Layer1Labs Silicon

LAYER1LABS SILICON INC. · DOCUMENT

ChronoFabric Measured Results

What ChronoFabric has measured on hardware, how each figure was obtained, and the limits published alongside it.

Version 1.0 · October 26, 2026 · Layer1Labs Silicon Inc.

This document collects the results ChronoFabric has published, in one place, each printed with the basis it was measured on. ChronoFabric is the hardware tier of the Layer1Labs architecture: it takes the same programs and constraint specifications that run in the ChronoHive software engine and enforces them on FPGA and hardware endpoints, with the same deterministic guarantees, delivered at hardware speed. It is available now as soft IP — a licensable hardware IP block — targeting Kria FPGA platforms.

Every figure here comes from the published measurement basis, stated once here and attached to each result where it appears: *measured on Kria FPGA evaluation boards (KR260/KV260) over 10-gigabit links — the beta platform.* Production-scale fabrics (100G/400G, ASIC) are the roadmap, not a measurement, and nothing in this document presents them as measured. Two sections (Sections 6 and 9) report a different kind of run, labeled wherever they appear: timing-budget equivalence runs, in which the per-packet timing budget was compressed to the cadences faster line rates would demand — in Section 6 on a board running at its native 1-gigabit rate, in Section 9 as a compressed interval on the two-board 10-gigabit link — validation against a target schedule, not a measurement at those line rates. Section 7 reports the same budgets verified in simulation, under that label.

See it live: the ChronoFabric page these results are published on — `https://layer1labs.ai/chronofabric/`

Documentation library: this document is published in the Layer1Labs documentation library — `https://layer1labs.ai/docs/`

Figures in this document are drawn from the results published at layer1labs.ai/chronofabric/ and the answers published at layer1labs.ai/faq/, both read on October 26, 2026, together with the April–May 2026 equivalence runs, the completed 24-hour table, supporting results of the same months, and the June 2026 simulation verification report, all from Layer1Labs' own test records. Each figure is cited to its basis where it appears.

© 2026 Layer1Labs Silicon Inc. All rights reserved. See the Notices at the end of this document.

Contents

1 How to read these results 3

2 The headline result 3

3 Delivery, run by run 4

4 Departure precision 4

5 Scheduling under load 5

6 Timing-budget equivalence at higher-rate cadences 6

7 Verified in simulation at the higher-rate budgets 7

8 The 24-hour runs 7

9 Equivalence at the 400G budget, on the two-board link 8

10 The results at a glance 9

11 Limits, stated plainly 10

12 What these results do not establish 10

Colophon 11

Notices 12

1 How to read these results

The results in this document carry three labels, and the differences matter:

- **Measured** — the figure comes from runs that were actually executed, on the hardware stated with it. The ChronoFabric page publishes its results under the heading "Measured on Hardware," and the results in Sections 2–5 and 8 are of this kind, each printed with its run basis.
- **Timing-budget equivalence** — the runs in Sections 6 and 9 were also executed on hardware, but what they validate is narrower: the timing mechanism was checked against its target schedule at per-packet budgets compressed to the cadences faster line rates would demand — in Section 6 on a board whose link ran at its native 1-gigabit rate throughout, in Section 9 as a 25-microsecond interval on the two-board 10-gigabit link. An equivalence result is not a measurement of jitter at the faster rates, and it is not data transport at those rates; the document states that boundary wherever the result appears.
- **Verified in simulation** — the results in Section 7 come from simulation: the gate's own design was exercised in a software simulator at the timing budgets the faster rates would impose. A simulation result verifies how the design behaves at those budgets; it is not a run on a board, and it is not a physical link at those rates. The document keeps it distinct from both labels above — it is a second, independent check on the same budgets, reported as what it is.

No ChronoFabric figure on the website carries a modeled or demonstrated label, and this document invents none: there are no modeled ChronoFabric results. The roadmap items the page names — production-scale fabrics (100G/400G, ASIC), embedded chiplets and merchant silicon, dedicated production hardware — are not results, and appear here only where they are labeled roadmap.

Three standing rules govern everything that follows. First, **the basis travels with the figure**: the boards, the link, and the run are stated every time a result is stated. Second, **no stacking**: the results come from separate runs, and no figure is merged with another to make a combined claim. Third, **no generalizing from one configuration**: where a result was measured on one board in one configuration — each of the 24-hour runs in Section 8 is the case in point — it is reported as exactly that, never as a platform-wide figure.

2 The headline result

21,000+ scheduled packets delivered — zero missed, zero dropped, a 100% delivery rate.

Measured in validation runs between two Kria FPGA boards over a 10-gigabit Ethernet link, including a continuous 10,000-packet endurance run.

This is the result the ChronoFabric page leads with, and it is the right one to lead with: a deterministic scheduler's first promise is that work scheduled for a tick leaves on that tick. Across more than 21,000 scheduled packets, ChronoFabric missed no departure and dropped no packet — and the total includes a continuous 10,000-packet endurance run, so the figure is a sustained result, not a short burst sampled at its best.

© 2026 Layer1Labs Silicon Inc.

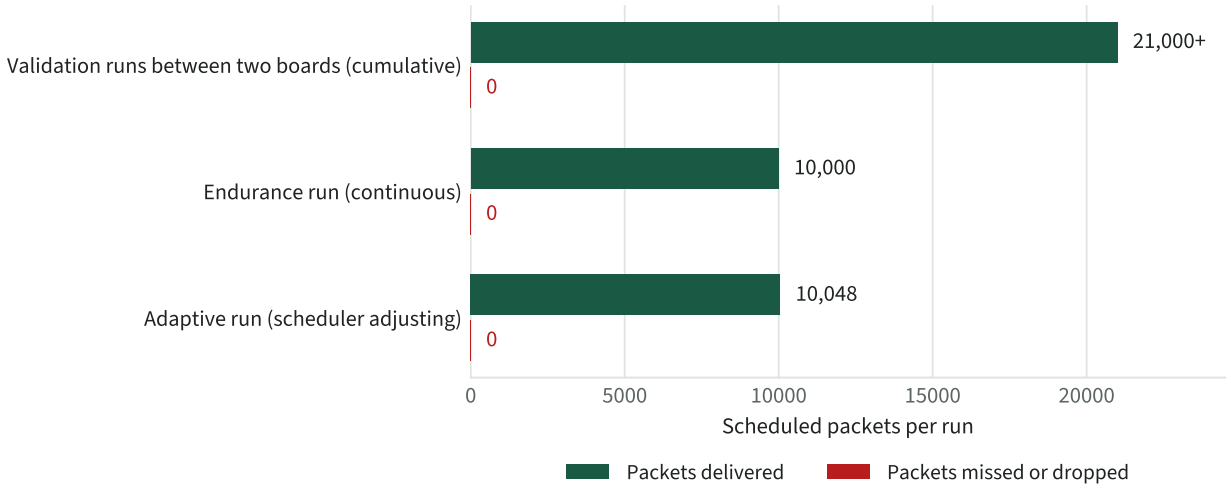


Figure 1: Packets delivered and packets missed or dropped, by run. Measured between two Kria FPGA boards over a 10-gigabit Ethernet link. The validation total is cumulative across the validation runs and includes the endurance run. Source: layer1labs.ai/chronofabric/, "Measured on Hardware."

3 Delivery, run by run

The delivery record has three parts. The **validation runs** between the two boards account for the cumulative total: more than 21,000 scheduled packets, zero missed departures, zero dropped. The **endurance run** is the sustained case inside that total: 10,000 packets in one continuous run, every one delivered. The **adaptive run** (Section 5) delivered all 10,048 of its packets while the scheduler was adjusting its own timing — delivery held not only in steady conditions but while the schedule itself was changing.

Burst groups, inside their windows. In burst tests, complete groups of packets were in place ahead of their scheduled instant, and every packet departed inside its window: 69 of 69 in a 100 KB group — all inside an 85-microsecond window in the window-enforcement run — and 700 of 700 in a 1 MB group, the larger burst finishing with about 115 microseconds of its 1-millisecond window to spare.

4 Departure precision

Delivery is the first promise; precision is the second. Packets are programmed to leave at an exact tick, and the page reports how closely the departures matched the program:

- **Median and p99 exactly on the tick.** Across 2,000-packet runs, the median and 99th percentile departure interval were exactly the programmed 1 ms — 1,000,000 ns.
- **Jitter as low as 61.1 ns.** The spread of departure intervals (standard deviation) was as low as 61.1 ns — roughly 160× tighter than the same boards scheduling from the CPU.
- **Two boards, one tick.** In cross-board tests, two ChronoFabric boards fired their packet bursts at exactly the same hardware tick — a measured collective departure spread of 0 ns.
- **A faster cadence, across two boards.** Coordinated scheduling across the two boards also held at a faster cadence: departure spread measured 0.0 ns at a 100-microsecond interval, as at 1 ms.

© 2026 Layer1Labs Silicon Inc.

- **Arrivals at the receiving board.** In the full transmit-and-receive dataset (10,000 packets, May 2026), arrivals at the receiving board held a trimmed standard deviation of 6.2 ns in quiet conditions and 6.3 ns with the processors under stress — inside one 10-nanosecond hardware clock tick. Ungated sending under the same stress degraded to millisecond-scale arrival spread.

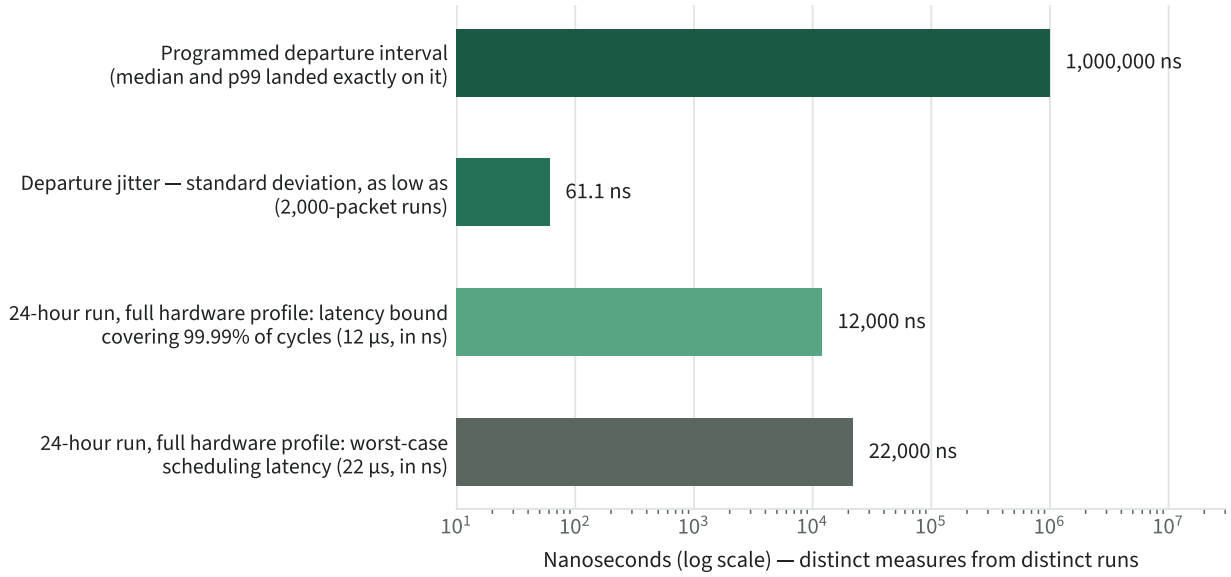


Figure 2: The published timing figures on one scale. These are distinct measures from distinct runs — the programmed interval and the departure jitter from the 2,000-packet precision runs, the latency figures from the full-hardware-profile 24-hour run of Section 8 — printed together to show the orders of magnitude involved, and never combined into a single claim. Source: layer1labs.ai/chronofabric/, “Measured on Hardware.”

5 Scheduling under load

The runs in this section test the scheduler where a software scheduler is weakest: with the host processor fully busy, under competing traffic, across processor power states, and with the schedule itself in motion.

Saturation. With all four CPU cores held at 100% load, ChronoFabric’s scheduled departures still landed on the programmed tick — median and p99 exact. On the same boards, CPU-scheduled departures degraded to millisecond-scale jitter under the same load. The reason is structural: the hardware scheduler issues no per-packet work to the CPU at all, so a saturated processor has nothing to delay.

Adaptive scheduling. In a 10,048-packet run, the fabric’s adaptive scheduler adjusted its own timing 53 times in response to on-chip constraint signals — and still delivered every packet: 10,048 of 10,048, zero misses. Adaptation did not cost delivery.

Under a packet flood. Under a flood of 50,000 competing packets, the scheduled flow still departed on its programmed 1-millisecond tick at the median and at the 99th percentile, hitting its target for 99.82% of packets. The two scheduled flows in the test added 3.8 mW of power draw.

Across power states. Timing held across processor power states. With the board’s processors in performance, power-saving, and partially-offline states, and while cycling under stress, scheduled departures hit their target in 99.94% to 100% of packets, with median and 99th-percentile intervals exactly the programmed 1 ms in every state.

© 2026 Layer1Labs Silicon Inc.

The ring step. In a two-board ring test — the step pattern a collective operation repeats — scheduled steps held a median and 99th-percentile round trip of exactly 1 ms across 999 steps, while unscheduled steps took about 645 microseconds longer per step on average (median round trip about 1.65 ms).

6 Timing-budget equivalence at higher-rate cadences

A scheduler that holds its timing at one line rate should also hold the tighter per-packet budgets that faster line rates demand: the faster the link, the less time there is between one packet's deadline and the next. In April 2026, timing-budget equivalence runs tested exactly that. On a single Kria KR260 board, using an on-board loopback path and the board's native 1-gigabit Ethernet, the per-packet timing budget was compressed to the cadences faster line rates would demand:

Equivalent line rate	Per-packet timing budget
100G-equivalent	10,000 ns
400G-equivalent	2,500 ns
800G-equivalent	1,250 ns
1.6T-equivalent	620 ns

At each level, 5,000 packets were run, first in quiet conditions and again under CPU stress. At every level, in both conditions, the schedule was hit exactly: a target-hit rate of 100.0%, zero stale targets, and a standard deviation of 0.0 ns against the target schedule.

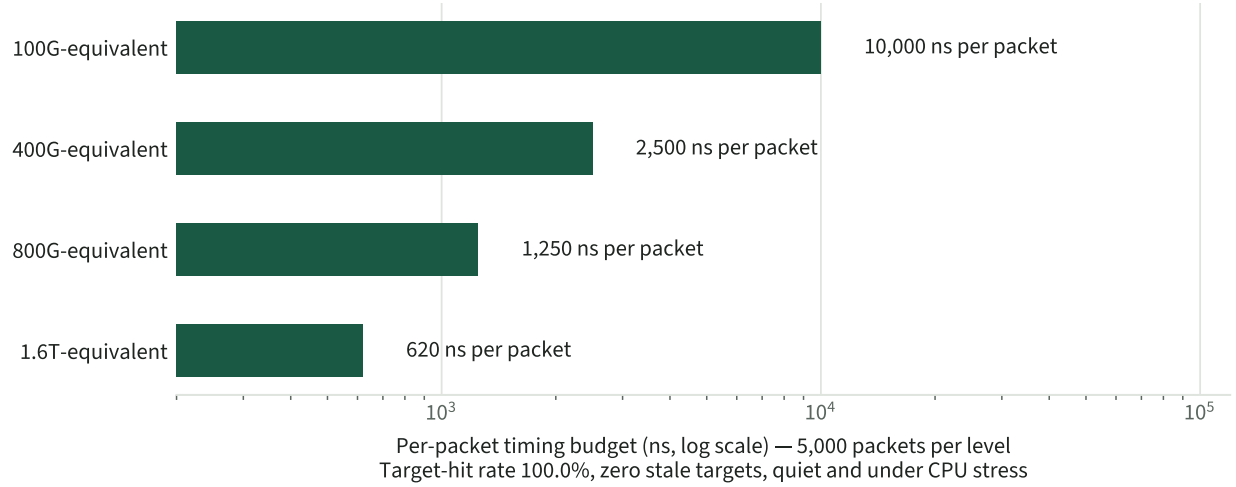


Figure 3: The timing-budget equivalence levels: the per-packet budget at each compressed cadence. One Kria KR260 board, on-board loopback, native 1-gigabit Ethernet; 5,000 packets per level, in quiet conditions and under CPU stress; target-hit rate 100.0% and zero stale targets at every level. These runs validate the timing mechanism against the target schedule at those budgets — they are not a measurement of jitter at those line rates, and not data transport at 100G, 400G, 800G, or 1.6T.

The boundary of this result is stated in the same breath as the result itself, because it is part of the result: **these runs validate the timing and coordination mechanism against the target schedule at those budgets. They are not a measurement of jitter at those line rates, and they are not data transport at 400G, 800G, or 1.6T** — the board's link ran at its native 1-gigabit rate throughout. What they establish is that the timing mechanism does not depend on having a full millisecond between packets: it holds its

schedule at the cadence a 1.6T link would impose, with the processor quiet and with the processor under stress alike. The same budgets were also verified a second way — in simulation, against the gate design itself — in Section 7.

7 Verified in simulation at the higher-rate budgets

The budgets of Section 6 were also verified a second way: in simulation, against the gate design itself, before any high-rate hardware existed to run them on. The gate was exercised in a cocotb regression suite — cocotb is the standard open-source Python framework for verifying chip designs, and here it drove the gate's actual design description in the GHDL simulator — at simulated link-rate timing budgets from 1 gigabit through the 1.6-terabit equivalent:

Simulated link rate	Per-packet timing budget
1G-equivalent	1,000,000 ns
10G-equivalent	100,000 ns
100G-equivalent	10,000 ns
400G-equivalent	2,500 ns
800G-equivalent	1,250 ns
1.6T-equivalent	620 ns

At every simulated rate, the gate held its precision: departures landed on exactly the target tick, with a departure spread of zero at all rates. At the tightest budget — the 620-nanosecond, 1.6T-equivalent cadence — the suite recorded zero missed gates, and burst timing held under the same compression. The suite's formal verification report (June 2026) records 46 tests in all: 45 passing, one skipped — a test that had been relocated to another part of the suite — and none failing.

The boundary, again in the same breath: **this is verification of the design in simulation at those budgets. It is not a physical link at 100G, 400G, 800G, or 1.6T, and it is not a measurement of silicon** — no board carried these rates. What it establishes is that the gate's design, as written, behaves correctly at the tightest budgets the faster rates would impose: the rates are not an untested extrapolation of the design. The same budgets were separately exercised on hardware in Section 6, where the gate also held its schedule exactly. The two legs agree; they are reported separately because they are different kinds of evidence, and neither is merged into the other.

8 The 24-hour runs

Short runs show what a scheduler can do; a full day shows whether it keeps doing it. The 24-hour campaign ran three configurations, each for a full 24 hours on a Kria KV260 board — 864 million timing cycles per run — and recorded worst-case scheduling latency on the isolated core:

- **Configuration A — stock software:** the board's standard software setup, with no real-time measures and no hardware timing profile. This is the baseline every other number is compared against.
- **Configuration B — real-time software only:** real-time software measures alone — a real-time kernel with isolated processors — without the hardware timing profile.
- **Configuration C — full hardware profile:** hardware timing enforcement together with the real-time software setup. This is the configuration the website publishes.

© 2026 Layer1Labs Silicon Inc.

Configuration	Worst case	99.99th percentile	Result
A — Stock software	904 μs	25 μs	Measured
B — Real-time software only	33 μs	15 μs	Pass
C — Full hardware profile	22 μs	12 μs	Pass

Read across the table, the three runs show what each layer buys. Stock software held its 99.99th percentile to 25 μs but let its worst case reach 904 μs over a full day. Real-time software alone brought the worst case down to 33 μs. The full hardware profile held the worst case to 22 μs, with 99.99% of cycles at or under 12 μs — and it is that configuration, run on one board, that the website reports.

Each row is reported the way it was run: **one board, one configuration, one full day.** No row is generalized — not to other boards, and not to a platform-wide latency figure.

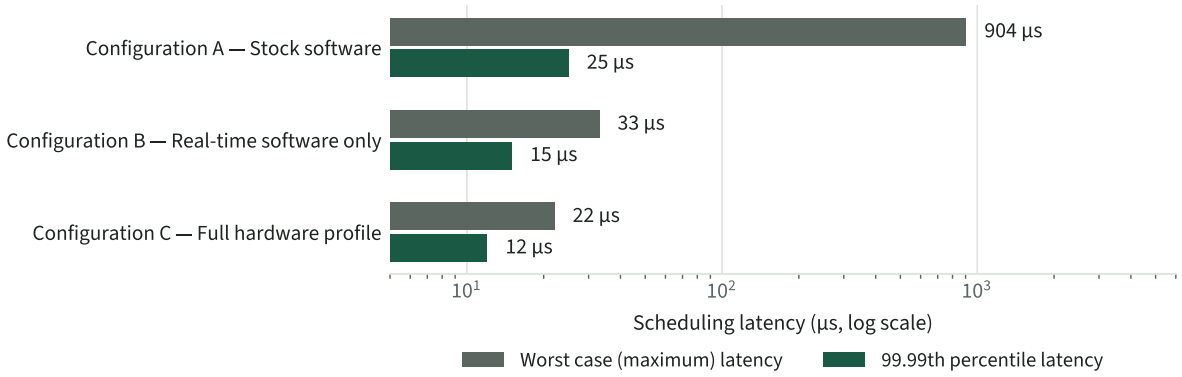


Figure 4: The 24-hour runs: worst-case and 99.99th-percentile scheduling latency for each configuration. Each run lasted 24 hours — 864 million timing cycles — on one Kria KV260 board. Configuration C is the result the website publishes as its 24-hour figure.

9 Equivalence at the 400G budget, on the two-board link

The equivalence runs of Section 6 compressed the per-packet budget on one board, in loopback, at the board's native 1-gigabit rate. A separate run compressed the interval on the two-board link itself. It is reported separately, in its own section, because its baseline is different — and the two results are not merged.

On the two-board 10-gigabit link, the same scheduling ran at a 25-microsecond interval — the timing budget of a 400-gigabit link — for 500 packets with zero misses, holding departure jitter to about 2.4 microseconds at that cadence.

That is an equivalent timing budget exercised on a 10-gigabit link, not a 400-gigabit measurement: no 400-gigabit link was run. What the run establishes is that the scheduling held the 400G cadence between the two boards, under the two-board conditions the rest of this document reports.

© 2026 Layer1Labs Silicon Inc.

10 The results at a glance

Result	Figure (Measured)	Basis
Delivery	21,000+ packets, 0 missed, 0 dropped — 100%	Validation runs, two Kria FPGA boards, 10-gigabit Ethernet link
Endurance	10,000 packets, every one delivered	One continuous run, inside the validation total
Burst groups	69 of 69 packets inside an 85 μs window; 700 of 700 inside a 1 ms window, $\sim$115 μs to spare	Burst tests: a 100 KB group in the window-enforcement run and a 1 MB group; every packet departed inside its scheduled window
Departure precision	Median and p99 exactly 1,000,000 ns	2,000-packet runs, programmed interval 1 ms
Departure jitter	Standard deviation as low as 61.1 ns	Same precision runs; $\sim$160$\times$ tighter than CPU scheduling on the same boards
Receiver-side precision	Arrival spread (trimmed standard deviation) 6.2 ns quiet, 6.3 ns under processor stress	Full transmit-and-receive dataset, 10,000 packets, May 2026; arrivals timed at the receiving board of the two-board link
Faster cadence	Departure spread 0.0 ns at a 100 μs interval	Coordinated two-board scheduling runs; the same 0.0 ns spread as at 1 ms
Saturation	Median and p99 exact under load	All four CPU cores at 100%; CPU scheduling on the same boards degraded to millisecond-scale jitter
Adaptive scheduling	10,048 of 10,048 delivered, 53 adjustments	One run; scheduler adjusting to on-chip constraint signals
Under a packet flood	Median and p99 departures exactly on the 1 ms tick; target hit for 99.82% of packets	One scheduled flow under a flood of 50,000 competing packets
Power states	Target hit for 99.94%–100% of packets; median and p99 exactly 1 ms in every state	Board processors in performance, power-saving, and partially-offline states, and cycling under stress
Ring step	Median and p99 round trip exactly 1 ms across 999 steps; unscheduled steps $\sim$645 μs longer per step	Two-board ring test — the step pattern a collective operation repeats
Timing-budget equivalence	Target-hit rate 100.0%, zero stale targets, standard deviation 0.0 ns against the target schedule, at per-packet budgets of 10,000 / 2,500 / 1,250 / 620 ns (100G / 400G / 800G / 1.6T-equivalent)	Equivalence runs, April 2026: one Kria KR260 board, on-board loopback, native 1-gigabit Ethernet; 5,000 packets per level, quiet and under CPU stress. Validates the timing mechanism against the target schedule at those budgets — not a measurement of jitter at those line rates, and not data transport at 400G, 800G, or 1.6T
Verified in simulation	Gate precision held at every simulated budget; departure spread zero at all rates; zero missed gates at the 620 ns budget; burst timing held	Simulation verification, June 2026: a cocotb regression suite (the standard Python chip-verification framework) driving the gate design in the GHDL simulator at budgets from 1G to 1.6T equivalents; formal report 45 of 46 tests passing, one skipped, none failing. Verifies the design at those budgets — not a physical link at those rates
Equivalence on the link	500 packets at a 25 μs interval (400G-equivalent budget), zero misses; departure jitter $\sim$2.4 μs	Equivalence run on the two-board 10-gigabit link: the timing budget of a 400-gigabit link, exercised at 10 gigabits — not a 400-gigabit measurement
24-hour run, stock software	Worst case 904 μs; 99.99th percentile 25 μs (Measured)	One Kria KV260 board; 24 hours, 864 million cycles; standard software setup, no real-time measures
24-hour run, real-time software only	Worst case 33 μs; 99.99th percentile 15 μs (Pass)	One Kria KV260 board; 24 hours, 864 million cycles; real-time kernel, isolated CPUs, no hardware timing profile
24-hour run, full hardware profile	Worst case 22 μs; 99.99th percentile 12 μs (Pass)	One Kria KV260 board; 24 hours, 864 million cycles; hardware timing enforcement with the real-time software setup
Cross-board	Collective departure spread 0 ns	Two boards firing at the same hardware tick

© 2026 Layer1Labs Silicon Inc.

11 Limits, stated plainly

- **The beta platform is the measurement platform.** Every figure in this document was measured on Kria FPGA evaluation boards — over 10-gigabit links, except the timing-budget equivalence runs: those of Section 6 ran on one board at its native 1-gigabit rate by design, and those of Section 9 ran at a compressed interval on the two-board 10-gigabit link. Production-scale fabrics (100G/400G, ASIC) remain the roadmap, not a measurement — at those rates, the gate design is verified in simulation (Section 7), not measured.
- **Equivalence is not transport.** The equivalence runs (Sections 6 and 9) validate the timing mechanism against its target schedule at faster-rate budgets. No link at 100G, 400G, 800G, or 1.6T was run, and no throughput at those rates was measured.
- **Simulation is not silicon.** The results of Section 7 verify the gate's design in a simulator at the higher-rate budgets. They are not measurements of a board running at those rates, and they do not substitute for a physical link at those rates when one is built.
- **The supporting results are single datasets.** The burst, receiver-side, flood, power-state, cadence, and ring-step results of Sections 3–5 each come from the single run or dataset stated with them. They are reported as run, and they are not combined with the headline figures.
- **Each 24-hour result is one board and one configuration.** The three configurations of Section 8 were each run once, for one full day, on a Kria KV260 board; no row is generalized beyond that run.
- **These are evaluation-platform runs.** The results are validation and test runs on the beta platform — not a production deployment, and not a customer's workload.
- **The roadmap is not measured.** Embedded chiplets and merchant silicon are future offerings on the staged hardware roadmap and are not part of the measurements above.

12 What these results do not establish

These results do not establish how ChronoFabric performs at production scale: no 100G or 400G fabric and no ASIC implementation has been measured, and the roadmap items are not results. In particular, the timing-budget equivalence runs of Sections 6 and 9 do not establish jitter at 100G, 400G, 800G, or 1.6T line rates, and they do not establish data transport or throughput at any of those rates — they validate the timing mechanism against its target schedule at the budgets those rates would impose, on a board running at 1 gigabit. The simulation results of Section 7 likewise establish no physical link at those rates and no silicon behavior at them; they verify the design's behavior at those budgets. They do not establish the 24-hour latency behavior of any board or configuration other than the three measured, each on the single board and single run reported. They do not establish what a particular customer's workload would see — that is what the beta program is for: ChronoFabric is complete and ready for beta customers, participation is arranged directly with the team (technical and beta questions: info@layer1labs.ai; partnerships and licensing: licensing@layer1labs.ai), and it is offered for commercial licensing by written agreement.

What they do establish is narrower: on the beta platform, across more than 21,000 scheduled packets, the hardware scheduler delivered every packet — burst groups included, each inside its window — departed on the programmed tick with nanosecond-scale spread, measured at both ends of the link, kept doing so with the host processor saturated, under a 50,000-packet flood, across the processor's power states, and while adjusting its own schedule, held a two-board ring step to exactly the programmed round

© 2026 Layer1Labs Silicon Inc.

trip, held its target schedule at per-packet budgets down to the 1.6T-equivalent cadence and at the 400G budget on the two-board link itself, had its design verified in simulation at all six rate budgets with its precision intact, and sustained its latency behavior over full-day runs in all three configurations measured.

A companion document, *ChronoFabric Technical Brief*, presents the architecture these results come from — the two tiers, the hardware-enforcement model, and the staged hardware strategy. The software tier's results are carried in the companion *ChronoHive Measured Results*, and are not repeated here.

Colophon

Sources: `https://layer1labs.ai/chronofabric/` (the "Measured on Hardware" section) and `https://layer1labs.ai/faq/`, both fetched live on October 26, 2026. The timing-budget equivalence results (Sections 6 and 9), the completed three-configuration 24-hour table (Section 8), and the supporting results of Sections 3–5 are printed from Layer1Labs' own hardware test records of April–May 2026 and of the completed 24-hour campaign, at the values recorded there. The simulation verification results (Section 7) are printed from Layer1Labs' own verification report of June 2026. Figures 1–4 are script-generated from those values. Build: Version 1.0, October 26, 2026; LuaLaTeX, in the Layer1Labs document family; no figure was hand-drawn or hand-typed.

© 2026 Layer1Labs Silicon Inc.

Notices

Copyright. © 2026 Layer1Labs Silicon Inc. All rights reserved.

Sharing. This document may be shared in its complete, unmodified form, with attribution to Layer1Labs Silicon Inc. No other reproduction, modification, adaptation, or derivative use — in whole or in part — is permitted without the prior written permission of Layer1Labs Silicon Inc.

No license. Nothing in this document grants, by implication or otherwise, any license or right under any patent, copyright, trademark, or other intellectual property of Layer1Labs Silicon Inc. Publication of this document is a disclosure of information only.

Patents. The technology described in this document is the subject of pending patent applications, including 9 provisional patent applications filed with the U.S. Patent and Trademark Office. No statement in this document is a representation as to the scope, validity, or enforceability of any application or of any patent that may issue from one.

Trademarks. Layer1Labs, ChronoHive, and ChronoFabric are trademarks of Layer1Labs Silicon Inc. Other names mentioned in this document may be trademarks of their respective owners; their use here is for identification only and claims no affiliation or endorsement. In particular, Lingua Franca is an open-source research project of its own community.

Measurements and basis. Figures in this document are drawn from the project's measurement records and are cited where they appear; where a figure is modeled rather than measured, it is labeled as modeled. This document is provided for information only, on an "as is" basis, without warranty of any kind, express or implied, including as to accuracy, completeness, or fitness for a particular purpose. It describes a demonstration system; it is not a specification, an offer, or a forecast of results in any other setting.

Current version. Documents in this family are revised in place under version numbers. The current version of this document is the one published at layer1labs.ai; earlier versions are superseded.