

**DN-500CB**

**Serial Command Protocol Guide**

Ver. 1.00

October 13, 2016

Denon Professional

inMusic Brands, Inc.

Contents

[2 General 3](#_Toc464115733)

[2.1 RS-232C Control 3](#_Toc464115734)

[3 Specification 4](#_Toc464115735)

[3.1 RS-232C Control 4](#_Toc464115736)

[4 Communication Protocol 5](#_Toc464115737)

[4.1 Packet Structure 5](#_Toc464115738)

[4.2 ACK (Acknowledge) and NACK (Not Acknowledge) 5](#_Toc464115739)

[4.2.1 ACK [Acknowledgement]: 5](#_Toc464115740)

[4.2.2 NACK [Negative Acknowledgement]: 6](#_Toc464115741)

[4.3 Communication Rules 7](#_Toc464115742)

[4.3.1 Initiation of Communication 7](#_Toc464115743)

[4.3.2 Communication Sequence 7](#_Toc464115744)

[4.3.3 Timeout 7](#_Toc464115745)

[4.3.4 ACK 8](#_Toc464115746)

[4.3.5 NACK 8](#_Toc464115747)

[4.3.6 Command Interval time 9](#_Toc464115748)

[4.4 Basic Control Flow 10](#_Toc464115749)

[4.4.1 Device Control Flow 10](#_Toc464115750)

[4.4.2 Status Request Flow 11](#_Toc464115751)

[4.4.3 Status Notification Flow 12](#_Toc464115752)

[5 Command Table 0](#_Toc464115753)

[5.1 Control Command/ Status Request Command List 0](#_Toc464115754)

[5.1.1 Key Control 0](#_Toc464115755)

[5.1.2 Current Status Information 2](#_Toc464115756)

[6 Appendix 4](#_Toc464115757)

[6.1 Acceptable Character 4](#_Toc464115758)

[6.1.1 Acceptable Character Type1 4](#_Toc464115759)

# 

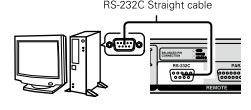
# General

The Serial Remote control function is used by connecting the host machine such as PC to the device (our product). The host can control the device by sending the Control command. It can determine the current device status by sending the Status Request command. In this document, the controlled equipment is called the “Device” (our pro duct) and the controlling equipment is called the “Host (PC etc.)”.

## RS-232C Control

The host can control the device by connecting the RS-232C straight cable to the D-sub 9 pins connector on the device back panel as shown in the following diagram.

Figure 2-1 RS-232C Connection



# Specification

## RS-232C Control

• Transmission type： Asynchronous / Full duplex

• Connector type： 9 pin D-sub female connector (Straight cable)

• Transfer rate： 9,600 / 38,400/ 115, 200 bps Selectable (via “Serial Bit Rate” in the “System Setting” menu).

• Clock accuracy： < +/- 2.0%

• Data length： 8 bits

• Parity： None

• Start bit： 1 bit

• Stop bit： 1 bit

• Flow Control： None

• Maximum data length：600 Bytes (Start character to End character is included.)

Figure3-1 Pin arrangement

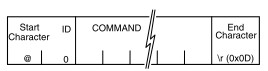
|  |  |
| --- | --- |
| Pin Number | Signal Name |
| 1 | GND |
| 6 | NC |
| 2 | TxD |
| 7 | RTS\* |
| 3 | RxD |
| 8 | NC |
| 4 | NC |
| 9 | NC |
| 5 | S. GND |

\*5V/500mA power supply can be used for RTS

# Communication Protocol

## Packet Structure

A packet must start with the Start Character ‘@’ and must be terminated with the End Character ‘\r’ (0x0D).

Figure 4-1Packet Structure

There are three kinds of packet, "COMMAND", "REQUEST", and "ANSWER /NOTIFICATION/ERROR".

## ACK (Acknowledge) and NACK (Not Acknowledge)

The device sends ACK or NACK to a host according to the following:

### ACK [Acknowledgement]:

It is an affirmative reply sent to a host from a device. When data transfer completes properly, the device responds and notifies the host.

### NACK [Negative Acknowledgement]:

It is a negative reply sent to a host from a device. When data transfer is not complete or it is not proper, the device respond and notifies the host.

The following table summarize the ACK and NACK messages

Table 4-1 ACK and NACK

|  |  |  |
| --- | --- | --- |
| Name | Value (HEX) | Transmission Requirement |
| ACK | 0x06 | The device acknowledged that the command was received normally from the host. |
| NACK | 0x15 | • The device received the End Character ‘\r’ (0x0D) before receiving the Start Character ‘@(0x40)’.  • The device did not receive ID’0(0x30)’ just after Start Character ’@(0x40)’  • The device received an unknown character just after Start Character ’@ (0x40)’ and ID’0 (0x30)’  • The device received an unknown command just after Start Character ’@ (0x40)’ and ID’0 (0x30)’.  • Parameter is out of range.  • The size of data is abnormal.  • 5msec passed before the device receives the next code necessary to complete the command. |

## Communication Rules

### Initiation of Communication

In the communication between a host and a device, the host must have the initiative. However, the Status Information notification automatically from the device is an exception.

When the device receives a command from the host, the device returns the following.

・When receiving the command which doesn’t require a status information ACK

・When receiving the command which requires a status information ACK + Status information (ANSWER)

・For the communication failure or an unknown commands etc.(Refer to Table 4-1): NACK

### Communication Sequence

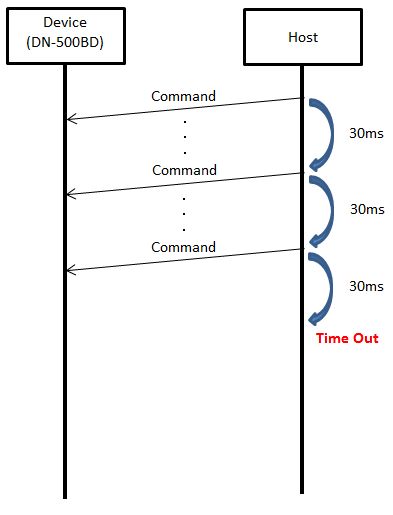
The host must not send any new command after the previous command before the host receives ACK or NACK, or the timeout (see”4-3-3

Timeout”) has expired.

### Timeout

After a host sends a message to a device, the host waits for a reply from the device for 300ms. When the host does not receive a reply over 300ms from the device, the host sends the same message to the device. However, when the host does not receive a reply from the device after sending the same message 3 times (that means Tim Out), the host sends End character ‘\r’ (0x0D) to the device. After that, the host should execute the recovery process such as retry.

Figure 4-2 Time Out



**(DN-500CD)**

Retry

Retry

### ACK

Refer to ” 4-2-1 ACK [Acknowledgement)”.

### NACK

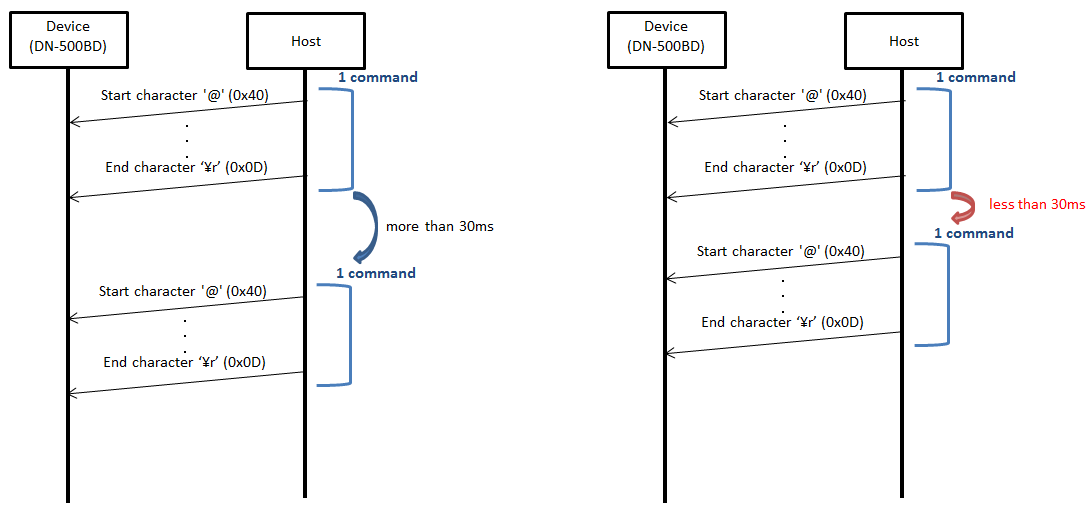
The device returns a NACK immediately after detecting a failure in the communication with the host (see（”4-2-2 NACK [Negative Acknowledgement)”). If the host receives a NACK, it must stop sending the current remaining command immediately and execute the recovery process such as retry.

### Command Interval time

・Interval time between Characters from a host must be less than 5ms. The device sends NACK when 5msec passed before the device receives the next character code.

・Interval time between Commands is more than 30ms.

Figure 4-3 Interval time between each command



**(DN-500CD)**

**(DN-500CD)**

In case of less than 300ms as interval time between commands,

1) Executing the subsequent command is not guaranteed.

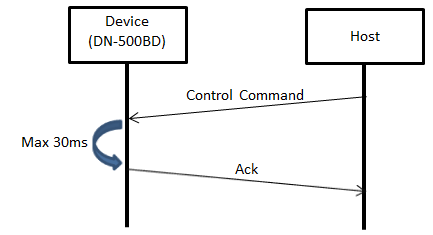
2) When there are buffer spaces of a device for a command, the device will execute the command.

3) When there is no buffer space of a device for a command, the device does not execute the command, and will send Busy (@0BDERBUSY) to the host instead.

## Basic Control Flow

### Device Control Flow

The device sends the host an ACK (Acknowledgement) and executes that command when the device receives a Control command from the host. The list of the Control command is shown in “Control Command List”.

Figure 4-4 Device Control Flow

**(DN-500CD)**

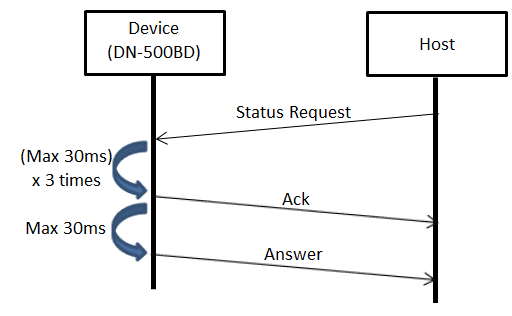
**MAX 300ms**

If the device receives an unknown command or an End Character ‘\r’ (0x0D) without a Start Character ‘@’, it causes transmission failure and the device returns a NACK (Not Acknowledgement). Refer to ”4-2-2 NACK [Negative Acknowledgement)” about NACK transmission condition.

### Status Request Flow

The device returns an ACK and the ANSWER requested by the host when the device receives the Status Request from the host. The list of the Status Request and the corresponding answer is shown in “Status Request List”.

Figure 4-5 Status Request Flow



**(DN-500CD)**

When a device receives Status Request from a host, the device sends an ACK to the host. After that, the device gets the current status, and then sends it to the host.

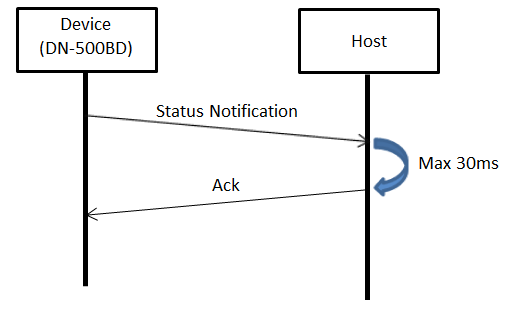
**MAX 300ms**

If the device receives an unknown command or an End Character ‘\r’ (0x0D) without a Start Character ‘@’, it causes transmission failure and the device returns a NACK (Not Acknowledgement). Refer to ”4-2-2 NACK [Negative Acknowledgement)” about NACK transmission condition.

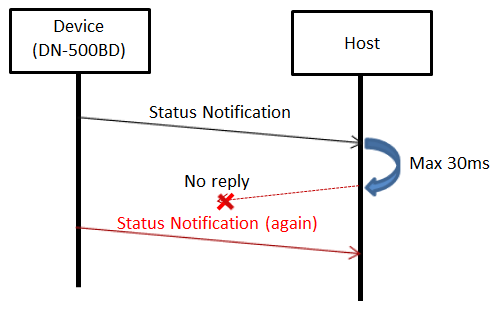
### Status Notification Flow

A device notifies of Status Information listed in “Status Information List” whenever the status of the device is changed, such as the transport status, the current track, the storage media status, and other status.

Figure 4-6 Status Notification Flow



**(DN-500CD)**

A device sends Status Notification to a host. The host replies ACK to the device. The device waits for the ACK for Max 300ms. When the device does not receive the ACK from the host, the device sends the same Status Notification to the host again. After that, the device does not send the same Status Notification even if it does not receive an ACK from the host.

**MAX 300ms**

**(DN-500CD)**

The Status Information is same as the answer for the Status Request listed in “Status Request List”.

# Command Table

*Italic* characters of command mean parameter. (Ex: Frame @0fr*XX* ‘\r -> Parameter: *XX*)

## Control Command/ Status Request Command List

### Key Control

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| # | Function | Command/Response | Status | Description |
|  | Power On | @0PW00 |  | Power On (Note-1) |
|  | Power Off | @0PW01 | - | Power Off (Note-1) |
|  | Stop | @02354 | - | Stop |
|  | Play | @02353 | - | Play |
|  | Play Pause | @02348 | - | Playback is Paused |
|  | Track | @0Tr*nnnn* | - | *nnnn*: Track No (‘0001’-‘2000’) |
|  | Track/Jump Next | @02332 | - | Track skip forward |
|  | Track/ Jump Prev | @02333 | - | Track skip reverse |
|  | Time Mode Code | @0PCTMD*XX* | @0?PCTMD | *XXXX: Time Mode Code* |
| ‘TL’: Total Elapsed, |
| ‘TR’: Total Remain, |
| ‘EL’: Elapsed, |
| ‘RM’: Remain, |
|  | Disc | @0PCDTRY*OP* | - |  |
| *XX*: Disc Tray Open/Close |
| ‘OP’: Empty |
| ‘CL’: Disc |
|  | Ten Key | @0PCTKEY*X* | - | Inputs 0-9 |
| *X*: Number |
| ‘1‘: 1, ‘2‘: 2, ‘3‘: 3, ‘4‘: 4, ‘5‘: 5, ‘6‘: 6, ‘7‘: 7, ‘8‘: 8, |
| ‘9‘: 9, ‘0‘:　0 |
|  | Slow/Search | @0PCSLS*d* | @0?PCSLS*d* | *d*: Direction |
| 'F' : Forward |
| ‘R’: Reverse |
|  |
|  | MUTE | @0mt*XX* | @0?mt | *XX*: On/Off |
| ‘00’: Mute on. |
| ‘01’: Mute off. |

Note-1: the power switch of the unit must be in the ON position.

### Current Status Information

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| # | Request | Command | Answer | Command | Description | Notification |
|  | Power Status | @0?PW | On |  | [See “Key Control”](#_Key_Control) | No |
| ACK |
| Off | No response |
|  | Media Status | @0?CD | No Disc | @0CDNC | There is not media | No |
| Disc In | @0CDCI | There is media. |
|  | Status | @0?ST | Play | @0STPL | [See “Key Control”](#_Key_Control) |
| Pause | @0STPP | [See “Key Control”](#_Key_Control) |
| Fast Play | @0STDVF*X* | Show scanning in process.  X: Direction  ‘R’ : Reverse, ‘F’: Forward |
|  | Total Track Number(4digit) | @0?Tt | Total Track Number | @0Tt*XXXX* | *XXXX*: Total Track  ‘0000’ to ‘9999’  ‘UNKN’: Unknown | No |
|  | Track Number | @0?Tr | Track Number | @0Tr*XXXX* | *XXXX*: Track No  ‘0000’ to ‘9999’  ‘UNKN’: Unknown | No |
|  | Elapse Time | @0?ET | Elapse Time | @0ET*hhhmmss* | *hhhmmss*: Time | No |
|  | Remain Time | @0?RM | Remain Time | @0RM*hhhmmss* | *hhhmmss*: Time | No |
|  | Current Track  Time | @0?tl | Current Track Time | @0tl*MMMSS* | *MMM*: Minute (‘000’-‘999’)  *SS*: Second (‘00’-‘59’) |  |
|  |
|  | Artist of Current Track | @0?at | Artist name | @0at*xxx* | *xxx*: Artist | *No* |
| (64 bytes max, \*Note-2) |
|  | Title of Current Track | @0?ti | Title | @0ti*xxx* | *xxx*: Title | *No* |
| (64 bytes max, \*Note-2) |
|  | Album of Current Track | @0?al | Album name | @0al*xxx* | *xxx*: Album | *No* |
| (64 bytes max, \*Note-2) |

(Note-2: Please refer to [6. Appendix](#_Appendix))

# Appendix

## Acceptable Character

The acceptable character set is ISO/IEC 8859-1.

### Acceptable Character Type1

* Acceptable characters are shown in Table 6‑1. (The characters that are colored in gray are not acceptable.)

Table 6‑1-1 Acceptable Character

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | X0 | X1 | X2 | X3 | X4 | X5 | X6 | X7 | X8 | X9 | XA | XB | XC | XD | XE | XF |
| 0X |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 1X |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 2X | *SP* | **!** | **“** | **#** | **$** | **%** | **&** | **‘** | **(** | **)** | **\*** | **+** | **,** | **-** | **.** | **/** |
| 3X | **0** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **:** | **;** | **<** | **=** | **>** | **?** |
| 4X | **@** | **A** | **B** | **C** | **D** | **E** | **F** | **G** | **H** | **I** | **J** | **K** | **L** | **M** | **N** | **O** |
| 5X | **P** | **Q** | **R** | **S** | **T** | **U** | **V** | **W** | **X** | **Y** | **Z** | **[** | **\** | **]** | **^** | **\_** |
| 6X | **`** | **a** | **B** | **c** | **D** | **e** | **f** | **g** | **h** | **I** | **j** | **k** | **l** | **m** | **n** | **o** |
| 7X | **p** | **q** | **R** | **s** | **T** | **u** | **v** | **w** | **x** | **Y** | **z** | **{** | **|** | **}** | **~** | *DEL* |
| 8X |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 9X |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| AX | *NBSP* | **¡** | **¢** | **£** | **¤** | **¥** | **¦** | **§** | **¨** | **©** | **ª** | **«** | **¬** | **­** | **®** | **¯** |
| BX | **°** | **±** | **²** | **³** | **´** | **µ** | **¶** | **·** | **¸** | **¹** | **º** | **»** | **¼** | **½** | **¾** | **¿** |
| CX | **À** | **Á** | **Â** | **Ã** | **Ä** | **Å** | **Æ** | **Ç** | **È** | **É** | **Ê** | **Ë** | **Ì** | **Í** | **Î** | **Ï** |
| DX | **Ð** | **Ñ** | **Ò** | **Ó** | **Ô** | **Õ** | **Ö** | **×** | **Ø** | **Ù** | **Ú** | **Û** | **Ü** | **Ý** | **Þ** | **ß** |
| EX | **à** | **á** | **â** | **ã** | **Ä** | **å** | **æ** | **ç** | **è** | **É** | **ê** | **ë** | **ì** | **í** | **î** | **ï** |
| FX | **ð** | **ñ** | **ò** | **ó** | **Ô** | **õ** | **ö** | **÷** | **ø** | **Ù** | **ú** | **û** | **ü** | **ý** | **þ** | **ÿ** |