UNRELEASED / CONFIDENTIAL

INTRODUCTION

The **Front Desk Bus** is a method and protocol for interconnecting computers with human input and other devices. This specification covers the **Physical,Datalink**, and **Network** layers of the **Front Desk Bus**. In this specification the computer is referred to as the host. Peripherals connected to the bus are referred to as devices.

The host is the undisputed bus master. It controls the flow of data by issuing **Commands** and it is the only device permitted to issue them. **Talk** is the command used for a data transaction from a device to the host. **Listen** is the command used for a data transaction from the host to a device.

PHYSICAL LAYER

Interconnection:

All devices will communicate with the host via a 3.5 mm mini phono jack, as specified in Apple Specification,T.B.D., with the following connector assignments; Tip-Power, Ring-Data, Sleeve-Power Return They will be interconnected with three conductor cables terminated with 3.5 mm mini phono plugs, as specified in Apple Specification, T.B.D.

Signal Levels:

Input Signals:

Data:

The data line will be pulled up by the host with a 10 K Ohm resistor to power. A "**High**" is 2.4 V minimum . A "**Zero**" is 0.8 V maximum .

Power:

The Host will supply 5.0 Vdc ± 10% to the devices. The power line will be current limited by the host to prevent systems damage in the event of a Power to Power Return short.

TOTAL BUS POWER ?

Output Signals:

A "**High**" is the voltage on the Tip connection. Each device in the inactive or "**High**" state must source less than 50 uA. A "**Zero**" is 0.4 V maximum at 1.6 mA minimum. Devices will provide current limiting on the data line to prevent damage to the device in the event of a Power to Data line short.

Modulation:

There are three forms of modulation on the bus, **Normal** modulation which transmits commands and data, **High Speed** modulation which transmits data, and **Signals** which broadcast global messages such as **Service Request** and **Reset**.

Normal Modulation:

An RZ code for modulation has been adopted for the Front Desk Bus. Each bit cell boundary is signified by a falling edge on the bus. The period of each bit cell is the time between two falling edges on the bus. The time for a normal modulation bit cell, T_{cyc}, is 100 usec ± 30%. All devices must support normal modulation for data transactions.

The data is encoded as the ratio of low to high time of each bit cell. Thus a "0" is encoded as a bit cell in which the low time is greater than the high time. Conversely, a "1" is encoded as a bit cell in which the low time is less than the high time. A **Start** is defined as a "1".A **Stop** is similar to a "0", in that it has a low time of T0 , but it does not have another negative edge to define the bit cell time. It is used to synchronize the stopping of a transactions.

High Speed Modulation:

High speed modulation is only used for data and not commands.The time for a high speed modulation bit cell is 50 usec ± 1%.

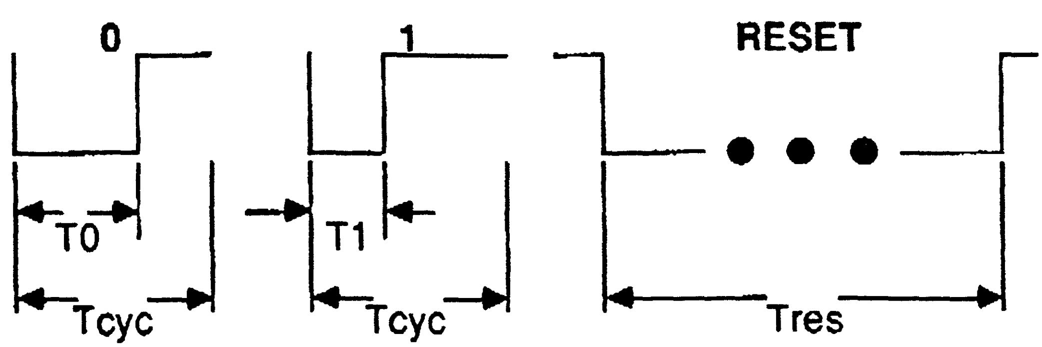


Fig. 1 FDB "0", "1" and RESET Times

Signals:

Certain transactions fall under the category of neither commands nor data transactions. These are special transactions which globally broadcast status to devices on the bus. There are four special transactions in this group.

Attention and Sync:

To signal the start of a command, a long **attention** pulse is sent. This is followed by a **synch** pulse to give the initial bus timing. The falling edge of the **synch** pulse is used as a timing reference for the first bit of the command .

Reset:

Reset issues a break on the bus by holding the bus low for a minimum of T_{res}.

Service Request:

Service Request is a transaction that devices can use to signal the host that they require service, i.e. have data to send. Following any command transaction, a requesting device can signal by holding the bus low during the low portion of the stop bit of the **Command** transaction. The requesting device holds the bus low T_{int} beyond the bit cell boundary to signal.

Once a device has requested service, it shall **Request Service** repeatedly until serviced. When the requesting device is addressed to **Talk**, it shall be considered serviced and not **Request Service** again until it again needs to be serviced. The ability for a device to **Request Service** can be enabled and disabled by the host.

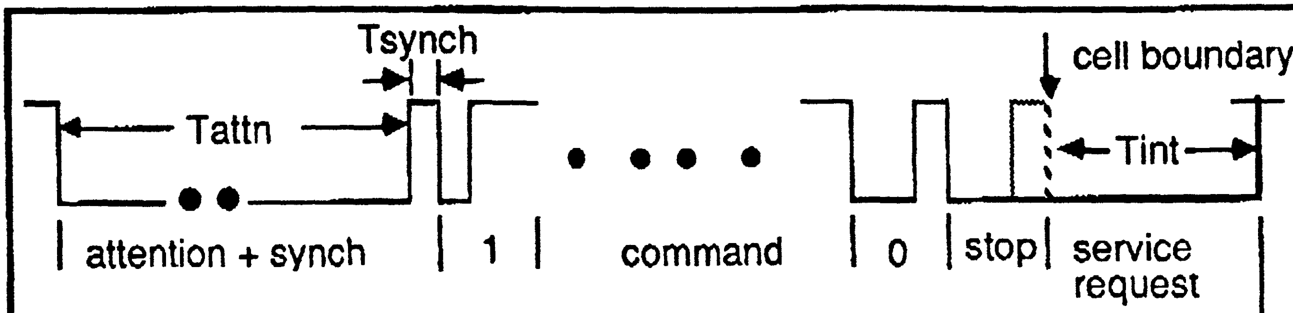


Fig. 2 Service Request Transaction

FDB Interface Characteristics						
Symbol	**Parameter**	**Min.**	**Max.**	**Unit**	**Fig.**	**Definition**
T0	"0" low time	60	70	% Tcyc	1	
T1	"1" low time	30	40	% Tcyc	1	
Tattn	ATTENTION signal	560	1040	usec	2	8 * Tcyc
Tcyc	FDB bit cell time	70	130	usec	1	
Tint	INTERRUPT signal	140	260	usec	2	2 * Tcyc
Tres	RESET signal	2.8	5.2	msec	1	40 * Tcyc
Tsynch	Synch pulse width	30	40	% Tcyc	2	
Tlt	Stop to start time	140	260	usec	3	2*Tcyc

Transactions:

Commands:

The format of a **command** is an **attention** signal, followed by a **sync** signal, followed by eight data bits,and to synchronize the stopping of the transaction, a **stop** bit. Following the imaginary bit cell boundary after the **stop** bit, the transaction is complete and the host releases its active drive of the bus.

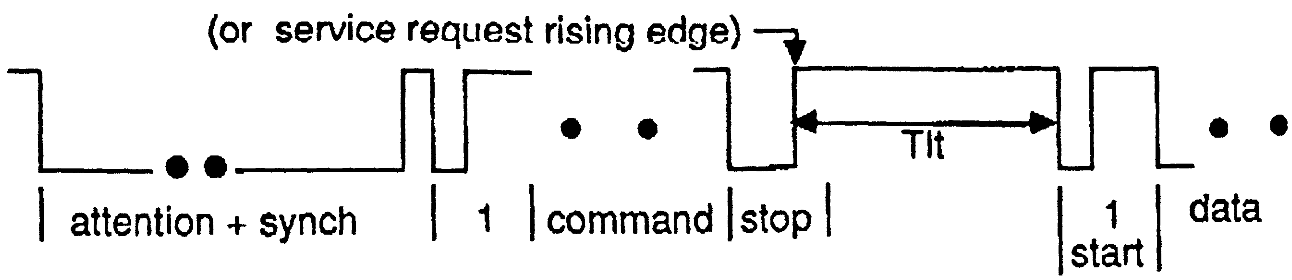


Fig. 3 Command Stop to Data Start

Data:

The format of a **data** transaction is a **start** bit ,followed by N times 8 bits of data with the most significant data bit sent first, where N may be from 2 to 64, followed by a stop bit.

UNRELEASED / CONFIDENTIAL

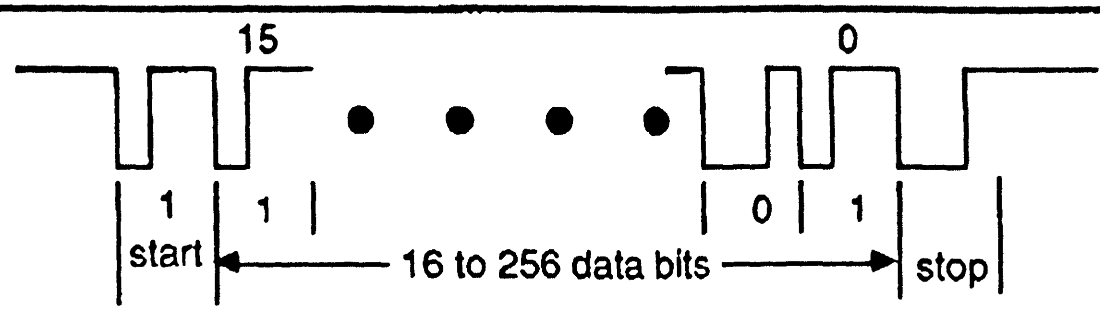


Fig. 4 Data Transaction Example

DATA LINK:

Each device on the bus has an address. There shall be only one active talker on the bus at any time, this may be the host or an addressed device. A device addressed to talk, with data to send, "untalks" itself after it sends its data. If a device has no data to send it "untalks" itself immediately and allows the bus to time-out.The host may also sent data after a command.

Front Desk Bus Peripherals:

Addresses:

All devices have a four bit command address which is defined by device type assignment. A device will always respond to that address upon either power on or a reset signal. The Addresses are assigned as follows:

UNRELEASED / CONFIDENTIAL

Device Table			
Address	**Device type**	**Extended address**	**Example**
0000 (0)	ADAPSO keys	yes	
0001 (1)	Appliances	yes	
0010 (2)	Coded devices	-	Keyboard
0011 (3)	Relative devices	-	Mouse
0100 (4)	Absolute devices	-	Tablet
0101 (5)	Reserved	-	
0110 (6)	Reserved	-	
0111 (7)	Reserved	-	
1000 (8)	Soft addressed	-	
. .	. .	. .	
1111 (15)	Soft addressed	-	

Registers:

All devices have at most four locations to receive data, and at most four locations to send data. These locations are called **registers** and are referred to as R0 to R3. They are defined as follows:

Register 0 Talk: Data Register, Device specific as to meaning.

Register 0 Listen: Data Register, Device specific as to meaning.

Register 1 Talk: Data Register, Device specific as to meaning.

Register 1 Listen: Data Register, Device specific as to meaning.

Register 2 Talk: Data Register,Device specific as to meaning.

Register 2 Listen: Soft Addressed Devices: Device specific as to meaning.
Extended Address Devices: Enabling Extended Address

UNRELEASED / CONFIDENTIAL

Register 3 Talk: Status information, ie: device address, handler.

Register 3 Listen: Status information, ie: device address, handler.

Commands:

Commands may be sent only by the host. There are four commands; Talk, Listen, SendReset,and Flush. A command is an eight bit value with the following syntax: First, there is a four bit device address field which specifies the address of the desired device. The most significant nibble is the address which ranges from 0 - 15 (A3-A0). The next two bits form the command.The last field is a two bit register address field (RB,RA). This field, which is optional, allows a specific register, **R0** to **R3** within an addressed device to be specified. An example of where this might be used is to differentiate a data register (in a keyboard, the specific keystroke) from a status or configuration register (in a keyboard, a response that signifies the model of the keyboard).

Command Syntax			
7654	32	10	Command
XXXX	00	00	SENDRESET *
A3 - A0	00	01	FLUSH
XXXX	00	10	RESERVED
XXXX	00	11	RESERVED
XXXX	01	XX	RESERVED
A3 - A0	10	RB RA	LISTEN
A3 - A0	11	RB RA	TALK

* forces RESET signal on FDB

To allow for future expansion of the command structure, a group of "place holder" Reserved instructions has been defined. These instructions shall be treated as no-ops. → [illegible] Release

As a specific example, a **Talk** command to **Register 0** of device **5** would be encoded as "01011100". The bus would be modulated with the following:

UNRELEASED / CONFIDENTIAL

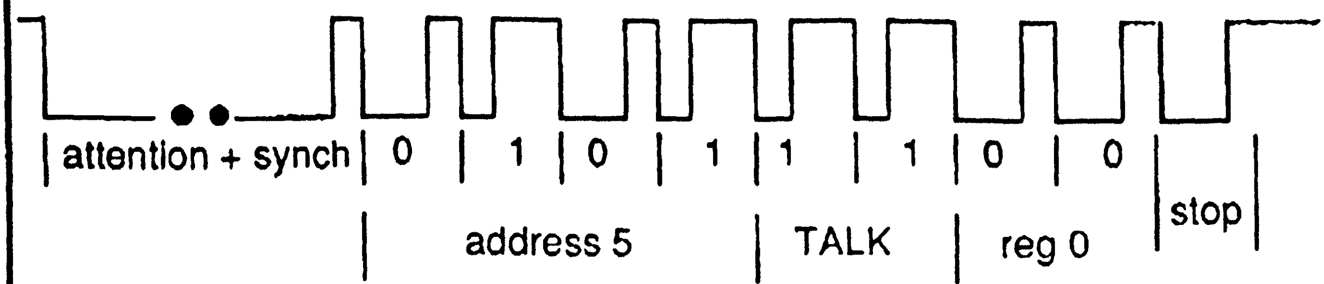


Fig. 5 TALK Command Example

Talk:

All devices on the bus must support **Talk** and **Listen** commands. When a device is addressed to **Talk**, it must respond before being timed out by the host. This timeout shall be T_{lt} max. after the rising edge of the stop bit of the **Talk** command.

The selected device, if it does not timeout, becomes active on the bus. It performs its data transaction no sooner then T_{lt} min after the rising edge of the stop bit of the **Talk** command then "untalks" itself and goes inactive on the bus.

Listen:

When a device is addressed to **Listen**, it is enabled to receive the data bits that are placed on the bus by the host. The host performs its data transaction within T_{lt} after the rising edge of the stop bit of the **Listen** command After the data bits are received, the transaction is complete and the device "unlistens" itself. If a device is addressed to **Listen** and it receives another command on the bus before it receives any data, then by definition the transaction is immediately complete and the device "unlistens" itself.

SendReset:

The **SendReset** commmand causes a **Reset** signal to be put on the bus. The **Reset** has the effect of reseting all pending **Service Requests**; turning the service request mode of all devices to **enable**; and in general puts the devices in a mode in which they will accept commands.

Flush:

The effect of the **Flush** command is defined by the device. It can be used for such functions as clearing a fifo and reseting all keys on a keyboard so that they will be sent again.

UNRELEASED / CONFIDENTIAL

Collision Detection:

All devices will detect a collision of data. If a device is trying to output a one and the data line is or goes to a zero, it has lost a collision to another device. If another device sends data before the device is able to assert its start bit it has lost a collision. The losing device should immediately "untalk" itself and preserve the data which was being sent for retransmission. The device will set an internal flag if it looses a collision .

In order to aid in collision detection, devices using internal clocks which operate in within ± 1% should attempt to assert their start bit at a random time within the limits of the line turn around time, T_{lt}.

Error Conditions:

If the data line gets hung low for T_{res}, all devices will reset themselves and output a one. If a command transaction is incomplete by staying high beyond the maximum bit cell time, all devices will ignore the command and seek another attention signal.

NETWORK LAYER:

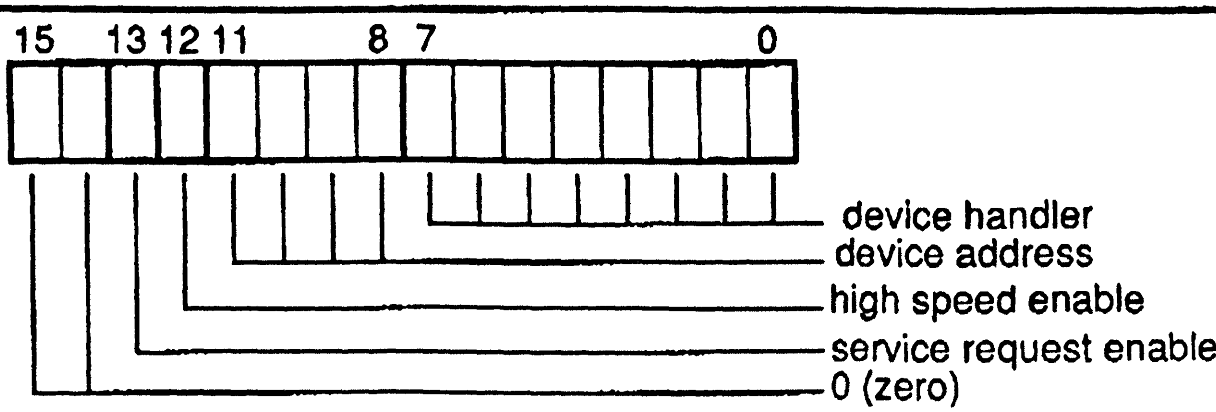
Front Desk Bus Peripheral Types:

Normal devices:

Normal device will optionally have a device on them to indicate activity, which is called the **activator**. The **activator** can be a special key on a keyboard or a mouse button. In order to aid in collision detection they will also replace the address portion of the address field of **Register 3** with a random number in response to a **Talk R3** command. **Normal devices** will change their **Register 3** to the data received when they receive a **Listen R3** command and the condition of no collision detected and **activator** inactive is true.

At the systems level a host can change the address of normal devices by forcing the collision of devices sharing the same address. By issuing a **Talk R3** command and following it with a **Listen R3** command, with a new address in bits 8 to 11 of the data, all devices which detected collisions will be moved to the new address. This process can be repeated at new addresses until the response to the **Talk R3** command is a time out. This can be used to identify and relocate multiple devices of the same type after initialization of the system.

At the applications level addresses can by changed by displaying a message requesting a user to use the **activator**. The host then issues a **Listen R3** command to a new address and all devices except the one with the **activator** being used are moved. This can be used to identify and locate individual devices in multi-user applications.



Register address 11 (3)

Extended Address devices:

These devices all have the same command address as well as a unique 16 bit extended address which is stored in the device. Their command address may not be changed. On power up or after **Reset** they will only accept the **Listen R2** command. They are enabled to talk and listen only after receiving a **Listen R2** command in which the data matches their stored address. Once enabled they will respond to all commands addressed to them. These devices become disabled after receiving a **Listen R2** command in which the data does not match their stored address.

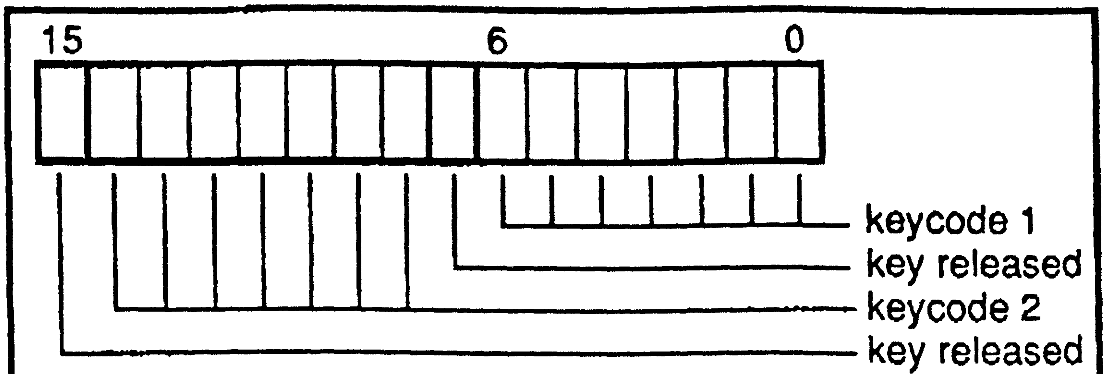
Register 3:

Handlers:

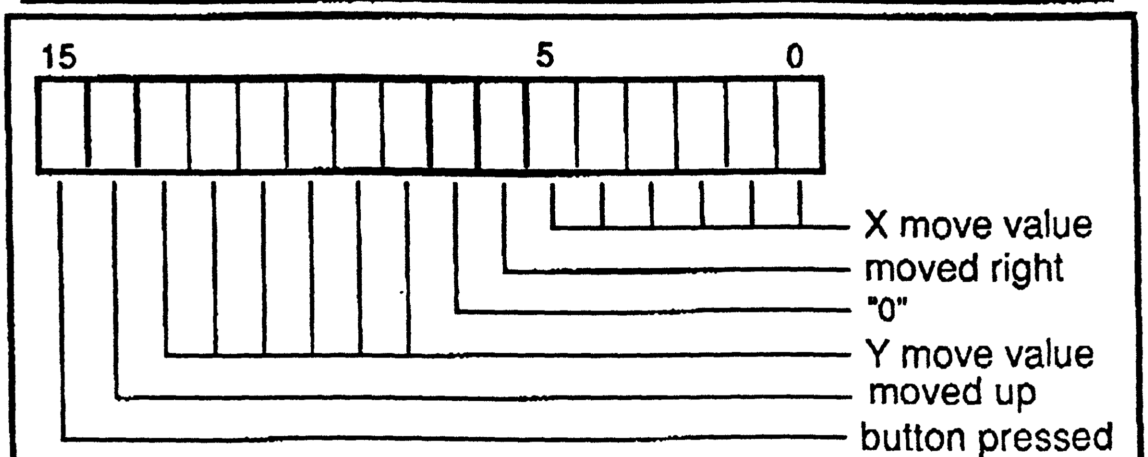
The function of a device and the use of data from them by the host is defined by a handler which is stored by the device in **Register 3**. The host is able to change the handler with a **Listen R3** command. If the receiving device is able to function to the new handler it will be stored and sent in response to a **Talk R3** command.

UNRELEASED / CONFIDENTIAL

The handler "FF" hex is reserved for the self test mode for all devices. The handler "00" hex in response to a talk is reserved to indicate a failed self test. The handler "00" hex sent with a listen is reserved to indicate that the device is only to change the address portion of R3. The following are examples of how the handlers for keyboards (coded device) and mice (relative device) could interpret data recived from a Talk R0 command. For specific information refer to the individual device specifications.



Keyboard register address 00 (0)



Mouse register address 00 (0)

Service Request and High Speed Enabling:

The Listen R3 command is also used to enable and disable Service Request and High Speed modulation. They are enabled by setting the appropriate bit in R3 to a one and disabled by setting the appropriate bit to a zero.

UNRELEASED / CONFIDENTIAL

The Service Request enable bit deals with the ability of devices on the bus to Request Service from the host. Setting the bit allows the device to signal a Service Request on the bus, or conversely, clearing the bit disables the signalling of a Service Request. This is useful in systems where the Service Request response time in a polling system is longer than desired. When only specific devices are required for an application the others could be disabled.

