iVerilog [15] and cocotb [16] to run included testbenches.
3) Documentation: generates a preview of the documentation.
4) Precheck: runs design rule check (DRC) tests to ensure the design can be integrated into the multi project chip.

Successful GDS, Documentation, and Precheck job completion are all required for a design to be submitted to a shuttle for production. Verification is optional but highly encouraged. Submissions designed in Wokwi are able to make use of its integrated truth table testing system [17].

Routing stats

Utilisation (%)	Wire length (um)
51.7	16572

Cell usage by Category

Category	Cells	Count
Fill	decap fill	1145
Combo Logic	o21ai nand3b a221o o31a or2b a21o a21bo nor2b o31ai a41o a21oi a211o o21ba o21a o21bai a2bb2o or3b o221a a31o a22o o32a o22a a32o a2111o and2b and3b or4b or4bb o211a a2111oi o2bb2a a211oi and4bb o211ai a22oi a31oi o311a	249
Tap	tapvpwrvgnd	246
Flip Flops	dfxtp	146
Buffer	buf clkbuf	127
AND	and2 a21boi and3 and4	97
Misc	dlygate4sd3 dlymetal6s2s conb	84
OR	or3 xor2 or2 or4	81
NOR	xnor2 nor2 nor3 nor4	64
NAND	nand2 nand3 nand4 nand2b	52
Inverter	inv	37
Multiplexer	mux2 mux4	9
Diode	diode	1

947 total cells (excluding fill and tap cells)

Fig. 3. A summary table from the GDS continuous integration job.

Fig. 4. The interactive 3-D viewer.

While the Tiny Tapeout continuous integration system can be run entirely in the user's web browser, it is also possible to install a local copy of the tools [18] on a participant's computer. Locally installed tools can help to reduce the time between design iterations, especially for the test and verification jobs.

III. SCAN CHAIN ARCHITECTURE

Tiny Tapeout started as an experiment in fitting as many designs as possible into the 10 mm^2 available on the Google lottery shuttles (Fig. 5). To rapidly prove the concept, initial designs were based on a scan chain architecture to simplify testing. Each Tiny Tapeout 01 design has eight inputs and eight outputs. Clock and reset signals were optional and not treated specially. The chain was formed of scan flops [14], a

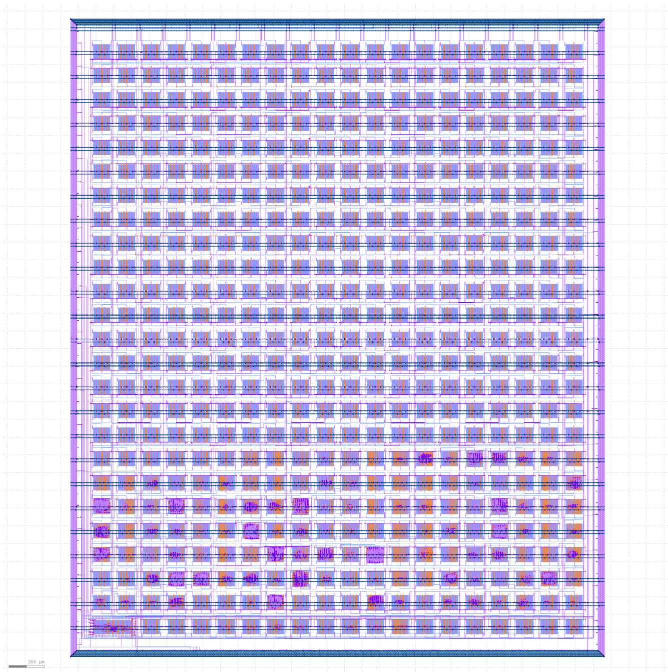

Fig. 5. 500 designs connected in a chain for Tiny Tapeout 01; the scan chain driver can be seen in the lower left corner.

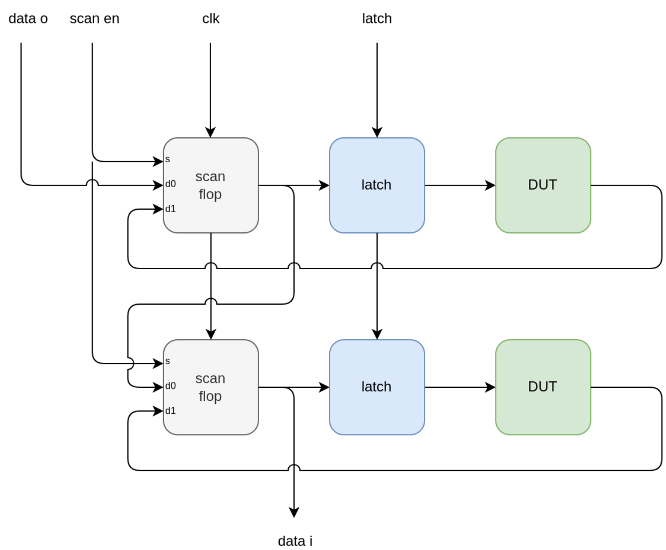

Fig. 6. A simplified view of two Tiny Tapeout 01 designs in the scan chain.

type of flip flop with an multiplexer integrated at its input. An example showing a two design scan chain is shown in Fig. 6.

Each design sends data into the secondary input of the scan flop and receives its own input from the output of the flop via a latch. The chain is built [19] by sending data from the output of the previous scan flop into the primary input of the next scan flop. This arrangement allows the loading of data into any of the designs, followed by the capturing of the output and its clocking through the rest of the chain to the overall chain output.

While relatively easy to implement, a scan chain architecture has a downside: high latency. As more designs are

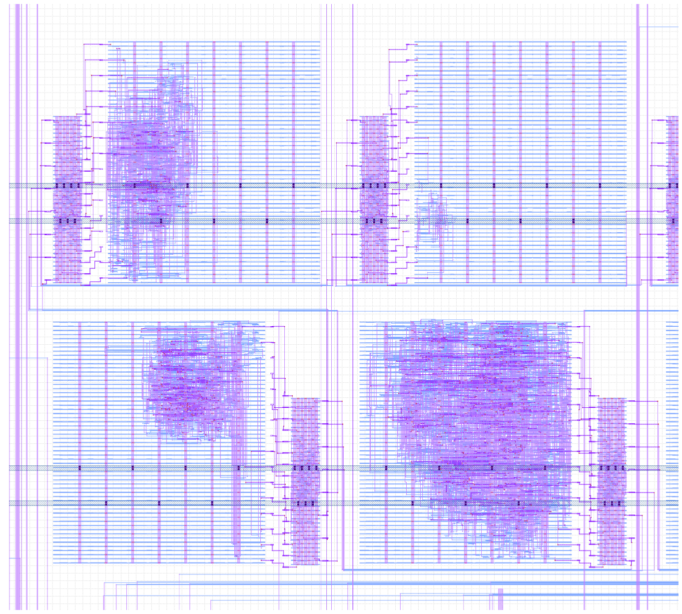

Fig. 7. Tiny Tapeout 02 designs, showing the discrete scan chain blocks.

added to the chain it takes longer to send and receive data through it. For example: assuming a 50 MHz scan chain clock with 250 designs each having eight inputs and eight outputs, the maximum refresh rate of the resulting chain is $50\,\mathrm{MHz}/(8 \times 250) = 25\,\mathrm{kHz}$.

The Tiny Tapeout 01 scan chain was embedded into each design, meaning a user could unintentionally remove it and break the chain. This risk was mitigated with a formal [20] equivalence check which proves the chain was present in each submitted design. For Tiny Tapeout 02 and Tiny Tapeout 03 the scan chain was separated into a discrete macro block which participants cannot modify.

Another concern with the scan chain design was hold violations, due to the large number of serially connected flops and potentially large clock skews over long signal wires. This was mitigated by reclocking the output data with a negative edge (negedge) flop, providing substantially more hold margin.

Following static timing analysis (STA) it was discovered that the clock duty cycle could change substantially due to the 500 sequential clock drivers in the chain. Depending on the clock buffers and capacitance between each design the clock duty cycle could either increase or decrease, with this effect accumulated over the chain.

For Tiny Tapeout 01 and Tiny Tapeout 02 each design used two clock buffers, with the internal flops driven after the first buffer. Tiny Tapeout 03 used inverting clock buffers, with only one between the clock input and output. Fig. 8 shows a comparison between the TT02 and TT03 clock buffer designs. By inverting the clock between each design any asymmetry in the clock pulse is evenly spread across the negative and positive cycles.

The verification effort [21] for the scan chain was broad and included a community review, register transfer level (RTL) and gate level (GL) simulation, formal verification [22], STA, layout vs. schematic (LVS), DRC, and device level static

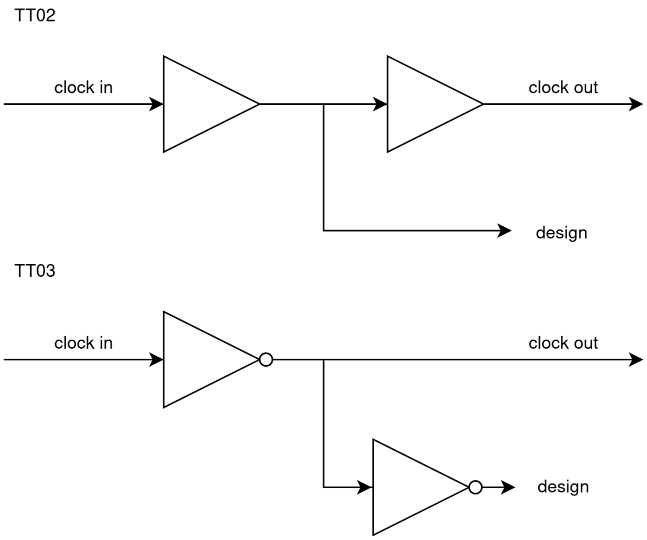

Fig. 8. The Tiny Tapeout 03 architecture buffers the output from the clock network into each design. Clock polarity is alternated between designs to minimize asymmetry between positive and negative cycles.

verification [23].

IV. Circuit Boards

After manufacture, Tiny Tapeout chips are mounted onto small carrier boards with 0.1 inch pin headers. These carriers allow people with limited surface mount technology (SMT) assembly experience to build their own demonstration boards around the chips.

The carrier fits onto the demonstration board shown in Fig. 9. The demonstration board is designed primarily for ease of use by beginners, though offers enough flexibility for power users. As all signals are below 50 MHz, no special layout was needed in the design of the demonstration board.

The demonstration board provides:

- USB Type-C as a power connection,
- 1.8 V and 3.3 V power supplies for core and IO,
- 20 MHz oscillator,
- Buttons for reset and single step clock,
- An eight way DIP switch for inputs,
- A nine way DIP switch for design selection,
- A seven segment LED display for the outputs,
- Headers for all IO, including two standard Digilent ports (PMOD),
- A header to select the internal clock or to provide one externally,
- A header to select an internal or external scan chain driver,
- A header to engage an automatic clock divider in input pin zero.

V. Scan Chain Silicon Results

TT02 chips were received in October 2023, 11 months after the chips were submitted for manufacture on Efabless chipIgnite 2211Q. The chips were tested for the first time in public on a livestream [25]. The chain was validated, and a few of the designs were shown to be working.

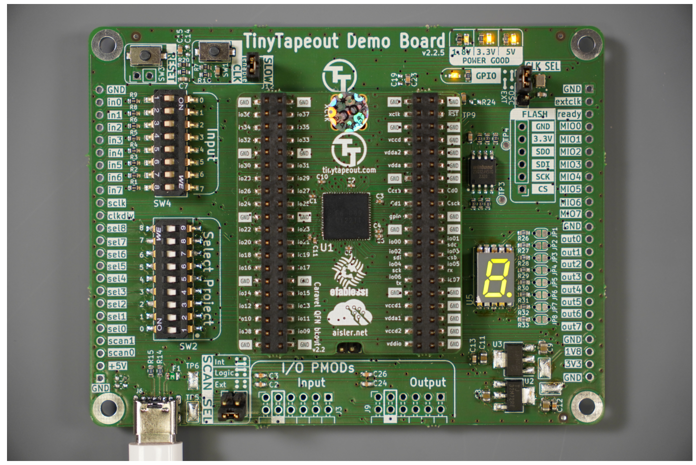

Fig. 9. The demonstration board, which has been recorded as Certified Open Source Hardware ES000040 [24].

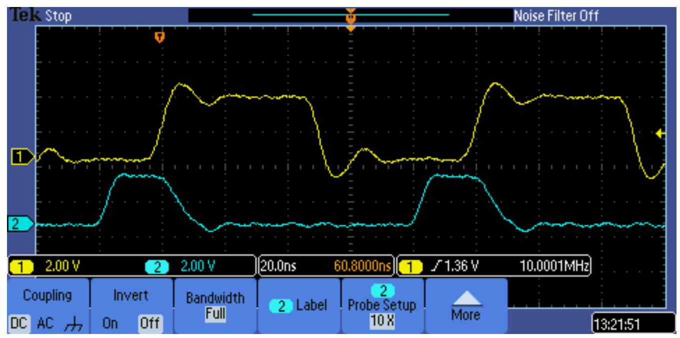

Fig. 10. Measurement from TT02 silicon, with input clock in yellow and the distorted output clock in blue.

In the following days another 30 designs were tested and shown to be working.

After measuring the clock asymmetry (Fig. 10) and maximum frequency it was decided to run the production boards with a 20 MHz oscillator, resulting in a 10 MHz scan chain clock and a 5 kHz IO update rate.

Some designs didn't function as expected, which in most cases was due to faults in the submitted design.

As well as 82 Verilog designs, 64 used the Wokwi graphical editor, 6 used alternative HDLs like VHDL, Amaranth [26] and Chisel [27]. Some Wokwi designs using combinational logic in clock paths (Fig. 11) worked in simulation but failed in hardware. This was due to the lack of timing data in the simulation, and wasn't detected by STA because the clock paths were not known. A detailed analysis has yet to be carried out.

At the time of writing, PCBs are in production and are expected to ship to customers by the end of January 2024.

Tiny Tapeout 3 silicon was received in January 2024, and the updated scan chain shows a more symmetric (Fig. 12) output clock at the end of the chain. This will allow a faster scan chain clock, resulting in a faster update frequency.

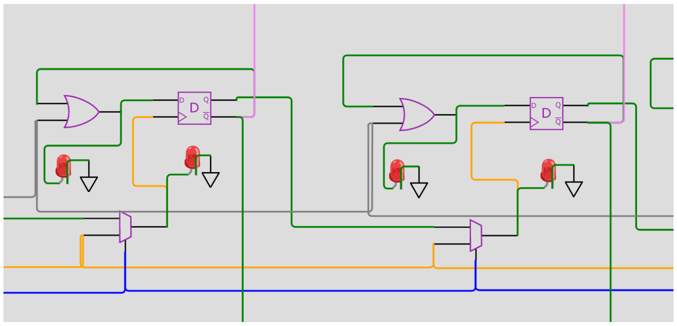

Fig. 11. Combinational logic in the clock path of one of the failed designs.

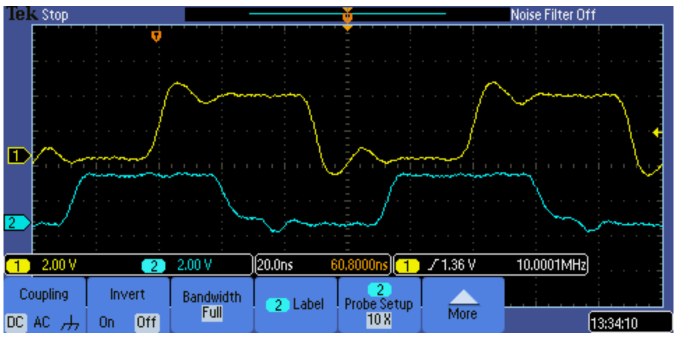

Fig. 12. Measurement from TT03 silicon.

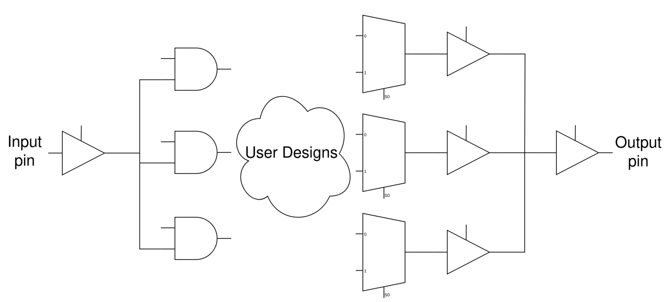

Fig. 13. Simplified diagram of the multiplexer architecture.

VI. Beyond the Scan Chain

The biggest limitation of the scan chain based architecture was the IO bandwidth and latency. A new architecture was needed for TinyTapout 4, so proposals were gathered from the community. An online video call was held and the 10 proposals discussed. The winning design was a fairly straightforward multiplexer design shown in Fig. 13. It was chosen as the simplest to implement while providing the most benefit in terms of additional IOs and higher bandwidth.

The physical layout (shown in Fig. 14) consists of a central controller connected up and down to two vertical spines. Twenty-four horizontal muxes connect to the spine with each supporting 16 designs. This allows up to 384 separate single tile designs. Multiple tile designs were also enabled, allowing a maximum project size of 8×2 tiles or $1359 \times 225\,\mu m^2$—around 20 000 logic cells. Table II shows the key differences between TT03 and TT04.

Another major limitation of TT01 to TT03 was the small number of IO. The scan controller used 9 GPIOs to select the currently active design, which, while simplifying the demo board, wasted valuable pins. With TT04, the parallel design selection was dropped in favor of a serial protocol. The extra pins were then used as bidirectional pins, giving each design clock, reset, and 24 IO.

An invite-only experimental shuttle [28] was submitted with 32 designs to Efabless chipIgnite 2306C. Two of the designs included a power gate as a stepping stone to supporting analog and mixed-signal designs.

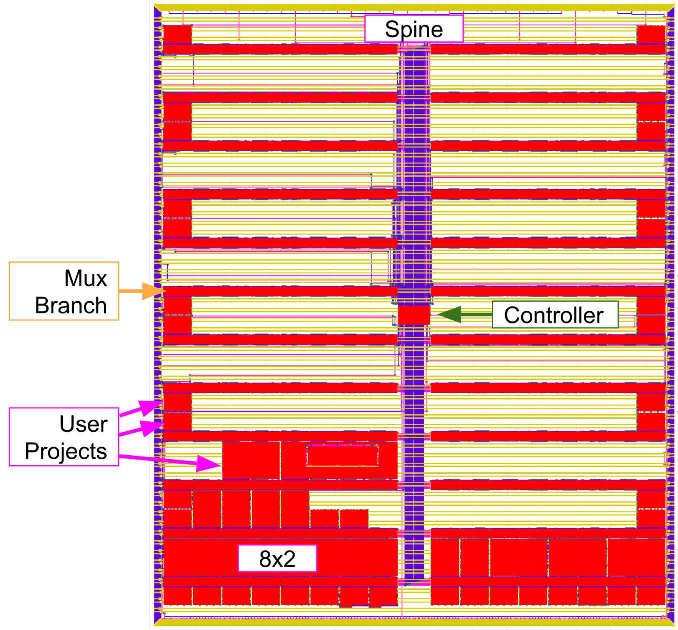

Fig. 14. The TT03.5 test design.

TABLE II
COMPARISON BETWEEN TT03 AND TT04

Parameters	TT03	TT04
Max clock speed	8 kHz	50 MHz
Max design size	$150 \times 170\,\mu m^2$	$1359 \times 225\,\mu m^2$
Input pins	8	10
Output pins	8	8
Bidirectional I/O pins	None	8
Custom GDS file	✗	✓

VII. Multiplexer Silicon Results

After silicon was received, the worst round trip latency was measured to be 20 ns as shown in Fig 15 and 16. Some designs have been validated, including a VGA clock project (Fig. 17) that takes advantage of the new higher speed IO.

The overhead of multiplexing multiple tiles makes power consumption of the infrastructure only a minor concern at this point. We don't have a direct comparison of the power consumption impact of the multiplexer vs. the scan-chain architecture. The motivation of the multiplexer approach is to substantially increase the bandwidth of the IOs.

The new chip pinout and serial design selection required a new demonstration board (Fig. 19) that included an easy way to select the design. The RP2040 microcontroller was chosen as a co-processor as it allows:

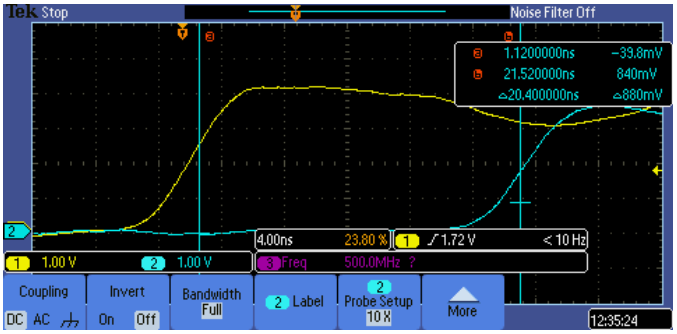

Fig. 15. Round trip latency on a rising edge of about 20 ns.

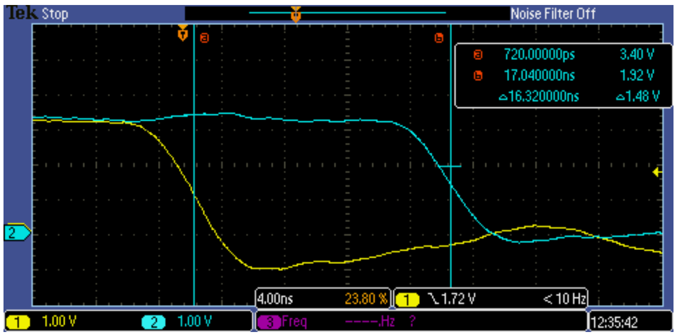

Fig. 16. Round trip latency on a falling edge of about 16 ns.

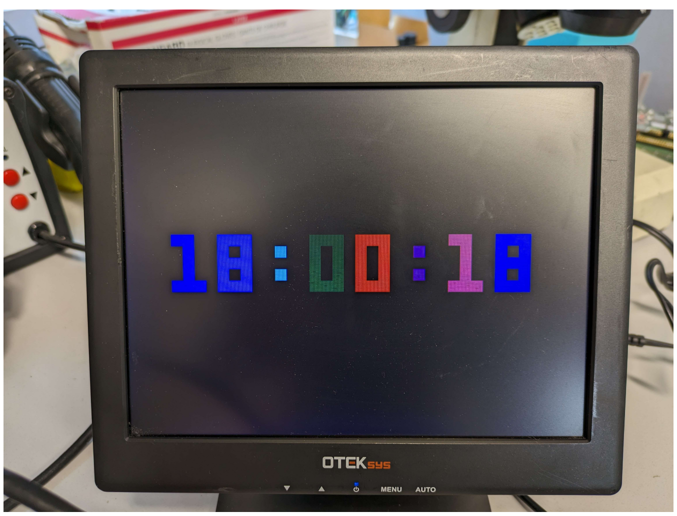

Fig. 17. VGA clock design running on TT03.5 silicon.

```
enabling design tt_um_test by sending 102 [0b01100110] pulses
design repo https://github.com/TinyTapeout/tt03p5-test @ 434c5d508d20053bea346881a61355f87ea1ca91
0 0 0 0
1 0 0 0
0 1 0 0
1 1 0 0
0 0 1 0
1 0 1 0
0 1 1 0
1 1 1 0
0 0 0 1
```

Fig. 18. A MicroPython program [30] enabling a design, clocking it, and printing the results.

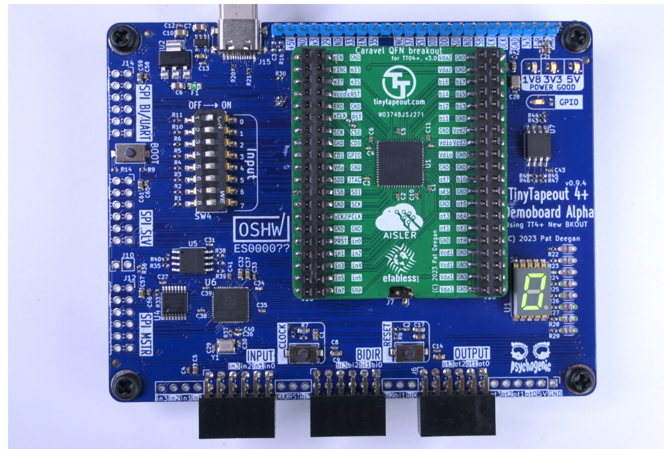

Fig. 19. The TT04+ demo board [31].

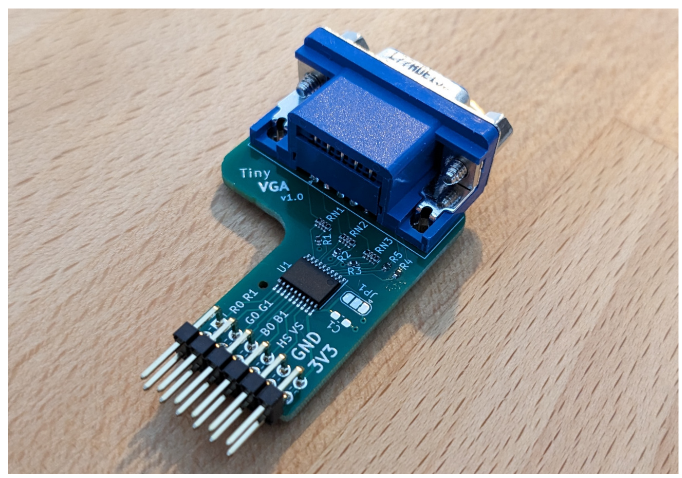

Fig. 20. A user-contributed VGA output PMOD.

- Drag and drop firmware updates on any OS,
- Runs MicroPython [29], ideal for beginners to enable and test their designs (Fig. 18),
- External memory emulation via PIO and DMA.

An additional PMOD expansion port was added for the bidirectional pins, and the community has started to standardize on pinouts [32] making it easier to test each other's designs. A new repository was created to house user-contributed PMODs [33], for example the VGA PMOD shown in Fig. 20.

An additional set of 3 PMOD expansion ports were added that mixed input and outputs, allowing the most common standard PMODs to be used. For more information about the circuit board, pinout and PMOD support see the repository [31].

VIII. Improving the Multiplexer and Mixed Signal Support

TT05 split the mux into two parts to improve performance. Because of the split, the controller was also updated to mux between the two halves. Some other small changes to the mux including improving STA results by tweaking, adding or removing buffers. As each spine segment is now half as long, it will have half the capacitance. We expect to reduce the round trip latency to around 10 ns.

For TT06, the Caravel harness will be replaced by OpenFrame [34], an alternative harness provided by Efabless that

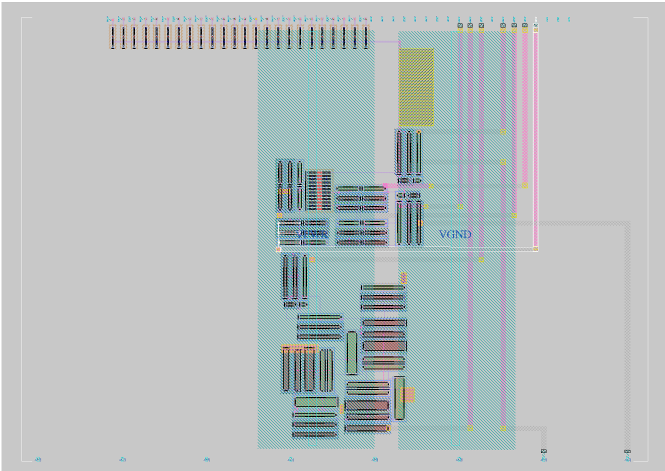

Fig. 21. Ring oscillator and DAC design submitted to TT05 with a transmission gate highlighted (auto-placed and auto-routed using an experimental analog P&R tool).

uses the same padring but removes the RISC-V coprocessor. This results in $5\,\text{mm}^2$ more space for user designs, and 12 more pins that will be used for analog signals.

For increased safety, all designs will be power-gated, which allows designers to take more risks or use custom flows.

Analog and mixed-signal designs will be enabled by adding an analog multiplexer based on transmission gates [35]. This allows up to 192 designs to share the analog pins between them. The transmission gates were tested as part of an experimental analog submission to TT05 shown in Fig. 21.

Noise coupling between analog and digital power domains is a critical concern. However, due to other limitations of our current setup like limited number of low bandwidth analog interfaces, we target educational low-medium performance analog and mixed-signal designs, where noise coupling is a lesser concern.

TT06 is planned to open for digital designs at the end of January 2024, for analog designs at the end of February, and to close on April 19th, 2024.

IX. Silicon Showcase

A small sample of the types of designs possible with Tiny Tapeout are listed below:

- Serial FPGA (Link)
- Synthesizable Digital Temperature Sensor (Link)
- 395 standard cells with mux (Link)
- FM transmitter with I2S input (Link)
- USB full speed - (Link)
- A Linux capable RISCV CPU - (Link)

An index of all submitted designs can be found at TinyTapeout.com/runs/ [3].

X. Acknowledgements

1) Uri Shaked for Wokwi development and lots more.
2) Sylvain Munaut for help with scan chain improvements and multiplexer design.
3) Mike Thompson and Mitch Bailey for verification expertise.
4) Jix for formal verification support.
5) Proppy for help with GitHub actions.
6) Maximo Balestrini for the amazing renders and the interactive GDS viewer.
7) Harald Pretl for advice and analog support.
8) The team at YosysHQ and all the other open source EDA tool makers.
9) Efabless for running the shuttles and providing OpenLane and sponsorship.
10) Tim Ansell and Google for supporting the open source silicon movement.
11) Pat Deegan for PCB development.
12) The Tiny Tapeout community for all their contributions.
13) The Zero to ASIC course community for all their support.

References

[1] M. Venn, "PDK, Zero to ASIC Course," https://www.zerotoasiccourse.com/terminology/pdk/, (accessed March 7, 2024).

[2] M. Venn, "Datasheet PDF, Tiny Tapeout Repository," https://github.com/TinyTapeout/tinytapeout-02/raw/tt02/datasheet.pdf, (accessed March 7, 2024).

[3] M. Venn, "Tiny Tapeout Runs," https://tinytapeout.com/runs/, (accessed March 7, 2024).

[4] M. Venn, "First Shuttle, Tiny Tapeout Runs," https://tinytapeout.com/runs/tt01/, (accessed March 7, 2024).

[5] E. Mahintorabi, J. Euphrosine, and A. Cunningham, "Announcing the GlobalFoundries Open MPW Shuttle Program, Google Open Source Blog," https://opensource.googleblog.com/2022/10/announcing-globalfoundries-open-mpw-shuttle-program.html, (accessed March 7, 2024).

[6] Efabless, "Efabless Home Page," https://efabless.com/, (accessed March 7, 2024).

[7] U. Shaked, "Wokwi home page," https://wokwi.com/, (accessed March 7, 2024).

[8] M. Venn, "Tiny Tapeout Home Page," https://tinytapeout.com/, (accessed March 7, 2024).

[9] GitHub, "GitHub Home Page," https://github.com/, (accessed March 7, 2024).

[10] M. Venn, "GDS2 File Format, Zero to ASIC Course," https://www.zerotoasiccourse.com/terminology/gds2/, (accessed March 7, 2024).

[11] U. Shaked, "Verilog Template, Tiny Tapeout Repository," https://github.com/TinyTapeout/tt06-verilog-template, (accessed March 7, 2024).

[12] GitHub, "GitHub Actions Documentation," https://docs.github.com/en/actions, (accessed March 7, 2024).

[13] Efabless, "OpenLane Documentation," https://openlane.readthedocs.io/en/latest/, (accessed March 7, 2024).

[14] S. P. Authors, "Skywater PDK Documentation," https://skywater-pdk.readthedocs.io/en/main/contents/libraries/sky130_fd_sc_hdll/cells/sdfxtp/README.html, (accessed March 7, 2024).

[15] S. Williams, "iVerilog, GitHub Repository," https://github.com/steveicarus/iverilog/, (accessed March 7, 2024).

[16] F. Foundation, "cocotb Home Page," https://www.cocotb.org/, (accessed March 7, 2024).

[17] P. Deegan, "Wokwi Automated Testing, Tiny Tapeout," https://tinytapeout.com/digital_design/wokwi_automated_testing/, (accessed March 7, 2024).

[18] U. Shaked, "Locally Installing the Tools," https://docs.google.com/document/d/1aUUZ1jthRpg4QURIIyzlOaPWlmQzr-jBn3wZipVUPt4, (accessed March 7, 2024).

[19] M. Venn, "Updating Inputs and Outputs of a Design, Tiny Tapeout Repository," https://github.com/TinyTapeout/tinytapeout-02/blob/tt02/INFO.md#updating-inputs-and-outputs-of-a-specified-design, (accessed March 7, 2024).

[20] J. Harder, "Tiny Tapeout Scan, GitHub Repository," https://github.com/jix/tinytapeout_scan, (accessed March 7, 2024).

[21] M. Venn, “Verification Documentation, Tiny Tapeout Repository,” https://github.com/TinyTapeout/tinytapeout-02/blob/tt02/VERIFICATION.md, (accessed March 7, 2024).

[22] YosysHQ, “SymbiYosys (SBY), YosysHQ Repository,” https://github.com/YosysHQ/sby, (accessed March 7, 2024).

[23] M. Bailey, “CVC, Verification Documentation, Tiny Tapeout Repository,” https://github.com/TinyTapeout/tinytapeout-02/blob/tt02/VERIFICATION.md#cvc, (accessed March 7, 2024).

[24] OSHWA, “Open Source Hardware Certification, Tiny Tapeout,” https://certification.oshwa.org/es000040.html, (accessed March 7, 2024).

[25] M. Venn, “TT02 Silicon is Alive!, Zero to ASIC Course Blog,” https://www.zerotoasiccourse.com/post/tt02-silicon-is-alive/, (accessed March 7, 2024).

[26] C. 'whitequark', “Amaranth Documentation,” https://amaranth-lang.org/docs/amaranth/latest/, (accessed March 7, 2024).

[27] ChipsAlliance, “Chisel Home Page,” https://www.chisel-lang.org/, (accessed March 7, 2024).

[28] M. Venn, “Tiny Tapeout 03p5, GitHub Repository,” https://github.com/TinyTapeout/tinytapeout-03p5, (accessed March 7, 2024).

[29] D. P. George, “MicroPython Home Page,” https://micropython.org/, (accessed March 7, 2024).

[30] M. Venn, “Demo Firmware Test Script, Tiny Tapeout Repository,” https://github.com/TinyTapeout/tt3p5-demo-fw/blob/main/tt3p5-test/test.py#L119, (accessed March 7, 2024).

[31] P. Deegan, “TT04+ Demoboard PCB, Tiny Tapeout Repository,” https://github.com/TinyTapeout/tt-demo-pcb, (accessed March 7, 2024).

[32] L. Moser, “Pinouts Specifications, Tiny Tapeout,” https://tinytapeout.com/specs/pinouts/, (accessed March 7, 2024).

[33] U. Shaked, “Awesome Tiny Tapeout PMODs, GitHub Repository,” https://github.com/TinyTapeout/awesome-tinytapeout-pmods, (accessed March 7, 2024).

[34] Efabless, “Caravel Openframe Project, GitHub Repository,” https://github.com/efabless/caravel_openframe_project, (accessed March 7, 2024).

[35] H. Pretl, “TT05 Analog Test, GitHub Repository,” https://github.com/iic-jku/tt05-analog-test, (accessed March 7, 2024).