Rapid Eye Analysis
Quick Start Guide

A hands-on tour of every core feature, using the bundled demo assets

Software version v2.0.28

2026-10-07

Contents

Before You Start: How to Use This Guide 3

Step 1 Your First Eye Diagram (5 minutes) 4

Step 2 Reading the Metrics 6

Step 3 Equalizers: Rescuing a Closed Eye 7

Step 4 Crosstalk, Jitter and Noise 10

Step 5 One-Click COM Import 11

Step 6 Saving and Loading Configurations 12

Step 7 The S-Parameter Toolbox: All Six Windows 13

- 7.1 TDR: impedance vs. distance . . . 13
- 7.2 Smith chart: reflection trajectory . . . 14
- 7.3 Cascade: chaining segments . . . 14
- 7.4 De-embed: removing the fixtures . . . 15
- 7.5 Split: halving a long channel . . . 16
- 7.6 Format conversion: S4P differential extraction . . . 16

Step 8 Expert Mode and Optical Links (a Preview) 17

Step 9 Automation (SCPI Remote Control) 22

Step 10 Pre- and Post-FEC BER (optional) 22

Reference: How the 500 Trial Runs Are Counted 23

Where to Go Next 25

Before You Start: How to Use This Guide

Welcome to Rapid Eye Analysis (REA). From a single channel S-parameter file (.s4p / .s2p, as exported by a VNA or an EM simulator), REA computes the eye diagram of a high-speed signal over that channel and helps you tune the link to its best. This guide assumes you have never used the software: follow the 10 steps below and you will generate your first eye, read the metrics, rescue a closed eye with equalizers, inject crosstalk and jitter, import an IEEE 802.3 COM configuration in one click, save/load configurations, exercise all six S-parameter tool windows, and take a peek at Expert mode and optical links.

Everything uses the **bundled demo assets** in the **demo** folder of the installation directory (when running from source: demo\ at the project root). Every screenshot in this guide was taken with those files — what you see should match.

File in demo\sparams\	Used for
demo_channel_clean.s4p	Clean short channel — first eye (Step 1)
lossy_channel_10dB.s4p	Lossy channel — equalizer tutorial (Step 3)
xtalk_aggressor.s4p	Crosstalk aggressor (Step 4)
demo_trace.s2p	2-port trace — TDR / Smith chart (Step 7)
fixture_2inch.s2p	Test fixture — cascade / de-embed (Step 7)
trace_with_fixture.s2p	Trace "as measured with fixtures" (Step 7)
long_channel.s4p	Long channel — split tutorial (Step 7)
Other demo\ content	**Used for**
com_import\ (three scenarios)	One-click COM import (Step 5)
configs\ (Normal / Expert .json files)	Reproducible baselines and optional exercises (Steps 6 / 8)
ibis\ (reference IBIS-AMI models)	IBIS-AMI device models (Step 8)
scpi_example.txt	SCPI automation example (Step 9)

Tip: Suggested route: complete Steps 1-3 first: generate, read and jointly equalize an eye. Then try Step 10 for FEC or Steps 5-6 for COM / JSON reuse. Choose toolbox, Expert / optical / IBIS and SCPI exercises as needed. Reload each case's baseline so earlier settings cannot leak into it. Compare eye count and trends with the pictures; runtime and last digits vary with hardware.

> **Tip:** Unfamiliar with a term? Press F1 to open the full User Manual — Appendix B is a plain-language glossary. Each step below also points to the manual chapter that goes deeper.

Step 1 Your First Eye Diagram (5 minutes)

Step 1 Launch RapidEyeAnalysis.exe and select **Mode** → **Normal**. Parameters are on the left and charts on the right (Fig. 1-1). Save any existing work, then clear the Input Files list so an old channel is not automatically cascaded with the demo.

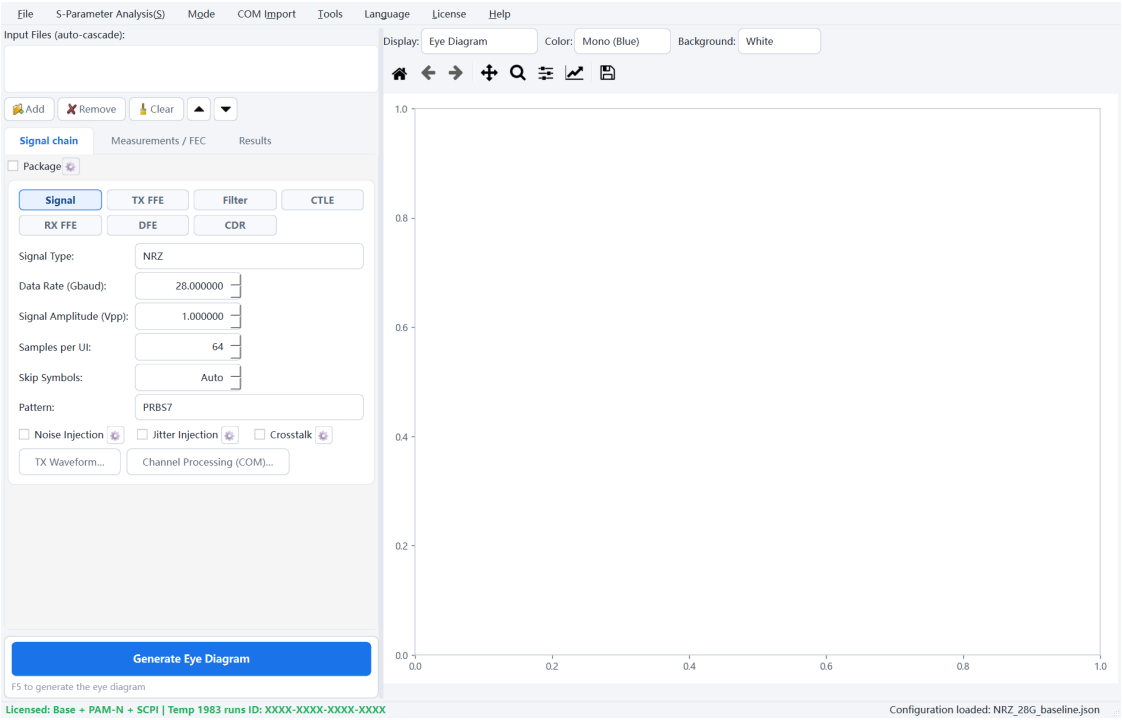

Fig. 1-1 The main window right after launch

Step 2 Under "**Input Files (auto-cascade):**" at the top-left, click the add button and pick demo\sparams**demo_channel_clean.s4p**.

→ GUI feedback: The file appears in the list; hovering shows the port count (4) and frequency range.

Step 3 Select **File** → **Load Configuration...** and load demo\configs**NRZ_28G_baseline.json**. Check Signal chain → Signal: **NRZ, 28 GBd, PRBS7, 64 samples/UI, 1 V amplitude**, with cross differential pairing. Equalizers, crosstalk, jitter and FEC are off. Only the clean channel should be listed (Fig. 1-2). REA remembers settings: restarting alone does not restore defaults.

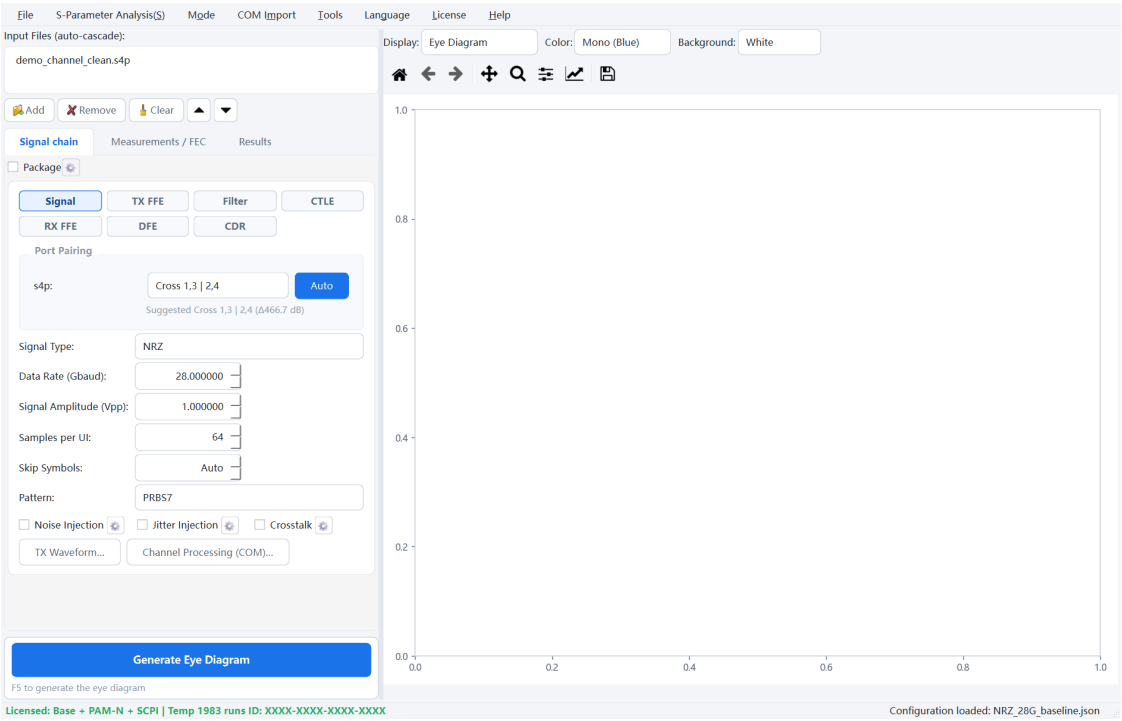

Fig. 1-2 File selected, all defaults — right before generating

Step 4 Click the fixed bottom "**Generate Eye Diagram**" button, or press F5. When calculation finishes, your first eye appears (Fig. 1-3).

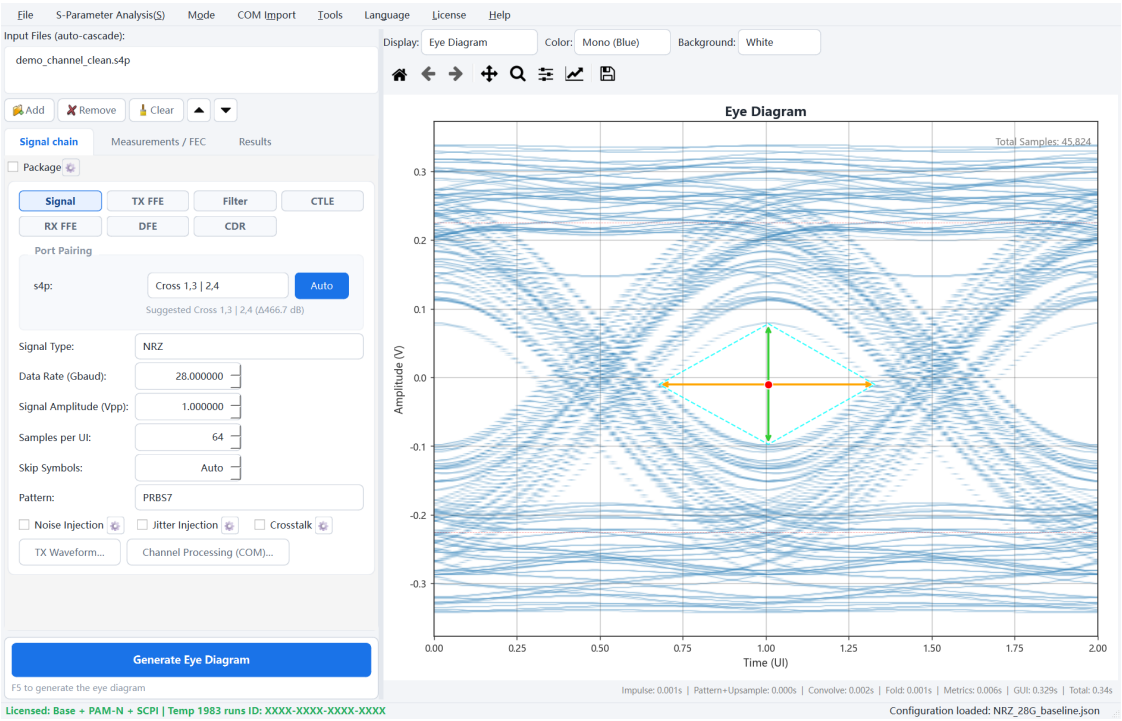

Fig. 1-3 Clean channel + defaults: a wide-open eye

Tip: What is an eye diagram? Overlay thousands of transmitted bit waveforms period by period: the wider the "eye" opening in the middle, the cleaner the signal and the easier it is for the receiver to make correct decisions. A closed eye = a broken link.

> **Tip:** The input list supports **multiple files with automatic cascading**: add cable, connector, and fixture S-parameters in order and the software chains them into one link before analysis.

More on file loading: Manual Chapter 3.

Step 2 Reading the Metrics

After generating an eye, open the left **Results** page to read eye height, eye width and COM; the eye stays visible on the right. Start with the three most used numbers:

Metric	Plain meaning	How to read
Eye Height (V)	Vertical opening = voltage margin against noise	Bigger is better
Eye Width (UI)	Horizontal opening = timing margin against jitter	Bigger is better
COM (dB)	The IEEE 802.3 "channel operating margin" score — loss, reflections, crosstalk and noise folded into one number	Compare with the selected standard
VEO / SNDR	Vertical eye opening and signal-to-noise-and-distortion — COM companions	Bigger is better
Channel @Nyq lines	The channel's own report card: Nyquist insertion loss, ILD, return loss, ICR, pulse cursors h0/h±1 (Manual 10.17.5)	Less loss is better; RL/ICR bigger is better
Bathtub / Mask margin	Expert bathtub mode shows one probe at a time; use the probe picker or result table to switch, keeping all eyes of that probe visible. The "Display" combo switches to separate bathtub rows per eye: phase (UI) left, voltage (V) right, dotted = extrapolated. Drag the BER line to adjust all eyes and read EH/EW; or use a mask overlay ("Mask…" sets shape and size) showing PASS/FAIL and four margins (Manual 10.12.1)	Bigger opening at the target BER is better; PASS needs hits = 0

> **Tip:** Missing bathtub eye data shows the usable eye count and insufficient data; overall opening is unavailable. A mask PASS also requires enough data for every eye. The Step 1 teaching baseline is approximately 0.176 V eye height, 0.645 UI eye width and 1.39 dB COM. It checks the workflow and open eye; it is not a compliance certification case. Acceptance criteria depend on the selected standard. Open **Measurements / FEC** → **Decision Parameters** on the left to adjust sampling time or thresholds. COM Noise Budget on the same page explains the margin contributions. Details: Manual Chapter 4.

Step 3 Equalizers: Rescuing a Closed Eye

Real high-speed channels are lossy — by the time the signal reaches the receiver it is smeared. In this step we stage an accident on purpose, then rescue it with equalizers. This is the core workflow of the software.

Step 1 Return to Normal, clear "Input Files" and add demo\sparams**lossy_channel_10dB.s4p** (roughly 10 dB of loss). Load demo\configs**NRZ_28G_baseline.json** to start without equalization, even if you skipped Steps 1 and 2.

Step 2 On the "Signal" tab set **Signal Type** to PAM4 and **Data Rate (Gbaud)** to 53.125 — the real per-lane rate of 100G Ethernet.

Step 3 Click "Generate Eye Diagram". This time all three eyes are nearly shut (Fig. 3-1) — heavy loss plus high speed causes severe inter-symbol interference.

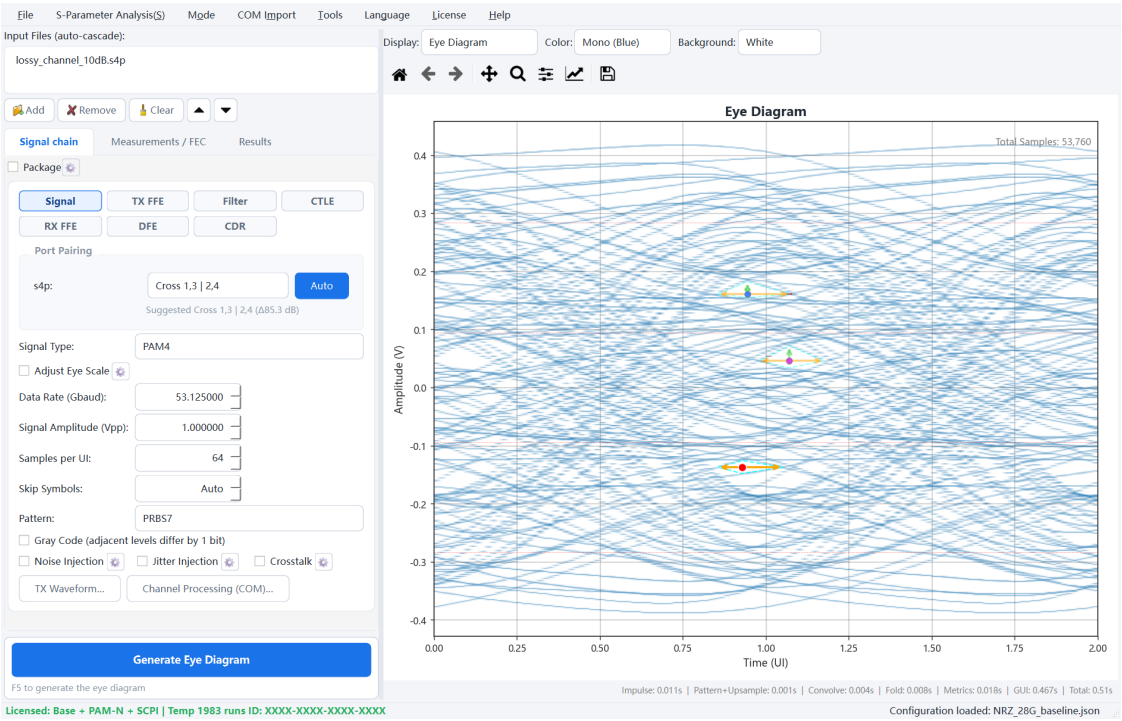

Fig. 3-1 Lossy channel, no equalization: eyes closed

Step 4 Switch to the **TX FFE** tab (Fig. 3-2), tick enable, then click "**Optimize TX FFE**". The software searches for the best transmitter pre-emphasis taps automatically.

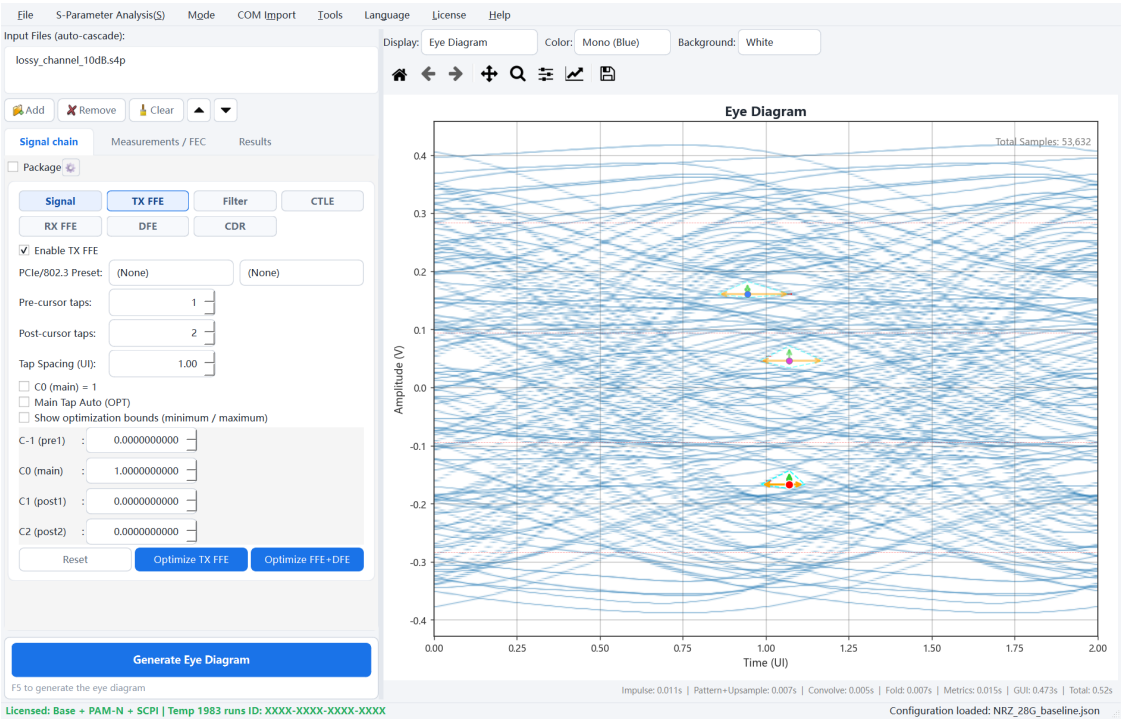

Fig. 3-2 TX FFE panel: enable switch and Optimize

Edit the tap coefficients directly for routine adjustments. Tick "**Show optimization bounds**" to inspect or change their limits; TX FFE, RX FFE and DFE all provide this option.

Step 5 Switch to **CTLE**: choose **IEEE 802.3** in the first preset dropdown, then choose the **802.3ck-C2C** preset from the drop-down (the 53 GBd class — a CTLE only helps when its poles and zeros match the data rate; that is what presets are for), tick enable, then click "Optimize CTLE".

Step 6 Do the last two stages together: tick enable on the **DFE** tab, then on the **RX FFE** tab tick enable and click "**Optimize RX FFE+DFE**" (joint). Why not one at a time? The 802.3ck-C2C preset gives the first DFE tap a lower bound of 0.3, as the standard does; if the RX FFE is first optimized alone until the inter-symbol interference is gone, a DFE optimized afterwards can only work at that lower bound and closes the eye again (the optimization summary reports the loss). The joint optimization splits the work between the RX FFE and the bounded DFE in one go. After each stage you can regenerate the eye to watch it open up.

Step 7 With all stages done, click the fixed bottom "Generate Eye Diagram" button (or press F5) once more: all three eyes are open (Fig. 3-3) and COM has climbed from negative to about 9 dB.

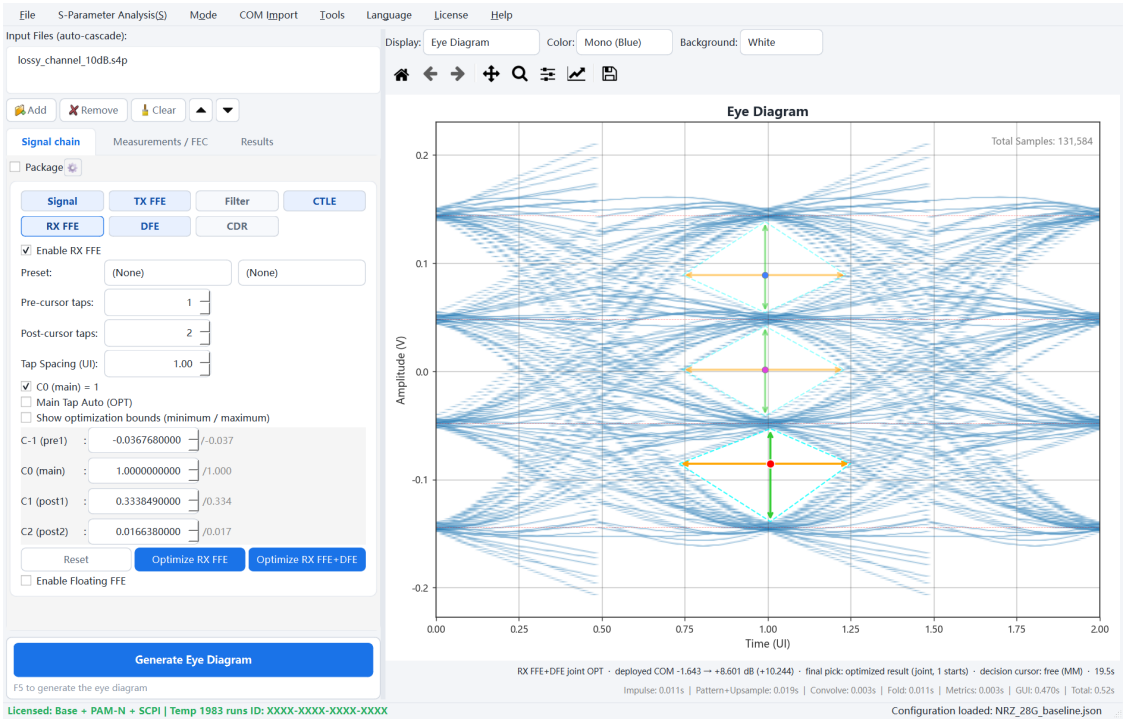

Fig. 3-3 Full equalizer chain: three open eyes

> **Tip:** What "Optimize" does: an automatic search for the best parameters of that equalizer, scored by COM. Recommended order for this example is TX FFE → CTLE → **joint RX FFE+DFE**, and the CTLE / TX FFE tabs ship **presets** for 802.3 / PCIe / USB — pick the right rate class before optimizing. When an optimization finishes, an **optimization summary** line appears to the right of the eye: COM before → after, whether the optimized result or the previous parameters were kept, and the elapsed time. Every optimization is also double-checked: if a stage truly cannot help (no improvement), the summary and the status bar say so and the parameters stay unchanged; if bypassing the CTLE is better, the software switches it off. After changing tap structure, switches or search bounds, compare the deployed result in the summary and eye diagram. Fixed and floating taps are optimized separately, preserving the other segment. A failed optimization is explained in red in the summary.

Equalizer theory and manual tuning: Manual Chapter 10.

Step 4 Crosstalk, Jitter and Noise

Real links live next to other channels that talk over them, and clocks always jitter. REA can inject these impairments so you can see how much margin survives.

Step 1 Return to Normal, clear "Input Files", add demo\sparams**demo_channel_clean.s4p**, then load demo\configs**NRZ_28G_baseline.json**. Generate an unimpaired eye first: it should match the trend in Fig. 1-3. Changing only the channel leaves the PAM4 rate and equalizers from Step 3 active.

Step 2 On Signal chain → Signal tick "**Crosstalk Injection**", open its settings window and add the FEXT aggressor file demo\sparams**xtalk_aggressor.s4p**, setting its amplitude to 0.1 V (real aggressors carry only a fraction of the main signal).

Step 3 Confirm crosstalk settings, then tick "**Jitter Injection**", open its settings and set RJ to **0.01 UI**. Confirm the settings.

Step 4 Generate the eye and compare with Fig. 1-3: the eye is fuzzier and margin is reduced (Fig. 4-1) — that is the margin eaten by crosstalk and jitter.

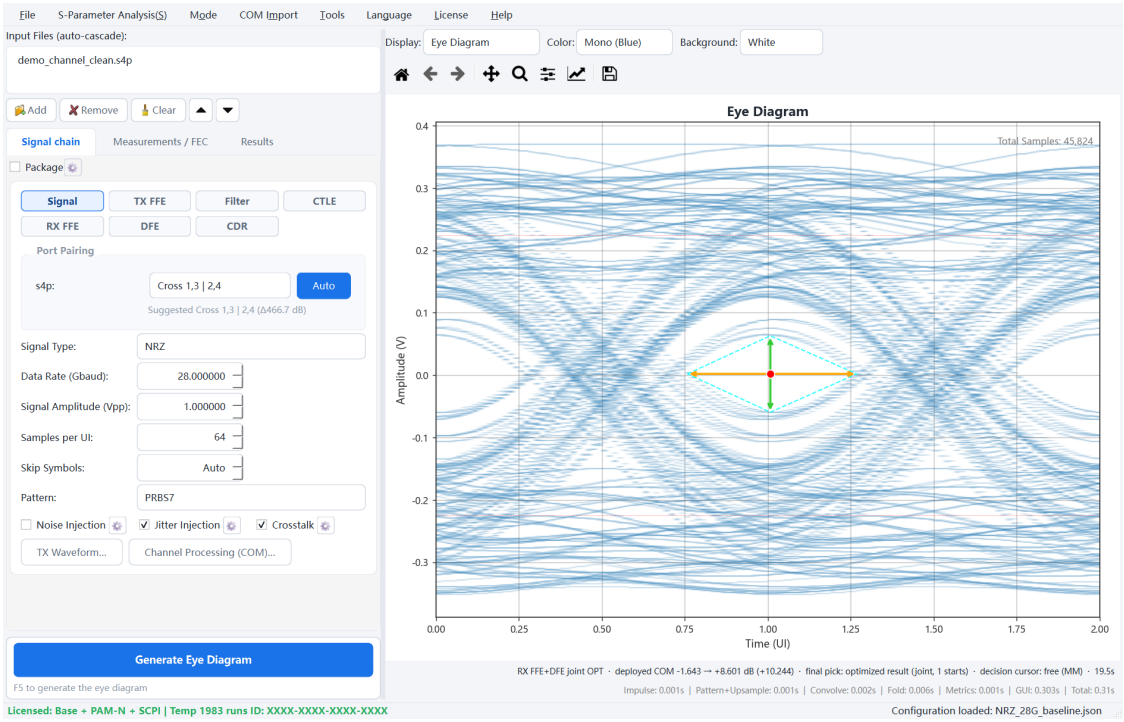

Fig. 4-1 The same channel with crosstalk + jitter

> **Tip:** The same area offers "Noise Injection" switches for white noise at selectable positions. Untick everything when done so later steps start clean. Details: Manual Chapter 10.

Step 5 One-Click COM Import

If your team uses the IEEE 802.3 COM reference tool, REA reads its result files directly and imports every parameter — signal, equalizers, package, noise — in one click, then generates the baseline eye automatically. No numbers to copy by hand.

Step 1 Return to Normal and choose **COM Import** on the menu bar to open a three-field dialog (Fig. 5-1).

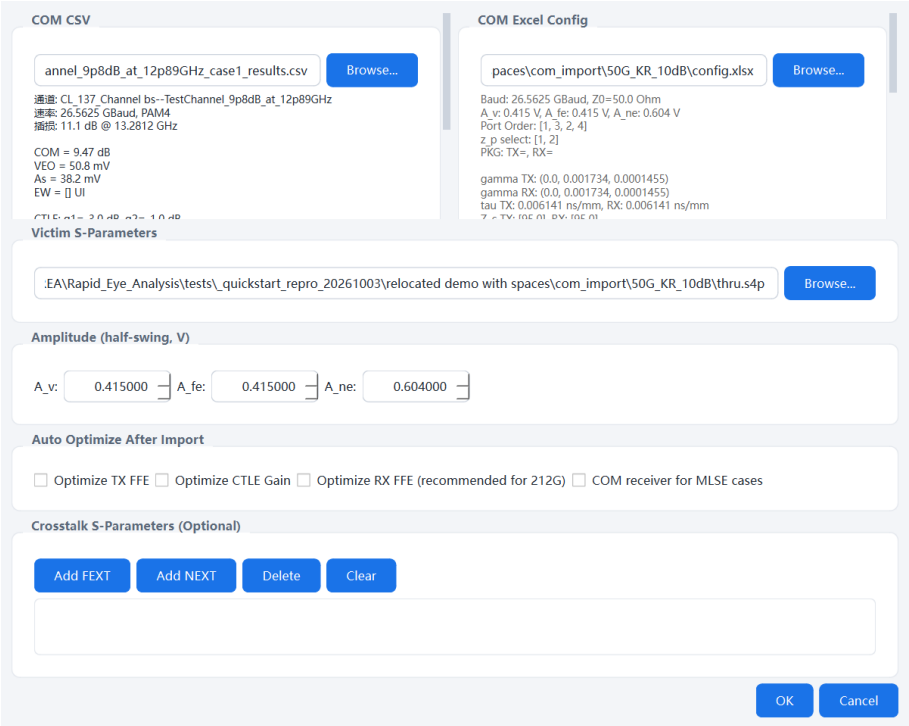

Fig. 5-1 COM import dialog

Step 2 Pick the three files from demo\com_import\50G_KR_10dB\: the results CSV (*case1_results.csv), config.xlsx, and thru.s4p.

Step 3 Leave every Auto Optimize After Import option off, then click OK. The baseline eye is generated automatically from the imported parameters (Fig. 5-2); further optimization is a separate exercise.

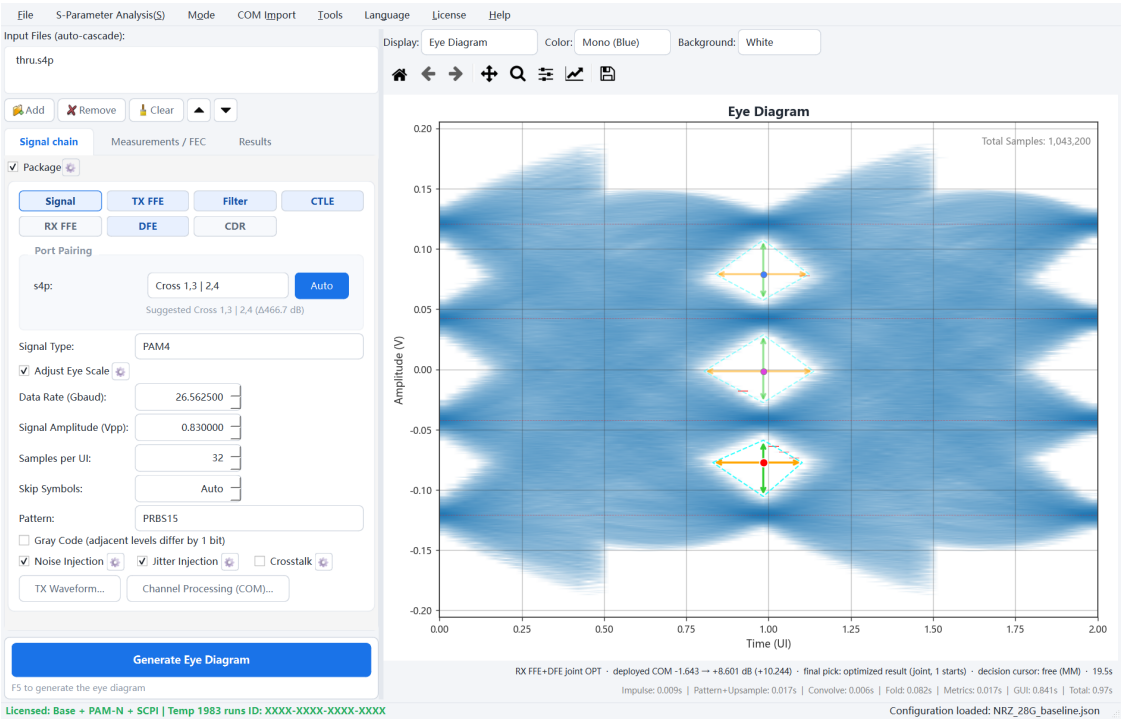

Fig. 5-2 Baseline eye right after COM import

> **Tip:** The dialog also offers "Auto Optimize After Import" to let REA keep tuning from the imported starting point. Two more scenarios (120G and 212G) are provided under demo\com_import\. Details: Manual Chapter 5.

Step 6 Saving and Loading Configurations

Floating FFE/DFE coefficients, positions and decision timing are also saved. For the Floating FFE examples, read demo/floating_ffe/README.txt, load a JSON from configs, and regenerate the eye to compare the reference result. Keep the relative locations of channels and configs when moving the examples.

A fully tuned parameter set (signal + every equalizer + advanced settings) can be saved as a single .json file — restore it later, or send it to a colleague to reproduce your result.

Step 1 Menu **File** → **Save Configuration...** stores the current panels to disk.

Step 2 Now try loading: return to Normal, clear "Input Files" and add only demo\sparams**lossy_channel_10dB.s4p**, then **File** → **Load Configuration...** and pick demo\configs**PAM4_53G_full_EQ.json**.

Step 3 Click "Generate Eye Diagram" — you get a fully equalized link immediately (Fig. 6-1), on par with what you tuned by hand in Step 3.

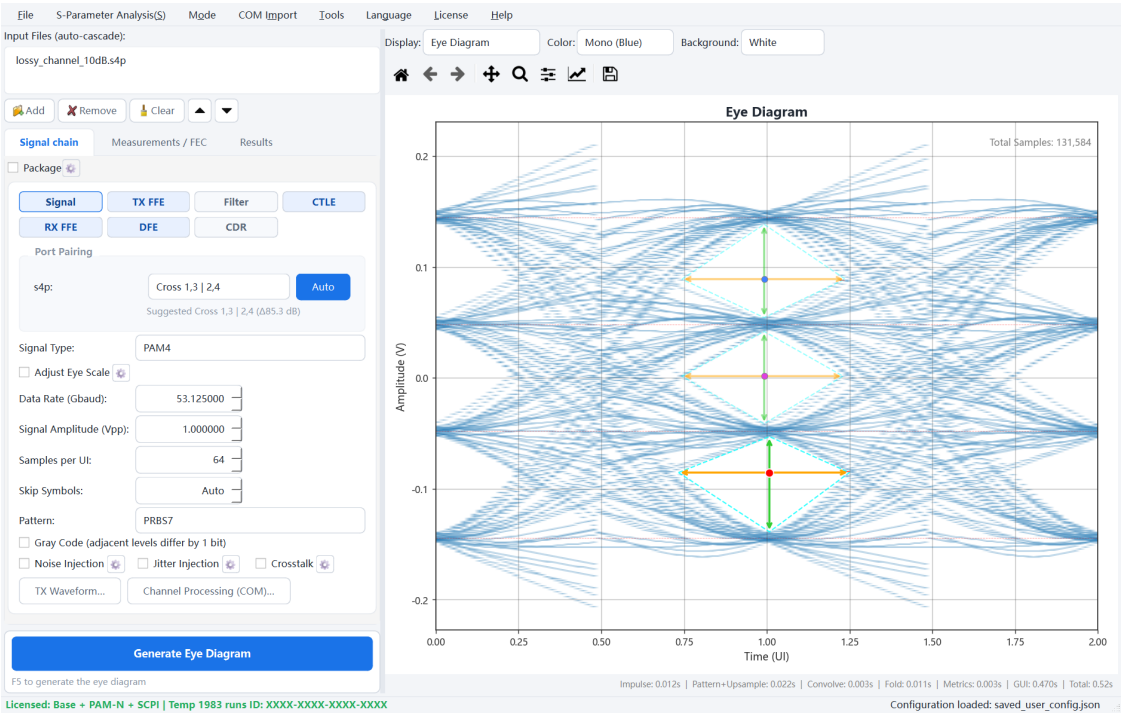

Fig. 6-1 After loading the demo configuration

> **Note:** Configuration files do not embed the channel itself: load the matching channel into "Input Files" first, then the config. demo\README.txt lists which channel pairs with which config. Sharing across machines: put the config JSON and the channel files in one folder - stale paths are re-resolved by filename in the folder the config was loaded from.

Step 7 The S-Parameter Toolbox: All Six Windows

The eye diagram shows what the signal looks like after the channel; the six windows under the **S-Parameter Analysis** menu examine the channel itself. Try each one with the demo files.

7.1 TDR: impedance vs. distance

Step 1 Menu **S-Parameter Analysis** → **TDR**, pick demo\sparams**demo_trace.s2p**, tick S11, compute — impedance discontinuities along the trace stand out immediately (Fig. 7-1).

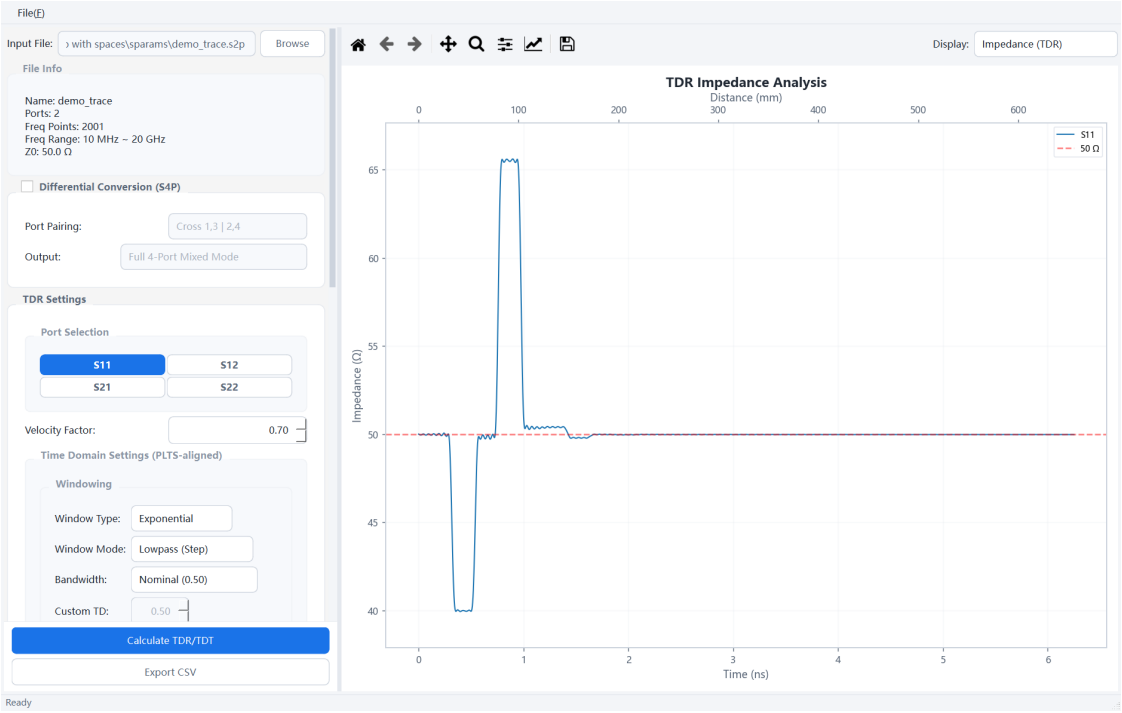

Fig. 7-1 TDR impedance profile

7.2 Smith chart: reflection trajectory

Step 1 Menu **S-Parameter Analysis** → **Smith Chart**, same file, plot (Fig. 7-2).

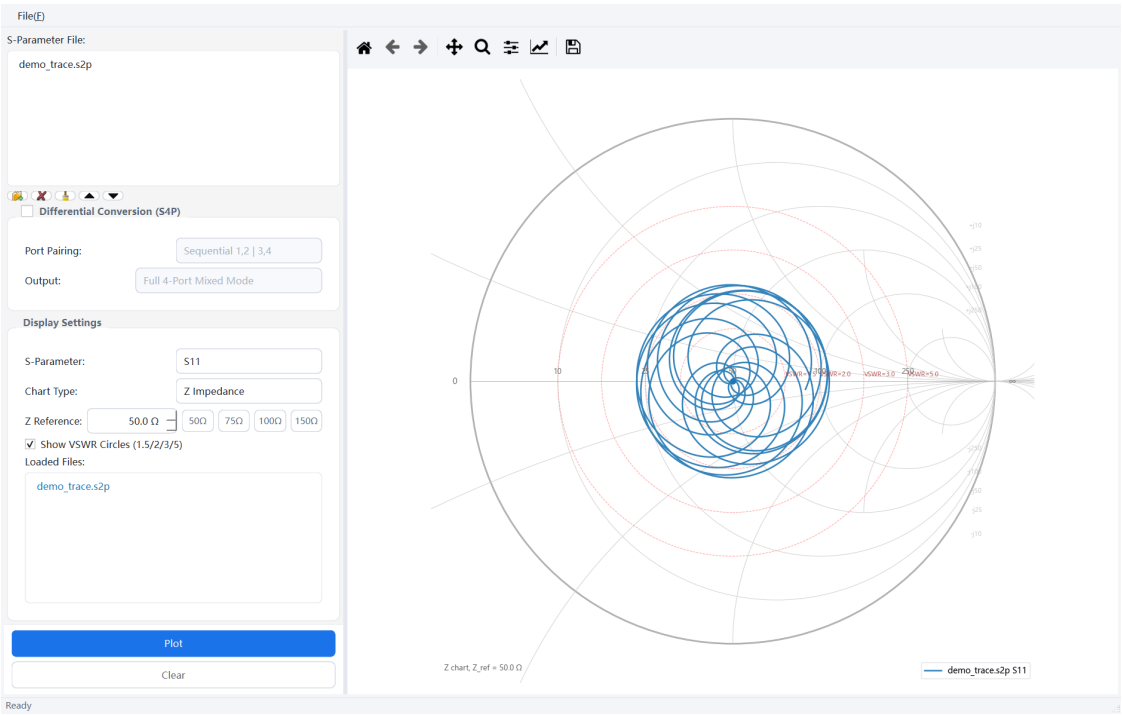

Fig. 7-2 Smith chart

7.3 Cascade: chaining segments

Step 1 Menu **S-Parameter Analysis** → **Cascade**, add three files in order: **fixture_2inch.s2p**, **demo_trace.s2p**, **fixture_2inch.s2p** — a realistic fixture + trace +

fixture measurement chain — then cascade (Fig. 7-3). The result can be saved as a new S-parameter file.

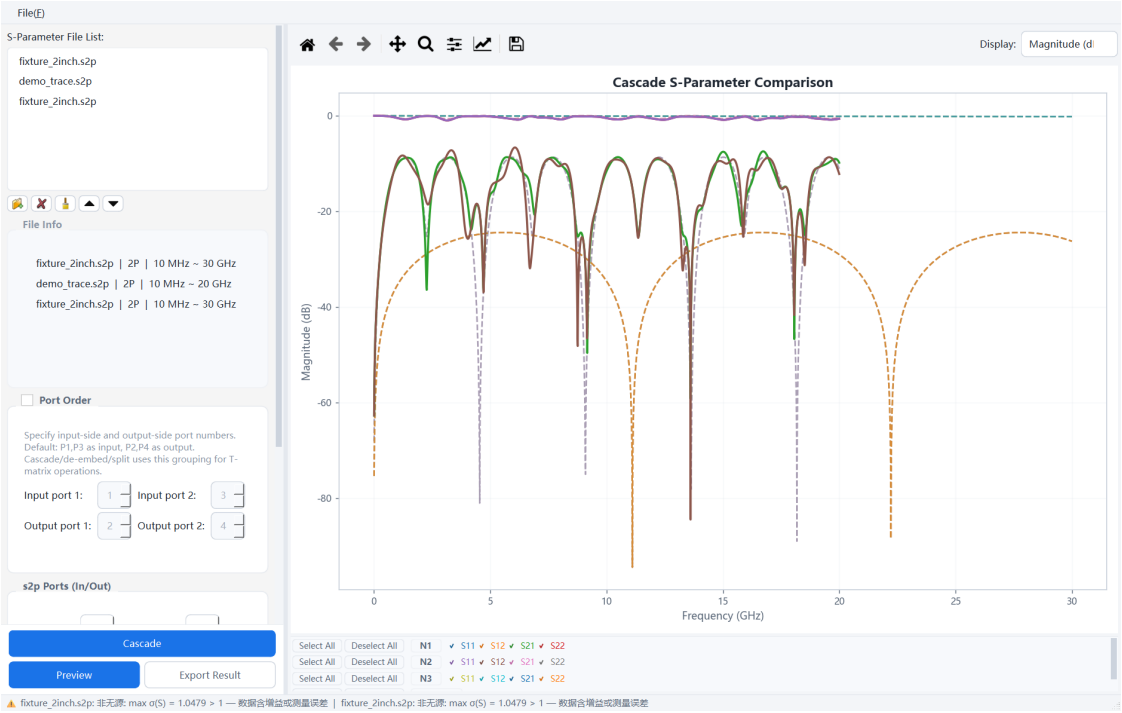

Fig. 7-3 Cascade: fixture + trace + fixture

7.4 De-embed: removing the fixtures

Fixtures inevitably contaminate a measurement. De-embedding subtracts them to recover the device under test. The demo file trace_with_fixture.s2p is exactly such a "measured with fixtures" trace:

Step 1 Menu **S-Parameter Analysis** → **De-embed**: DUT = **trace_with_fixture.s2p**; enable the left and right fixtures and set both to **fixture_2inch.s2p**; de-embed (Fig. 7-4). The result is the clean demo_trace itself.

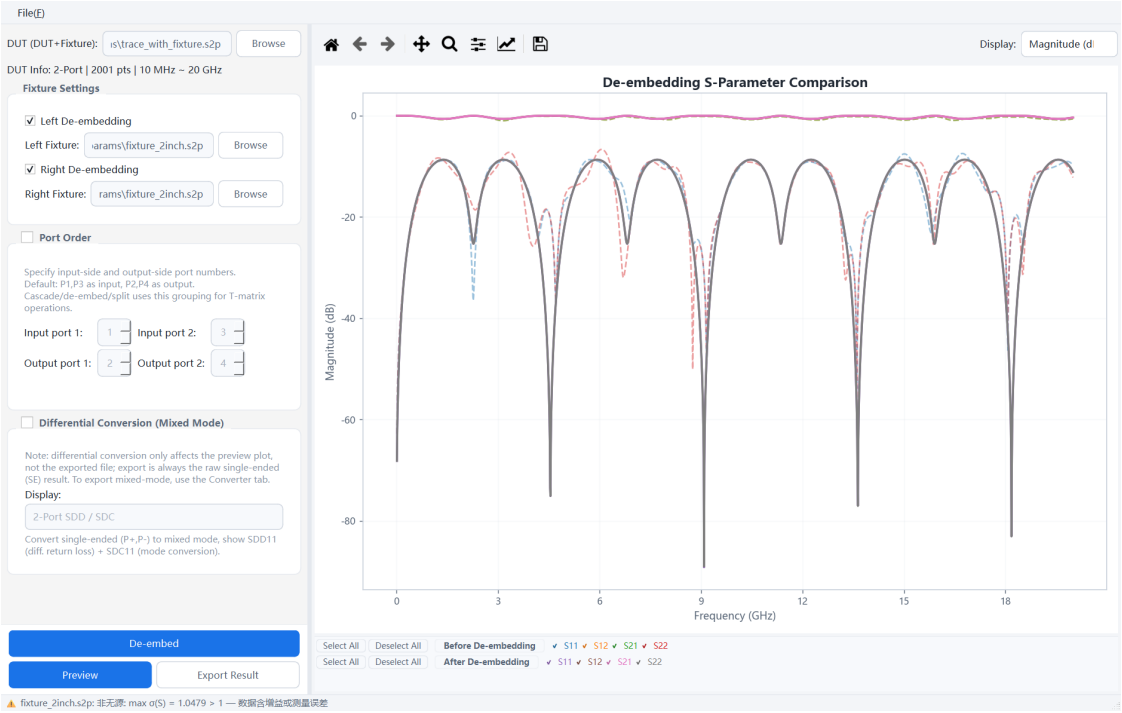

Fig. 7-4 De-embed: both fixtures removed

7.5 Split: halving a long channel

Step 1 Menu **S-Parameter Analysis** → **Split**, pick **long_channel.s4p**, 2 segments, click Split, then Preview to plot both halves (Fig. 7-5). The 2x-thru method cuts the channel into halves; each half can be saved and analyzed on its own.

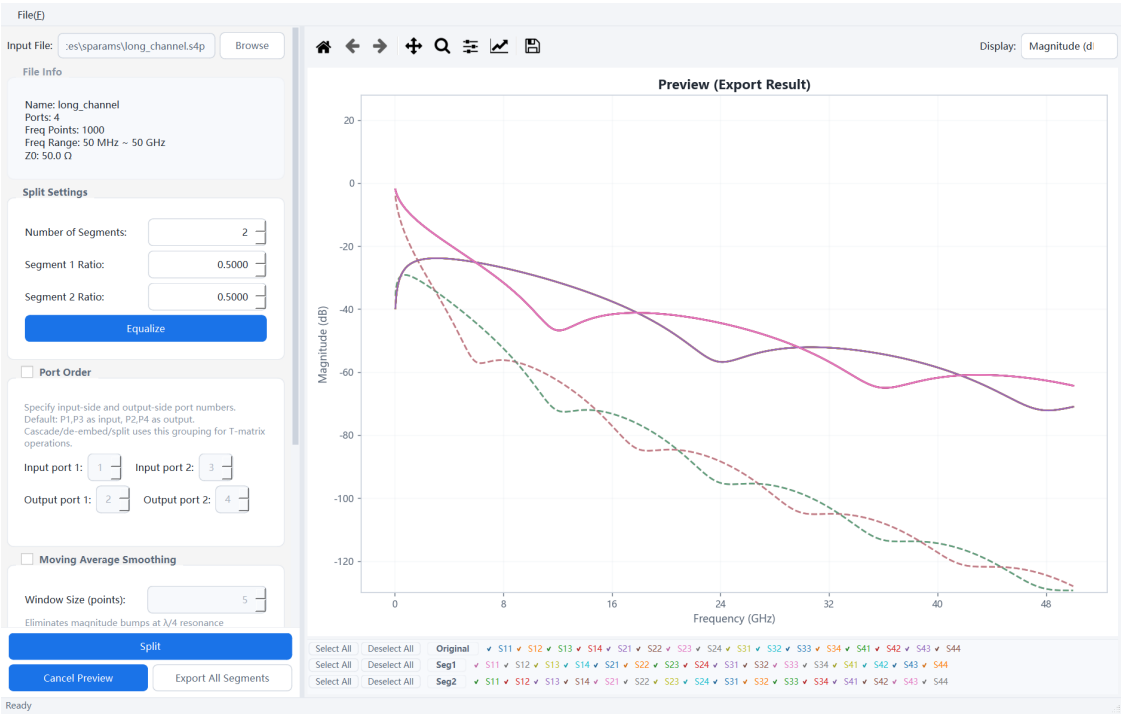

Fig. 7-5 Split: one channel, two halves

7.6 Format conversion: S4P differential extraction

Step 1 Menu **S-Parameter Analysis** → **Format Conversion**, pick **demo_channel_clean.s4p**, enable the differential mixed-mode group, use cross 13/24 pairing, select differential (SDD) output and preview (Fig. 7-6) — turning a 4-port file into a 2-port differential file for tools that only accept 2-port data.

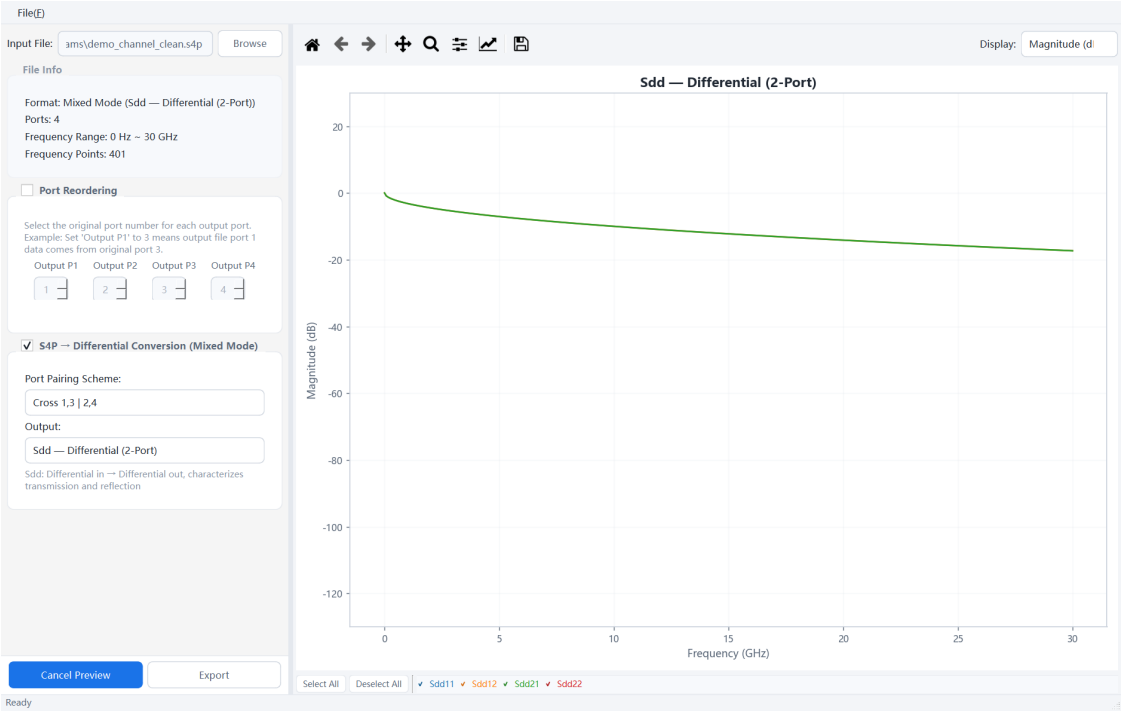

Fig. 7-6 Conversion: S4P → SDD

Full documentation of all six windows: Manual Chapters 6–9.

Step 8 Expert Mode and Optical Links (a Preview)

Normal mode uses a fixed signal-chain order; **Expert mode** turns the link into a free-form node canvas — multiple branches, multiple probes, and optical devices.

Step 1 First follow Step 6 to load **PAM4_53G_full_EQ.json** with the lossy channel and generate an eye. Then select **Mode** → **Copy Normal Config to Expert**. This overwrites the existing Expert canvas, so save any work you need first.

Step 2 Menu **Mode** → **Expert — Pipeline Designer**, then click Run: the link appears as a node graph, with one eye per probe (Fig. 8-1). Double-click a node to edit its parameters.

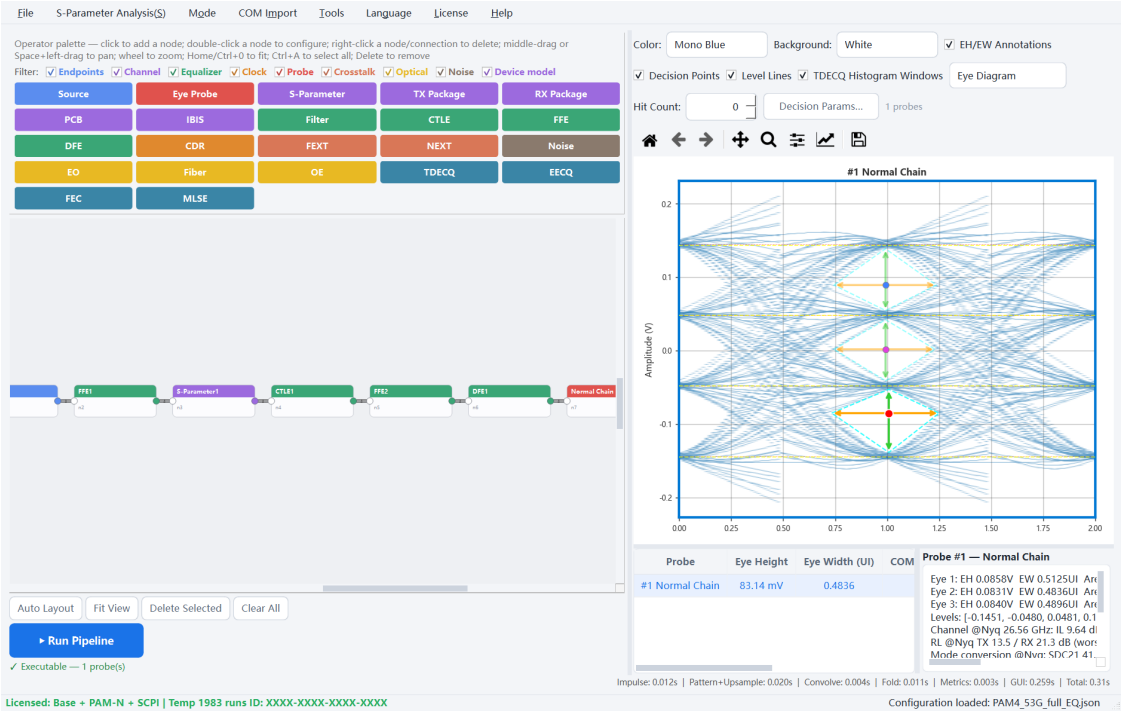

Fig. 8-1 Expert canvas: the link as a node graph

Step 3 Long runs (optical / TDECQ / FEC) take a while — while a run is in flight, a **progress bar** below the canvas shows percent, elapsed and estimated remaining time, the status line names the current stage, and the "**■ Pause**" / "**✕ Cancel**" buttons freeze or stop the run at any time (Fig. 8-2). Cancelling does not consume a trial run; the previous results stay on screen.

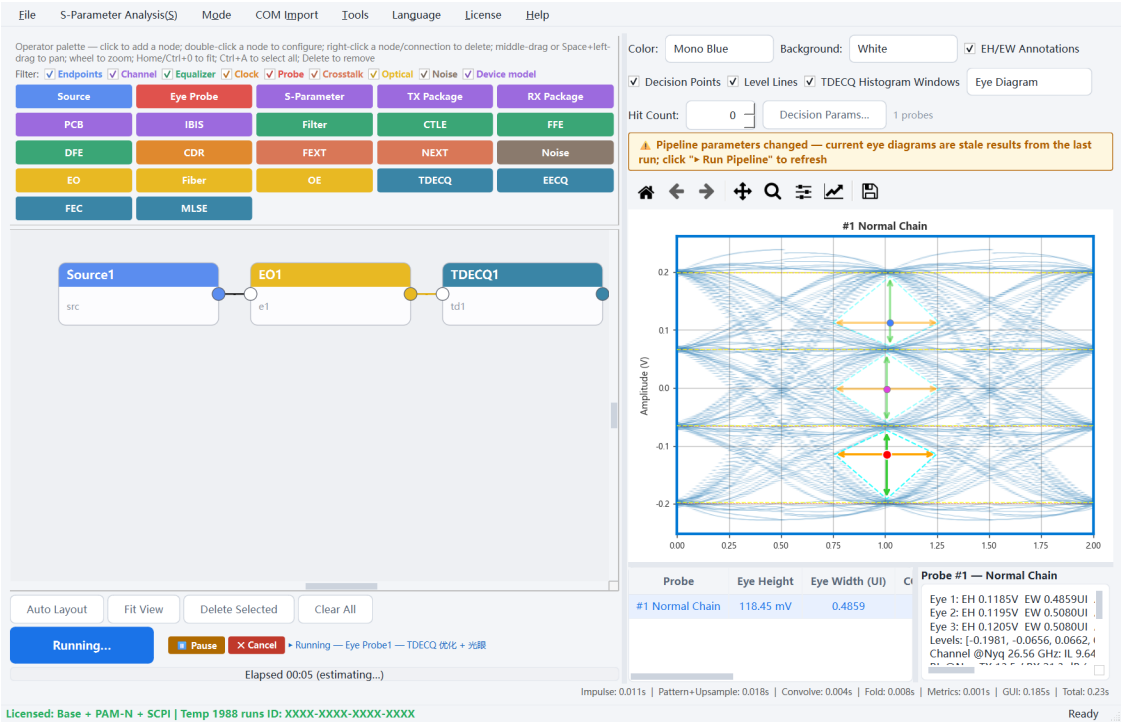

Fig. 8-2 A run in flight: progress bar + Pause / Cancel

Step 4 For an optical eye, use **File** → **Load Configuration...** to load demo\configs**QuickStart_Optical_TDECQ.json**. It opens Expert with **Source** → **EO** → **TDECQ** → **Probe**: PAM4 / 53.125 GBd / PRBS13 / 32 samples/UI, EO bandwidth 40 GHz and extinction ratio 6 dB; reference FFE optimization is enabled. Click Run Pipeline and wait for optical eyes and a finite TDECQ reading (Fig. 8-3). This is a minimal modulator example without fiber or OE; build a longer optical link separately when needed.

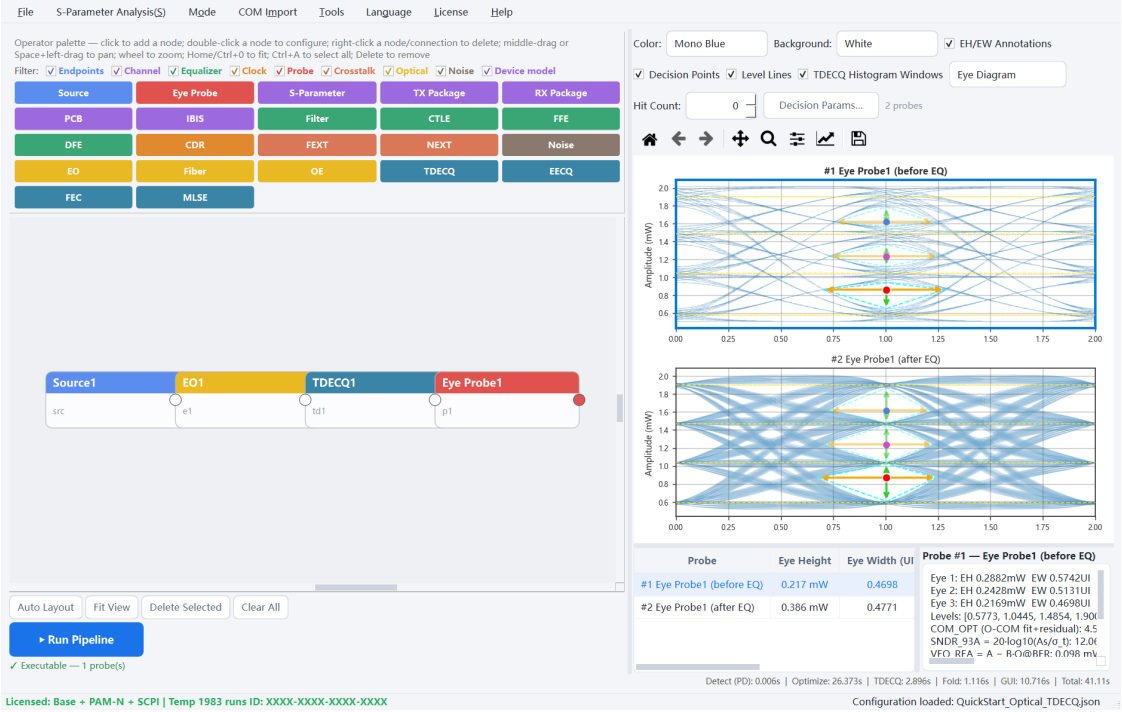

Fig. 8-3 Optical link + TDECQ reference receiver

Step 5 For **IBIS-AMI device models**, load demo\configs**QuickStart_IBIS_AMI.json** and click Run Pipeline. The chain is **Source** → **IBIS TX** → **S-parameter** → **IBIS RX** → **Probe**: NRZ / 53.125 GBd / PRBS13 / 32 samples/UI; TX pin A1, RX pin B1, both using GetWave. The configuration locates demo\ibis\rea_ref.ibs and its AMI / DLL files, plus lossy_channel_10dB.s4p (Fig. 8-4). These are synthetic reference models authored by REA, not real silicon. Double-click an IBIS node to inspect its pin, model, package and C_comp summary (Fig. 8-5). Vendor models and advanced IBIS 8.0 features are covered in Manual 11.6.

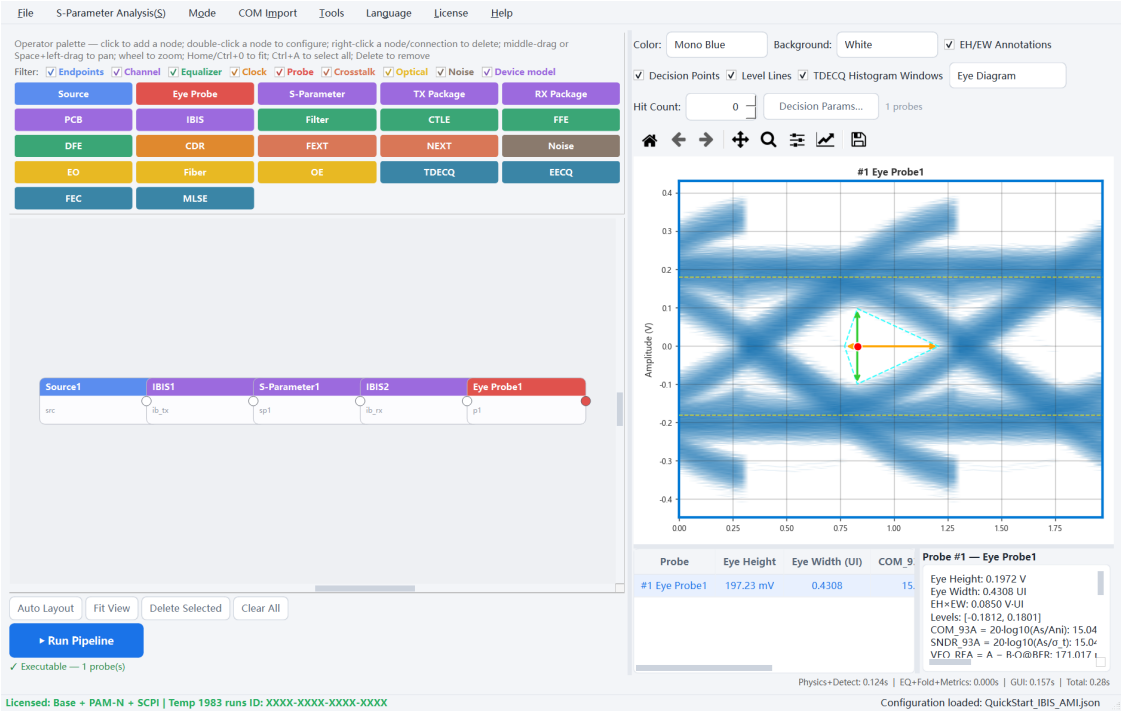

Fig. 8-4 IBIS-AMI device models: TX / RX nodes around the channel, the AMI receiver brings its own equalisation and CDR

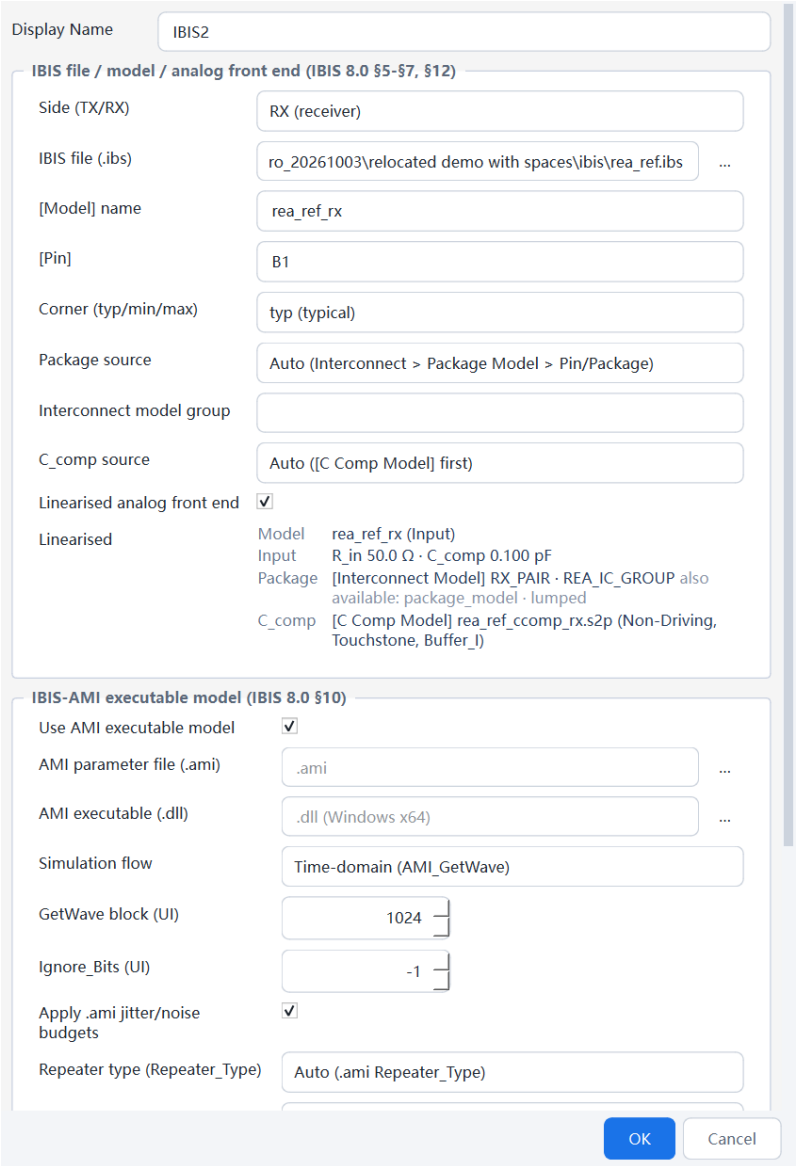

Fig. 8-5 IBIS node dialog: package source / interconnect model group / C_comp source and the linearised summary (incl. [C Comp Model])

Tip: Switching back to Mode → Normal never touches your Normal settings — the two modes are fully independent. Expert mode: Manual Chapter 11 (IBIS: 11.6); optical links and TDECQ: Manual Chapter 12.

Step 9 Automation (SCPI Remote Control)

REA can be driven like a bench instrument: enable **Tools** → **Enable SCPI Remote Control** and the program listens for standard SCPI commands on TCP port 5025 — loading files, changing parameters, running optimizations and reading metrics are all scriptable.

Step 1 Open demo**scpi_example.txt** — a complete command sequence from loading a channel to reading back COM. Replace <DEMO> with the absolute demo folder path. Send each command with a newline; read each query response before sending the next command. Start with the section labelled "First NRZ eye"; run other sections independently.

Note: INIT, optimization and pipeline execution require the **SCPI option license**; GUI trial runs cannot substitute for it. Queries and parameter changes can be tried directly.

→ GUI feedback: CONF:SIGN:RATE 28 immediately updates the GUI rate to 28 GBd. After INIT and *OPC? returning 1, the GUI shows the eye, and FETC:EYE:ALL? retrieves the same result.

Tip: The full command reference (500+ commands) is in Manual Chapter 13.

Step 10 Pre- and Post-FEC BER (optional)

What ultimately decides whether a link is good enough is the error rate before correction and what remains after it. One tick box puts both numbers right below the eye diagram, judged against the relevant IEEE 802.3 / PCIe specification.

Step 1 Restore **Normal + lossy_channel_10dB.s4p + PAM4_53G_full_EQ.json** as in Step 6 and generate the three equalized PAM4 eyes. This uses a PAM4 baseline rather than mixing Step 1 NRZ with the Electrical preset.

Step 2 Open **Measurements / FEC** on the left, choose **IEEE 802.3dj D3.1 Electrical**, set LANes to **4**, leave BERadded blank for the preset default, then tick **FEC analysis** and wait for the cards to update. If no result is shown, generate the eye. For an NRZ exercise, use the Step 1 baseline with Clause 91 KR4.

→ GUI feedback: A FEC results strip appears below the eye: four metric cards plus an error-distribution histogram

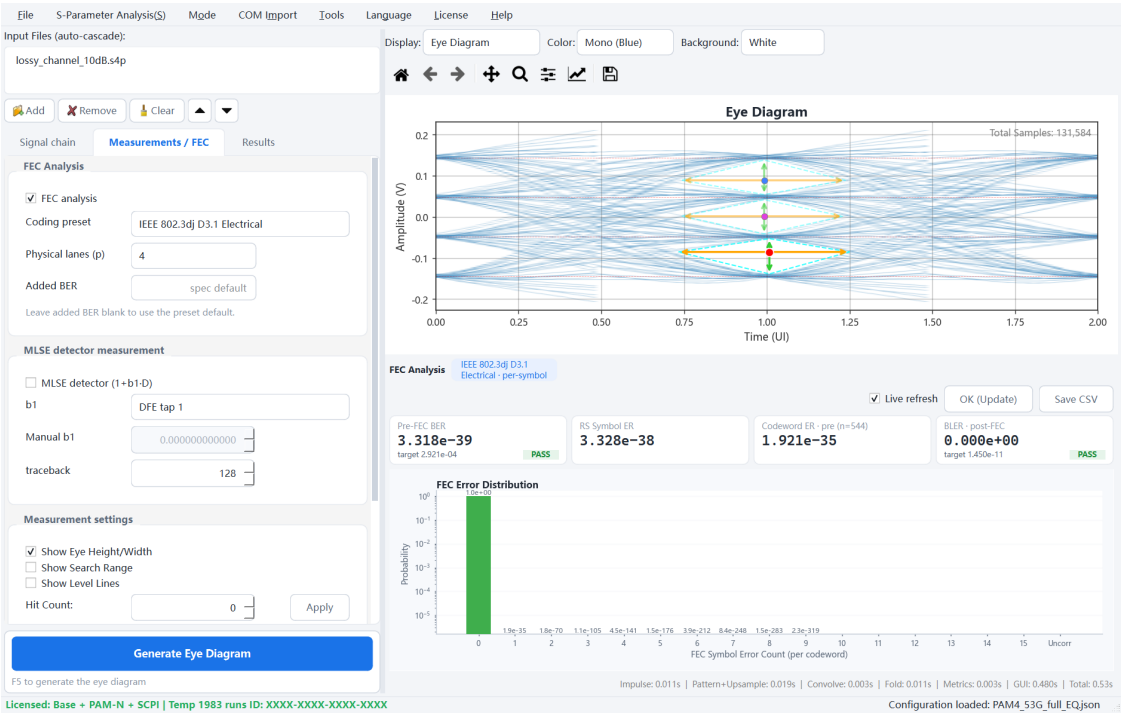

Figure 10-1 FEC results below the eye

The four cards are, from left to right, the pre-correction bit error ratio, the RS symbol error ratio, the codeword error ratio, and the **post-correction** codeword error ratio (BLER). That last one is the real Ethernet criterion, and the card prints the target together with a PASS / FAIL verdict.

> **Tip:** Open Decision Parameters, change the threshold and apply: FEC is recalculated. A worse threshold increases errors; reload the configuration to recover the baseline. Other presets that use Inner FEC currently give a hard-decision estimate, not a complete soft-decision decoder; see Manual Chapter 15.

Standard presets, BERadded, Expert-mode usage and the SCPI commands are covered in Manual Chapter 15.

Reference: How the 500 Trial Runs Are Counted

A fresh installation starts in trial mode with **500 runs**. NRZ and PAM-N share this one counter. Walking through the 10 steps below costs a few dozen runs, depending on automatic generation when parameters are enabled and how often you repeat an exercise. The table shows what does and does not cost a run.

What you do	Does it cost a run?
Generate one eye diagram (the button, or one triggered automatically)	1 run
Generate an eye with PAM4 / PAM-N	1 run (available during the trial)

What you do	Does it cost a run?
Press an optimize button (CTLE / FFE / DFE / joint / TDECQ)	Optimization itself consumes no runs; the eye generated when it finishes costs 1 run
Import COM results in one click (incl. auto optimization)	**Exactly 1 run for the whole flow**
Run an Expert mode pipeline	Same price as Normal: 1 run
Compute and view S-parameters (cascade / de-embed / split / convert / TDR / Smith)	**No runs consumed**
Export an S-parameter result to a file	**10 exports consume 1 run**
Change parameters, zoom, save/load configs, export images	**No runs consumed**
GPU / CUDA acceleration	**No runs consumed**; enabled by default, no separate option

Tip: Running out does not lock the software. S-parameters are still computed and displayed; you simply cannot export result files or generate new eyes. Activating any valid key restores everything immediately, and none of your settings are lost.

Note: SCPI remote control is not part of the trial. The SCPI server is always reachable for queries and configuration, but commands that actually run something - INIT / MEAS / optimization / COM import - require the separate SCPI option key; trial runs cannot substitute for it. Step 9 covers this.

To request a trial key, open "License -> License Management" and send the **Host ID** to the download site consultation: 106.14.76.130. Self-service grants 200 / 100 / 50 runs, once per tier, up to 350 in total; this is separate from the installation's built-in 500-run trial. For additional runs or commercial licensing, contact the author there or via the WeChat QR code in the license dialog. The complete licensing matrix is in Chapter 4 of the User Manual.

Where to Go Next

I want to…	Go to
Learn every panel and parameter systematically	Press F1 — the User Manual (15 chapters + glossary)
Understand licensing and metering in full	Manual Chapter 4 (per-feature metering table)
See what else the demo assets can do	demo\README.txt
Package models, CDR, noise budget and other advanced topics	Manual Chapter 10 onwards
Write automation scripts	Manual Chapter 13 + demo\scpi_example.txt

Enjoy!