EW

# DATALINK COMMUNICATIONS PROTOCOL - Version 5

Mike Saari 8/30/83
File: protocol.tp2

This describes the latest version of a protocol between the Apple, host transceiver, and Topo.

Version 5 changes- Apple/Topo message formats deleted. This document now only covers datalink protocols. Message formats will be covered separately. All data words are 8 bits. They are sent as 4 bit nibbles between Apple and datalink, then reassembled. Datalink commands are single characters, with some command letter changes. IR packets are increased to 8 characters.

Version 4 changes- Topo carrier timeout time reduced to 2 seconds. Carrier sent every 300 msec. No ACK error not reported, garbled ACK error changed to "Topo not responding" - only activated after 32 tries (approx. 2 seconds). Saywhat enable deleted from CONFIGURE PACKET, and default channel set to 0, private. General status bits rearranged. Say model# deleted. Get queue remaining added. Abort deleted. Delay deleted.

Version 3 changes- Apple/datalink message format is changed to delete message counts. Command #'s are sent with the message instead of in Configure Packet. Headswitch tone selection is deleted.

Version 2 changes- Substantial changes are made in the host datalink protocol. All datalink commands are uniquely identified by preceding a command with a control-code. The Apple/Topo commands are unscathed.

This document will deal strictly with host datalink ( H.D. ) protocols. Commands passing between the Apple and Topo are covered in a separate document.

## Packet Format

All messages are sent over the IR link in packets. The packet format itself has not been changed, i.e. one bit every 1/2 ms, starting with 5 on's and 1 off, then characters coded as 8 data bits and 1 parity bit - odd parity. All packets in the system are always 8 characters, expect for short ACK's which are only a single character. The ~~4~~8 characters in the packet are ordered as follows:

ch#, proc#, cmd#, d,d,d,d, cksum .

D,d,d,d are 4 characters of data; ch# is the channel number to indicate which Topo(s) should respond; proc# is the on-board process number; cmd# is the command for the given process; and cksum is the twos complement of the sum of all the previous characters.

All packetizing, retries, and IR error handling is handled by the H.D.

## 8-bit Data and Nibbling

All data in the system, whether on the Apple or onboard Topo, consists of full 8-bit words. There is a problem, however, in that most serial cards deal with 7-bit ASCII, and also trap certain control codes. To avoid this, all data sent between the Apple and H.D. will be broken up into 4-bit nibbles, sent through the serial card, and then reassembled. (The full 8-bits are sent over the IR.) The 4-bit nibbles are encoded as ASCII 0-9 and A-F. For example, data 10100011 would be sent as A3 (hex 41, hex 33). A letter G (hex 47) would be sent as 47 (hex 34, hex 37).

## Apple/H.D. Command Syntax

Handshaking commands between the Apple and H.D. are not broken up into nibbles. All commands and responses are standard ASCII characters, excluding 0-F and excluding all control codes.

## Apple/H.D. Handshaking

A partial handshaking system is used between the Apple and the H. D. Before sending any message or command to the H.D., the Apple must first send a "QUERY" (ASCII Q). The H. D. must respond with one of several answers to indicate readiness to receive the message. The required response time is within 5 ms + 1 character time (dependent on the baud rate). (This protocol will run at 9600 baud, which gives a character time of 1 ms.) The possible responses are:

```
   Bit 3 - BUSY - Apple may not send more data yet
       2 - MESSAGE WAITING (from Topo to Apple)
       1 - ERROR - Topo not responding
       0 - ERROR - invalid message from Apple to H.D.
 all off - READY - OK to send
```

The response is a single character, where the indicated bit is active. The four high-order bits are always 1110, so responses are E0-EF (ASCII lowercase blank,a,b...m,n,o). Thus MESSAGE WAITING is 11100100, or hex E4, and BUSY is 11101000 or hex E8.

The H.D. may interrupt a long message coming in by sending a BUSY back to the Apple, which must then stop sending and QUERY until a ~~CONTINUE~~ READY is received. (The Apple always checks before sending any character to see if a BUSY has been received.)

## Apple/H.D. Command Formats

The standard message format from Apple to H.D. is: MESSAGE START (ASCII M), Topo process number, Topo command number, d,d,d ... d, and END-OF-MESSAGE (ASCII Z). This is used for every message which is to be sent from the Apple to Topo. If an extra message-start or message-end is received, ~~or a parity error is received,~~

(h#Ø)

then the bad message error code is set for the next query. Remaining characters at the end of a message, which fall short of filling a packet, are sent out in a final packet padded with trailing spaces (L+2Ø)

A single command is used to configure the H.D. for packet headers and protocol. It consists of : CONFIGURE PACKET (ASCII P), channel #, and public flag (T=public, F=private). When the public flag is set, no ACK's are expected. The power-on default value is channel# 0, private.

GET LAST RESPONSE (ASCII R) will cause the H.D. to return (d,d,d,d) that was last received back from Topo.

RESET LINK (ASCII X) cancels the current packet being attempted, clears the message buffer, resets the ERROR flags, and resets the channel number to its default. Equivalent to a power-on reset.

In the middle of a message the only legal H.D. commands are QUERY and RESET LINK, otherwise just data and END-OF-MESSAGE ( i.e. after a CONTINUE has been received.) In fact, whenever any BUSY has been received, the only legal commands are QUERY and RESET LINK.

<u>ACK0,ACK1, and Saywhat Protocol</u>

Every message received by any given Topo (except for public channels), is acknowledged by an ACK0 or an ACK1. An ACK0 "packet" is a single letter Q; an ACK1 packet is a single letter R. These must alternate to insure valid transmissions. The MSB of the channel# being transmitted indicates whether Topo should send an ACK0 or ACK1 - 0 means ACK0, 1 means ACK1. The H.D. is required to keep track of the current ACK0/ACK1 status for each private channel.

A valid ACK should be received by the H.D. within 50 40 ms of the end of the message being sent. If no ACK is received within that time, or a garbled transmission or the wrong ACK is detected, then a "saywhat?" command ( format to be determined) is given, telling the Topo to repeat its last ACK. Receiving the correct ACK means to continue normally, i.e. the original message has been correctly handled. Receiving the wrong ACK means the original message was mishandled and should be resent. Receiving nothing or a garbled reply should prompt another "saywhat?". After 3 or more saywhat's (corresponding to a period of 2 seconds) without a reply, the appropriate QUERY error flag (Topo not responding) is set while the saywhat's continue. The error flag will only be reset after the message is eventually received, or by doing a RESET LINK.

There are conceivable circumstances in which Topo can receive a command but is temporarily unable to respond within the ACK timeout time. Examples could include motion status or sensor status requests coupled with a lot of unrelated bus activity. In these cases, Topo should respond with the incorrect ACK, meaning "Topo busy, try again". This would initiate the aforementioned "saywhat?" sequence to keep trying until Topo is able to respond correctly...

When Topo sends a message to the H.D. ( generally a status requested by the Apple), the format is:

H.D. channel #,
H.D status return process #, ?
Q/R (appropriate ACK),
d,d,d,d,
cksum.

<u>IR Carrier</u>

A carrier signal tells all Topos if they go out of range of the H.D. The H.D. is required to send out some command at least once every 300 msecs. If no command has been sent after 300 msecs has elapsed, the H.D. sends out a dummy message on channel A ( the default public channel) of (format to be determined). Any Topo which hears no valid commands for 2 seconds will PARK ( abort the current motion command and flush the motion command queue ).

<u>Real-time Joystick</u>

Since the packets are relatively short ( 8 characters at 2000 baud IR gives 40 ms per packet), this should work for real-time joystick response. Since short ACK's take only another 5 ms (plus computation time), it may well work with ACK's, too. The typical configuration will be to use a public channel so as to ignore any message errors, since we are better off simply sending the newest command, instead.

Saywhat => resend last msg.

TOPO2 COMMUNICATIONS PROTOCOL - Version 4
Mike Saari 8/19/83
File: protocol.tp2

This describes the latest version of a protocol between the Apple, host transceiver, and Topo.
Version 4 changes- Topo carrier timeout time reduced to 2 seconds. Carrier sent every 300 msec. No ACK error not reported, garbled ACK error changed to "Topo not responding" - only activated after 32 tries (approx. 2 seconds). Saywhat enable deleted from CONFIGURE PACKET, and default channel set to 0, private. General status bits rearranged. Say model# deleted. Get queue remaining added. Abort deleted. Delay deleted.
Version 3 changes- Apple/datalink message format is changed to delete message counts. Command #'s are sent with the message instead of in Configure Packet. Headswitch tone selection is deleted.
Version 2 changes- Substantial changes are made in the host datalink protocol. All datalink commands are uniquely identified by preceding a command with a control-code. The Apple/Topo commands are unscathed.

The first portion will deal strictly with host datalink ( H.D. ) protocols. Commands passing between the Apple and Topo are covered later.

## Packet Format

All messages are sent over the IR link in packets. The packet format itself has not been changed, i.e. one bit every 1/2 ms, starting with 5 on's and 1 off, then characters coded as 8 data bits and 1 parity bit - odd parity. All packets in the system are always 7 characters, expect for short ACK's which are only a single character. The 7 characters in the packet are ordered as follows:

ch#, cmd#, d,d,d,d, cksum .

D,d,d,d are 4 characters of data; ch# is the channel number to indicate which Topo(s) should respond; cmd# commands a device on board Topo, and cksum is the twos complement of the sum of all the previous characters.

All packetizing, retries, and IR error handling is handled by the H.D.

## Apple/H.D. Command Syntax

All commands from the Apple to H.D. consist of a single unique control code followed by a single letter. For now, we will use ASCII CTRL-U ( 15 hex) , although this may change for the final product based on marketing studies of other computer's serial cards. This document will simply refer to "ctrl-code", thus "ctrl-code Q" means CTRL-U Q, or hex 15 51 .

## Apple/H.D. Handshaking

A partial handshaking system is used between the Apple and the H. D. Before sending any message or command to the H.D., the Apple must first send a "QUERY" ( ctrl-code Q). The H. D. must respond with one of several answers to indicate readiness to receive the message. The required response time is within 5 ms + 1 character time (dependent on the baud rate). (This protocol will run at 9600 baud, which gives a character time of 1 ms.) The possible responses are:

Bit 5 - CONTINUE - OK to continue last message
4 - MESSAGE WAITING (from Topo to Apple)

3 - BUSY - Apple may not send more data yet
2 - not used (OFF)
1 - ERROR - Topo not responding
0 - ERROR - last message invalid from Apple to H.D.
all off - READY - OK to start new message

The response is a single character, where the indicated bit is active. Bits 6 and 7 are always on. Thus MESSAGE WAITING is 11010000, or hex E0, and BUSY is 11001000 or hex C8 .

The H.D. may interrupt a long message coming in by sending a BUSY back to the Apple, which must then stop sending and QUERY until a CONTINUE is received. (The Apple always cnecks before sending any character to see if a BUSY has been received.)

## Apple/H.D. Command Formats

The standard message format from Apple to H.D. is: MESSAGE START (ctrl-code M), Topo command number, d,d,d ... d, and END-OF-MESSAGE (ctrl-code Z). This is used for every message which is to be sent from the Apple to Topo. If the end-of-message character is not in its expected place at the end of a message, the bad message error code is set for the next query. Remaining characters at the end of a message, which fall short of filling a packet, are sent out in a final packet padded with trailing spaces.

A single command is used to configure the H.D. for packet headers and protocol. It consists of : CONFIGURE PACKET (ctrl-code C), channel #, and public flag (T=public, F=private). When the public flag is set, no ACK's are expected. The power-on default value is channel# 0, private.

GET LAST RESPONSE (ctrl-code R) will cause the H.D. to return (d,d,d,d) that was last received back from Topo.

RESET LINK (ctrl-code A) cancels the current packet being attempted, clears the message buffer, resets the ERROR flags, and resets the channel number to its default. Equivalent to a power-on reset.

In the middle of a message the only legal H.D. commands are QUERY and RESET LINK, otherwise just data and END-OF-MESSAGE ( i.e. after a CONTINUE has been received.) In fact, whenever any BUSY has been received, the only legal commands are QUERY and RESET LINK.

## ACK0,ACK1, and Saywhat Protocol

Every message received by any given Topo (except for public channels), is acknowledged by an ACK0 or an ACK1. An ACK0 "packet" is a single letter Q; an ACK1 packet is a single letter R. These must alternate to insure valid transmissions. The MSB of the channel# being transmitted indicates whether Topo should send an ACK0 or ACK1 - 0 means ACK0, 1 means ACK1.

A valid ACK must be received by the H.D. within 40 ms of the end of the message being sent. If no ACK is received within that time, or a garbled transmission is detected, then a "saywhat?" command ( ASCII 0, or 30 hex) is given, telling the Topo to repeat its last ACK. Receiving the correct ACK means to continue normally (the original ACK was lost) while receiving the wrong ACK means the original message was lost and should be resent. The H.D. is required to keep track of the current ACK0/ACK1 status for each private channel.

Example: Suppose a message is sent and an ACK1 was expected but not received. After a saywhat?, an ACK0 means that the Topo never received the original message, so it should be re-sent. After the saywhat?, an ACK1 means that the original, expected ACK1 was sent but not received by the H.D., so we know the message has been received. A garbled or missing reply after a saywhat? should prompt another saywhat?. After 32 or more saywhat's (corresponding to a period of 2 seconds) without a reply, the appropriate QUERY error flag (Topo not responding) is set while the saywhat's continue. The error flag will only be reset after the message is eventually received, or by doing a RESET LINK.

When Topo sends a message to the H.D. ( generally a status requested by the Apple), the format is:
P (H.D. channel #), Q/R (appropriate ACK), d,d,d,d, cksum.

## IR Carrier

A carrier signal tells all Topos if they go out of range of the H.D. The H.D. is required to send out some command at least once every 300 msecs. If no command has been sent after 300 msecs has elapsed, the H.D. sends out a dummy message on channel A ( the default public channel) of A, nul, nul, nul, nul, nul, cksum. Nul is an ASCII 00. Any Topo which hears no valid commands for 2 seconds will PARK ( abort the current motion command and flush the motion command queue ).

Real-time Joystick

Since the packets are relatively short ( 7 characters at 2000 baud IR gives 35 ms per packet), this should work for real-time joystick response. Since short ACK's take only another 5 ms (plus computation time), it may well work with ACK's, too. The typical configuration would set the number of saywhat's after an error to zero, since we are better off simply sending the newest command, instead.

Apple/Topo Commands

This next section specifies all of the currently envisioned commands to be sent between the Apple and Topo. The H.D. should not need to know any of this. The final command formats will be included at the end. (dddd) represents four characters of data being transmitted. Numbers are twos complement unless otherwise specified.

Topo General Commands

Reset all - acts like a power-on reset. All parameters are set to their original values.

General status - returns data ( dddd) with the following :
3 char1 - queue depth (0-F)
2 char2 - queue space remaining (0-F)
1 char3 - bits7-4 -unimplemented additional switches
bits0-3 - head switches 0-3 after latching
0 char4 - bit7 - talking?
bit6 - moving?
bit5 - tone sounding?
bit4 - power on flag set?
bit3 - speech buffer full?
bit2 - head switch any?

read & clear?

Set power on flag (T/F) - sets flag to indicated value ( generally false). Automatically reset to true on power up, or after a reset all. Used to indicate to the Apple if a Topo has been inadvertantly reset.

Version # - returns the version number of Topo (1-F)

Communication Commands

Saywhat? - retransmit last reply, whether ACK, message, or whatever. Should only be sent by H.D.

Set public channel # P2,P3,or P4 (T/F) enable or disable the Topo to listen on the given channel. Has no effect on other channels.

### Head Switch Commands

Set head sw. auto-stop (T/F) - enable or disable an automatic motion stop after any head switch is pressed.

Latch head switch - latch head switch data for reading.

### Sound Generation Commands

Say (dddd) - send ( d,d,d,d ) to the ECHO module

Reset speech - send a reset signal to the ECHO module

Enable tone (pp ll) - activate.the Topo tone generator for the given pitch (00-FF) and length (00-FF intervals of 32 ms)

### Motion Commands

Park - cancel the currently executing motion command, and flush the motion command queue

Reset motion - warmstart, and reset all motion parameters to defaults

Set S (dddd) - assign dddd to the distance parameter S, measured in cm

Set A (dddd) - assign dddd to the angle parameter A, measured in degrees.

Set V (dddd) - assign dddd ( positive only ) to the target velocity parameter V, measured in cm/sec

Set ramp (dd—) - assign dd to the ramp rate parameter, measured in cm/sec/sec

Move - start a distance motion, using current values for S,A,V, and ramp parameters

Joy (dd1,dd2) -directly set wheel speeds, where dd1 controls forward velocity (S.) in cm/sec, and dd2 controls rotation rate (A.) in degrees/sec

Read current S - return the current elapsed value of S in cm

Read current A - return the current elapsed value of A in degrees

Read target S.A. - returns dd1=S. , dd2=A. from the velocity control sequencer. Intended for development purposes only

<u>Command Character Format</u>

The second character of a packet is the command character. Of its 8 bits, the MSB is always set (don't care), the next 3 bits is a device code valid for 2-7 (to ensure not having ASCII control codes), and the lower 4 bits are a 1 of 16 command for the given device. Valid devices are :

2 = Topo general
3 = Communication
4 = Motion
5 = Sound
6 = Head switch

The following table shows the current command matrix.

<u>COMMAND CHARACTER MATRIX</u>
<u>Devices</u>

| <u>Command</u> | | 2 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|---|
| <u>Types</u> | | Topo | Comm | Motion | Sound | Head Sw |
| Status1 | 0 | STATUS | SAYWHAT? | Current S | | |
| Status2 | 1 | Version# | | Current A | | |
| Status3 | 2 | | | Current S.A. | | |
| Status4 | 3 | | | | | |
| DO1 | 4 | Say Model# | | MOVE | | LATCH |
| DO2 | 5 | | | | | |
| DO3 | 6 | | | PARK | | |
| RESET | 7 | RESET ALL | | RESETmot. | RESETspeech | |
| SET1 | 8 | Pow.FlagT/F | | S (dddd) | | AutoStpT/F |
| SET2 | 9 | | CH#P2(T/F) | A (dddd) | | |
| SET3 | A | | CH#P3(T/F) | V (dddd) | | |
| SET4 | B | | CH#P4(T/F) | Ramp(dddd) | | |
| Set&DO1 | C | | | JOY(ssdd) | SAY(dddd) | |
| Set&DO2 | D | | | | TONE(ppll) | |
| Set&DO3 | E | | | | | |
| Set&DO4 | F | | | | | |