

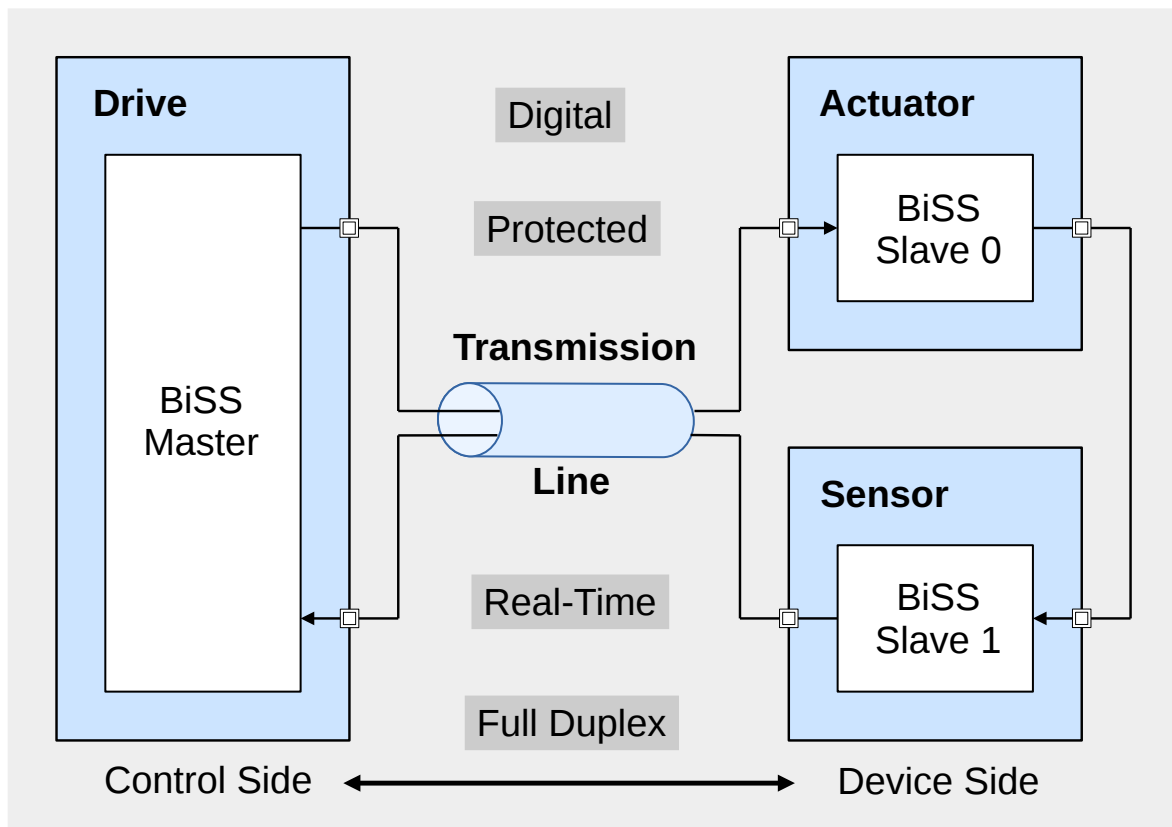
### 特性

- 传感器/执行器接口
- 异步、实时数据传输
- 快速、串行
- 永久双向
- 点对点或多从属网络
- 结构紧凑，成本效益高
- 开放式标准

### 应用

- 驱动器
- 旋转/线性编码器
- 机器人
- 智能传感器
- 安全执行器

### 模块框图



### 目录

<b>概述</b>	<b>3</b>	<b>CMD = "11" (用户定义/保留)</b>	<b>19</b>
<b>点对点连接</b>	<b>4</b>	短 <i>BiSS</i> 帧	19
<b>BiSS 帧</b>	<b>6</b>	精简 <i>BiSS</i> 帧	19
闲置	6	<b>BiSS 内存图</b>	<b>20</b>
数据头	6	寄存器保护级别	20
数据通道	7	存储模块选择	20
超时	7	EDS 存储模块	21
静态超时	7	电子数据表	21
自适应超时	7	用户存储模块 (可选)	21
线路延迟	8	<b>BiSS 类别ID</b>	<b>21</b>
BiSS 周期	9	设备序列号	21
<b>过程数据通信</b>	<b>9</b>	空闲的寄存器	21
锁存点	9	制造商ID和设备ID	22
处理时间	10	<b>总线连接</b>	<b>23</b>
菊花链	10	<b>应用提示</b>	<b>26</b>
总线复位/初始化	12	初始化示例	26
传播延迟	12	BiSS周期时间计算	27
空值	13	N个从机的一般计算方法:	27
<b>控制通信</b>	<b>13</b>	单个从机的简化计算方法:	27
寄存器通信	15	总线耦合器	28
寄存器读取访问	15	<b>特性</b>	<b>29</b>
寄存器写入访问	17	<b>缩略语表</b>	<b>31</b>
<i>BiSS</i> 命令	17	<b>修订历史</b>	<b>32</b>
CMD = "00" (单周期数据通道)	19		
CMD = "01" (控制通信)	19		
CMD = "10" (总线耦合器/用户定义)	19		

### 概述

**BiSS (双向/串行/同步)** 是一种用于快速、安全地传输异步过程数据的数字式串行接口协议，尤其适用于电机反馈系统。在实时接收传感器过程数据和传输执行器过程数据的同时，**BiSS** 协议能够在不中断过程数据流的情况下进行寄存器数据传输。

#### i

如今，**BiSS** 接口 (缩写为 "**BiSS**") 指的是 2007 年推出的 **BiSS C** 协议。新设计不推荐使用以前的协议定义 (如 **BiSS B**)。

驱动装置中的一个 **BiSS** 主模块与一个或多个 **BiSS** 从模块相连，这些从模块包含在系统的传感器和执行器设备中。一般情况下，**BiSS** 从模块通过三个信号与 **BiSS** 主模块连接：用于同步的时钟输入信号 (**Clock**)、传输至 **BiSS** 从模块的数据输入信号 (**Data In**) 和从 **BiSS** 从模块接收的数据输出信号 (**Data Out**)。在下面的文字中，输入和输出的信号方向始终参照前面所述的 **BiSS** 从属模块。

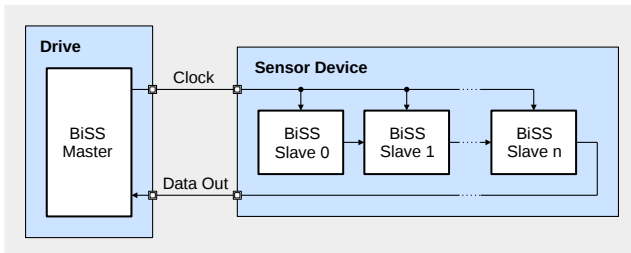


图 1：点对点连接：主设备连接单个传感器设备

在适用于仅由一个传感器设备组成的系统的点对点连接中，驱动器仅通过时钟线和数据输出线与传感器设备连接。数据输入线被省略。因此，点对点配置的 **BiSS** 设备在硬件上与仅支持 **SSI** 协议的驱动器兼容。

如图 1 所示，通过将数据输出线转发到每个后续 **BiSS** 从模块，最后再返回到 **BiSS** 主模块，可以在一个传感器设备内以菊花链方式连接多个 **BiSS** 从模块。

随后，**BiSS** 主机和 **BiSS** 从机分别表示 **BiSS** 主模块和 **BiSS** 从模块，与 **BiSS** 设备相对应。

**BiSS** 主机和 **BiSS** 从机之间的数据交换由连续的 **BiSS** 帧控制，这些帧在可配置的 **BiSS** 周期内异步触发。一个 **BiSS** 帧分为多个逻辑进程数据通道，这些通道分别受到循环冗余校验 (**CRC**) 的保护。每个 **BiSS** 从机占用一个逻辑进程数据通道，每个 **BiSS** 帧都通过整个菊花链串行传输。因此，**BiSS** 的最短周期时间取决于当前的进程数据通道配置。

在寄存器数据传输中，每个 **BiSS** 帧都包括一个控制数据位，分别向和从 **BiSS** 从机方向传输。这些比特从连续的 **BiSS** 帧中收集，形成流量较慢的次级传输帧。控制数据帧用于与特定 **BiSS** 从机之间的次级数据传输。每个 **BiSS** 从机都会自动分配一个唯一的标识号，该标识号由其在菊花链中的位置决定。

寄存器数据传输通常用于配置/校准目的以及电子数据表形式的设备信息。不过，控制数据帧也可用于传输更多静态的传感器数据，如温度。

除寄存器数据传输外，控制数据帧还提供预定义和自定义命令，这些命令可寻址到特定的 **BiSS** 从机或广播到连接到系统的每个 **BiSS** 从机。**BiSS** 命令可用于简化 **BiSS** 主机的启动程序，或同步与整个系统的特定应用有关的特定事件，例如位置数据预设。为了确保即使在高干扰环境下也能正常传输数据，除了 **BiSS** 帧的数据保护外，每个控制数据帧还受到自身 (**CRC**) 循环冗余校验的保护。

对于拥有多个传感器设备的系统，则使用总线连接。与点对点连接不同，总线连接还可以将数据传输到执行器设备。因此，用于数据输入的第三条线会连接到 BiSS

从机。图 2 显示了使用总线配置连接到 BiSS 主机的几个 BiSS 从机。

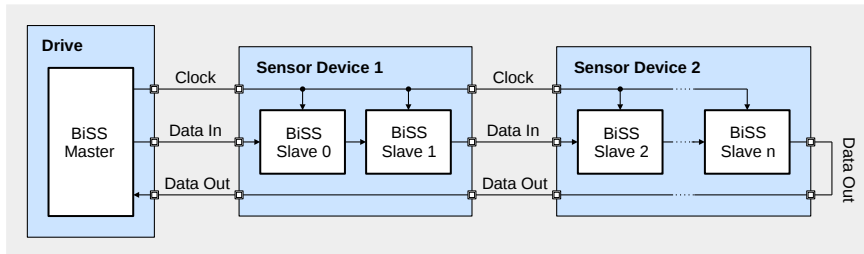


图 2: 总线连接: 主设备连接多个菊花链式传感器设备

由于菊花链实际上是一个长移位寄存器，因此执行器数据被移入菊花链，而传感器数据被移出菊花链。因此，代表执行器的 BiSS 从机使用相同的过程数据通道，其配置属性与点对点连接相同。总线连接甚至可用于菊花链整个 BiSS 设备，其中包括多个 BiSS 从属模块，以创建更复杂的 BiSS 系统。

该机制旨在减少针对高精度控制回路应用的时间抖动。此外，还可以在 BiSS 帧开始时延迟过程数据传输，以支持需要额外内部处理时间的设备。

为了同步采集和应用整个系统的过程数据，BiSS 协议提供了一种机制，可用于在连接到 BiSS 主机的每个 BiSS 从机内同时触发数据处理。

BiSS 周期通过预定义的超时来同步，每个 BiSS 从机在 BiSS 帧结束时各自识别超时。为了便于执行，BiSS 主机自动补偿因传输线长和线路驱动器慢而造成的线路延迟。由于具有广泛的数据保护功能，BiSS 协议还适用于高达 SIL3 的安全关键型应用，符合 **BiSS Safety Profile**。

### 点对点连接

如 "概述" 一章所述，BiSS 主要用于实时控制回路应用中的多从机异步过程数据传输。为此，一个 BiSS 主设备连接一个或多个 BiSS 从设备。单个 BiSS 主设备只包括一个 BiSS 主设备，而 BiSS 从设备则可能包括多个 BiSS 从设备。

不过，在点对点配置中，BiSS 主设备只连接到一个 BiSS 从设备。最常见的有且仅有一个 BiSS 从设备的点对点连接，如图 3 所示。

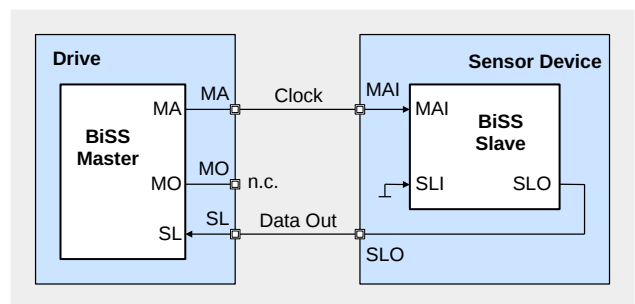


图3: 单个 BiSS 从机的点对点连接



在点对点连接中，只有一个 BiSS 从设备与主设备相连。BiSS 从机设备只能包含传感器。

BiSS 主机通过 MA 引脚与时钟线相连，通过 SL 引脚与数据输出线相连。数据输入线的 MO 引脚不使用，因此无法在点对点配置中使用执行器 BiSS 从机。

传感器设备内的 BiSS 从机通过其 MAI 引脚与时钟线相连, 通过其 SLO 引脚与数据输出线相连。由于不使用数据输入信号, BiSS 从机的 SLI 引脚接地。

**i** 请注意, 在仅用于点对点连接的设备中, 数据输入引脚 SLI 在设备外部不可访问的。该引脚内部接地。

MA 引脚产生的时钟信号控制通信帧。它为 BiSS 从机提供时钟源, 并定义传输周期的开始和结束。此外, 时钟信号还用于从 BiSS 主机向 BiSS 从机传输单个控制数据位, 以进行控制通信。有关时钟信号和控制通信的更多信息, 请参见 "BiSS 帧" 和 "控制通信" 两章。

引脚 SLO 生成的数据输出信号用于将 BiSS 从机的传感器数据传输回 BiSS 主机。此外, 数据输出信号还用于确定适当的 BiSS 主机的数字采样点, 并传输 BiSS 从机的响应以进行控制通信。基于考虑长传输线而造成的信号延迟有关的数字采样点灵活性的更多信息, 请参阅 "线路延迟" 一章。

一般来说, BiSS 是一种数字协议, BiSS 主机和 BiSS 从机在时钟信号 MA 的上升沿产生数据信号。BiSS 从机的采样点位于 MA 的下降沿。如前所述, BiSS 主机对时钟信号 MA 的采样时间并不固定, 可用于补偿线路延迟。不过, 作为介绍的开始, 我们可以假设系统不引入任何线路延迟, 并且 SL 在 MA 的时钟周期内的适当时间点采样, 如图 4 所示。

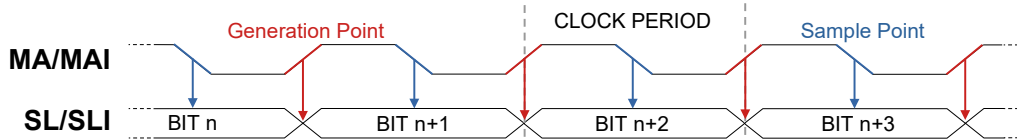


图 4: BiSS 数据的生成和采样

除其他因素外, 图 4 中的时钟周期会影响 BiSS 的数据传输速率。时钟的频率范围和占空比在 "特性" 一章中作了规定。除了 BiSS 协议本身的限制外, 时钟信号也会收到传输线的影响。

如图 5 所示, 建议遵循标准 RS422 协议进行正确的数据传输。有关 RS422 的更多信息, 请参阅 [Telecommunications Industry Association](#) 的 TIA-422 标准。

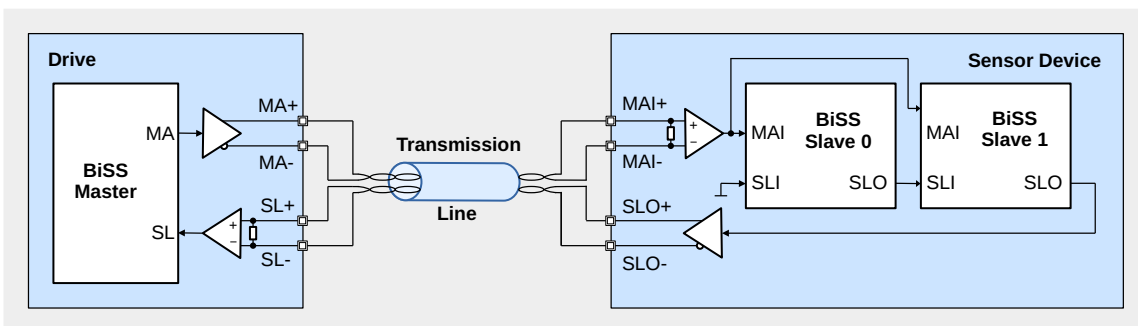


图 5: RS422 点对点连接

如有必要, 即使是点对点连接, 也可以菊花链方式连接多个 BiSS 从机。图 5 显示了一个带有两个互联 BiSS 从机的传感器设备示例。但是, 对于点对点连接, 则无法将多个传感器设备连接在一起。

更多有关于 BiSS 从机菊花链连接方式的信息, 请参阅 "过程数据通信" 一章。

### BiSS 帧

由于 *BiSS* 帧不会因为不同的物理连接而发生较大变化，因此可以参照第 4 页图 3 所示的简单系统来介绍其结构。该系统下的一个 *BiSS* 帧示例可以参考图 6。下文描述与 *BiSS* 从机的 MAI 和 SLO 信号相对应。

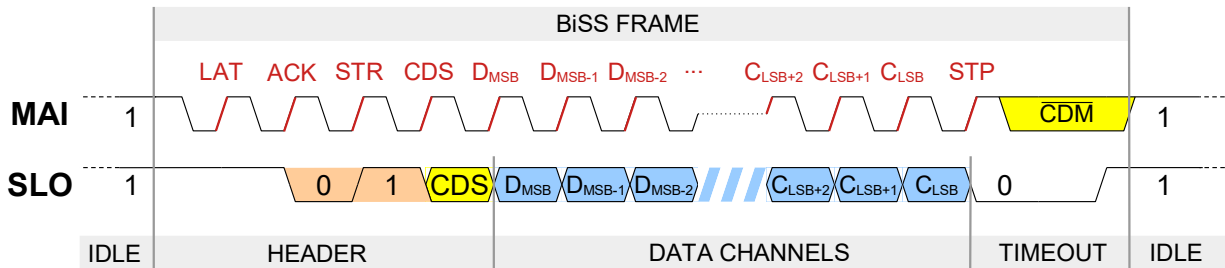


图 6: 一个简单的 *BiSS* 帧

如“概述”一章所述，*BiSS* 帧由 *BiSS* 主机通过 MAI 引脚上的时钟信号控制。MAI 序列可分为四个主要阶段：

- 闲置
- 数据头
- 数据通道
- 超时

闲置期间不传输数据。数据头初始化新的 *BiSS* 帧，数据通道用于传输过程数据。超时使 *BiSS* 从机同步，为下一个 *BiSS* 周期做好准备。

每个阶段的长度实际上取决于 *BiSS* 从机的特性和当前的 *BiSS* 配置。每个阶段（数据头、数据通道和超时）决定 *BiSS* 系统支持的最短 *BiSS* 周期时间。*BiSS* 主机需要进行相应配置。有关 *BiSS* 周期的更多信息，请参阅 *BiSS* 周期一节。不过，为了便于理解，最好理解最简单的情况。下文将讨论图 6 中的不同阶段。

#### 闲置

由于 *BiSS* 帧通常在有规律的周期内传输，因此在当前周期结束后和下一帧开始前会有一个阶段。这个阶段称为空闲。通常情况下，空闲期间不传输数据，MAI 和 SLO 引脚均持续为高电平 (1)。唯一的例外是 Hold-CDM 功能，该功能将在“控制通信”一章中介绍。

#### 数据头

图 7 显示了一个典型的 *BiSS* 帧中的数据头，它启动了一个新的 *BiSS* 周期，并准备通过 SLO 来传输 *BiSS* 从机中的数据。

*BiSS* 帧由 MAI 传输线上的下降沿和上升沿启动。第一个上升沿 LAT（锁存）被用作触发器，在 *BiSS* 从机内锁存或生成过程数据。有关数据同步的更多信息，请参阅“锁存点”一节。

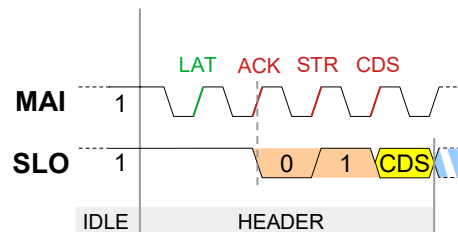


图 7: 一个简单的 *BiSS* 帧的数据头

如“点对点连接”一章所述，*BiSS* 从机在 MAI 上的时钟信号上升沿在 SLO 上产生响应。因此，除 LAT 外，每个上升沿代表 *BiSS* 帧的一个比特。第一位 (ACK) 用于 *BiSS* 从机向 *BiSS* 主机确认新 *BiSS* 周期的开始。由于 SLO 的空闲阶段为“1”，因此 ACK 对应的确认位始终为“0”。有关 *BiSS* 主机如何处理确认位的详细信息，请参阅“线路延迟”一节。

起始位 (STR) 表示 *BiSS* 从机已准备好向 *BiSS* 主机传输过程数据。如“过程数据通信”一章中的“处理时间”一节所述，对于给系统带来处理时间的设备，起始位可以延迟。在起始位之后，实际数据传输从控制数据从属位 (CDS) 开始。与过程数据不同，CDS 位是控制通信的一部分，因此同样属于 *BiSS* 帧当中的数据头部分。关于控制通信部分的解释请参见“控制通信”一章。

### 数据通道

数据头之后是数据通道, 如图 8 所示。在 CDS 位之后, 数据通道为每个连接的 BiSS 从机传输多达 64 位的处理数据。由于上述示例中只包含了一个 BiSS 从机, 因此该 BiSS 帧只传输一个数据通道 ( $D_{MSB} \dots D_{LSB}$ )。每个数据通道通常由最多 16 位的循环冗余校验保护 ( $C_{MSB} \dots C_{LSB}$ )。CRC 位在传输前被反转。数据通道在每个 BiSS 周期内被全部传输, 因此处理能力最强。有关数据通道的更多信息, 请参阅“过程数据通信”一章。

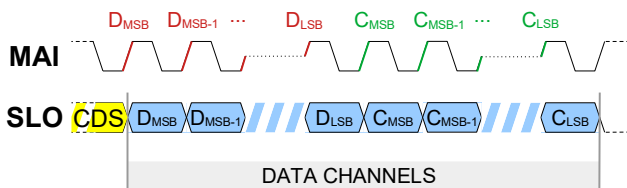


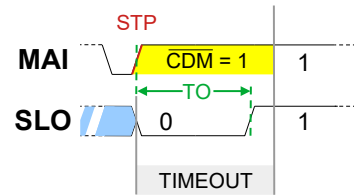
图 8：简单 BiSS 帧的数据通道

### 超时

BiSS 帧的最后一个阶段是超时。超用于同步 BiSS 从机和 BiSS 主机的 BiSS 周期结束。为此, 提供了一个握手机制。BiSS 主机停止 MAI 上的时钟信号, 每个 BiSS 从机单独测量超时 (TO)。超时后, BiSS 主机将通过 SLO 的上升沿来指示超时, 如图 9 所示。因此, BiSS 从机在 MAI (STP) 的最后一个上升沿产生停止位。为了传输用于控制通信的主机数据控制位 (CDM), BiSS 从机在超时结束时对时钟线 MAI 进行采样。

握手机制确保 BiSS 主机在所有连接的 BiSS 从机都准备好处理另一个 BiSS 帧之前, 不会启动另一个 BiSS 周期。如前所述, 每个 BiSS 从机都能独立确定自己的超时时间间隔。因此, 在时钟线 MAI 停止切换时, BiSS 从机会启动一个内部定时器。定时器到期时, 数据输出 SLO 设置为 "1", CDM 位被评估, BiSS 从机回到空闲状态。如果正确生成了停止位 "0", BiSS 主机将收到 SLO 上升沿超时通知, 并进入空闲状态。现在, 整个系统已准备就绪, 可以开始新的 BiSS 周期。

### CDM = 0:



### CDM = 1:

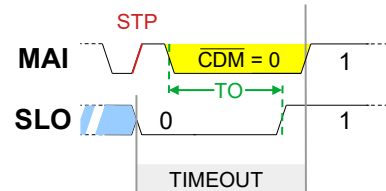


图 9：简单 BiSS 帧的超时

标准 BiSS 系统支持两种类型的超时:

- 静态超时
- 自适应超时

静态超时是 BiSS 从机中预先定义的一个恒定时间间隔 (通常约为 20s)。而自适应超时则在每个 BiSS 帧开始时根据当前 BiSS 时钟频率进行调整, 为高性能应用的尽量缩短周期时间。

### 静态超时

图 10 显示了静态超时的 BiSS 帧。对于高 MAI 频率, 超时间隔比时钟周期 ( $T_{MA}$ ) 长的多。

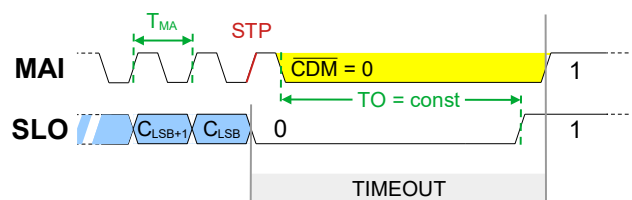


图 10：静态超时

静态超时的限制在“特性”一章中规定。由于每个 BiSS 从机至少需要包含有静态超时, 因此 BiSS 主机可以随时取消 BiSS 帧。在等待预定义的最大静态超时间隔后, 每个 BiSS 从机必须处于闲置状态, 并准备好开始新的 BiSS 周期。

### 自适应超时

如图 11 所示, 自适应超时取决于 MAI 当前配置的时钟频率。

自适应超时时间间隔是 MAI 时钟周期 ( $T_{MA}$ ) 的 1.5 倍, 因此也兼容整个 BiSS 频率范围。

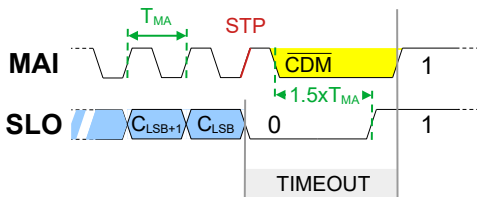


图 11: 自适应超时

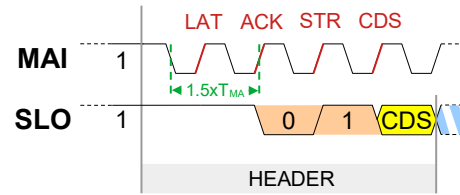


图 12: 测量自适应超时

与图 10 相比, 超时阶段更短, 因此 BiSS 周期时间更短, 吞吐量更高。不过, 自适应超时的精确度仅与 BiSS 从机的采样时钟相当。图 12 显示了 BiSS 数据头阶段自适应超时的测量结果。慢采样时钟可能会产生较长的自适应超时。支持自适应超时的 BiSS 设备在测量的前 1.5 个时钟周期时, 依然将静态超时设为极限值。

### 线路延迟

如前所述, 图 6 所示的 BiSS 帧表示 BiSS 从机的 MAI 和 SLO 信号。在实际应用中, BiSS 主机产生的时钟信号 MA 和 BiSS 从机产生的数据输出 SLO 通过传输线传输, 这可能会增加 BiSS 通信系统的信号延迟。因此, 上述示例中 BiSS 主机的 BiSS 帧可能如图 13 所示。

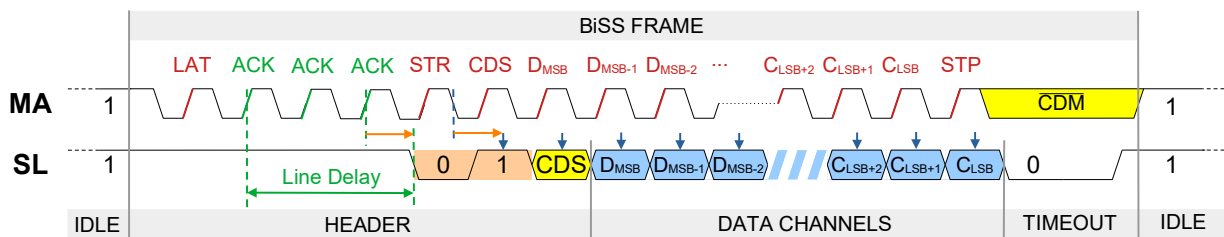


图 13: BiSS 主机中带有线路延迟的 BiSS 帧

由于 BiSS 从机对确认位 (ACK) 的响应延迟, 数据头的长度延长了两个时钟。BiSS 主机会在数据头中插入额外的 MA 时钟, 直到 SL 上收到确认位 “0”。此外, BiSS 主机会分析最后一个确认时钟周期内的延迟, 以调整其对剩余数据位的采样点。由于额外的 MA 时钟会移动整个帧, 因此接下来的 BiSS 通信顺序与之前描述的不同。

线路延迟补偿适用于具有长传输线的系统。此外, 甚至可以在传输线路中加入协议转换器, 而不会干扰 BiSS 通信。不过, BiSS 从机必须在一定的时间间隔内通过确认位做出响应。如果超过该时间间隔, 驱动器必须将其视为错误。关于标准 BiSS 系统允许的最大线路延迟请参考“特性”一章。



### BiSS 周期

由于 BiSS 可用于高精度控制应用中的异步过程数据传输，因此重复启动具有预设周期的 BiSS 帧是合理的。通常情况下，BiSS 主机会被设置去执行固定的 BiSS 周期。BiSS 周期时间由应用确定。不过，需要同时考虑最大 BiSS 帧的长度，从而确定可能的最短周期时间。更多最短周期时间的计算方法请参见“应用提示”一章当中“计算 BiSS 周期时间”部分。

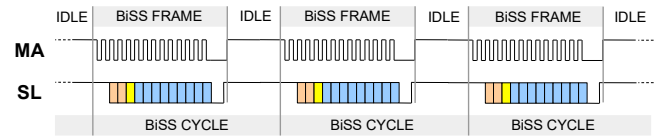


图 14: BiSS 异步周期

图 14 显示了几个由 BiSS 主机异步触发的 BiSS 周期。

**i** 在启动下一个 BiSS 周期之前，BiSS 帧必须在 SLO 上升沿指示的超时时间内结束。

### 过程数据通信

BiSS 帧一章中介绍到，数据通道被用于受到保护的，高吞吐量的异步过程数据传输。与 BiSS 主机连接的每个 BiSS 从机都可配置占用一个最多 64 位的数据通道，该通道受最多 16 位的循环冗余校验 (CRC) 保护。通常情况下，数据通道用于传感器和执行器数据传输，这些数据需要在每个 BiSS 周期内完整传输。为便于理解，本章只讨论传感器数据。执行器数据将在总线连接一章中解释。

图 15 显示了 BiSS 从机传输旋转编码器多圈和单圈位置数据的典型示例。不过，数据通道也可用于任意数据。BiSS 协议并未规定通过过程数据通信传输的数据类型和格式。为实现标准化，可使用预定义的 BiSS 配置文件和电子数据表 (EDS)。

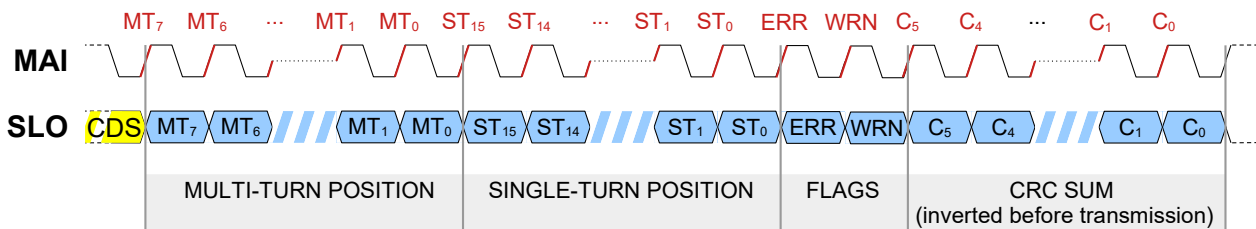


图 15: 典型的数据通道内容

参考第 4 页图 3 中的简单示例，该处只占用了—个数据通道，该数据通道被配置为 8 位多圈和 16 位单圈位置数据。此外，数据通道配有错误和警告标志的形式传输状态信息。整个数据通道受 6 位循环冗余校验保护。如果激活了另一个数据通道，附加数据位将直接跟随第一个数据通道。有关多数据通道的信息，请参阅菊花链部分。

对于高精度控制应用而言，定期生成新的传感器数据非常重要。因此，BiSS 帧提供了一种在每个 BiSS 周期开始时同步触发生成新传感器数据的机制，更多信息请参见“锁存点”部分描述。

如果 BiSS 设备需要大量时间来生成或准备其处理数据，则数据通道的传输可以延迟。数据头中的起始位会通知 BiSS 主机所需的处理时间。有关起始位延迟的更多信息，请参阅“处理时间”部分。

### 锁存点

对于高精度控制应用而言，在特定时间点生成传感器数据非常重要。因此，过程数据会在每个 BiSS 周期开始时，在 MAI 的第一个上升沿触发后被生成（见第 6 页图 7 中的 LAT）。

如果 BiSS 主机以异步方式触发 BiSS 周期，则 BiSS 帧 LAT 的采样边沿和过程数据的锁存点也以异步方式出现。如果 BiSS 帧的采样边沿经过处理，使过程数据的生成仅具有恒定延迟，则控制装置总是从 BiSS 周期开始时的一个固定时间点接收传感器数据。因此，数据生成的时间抖动被减小到 BiSS 从机设备的处理性能。图 16 显示了参照当前 BiSS 帧采样边沿的数据生成恒定延迟。

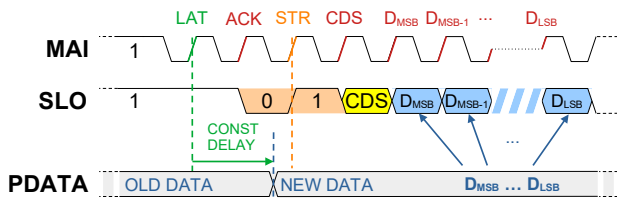


图 16: 数据生成的恒定延迟

数据生成过程由数据头内的 LAT 上升沿触发。BiSS 从机设备需要时间进行数据采集并将新的过程数据锁存到 BiSS 从机 (PDATA)。新的过程数据将在 CDS 位之后的数据通道中传输回 BiSS 主机。

在点对点连接中，时钟信号 MA 将并行应用于连接到系统的任何 BiSS 从机。锁存点可用于同步整个菊花链的过程数据采集。有关用点对点连接方式连接多个 BiSS 从机的更多信息，请参阅菊花链部分。

### 处理时间

从锁存点 (LAT) 到起始位之间的时间定义为处理时间 ( $t_{busy}$ )。由于帧的特性，BiSS 从机设备的处理时间至少为两个时钟周期。对于处理时间较长 (超过两个时钟周期) 的设备，起始位在数据头内的生成是延迟了的。此功能可使 BiSS 系统在 BiSS 周期开始时锁存传感器数据，并在传输前对其进行处理。图 17 展示了一个需要两个以上时钟周期来生成过程数据的 BiSS 从机的示例。

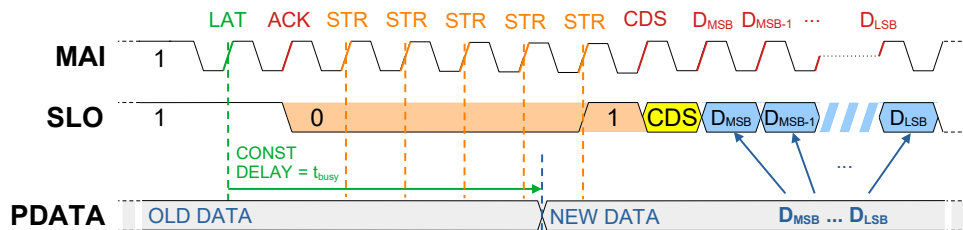


图 17: 处理时间较长的延迟 BiSS 帧

在点对点连接中，起始位的传输完全由 BiSS 从机控制。如果设备需要更多时间来生成过程数据，BiSS 从机在确认位 (ACK) 之后发送 "0"，而不是起始位 (STR)。直到过程数据准备好传输之前，起始位都会被抑制。关于主机必须支持的最大处理时间 ( $t_{busy}$ ) 请参考“特性”一章。

对于总线连接的系统，起始位延迟会由 BiSS 总机控制，更多信息请参考“总线连接”一章。

### 菊花链

即使在点对点连接中，也可以在 BiSS 从机设备中包含多个 BiSS 从机。时钟信号 MA 与每个 BiSS 从机并行连接。数据输出线 SLO 与后续 BiSS 从机的数据输入线 SLI 相连，因此最左侧 BiSS 从机的帧数据会在整个菊花链中移动。图 18 显示了在一个 BiSS 从机设备中菊花链连接的三个 BiSS 从机的示例，该设备以点对点连接方式连接到 BiSS 主机。

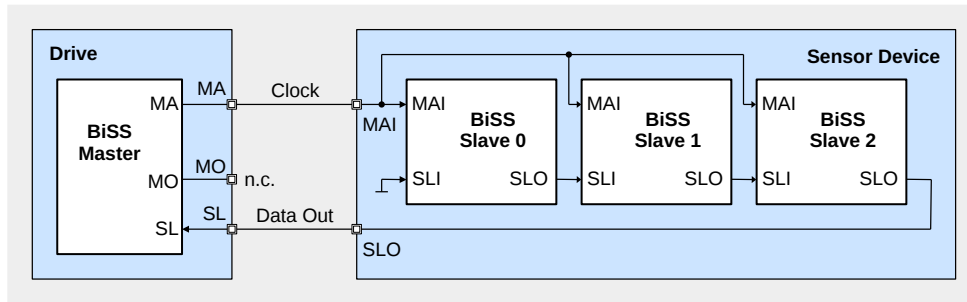


图 18: 点对点连接中 BiSS 从机内的菊花链

菊花链就像一个长移位寄存器，将每个 BiSS 从机的过程数据移回 BiSS 主机。由于最右边的 BiSS 从机 2 的数据输出信号 SLO 直接连接到 SL，因此 BiSS 主机首先收到其过程数据。在“控制通信”一章中会讲到识别方案的概念。由于识别方案的存在，连接菊花链

与 BiSS 主机的 BiSS 从机获得最高从机编号（此处为从机 2）。菊花链另一端的 BiSS 从机，其数据输入信号 SLI 接地，始终称为 BiSS 设备的从机 0。

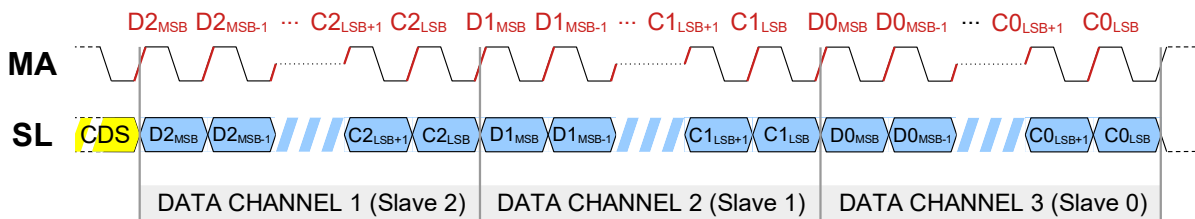


图 19: 菊花链式 BiSS 从机的数据通道

图 19 显示了菊花链式三个 BiSS 从机的数据通道。如上所述，三个数据通道依次传输，每个通道都有自己的 CRC 进行独立保护。如图 18 所示，BiSS 主机接收到的第一个数据（数据通道 1）属于从机 2。第二个和第三个数据通道（数据通道 2 和 3）分别属于从机 1 和从机 0。

**i** 过程数据通信的数据通道顺序与用于控制通信的从机 ID 的顺序相反。

造成这种相反顺序的原因是，识别方案的机制总是从第 0 号从机开始，但是 BiSS 主机首先接收到的数据通道是来自于直接连接到 SL 的 BiSS 从机。

由于时钟信号 MA 是并行连接的，因此每个 BiSS 从机都会在 LAT 上同时触发过程数据的生成。如果 BiSS 从机生成过程数据所需的处理时间不同，则由处理时间最长的 BiSS 从机决定起始位延迟，如“处理时间”一节所述。因此，在完成自己的数据生成后，每个 BiSS 从机都会将前一个从机的数据输出信号 SLO 移交给数据输入信号 SLI。实际上，只有 0 号从机生成起始位，并通过整个菊花链传回 BiSS 主机。

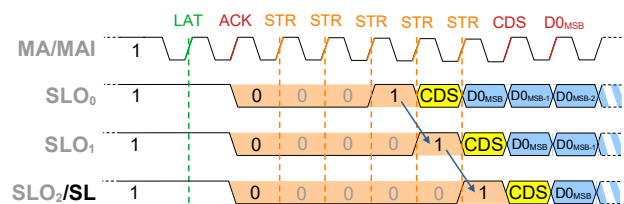


图 20: 菊花链中起始位延迟的 BiSS 从机

图 20 显示了前面所述的三个菊花链式 BiSS 从机的起始位延迟。确认位 (ACK) 强制将每个数据输出信号 SLO 置低 (0)。只有 0 号从机 (SLO<sub>0</sub>) 在准备就绪时主动在 SLO 上生成起始位。其他的 BiSS 从机 (SLO<sub>1</sub> 和 SLO<sub>2</sub>) 捕获数据输入信号 SLI，并在其过程数据生成完成后将其移交给相应的 SLO。2 号从机的数据输出信号通过 SL 连接至 BiSS 主机。由于这种捕捉机制会使每个从属设备的起始位延迟一个时钟，因此即使 BiSS 从机本身不需要额外的时间来生成过程数据，BiSS 主机也能始终检测到任意菊花链的处理时间。

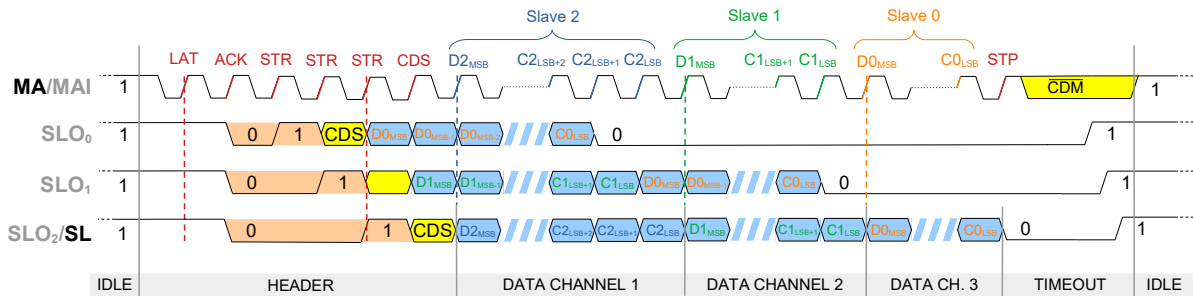


图 21: 通过菊花链波纹连接的整个 BiSS 帧

**i** 由于后面的 BiSS 从机在时钟输出其处理数据之前会等待起始位，因此处理时间最长的 BiSS 从站被置于 0 号从机。

图 21 显示了通过菊花链的整个 BiSS 帧。如上所述，虽然从机 0 (SLO<sub>0</sub>) 在生成处理数据时不会产生任何实际的起始位延迟，但 BiSS 主机会检测从机 2 的处理时间。从前者捕获 CDS 位后，菊花链中的每个 BiSS 从机都将其用于控制通信，然后再生成自己的 CDS 位。

有关控制通信的更多信息，请参阅“控制通信”一章。

在起始位之后输出自己的过程数据 (SLO<sub>2</sub> 上的数据通道 1, 蓝色) 时，从机 2 需要缓冲其前置过程数据 (SLO<sub>1</sub> 上的数据通道 2, 绿色。以及数据通道 3, 橙色)，这些数据与数据输入线 SLI (SLI<sub>2</sub>) 相连。另一方面，中间的从机 1 需要相应地缓冲从机 0 的帧数据。因此，菊花链中的每个 BiSS 从机都必须能够缓冲来自前一个从机的帧数据，缓冲区的大小必须达到其最大过程数据配置加上 CDS 的大小。这就是将处理时间最长的 BiSS 从机设置为从机 0 的原因。

当完成自身过程数据的输出后，每个 BiSS 从机会再次将从输入数据信号 SLI 接收到的帧数据转交，正如在每一帧开头的起始位机制中所描述的那样。因此，无需为整个菊花链中的单个 BiSS 从机配置帧长度。相反，只有从机 0 会主动生成用于超时指示的停止位。

当 BiSS 主机停止发送时钟信号 MA 时，任何 BiSS 从机的数据输出信号 SLO 都会自动置低 (0)。如“超时”一章所述，超时部分会由每个 BiSS 从机独立处理。不过，为了同步超时机制，每个 BiSS 从机只有其前者在 SLI 上指示检测到超时时才会进入空闲状态。否则，数据输出信号 SLO 将保持低电平，以便将整个菊花链的超时检测结果正确通知 BiSS 主机。

### 总线复位/初始化

上电后和报错后，BiSS 主机必须中断 BiSS 通信 40 μs 后才能启动新的 BiSS 帧，以确保 BiSS 超时已过和所有从机都已准备好进行数据传输。

在点对点连接中，BiSS 帧的起始位和停止位必须由 0 号从机生成。为了通知 0 号从站其激活的角色，它的数据输入线 SLI 需要连接到地，并在上电后对系统进行初始化。为此，如图 22 所示，BiSS 主机在 MA 处至少发送两个低脉冲。超时后，系统可进行 BiSS 通信，数据信号 SL 的空闲电平应为“1”。

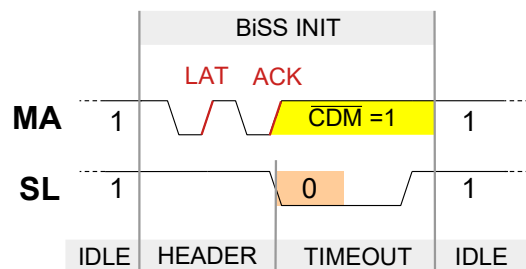


图 22: 典型初始化步骤序列示例

### 传播延迟

为了确保正确的数据通信，必须注意时钟输入信号 MAI 和数据输入信号 SLI 的传播延迟。如果 MAO 和 SLO 之间的延迟过大，菊花链中两个 BiSS 从站之间的同步数据传输可能会出现。由于没有对 BiSS 主机进行线路延迟补偿，考虑到与时钟线路 MAI 的并行连接，前一个主机的数据将在下一个下降沿被采样。如果数据输出信号 SLO 在下降沿之前没有稳定，则数据移位失败，从而导致 BiSS 帧处理过程中出现错误。



为防止 MAO 和 SLO 之间出现滞后, 可按照“总线连接”一章所述, 延迟 MAO 与 SLO 之间的同步。

在菊花链或总线拓扑结构中, 如果 BiSS 从机在 SLI 收到的数据无法 (完全) 缓冲, 也会传输空值。在这种情况下, 在传输空值之前可能会延迟起始位。

### 空值

在 BiSS 中, 空值 (所有数据位  $D_{MSB} \dots D_{LSB}$  等于 "0") 用于表示无效的传感器或执行器数据。当它由 BiSS 从机传输时, 表示其传感器数据自上一 BiSS 帧后未更新, 由 BiSS 主机传输时, 表示执行器数据未更新。

通常, 数据通道包含低电平有效的状态位 (如 BiSS Profile BP3 中定义的错误和警告位)。这样就可以输出等于零的有效传感器数据 (错误位为 "1")。如果传感器数据无效, 则所有位置和状态位均为 "0", 表示出错。

### 控制通信

与 SSI 协议不同的是, BiSS 能够进行双向数据传输, 即从 BiSS 主机到 BiSS 从机或反向传输。从 BiSS 主机到 BiSS 从机的数据传输可用于访问 BiSS 从机的内存资源。

例如, 用于传感器配置和校准。由于控制通信已嵌入 BiSS 帧, 因此这些数据传输不会中断过程数据传输。

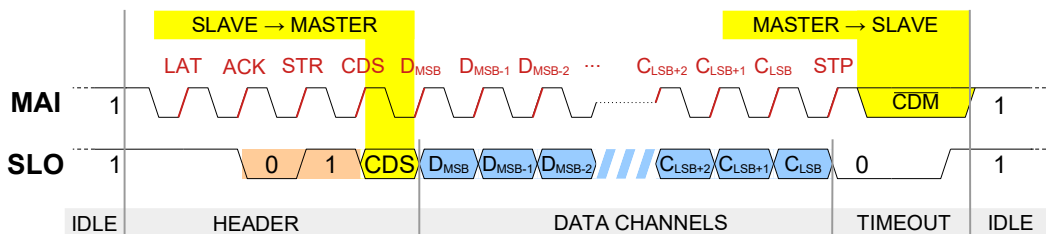


图 23: CDM 和 CDS 位

如“BiSS 帧”一章所述, 每个 BiSS 帧包括两个用于控制通信的位。图 23 显示了 CDM 位和 CDS 位, 这两个位除用于过程数据传输外, 还用于 BiSS 主机和 BiSS 从机之间的控制数据传输。

CDM 和 CDS 位在每个 BiSS 帧内传输, 并组成一个子数据帧, 其吞吐量低于过程数据传输。图 24 显示了几个传输多个 CDM 和 CDS 位用于控制通信的帧。

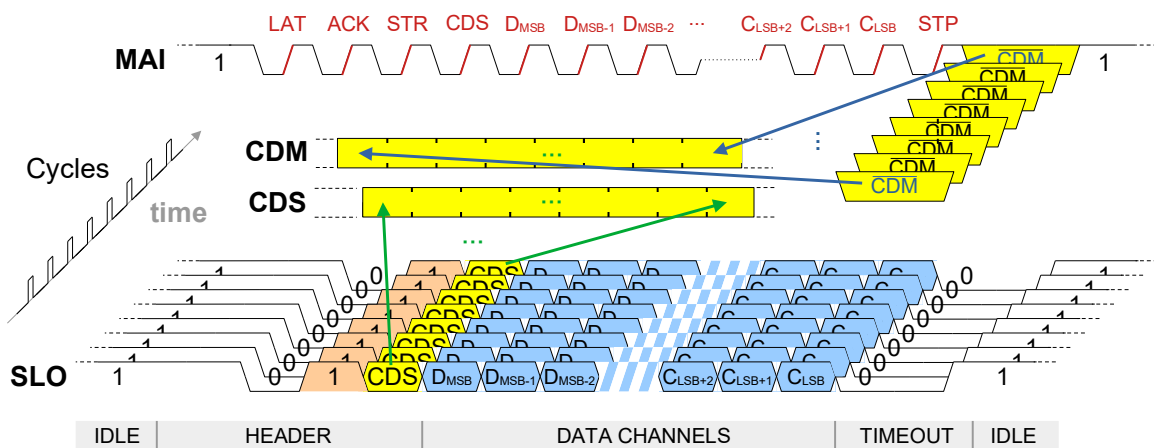


图 24: 多个 CDM 和 CDS 位组成控制数据帧

单 CDM 位 (传输前反向) 由 BiSS 主机在 BiSS 周期结束时生成, 并由 BiSS 从机在超时期间捕获 (如“BiSS 帧”章节所述)。

BiSS 从机处理传输的信息, 并通过在下一个 BiSS 周期开始时生成 CDS 位来响应 BiSS 主机。因此, BiSS 主机负责启动、定义类型和终止控制通信。

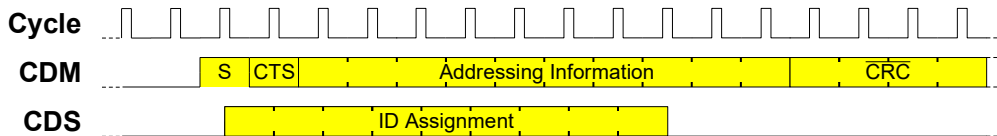


图 25: 控制数据帧头

图 25 显示了每个控制数据帧的数据头。只要没有控制数据帧正在处理, BiSS 主机就会发送 CDM = "0", BiSS 从机就会响应 CDS = "0"。BiSS 主机通过在 CDM 上发送控制数据启动位 (S) 触发新的控制通信, 从而开始在 BiSS 从机中处理新的控制数据帧。控制选择位 (CTS) 定义 BiSS 主机选择的控制通信类型。

- CTS=0: BiSS 命令 (第 17 页)
- CTS=1: 寄存器通信 (第 15 页)

在控制选择位之后, BiSS 主机传输 10 位寻址信息, 连接到菊花链的每个 BiSS 从机都要处理这些信息。每个控制数据帧的数据头都受到 4 位循环冗余校验 (多项式 0x13) 的保护。CRC 在传输前进行反向, 并考虑 CTS 和寻址位。计算 CRC 时不包括起始位。

为了提高 BiSS 系统的灵活性, 控制通信的寻址方案在 CDS 上每个新控制数据帧开始时动态执行。因此, 在接收到起始位后, 每个 BiSS 从机都会尝试根据其在菊花链中的位置获取下一个空闲的 BiSS 从机 ID。

控制通信最多可同时支持八个不同的 BiSS 从机 ID。BiSS 从机 ID 的分配与每个 BiSS 信号的起始位延迟和超时检测机制类似: 由于每个 BiSS 帧只有一个 CDS 位, 因此每个 BiSS 从机都会接收其前一站生成的 CDS 位。在 ID 分配期间, CDS 上的每个位都代表不同的 BiSS 从机 ID。如果 CDS 位被置 1, 则当前的 BiSS 从属设备 ID 已被占用, BiSS 从属设备将等待下一个周期, 如果可能的话并占用下一个 BiSS 从属设备 ID。

每个已获得 BiSS 从机 ID 的 BiSS 从站都会将未置位的 CDS 位转移给其后继的 BiSS 从机。经过九个周期后, ID 分配完成, 菊花链中的前八个 BiSS 从机可通过其相应的 BiSS 从机 ID 寻址。

第九位比特向 BiSS 主机表明, 连接到系统的 BiSS 从机数量已超过无需进一步测量即可寻址的最大数量。有关拥有超过 8 个 BiSS 从机的系统的 BiSS 从机 ID 分配的更多信息, 请参阅 BiSS 命令一节。图 26 显示了第 11 页图 18 中示例系统的 BiSS 从机 ID 分配。

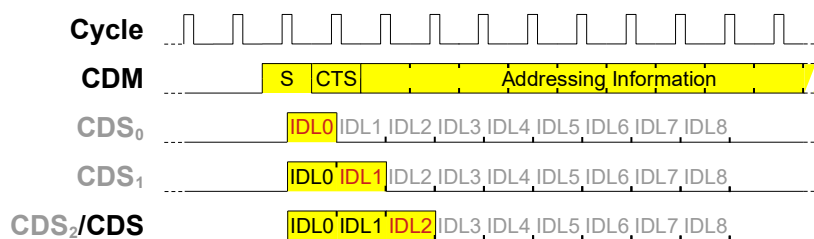


图 26: 3 个 BiSS 从机的 BiSS 从机 ID 分配

接收到起始位后, 从机 0 将锁定位 IDL0 设置为 "1", 并存储 BiSS 从机 ID = 0。从那时起, 从机 0 将应答任何与 BiSS 从机 ID = 0 的通信, 直到下一次 ID 分配。在同一 BiSS 周期内, 从机 1 和从机 2 将接收

IDL0=1, 并等待下一个 BiSS 周期以获得其 BiSS 从机 ID。在下一个 BiSS 周期中, 从机 0 不占用 CDS, 因此从站 1 将 IDL1 设置为 "1", 向从机 2 发出信号, 表明 BiSS 从机 ID = 1 已被占用。

在第三个 BiSS 周期中, IDL2 被从机 0 和从机 1 忽略, BiSS 从机 ID = 2 被分配给从机 2。

ID 分配完成后, 所有三个 BiSS 从机都可以通过相应的 BiSS 从机 ID 进行寻址。针对不同类型控制数据帧的进一步寻址机制将在后续章节中说明。

通过发送 14 个 BiSS 周期的 CDM = "0" 后, 控制数据帧可随时终止。每个连续 14 次接收 CDM = "0" 的 BiSS 从机必须重置为空闲状态, 并准备好处理新的控制数据帧。

重启后, 如果 BiSS 系统配置不正确, 可以通过电子数据表从连接的 BiSS 从属设备读取必要的通道配置。

为了确保控制通信 CDM 的正确采样, BiSS 主机应能够保持反相 CDM 位的电平, 直到下一个 BiSS 周期 (从机的超时需要过期), 而不是将其释放到空闲状态。图 27 显示了在 MA 处 "保持" CDM 电平的情况。如果 CDM 电平保持被激活且 CDM = "1", 则 MA 的上升沿将延迟至下一个 BiSS 帧开始。

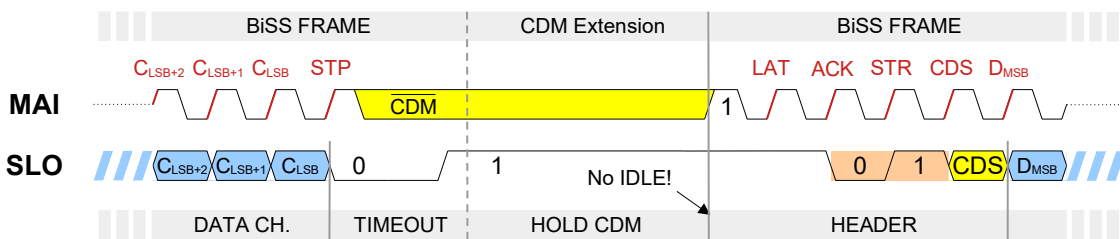


图 27: CDM 电平保持不变的连续两个 BiSS 帧

### 寄存器通信

如前所述, BiSS 协议支持两种类型的控制数据帧。寄存器通信用于对连接到 BiSS 系统的特定 BiSS 从机的内存映射进行字节寻址数据访问。这种数据访问用于读写 BiSS 从机的内存映射资源, 如设备配置校准数据和 EDS 参数。

### 寄存器读取访问

寄存器通信可用于单字节数据访问, 如果设备支持, 还可用于整个地址范围的自动递增数据访问。数据访问由 BiSS 协议确认, 并受多个寄存器保护级别的保护。

**i** 在不中断过程数据传输的情况下, 可访问 BiSS 设备进行额外的寻址数据交换。

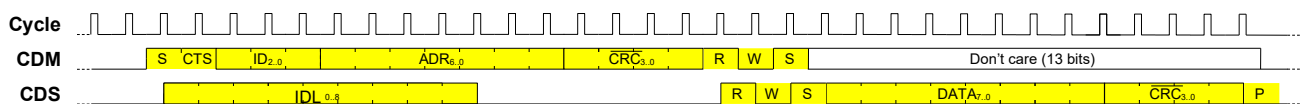


图 28: 单字节读取访问的寄存器通信

图 28 显示了一个控制数据帧, 用于从指定地址处具有相应 BiSS 从机 ID 的从机读取单字节。起始位 (S) 之后的 CTS 位 = 1 通知所有 BiSS 从机寄存器通信控制数据帧的开始。二进制编码的 3 位地址 ID 指定了帧的接收方。只有分配给由 ID 标识的 BiSS 从机 ID 的 BiSS 从机才能完全处理控制数据帧。其他 BiSS 从机在处理完帧数据头后会忽略控制数据帧的其余部分。

7 位地址 (ADR) 指定了数据请求所选 BiSS 从机内的资源。因此, BiSS 协议通过单个控制数据帧支持多达  $2^7 = 128$  字节的直接寻址访问。不过, 可以通过引入可在 BiSS 从机内切换的内存库来扩展可用地址空间。

有关内存库的更多信息, 请参阅“BiSS 内存图”一章。

在对数据头中的 4 位循环冗余校验进行验证后, 寄存器通信控制数据帧将继续读取 (R) 和写入 (W) 位。这些位由 BiSS 主机生成, 用于通知被寻址的 BiSS 从机有关请求的目的。读访问由 R=1 和 W=0 启动。

**i** 如果检测到 CRC 错误, BiSS 从机将视作未被寻址, 并忽略残余控制数据帧。

如图 28 所示, 被寻址的 BiSS 从机直接在 CDS 上重复 R 和 W 位。该机制用于确认 BiSS 主站的当前访问。只有当 R 和 W 位在 CDS 上重复相同时, 被寻址的 BiSS 从机才会接受数据访问。

如果所选地址不可用或不允许操作, BiSS 从机将 W 位 (1) 倒置, 并中止当前控制数据帧的进一步处理。BiSS 主机在检测到 W 位反向时, 也会中止当前控制数据帧。如果数据访问被 BiSS 从机接受, 则控制数据帧继续进行实际数据传输。

数据传输从 BiSS 主机生成的另一个起始位 (S) 开始, 并在 CDS 上再次重复。如果由于内部处理原因, BiSS 从机尚未准备好发送重新请求的数据, 则可以延迟重复起始位 (S) 以通知 BiSS 主机所需的处理时间。启动位延迟示例如图 29 所示。寄存器通信的最长处理时间在“特性”一章中规定。如果 BiSS 从机超过最大处理时间, BiSS 主机将中止控制数据帧。

**i** 一旦 BiSS 从机重复起始位 (S), 随后的 13 个 CDM 位与当前寄存器访问无关。不过, 建议在读取结束时发送 CDM = “0”, 重置第 14 个 CDM = “0”位的控制通信。

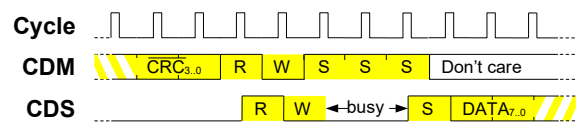


图 29: 单字节读取访问的起始位延迟

对于不支持写位反转的而导致上述无效读取访问的设备, 也可以使用起始位延迟来通知 BiSS 主机寄存器通信失败。但是, 为了正确检测无效访问, BiSS 主机必须等待寄存器通信所规定的最长处理时间。

从 BiSS 从机接收到重复的起始位 (S) 后, BiSS 主机将在 CDS 上收到请求的数据字节, 然后再进行 4 位 CRC 校验 (多项式 0x13), 以防止传输线路上的干扰。CRC 在传输前被反转并且只考虑数据字节。计算 CRC 时不包括起始位。单字节读取的控制数据帧由 BiSS 从机生成的停止位 P=0 结束。

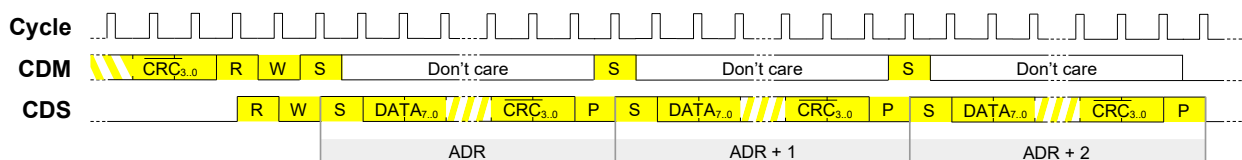


图 30: 用于顺序读取访问的寄存器通信

BiSS 主机收到停止位 (P) 后, 如果所连接的设备支持, 可通过 CDM 发送另一个起始位 (S), 直接开始另一次数据传输。这样, 就可以传输多个数据字节, 而无需重新启动整个帧, 从而提高吞吐量。在这种情况下, 被寻址的 BiSS 从机会自动递增接收到的地址, 并响应下一个数据字节。图 30 显示了读取 3 个连续数据字节的控制数据帧示例。

每次数据传输结束时的停止位 (P) 的使用与写入位 (W) 具有相同的确认机制。如果所选操作接受以下地址, 则停止位 P=0, 否则, BiSS 从机在 CDS 上发送 P=1, 通知 BiSS 主机终止该帧。

例如, 图 30 中的第四字节不能用于读取, 因此用 P=1 确认。

由于顺序寄存器访问时 R 位和 W 位不能重复, 因此在不启动新的控制数据帧的情况下, 无法改变当前控制通信的访问模式。顺序的寄存器访问的每个字节也支持单字节访问时的起始位延迟。顺序的寄存器访问的最大字节数为 64。



### 寄存器写入访问

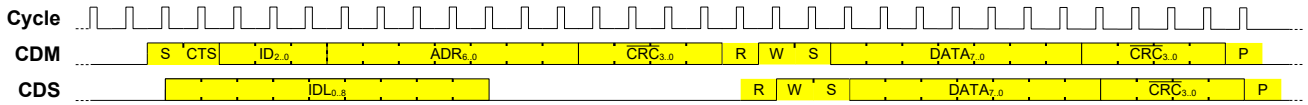


图 31: 单字节写入访问的寄存器通信

如图 31 所示, 与用于寄存器读取访问的控制数据帧不同, 用于启动寄存器写入访问的读写位分别为  $R=0$  和  $W=1$ 。帧头完全相同。

与之前一样, BiSS 从机通过重复  $R$  和  $W$  位确认所请求资源的可用性, 如果传输的地址未被接受, 则将  $W$  位反相 (0)。读取和写入同一地址的权限可以不同。有关 BiSS 支持的寄存器保护级别的详细信息, 请参阅“BiSS 内存图”一章。

收到起始位 (S) 后, BiSS 主机将数据字节写入所选地址。第一个数据字节前面的起始位不得延迟。如果需要处理时间, 可延迟顺序写入访问中的后续起始位。届时 BiSS 主机重复起始位, 直到 BiSS 从机确认为止。

为了进行确认, BiSS 从机重复接收到的数据信息和 4 位 CRC, 接收方必须对其进行验证。

**i** 为确保写入操作成功, 建议通过回读相应地址的数据来验证写入访问。

与寄存器读取访问一样, 传输以停止位  $P$  结束, 该位用于连续寄存器写入访问的访问确认。

图 32 显示了向编址 BiSS 从机写入三个连续数据字节的示例。由于 BiSS 从机的处理时间, 第二和第三个数据字节的起始位 (S) 会延迟。与之前一样, 从传输的地址信息中计数的第 4 个地址不可用于写入访问 ( $P=1$ )。

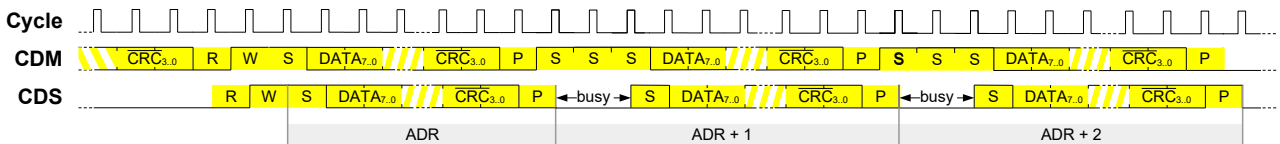


图 32: 带有起始位延迟的顺序写入寄存器通信

### BiSS 命令

为了同时触发多个 BiSS 从机的特定功能, 除了协议特定命令外 (本节稍后将解释), BiSS 还在协议中提供了一个命令机制。

甚至可以执行 BiSS 设备单独的特定从机命令。按照控制数据帧头的规范, BiSS 命令帧的定义如图 33 所示。

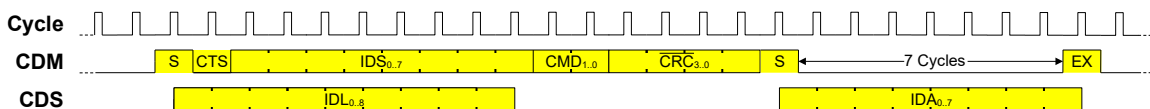


图 33: BiSS 命令帧

BiSS 命令帧由  $CTS=0$  启动位 (S) 启动。与寄存器通信帧不同, BiSS 命令帧不需要向接收者发送地址。相反, 帧头的寻址信息用于线性选择 8 位 ID (IDS) 和 2 位命令 (CMD) 本身。

从 BiSS 从机  $ID = 0$  开始, 每个 IDS 位代表一个自动分配的 BiSS 从机 ID。线性选择码使 BiSS 主机能够在控制数据帧内寻址多个 BiSS 从机, 并在必要时向多个 BiSS 从机发送一条 BiSS 命令。如果未设置 IDS 位, 则发送的命令应由连接到系统的所有 BiSS 从机执行。

即使是拥有 8 个以上 BiSS 从机的系统, 也可使用 BiSS 命令帧广播。因此, 当前未分配到 BiSS 从机 ID 的 BiSS 从机也需要处理即将到来的 BiSS 命令帧。有关 BiSS 命令帧广播的更多信息, 请参阅下文。

2 位的 CMD 定义了应在所选 BiSS 从机上执行的命令。因此, 可以向连接到系统的一个或多个 BiSS 从机触发 4 种不同的命令。有关不同 BiSS 命令的详细说明如下。如前所述, BiSS 命令帧头以 CDM 上传输的 4 位 CRC 校验结束。

传输完控制数据帧的数据头后, BiSS 主机将会继续传输 CDM 上的另一个起始位 (S) 以验证命令信息的接收情况。

因此, 所有 IDS 寻址的 BiSS 从机都会通过 ID 确认位 (IDA) 向 BiSS 主机确认接收并接受传输的命令。

与 IDL 和 IDS 位一样, IDA 位从 BiSS 从机 ID = 0 开始, 按升序代表 BiSS 从机。根据 BiSS 命令帧数据头的 IDS 位正在处理 BiSS 命令且已准备好执行的 BiSS 从机, 将相应的 IDA 位设置为 "1"。分配给有效 BiSS 从站 ID 的其他 BiSS 从站发送 IDA = "0"。这样, BiSS 主机就能在使用 CDM 上的执行位 (EX) 触发实际执行之前, 检查所请求的 BiSS 从机是否接受传输的 BiSS 命令。

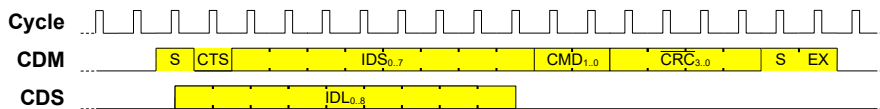


图 34: 广播 BiSS 命令帧

与针对特定 BiSS 从机的 BiSS 命令帧不同, 广播传输 (图 34) 以 IDS = "0000 0000" 启动。传输的 BiSS 命令将由连接到系统的所有 BiSS 从机执行, 即使是那些未分配有效 BiSS 从机 ID 的从机。因此, IDA 位被省略, BiSS 主机直接发送 CDM 上起始位 (S) 之后的 EX 位。

由于预定义 BiSS 命令的操作会因广播 BiSS 命令帧而改变, 因此一个 BiSS 命令帧总共可触发八种功能。表 1 列出了控制通信支持的所有 BiSS 命令。



BiSS 命令是可选的。建议执行 BiSS 命令 "00" 和 "01", 特别是在具有多机和总线功能的设备中。

BiSS 协议命令	
CMD	功能
<b>寻址</b>	
00	激活被寻址从站的过程数据通道
01	停用寻址从站的控制通信
10	激活总线耦合器的 "标准操作" (或用户定义的 "标准操作")
11	用户自定义
<b>广播 (对所有从机) IDS="0000 0000"</b>	
00	停用所有从属设备的过程数据通道激活
01	所有从属设备的控制通信
10	激活总线耦合器的 "反馈操作" (或保留)
11	保留

表 1: BiSS 命令

### CMD = "00" (过程数据通道)

考虑到一般启动程序, 可以使用广播命令 CMD = "00" 来停用连接到系统的所有 BiSS 从站的过程数据。如果执行该命令, 则可启用最短的 BiSS 帧, 用于双向控制通信。如短 BiSS 帧一节所述, 短 BiSS 帧适用于快速传输寄存器数据, 例如在初始化或传感器校准期间读取电子数据表数据。

针对特定 BiSS 从机的 CMD = "00" 命令仅针对分配给相应 BiSS 从机 ID 的 BiSS 从机重新激活过程数据。该命令可用于为所需应用初步设置 BiSS 帧。

### CMD = "01" (控制通信)

针对特定 BiSS 从机的 CMD = "01" 将停用系统所选 BiSS 从站的 ID 分配。该功能对于拥有 8 个以上 BiSS 从机的系统至关重要。例如, 为寄存器通信寻址菊花链中第九个 BiSS 从机的唯一方法是停用占据有效 BiSS 从机 ID 的前八个设备中的一个。

广播命令 CMD = "01" 可重新激活所有已连接的 BiSS 从机的寄存器通信。因此, 如果连接了九个 BiSS 从机, 则只能寻址前八个 BiSS 从机。启动后, 每个 BiSS 从机都会尝试在控制数据帧的 ID 分配期间占用一个有效的 BiSS 从机 ID (无需通过广播命令 CMD = "01" 激活)。

### CMD = "10" (总线耦合/用户定义)

广播命令 CMD = "10" 用于激活所有装有总线耦合器的 BiSS 从机的 "反馈操作" 模式。对于不带总线耦合器的 BiSS 从机, 广播命令 CMD = "10" 将被禁用。

CMD = "10" 可激活所选的带总线耦合器的 BiSS 从机的 "标准操作" 模式。如果未安装总线耦合器, 则该命令可用于特定设备功能, 例如传感器预设。



有关总线耦合器的详细信息, 请参见“应用提示”一章

### CMD = "11" (用户定义/保留)

针对特定 BiSS 从机的命令 CMD = "11" 也用于用户定义功能。通过寄存器通信更改活动命令, 可实现多个用户定义命令。广播命令 CMD = "11" 未被使用, 留待将来使用。

### 短 BiSS 帧

短 BiSS 帧是一个完整的帧, 如“BiSS 帧”一章所述, 但数据通道已停用。因此, BiSS 帧在传输停止位后结束。图 35 显示了通过广播命令 CMD = "00" 激活的短 BiSS 帧。

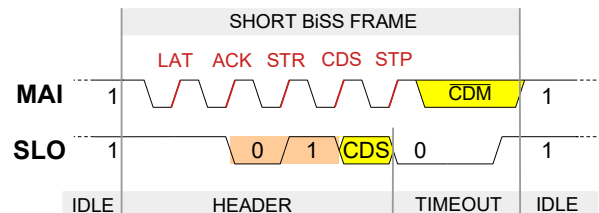


图 35: 短 BiSS 帧

由于短 BiSS 帧传输位 CDS 和 CDM 的存在, 因此可用于双向控制通信。短 BiSS 帧支持 BiSS 帧一章所述的自适应 BiSS 超时, 并可在 BiSS 主机进行适当的自动超时检测。短 BiSS 帧使 BiSS 周期时间最短, 从而实现最快的控制通信。

### 精简 BiSS 帧

与短 BiSS 帧相比, 精简 BiSS 帧不传输用于双向寄存器通信的 CDS 位。相反, 即使 BiSS 从机没有任何响应, 该帧也会在 MA 的确认边沿后结束。图 36 显示了上述精简 BiSS 帧。

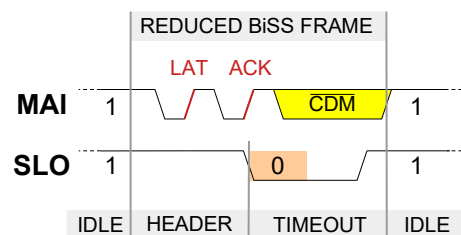


图 36: 精简 BiSS 帧

精简的 BiSS 帧仅由 BiSS 主机在确认位后停止切换 MA 时启动。由于不传输 CDS 位, 精简 BiSS 帧只能用于广播 BiSS 命令, 不需要 BiSS 从机做出任何响应。因此, 当 BiSS 主机未根据所连接的 BiSS 从机的正确参数进行正确配置时, 精简 BiSS 帧应该用于系统初始化。为了实现快速寄存器通信, 可以使用精简 BiSS 帧激活短 BiSS 帧, 例如广播命令 CMD = "00"。

### BiSS 内存图

如控制通信一章所述，寄存器通信可用于寻址每个 BiSS 从机的 128 字节内部数据资源。上面 64 个字节（地址 0x40 至 0x7F）称为固定地址，直接映射到 BiSS 从机最重要的数据资源。其中一些固定地址是为 BiSS 协议本身的必要信息预先定义的，其他地址则可自由用于设备特定信息。地址范围的其余部分（地址 0x00 至 0x3F）可用于其他数据资源，这些数据资源被组织成内存库。

因此，固定地址空间包括一个存储模块选择字节（BSEL）。必要时，可使用存储模块选择字节切换地址 0x00 至 0x3F 所代表的地址，以扩展可寻址的数据资源。因此，寄存器存储模块空间总共可寻址  $256 \times 64 = 16384$  个字节。例如，寄存器存储模块可以包含配置参数、电子数据表（EDS）或用户数据（如电机信息表）。因此，每个寄存器存储模块的访问级别由设备制造商根据其内容确定。表 2 提供了 BiSS 内存图的概览。

地址	名称	寄存器保护级	内容
<b>寄存器地址</b>			
0x00 ...0x3F	Register 1 ... Register 64	R/W, R, N/A	带有 64 个寄存器的寄存器存储模块（如配置数据、EDS、用户数据）。寄存器存储模块由 BSEL 选择。每个寄存器的访问级别由设备制造商确定。
<b>固定地址</b>			
0x40	BSEL	R/W, N/A	存储模块选择，如果执行的存储模块不止一个。
0x41	EDS_BANK	R, N/A	指向 EDS 内存库的指针
0x42 0x43	BP_ID	R	BiSS 类别 ID
0x44 ... 0x47	DEV_SN	R, N/A	设备序列号
0x48 ...0x77	Free Registers	R/W, R, N/A	与项目相关的自由寄存器（如状态信息、设备专用指令寄存器等）。每个寄存器的访问级别由设备制造商确定。
0x78 ... 0x7D	DEV_ID	R	设备 ID
0x7E ... 0x7F	MFR_ID	R	制造商 ID
<b>备注:</b> 1) 多于一个字节的参数以大端序方式保存，即最高值字节位于最低值地址。			

表 2: BiSS 内存映射表

### 寄存器保护级别

为了保护数据和配置参数，BiSS 协议提供了三级寄存器保护级别（RPL）。这些阶段如表 3 所示，可分配给 BiSS 地址空间中的任何地址，如上文所述。

寄存器保护级别	
RPL	描述
R/W	可以读取和写入相应的寄存器地址。
R	只能读取相应的寄存器地址。写入访问将被拒绝。
N/A	相应的寄存器地址拒绝读写访问。

表 3: 寄存器保护级别

BiSS 地址范围内的每个地址都可以完全访问、只读或完全保护。由于指示机制相同，因此无法区分完全保护和未执行两种情况。

受保护寄存器的访问指示参见控制通信一章。

### 存储模块选择

存储模块选择（BSEL）字节被映射到从机地址 0x40。BSEL 最多可选择 256 个存储模块，每个存储模块有 64 个数据字节，编址为 0x00 至 0x3F。为了切换当前映射到可访问地址空间的存储模块，需要使用寄存器通信帧进行写访问。BSEL 可以被读取以检查当前激活的存储模块。如果当前只有一个模块被选择，则拒绝对 BSEL 进行读写访问。

BSEL(7:0)		地址 0x40; 位 7:0	R/W
编码	描述		
0x00 ...0xFF	可通过地址 0x00...0x3F 访问所选存储模块的寄存器。		
备注	如果只有一个存储模块被执行，则拒绝对 BSEL 进行读写访问（RPL=N/A）。		

表 4: 存储模块选择

### EDS 存储模块

地址 0x41 是 **EDS\_BANK** 指针。该字节为只读字节, BiSS 主机可通过访问该字节获取 BiSS 设备电子数据表 (EDS) 的存储模块号。EDS 存储空间在 "电子数据表" 一节中进行了描述。每个 BiSS 设备都应安装 EDS。如果无法安装 EDS (例如由于内存空间不足), 则至少必须专配一个类别 ID。

EDS_BANK(7:0) 地址 0x41; 位 7:0		R
编码	描述	
0x00	未选择 EDS。	
0x01	包含电子数据表 (EDS) 的存储模块编号。	
...0xFF		
备注	在遗留产品中, 如果没有选择 EDS, 访问 EDS_BANK 可以被拒绝 (RPL=N/A)。	

表 5: EDS 存储模块

### 电子数据表

电子数据表 (EDS) 可用于根据所连接的 BiSS 从机对 BiSS 主机进行自动配置。EDS 包含建立 BiSS 通信所需的所有从机特性, 例如数据通道配置、最大时钟频率和最短 BiSS 周期时间。EDS 的定义可以参考 [www.biss-interface.com](http://www.biss-interface.com)。

### 用户存储模块 (可选)

用户存储模块保存有关 BiSS 设备所执行系统的特定信息。这些可选存储模块通常由在最终系统中实施 BiSS 设备的用户定义。例如, 在电机反馈应用中, 用户存储模块可用于存储电机信息表, 其中包含有关电机及其特定限制的详细信息。因此, 最终产品的制造商需要能够写入用户存储模块, 并通过将寄存器保护级别提高到只读来防止对其进行更改。



强烈建议使用 EDS 和类别 ID。只有在没有足够的非易失性内存来存储 EDS 的情况下, 才可以忽略 EDS。在这种情况下, 必须至少执行一个有效的类别 ID (≠ 0x0000)。

### BiSS 类别 ID

为了简化 BiSS C 设备的兼容性并支持 BiSS 产品的互操作性, 针对常见应用和经常需要的设备类型制定了配置文件定义。

BiSS 类别文件定义了兼容系统应支持的数据通道配置和保护机制。

BiSS 类别 ID (**BP\_ID**) 是一个 16 位大端序数字, 可从地址 0x42 和 0x43 读取。如果没有 BiSS 类别被支持, **BP\_ID** 等于 0x0000。由于配置文件描述的是单个数据通道, 因此每个 BiSS 从属设备都需要执行一个类别 ID。BiSS 类别仅由 iC-Haus GmbH 或 BiSS Association e.V. 定义。更多关于 **BP\_ID** 的详细信息请参见 [www.biss-interface.com](http://www.biss-interface.com)。

BP_ID(15:8) 地址 0x42; 位 7:0		R
BP_ID(7:0) 地址 0x43; 位 7:0		R
编码	描述	
0x00	BiSS 类别 ID 由 iC-Haus GmbH 或 BiSS Association e.V. 确定。	
...0xFF		
备注	如果不支持 BiSS 类别 ID, 则为 BP_ID=0x0000。	

表 6: BiSS 类别 ID

### 设备序列号

为了在现场识别每个 BiSS 设备, 可以在位置 0x44 到 0x47 之间分配一个 32 位序列号 (**DEV\_SN**)。这个序列号会采用大端序的方式储存。和参数 **MFR\_ID** 以及 **DEV\_ID** 一起, 例如, 序列号为错误跟踪和批次的识别提供了一个全球唯一的标签。0x00000000 和 0xFFFFFFFF 是预留值, 不能指定为有效序列号。如果 BiSS 设备不提供序列号, 则必须将其值设置为 0x00000000, 或将地址 0x44 至 0x47 设置为不可访问。

DEV_SN(31:24) 地址 0x44; 位 7:0		R
DEV_SN(23:16) 地址 0x45; 位 7:0		R
DEV_SN(15:8) 地址 0x46; 位 7:0		R
DEV_SN(7:0) 地址 0x47; 位 7:0		R
编码	描述	
0x00	设备序列号由设备制造商定义	
...0xFF		
备注	如果不支持设备序列号, 则为 DEV_SN=0x00000000。在遗留产品中, 如果无法执行设备序列号, 访问 DEV_SN 可能会被拒绝 (RPL=N/A)。DEV_SN=0xFFFFFFFF 被保留。	

表 7: 设备序列号

### 空闲的寄存器

从地址 0x48 到 0x77 的地址空间包含未使用的寄存器。由于这些寄存器没有为任何 BiSS 参数进行预定义, 因此可用于设备特定信息, 如状态信息或设备命令。寄存器的访问级别由设备制造商确定。如果设定了该参数, 那么一个寄存器必须独立于 **BSEL** 而可使用。

**i** 多于一个字节的参数以大端序方式保存, 即最高值字节位于最低值地址。

建议为具有不同 BiSS 接口配置 (如不同的数据长度或定时参数) 的设备分配不同的设备 ID。只有唯一可识别的 BiSS 设备才有可能执行任何自动系统初始化程序。

### 制造商 ID 和设备 ID

地址为 0x7E 和 0x7F 的 16 位制造商 ID 用于识别 BiSS 设备的制造商。它是唯一的, 由 BiSS Association e.V. 分配。制造商 ID 可以联系 [www.biss-interface.com](http://www.biss-interface.com) 来获取。

**i** EDS 和 BiSS 类别 ID 由 BiSS Association e.V. 标准化, 而使用制造商和设备 ID 的识别程序则需要制造商提供 XML 文件进行解码。

MFR_ID(15:8)	地址 0x7E; 位 7:0	R
MFR_ID(7:0)	地址 0x7F; 位 7:0	R
编码	描述	
0x00 ...0xFF	由 iC-Haus GmbH 或 BiSS Association e.V. 确定的唯一制造商 ID。	
备注	制造商 ID 必须被设定。只能使用 iC-Haus 或 BiSS 协会分配的制造商 ID。更多信息请联系 <a href="http://www.biss-interface.com">BiSS 协会</a> 。	

表 8: 制造商 ID

与制造商 ID 类似, 48 位设备 ID 可唯一标识 BiSS 设备的产品类型。BiSS 主机可读取地址 0x78 至 0x7D, 以检查设备是否按预期响应。

DEV_ID(47:40)	地址 0x78; 位 7:0	R
DEV_ID(39:32)	地址 0x79; 位 7:0	R
DEV_ID(31:24)	地址 0x7A; 位 7:0	R
DEV_ID(23:16)	地址 0x7B; 位 7:0	R
DEV_ID(15:8)	地址 0x7C; 位 7:0	R
DEV_ID(7:0)	地址 0x7D; 位 7:0	R
编码	描述	
0x00 ...0xFF	设备制造商定义的设备 ID。	
备注	如果设备 ID 不被支持, 则为 DEV_ID=0x000000000000。在遗留产品中, 如果没有设定设备 ID, 访问 DEV_ID 可能会被拒绝 (RPL=N/A)。	

表 9: 设备 ID

### 总线连接

与第 4 页 "点对点连接" 一章中描述的点对点连接不同, 使用总线连接的系统可以菊花链方式连接多个 BiSS 设备, 包括传感器和执行机构从机。用于数据输入 MO 的第三条 BiSS 信号线用于从 BiSS 主机向 BiSS 从机传输执行器数据和/或根据所连接 BiSS 从机的最大处理时间延迟起始位。图 37 显示了一个使用总线连接的简单 BiSS 系统示例, 该系统用于带单个 BiSS 从机的执行器设备。

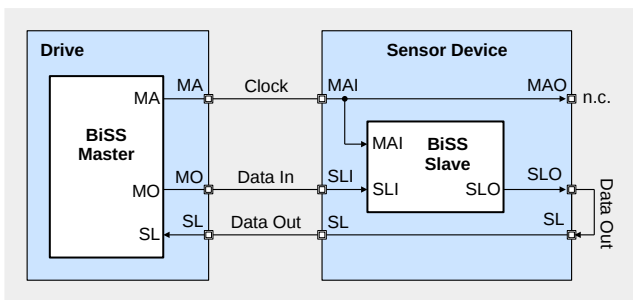


图 37: 带单个执行器从机的总线连接

一般来说, 点对点连接和总线连接的 BiSS 从机操作是相同的。只有从站 0 的 SLI 信号没有接地。这样, BiSS 主机本身就不再采用点对点连接的控制机制。因此, 起始位生成和超时检测都是在 MO 上启动的。

执行器设备要处理的数据被传送到 SLI。然后通过时钟传送, 供 BiSS 从机访问。

数据输出端 SLO 通过 SL 从外部反馈给 BiSS 主机。对于执行器设备, BiSS 从机通过 SLO 返回上一个 BiSS 周期的过程数据。

引脚 MAO 输出 BiSS 时钟, 这是连接多个设备所必需的, 如下所述。时钟转发是点对点连接一章中讨论的。图 38 显示了上述系统的相应 BiSS 帧。由于 SLI 在 BiSS 帧开始时未接地, 因此 BiSS 从机不会生成起始位。相反, BiSS 从机在输出新的进程数据之前, 会等待在确认位之后通过 MO 传送起始位。

由于 BiSS 主机的行为类似于从机 0, 因此必须传输一个有效的 CDS 位供 BiSS 从机进一步处理。因此, 下一位总是 CDS = "0", 如图 38 所示。CDS = "0" 对控制通信或自动 ID 分配 (如控制通信一章所述) 没有影响。

数据头的最后一位是 MO 停止位 "0"。停止位对于 BiSS 帧结束时的超时检测非常必要。由于每个 BiSS 从机在传输完自己的进程数据后会输出在 SLI 上采样的数据, 因此停止位可确保在超时 SLO 上的上升沿能被 BiSS 主机检测到。对于点对点连接, 通过将从机 0 的 SLI 连接到地, 自然会产生停止位。

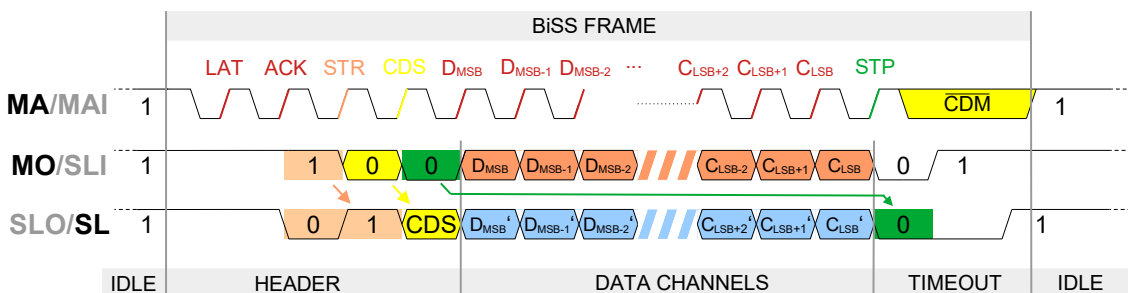


图 38: 单执行器总线系统的 BiSS 帧

总线连接中的数据通道完美地诠释了所有 BiSS 设备的移位寄存器机制。MO 上的执行器数据 (D0-C15) 被移入寄存器, 而在上一个 BiSS 周期中接收到的过程数据 (D0'-C15') 则被移出寄存器。

也就是说, 对于执行器从机, 过程数据是在 MAI 上的 BiSS 时钟期间进行交换的。执行器数据通道的配置等同于点对点连接中传感器数据通道的配置, 同时也可选 CRC 校验保护, 详见 "过程数据通信" 一章。

如前所述, BiSS 帧以 MO 上的停止位结束, 随后在传输 CDM 位时出现上升沿。总之, 总线连接与点对点连接相比, SLO 上的 BiSS 帧没有区别, 连接的 BiSS 从机的行为也没有区别。

与点对点连接不同, 总线连接可用于多个 BiSS 从机的菊花链连接, 甚至可分布在多个设备中。

每个 BiSS 从机既可以是传感器从机, 也可以是执行器从机, 这样就能在电机反馈应用中使用相同的协议和传输线路进行完整的控制信号处理。

图 39 显示了带有两个 BiSS 设备的总线示例。BiSS 设备 1 包含两个 BiSS 从机, BiSS 设备 2 包含一个 BiSS 从机。虽然单个 BiSS 从机的行为完全相同, 但在实施总线连接中包含多个设备的系统时, 需要考虑一些影响。

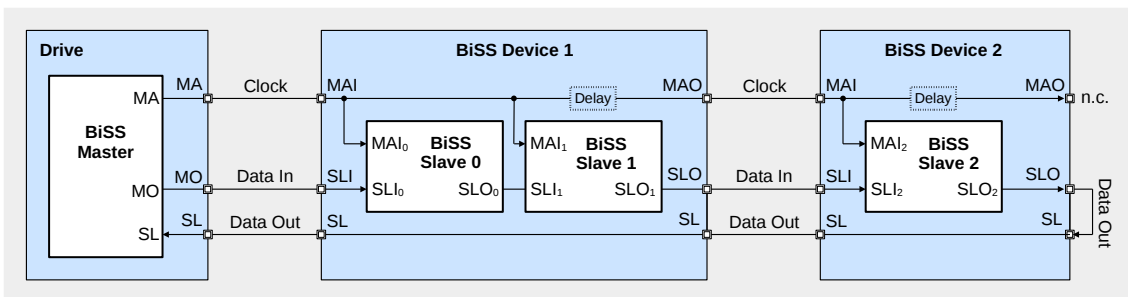


图 39: 总线连接中的菊花链

首先, 必须考虑菊花链式连接的 BiSS 从机之间的线路延迟。第 8 页 "线路延迟" 一节中所述的线路延迟补偿功能仅适用于将信号传输回 BiSS 主机。例如, 从 BiSS 设备 1 到 2 的信号不进行任何延迟补偿。由于 MA 是通过 BiSS 设备 1 传输的, 因此必须将 BiSS 时钟的延迟与 SLO 上数据信号的延迟相匹配。如果 BiSS 设备 2 上的时钟和数据信号不同步, 菊花链可能无法正常工作。MAO 和 SLO 之间的最大延迟时间在 "特性" 一章中定义。

不同设备内 BiSS 从机的 ID 分配如前所述。BiSS 从机 ID 的计算不考虑设备总数, 而只考虑设备内的 BiSS 从机模块。总之, 控制通信也适用于总线连接, 详见控制通信一章。图 40 显示了图 39 所示示例系统的相应 BiSS 帧。



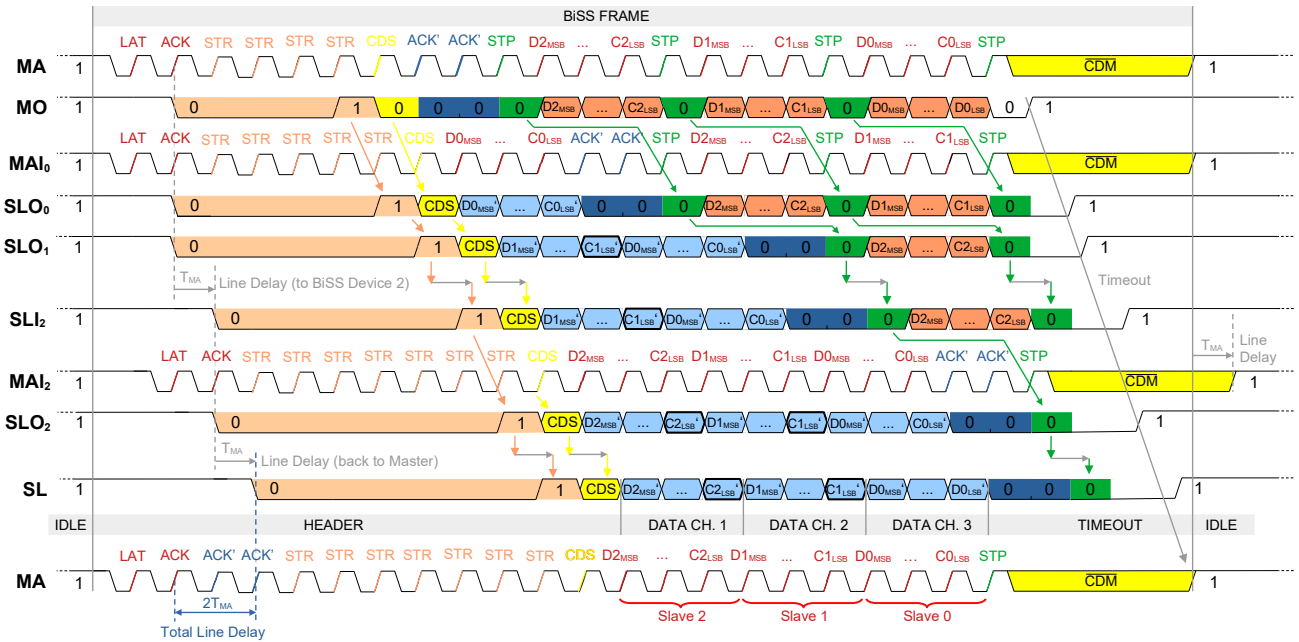


图 40: 总线连接中菊花链的 BiSS 帧与处理时间

与通常一样, BiSS 帧以 MA 开始。如前所述, 所有传感器过程数据都在 LAT 处采样。最佳做法是在 LAT 时也启用执行器数据, 以实现无抖动控制回路应用的确定性时序。

在确认位之后, 如上所述在 MO 上产生起始位。与点对点连接不同的是, 启动位不是由第 10 页 "处理时间" 一节中所述的从 0 产生。相反, BiSS 主机根据系统中速度最慢的 BiSS 从机的最大处理时间, 配置为延迟 MO 上的启动位。这样, 如图 40 所示, 每个 BiSS 从机都会在菊花链中发出启动位, 而不会再有任何延迟。

处理完 CDS 位后, 过程数据的交换方式与图 38 中单个数据通道的交换方式相同。MO 上的每个执行器数据通道以一个停止位结束, 该停止位需要时钟进入移位寄存器, 以便每个 BiSS 从机进行正确的超时检测。

每个 BiSS 从机每一帧结束时都会在 SLO<sub>x</sub> 上输出停止位。由于采用移位寄存器机制, 在 SLO<sub>2</sub> 向 BiSS 主机传输的执行器数据通道之间不输出停止位。

由于 BiSS 设备之间存在线路延迟, 因此与 BiSS 设备 1 的输出信号相比, BiSS 设备 2 的输入信号会出现延迟。如前所述, 为了正确处理接收到的帧数据, 必须匹配 MAI<sub>2</sub> 和 SLI<sub>2</sub> 信号的时间。因此, 可能有必要在 BiSS 设备 1 的 MAI 和 MAO 之间对时钟实施额外延迟。不过, 在任何情况下, 每个 BiSS 设备的时钟和数据信号都最好使用相同的总线驱动器。如 BiSS 帧一章, 线路延迟一节所述, SLO<sub>2</sub> 的总线路延迟会由 BiSS 主机自动补偿。

### 应用提示

### 初始化示例

以下序列提供了一个在 BiSS 主机中实施的初始化示例，用于与单个 BiSS 从机进行点对点连接。请注意，BiSS 设备的单独启动时间未考虑在内。

1. 设置时钟频率：  
 $250 \text{ kHz} \leq f_{\text{MA}} \leq 1,000 \text{ kHz}$   
(根据特性 No.2).
2. 设置周期时间： $t_{\text{Cycle}} = 250 \mu\text{s}$   
(根据特性 No.12).
3. 禁用 BiSS 主机配置中的所有数据通道
4. 启用 Hold-CDM (如图 27 所示)。这可确保在数据通道配置已知之前，控制通信正常进行。
5. 等待  $40 \mu\text{s}$  后再启动第一个 BiSS 帧。这将确保 BiSS 超时已过，所有 BiSS 从机都已准备好进行通信。
6. 发送 INIT 序列 (如图 22 所示)。为确保所有 BiSS 从机的数据输出信号 SLO 均为 "高" 电平，将发送初始化序列 (MA 处至少有两个低脉冲的帧)。
7. 等待  $40 \mu\text{s}$  以确保 BiSS 从机的超时已过。
8. 读取 BiSS 从机 (从机 ID = 0) 的电子数据表 (EDS)。如果没有 EDS，请继续执行步骤 9。
  - 读取 BiSS 从机的寄存器地址  $0x41$ 。该寄存器指向包含电子数据表(EDS)的寄存器存储模块。
  - 将地址  $0x41$  的内容写入  $0x40$ ，以选择 EDS 寄存器存储模块。
  - 读取地址为  $0x00 \dots 0x3F$  的 EDS。注意，内容取决于 EDS 版本。
  - 根据 EDS 配置 BiSS 主机 (如数据长度、CRC、时钟频率、周期时间)。
9. 如果没有 EDS，则读取 BiSS 从机 (从机 ID = 0) 的 BiSS 类别 ID。
  - 读取 BiSS 地址  $0x42$  和  $0x43$  的寄存器。这些寄存器包含 BiSS 类别 ID。
  - 根据 BiSS 配置文件配置 BiSS 主机 (如数据长度、CRC)。注意，时序参数 (如时钟频率、周期时间) 不是由 BiSS 配置文件定义的，需要手动配置。

### BiSS 周期时间计算

最短周期时间  $T_{\text{Cycle\_min}}$  由 BiSS 主机计算得出, 计算时应考虑以下因素

- 从机的特性
  - 处理时间 ( $t_{\text{busy}}$ , 向上舍入)
  - 额外的启动位延迟 ( $\text{busy\_s}$ , 时钟单位)
  - 每个从机的数据长度 ( $DLENx$ )
  - 每个从机的 CRC 长度 ( $CRCLLENx$ )
  - 超时 ( $t_{\text{TO}}$ , 向上舍入)
  - MAI 的最小时钟周期 ( $T_{\text{MA\_min}}$ )
- 传输线的特性 ( $t_{\text{LineDelay}}$ )
- 协议的特性 (附加数据头位)
  - 第一个时钟周期 ( $t_{\text{FirstClock}}$ )
  - CDS 位 ( $t_{\text{CDS}}$ )
  - 每个从机一个起始位的位移 ( $t_{\text{StartBitShift}}$ )
- 当前应用的 MA 时钟周期 ( $T_{\text{MA}}$ ).

从机的特性可从其电子数据表中读取。主机应用的时钟周期不得低于从机的最小时钟周期 ( $T_{\text{MA}} \geq T_{\text{MA\_min}}$ )。线延迟通常可由 BiSS 主机测量。



计算出的最短周期时间不得超限。此外, 从机可根据其执行情况定义绝对最短周期时间。计算出的最短周期时间和绝对最短周期时间都不得超限。

### N 个从机的一般计算方法:

$$T_{\text{Cycle\_min}} = \underbrace{T_{\text{MA}}}_{\text{First Clock}} + \underbrace{t_{\text{LineDelay}}}_{\text{Line Delay}} + \underbrace{t_{\text{busy\_max}}}_{\text{Processing Time}} + \underbrace{T_{\text{MA}} + \text{busy\_s\_max}}_{\text{Additional Start Bit Delay}} + \underbrace{T_{\text{MA}}}_{\text{CDS}} + \underbrace{T_{\text{MA}} * \sum_{x=1}^n [1 + DLENx + CRCLLENx]}_{\text{Data Channels (incl. Start Bit Shift)}} + \underbrace{t_{\text{TO}} + T_{\text{MA}}}_{\text{Timeout}}$$

$$= 4T_{\text{MA}} + t_{\text{LineDelay}} + t_{\text{busy\_max}} + \text{busy\_s\_max} + T_{\text{MA}} * \sum_{x=1}^n [1 + DLENx + CRCLLENx] + t_{\text{TO}}$$

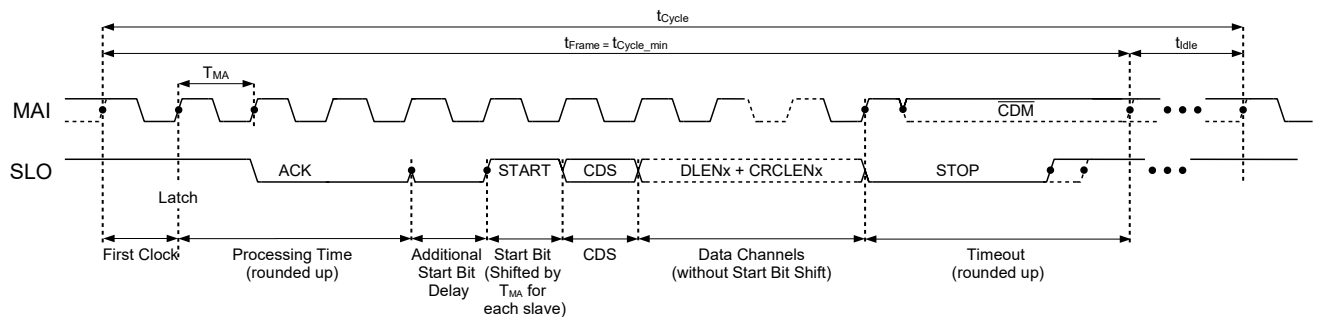


图 41: 用于计算最短周期时间的 BiSS 帧组成 (不包括线延迟)

### 单个从机的简化计算方法:

$$T_{\text{Cycle\_min}} = T_{\text{MA}} * (5 + DLEN + CRCLLEN) + t_{\text{LineDelay}} + t_{\text{busy\_max}} + \text{busy\_s\_max} + t_{\text{TO}}$$

### 总线耦合器

总线耦合器是 BiSS 从机的一项功能，可将设备的 SLO 线路直接反馈到 SL 线路。同时，后续 BiSS 设备的数据输出与 BiSS 主机断开。因此，如图 42 所示，BiSS 从机实现了两种工作模式的切换。

- **标准操作:** SLO<sub>x</sub> 连接至 SLI<sub>x+1</sub>, SL 输入连接至 SL 输出。
- **反馈操作:** SLO<sub>x</sub> 连接至 SL 从而使得下游设备旁路。

总线耦合器适用于硬件诊断，例如检测总线连接中的短路或开路。通过广播命令 **CMD = "10"**，所有总线耦合器均切换为“反馈操作”（无总线耦合器的 BiSS 从机忽略此命令）。然后，使用寻址命令 **CMD = "10"**，从第一个从机 (SLO<sub>0</sub>) 开始，将每一个总线耦合器切换为“标准操作”。通过检查每个实例的响应，可检测出故障设备并进行检修。

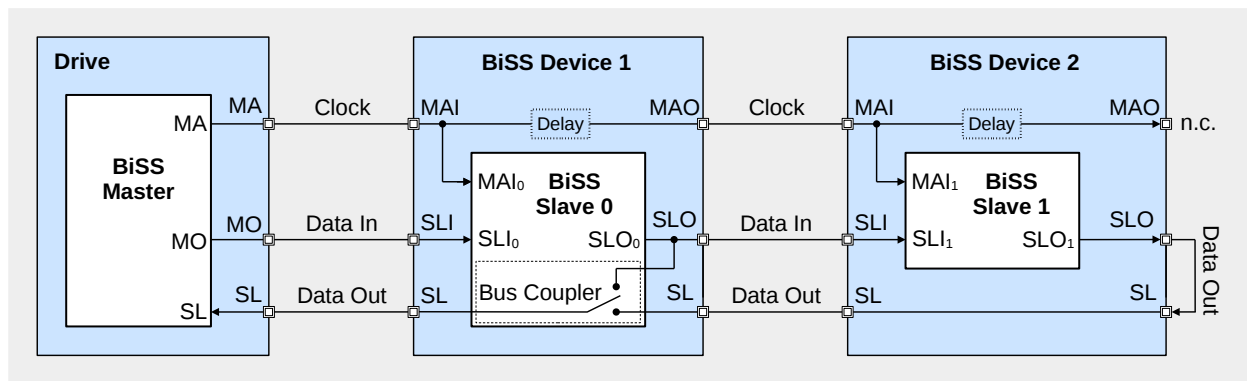


图 42: 设备 1 中的总线耦合器

### 特性

序号	标识	参数	条件	最小值	最大值	单位
01	$1/T_{MA}$	允许时钟频率		80	10000 <sup>1)</sup>	kHz
02	$1/T_{MA}^{2)}$	所需时钟频率	最大电缆长度100米	250	1000	kHz
03	$t_{MAI\_lo}$	时钟信号低电平持续时间	接收端 MAI = "0"	40		% $T_{MA\_min}^{1)}$
04	$t_{MAI\_hi}$	时钟信号高电平持续时间	接收端 MAI = "1"	40		% $T_{MA\_min}^{1)}$
05	$t_{TO}$	静态 BiSS 超时		12.5	40	$\mu s$
06	$t_{TOA}$	自适应 BiSS 超时	$T_{MA}$ 上具有自动 BiSS 超时适应功能的从机	$t_{TOA\_ref}$	$t_{TOA\_ref} + 3 \cdot T_{CLK}^{3)}$	
07	$t_{LineDelay}$	线路延迟 MA → SL	测量在接收端从 MA 处的第二个上升沿到 SL 处的第一个下降沿	0	40	$\mu s$
08	$t_{LineJitter}$	线路延迟抖动 MA → SL	接收端 $t_{LineDelay}$ 参数允许的偏差	-25	25	% $T_{MA}$
09	$t_{Lag\_SLO}$	SLO 滞后	对于总线配置, 以 MAO 上升沿为基准	-50	50	% $t_{MAlo}$ % $t_{MAhi}$
10a	$t_{busy}$	处理时间		$2T_{MA}$	$40\mu s$	
10b	$busy\_s$	额外的起始位延迟 (以时钟为单位)	$t_{busy} + busy\_s \leq 40\mu s$	0	8	$T_{MA}$
11	$t_{busy\_r}$	寄存器访问处理时间		0	20	ms
12	$t_{Cycle\_min}$	最小周期时间	支持所有从机		250	$\mu s$

#### 备注:

- 1) 最大的时钟频率取决于传输介质和各个设备。相应的最小时钟周期  $T_{MA\_min}$  通常被存储在 EDS 中。
- 2) 任何所需要的时钟频率都必须被 BiSS 从机所支持。BiSS 主机可将其用于初始操作, 以读取从机的 EDS。
- 3)  $1/T_{CLK}$  是 BiSS 从机最小的采样频率。  $t_{TOA\_ref}$  是 MAI 从第一个下降沿到第二个上升沿的测量值。

表10: 特性表

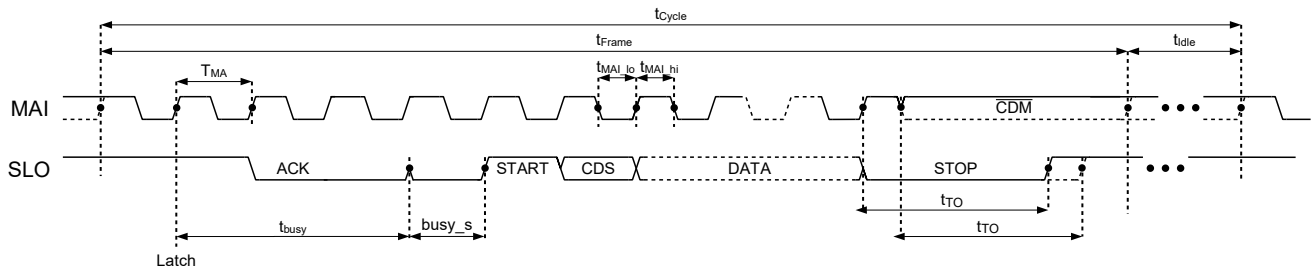


图 43 : 点对点配置中的 BiSS 从机时序

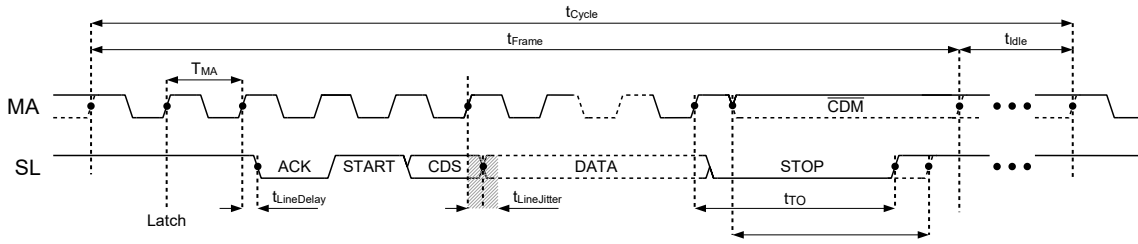


图 44: 点对点配置的 BiSS 主机时序

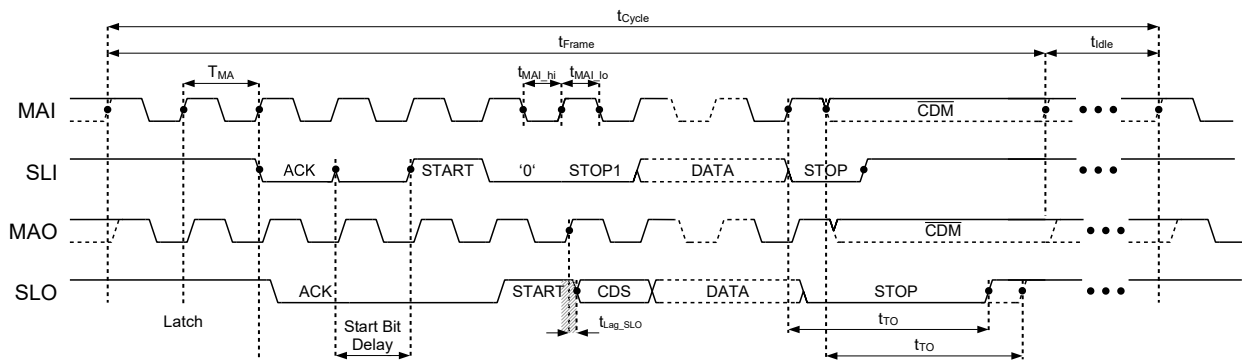


图 45: 总线配置中的 BiSS 从机时序

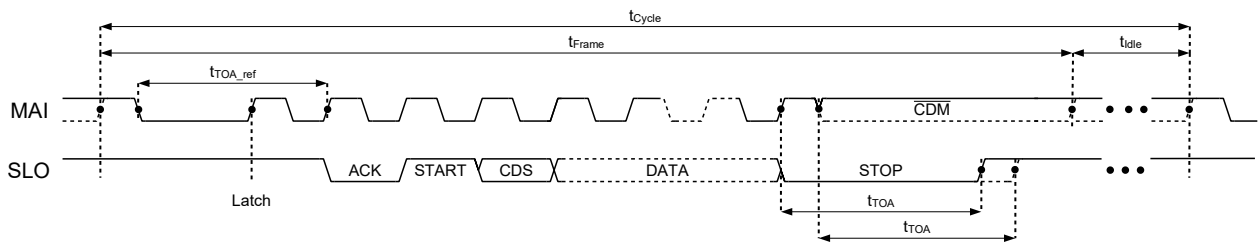


图 46: 自适应 BiSS 超时

### 缩略语表

<b>ACK</b>	BiSS 从机发送的确认位, 用于确认新的 BiSS 周期的开始	<b>MA</b>	主机时钟输出引脚 (连接至从机 MAI)
<b>ADR</b>	BiSS 主机发送的用于寄存器访问的寄存器地址	<b>MAI</b>	从机的时钟输入引脚 (连接至主机的 MA)
<b>BiSS</b>	双向/串行/同步	<b>MO</b>	主机的数据输出引脚 (连接至从机的 SLI)
<b>CDM</b>	BiSS 主机发送的控制数据位 ( $\overline{\text{CDM}}$ 是反向的控制数据位)	<b>P</b>	BiSS 从机发送的用于控制通信的停止位
<b>CDS</b>	BiSS 从机发送的控制数据位	<b>R</b>	BiSS 主机/从机发送的寄存器读取访问位
<b>CMD</b>	BiSS 主机发送的 BiSS 命令	<b>S</b>	BiSS 主机发送控制通信的起始位
<b>CRC</b>	循环冗余校验 ( $\overline{\text{CRC}}$ 表示反向 CRC)	<b>SL</b>	主机的数据输入引脚 (连接至从机的 SLO)
<b>CTS</b>	BiSS 主机发送的用于控制通信的控制选择位	<b>SLI</b>	从机的数据输入引脚 (连接至主机的 MO)
<b>EX</b>	BiSS 主机发送的 BiSS 命令的执行位	<b>SLO</b>	从机的数据输出引脚 (连接至主机的 SL)
<b>ID</b>	BiSS 主机发送的用于寄存器访问的从机 ID	<b>SSI</b>	同步串行接口
<b>IDA</b>	BiSS 从机发送的 BiSS 命令的 ID 确认位	<b>STP</b>	BiSS 从机发送的停止位, 指示 BiSS 帧结束
<b>IDL</b>	BiSS 从机发送的控制通信的 ID 锁定位	<b>STR</b>	BiSS 从机发送的起始位, 指示传感器数据已准备好传输
<b>IDS</b>	BiSS 主机发送的 BiSS 命令所对应的从机 ID	<b>TO</b>	BiSS 从机超时
<b>LAT</b>	表示 BiSS 传感器捕获当前数据的“锁存点”	<b>W</b>	BiSS 主/从机发送的用于寄存器访问的写访问位

### 修订历史

版本	修订日期*	章节	修改	页次
A1	2007-07-09		初版	全部页

版本	修订日期*	章节	修改	页次
C1 ...C5	2007 ...2016		有关修改请参阅相应的数据手册	

版本	修订日期*	章节	修改	页次
C6	2016-02-26		应用中增加了驱动控制器、旋转编码器、线性编码器和通信看门狗	1
		操作说明	图 3 已更新: T <sub>busy</sub> 始于 MA (锁存) 的第一个上升沿, 而不是 SLO 的第一个下降沿	4
		控制通信	更新了图 10 的参考资料	9
		控制通信	添加了精简 BiSS 帧的详细信息	10
		控制通信	新增了 BiSS 命令 0b00 和 0b01	11
		控制通信	新增 8 个 ID 以上的应用细节	11
		控制通信	新增寄存器保护细节	13
		术语和缩略语	缩略语表更新	20
		修订历史	新增章节	21
			少量文字更正	5, 8, 11, 13, 14, 15, 17

Rel.	修订日期*	章节	修改	页次
D1	2023-08-08	全部章节	全面更新, 改进了所有章节的说明和图	全部页
		BiSS 内存图	改进了描述, 以明确必要的 BiSS 寄存器和内容。	20
		应用提示	新增了应用提示	26
		特性	改进参数描述和条件。 No.01 更新为“允许的时钟频率”, 调整脚注 No.02 更新为“所需的时钟频率”, 调整了最小值 (80 kHz → 250 kHz) 和脚注 No.03 已更新, “时钟信号低电平持续时间”最小值为 40 ns No.04 已更新, “时钟信号高电平持续时间”最小值为 40 ns No.05 已更新符号和参数 No.06 删除了缩短的超时的特性 No.07 更新了符号, 简化了公式, 增加了脚注 No.08 改进了参数说明和条件 No.09 增加了 SLO 和 MAO 信号之间的滞后特性 No.10 更新为 No.10a/10b 特性 No.12 添加了“最短周期时间” 添加了时序图	29
		缩略语列表	新增了缩略语列表	31

Rel.	修订日期*	章节	修改	页次
D2	2023-10-26	过程数据通信	描述稍作修改并更新了图 21 (图 19 显示了两次)。	11
		控制通信	寄存器读取访问: 添加信息, 数据头中出现 CRC 错误时从属设备的行为。 更新了图 28、29 和 30: “S”后的 CDM 为 “Don't Care”。 在图 29 下方添加了信息。 更名为 “单周期数据” → “过程数据”。 用带有处理时间的图片取代图 32, 并更新了说明。	13
		特性	线路延迟抖动 (No.8) 更新为 25% (之前为 20%)。	29
		缩略语表	更正 CRC 描述中错别字。	31

\* 发布日期格式: 年-月-日