

一种可扩展的高精度可靠 IO 时序测试方法

苏平, 高飞, 刘凯, 张云飞, 李兰兰

(北京航天自动控制研究所, 北京 100854)

摘要: 针对现有 IO 测试方法时序测量数量少、精度和可靠性低的问题, 提出了一种可扩展的高精度可靠 IO 时序测试方法, 实现了基于层次化的混合网络架构的统一时钟的可扩展大规模分布式 IO 时序测试系统。试验表明该系统在有效提高了 IO 时序容量规模的同时, 保证了每个时序的测量时间精度, 同时这种测量精度不会随层级增加而积累, 各层级间误差一致性较高, 可在满足大规模异步 IO 时序测试的同时, 兼顾高精度和高可靠性, 具有一定的实用价值。

关键词: 时序测试; 分层网络; 时间同步

中图分类号: TP368.1

文献标识码: A

DOI: 10.19358/j.issn.2097-1788.2023.07.013

引用格式: 苏平, 高飞, 刘凯, 等. 一种可扩展的高精度可靠 IO 时序测试方法 [J]. 网络安全与数据治理, 2023, 42(7): 79-84.

A scalable high precision and reliable IO timing sequence testing method

Su Ping, Gao Fei, Liu Kai, Zhang Yunfei, Li Lanlan

(Beijing Aerospace Automatic Control Institute, Beijing 100854, China)

Abstract: In this paper, a scalable high-precision and reliable IO timing test method is proposed to solve the problems of small number of timing measurements and low accuracy and reliability of existing IO testing methods, and a scalable large-scale distributed IO timing test system based on hierarchical hybrid network architecture and unified clock is implemented. The test shows that the system not only effectively improves the capacity scale of IO timing sequence, but also guarantees the measuring time accuracy of each timing sequence. At the same time, the measuring accuracy will not accumulate with the increase of levels, and the error consistency between different levels is high. It can meet the requirements of large-scale asynchronous IO timing sequence testing while giving consideration to high precision and high reliability, which has certain practical value.

Key words: timing sequence testing; hierarchical network; time synchronization

0 引言

IO 型数字量由于其实现简单可靠、成本低廉, 常常作为嵌入式系统中获取系统时序反馈、运行状态的一种即时有效的采样形式。但它常常受到芯片或者集成电路的体积限制, 使得能够同时采用这种形式的信号数量较少, 无法较好适配采集 IO 型数字量规模较大的甚至跨地域分布的自动化控制系统, 采集的实时性也常常无法与其在小规模系统中的应用效果相比。

王琪、章文晋^[1]提出了一种基于单片机的时序测试方法, 王乔、王民钢等^[2]针对特定系统的时序采集进行了设计, 盖猛、王吉平和留黎钦、林志雄等^[3-4]从硬件实现角度设计了一种数字量采集电路方法, 这些传统 IO 型数字量采集系统都是基于单处理器和单网络, 受本身处理器接口的限制, 只可以解决少量 IO 型数字量的采

集, 无法满足大规模的 IO 型数字量采集。曹洪龙、胡剑凌和刘素华等^[5-7]设计了一种基于多路开关量信号检测板卡的采集系统, 扩展了采集路数, 但板卡间没有同步机制, 对实时性要求高的场景并不适用。郑来波、李泰强和何春华、鄢仁成等^[8-9]针对采样可靠性从方面软件提出消抖等算法。许天骄、吴胜华和陈晨^[10-11]从硬件角度提出了提高采集信号可靠性的方法, 但方案复杂, 成本较高。因此, 一种既能满足大规模需求、又能满足实时性、可靠性要求的 IO 型数字量采集系统显得尤为重要。

本文提出了一种基于层次网络架构进行扩展的分布式 IO 时序实时监测方法, 该方法通过将待采集的 IO 时序进行分组, 并将不同的 IO 时序采集组挂接到不同层次中的处理节点中, 从而获得一种适应性较强的扩展结构。在这个 IO 时序采集层次结构中, 系统时间信息周期性自

顶向下流向每个层及各个节点,从而确保系统中的各个节点的时钟始终与系统时钟保持一致,在所监测的 IO 信号发生变化时,能够以系统时钟对变化时刻进行标记。

1 分层扩展的 IO 时序监测网络

大规模的 IO 时序监测系统需要解决三个方面问题: IO 时序监测口的扩展,即 IO 时序监测口的接入问题;时序测试结果信息的时间一致性问题,即让每个接入点具备系统统一的时间;以及各监测节点 IO 状态监测的可靠性问题,即应避免由于采集时刻 IO 发生状态变化引起的读写冲突,导致采样结果的异常。

基于层次架构具有可低成本进行扩展且扩展数量不受硬件指标限制的特点,采用层次化的网络结构对 IO 时序监测口进行扩展。每层网络对应一个本层网络控制节点和一组 IO 时序监测节点及若干下层网络 IO 时序监测控制节点。从而横向上可以在同层网络上扩展 IO 监测节点,当同层网络站点容量受限时,可以通过增加一层下层监测网络,从而在纵向扩展 IO 监测节点。图 1 为一个三层 IO 时序监测网络示意图。

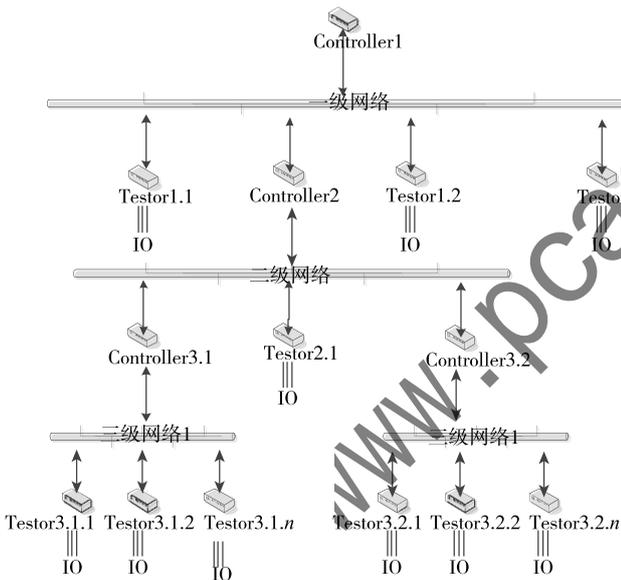


图 1 三层 IO 时序监测网络示意图

IO 时序监测网络中,每层测试网络对应的控制节点负责本层网络节点的时间同步管理,收集本层网络中测试节点的测试结果,并将测试结果由本层网络的测试时间转换为上层网络时间,报送上层网络的控制节点。每层测试网络对应的测试节点完成 IO 输入口的电平状态监测,并将每一次电平状态的变化记录下来,同时记录变化时刻的时间信息。

为了保证各层网络的时间同步精度,且避免时间同步误差不在层间跨层累计,各层采用层内局部时间对本层节点进行时间同步,层间时间转换由各层对应的控制

节点完成。当测试启动后,各层对应的控制节点周期性向其所在网络层中的其他各个节点发送本层网络局部时间同步信息,其他网络节点(包括 IO 监测节点和下层网络控制节点)接收到同步信息后,更新自己收到的时间同步状态,并用本次更新状态后的时间信息对本次同步之后、下次同步前采样到的 IO 时序结果进行标记,并将结果发送到本层网络的控制节点,由本层控制节点将本层时间转换为上层网络时间。

每层的监测节点实时监测其对应的 IO 时序的上升沿和下降沿状态,并以本层网络时间记录每个沿监测到的时刻,最后发送到本层的控制节点。为了实时监测敏感 IO 时序状态的变化同时对 IO 变化过程进行记录,选择 BM3101 型 SoC^[12]作为 IO 监测节点的处理节点,将 IO 时序变化以中断信号的形式提交到监测节点 SoC 中,确保了 IO 时序变化响应的实时性,同时将若干次 IO 变化历史记录在 IO 状态变化寄存器中,防止 SoC 对 IO 变化状态查询不及时,导致时序解析丢失信息。

为了提高每层控制节点的可靠性,对每层网络可以设置一主一从两个功能互为冗余的控制节点。在无故障的状态下,由主控制节点完成对本层网络的管理功能。主从控制节点以心跳信息作为健康依据,从控制节点实时读取主控制节点的时间同步信息和心跳信息。当心跳信息和时间同步信息多次不一致时,则判定主控制节点发生故障,控制权由当前主控制节点切换至从控制节点。心跳信息和时间同步信息不一致次数阈值可根据系统的可靠性的兼容性需求设计确定。

在 IO 时序监测网络结构中,每一层的网络可以选择不同的物理总线实现,如 1553B、GLink、串行总线、以太网等。

通过本方法进行 IO 时序监测网络分层扩展架构设计,各层网络和相邻网络局部独立,可以对网络和节点进行无限扩展,从而实现大规模分布式 IO 监测。

2 多级异构网络的时间同步策略

为了使系统中的各个子设备能够按照同样的时钟节拍运行,使各个设备的动作先后顺序具有确定性,同时使测试结果数据具有可比较性,特别是来自不同层网络、不同监测节点的时序测试结果具有整体表征意义,需要对系统中的各个设备进行时间同步。为了兼容父子层的不同网络类型,特别是不具备广播性质的两类网络间,时间同步需要能够在不同类型的网络间完成,同时在时间同步的过程中,不应影响监测节点基于时间的控制功能。

将 IO 时序监测网络的相邻两层定义为父层节点和子层节点,本节描述了一种混合网络中的级联式时间同步

方法，解决了混合异型混合网络中多层网络节点的时间同步问题，摆脱了单层网络同步技术对时间同步的站点数量的限制。

混合网络之间的时间同步转换过程如下所示。

(1) 父层网络控制站点向父层网络中的其他站点发送时间同步消息，时间同步消息包括预估的其他站点预计收到所述时间同步消息时刻的父层网络时间 T_{syn} ；

(2) 父层中非本层网络控制站点的其他节点在收到时间同步消息之后，解析所收到的同步时间 T_{syn} ，同时记录收到时间同步消息时刻对应的本地计时时间 T_{local_syn} ；

(3) 父层中的子层控制站点根据自身的子层系统时间 T_{local} ，对其控制的子层网络中的其他站点进行时间同步，方式同步步骤 (1) 和步骤 (2)；

(4) 子层控制站点收到子层网络中的时序监测站点发送的带有子层网络时间 t_{sub} 标记的时序结果数据后，采用最新得到的父层时间系统时间 T_{syn} 和相应的时间同步消息收到时刻的本地时间 T_{local_syn} ，将 t_{sub} 转换为父层网络中的时间值 $T_{afterSyn}$ ，形成父层网络时间，然后用其修正所收到的时序结果数据中的时间信息，并反馈至父层网络的控制节点，从而实现混合网络之间的时间同步转换。

为了保证监测可以持续的时间不受限制，父层网络向本网络时序监测站点和本网络中子层网络控制节点同步时间和子层网络向父层网络发送时序测试结果的时间格式定义如表 1 所示，即以年、月、日、时、分、秒、毫秒为表示形式的时间值。在子层网络进行时间同步转换时，以毫秒为单位定义时间修正量，将本层网络修正为父层网络时间。

表 1 时间信息结构定义

序号	字名称	数据类型	说明
1	年月	字	高字节为年，0 表示 2000 年，最大值 255 表示 2255 年。低字节为月，值域为 [1, 12]
2	日时	字	高字节为日，值域为 [1, 31]。低字节为时，值域为 [0, 23]
3	分秒	字	高字节为分，值域为 [0, 59]。低字节为秒，值域为 [0, 59]
4	毫秒	字	值域为 [0, 655 35]
5	CRC 校验码	字	

将父层网络时间标记转换成毫秒单位时间的转换操作定义为 $Tr(t)$ ，转换后的毫秒时间可以时间同步周期的长度作为舍入依据，进行截断舍入；毫秒时间标记转换成父层网络时间标记表达方式的转换操作为 $Tr^{-1}(t)$ ，则 $T_{afterSyn}$ 可以表示如下：

$$T_{afterSyn} = T_{syn} + Tr^{-1}(Tr(t_{sub}) - Tr(T_{local_syn})) \quad (1)$$

其中， $Tr^{-1}(Tr(t_{sub}) - Tr(T_{local_syn}))$ 为网间交换时间差。

通过本方法将时间信息周期性自顶向下，从总到分，传输到每个层和节点，确保系统中各个节点的时间始终与系统保持一致，各层网络计时局部独立，不会对时间误差积累和传播，从而保证所检测的 IO 时序标记的时间信息精度。

3 异步 IO 时序的实时测量

常见的 IO 时序监测方法可以分为两类：循环采样法和变化检测法。循环采样法的基本思路为通过高密度采样被检测的 IO 时序，并记录每次采样的结果，通过对采样结果进行统计来获得 IO 时序在高低电平的时域分布。这种方法的优点在于可以方便于实时获知每一时刻 IO 时序的电平状态，这种方法能够监测的 IO 时序的最小脉宽依赖于采样处理效率，对于实时统计 IO 时序历史变化规律，电平持续时长以及电平变化时刻就显得有些力不从心了，获得较好的噪声监测结果通常需要较复杂的硬件和软件配合才能完成；另一类变化检测法通过中断来响应 IO 时序变化的上下沿，通过检测 IO 时序的上升沿和下降沿来判别 IO 时序的状态变化时刻，这种方法只记录上下沿的变化状态，不统计每个采样时刻的信息，从而可以在误差范围内适当降低采样频率要求，允许多任务交叉运行，但这种方法能够监测的 IO 时序的最小脉宽较大依赖于中断服务处理的时间。若在第一个 IO 时序变化还未处理完毕时再发生新的 IO 时序，则会造成新变化状态的丢失。

为了克服软件响应不及时导致在一次中断延时中发生的多次 IO 时序变化丢失情况，以及在异步解析数字量变化信息和记录数字量变化信息的读写冲突所导致的数字量信息丢失或变化时间信息记录不正确所带来的技术困难，实现高实时可靠地对系统中 IO 时序的变化信息进行监测，本节将循环采样和中断的方法相结合，提出了一种有限记录并查询历史状态信息的 IO 状态采样方法，该方法针对同一组 IO 输入，定义了两类采样端口：IO 时序状态端口和 IO 时序异步数据端口。IO 时序状态端口用于实时记录 M 个并行的 IO 时序状态， $M \geq 1$ ；IO 时序异步数据端口用于实时存储最多 N 次最近的 IO 时序状态端口值和变化次数，当 IO 时序状态端口发生变化时，IO 时序异步数据端口立即更新，并产生中断请求。

当每次检测到中断请求时，按如下方式分析 IO 的状态变化过程。

(1) 通过读取 IO 时序异步数据端口，同时获取最多 N 次最近的 IO 时序状态端口值和变化次数，将所述变化次数记为 $nCount$ ，SoC 对 IO 时序异步数据端口进行读操

作对 IO 时序异步数据端口中存储的变化次数具有“读清”功能,且清零操作优先级高于 IO 时序异步数据端口更新时的写入操作。

(2) 读取 IO 时序状态端口,获取 IO 时序状态端口值。

(3) 根据最近获得的 IO 时序异步数据端口所记载的最多 N 次最近的 IO 时序状态端口值和当前的 $nCount$ 值,分析 IO 时序异步数据端口所记载的每个 IO 时序最近 $nCount$ 次翻转变化情况。

(4) 比较最近获得的 IO 时序异步数据端口所记载的最新的 IO 时序状态端口值与最近从 IO 时序状态端口获取的 IO 时序状态端口值是否一致,如果一致,结束中断处理过程;否则,转入步骤(5)。

(5) 将最近获得的 IO 时序异步数据端口所记载的最多 N 次最近的 IO 时序状态端口值和 $nCount$ 保存,按照步骤(1)的所述的方式再次读取 IO 时序异步数据端口,获得 IO 时序异步数据端口所记载的最多 N 次最近的 IO

时序状态端口值和变化次数。

(6) 再次读取 IO 时序状态端口,获取 IO 时序状态端口值。

(7) 对比最近两次读取 IO 时序异步数据端口所获取最多 N 次最近的 IO 时序状态端口值,分析得到最近两次读取 IO 时序异步数据端口操作之间 IO 时序状态端口值的真实变化次数,将此变化次数记为 $nCount$,重新执行步骤(3)~步骤(7)。

处理流程如图 2 所示。

在最近获得的 IO 时序异步数据端口所记载的最近 $nCount$ 次翻转的 IO 时序状态端口值中,按照从旧到新的顺序,将每一个 IO 时序状态端口值与该 IO 时序状态端口值变化前的 IO 时序状态端口值进行位比较,得到每一次变化时每一个 IO 时序翻转变化情况,直至所有 $nCount$ 次最近的 IO 时序状态端口值都对完毕,得到所有 $nCount$ 次 IO 时序翻转变化情况,第一次执行本步骤时,默认 IO 时序状态端口值变化之前的值为 IO 时序初始状态值。

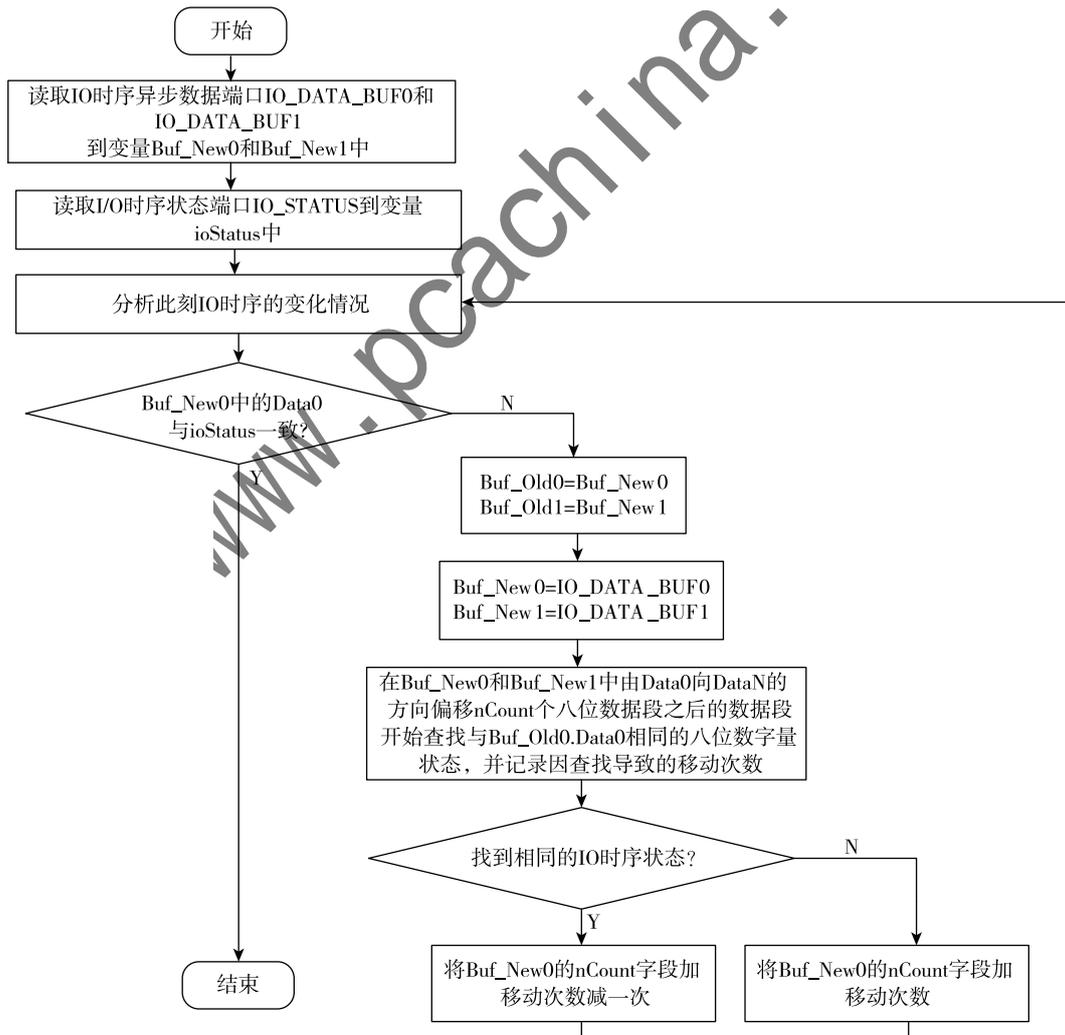


图 2 异步 IO 型时序的实时测量

对比最近两次读取 IO 时序异步数据端口所获取最多 N 次最近的 IO 时序状态端口值，分析得到最近两次读取 IO 时序异步数据端口操作之间 IO 时序状态端口值的实际变化次数的具体处理过程如下所示。

(1) 提取最近读取的 IO 时序异步数据端口所记载的变化次数，将该变化次数作为最近两次读取 IO 时序异步数据端口操作之间 IO 时序状态端口值实际变化次数估值 C 。

(2) 在最近读取的 IO 时序异步数据端口记载的 IO 时序状态端口数据中，找到 IO 时序状态端口发生 C 次变化之前的最新的 IO 时序状态端口值，并判断该值与次近读取的 IO 时序异步数据端口所记载的最新的 IO 时序状态端口值是否相同，如果相同，则认为当前实际变化次数估值 C 就是两次读取 IO 时序异步数据端口之间 IO 时序状态端口值实际发生的变化次数，结束；否则，执行步骤 (3)。

(3) 采用当前实际变化次数估值 C 加 1，更新实际变化次数估值 C ，当更新后的实际变化次数估值 C 等于 N 时，将 N 作为最近两次读取 IO 时序异步数据端口操作之间 IO 时序状态端口值的实际变化次数，结束；当所述更新后的实际变化次数估值 C 小于 N 时，重复执行步骤 (2) ~ 步骤 (3)。

为了提高 IO 时序监测的可靠性，IO 时序状态端口值应取对 IO 时序异步端口进行滤波处理去除毛刺或者干扰之后的值。

本方法利用软件设计的策略，可有效避免在中断响应时延的 IO 时序问题以及在异步 IO 读写冲突时造成的 IO 时序状态丢失问题，在无需增加额外硬件功能支撑的情况下，有效提高了时序 IO 时序测试的可靠性。

4 实验结果分析

时序测试的时间精度和多层间时序测试精度的一致性是两个重要的性能指标。前者反映了时序测试的准确性，其表征了系统运行的健康状态，也是进行时序测试重要意义所在；后者通过对扩展过程引入的误差进行分析，可以间接反映出大规模 IO 时序监测性能指标的偏差上限。

基于某航天型号控制系统的 IO 时序监测系统，如图 3 所示，对本文方法的上述两个指标进行了分析和讨论。

该 IO 时序监测系统分为三层网络，第一层网络为 1553B 总线层，第二层为 RS485 串行总线层，第三层为 Glink 总线层，系统可监测 112 路时序，每个测试节点模块可测 16 路时序，第一层有两个测试节点，可监测 32 路

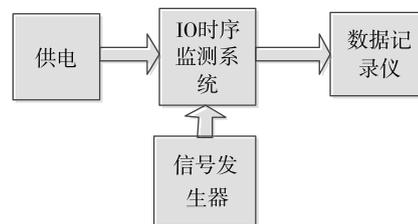


图3 IO时序监测测试系统框图

时序，第二层有三个测试节点，可监测 48 路时序，第三层有两个测试节点，可监测 32 路时序。时序监测系统框图如图 4 所示。

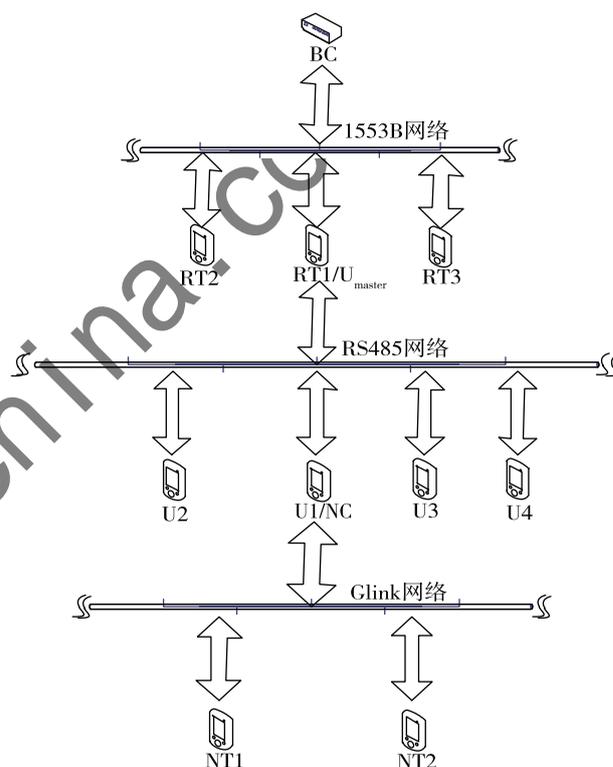


图4 IO时序监测系统框图

4.1 单路 IO 时序精度测试

利用波形发生器产生占空比 50%，频率 10 Hz 的 PWM 波输入采集系统的一路 IO，测试时长 20 s，通过 100 个时序开始上升沿的时刻标准值与 IO 时序监测系统实测值比较，测出时序精度 2 ms，统计的误差如图 5 所示。

4.2 多路 IO 时序测试误差一致性测试

利用波形发生器产生占空比 50%，频率 10 Hz 的 PWM 波，同时输入 IO 监测系统的所有 IO 采集口，通过各个时序开始时刻的标准值与采集系统的实测值比较，对每级网络各节点的误差和各网络的节点误差统计分析，测得同层之间节点时序测试误差 2 ms，不同网络层节点

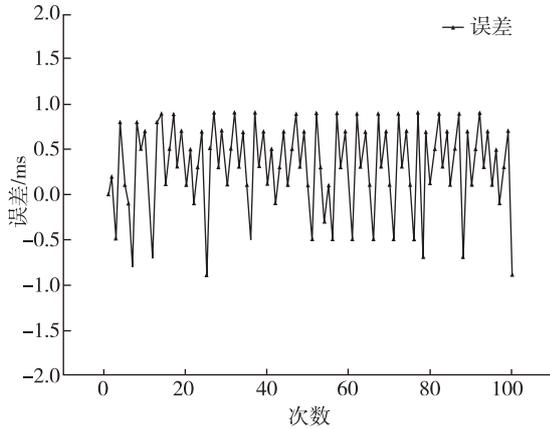


图5 单 IO 时序测量精度统计

时序测试误差 2 ms, 不同网络层节点的精度误差一致性较好, 统计的误差如图 6 所示。

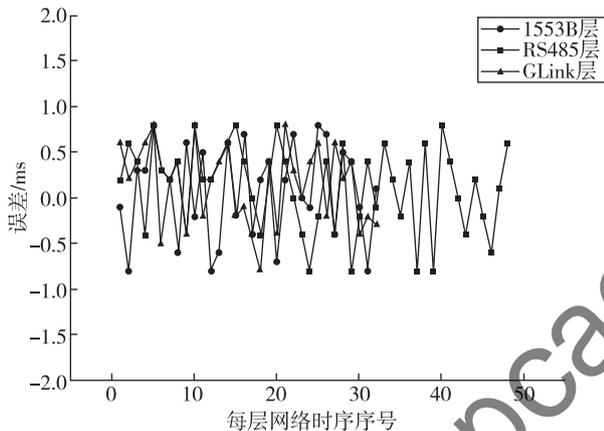


图6 多路 IO 时序测量误差一致性统计

现有的 IO 时序测试系统基于单个芯片或者处理器, 受到芯片或处理器体积和接口数量限制, 使得能够同时采用这种形式的信号数量较少, 不能大规模扩展, 并且受限于采集周期, 现有系统测试误差都在 ± 10 ms, 并且对异步时序采集可能出现 IO 时序丢失问题。

综上所述, 本文 IO 时序监测方法即可保证 IO 时序测试精度, 也不会因网络不同和层次的增加而产生误差积累, 各层级间误差一致性较高。

5 结论

本文提出了一种可扩展的高精度可靠 IO 时序测试方法, 该方法基于层次化的网络架构, 使得 IO 时序监测可以从单机内部或者单网络规模方便地扩展到任意规模, 同时混合网络间的时间同步方法可以适应分布式的 IO 时序监测具有统一的时钟基准的监测需求。通

过综合分析 IO 时序状态历史缓存和实时状态记录, 解决了因处理器响应时延以及 IO 时序口读写冲突等情况产生的时序测量异常问题, 提高了 IO 时序状态采样的可靠性。

实验结果表明, 本文方法在有效提高了 IO 时序容量规模的同时, 保证了每个时序的测量时间精度, 同时这种测量精度不会随层级增加而积累, 各层级间误差一致性较高。

参考文献

- [1] 王琪, 章文晋, 李建宏. 基于单片机的时序测试系统设计 [J]. 电子测量技术, 2015, 38 (4): 74-77.
- [2] 王乔, 王民钢, 路建中. 飞控计算机硬件及全时序综合测试装置设计 [J]. 计算机测量与控制, 2018, 26 (5): 29-32.
- [3] 盖猛, 王吉平, 曾祥来. 一种安全型数字量采集电路 [J]. 电子质量, 2021 (1): 18-25.
- [4] 留黎钦, 林志雄, 江永增. 一种基于 CAN-BUS 的数字量采集器设计 [J]. 计算机系统应用, 2014, 23 (8): 218-221.
- [5] 曹洪龙, 胡剑凌, 邵雷, 等. 新型便携式多路开关量信号检测设计 [J]. 仪表技术与传感器, 2014 (12): 127-130.
- [6] 刘素华. 多通道模拟与数字接口混合采集装置设计与实现 [D]. 太原: 中北大学, 2012.
- [7] 于洋, 刘晶, 王迪, 等. 多格式信号采集处理模块的研究实现 [J]. 电子设计工程, 2020, 28 (21): 93-97.
- [8] 郑来波, 李泰强. 多路高速开关信号采集中的一种软件消抖算法 [J]. 山东工业大学学报, 2002, 32 (1): 48-51.
- [9] 何春华, 鄢仁成. 开关量的实用软件处理方法 [J]. 计算机测量与控制, 2005, 13 (4): 399-400.
- [10] 许天骄, 吴胜华, 张楠楨, 等. 提高数字量输入信号采集回路安全性的方法分析 [J]. 仪器仪表用户, 2021, 28 (8): 26-28.
- [11] 陈晨. 车载 ATP 数字量采集模块的研究与设计 [D]. 北京: 北京交通大学, 2011.
- [12] SPARC International, Inc. The SPARC architecture manual version 8. New Jersey: Prentice-Hall, 1992.

(收稿日期: 2023-05-04)

作者简介:

苏平 (1986-), 通信作者, 男, 硕士, 工程师, 主要研究方向: 嵌入式系统等。E-mail: spfeing@126.com。

高飞 (1981-), 男, 博士, 高级工程师, 主要研究方向: 计算机科学与技术、机器学习等。

刘凯 (1987-), 男, 硕士, 工程师, 主要研究方向: 嵌入式系统。