arXiv:2603.03470v1 [cs.DC] 3 Mar 2026

双同步 FIFO 与 FITO 范畴错误：用于分布式协调的经硅验证交互原语

Paul Borrill
DÆDÆLUS
paul@daedaelus.com

2026年3月

摘要

双同步 FIFO——在彼此独立的时钟域之间传输数据、且无需共享全局时间基准的硬件缓冲区——四十多年来一直在被设计、正式验证，并已在硅片上实现商业部署。我们梳理了相关文献，从 Chapiro 1984年的 GALS 学位论文，到 Cummings 的格雷码指针技术、Chelcea 和 Nowick 的混合时序接口、Greenstreet 的 STARI 协议，以及 2015年的英伟达可暂停双同步 FIFO，并指出，这些成果构成了经硅验证的存在性证明，足以反驳在分布式系统中普遍存在的仅向前时间（FITO）假设。核心论点是，基于交互的同步原语——握手、互斥和因果流控——能够在数字工程要求最严苛的层面取代基于时间戳的协调，直接挑战 PTP、TSN 和传统以太网等协议中的 FITO 假设。我们指出，片上双同步协调与开放原子以太网（OAE）架构在结构上具有相似性，并认为，在独立因果域之间进行协调时，基本原语应是握手，而非时间戳。

目录

1 引言：同步问题 **3**

2 历史发展 **4**

2.1 先驱工作：跨越时序边界的 FIFO .42.2 IEEE P896 Futurebus（1979–1987年）：背板上的握手协议 . . . 42.3 Chapiro（1984年）：GALS 学位论文 .42.4 Cummings（2001–2008年）：格雷码双同步 FIFO.52.5 Chelcea 和 Nowick（2001–2004年）：混合时序接口 .52.6 Greenstreet 和 Chakraborty（1990–2003年）：STARI 与自定时接口 . . 62.7 Keller、Fojtik 和 Khailany（2015年）：英伟达可暂停双同步 FIFO 62.8 Miro-Panades 和 Greiner（2007）：用于片上网络的双同步 FIFO 7

3 同步方法分类 **7**

4 对 FITO/范畴错误争论的意义 **7**

4.1 分布式系统中的 FITO 假设 .74.2 双同步 FIFO 作为存在性证明.84.3 握手作为基本原语 .84.4 从片上到芯片间：OAE 的平行对应 .9

4.5 Lamport 逻辑时钟：桥梁 .104.6 对高频交易的影响 .10

5 个开放问题与未来工作 10

6 结论 11

1 引言：同步问题

任何包含多个时钟域的数字系统都面临一个根本问题：如何安全地将数据从以某一速率（或相位）工作的生产者传送到以另一速率（或相位）工作的消费者？一种天真的做法是用接收端时钟对输入信号采样，但这会引发亚稳态——触发器因输入违反建立时间或保持时间要求而进入不确定状态。Chaney 和 Molnar 于 1973年发表的关于同步器异常行为的报告 [1] 首次记录了由模糊时序驱动的仲裁器可能无限期振荡；而 Ginosar 后来的教程 [18] 则总结了现代观点：亚稳态无处不在，只能通过概率方法加以管理，而无法消除。这个问题并非仅仅是理论上的。亚稳态故障曾导致现实世界中的系统崩溃，相关文献也以与其风险相称的严肃态度讨论这一主题。事实上，亚稳态和传输线信号异常造成的实际后果，是 IEEE P896 Futurebus 标准设计中的核心考量 [25], 。该标准采用了基于握手的同步协议和明确的时序保护机制（例如，要求线或信号持续足够长的时间 $2 \times T_{p0}$，以便被识别为有效信号），正是为了避免依赖时序各异的背板模块共享同一个全局时钟。

传统的时钟关系分类法沿用 Teehan、Greenstreet 和 Lemieux [13],，区分如下：

同步
频率相同，相位已知且固定。

同频异相
频率相同，相位未知（可能漂移）。

近同步
频率几乎相同，相位缓慢漂移。

异步（异步）频率和相位互不相关。

一种**双同步 FIFO** 是先进先出缓冲区，旨在确保在后三种情况中的任何一种下都能正确运行，也就是说，在两个彼此独立、频率或相位关系均无保证的时钟域之间运行。“双同步”一词强调，FIFO 的每一侧都与各自的时钟局部同步，而两侧之间的接口则是异步的。

从 FITO/范畴错误之争 [27, 31], 的角度看，双同步 FIFO 的非凡之处在于它们不 需要什么：它们不需要全局时钟、共同时间基准、同步时间戳，也不需要两个时钟域之间达成任何形式的日时一致。相反，它们仅凭**基于交互的** 原语就能实现正确、无损且有序的数据传输：握手信号、指针比较、互斥元件和因果流控协议。

本文综述了双同步 FIFO 相关文献，并论证其构成了一个成熟且经硅验证的存在性证明，足以反驳分布式系统中的 FITO 假设。论证分为四个阶段：我们追溯双同步设计的历史发展（第 2 节），将相关文献整理为同步机制的分类体系（第3 节），阐明其对 FITO/范畴错误之争的影响（第 4 节），并指出将片上协调与网络规模协议设计联系起来的开放问题（第5 节）。

2 历史发展

2.1 先驱工作：跨越时序边界的 FIFO

FIFO 缓冲区能够桥接独立计时的时钟域，这一理念早于全局异步、局部同步范式。早在1974年，Lim 为美国国家航空航天局撰写的关于“准完美”FIFO 的技术报告 [2] 就探讨了迭代式 FIFO 实现，使其能够以同步模式或异步模式运行于 UNICON 激光存储器控制器中——这很早就认识到，时钟域之间的边界是一个需要通过结构化方案解决的设计问题，而不能依赖时序假设。

2.2 IEEE P896 Futurebus（1979–1987年）：背板层级的握手协议

与学术界的 GALS 计划同步推进，并在某些方面走在其前面的是 IEEE P896 Futurebus 工作组（成立于 1979 年；从 1982 年起开展积极的设计工作），该工作组在背板层级直面同样的同步问题。Borrill 和 Theus 于 1984 年在 *IEEE Micro* [24] 发表的论文介绍了为拟议中的 Futurebus 设计的一种先进通信协议，该协议完全依赖基于握手的同步，而非共享总线时钟。该协议定义了一套全互锁握手模式分类体系——单源、单目的地、源与目的地（四事件和两事件变体），以及用于多方协调的 TRI（三方）握手——每种模式都能确保在时序彼此独立的背板模块之间正确传输数据。

Futurebus 相关工作在历史上具有多方面的重要意义。首先，它早于 Chapiro 于 1984 年完成的 GALS 学位论文（该工作组自 1979 年起便开展工作，而 Borrill 的分类体系则是在 20 世纪 80 年代初形成的）。其次，该协议利用了握手信号的两个边沿——这种双边沿信号传输可能是后来被称为 DDR（双倍数据速率）的技术在存储器行业中最早的实例之一。第三，Borrill 后来的博士论文 [25], 对握手分类体系及相关的传输线分析进行了形式化阐述，之后在挑战 Rambus DDR 存储器专利的专利诉讼中被引为现有技术，足以证明这项早期基于交互的设计工作具有实际和法律上的分量。

IEEE P896.1 标准于1987年发布，确立了与技术无关的总线同步的基础概念，并影响了20世纪80年代末和20世纪90年代的后续互连标准。

2.3 Chapiro（1984年）：GALS 学位论文

全局异步、局部同步（GALS）范式由 Daniel Chapiro 在其1984年于斯坦福大学撰写的博士学位论文 [3] 中提出。Chapiro 意识到，随着设计规模不断扩大，强制整个芯片共用一个全局时钟已变得难以为继，因此提出系统应由通过异步方式通信的同步岛构成。他还提出了可暂停时钟方案：当异步输入在不合适的时刻到达时，可暂时停止本地时钟发生器，以避免亚稳态。

Chapiro 的洞见从根本上说是架构层面的：全局时钟是一种便于使用的抽象，并非物理上的必需条件。每个同步岛都是一个独立的因果域；正确性的关键不在于事件是否共享同一时间戳，而在于数据跨越边界时是否安全——也就是说，不会出现亚稳态，也不会丢失数据。值得注意的是，IEEE P896 工作组几年前也曾独立地在背板层级得出基本相同的结论，只不过其出发点是工程实践，而非理论研究。

2.4 Cummings（2001–2008）：格雷码双同步 FIFO

Clifford E. Cummings 在一系列颇具影响力的 SNUG（Synopsys 用户组）论文中 [4, 5],确立了后来成为业界异步先进先出队列设计标准的方法。其关键贡献是采用格雷码指针——一种每次递增时仅有一位发生变化的二进制计数器——并使用双触发器（“蛮力”）同步器，使其在时钟域之间完成同步。

其工作原理如下：双端口 RAM 用于存储 FIFO 数据。写指针在写时钟域中维护，读指针则在读域中维护。要计算满标志，必须在写域中将读指针与写指针进行比较；空标志的计算方式则相反。格雷码指针经由两级同步器触发器传入对侧时钟域。由于每次递增时只有一位发生变化，任何亚稳态事件最多只会导致同步后的指针比真实值落后一次递增——这是一种保守误差，会使 FIFO 显得比实际稍满（或稍空），但绝不会导致溢出或下溢。

Cummings 的论文以注重对正确性属性进行形式化推理而著称。关键不变式是：指针的同步副本始终等于或落后于真实指针。因此，满标志可以保守地置位（即 FIFO 尚未真正满时就置位），但绝不会漏报；空标志的情况则相反。换言之，FIFO 的每一侧都对远端指针持有有界的过时认知：它并非即时的真实状态，而是经过延迟和滤波、且保证安全的观测。满/空标志是一种认知性陈述——“根据同步延迟后我当前掌握的信息，FIFO（不）满”——而非对共享时间下全局状态的本体论断言。

这是一个**因果安全**论证，而非时序论证。正确性不取决于两个时钟的相对频率、不取决于时钟偏斜的任何界限，也不取决于任何时间戳比较。它完全取决于格雷码的结构特性以及同步器触发器的亚稳态消解特性。

蛮力同步的代价是延迟：每次指针跨越至少需要两个接收端时钟周期才能完成消解，因此会增加 $4+$ 个时钟周期的往返同步延迟。

2.5 Chelcea 和 Nowick（2001–2004年）：混合时序接口

哥伦比亚大学的 Tiberiu Chelcea 和 Steven M. Nowick 开发了一系列低延迟混合时序 FIFO，可连接任意组合的同步系统和异步系统 [6, 7]。其设计采用连接到公共数据总线的循环单元数组；数据项入队后保持不动，并通过游动令牌进行访问。

Chelcea 和 Nowick 的工作有两个特别值得关注的关键特征。首先，他们的设计是**模块化**的：每个单元的放入接口和取出接口彼此独立，可配置为同步（时钟控制）或异步（握手）操作。这种模块化意味着同一个 FIFO 单元可以实现同步到同步、同步到异步、异步到同步或异步到异步通信。

其次，他们将 Carloni等人的“延迟无关”协议 [22]——最初是为带有长导线的单时钟系统设计的——改造后应用于混合时序域。这一迁移证明，基于交互的流控制（带反压的 valid/ready握手）可以自然地推广到时钟域边界之间。

稳健性保证以亚稳态概率来表述：通过增加同步器级，设计可以达到“任意高的稳健性”，代价是延迟增加，而失效概率则呈指数级下降。

2.6 Greenstreet 和 Chakraborty（1990–2003）：STARI 与自定时接口

Mark Greenstreet 的博士研究提出了用于中同步通信的 STARI（接收端输入自定时）协议 [9]。其核心思想是在两个时钟域的交界处放置一个自定时 FIFO。该 FIFO 吸收两个时钟之间的相位差：只要偏斜不超过 FIFO 的深度，数据就能无误通过。

Greenstreet 证明了 STARI 的正确性，包括其不会发生同步故障——这一结果超越了概率论证，提供了形式化保证。在随后与 Chakraborty 合作的研究中， [8], 这一方法被推广到单级 FIFO，这类 FIFO 利用已知的时钟关系（有理数倍、频率高度匹配）以最少的硬件实现同步。这项研究荣获 ASYNC 2003 最佳论文奖。

STARI 理念意义重大，因为它将同步视为数据流与本地时钟之间的**物理交互**，并由 FIFO 的弹性存储予以调节。它不涉及挂钟时间、时间戳或外部协调协议。FIFO 只需吸收这种不匹配，而握手协议则确保数据不会在写入前被读取，也不会在读取前被覆盖。

2.7 Keller、Fojtik 和 Khailany（2015年）：英伟达可暂停双同步 FIFO

这篇发表于 IEEE 国际异步电路与系统研讨会的 2015年 英伟达论文 [10], 代表了双同步 FIFO 设计的最先进水平，其尤为重要的一点是，设计时充分考虑了工业应用。

该设计将 Chapiro 的可暂停时钟与 Cummings 的双端口同步 FIFO 结构相结合。它没有通过强制同步器来同步多位格雷码指针，而是使用**两相递增／确认信号** 在不同域之间传递指针更新。每个递增信号都通过基于互斥器的可暂停时钟网络进行同步；如果输入信号在不安全的相位到达，该网络就会暂时停止接收端时钟。

其主要特点包括：异步接口上的平均延迟为 1.34 个时钟周期（而蛮力同步约需 4 个周期）；通过可暂停时钟实现无错误运行，从而消除亚稳态，而非仅仅降低其发生概率；兼容标准工具流程，因为 FIFO 存储器采用常规双端口同步 RAM，只有互斥器元件需要定制设计；以及借助多组递增／确认信号对实现可扩展流量控制，即使时钟频率相差最多 2:1，也能达到满吞吐量。

流量控制协议值得详细说明。当发送器向 FIFO 写入一个字时，写指针逻辑会切换两条双相写指针递增线中的一条。该切换通过可暂停同步器传播到接收端时钟域。如果切换在安全时钟相位到达，就会立即通过。如果在不安全相位到达，互斥器便会介入，时钟暂停，直到信号得到安全处理。同步完成后，接收方的读指针逻辑会更新对写指针的跟踪，然后即可继续读取。

整个协议都基于**因果握手**：发送器只有在确认接收方已应答后才能继续推进，而接收方只有在确认发送器已写入后才能读取。这里没有全局时间、没有时间戳，也没有时钟同步协议，只有交互。

2.8 Miro-Panades 和 Greiner（2007）：用于片上网络的双同步 FIFO

Miro-Panades 和 Greiner [11] 针对全局异步、局部同步系统中的片上网络（NoC）架构，提出了一种双同步 FIFO 设计。他们的设计引入了一种新颖的指针比较编码算法，无需进行格雷码转换即可避免其开销，同时确保安全性。

这项工作的意义在于，它明确将双同步 FIFO 定位为全局异步、局部同步架构中片上网络通信的基础构建模块——这与 OAE 在多设备系统芯片间网络通信中所发挥的基础构建模块作用直接对应 [30]。

3 同步方法分类

文献中对时钟域跨越方法进行了清晰的分类，依据是所采用的安全保障机制：

蛮力同步
多触发器链允许亚稳态以概率方式消解。延迟高（每次跨越 2+ 个周期），且失效概率非零。Cummings 先进先出队列 [4]。

可暂停时钟
当异步输入在不安全的相位到达时，基于互斥锁的电路会暂停本地时钟。失效概率为零（时钟只需等待），平均延迟极低（约 1.3 个周期）。英伟达可暂停双同步 FIFO [10]。

自定时／基于握手
全异步先进先出队列（微流水线、GasP、Mousetrap）使用请求／应答握手进行流量控制。FIFO 本身完全没有时钟。Sutherland 的微流水线 [15],Sutherland 和 Fairbanks 的 GasP [19]。

相位预测
利用时钟相位关系的已知信息，在安全时刻进行采样的电路。Chakraborty 和 Greenstreet 的有理时钟接口 [8]。延迟低，但需要假设时钟稳定。

混合时序模块化
Chelcea 和 Nowick 的 FIFO 系列 [7], 提供可互换的同步和异步接口。可跨时序域适配的延迟无关协议。

所有这些方法的共同点——也是它们与分布式系统协议（如精确时间协议、时间敏感网络和传统以太网）中基于时间戳的协调方式的区别——在于，**它们都不依赖共享时间概念**。它们都通过结构性和因果性机制实现正确性：握手、互斥、指针不变量和流量控制。

4 FITO／类别错误争论的重要性

4.1 分布式系统中的 FITO 假设

仅向前时间（FITO）假设，正如 DÆDÆLUS 对传统网络通信的批判 [27, 33, 28]，所阐述的那样，认为分布式系统协议的设计基于这样一个前提：事件可以通过分配来自同步时钟的时间戳来排序，且这些时间戳单调地向前推进。这一假设体现在 PTP（IEEE 1588）及其变体中：它们试图在网络中同步时钟，以达到

纳秒级精度；时间敏感网络（TSN）根据共享时间基准调度流量；传统以太网依赖超时和重试来进行错误恢复；高频交易系统则依靠时间戳排序来确定交易优先级，尽管狭义相对论表明，对空间上相隔的事件进行同时排序在物理上毫无意义 [32]。

“范畴错误”——按 Ryle 的原初意义 [23], 将一个本体论范畴的概念误用于另一个范畴——在 DÆDÆLUS 批判中，是对因果排序 与时间排序的混淆。Lamport 1978 年关于逻辑时钟的论文 [17] 已经认识到，在分布式系统中，唯一具有物理意义的排序是由消息传递所定义的先发生关系。然而，占主导地位的工业实践仍然是施加一个全局时间坐标，并按时间戳对事件排序——这种方法很脆弱（时钟会漂移）、代价高昂（同步协议消耗带宽并增加延迟），并且在相对论极限下在物理上并不自洽 [31, 26]。

4.2 双同步 FIFO 作为存在性证明

双同步 FIFO 文献提供了一个具体的、经硅验证的存在性证明：范畴错误不仅是哲学上的异议，而且是一个具有切实替代方案的实践问题。考虑 NVIDIA 可暂停双同步 FIFO [10]实现了什么：

1. **正确、无损、有序的数据传输**在两个完全独立的时钟域之间。2. **无共享时间基准** 任何形式——甚至没有共同频率。3. **无时间戳**——指针更新通过交互（经由互斥锁的两相握手）传递，而不是通过编码时间值。4. **形式化正确性保证** 是结构性的，而非概率性的（在可暂停情形中），或者在最坏情况下以指数下降的失败率受到概率界定（在暴力法情形中）。5. **性能可与同步接口媲美**——平均延迟 1.34 个周期，满吞吐量，能耗低于暴力法。

如果能在单个时钟周期的层面上实现这一点——这正是整个工程领域中要求最高的同步环境，也就是硅片上——那么，声称分布式系统需要全局时间同步才能实现可靠、有序的通信，就显然是错误的。双同步 FIFO 就是一个反例。

有人可能会反驳说，FIFO 本身体现了仅向前假设：数据从写端口进入，再从读端口按严格顺序输出，而且这种顺序不可逆。但这是把因果顺序与时间顺序混为一谈——恰恰就是本文所讨论的范畴错误 [23]。FIFO 的先写后读不变量是一种结构性约束：写指针必须先前进，读指针才能跟进。这说的是已经发生了什么（因果事实），而不是事情发生在何时（时间上的主张）。这里不需要参考任何时间戳；指针比较是基于交互的检查，而非时刻比较。FIFO 的顺序归根结底完全是因果顺序。

4.3 握手作为基本原语

双同步 FIFO 相关文献带来的更深层启示是：在独立因果域之间进行协调时，自然的基本原语是**握手**，而不是时间戳。每一种双同步 FIFO，无论其具体架构如何，都依赖某种形式的握手协议：

- 在 Cummings 的设计中， [4], 穿越同步器触发器的格雷码指针构成一种隐式握手：指针值由发送方“提供”，并在同步延迟后由接收方“接受”。

- 在可暂停设计中 [10], 两阶段递增/确认协议是一种显式握手。
- 在 Chelcea 和 Nowick 的设计中， [7], 放置/获取接口实现了请求/确认协议。
- 在 Sutherland 的微流水线中， [15], Muller C 元件在相邻流水线级之间实现了会合握手。
- 在 STARI 中， [9], 自定时 FIFO 通过弹性缓冲，在发送器的数据速率与接收端时钟之间进行协调——这是信号传播与时钟边沿之间的一种物理握手。

这种趋同并非偶然，而是反映了一个基本的物理现实：两个独立的因果域只能通过交互来协调，而不能依靠对时钟的共享观测。握手是建立发送方与接收方之间因果排序的最小交互。用 OAE 项目的术语来说，它是一种原子交换——不可分割的信息交换，由此产生一个新的因果事实 [30, 29]。

同一原理早在背板层级便已由 IEEE P896 Futurebus 标准确立 [25], ：亚稳态和传输线现象（反射、串扰和线或毛刺）促使人们采用基于交互的同步协议。Futurebus 定义了同步、异步和独立定时模式的分类，并规定了在时钟彼此独立的背板模块之间采用基于握手的流量控制。在板级工程中，这恰好预示了 GALS 社群后来为片上时钟域交叉正式确立的设计理念。Futurebus 的经验表明，握手原语能够跨物理域扩展：从片上触发器同步器，到背板总线协议，再到网络规模的原子交换。

4.4 从片上到芯片间：OAE 的平行关系

双同步 FIFO 在芯片上的两个时钟域之间进行协调。IEEE P896 Futurebus [25]则在背板上协调具有独立时序的模块。OAE 在网络中的节点之间进行协调。这种结构上的平行关系跨越三个物理尺度：

概念	**双同步 FIFO**	**Futurebus**	**OAE**
独立域	时钟域	背板模块	网络节点
排序机制	指针握手	总线握手	原子交换
流量控制	满/空标志	线或协议	基于信用的反压
数据完整性	格雷码不变量	$2 \times T_{p0}$ 保护条件	ACID 事务
需要全局时间	No	No	No
需要超时/重试	No	No	No

GALS 文献的发展轨迹——从 Chapiro 指出全局时钟并非必需 [3], ，到数十年来对基于交互的同步不断改进，再到性能超越同步对应设计的经硅验证设计——预示了 OAE 为网络通信提出的发展轨迹：从指出全局时间同步是一种范畴错误 [27, 31], ，到开发基于交互的网络原语，再到实现无时间戳原子性的协议栈。

然而，GALS 文献在某种重要意义上，尚未达到最强协调原语。Chelcea 和 Nowick 的“put/get”接口 [7]——事实上，每一种双同步 FIFO 都是如此——都将交换分解为两个按时间先后发生的半操作：生产者将数据放入缓冲区，而消费者则在稍后的某个时间取出数据。从放入到取出之间，存在一个中间状态：发送方已提交数据，但接收方尚未消费数据。满/空标志用于管理这一窗口，却无法将其消除。OAE 原子交换 [30, 29] 将这两个半操作合并为

单个不可分割的因果事件：数据传输与确认作为一个原子操作同时发生，双方都看不到中间状态。换言之，双同步 FIFO 证明，在独立域之间进行协调并不需要时间；OAE 更进一步，表明甚至不需要将写操作和读操作分开的两阶段分解。原子交换是 GALS 文献开创的这一发展轨迹自然而然的终点。

4.5 Lamport 的逻辑时钟：桥梁

值得注意的是，Lamport 1978 年论文 [17]处于一个颇为特别的中间位置。Lamport 认识到，在分布式系统中，唯一具有物理意义的顺序，是由消息传递定义的先发生关系——这是一种从根本上基于交互的洞见。然而，他的论文也提出了使用逻辑 时间戳 (单调计数器)来表示这种顺序的想法，并进一步提出通过物理时钟同步来处理“异常”情况：此时，逻辑顺序与外部观察者感知到的物理顺序不一致 [31]。

双同步 FIFO 相关研究通过证明正确排序甚至不需要逻辑时间戳，化解了这种矛盾。指针不是时间戳，而是因果序列中的位置。写指针表示“这是下一个要写入的槽位”；读指针表示“这是下一个要读取的槽位”。比较二者即可确定 FIFO 是满还是空。整个过程中完全不涉及“何时”的概念，只有“已经完成了什么”和“还有什么待完成”。

这正是双同步 FIFO 体现基于交互的 协调方式之处：相关信息不是“现在几点？”，而是“我们是否交互过？”指针握手是因果连接的物理记录，而非时间上的断言。

4.6 对高频交易的影响

双同步 FIFO 相关文献也进一步强化了对 HFT 的批评 [32]。如果两个交易所作为彼此独立的时钟域运行（它们在物理上确实如此，中间隔着光速延迟），那么按时间戳排列交易，恰恰是双同步 FIFO 设计者几十年前就学会避免的范畴错误。正确的做法是按照因果交互来排列交易——也就是说，哪笔交易先被撮合引擎实际接收并确认，而不论发送方附带了什么时间戳。撮合引擎的接收缓冲区本质上就是一个双同步 FIFO：数据从外部时钟域（交易者的系统）到达，必须安全地写入引擎的本地域。排序由 FIFO 中的到达顺序决定，而非任何时间戳。

5 个开放问题与未来工作

双同步 FIFO 设计与 FITO/范畴错误之争的交汇处仍有几个悬而未决的问题：

扩展至多跳网络。片上全局异步局部同步系统在相邻时钟域之间通信。OAE 必须能够处理多跳通信，其中独立域的数量会随着网络规模的扩大而增加。片上网络领域的文献 [11, 20] 已探讨了这一问题；如何将这些成果扩展至数据中心规模网络架构，是核心挑战。

协议层的形式化验证。双同步 FIFO 文献在电路层拥有强有力的形式化验证结果（Greenstreet 的正确性证明 [9]）。将这些方法扩展到 OAE 协议栈——证明基于交互的原语

能够在由双同步链路构成的网络中维持 ACID 属性——将是一项重要贡献。

弥合 SerDes 差距。 现代高速串行链路（PCIe、以太网 SerDes）已在物理层执行时钟域跨越，通常使用功能上即为双同步 FIFO 的弹性缓冲区。使这种隐式 FIFO 行为显式化并可供更高协议层访问——而不是将其隐藏在同步抽象之后——是 OAE SerDes 层规范的一部分。

范畴论形式化。 双同步 FIFO 与 OAE 原子交换之间的结构平行表明，二者可能都是更一般的范畴论构造的实例（或许与 Pratt 的偏序多重集或并发进程的幺半范畴相关）。将这一联系形式化，可为在所有尺度上推理基于交互的协调提供统一框架。

与无时间方向的消息传递的联系。 消息传递的约束语义方法 [34] 提供了一个形式框架，其中消息是边界约束而非时间事件。双同步 FIFO 的指针握手是这种约束语义的物理实例化：写指针约束读取方可访问的内容，读指针约束写入方可覆盖的内容，双方均不引用时间。

6 结论

双同步 FIFO 是数字工程中研究最为透彻、实践验证最为充分的结构之一。四十多年来，GALS 社群开发、正式验证并在商业产品中部署了相关设计，仅凭基于交互的原语——握手、互斥、指针不变量和因果流控——便实现了完全独立的时钟域之间正确、高性能的数据传输。无需全局时间、时间戳或时钟同步协议。

这项成果有力地证明了 DÆDÆLUS 对传统分布式系统的批判 [27]：在分布式系统协议中依赖全局时间同步是一种范畴错误；基于交互的原语能够取代基于时间戳的协调，同时不牺牲正确性或性能。双同步 FIFO 以经验证的硅片证明，在相互独立的因果域之间进行协调，正确的方法不是同步它们的时钟，而是同步它们的交互。Open Atomic Ethernet [30,29] 提议将这一经验——在工程领域要求最严苛的同步环境中积累的经验——应用于网络协议设计。

致谢

本文在 AI 工具（Claude、Anthropic）的协助下完成了文献综述组织、撰写和 LaTeX 准备工作。研究计划及所有技术主张均由作者独自负责。FITO 范畴错误框架源自作者在 Earth Computing（2012–2020年）开展的硬件验证的确定性网络研究，并在 DÆDÆLUS 研究计划下继续推进。

参考文献

[1] T. J. Chaney 和 C. E. Molnar，《同步器与仲裁器电路的异常行为》，*IEEE*计算机学报，第 C-22 卷，第 4 期，第 421–422 页，1973年4月。[2] R. S. Lim，《准完美先进先出队列：同步或异步及其在 UNICON 激光存储器控制器设计中的应用》，美国国家航空航天局技术报告 CR-137841，1974年。[3]D. M. Chapiro，《全局异步、局部同步系统》，博士论文，斯坦福大学，1984年。[4] C. E. Cummings，《异步先进先出队列设计的仿真与综合技术》，SNUG圣何塞会议，2002。[5] C. E. Cummings，《使用 SystemVerilog 的时钟域跨越（CDC）设计与验证技术》，SNUG波士顿会议，2008。[6] T. Chelcea 和 S. M. Nowick，《混合时序系统的鲁棒接口及其在延迟无关协议中的应用》，载于 设计自动化会议论文集，2001。[7] T. Chelcea 和 S. M. Nowick，《混合时序系统的鲁棒接口》，*IEEE*超大规模集成电路系统学报，第 12 卷，第 8 期，第 857–873 页，2004年8月。[8] A. Chakraborty 和 M. R. Greenstreet，《跨时钟域的高效自定时接口》，载于 异步设计会议论文集，2003。（最佳论文奖。）[9] M. R. Greenstreet，《STARI芯片的实现》，载于国际计算机设计会议论文集，1995年。[10] B. Keller、M. Fojtik 和 B. Khailany，《面向全局异步、局部同步系统的可暂停双同步先进先出队列》，载于 异步设计会议论文集，2015年。[11] I. Miro-Panades 和 A. Greiner，《适用于全局异步、局部同步架构中片上网络的同步电路通信双同步先进先出队列》，载于片上网络会议论文集，2007。[12] J. Muttersbach、T. Villiger 和 W. Fichtner，《全局异步、局部同步系统的实用设计》，载于异步设计会议论文集，2000。[13] P. Teehan、M. Greenstreet 和 G. Lemieux，《全局异步、局部同步设计风格的综述与分类》，*IEEE*设计与测试，第 24 卷，第 5 期，第 418–428 页，2007。[14] K. Yun 和 R. Donohue，《可暂停时钟：迈向异构系统的第一步》，载于 国际计算机设计会议论文集，1996。[15] I. E. Sutherland，《微流水线》，美国计算机协会通讯，第 32 卷，第 6 期，第 720–738 页，1989年。[16] S. M. Nowick 和 M. Singh，《异步设计——第 1 部分：概述与最新进展》，*IEEE*设计与测试，第 32 卷，第 3 期，2015年。[17] L. Lamport，《分布式系统中的时间、时钟与事件排序》，美国计算机协会通讯，第 21 卷，第 7 期，第 558–565 页，1978年。[18] R. Ginosar，《亚稳态与同步：教程》，*IEEE*设计与测试，第 28 卷，第 5 期，第 23–35 页，2011年。[19] I. Sutherland 和 S. Fairbanks，《GasP：最小化先进先出队列控制》，载于 异步设计会议论文集，2001。[20] N. Dimitrakopoulos，《CrossOver：虚通道流量控制下的时钟域跨越》，载于 设计、自动化与测试欧洲会议论文集，2016。

[21] D. Verbitsky、R. Dobkin、R. Ginosar 和 S. Beer，《StarSync：可扩展的标准单元同频异相同步器》，《集成：超大规模集成电路期刊》，2013年。[22] L. P. Carloni、K. L. McMillan 和 A. L. Sangiovanni-Vincentelli，《延迟无关设计理论》，*IEEE*计算机辅助设计汇刊，第20卷第9期，第1059–1076页，2001年。[23] G. Ryle，《心灵概念》。伦敦：Hutchinson，1949年。[24] P. Borrill 和 J. Theus，《拟议 IEEE 896 Futurebus 的高级通信协议》，*IEEE Micro*，第4卷第4期，第42–56页，1984年8月。[25] P. Borrill，《微处理器总线结构与标准》，博士论文，伦敦大学学院，1986年。（第5–6章讨论了 IEEE P896 Futurebus 标准中的亚稳态、基于握手的同步协议和传输线现象。）[26] P. Borrill，《地月空间时间的范畴错误：为什么美国国家航空航天局无法同步不存在的事物》，arXiv:2602.18641 [cs.DC], 2026.[27] P. Borrill，《分布式计算所犯的错误：将设计选择变成自然法则的范畴错误》，arXiv:2602.18723 [cs.DC], 2026.[28]P. Borrill，《iCloud为何失败：云同步的范畴错误》，arXiv:2602.19433 [cs.DC], 2026.[29] P. Borrill，《利用 Open Atomic Ethernet 规避 FLP不可能性结果》，arXiv:2602.20444 [cs.DC], 2026.[30]P. Borrill，《利用 Open Atomic Ethernet 规避 CAP定理》，arXiv:2602.21182 [cs.DC], 2026.[31]P. Borrill，《Lamport时间之箭：逻辑时钟中的范畴错误》，arXiv:2602.21730 [cs.DC], 2026.[32] P. Borrill，《工程化同时性：跨类空分离的交易所进行综合价格发现的物理不可能性》，arXiv:2602.22350 [cs.DC], 2026.[33] P. Borrill，《Unix工具与 FITO 范畴错误：崩溃一致性与持久性的协议本质》，arXiv:2603.01384 [cs.DC], 2026.[34] P. Borrill，《没有时间方向的消息传递：约束语义与 FITO 范畴错误》，arXiv:2603.01405 [cs.DC], 2026.[35]P. Borrill，《语义时间之箭（第一部分）：从爱丁顿到以太网》，arXiv:2603.01440 [cs.DC], 2026。

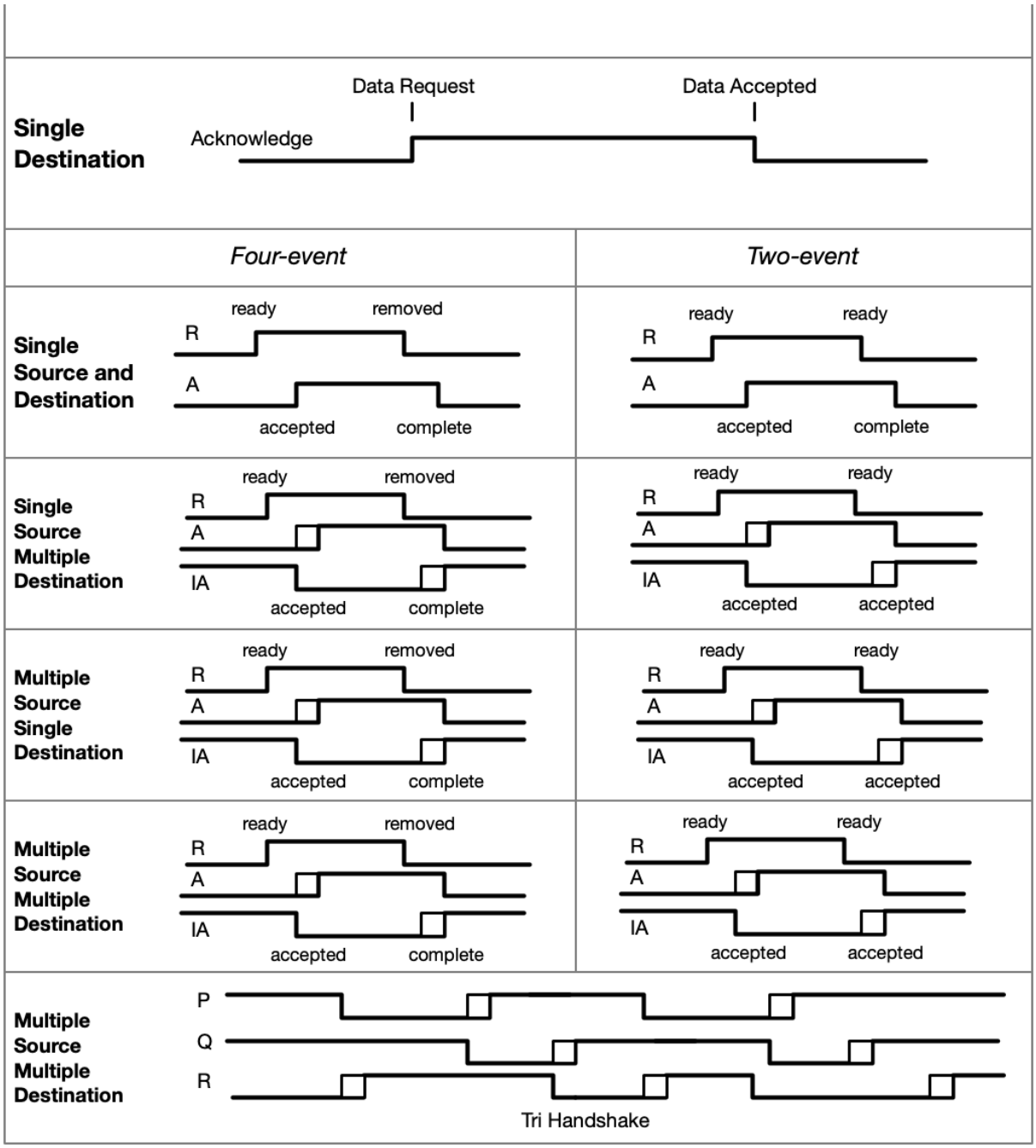

图 1：Borrill 1986年分类法中的完全互锁握手协议 [25], 最初为 IEEE P896 Futurebus 标准开发（工作组成立于 1979年；另见 [24]）。该图将单源、单目的地、源与目的地（四事件和两事件变体）以及用于多方协调的 TRI 握手进行了分类。矩形脉冲宽度表示线或协议窗口：前沿标志着最快的模块驱动导线的时刻，而后沿标志着最慢的模块释放导线的时刻。这与源定时协议存在根本区别；在此类协议中，发送器会在固定的、依赖于技术的时间间隔内保持信号有效。线或握手在设计上与技术无关：由于导线本身会报告所有参与者何时均已响应，该协议无需修改即可适用于任意长度的背板以及未来的任何硅技术。这一分类法比异步超大规模集成电路文献中对四相和两阶段握手的形式化描述早了数年（Sutherland 1989年、Chelcea 和 Nowick 2001年）。