

USB 总线接口芯片 CH375

中文手册（二）：USB 基本传输命令

版本： 4

<http://wch.cn>

1、附加命令

代码	命令名称	输入数据	输出数据	命令用途
04H	SET_USB_SPEED	总线速度		设置 USB 总线速度
0AH	GET_DEV_RATE	数据 07H	数据速率	获取 USB 设备的数据速率类型
0BH	SET_RETRY	数据 25H		设置 USB 事务操作的重试次数
		重试次数		
0FH	DELAY_100US		延时状态	延时 100uS
13H	SET_USB_ADDR	地址值		设置 USB 地址
1CH	SET_ENDP6	工作方式	(等 3uS)	设置 USB 主机端点的接收器
1DH	SET_ENDP7	工作方式	(等 3uS)	设置 USB 主机端点的发送器
27H	RD_USB_DATA0		数据长度	从当前 USB 中断的端点缓冲区 读取数据块
			数据流	
41H	CLR_STALL	端点号	产生中断	控制传输：清除端点错误
45H	SET_ADDRESS	地址值	产生中断	控制传输：设置 USB 地址
46H	GET_DESCR	描述符类型	产生中断	控制传输：获取描述符
49H	SET_CONFIG	配置值	产生中断	控制传输：设置 USB 配置
4DH	AUTO_SETUP		产生中断	自动配置 USB 设备
4EH	ISSUE_TKN_X	同步标志	产生中断	发出同步令牌，执行事务
		事务属性		
4FH	ISSUE_TOKEN	事务属性	产生中断	发出令牌，执行事务
50H	DISK_BOC_CMD		产生中断	执行 BulkOnly 传输协议的命令
52H	DISK_RESET		产生中断	复位 USB 存储设备
5DH	DISK_MAX_LUN		产生中断	获取 USB 存储设备的最大单元号

如果输入数据是 USB 端点的收发器的工作方式，参考下表。

工作方式字节	名称	工作方式的位分析说明	
位 7~位 6	同步触发标志	如果位 7 为 1 则位 6 为新的同步触发标志：	
		00 或者 01=保持当前同步触发标志不变	
		10=同步触发标志置 0	11=同步触发标志置 1
位 5~位 4	(保留位)	(未定义，必须为 0)	
位 3~位 0	事务响应方式	必须是 0000	

1.1. 命令 SET_USB_SPEED

该命令设置 USB 总线速度（部分型号的芯片不支持该功能）。该命令需要输入 1 个数据，用于选择 USB 总线速度，00H 对应于 12Mbps 全速方式，02H 对应于 1.5Mbps 低速方式。CH375 的 USB 总线速度默认为 12Mbps 全速方式，并且在执行 SET_USB_MODE 命令设置 USB 工作模式后也会自动恢复到 12Mbps 全速方式。

1.2. 命令 GET_DEV_RATE

该命令获取当前连接的 USB 设备的数据速率类型。该命令需要输入 1 个数据 07H，输出为数据速

率类型，其位 4 为 1 则是 1.5Mbps 低速 USB 设备，否则是 12Mbps 全速 USB 设备。该命令仅在 USB 模式 5（已启用的 USB 主机方式，不产生 SOF 包）状态下有效。

1.3. 命令 SET_RETRY

该命令设置 USB 事务操作的重试次数。该命令需要输入两个数据，分别是数据 25H 和重试次数。

重试次数的位 7 和位 6 指定 CH375 收到 NAK 应答时的处理方式，位 7 为 1 并且位 6 为 0 则无限重试（可以用 ABORT_NAK 命令临时放弃当前的重试），位 7 为 1 并且位 6 为 1 则有限重试 200ms 到 2s 左右，位 7 为 0 则将 NAK 作为结果通知单片机或者作为错误处理。重试次数的位 5~位 0 指定当 USB 设备应答超时后 CH375 的重试次数，为 0 则超时后不重试。

芯片复位后或者重新设置 USB 模式后的默认重试次数是 85H，所以收到 NAK 应答后将无限重试，USB 设备应答超时后将重试 5 次。

1.4. 命令 DELAY_100US

该命令用于延时 100 μ s，只支持并口方式。在延时期间并口输出数据 0，延时结束后并口输出数据为非 0（通常是芯片版本号），单片机根据读出的数据判断延时是否结束。

1.5. 命令 SET_USB_ADDR

该命令设置 USB 设备地址。该命令需要输入 1 个数据，用于选择被操作的 USB 设备的地址。复位后或者 USB 设备连接或者断开后，USB 设备地址总是 00H，单片机通过默认地址 00H 与 USB 设备通讯，如果通过标准 USB 请求设置了 USB 设备的地址，那么也必须通过该命令设置相同的 USB 设备地址，以便 CH375 通过新地址与 USB 设备通讯。

1.6. 命令 SET_ENDP6

该命令设置 USB 主机端点或者端点 2 的接收器。该命令需要输入 1 个数据，指定新的工作方式。例如，如果执行 IN 事务并希望收到 DATA0 而放弃 DATA1，那么必须通过该命令设置主机端点的接收器的同步触发标志为 0，相应的工作方式字节是 80H。通常情况下，该命令在 3 μ s 时间之内完成。

1.7. 命令 SET_ENDP7

该命令设置 USB 主机端点或者端点 2 的发送器。该命令需要输入 1 个数据，指定新的工作方式。例如，如果执行 SETUP 或者 OUT 事务并希望发送 DATA0，那么必须通过该命令设置主机端点的发送器的同步触发标志为 0，相应的工作方式字节是 80H。如果希望发送 DATA1，则工作方式字节是 C0H。通常情况下，该命令在 3 μ s 时间之内完成。

1.8. 命令 RD_USB_DATA0

该命令从当前 USB 中断的端点缓冲区中读取数据块。在 USB 主机方式下，该命令与 RD_USB_DATA 命令的功能完全相同，唯一区别是该命令的效率稍高。

1.9. 命令 CLR_STALL

该命令是清除端点错误的控制传输命令。该命令需要输入 1 个数据，指定将被清除错误的 USB 设备的端点地址，对于 OUT 端点，有效地址是 01H~0FH，对于 IN 端点，有效地址是 81H~8FH。该命令用于简化标准 USB 请求 CLEAR_FEATURE，CH375 在命令执行完成后向单片机请求中断，单片机可以读取中断状态作为该命令的操作状态。如果操作状态是 USB_INT_SUCCESS，则说明命令执行成功，否则说明命令执行失败。

1.10. 命令 SET_ADDRESS

该命令是设置 USB 地址的控制传输命令。该命令需要输入 1 个数据，指定新的 USB 设备地址，有效地址是 00H~7FH。该命令用于简化标准 USB 请求 SET_ADDRESS，CH375 在命令执行完成后向单片机请求中断，单片机可以读取中断状态作为该命令的操作状态。如果操作状态是 USB_INT_SUCCESS，则说明命令执行成功，否则说明命令执行失败。

1.11. 命令 GET_DESCR

该命令是获取描述符的控制传输命令。该命令需要输入 1 个数据，指定将要获取的描述符的类型，有效类型是 1 或者 2，分别对应于 DEVICE 设备描述符和 CONFIGURATION 配置描述符，其中，配置描述符还包括接口描述符和端点描述符。该命令用于简化标准 USB 请求 GET_DESCRIPTOR，CH375 在命令执行完成后向单片机请求中断，单片机可以读取中断状态作为该命令的操作状态。如果操作状态是 USB_INT_SUCCESS，则说明命令执行成功，否则说明命令执行失败。由于 CH375 的控制传输缓冲区只有 64 个字节，所以当描述符的长度超过 64 字节时，CH375 将返回操作状态 USB_INT_BUF_OVER，对于该 USB 设备，单片机可以通过 ISSUE_TOKEN 或者 ISSUE_TKN_X 命令自行处理控制传输。

1.12. 命令 SET_CONFIG

该命令是设置 USB 配置的控制传输命令。该命令需要输入 1 个数据，指定新的 USB 配置值，配置值为 0 则取消配置，否则应该取自该 USB 设备的配置描述符中。该命令用于简化标准 USB 请求 SET_CONFIGURATION，CH375 在命令执行完成后向单片机请求中断，单片机可以读取中断状态作为该命令的操作状态。如果操作状态是 USB_INT_SUCCESS，则说明命令执行成功，否则说明命令执行失败。

1.13. 命令 AUTO_SETUP

该命令用于自动配置 USB 设备。该命令用于简化普通 USB 设备的初始化步骤，相当于 GET_DESCR、SET_ADDRESS、SET_CONFIGURATION 等多个命令序列。CH375 在命令执行完成后向单片机请求中断，单片机可以读取中断状态作为该命令的操作状态。如果操作状态是 USB_INT_SUCCESS，则说明命令执行成功，否则说明命令执行失败。

1.14. 命令 ISSUE_TKN_X

该命令使 CH375 发出同步令牌，执行事务。该命令需要输入两个数据，分别是同步标志和事务属性。同步标志的位 7 为主机端点的接收器的同步触发标志，位 6 为主机端点的发送器的同步触发标志，位 5~位 0 必须为 0。事务属性的低 4 位指定事务的令牌 PID，高 4 位指定 USB 设备的目的端点号。CH375 在命令执行完成后向单片机请求中断，单片机可以读取中断状态作为该命令的操作状态。如果操作状态是 USB_INT_SUCCESS，则说明命令执行成功，否则说明命令执行失败。该命令与 ISSUE_TOKEN 命令的唯一区别是该命令在执行事务前总是先设置同步触发标志（相当于再加上 SET_ENDP?命令）。

1.15. 命令 ISSUE_TOKEN

该命令使 CH375 发出令牌，执行事务。该命令需要输入 1 个数据，作为事务属性。事务属性的低 4 位指定事务的令牌 PID，高 4 位指定 USB 设备的目的端点号。CH375 在命令执行完成后向单片机请求中断，单片机可以读取中断状态作为该命令的操作状态。如果操作状态是 USB_INT_SUCCESS，则说明命令执行成功，否则说明命令执行失败，单片机可以根据操作状态进一步分析失败原因。

对于发送数据的 SETUP 事务和 OUT 事务，应该先通过 WR_USB_DATA7 命令写入准备发送的数据，然后再通过 ISSUE_TOKEN 命令执行事务；对于接收数据的 IN 事务，应该先通过 ISSUE_TOKEN 命令执行事务，当事务执行成功后，再通过 RD_USB_DATA 命令读出已经接收的数据。

例如，事务属性字节为 09H 时，则 CH375 从 USB 设备的默认端点 0 接收数据；事务属性字节为 21H 时，则 CH375 向 USB 设备的端点 2 发送数据；事务属性字节为 29H 时，则 CH375 从 USB 设备的端点 2 接收数据，该端点的地址是 82H。

下面是 CH375 支持的 USB 令牌 PID。

PID 字节	名称	说明
0DH	DEF_USB_PID_SETUP	发起控制传输，发送建立数据
01H	DEF_USB_PID_OUT	执行 OUT 事务，发送数据
09H	DEF_USB_PID_IN	执行 IN 事务，接收数据

1.16. 命令 DISK_BOC_CMD

该命令对 USB 存储设备执行 BulkOnly 传输协议的命令。在执行该命令之前，单片机必须先通过 WR_USB_DATA7 命令向 CH375 写入相应的 CBW 包，CH375 在命令执行完成后向单片机请求中断，单片机可以读取中断状态作为该命令的操作状态。如果操作状态是 USB_INT_SUCCESS 则说明命令执行成功，对于有返回数据的操作，可以由 RD_USB_DATA 命令获取返回数据。

1.17. 命令 DISK_RESET

该命令通过控制传输复位 USB 存储设备。CH375 在命令执行完成后向单片机请求中断，单片机可以读取中断状态作为该命令的操作状态。如果操作状态是 USB_INT_SUCCESS 则说明命令执行成功。当 USB 存储设备发生错误时，CH375 会分析错误原因并根据需要自动选择是否复位 USB 设备。

完整的复位过程包括：通过该命令复位 USB 存储设备，通过 CLR_STALL 命令复位 Bulk-IN 端点，通过 CLR_STALL 命令复位 Bulk-OUT 端点。

1.18. 命令 DISK_MAX_LUN

该命令通过控制传输获取 USB 存储设备的最大逻辑单元号。CH375 在命令执行完成后向单片机请求中断，单片机可以读取中断状态作为该命令的操作状态。如果操作状态是 USB_INT_SUCCESS，那么可以由 RD_USB_DATA 命令获取数据，数据通常是 1 个字节。

2、外部固件

2.1. 概述

ISSUE_TOKEN 命令或者 ISSUE_TKN_X 命令用于执行基本的 USB 传输事务，是 USB 主机方式下固件编程中最基本的操作。

在此基础上，外部单片机可以按照 USB 协议的要求，自行处理 CH375 尚未直接提供简化命令的控制传输。再进一步，单片机可以按照 USB 协议的要求，自行处理各种特定类设备的 USB 协议，实现对 USB 设备的控制和数据交换。CH375 内置了 Mass-Storage 海量存储设备的 Bulk-Only 传输协议，对于使用 CBI 传输协议的 USB 存储设备，仍然需要外部单片机基于 ISSUE_TOKEN 命令或者 ISSUE_TKN_X 命令和控制传输命令自行处理。

2.2. 外部固件参考流程

在 CH375 评估板资料中提供了外部固件的参考程序，下述流程是外部单片机通过控制传输执行标准 USB 请求 GET_STATUS，获取 USB 设备的状态，供外部单片机设计固件程序时参考。

(1) 控制传输的建立阶段

- ① 发出 WR_USB_DATA7 命令将 8 个字节的请求数据写入输出缓冲区，请求数据依次是 80H、00H、00H、00H、00H、02H、00H，长度为 8。

- ② 发出 `ISSUE_TKN_X` 命令执行事务，同步标志为 `00H`，事务属性字节是 `0DH`，向默认端点 `0` 发出 `SETUP` 令牌及发送 `DATA0`。如果使用 `ISSUE_TOKEN` 命令执行事务，那么必须先用工作方式字节为 `80H` 的 `SET_ENDP7` 命令设置主机端点的发送器的同步触发标志为 `0`。
 - ③ 单片机等待事务完成中断或者等待中断通知。
 - ④ 事务完成后，CH375 将 `INT#` 引脚设置为低电平，向单片机请求中断；
 - ⑤ 单片机进入中断服务程序，或者在主程序中收到中断的通知后退出等待。
 - ⑥ 发出 `GET_STATUS` 命令获取中断状态。
 - ⑦ CH375 在 `GET_STATUS` 命令完成后将 `INT#` 引脚恢复为高电平，取消中断请求。
 - ⑧ 单片机分析获得的中断状态，如果不是 `USB_INT_SUCCESS` 则操作失败，进行异常处理；如果是 `USB_INT_SUCCESS` 则事务执行成功，建立阶段完成。
- (2) 控制传输的数据阶段
- ① 发出 `ISSUE_TKN_X` 命令执行事务，同步标志为 `80H`，事务属性字节是 `09H`，向默认端点 `0` 发出 `IN` 令牌及接收 `DATA1`。
 - ② 单片机等待事务完成；事务完成后，CH375 请求单片机中断。
 - ③ 发出 `GET_STATUS` 命令获取中断状态，CH375 撤消中断请求。
 - ④ 单片机分析获得的中断状态，如果操作失败则进行异常处理；如果是 `USB_INT_SUCCESS` 则事务执行成功。
 - ⑤ 发出 `RD_USB_DATA0` 命令获取 USB 设备返回的数据，并保存作为控制传输的返回结果。
 - ⑥ 因为该控制传输只需要一次 `IN` 事务，所以数据阶段完成。
- (3) 控制传输的状态阶段
- ① 发出 `WR_USB_DATA7` 命令将 `0` 长度的状态数据写入输出缓冲区，长度为 `0`。
 - ② 发出 `ISSUE_TKN_X` 命令执行事务，同步标志为 `40H`，事务属性字节是 `01H`，向默认端点 `0` 发出 `OUT` 令牌及发送 `DATA1`。
 - ③ 单片机等待事务完成；事务完成后，CH375 请求单片机中断。
 - ④ 发出 `GET_STATUS` 命令获取中断状态，CH375 撤消中断请求。
 - ⑤ 单片机分析获得的中断状态，如果操作失败则进行异常处理；如果是 `USB_INT_SUCCESS` 则事务执行成功，状态阶段完成。
- (4) 控制传输完成，在数据阶段中返回的数据作为标准 USB 请求 `GET_STATUS` 的返回数据，通常返回数据的长度是 `2` 字节。