



Innovative Design of Products and Systems

DATA FLOWS

Microsoft USB Simulator

Document Number 621F01
Revision 02

Prepared for Microsoft Corp.
Date: 4-Dec-2000, 3:55 PM

*This document is the property of INDesign, LLC and is considered **CONFIDENTIAL**.
© 2000 INDesign, LLC. All rights reserved.*

DATA FLOWS	INDesign, LLC 5640 Caito Drive Indianapolis, IN 46226-1346 Phone: 317.377.5450 FAX: 317.377.5455 http://www.INDesign-LLC.com
MICROSOFT USB SIMULATOR	

<i>Document Change Notes</i>			
Rev	Description	Date	Changed by
01	Rough draft for client review of concept	11/29/00	K. L. Payne
02	Updates from first review, added additional data flows, standardized command formats, added TCP/IP port info	12/04/00	K. L. Payne

DOCUMENT NUMBER 621F01	REV 02	<i>Confidential</i>	Page 2 of 17
---------------------------	-----------	----------------------------	--------------

TABLE OF CONTENTS

1.0 Introduction	4
2.0 Controller/Simulator Commands	4
2.1 USB Data Commands	5
2.2 USB Handshaking Commands	5
2.3 IP Query Commands	5
2.4 Simulator Status Commands	5
2.5 Simulator Setup Commands	6
2.5.1 Endpoint Setup Structure	7
3.0 Data Flow Diagrams	7
3.1 Get Device Descriptor	8
3.2 Set Address	9
3.3 Get Configuration Descriptor	10
3.4 Bulk Pipe	11
3.5 Interrupt Pipe	12
3.6 Isochronous Pipe	15

TABLE OF TABLES

Table 1: Command Categories	4
Table 2: General Command Format	4
Table 3: USB Data Command Format	5
Table 4: USB Handshaking Command Format	5
Table 5: IP Query Command Format	5
Table 6: Simulator Status Command Format	6
Table 7: Simulator Setup Command Format	6
Table 8: Endpoint Setup Structure	7

TABLE OF FIGURES

Figure 1: Get Device Descriptor	8
Figure 2: Set Address	9
Figure 3: Get Configuration Descriptor	10
Figure 4: Bulk Pipe (Transfer to Device)	11
Figure 5: Interrupt Pipe (no preloaded data, no autorepeat)	12
Figure 6: Interrupt Pipe (preloaded data, no autorepeat)	13
Figure 7: Interrupt Pipe (preloaded data, autorepeat on)	14
Figure 8: Isochronous Pipe (no preloaded data, no autorepeat)	15
Figure 9: Isochronous Pipe (preloaded data, no autorepeat)	16
Figure 10: Isochronous Pipe (preloaded data, autorepeat on)	17

MICROSOFT USB SIMULATOR	INDesign
Data Flows	

1.0 INTRODUCTION

This document provides the data flows for the USB Simulator. These flows show the data transfer to/from the USB port of the simulator, thru the Simulator, and to/from the Controller.

Each Simulator can contain a maximum of 4 USB boards. Each USB board is considered a USB device. Each USB device (board) can have a maximum of 8 endpoints (including the mandatory control endpoint 0). Each USB board will be logically connected to a specific TCP/IP port (201-204). The Simulator will respond to broadcast messages on TCP/IP port 200.

The goal is for the Simulator to remain as un-intelligent concerning data content as possible. It is, therefore, the responsibility of the Controller to maintain the setup of each USB device (board) and each endpoint within each device. As much as possible, the Simulator will simply pass data from the USB port(s) to the TCP/IP port(s).

2.0 CONTROLLER/SIMULATOR COMMANDS

To understand the flow of data within the Simulator, it is necessary to understand the commands that are used between the Simulator and the Controller. Commands between the Controller and the Simulator fall into 5 categories, as shown in Table 1, and are detailed in the following subsections. Three of the commands come from the Controller over the TCP/IP ports dedicated to a USB board. The other two commands come over the broadcast USB port (200).

Table 1: Command Categories

Command	Function	TCP/IP Port(s)
1	USB data	201, 202, 203, 204
2	USB handshaking	201, 202, 203, 204
3	IP Query	200
4	Simulator status	200
5	Simulator setup	201, 202, 203, 204

All commands have the format shown in Table 2. If a command is not endpoint-specific, the USB Endpoint field can be filled with any value

Table 2: General Command Format

Command	Subcommand	USB Endpoint	Data Size	Data
1 byte	1 byte	1 byte	2 bytes	variable

MICROSOFT USB SIMULATOR	INDesign
Data Flows	

2.1 USB DATA COMMANDS

USB Data Commands transfer USB received data from the Simulator to the Controller, and USB data to transmit from the Controller to the Simulator. Each command has the format shown in Table 3.

Table 3: USB Data Command Format

Command	Subcommand	USB Endpoint	Data Size	Data	Comment
0x01	0x00	0x00 – 0x07	Size of the Data field	Variable, first byte is PID	Same command, either direction

2.2 USB HANDSHAKING COMMANDS

USB Handshaking Commands transfer USB handshaking responses (ACKs/NAKs) from the Simulator to the Controller. These handshakes may be either to or from the Simulator. Each command has the format shown in Table 4.

Table 4: USB Handshaking Command Format

Command	Subcommand	USB Endpoint	Data Size	Data	Comment
0x02	0x01	0x00 – 0x07	0x01	0x4B or 0x5A	ACK or NAK
0x02	0x02	0x00 – 0x07	Size of the Data field	Variable	Other data

2.3 IP QUERY COMMANDS

IP Query Commands request IP responses from the Simulator. Each command has the format shown in Table 5. These commands are local to the Simulator and are not mirrored on the USB side. These commands will be handled over the TCP/IP 200 port.

Table 5: IP Query Command Format

Command	Subcommand	USB Endpoint	Data Size	Data	Comment
0x03	0x00	Unused	0x00	None	Request for IPs from Controller
0x03	0x00	Unused	0x04	4 bytes of IP address	Response from Simulator

2.4 SIMULATOR STATUS COMMANDS

Simulator Status Commands request/return status from the Simulator. Each command has the format shown in Table 6. These commands are local to the Simulator and are not mirrored on the USB side. These commands will be handled over the TCP/IP 200 port.

DOCUMENT NUMBER 621F01	REV 02	<i>Confidential</i>	Page 5 of 17
---------------------------	-----------	---------------------	--------------

Table 6: Simulator Status Command Format

Command	Subcommand	USB Endpoint	Data Size	Data	Comment
0x04	0x01	Unused	0x00	None (from Controller to Simulator)	Get Version
0x04	0x01	Unused	Size of the Data Field	Variable (from Simulator to Controller)	Returns Version
0x04	0x02	Unused	0x00	None (from Controller to Simulator)	Get MAC Addresses
0x04	0x02	Unused	0x04	4-bytes of MAC address (from Simulator to Controller)	Return MAC Addresses
0x04	0x03	Unused	0x00	None (from Controller to Simulator)	Get Device Status
0x04	0x03	Unused	0x01	1-byte bit-masked status of connected devices (from Simulator to Controller)	Returns Device Status

2.5 SIMULATOR SETUP COMMANDS

Simulator Setup Commands are from the Controller to the Simulator, only, and configure the operation of the Simulator as a USB device. Each command has the format shown in Table 7.

Table 7: Simulator Setup Command Format

Command	Subcommand	USB Endpoint	Data Size	Data	Comment
0x05	0x01	0x00 – 0x07	0x03	Structure (see Section 2.5.1)	Sets attributes of selected USB endpoint
0x05	0x02	0x00	0x01	0x00 – 0x7F	Sets the USB address
0x05	0x03	0x00	0x01	0x00, 0x01	0x00: Disconnect device (board) from USB 0x01: Connect device (board) to USB

2.5.1 Endpoint Setup Structure

This structure is used to configure a given endpoint (see command 0x05 0x01). It is the responsibility of the Controller to be sure that all endpoints for a given device (board) are compatible (i.e., can't have 4, 1024-byte ISOC pipes setup). If a structure is not received for a given endpoint, that endpoint is not enabled. This structure contains the fields shown in Table 8.

Table 8: Endpoint Setup Structure

Field	Content	Comment
FIFO size	0x00 – 0x03	0x00: non- ISOC = 16 bytes, ISOC = 64 bytes 0x01: non-ISOC = 64 bytes, ISOC = 256 bytes 0x02: non-ISOC = 8 bytes, ISOC = 512 bytes 0x03: non-ISOC = 32 bytes, ISOC = 1024 bytes note: the total FIFO size for all enabled endpoints can not exceed 1120 bytes
Set Endpoint type (disabled, ISOC, non-ISOC)	0x00 – 0x02	0x00 – disables endpoint 0x01 – sets endpoint for ISOC 0x02 – sets endpoint for non-ISOC
Autorepeat	0x00, 0x01	0x00 – sets endpoint to NOT auto send last packet 0x01 – sets endpoint to auto send last packet

3.0 DATA FLOW DIAGRAMS

This section shows Data Flow Diagrams for the different USB messages. In these diagrams, time advances from the top of the diagram to the bottom. While these diagrams show a sequential flow of data, it is possible that, due to the asynchronous nature of the data, the order of events may be intermixed. The USB side shows the PID that was transferred/received, the Controller side shows the commands, as outlined in Section 2.0 .

3.1 GET DEVICE DESCRIPTOR

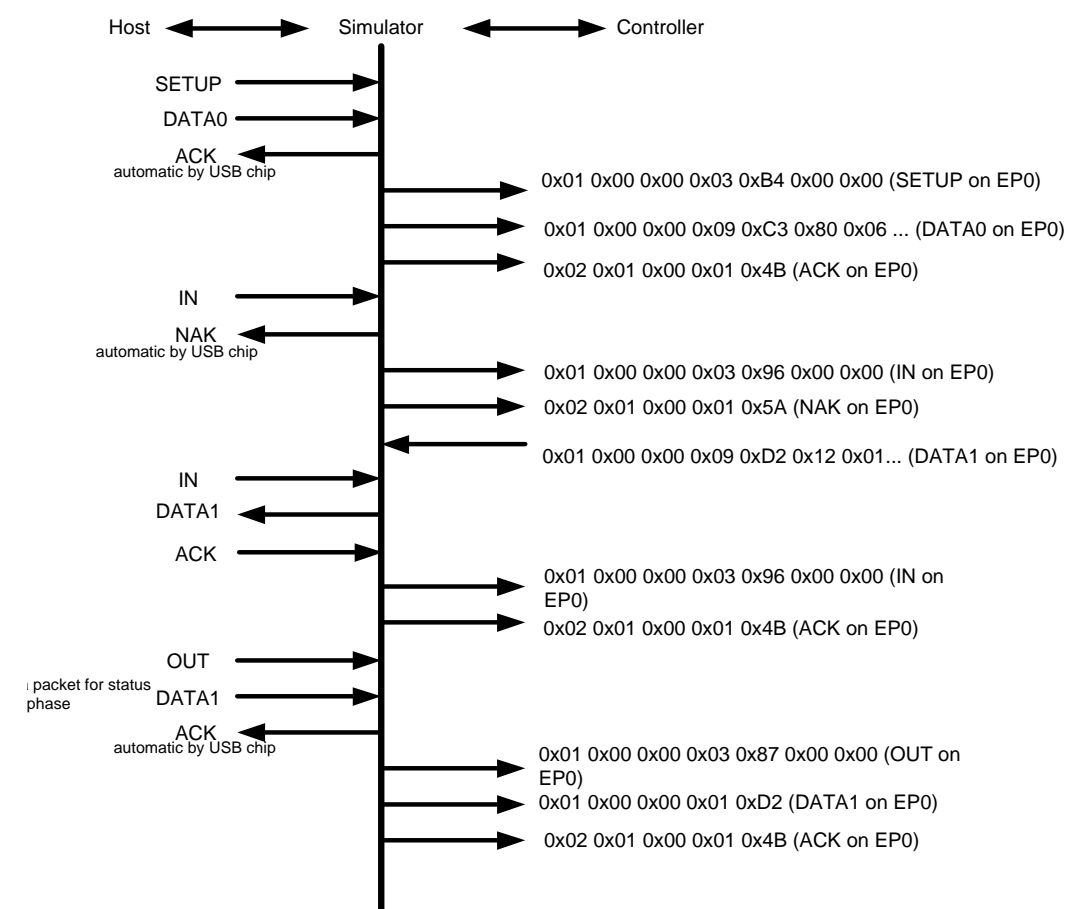


Figure 1: Get Device Descriptor

3.2 SET ADDRESS

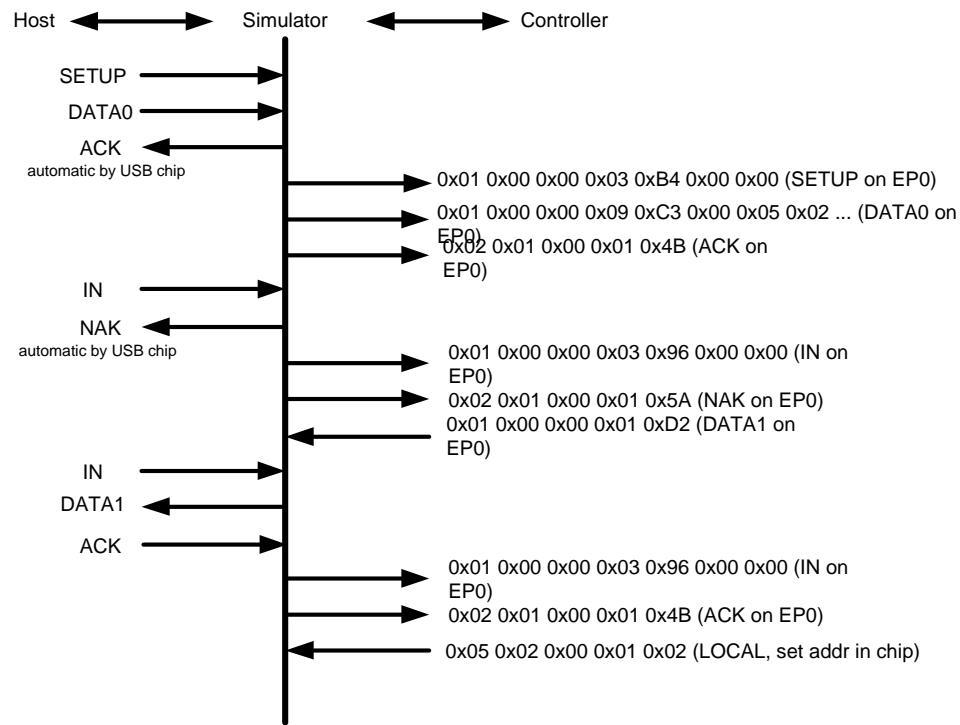


Figure 2: Set Address

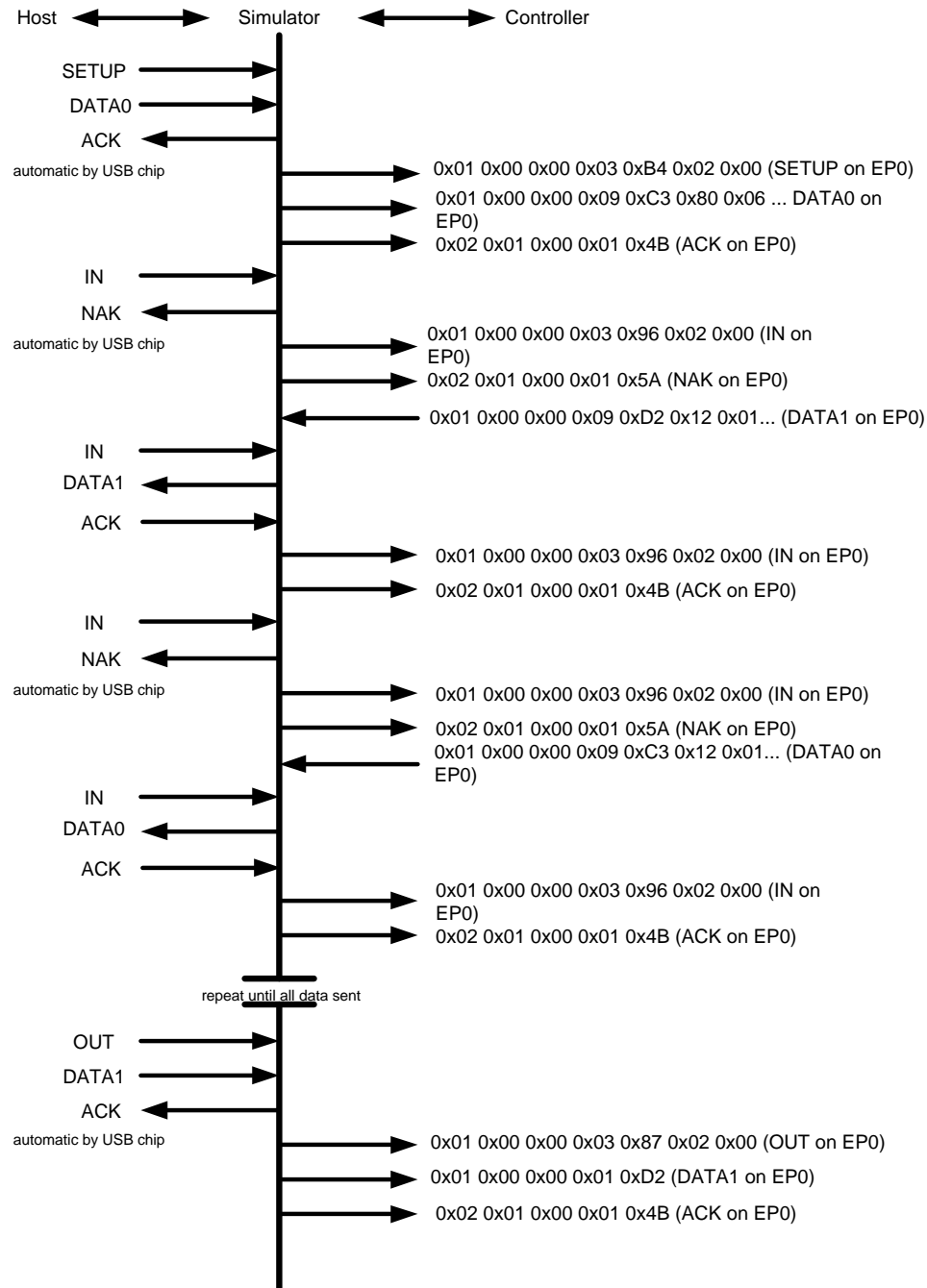


Figure 3: Get Configuration Descriptor

3.4 BULK PIPE

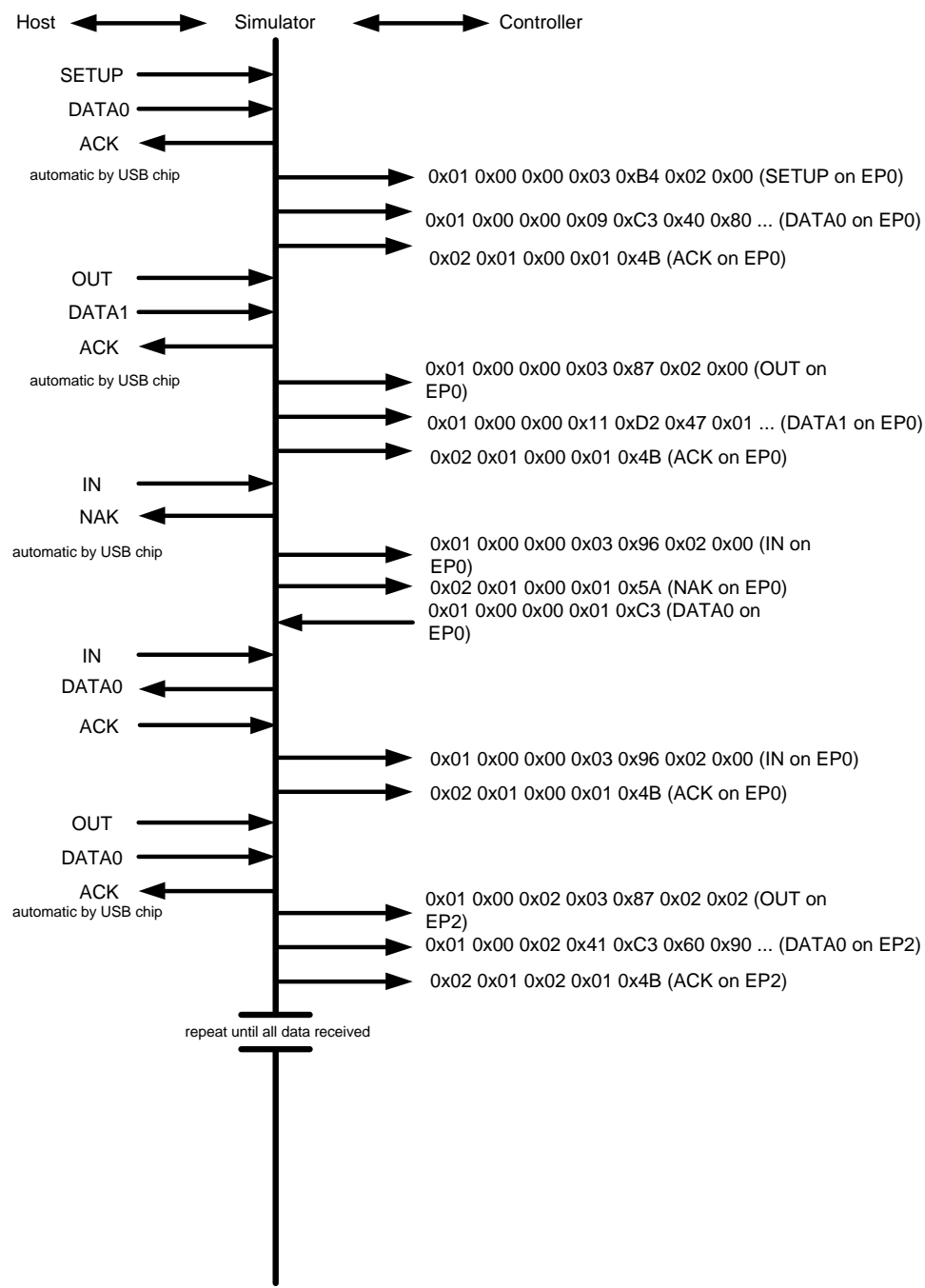


Figure 4: Bulk Pipe (Transfer to Device)

3.5 INTERRUPT PIPE

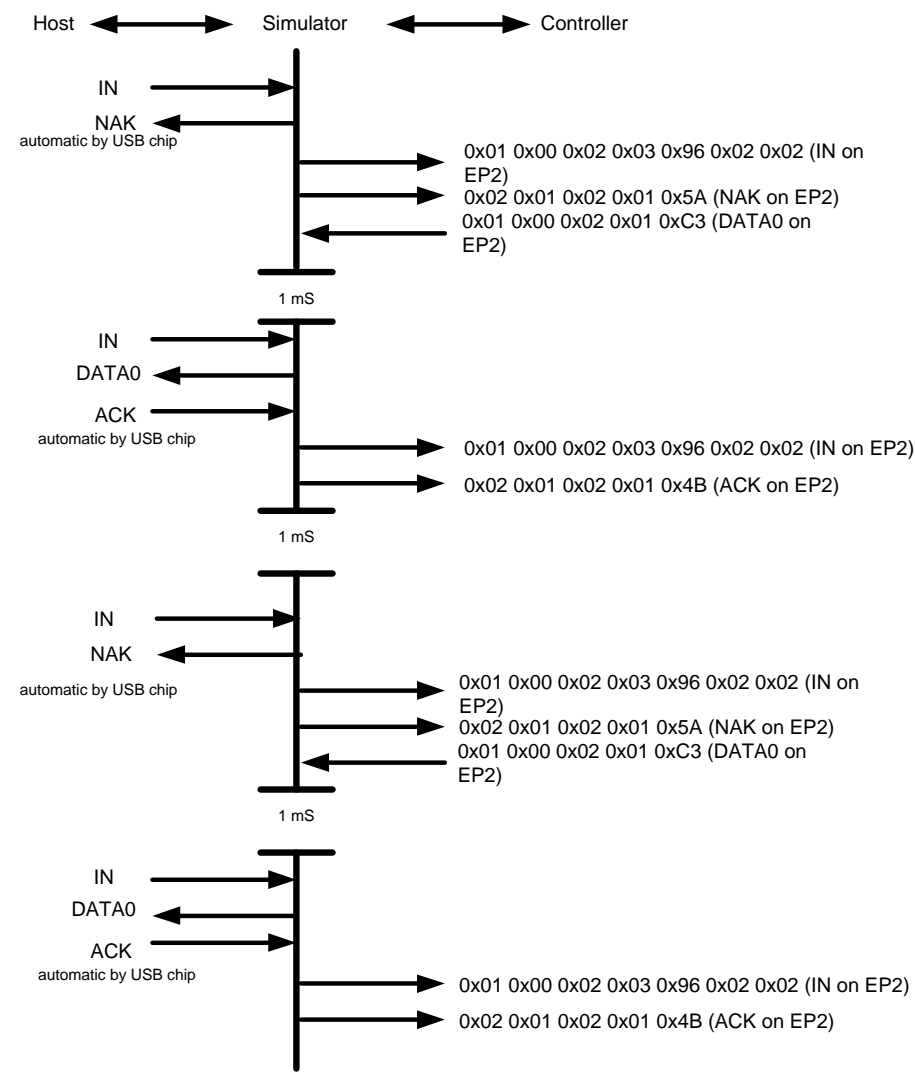


Figure 5: Interrupt Pipe (no preloaded data, no autorepeat)

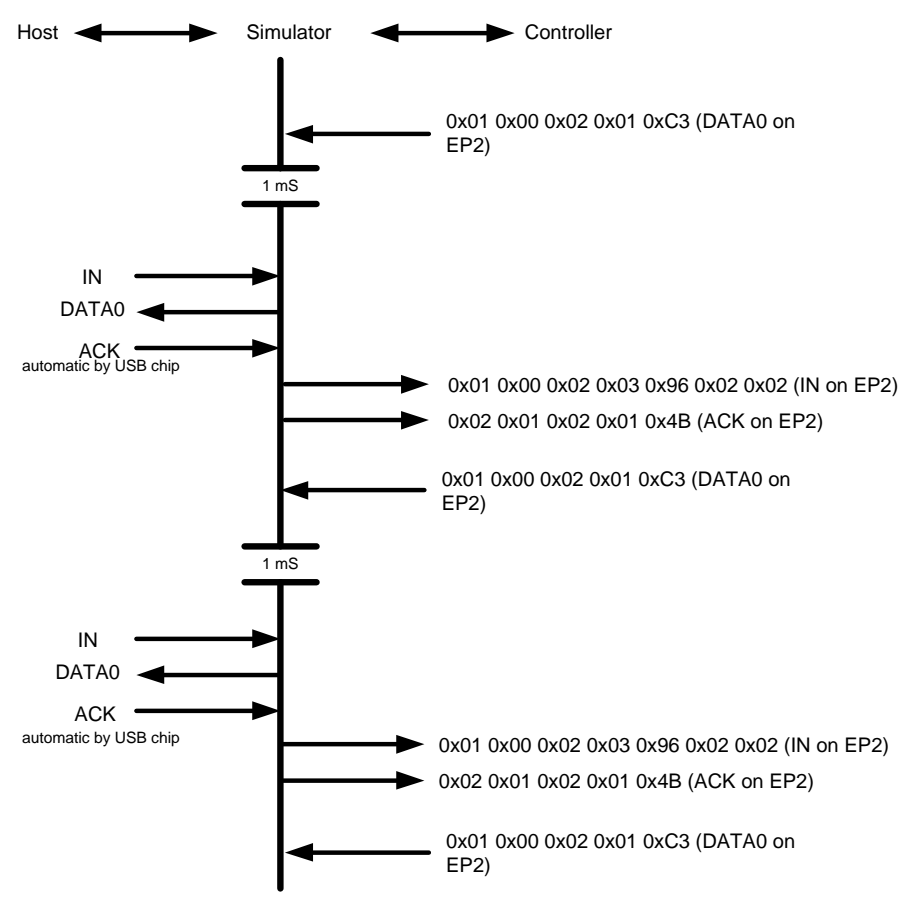


Figure 6: Interrupt Pipe (preloaded data, no autorepeat)

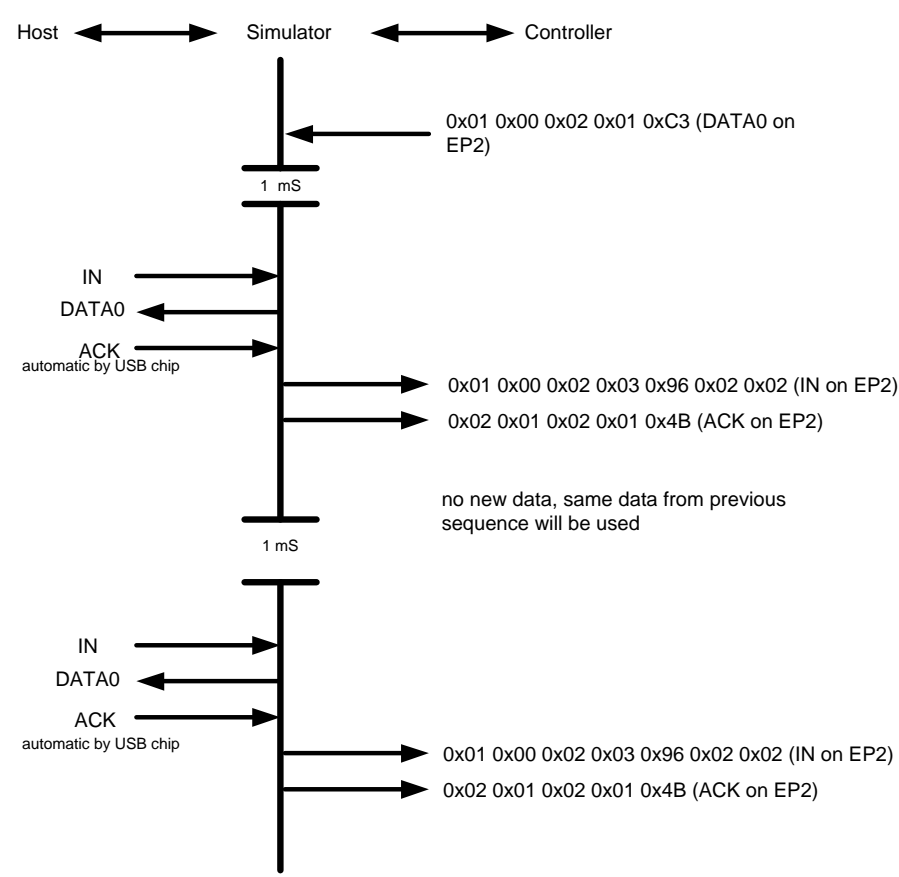


Figure 7: Interrupt Pipe (preloaded data, autorepeat on)

3.6 ISOCHRONOUS PIPE

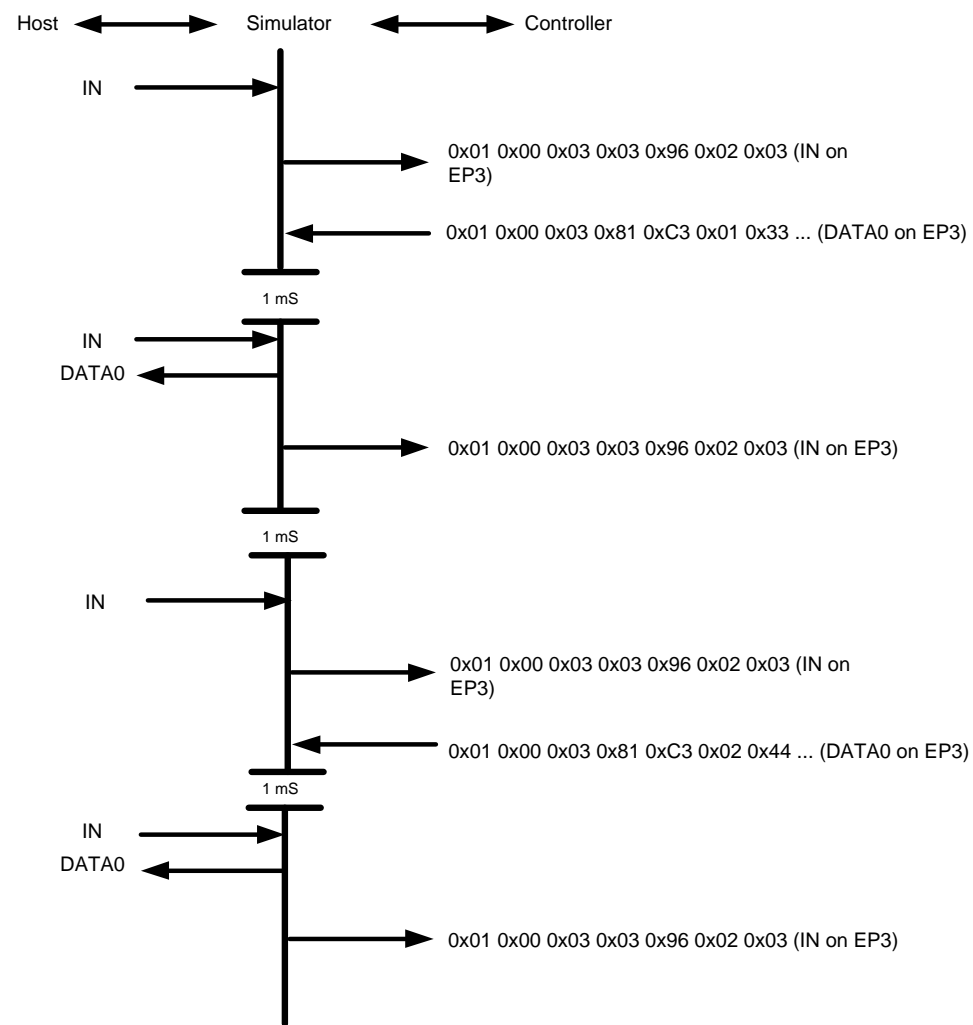


Figure 8: Isochronous Pipe (no preloaded data, no autorepeat)

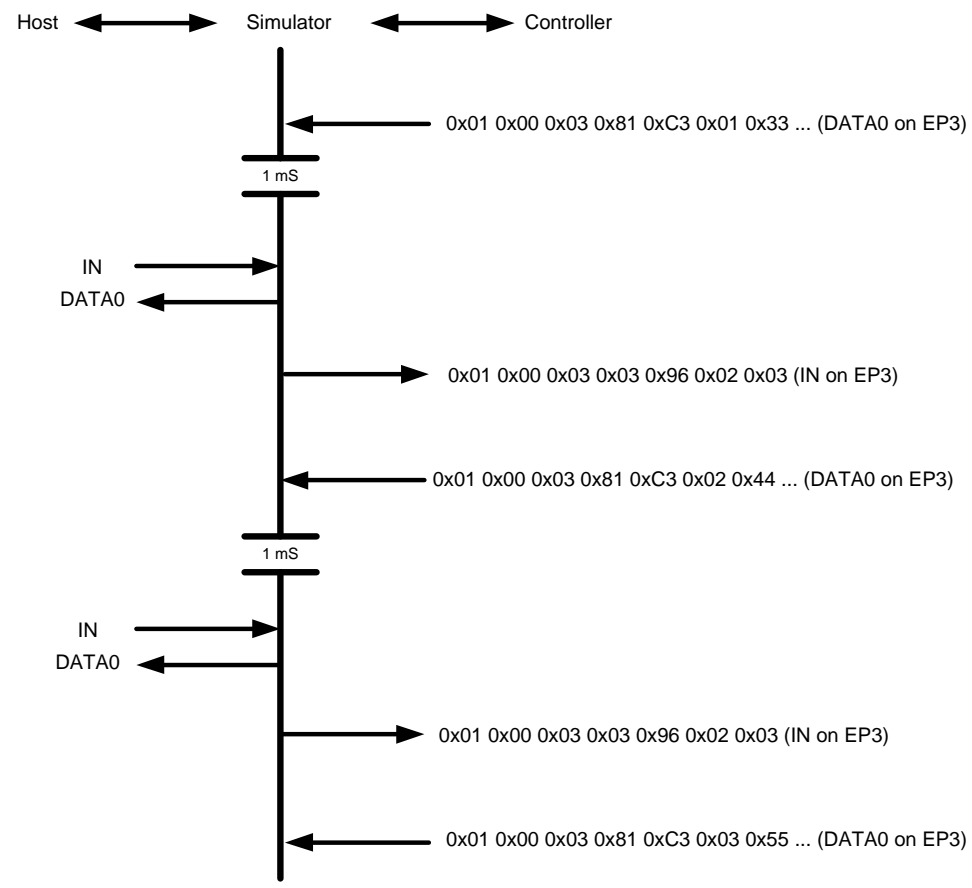


Figure 9: Isochronous Pipe (preloaded data, no autorepeat)

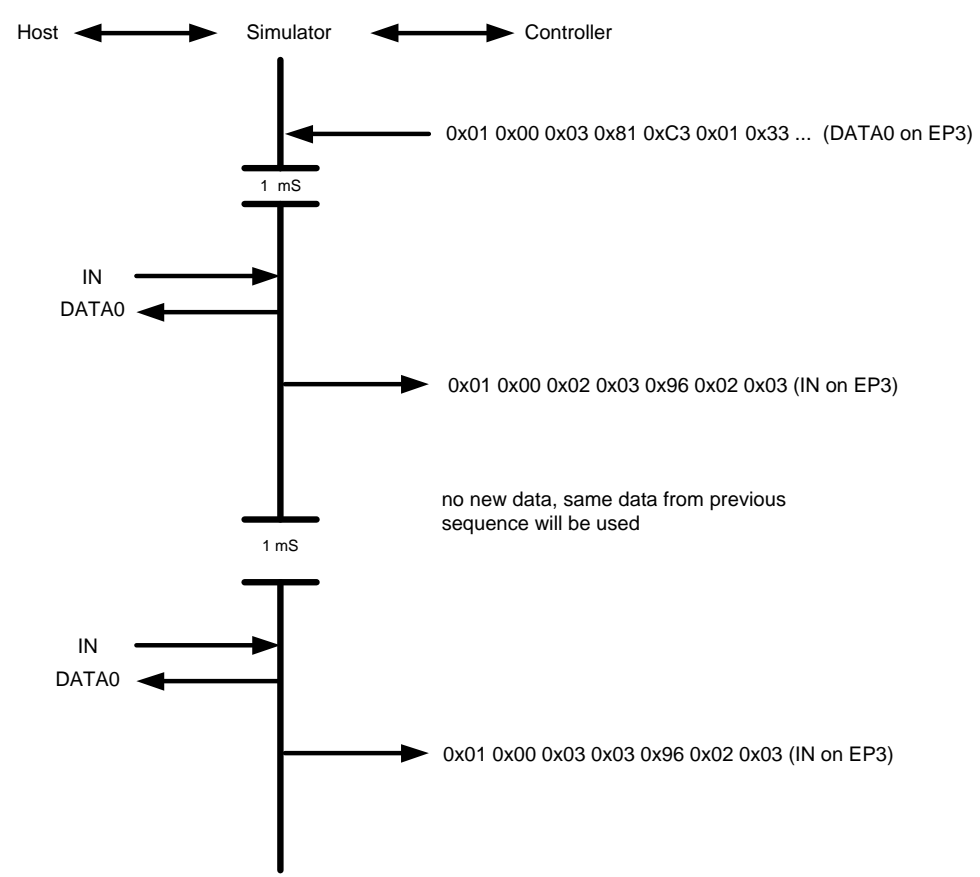


Figure 10: Isochronous Pipe (preloaded data, autorepeat on)