Link Quality Aware Pathfinding for Chiplet Interconnects

Aaron Yen*, Jooyeon Jeong*, and Puneet Gupta, *Fellow, IEEE*

***Abstract*—As chiplet-based integration advances, designers must select among short-reach die-to-die interconnect technologies with widely varying shoreline and areal bandwidth density, energy per bit, reach, and raw bit error rate (BER). Meeting stringent delivered BER targets in chiplet systems requires error-correcting codes (ECC), but incurs energy, area, and throughput overheads. We develop a flow centered around RTL synthesis power and area estimations to support pathfinding of inter-chiplet links under a stringent 10^{-27} delivered BER target. We synthesize a parameterized Reed-Solomon code with CRC-64 and Go-Back-N retry logic to estimate the correction overhead for different transceiver bit error rates.**

Results show that ECC can materially change link comparisons under common figures of merit and that CRC+ARQ can reduce the required RS strength (and decoder overhead) at moderate BERs while still meeting stringent delivered-BER targets. We present a CP-SAT-based link assignment formulation that uses these ECC-corrected metrics under reach, delivered-bandwidth, and shoreline constraints in system-level optimization.

I. Introduction

Chiplet-based integration improves yield and enables heterogeneous composition, but shifts many bottlenecks onto die-to-die interconnects. Advanced packaging supports various interconnects, creating a broad design space with wide ranges of (i) shoreline bandwidth density (Gbps/mm perimeter), (ii) areal bandwidth density (Gbps/mm^2 of chip area), (iii) energy/bit (pJ/bit), (iv) reach, and (v) raw bit error rate (BER). While emerging die-to-die standards like UCIe [1], AIB [2], and BoW [3] exist, system-level requirements span short-reach (¡1,mm) on-package links to longer links across large interposers and wafer-scale packages.

Prior work on advanced packaging proposed interconnect metrics like shoreline bandwidth density and signaling figures of merit for heterogeneous integration [4], [5]. However, despite strict BER and latency requirements for intra-package communication, these metrics for chiplet interconnects do not yet account for ECC-related power, area, and throughput overheads needed for reliable communication.

To meet strict delivered-BER targets, links often combine forward error correction (FEC) with a cyclic redundancy check (CRC) and automatic repeat request (ARQ). Early-stage link comparisons, however, typically focus only on nominal physical metrics and ignore error-correction overhead. As link energy reaches the sub-pJ/bit regime, ECC logic can consume a significant fraction of total link energy and area, altering comparisons under common figures of merit [4].

*These authors contributed equally to this work.

To achieve high reliability with low latency variability (and thus smaller retransmission buffers), we focus on Reed-Solomon FEC (RS-FEC) [6], which provides strong burst-error correction and predictable performance under harsh channels. Standard treatments discuss FEC/ARQ tradeoffs and hybrid schemes [7], and prior work studied energy–reliability tradeoffs between correction and retransmission for on-chip communication [8], [9].

For CRC-based detection, we use CRC-64/ECMA-182 [10] for its low undetected-error probability, which depends on message length, polynomial, and error structure. Koopman [11] and the FAA integrity report [12] provide guidance beyond the uniform 2^{-k} approximation. At the system level, chiplet placement and interconnect co-optimization have been studied in prior work [13]. This paper makes contributions:

- A synthesis-based ECC characterization flow for RS-FEC, CRC64, and Go-Back-N retry that quantifies energy, area, and throughput overhead as a function of raw BER under a delivered-BER target.
- A link metric calculator that produces ECC-corrected throughput, energy, and area metrics suitable for system-level optimization.
- A CP-SAT [14], [15]-based link assignment tool that maps link technologies to chiplet nets under reach, bandwidth, and shoreline constraints.

II. Motivation

Chiplet link pathfinding often uses uncorrected transceiver metrics (e.g., energy/bit, bandwidth density) and reach constraints. In practice, many systems need delivered BER far below raw BER across operating conditions. Bridging this gap requires ECC, which adds three coupled costs: (1) *code-rate overhead* reducing goodput, (2) *energy overhead* from encoder/decoder logic and retry control, and (3) *area/throughput overhead* requiring multiple ECC blocks at high line rates.

We distinguish *post-FEC BER* (bit error rate after RS decoding, before CRC) from *delivered BER* (payload-bit error rate after CRC and ARQ). FEC-only targets post-FEC BER directly (and thus delivered BER in the absence of retry), while the hybrid FEC+CRC+ARQ scheme targets delivered BER by bounding CRC-undetected residual errors.

These costs are nonlinear with BER. As shown in Section IV, the RS code strength required to meet a fixed delivered-BER target increases rapidly as input BER degrades, which increases decoder energy and area. Protocol choices also matter: CRC64 and retry can relax the required RS strength at moderate BERs while still meeting a delivered-BER target.

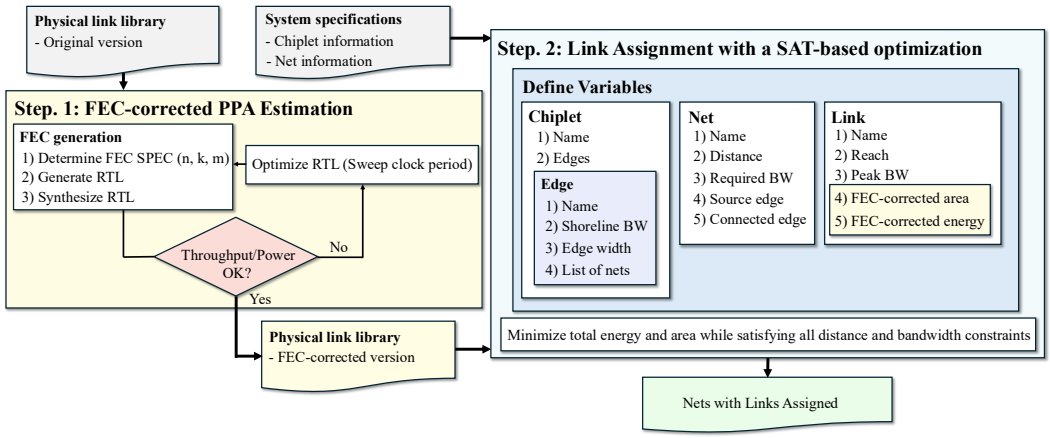


Fig. 1: Overall flow of link assignment.

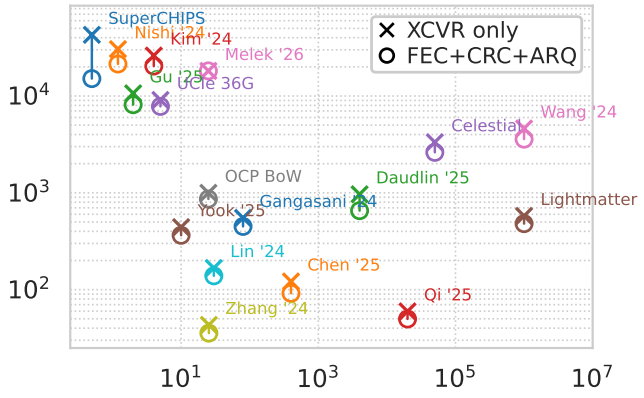


Fig. 2: FoM vs. reach (mm) for representative links. Crosses denote raw transceiver FoM, and open circles show ECC-corrected FoM using hybrid FEC+CRC+ARQ. FoM is defined as payload throughput density (Gbps/mm) divided by energy per delivered bit (pJ/bit).

Pathfinding decisions based only on raw link metrics can be misleading. Fair comparison requires incorporating BER-driven ECC overhead, and system-level link assignment must consider these corrected metrics along with physical layout constraints such as shoreline bandwidth and reach. Figure 2 illustrates how ECC correction can shift a common link figure of merit across representative reach regimes.[1]

III. Methodology

Figure 1 illustrates the overall workflow. The link metric calculator first characterizes each interconnect technology in terms of ECC-corrected metrics, which are then combined with net-to-edge mapping and physical distances. Finally, the CP-SAT optimization module assigns link technologies to nets.

A. Link Protection Stack and Metrics

We model a standard "FEC corrects, CRC detects, retry recovers" link protection stack: payload and protocol header bytes are protected by a cyclic redundancy check, then Reed-Solomon forward error correction parity is applied over the (header+payload+CRC) bytes. We use CRC64/ECMA-182 [10] and RS codes [6], [7] over GF(2^8). At the receiver, RS-FEC attempts to correct symbol errors, CRC64 detects remaining corruption, and a Go-Back-N (GBN) automatic repeat request mechanism retransmits frames that fail CRC.

We report two sets of link-level metrics: (1) *FEC-only*, where RS-FEC is sized to meet the delivered BER target without relying on retry and (2) *FEC+CRC+ARQ*, where CRC64 and retry are used to meet the delivered BER target and RS-FEC is only used as needed to keep the retry rate acceptably low. In the FEC-only mode we omit CRC and retry, so RS protects only header+payload bytes. In the FEC+CRC+ARQ mode, RS protects header+payload+CRC bytes.

B. Code Selection and Reliability Model

To evaluate the impact of bit errors on interconnects, we synthesize byte-oriented RS-FEC codes. We fix the codeword length to $N = 86$ symbols over GF(2^8), motivated by PCIe 6.0 FLIT-mode interleaving (≈85 B groups) [16], and vary the number of information symbols K such that the code can correct $t = \lfloor \frac{N-K}{2} \rfloor$ symbol errors per codeword.

For *FEC-only*, we select the highest-rate (largest K) RS$(86, K)$ code whose estimated post-FEC BER meets a target post-FEC BER. Without retry, post-FEC BER equals the delivered BER. Assuming i.i.d. symbol errors, the number of symbol errors per codeword X is $X \sim \text{Binomial}(N, p_{\text{sym}})$, with $p_{\text{sym}} = 1-(1-p_{\text{pre}})^M$. Post-FEC BER is approximated as the expected fraction of erroneous bits in uncorrectable codewords:

$$\text{BER}_{\text{post}} \approx \sum_{i=t+1}^{N} \left(\frac{1}{2} \cdot \frac{i}{N}\right) \Pr[X = i], \tag{1}$$

where we assume random bit corruption within each erroneous symbol (byte).

For *FEC+CRC+ARQ*, we account for two reliability failure modes: (1) *silent data corruption* (SDC) when CRC fails to

[1]Our modeling and assignment scripts are open sourced at https://github.com/nanocad-lab/chiplet-link-pathfinding

detect a residual error, and (2) *drop* when a frame exceeds a bounded number of CRC-detected retransmissions. As a simplifying collision model, we approximate the CRC undetected-error probability as p_{undet}. Unless otherwise noted we use $p_{\text{undet}} = 2^{-64}$ under random residual error patterns. Prior analyses show the undetected-error probability depends on message length, polynomial choice, and error structure [11], [12]. We apply a conservative "catastrophic" corruption model: if an undetected error is delivered, a fraction f_{wrong} of payload bits are incorrect.

Let $p_{\text{frame\_fail}}$ denote the per-attempt probability that an ARQ frame still contains corruption after RS decoding (and would therefore fail CRC absent a miss). A CRC-detected failure occurs with probability

$$p_{\text{det}} \approx p_{\text{frame\_fail}} \cdot (1 - p_{\text{undet}}). \tag{2}$$

Under independent attempts and retry-on-detect, the probability that a *delivered* frame is undetected is $p_{\text{frame\_fail}} \cdot p_{\text{undet}}/(1 - p_{\text{det}})$, giving

$$\text{BER}_{\text{delivered}} \approx f_{\text{wrong}} \cdot \frac{p_{\text{frame\_fail}} \cdot p_{\text{undet}}}{1 - p_{\text{det}}}. \tag{3}$$

To bound retries, we cap retransmissions to at most R (so the total number of attempts is $A = R + 1$). The frame-drop probability is then

$$p_{\text{drop}} \approx p_{\text{det}}^{A}. \tag{4}$$

To map drops into a bit-level budget compatible with $\text{BER}_{\text{target}}$, we use a 1-bit-equivalent model $\text{BER}_{\text{drop,eff}} = p_{\text{drop}}/(8P)$, where P is payload bytes.

Given $\text{BER}_{\text{target}}$, we derive a frame-fail budget $p_{\text{frame\_target}}$ as the tighter of the SDC and drop constraints, and select the highest-rate $\text{RS}(86, K)$ such that $p_{\text{frame\_fail}} \leq p_{\text{frame\_target}}$. Unless otherwise noted, we use $\text{BER}_{\text{target}} = 10^{-27}$ (UCIe-motivated [1]), $f_{\text{wrong}} = 0.5$, and $R = 1$ (one retry) for the FEC+CRC+ARQ mode.

C. Frame-Fail Probability and Goodput

We treat RS-FEC as a *streaming* byte code: parity overhead is attributed proportionally by code rate (K/N) without per-frame RS padding waste. Let P be payload bytes, H header bytes, and C CRC bytes (8 for CRC64). We use $P = 256\,\text{B}$ unless otherwise noted (PCIe 6.0 FLIT size [16]). The RS-protected data bytes per ARQ frame are:

$$D = P + H + C. \tag{5}$$

Given a raw (pre-FEC) bit error probability p_{pre}, we approximate the symbol error probability as

$$p_{\text{sym}} = 1 - (1 - p_{\text{pre}})^{M}. \tag{6}$$

This symbol-error mapping and the binomial tail model below assume independent bit errors. We model the probability of codeword decode failure as a binomial tail over symbol errors:

$$p_{\text{blk\_fail}} = \Pr[X > t], \quad X \sim \text{Binomial}(N, p_{\text{sym}}). \tag{7}$$

Under the streaming model, an ARQ frame spans an *effective* number of codewords $B_{\text{eff}} = D/K$ (which can be fractional). We then approximate the frame-fail probability as:

$$p_{\text{frame\_fail}} \approx 1 - (1 - p_{\text{blk\_fail}})^{B_{\text{eff}}}. \tag{8}$$

Fractional B_{eff} corresponds to cross-frame streaming RS coding and ignores per-frame padding overhead. The wire bytes transmitted per attempt under the streaming code-rate model are:

$$\text{wire\_bytes/attempt} \approx D \cdot \frac{N}{K}. \tag{9}$$

Assuming independent attempts and retry-on-detect, the expected number of attempts per delivered frame is $\mathbb{E}[\text{attempts}] \approx 1/(1 - p_{\text{det}})$ and the goodput is:

$$\text{goodput} = \frac{P}{(\text{wire\_bytes/attempt}) \cdot \mathbb{E}[\text{attempts}]}. \tag{10}$$

This retry model approximates per-frame independent retransmission. We synthesize a Go-Back-N controller with a replay window sized to the bandwidth–delay product, but in analysis we approximate retransmission cost using a geometric attempts factor and ignore go-back-N pipeline flushes or window-full stalls, which are negligible in our rare-retry target regime [17], but higher retry rates or larger RTT/BDP may require a BDP-aware ARQ throughput model.

D. Synthesis

Using the code selection method above, we choose a worst-case input pre-FEC BER of 10^{-3} and a post-FEC BER target of 10^{-27}, which sets the worst-case code to $\text{RS}(86, 44)$. We synthesize $\text{RS}(86, K)$ codes from $\text{RS}(86, 44)$ to $\text{RS}(86, 84)$ in steps of $t = 1$. Synthesis is performed with Synopsys Design Compiler [18] using the ASAP7 7nm predictive PDK [19]. We use a fixed clock period that meets timing for the worst-case code (0.8 ns).

In addition to RS-FEC, we synthesize CRC64 append/check blocks and a parameterized GBN retry controller. The CRC/retry sweep spans payload size and round-trip time (RTT), which determines the retry window depth and replay buffering. Reported GBN area/power includes both controller logic and the synthesized replay buffer (implemented using SRAM macros when available and inferred storage otherwise).

To compare different interconnect links, we normalize ECC area and energy by effective throughput. RS-FEC throughput is decoder-limited at the synthesized clock, while CRC/retry assumes single-frame-per-cycle at 500 MHz. Dynamic power is estimated with a vectorless activity model (randomized data, clock gating enabled, leakage excluded), so pJ/bit values should be interpreted as estimates. For RS-FEC receiver energy, we approximate correction activity by weighting incremental decoder energy beyond syndrome by $p_{\text{corr}} = \Pr[1 \leq X \leq t]$ under the i.i.d. symbol-error model.

We treat chiplet transceivers as largely mixed-signal blocks and do not scale published transceiver metrics across nodes. We synthesize ECC logic in ASAP7 (7 nm) and scale ECC energy and area to a 3 nm using public TSMC guidance.

E. Chiplet Link Assignment

We formulate a constrained optimization problem that maps logical interconnect nets to physical link technologies while minimizing system-level cost. Each net must be assigned exactly one link type that satisfies its reach and bandwidth requirements, and the aggregate shoreline width consumed on each chiplet edge must not exceed the available physical I/O width. In the chiplet, the available physical I/O width depends on which edge the connection is made to, as the required routing distance varies accordingly. We model the I/O-available length as half of the chiplet perimeter.

We solve this problem using the OR-Tools constraint programming satisfiability solver [14], [15]. Table I summarizes the notation. For each net $n \in \mathcal{N}$ and link type $l \in \mathcal{L}$, the binary variable $x_{n,l}$ indicates whether n uses link type l. The shoreline density BW_l^{link}, energy ($Energy_l$), and area density ($BW_{density}$) parameters incorporate ECC overhead, so that the assignment problem operates on delivered payload bandwidth and energy. We normalize the objective using the total power and total area (λ_P and λ_A).

TABLE I: Notation for CP-SAT-based I/O link assignment.

Input parameters	
$\mathcal{C}, \mathcal{L}, \mathcal{N}$	Set of chiplets, available link types and nets
$\mathcal{N}/\mathcal{W}_{c,e}$	Net/Edge width of chiplet c edge e
d_n	Actual physical distance required for net n [mm]
r_l	Maximum reach distance supported by link type l [mm]
$Energy_l$	Energy per bit for link type l [pJ/bit]
BW_l^{link}	Shoreline bandwidth density for link type l [Gbps/mm]
BW_n^{req}	Required bandwidth for net n [Gbps]
$BW_{density}$	Areal bandwidth density for link type l [Gbps/mm^2]
$\lambda_{P/A}$	Total system power/area of chiplets/wafer [W, mm^2]
Decision variables	
$x_{n,l}$	1 if net n uses link type l, 0 otherwise
$w_{n,l}$	Shoreline width allocated to net n if it uses link type l

The objective minimizes a weighted sum of total link power and total transceiver area:

$$\min \sum_{n\in\mathcal{N}} \sum_{l\in\mathcal{L}} \big((Energy_l \cdot BW_n^{req})/\lambda_P \cdot x_{n,l} + (BW_n^{req}/BW_{density})/\lambda_A \cdot x_{n,l}\big). \tag{11}$$

IV. Experimental Results

A. RS-FEC Energy, Rate, and Area Trends

Figure 3 shows how RS-FEC decoder energy per information bit and effective code rate vary with input BER for RS$(86, K)$ codes when selecting the highest-rate code that meets a post-FEC BER target of 10^{-27}. As BER degrades, stronger codes (smaller K) are required, reducing code rate and increasing decoder energy.

Figure 4 reports RS-FEC information-throughput density vs input BER for a post-FEC BER target of 10^{-27} (ASAP7, with decoder-limited throughput), normalized to both a 1-D (Gbps/mm) and 2-D (Gbps/mm^2) footprint. The Gbps/mm metric uses a square-footprint proxy (width $= \sqrt{\text{area}}$).

Algorithm 1 Chiplet I/O link assignment via CP-SAT

Initialize model: $\mathcal{M} \leftarrow$ CpModel()
for all nets $n \in \mathcal{N}$ and link types $l \in \mathcal{L}$ **do**
  Define binary variable $x_{n,l} \in \{0, 1\}$
  Define width variable $w_{n,l} \in [0, W_{\max}]$
  Constraint 1: Reachability
  if $d_n > r_l$ **then**
    $x_{n,l} = 0$
  end if
  Constraint 2: Bandwidth capacity
  $w_{n,l} \cdot BW_l^{link} \geq BW_n^{req}\, x_{n,l}$
end for
for all chiplets $c \in \mathcal{C}$ and edges $e \in \mathcal{E}$ **do**
  Constraint 3: Edge capacity limit
  $\sum_{n\in\mathcal{N}_{c,e}} \sum_{l\in\mathcal{L}} w_{n,l} \leq W_{c,e}$
end for
for all nets $n \in \mathcal{N}$ **do**
  Constraint 4: Single link per net
  $\sum_{l\in\mathcal{L}} x_{n,l} = 1$
end for
Objective: Minimize Eq. (11)
Solve model $\mathcal{M}$ with CP-SAT solver
Return: Assigned link variables $x_{n,l}$ and $w_{n,l}$

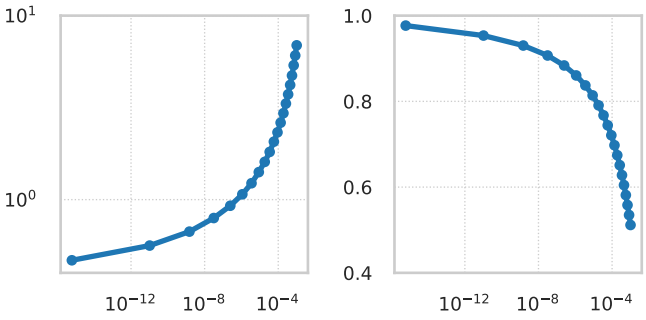


Fig. 3: Selected RS-FEC decoder energy per info bit and code rate vs input BER for a post-FEC BER target of 10^{-27} (ASAP7 synthesis sweep, highest-rate K meeting target). Left: energy per info bit [pJ/bit]. Right: code rate (K/N).

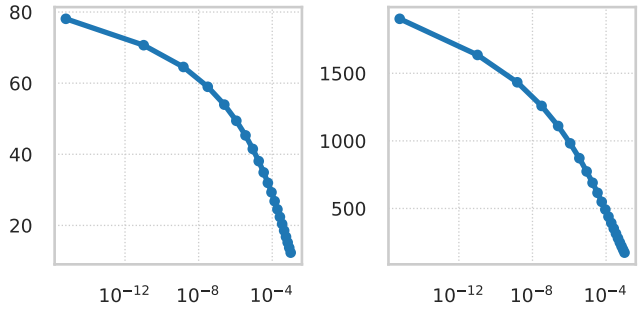


Fig. 4: RS-FEC throughput density vs. input BER (ASAP7, post-FEC BER = 10^{-27}). X-axis: pre-FEC BER. Left: throughput density (Gbps/mm). Right: throughput density (Gbps/mm^2).

B. FEC-only vs FEC+CRC+ARQ Tradeoffs

Figure 5 compares two protection modes for a representative frame size (P=256 B payload, H=8 B header, CRC64) under a delivered BER target of 10^{-27}: *FEC-only* (no retry) and

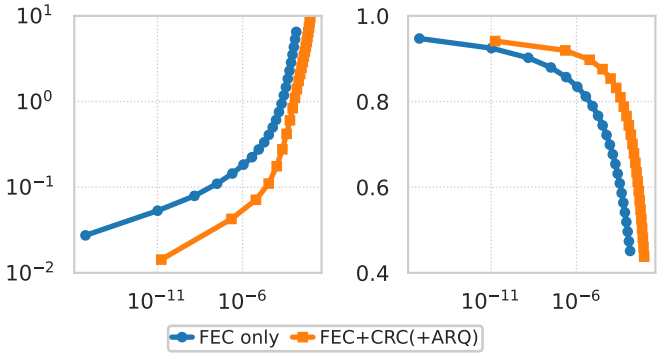


Fig. 5: ECC energy and goodput vs input BER (ASAP7), comparing FEC-only and a no-drop (unbounded-retry) FEC+CRC+ARQ hybrid. Left: energy per payload bit [pJ/bit]. Right: goodput rate.

FEC+CRC+ARQ (retry on CRC fail with RS-FEC sized to meet the retry budget). The FEC+CRC+ARQ scheme relaxes the required RS strength at moderate BERs, improving goodput and reducing ECC energy while still meeting the delivered BER target. At very low input BER, it can fall back to a no-FEC operating point ($K = N$).

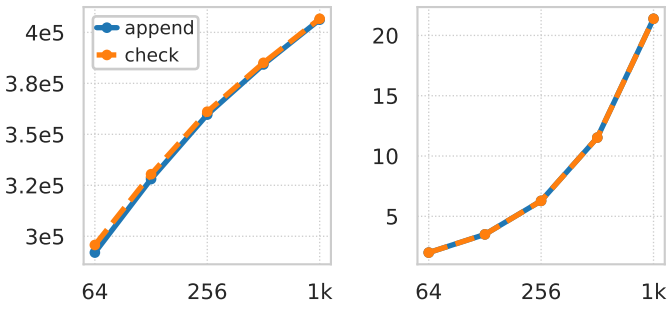
For example, at input BER $\approx 10^{-4}$, FEC-only selects RS$(86, 62)$. Under an unbounded-retry (no-drop) hybrid model, FEC+CRC+ARQ selects RS$(86, 78)$, reducing ECC energy from ≈ 0.61 to ≈ 0.18 pJ/payload-bit and improving goodput from ≈ 0.70 to ≈ 0.85. Enforcing a one-retry limit (R=1) and budgeting drops into the same 10^{-27} target instead selects RS$(86, 72)$ (ECC energy ≈ 0.31 pJ/payload-bit, goodput ≈ 0.79).

To ensure numerical stability for rare RS decode-failure probabilities, binomial tails are computed via log-domain summation (instead of $1 - \text{CDF}$ subtraction) and verified against high-precision references for representative RS$(86, K)$ points. Relative error remains below 10^{-12} for probabilities down to $\sim 10^{-27}$, validating the computation under the i.i.d. error assumption.

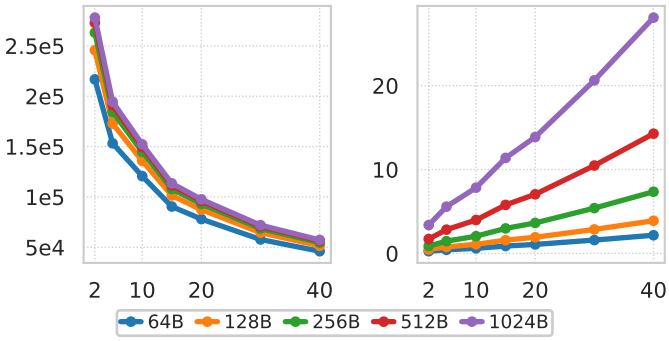
C. CRC64 and Go-Back-N Retry Synthesis

We synthesize CRC64 append/check and a parameterized GBN retry controller across payload size and RTT. The synthesized GBN block includes the replay buffer memory sized to the outstanding window implied by the bandwidth–delay product (BDP). Reported area/power includes this buffer (SRAM macros when available, inferred storage otherwise). For the baseline (P=256 B, H=8 B, CRC64, RTT=10 ns at 500 MHz), the window is 7 frames, corresponding to ≈1.9 KB of replay storage. Figure 6 shows representative ASAP7 sweeps.

Table II reports a concrete operating point (256 B payload, 8 B header, 10 ns RTT) showing that CRC64 and retry logic have small per-bit energy and area overhead relative to RS-FEC at moderate-to-high BERs. Energy per payload bit is computed from dynamic power as $E = (P_{\text{dyn}}/f_{\text{clk}})/(8P)$ at $f_{\text{clk}} = 500$ MHz (dynamic power only, leakage excluded).



(a) CRC64 append/check (left: throughput density [Gbps/mm^2] vs payload bytes, right: dynamic power [mW] vs payload bytes).



(b) GBN retry controller (left: throughput density [Gbps/mm^2] vs RTT [ns], right: dynamic power [mW] vs RTT [ns]).

Fig. 6: CRC64 and GBN retry synthesis sweeps (ASAP7, 500 MHz, vectorless activity model).

TABLE II: CRC64 and GBN retry synthesis cost at 256 B payload, 8 B header, 10 ns RTT (ASAP7, 500 MHz).

Block	**Area** [μm^2]	**Dyn. Pwr.** [mW]	**Energy** [pJ/payload-bit]
CRC append	2847	6.28	0.00614
CRC check	2836	6.28	0.00614
GBN retry	7071	2.05	0.00201

Power uses a vectorless activity model with randomized data inputs (static probability 0.5, toggle rate 0.5 toggles/cycle) and non-stalling control (valid/ready asserted).

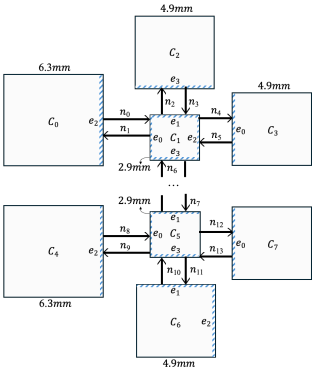
D. Link Assignment with Various Configurations

1) Experimental Setup and Link Characteristics: Table III summarizes the link library used in our case studies, including original process node, reach (mm), raw BER, and link type (E for electrical, O for optical). We report ECC-corrected delivered throughput density ($Gbps/mm^2$) and energy (pJ per delivered payload bit) using ASAP7-synthesized ECC (7 nm) and a 3 nm ECC scaling case study, while leaving transceiver metrics unchanged.

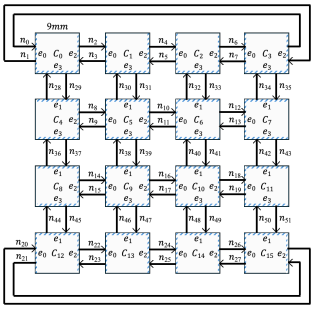
Overall, CRC reduces delivered energy/bit and increases delivered throughput density by allowing weaker RS codes at a fixed delivered-BER target. When the raw BER is already extremely low (e.g., Melek [32]), FEC-only and FEC+CRC become similar. Long-reach optical links remain higher-energy than short-reach electrical links for chiplet-scale distances.

TABLE III: Final link metrics comparison for representative links, using ASAP7-synthesized ECC (7 nm) and a 3 nm ECC scaling case study. Transceiver energy and bandwidth metrics are taken as published and are not scaled across nodes. Due to space limitations, $Gbps/mm$ and $Gbps/\mathrm{mm}^2$ are abbreviated as G/mm and G/mm^2 in the table.

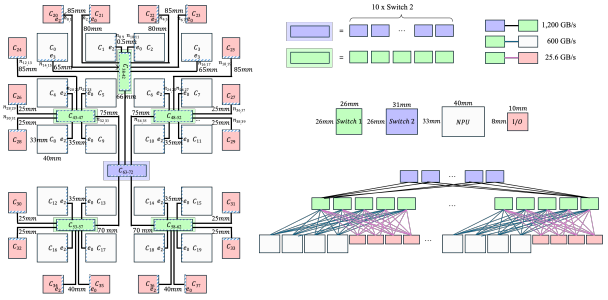
Link Name	Reach (mm)	Proc. (nm)	Raw BER	Link Type	7nm (FEC Only)			7nm (FEC+CRC)			3nm (FEC Only)			3nm (FEC+CRC)		
					G/mm	G/mm^2	pJ/b	G/mm	G/mm^2	pJ/b	G/mm	G/mm^2	pJ/b	G/mm	G/mm^2	pJ/b
SuperCHIPS [20]	0.5	16	1e-14	E	1144	1608	0.08	1103	1719	0.07	1144	4856	0.05	1103	5172	0.05
Hsu '21 [21]	1.0	7	1e-25	E	5187	1020	0.50	4998	2063	0.50	5187	1605	0.48	4998	2100	0.50
Nishi '23 [22]	1.2	5	1e-25	E	10744	1420	0.33	10353	5053	0.33	10744	2885	0.32	10353	5280	0.32
Nishi '24 [23]	1.2	5	1e-12	E	5530	1043	0.25	5332	1074	0.25	5530	1843	0.22	5332	1843	0.23
Kang '25 [24]	1.5	28	1e-12	E	17163	486	1.21	16547	483	1.24	17163	610	1.18	16547	595	1.22
Gu '25 [25]	2.0	3	1e-15	E	3661	1442	0.43	3530	1525	0.43	3661	3605	0.40	3530	3739	0.41
Kim '25/26 [26]	2.0	28	1e-12	E	8734	1560	0.34	8421	1662	0.34	8734	4442	0.31	8421	4690	0.32
Wang '25 [27]	3.0	28	1e-16	E	10256	1189	1.27	9882	2944	1.30	10256	2069	1.25	9882	3019	1.29
Kim '24 [28]	4.0	28	1e-12	E	11442	1526	0.53	11031	1623	0.54	11442	4181	0.51	11031	4390	0.52
GLink 2.3LL [29]	5.0	5	1e-20	E	13990	1345	0.33	13480	4189	0.33	13990	2591	0.32	13480	4343	0.33
UCIe 36G [30]	5.0	5	1e-20	E	3647	699	0.45	3514	1050	0.45	3647	931	0.43	3514	1059	0.44
Yook '25 [31]	10	4	1e-15	E	641	552	1.65	618	550	1.70	641	716	1.62	618	700	1.67
Melek '26 [32]	25	3	1e-27	E	5270	4216	0.29	5270	4216	0.29	5270	4216	0.29	5270	4216	0.29
OCP BoW [33]	25	12	1e-20	E	488	1564	0.54	471	7662	0.55	488	3549	0.52	471	8194	0.54
Vandersand '25 [34]	25	3	1e-27	E	448	393	0.52	448	393	0.52	448	393	0.52	448	393	0.52
Zhang '24 [35]	25.4	28	1e-12	E	59	505	1.54	56	502	1.59	59	639	1.51	56	624	1.57
Lin '24 [36]	30	28	1e-12	E	344	355	2.32	332	350	2.39	344	416	2.29	332	405	2.37
Poon '21 [37]	30	7	1e-12	E	830	496	1.35	800	493	1.39	830	625	1.33	800	610	1.37
Gangasani '24 [38]	80	5	1e-9	E	564	358	1.26	557	372	1.24	564	432	1.22	557	435	1.22
Chen '25 [39]	400	4	2.5e-6	E	539	186	6.55	564	209	6.15	539	218	6.44	564	234	6.10
Daudlin '25 [40]	4e3	28	6e-8	O	278	897	0.51	283	1146	0.43	278	1994	0.44	283	2370	0.40
Qi '25 [41]	2e4	45	1e-12	O	162	220	3.04	156	215	3.14	162	243	3.01	156	235	3.12
Celestial [42]	5e4	4	1e-8	O	7256	527	2.75	7179	563	2.74	7256	737	2.70	7179	754	2.71
Wang '24 [43]	1e6	28	1e-10	O	1860	1111	0.54	1839	1289	0.51	1860	2348	0.50	1839	2579	0.49
Lightmatter [44]	1e6	45	1e-9	O	1395	1015	2.87	1379	1161	2.87	1395	1959	2.83	1379	2113	2.85



(a) Example 1 [45]



(b) Example 2 [45]



(c) Example 3 [46]

Fig. 7: Link assignment abstractions: Examples 1 & 2 (top) and Example 3 (bottom). Each chiplet edge has a shoreline width budget, and each net connects two edges with a required bandwidth and floorplan distance.

2) Evaluated System Configurations: Figure 7 shows three experimental setups. Examples 1 & 2 come from a wafer-scale architecture [45]: Example 1 models a reduced two-tile subset (each tile with one compute die and two memory dies for more net diversity), and Example 2 extends to a 4×4 tile array with extra inter-row connections to study link selection in more complex topologies. Example 3 is from prior work [46], where die dimensions are unspecified, so dimensions are heuristically set to match the reported total area.

Example 1, Example 2, and Example 3 assume total areas/powers of 175.42 mm^2/11.27 W, 1,403.36 mm^2/90.12 W, and 70,000 mm^2/15,000 W, respectively. The values for Example 1 and Example 2 are linearly scaled from [45] based on tile count and are therefore approximate.

3) Results and Optimization Analysis: To evaluate our optimization framework, we compare it with a simplified baseline in Fig. 8. The Simple method selects links independently per net based only on reach constraints, choosing the feasible link with the highest peak bandwidth under shoreline limits. Although this greedy approach ensures feasibility, it can be inefficient in area and energy.

As shown in Fig. 8, for Examples 1 and 2, the All-Link-Types strategy yields the same selections as the Electrical-Only case. This is due to short chiplet distances (0.5–6 mm) and moderate bandwidth demands, where electrical links are more efficient. In contrast, Example 3 features longer distances (25–77 mm) and higher bandwidth requirements, leading to a more demanding topology. Under these conditions, the optimizer selects optical links to improve both energy and area efficiency.

Table IV presents the results considering ECC. For Example 3, we perform case studies by scaling bandwidth and net distances. Case (1) (BW $\times$ 1.4) yields the same link selections

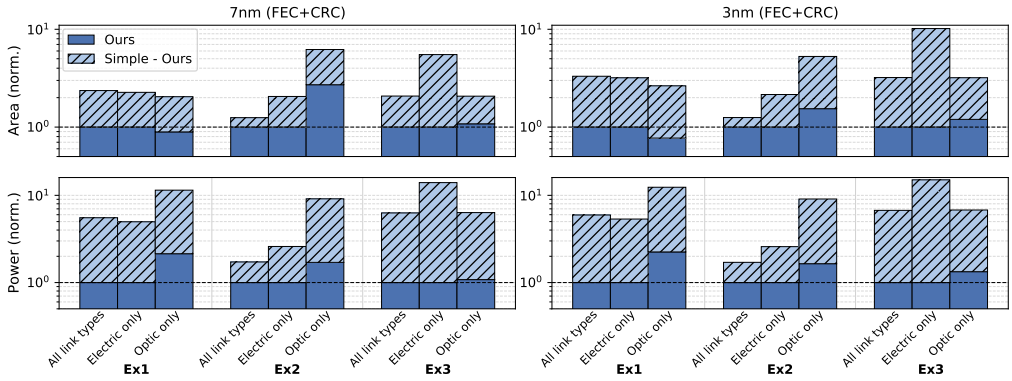


Fig. 8: Bar-graph summary of FEC+CRC results. Values are normalized per (Example, metric, node) to the "Ours" result under All-Link-Types. Bars are stacked: solid = Ours, hatched = (Simple−Ours), bar top = Simple.

TABLE IV: Integrated link selection results with total energy and area comparison for CRC-enabled links scaled to 7nm

Ex. / Case	# of Nets	Dist. (mm)	BW (Gbps)	All Link Types	Electrical Only	Optical Only
Example 1	12	0.5	2033–7228	SuperCHIPS [20] / Nishi '24 [23]	SuperCHIPS [20]/ Nishi '24 [23]	Wang '24 [43]
	2	6	1604	Melek '26 [32]	Melek '26 [32]	Wang '24 [43]
	Total Power [W] / Area [mm^2]			14.07 / 25.55	14.07 / 25.55	30.15 / 22.82
Example 2	48	5	1604	Melek '26 [32]	Melek '26 [32]	Daudlin '25 [40]
	4	60–80	1604	Daudlin '25 [40]	Gangasani '24 [38]	Daudlin '25 [40]
	Total Power [W] / Area [mm^2]			25.10 / 11.93	30.28 / 17.76	36.06 / 36.38
Example 3	80	0.5–25	26–600	Melek '26 [32]	Melek '26 [32]	Daudlin '25 [40]
	20	0.5–25	26–600	SuperCHIPS [20]	SuperCHIPS [20]	Daudlin '25 [40]
	730	35–75	26–1200	Daudlin '25 [40]	Gangasani '24 [38]	Daudlin '25 [40]
	50	35–75	26–1200	Wang '24 [43]	Gangasani '24 [38]	Wang '24 [43]
	Total Power [W] / Area [mm^2]			313.47 / 310.76	881.92 / 958.95	318.08 / 313.16
Case Studies of Example 3						
(1) BW × 1.4	100	0.5–25	36–840	Melek '26 [32] / SuperCHIPS [20]	Melek '26 [32] / SuperCHIPS [20]	Daudlin '25 [40]
	780	35–75	36–1680	Daudlin '25 [40]/ Wang '24 [43]	Gangasani '24 [38]/ Chen '25 [39]	Daudlin '25 [40]/ Wang '24 [43]
	Total Power [W] / Area [mm^2]			449.68 / 428.60	1482.23 / 1395.36	456.13 / 431.95
(2) BW × 0.5	100	0.5–25	13–300	Melek '26 [32] / SuperCHIPS [20]	Melek '26 [32] / SuperCHIPS [20]	Daudlin '25 [40]
	780	35–75	13–600	Daudlin '25 [40]	Gangasani '24 [38]	Daudlin '25 [40]
	Total Power [W] / Area [mm^2]			154.32 / 156.83	440.96 / 479.48	156.62 / 158.02
(3) Dist × 2	20	1–50	52–1200	Melek '26 [32]	Melek '26 [32]	Daudlin '25 [40]
	860	70–150	26–1200	Daudlin '25 [40]/ Wang '24 [43]	Gangasani '24 [38]/ Chen '25 [39]	Daudlin '25 [40]/ Wang '24 [43]
	Total Power [W] / Area [mm^2]			316.37 / 309.35	3899.79 / 1602.47	318.08 / 313.16
(4) Dist × 0.25	20	0.125–6.25	26–600	SuperCHIPS [20]	SuperCHIPS [20]	Daudlin '25 [40]/ Wang '24 [43]
	860	8.75–18.75	26–1200	Melek '26 [32]	Melek '26 [32]	Daudlin '25 [40]/ Wang '24 [43]
	Total Power [W] / Area [mm^2]			207.53 / 88.01	207.53 / 88.01	318.08 / 313.16

as the baseline because the same high-bandwidth links remain required to satisfy the stricter bandwidth constraints. Case (2) (BW $\times$ 0.5) removes some selected optical links, as reduced bandwidth demands make electrical links sufficient. Case (3) (Dist $\times$ 2) increases reliance on optical or high-performance links. For example, Chen '25 [39] is selected in case (3) to handle longer distances. Case (4) (Dist $\times$ 0.25) favors low-power electrical links such as Melek '26 [32] . Overall, these results show that the optimizer adapts link choices to minimize system-level power and area under varying constraints.

V. Conclusion

This paper presents a link-quality-aware pathfinding method that models ECC overhead for chiplet interconnects. Using RS$(86, K)$ FEC, CRC64, and Go-Back-N blocks, we quantify how ECC energy, goodput, and area/throughput vary with input BER under a delivered-BER target. Results show that ECC overhead can change link rankings and that CRC+ARQ can reduce RS strength—and decoder overhead—at moderate BERs while meeting stringent BER targets. We present a CP-SAT link assignment using ECC-corrected metrics with reach, bandwidth, and shoreline constraints, while leaving burst-error traces, BDP-aware ARQ throughput effects, latency, and routing congestion to future work.

References

[1] UCIe Consortium, "UCIe specifications."
[2] Intel Corporation, "Advanced interface bus (AIB) specification."
[3] Open Compute Project, "ODSA BoW PHY specification."
[4] S. Jangam and S. S. Iyer, "A signaling figure of merit (s-FoM) for advanced packaging," *IEEE Trans. Compon. Packaging Manuf. Technol.*, vol. 10, no. 10, pp. 1758–1761, Oct. 2020.
[5] S. Jangam and S. S. Iyer, "Silicon-interconnect fabric for fine-pitch ($\leq$10 µm) heterogeneous integration," *IEEE Trans. Compon. Packaging Manuf. Technol.*, vol. 11, no. 5, pp. 727–738, May 2021.
[6] I. S. Reed and G. Solomon, "Polynomial codes over certain finite fields," *J. Soc. Indust. Appl. Math.*, vol. 8, no. 2, pp. 300–304, 1960.
[7] S. Lin and D. J. Costello, *Error Control Coding: Fundamentals and Applications*, 2nd ed. Pearson, 2004.
[8] D. Bertozzi, L. Benini, and G. De Micheli, "Energy-reliability trade-off for NoCs," in *Networks on Chip*. Springer, 2003, pp. 107–129.
[9] D. Bertozzi, L. Benini, and G. De Micheli, "Error control schemes for on-chip communication links: the energy-reliability tradeoff," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 24, no. 6, pp. 818–831, 2005.
[10] Ecma International, "ECMA-182 (crc-64/ecma-182) standard," 1st edition, Dec. 1992.

[11] P. Koopman and T. Chakravarty, "Cyclic redundancy code (crc) polynomial selection for embedded networks," in *International Conference on Dependable Systems and Networks, 2004*. IEEE, 2004, pp. 145–154.

[12] P. Koopman, K. Driscoll, and B. Hall, "Selection of cyclic redundancy code and checksum algorithms to ensure critical data integrity," Carnegie Mellon University, Tech. Rep., 2015, prepared for the U.S. Federal Aviation Administration (FAA).

[13] A. Coskun *et al.*, "Cross-layer co-optimization of network design and chiplet placement in 2.5-d systems," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*.

[14] Google, "OR-Tools: Optimization tools."

[15] L. Perron, F. Didier, and S. Gay, "The CP-SAT-LP solver (invited talk)," in *Proc. CP 2023*, 2023, pp. 3:1–3:2.

[16] PCI-SIG, "PCI Express base specification revision 6.4."

[17] Y. Hayashida, "Throughput analysis of tandem-type go-back-n arq scheme for satellite communications," *IEEE Transactions on Communications*, vol. 41, no. 10, pp. 1517–1524, 1993.

[18] Synopsys, Inc., *Synopsys Design Compiler® User Guide*, 2019.

[19] L. T. Clark *et al.*, "Asap7: A 7-nm finfet predictive process design kit," *Microelectronics Journal*, vol. 53, pp. 105–115, 2016.

[20] S. Jangam and S. S. Iyer, "Silicon-interconnect fabric for fine-pitch ($\leq$ 10 μm) heterogeneous integration," *IEEE Transactions on Components, Packaging and Manufacturing Technology*, 2021.

[21] T. Hsu *et al.*, "A 7nm 0.46pJ/bit 20Gbps with BER 1e-25 die-to-die link using minimum intrinsic auto alignment and noise-immunity encode," in *IEEE Symposium on VLSI Circuits*, 2021.

[22] Y. Nishi *et al.*, "A 0.297-pJ/Bit 50.4-Gb/s/Wire inverter-based short-reach simultaneous bi-directional transceiver for die-to-die interface in 5-nm CMOS," *IEEE Journal of Solid-State Circuits (JSSC)*, 2023.

[23] Y. Nishi *et al.*, "A 0.190-pJ/bit 25.2-Gb/s/wire inverter-based AC-coupled transceiver for short-reach die-to-die interfaces in 5-nm CMOS," *IEEE Journal of Solid-State Circuits (JSSC)*, 2024.

[24] H. Kang *et al.*, "An 18-Tb/s/mm PAM-3 on-chip link with jitter-suppressing 3T4T coding and FFE-based crosstalk cancellation for memory interfaces," *IEEE Transactions on Circuits and Systems I (TCSI)*, 2025.

[25] J. Gu *et al.*, "A 32 Gb/s 0.36 pJ/bit 3 nm chiplet IO using 2.5-D CoWoS package with real-time and per-lane CDR and bathtub monitoring," *IEEE Journal of Solid-State Circuits (JSSC)*, 2025.

[26] S. Kim *et al.*, "A 42-Gb/s noise-tolerant single-ended clock-referenced PAM3 transceiver for chiplet interfaces," *IEEE Journal of Solid-State Circuits (JSSC)*, 2026.

[27] X. Wang *et al.*, "A 64Gb/s/wire 10.5Tb/s/mm/Layer single-ended simultaneous bi-directional transceiver with echo and crosstalk cancellation for a die-to-die interface in 28nm CMOS," in *IEEE International Solid-State Circuits Conference (ISSCC)*, 2025.

[28] S. Kim *et al.*, "A 0.458-pj/bit 24-gb/s/pin capacitively driven pam-4 transceiver with pam-based crosstalk cancellation for high-density die-to-die interfaces," *IEEE Journal of Solid-State Circuits*, vol. 59, no. 11, pp. 3730–3740, 2024.

[29] Global Unichip Corp. (GUC), *IGAD2DY04A Product Brief*, 2025.

[30] Global Unichip Corp. (GUC), *IGAD2DY08A Product Brief*, 2025.

[31] J. Yook *et al.*, "A 4–24 Gb/s/wire single-ended NRZ die-to-die transceiver with low latency for UCIe standard package in 4nm FinFET," in *2025 IEEE Asian Solid-State Circuits Conference (A-SSCC)*, 2025.

[32] A. Melek *et al.*, "A 0.29 pJ/b 5.27 Tb/s/mm UCIe advanced package link with 2.5-D CoWoS interposer and a 0.52 pJ/b 0.448 Tb/s/mm UCIe standard package link with organic substrate in 3 nm FinFET," *IEEE Journal of Solid-State Circuits (JSSC)*, 2026.

[33] Open Compute Project, *OCP – ODSA Project*.

[34] B. Vandersand *et al.*, "A 0.52pJ/bit 0.448Tbps/mm UCIe standard package die-to-die transceiver with low-latency TX clock alignment in 3nm FinFET," in *IEEE VLSI Symp.*, 2025.

[35] L. Zhang *et al.*, "A 61.4Gb/s/mm 1.42-pJ/b symmetric correlated coding wireline transceiver with single-ended crosstalk cancelling for D2D interconnection," in *European Solid-State Electronics Research Conference (ESSERC)*, 2024.

[36] C. Lin *et al.*, "7.7 a 2.16pJ/b 112Gb/s PAM-4 transceiver with time-interleaved 2b/3b ADCs and unbalanced baud-rate CDR for XSR applications in 28nm CMOS," in *IEEE International Solid-State Circuits Conference (ISSCC)*, 2024.

[37] K. Poon *et al.*, "A 1.24pJ/b 112Gb/s (870Gbps/mm) transceiver for in-package links in 7nm FinFET," in *IEEE Symposium on VLSI Circuits*, 2021.

[38] S. Gangasani *et al.*, "A 1.1pJ/b/lane, 1.8Tb/s chiplet over XSR-MCM channels using 113Gb/s PAM-4 transceiver with signal equalization and envelope adaptation using TX-FFE in 5nm CMOS," in *IEEE VLSI Symp.*, 2024.

[39] Y. Chen *et al.*, "A 212.5Gb/s DSP-based PAM-4 transceiver with 50dB loss compensation for large AI system interconnects in 4nm FinFET," in *IEEE International Solid-State Circuits Conference (ISSCC)*, 2025.

[40] J. Daudlin *et al.*, "Three-dimensional photonic integration for ultra-low-energy, high-bandwidth interchip data links," *Nature Photonics*, 2025.

[41] Z. Qi *et al.*, "A monolithically integrated DWDM Si-Photonics transceiver for chiplet optical I/O," *IEEE Journal of Solid-State Circuits (JSSC)*, 2025.

[42] P. Winterbottom, "Celestial ai photonic fabric module – the world's first soc with in-die optical io," Presented at HotChips 2025, 2025.

[43] Z. Wang *et al.*, "Co-designed silicon photonics chip I/O for energy-efficient petascale connectivity," in *Optical Fiber Communication Conference (OFC)*, 2024.

[44] D. Bunandar, "Passage m1000 a 3d photonic interposer for ai," Presented at HotChips 2025, 2025.

[45] S. Pal *et al.*, "Designing a 2048-chiplet, 14336-core waferscale processor," in *DAC*, 2021, pp. 1183–1188.

[46] S. Rashidi *et al.*, "Fred: A wafer-scale fabric for 3d parallel dnn training," in *Proc. ISCA*, 2025, pp. 34–48.