『Maple Bus 1.0』Peripheral Hardware Specifications

105 Key Keyboard for Italy

Revision 1.11

Issued by
Sega Enterprises, Inc.
CS Development and Production Dep't, Second Division

Revision :

0.50	1999/05/17	First release
1.00	1999/06/07	Added Key scan code
1.10	1999/06/21	Changed the specifications for the device ID keyboard language from UK to Italy
1.11	1999/06/24	Modified the position of [€] (the Euro) appearing in Fig. 4.1, “Keyboard layout,” and Fig. 4.3, “Keyboard List,” and deleted

Note:

Windows 95 does not support the new EU currency unit in Euro font.

In Windows 95, characters are displayed differently. (This has been confirmed for versions A, B, and C of Windows 95. Later versions are unverified.)

Windows NT 4.0 Workstation SP4 and Windows 98 are supported.

An update program must be downloaded from the Microsoft site in order for this software to display normally in Windows 95.

CONTENTS

1 The Keyboard

1.1 Keyboard Functions

The Keyboard function is an input type man/machine interface serving for input and editing of characters and numerals, and for control of screen operations. This peripheral conforms to the "Maple Bus" Standard Specifications, belonging to function type " FT_6".

1.2 Outline of Operation

(1) Key scan

Keys are scanned constantly and data on key status is kept up to date in order to enable good response to requests from the host on the status of keys.

(2) Optimization and conditions

a) When no keys are pressed, simultaneous depression of up to three keys can be detected.

b) If three or more keys are being pressed, up to three additional keys can be detected provided that they are combinations that do not result in sneak current. However, if sneak current is produced, and error code results for that key and all subsequent keys pressed.

c) If a seventh key is pressed while six keys are already being pressed under the conditions given in b) above, an error code results for all further keyboard functions.

d) If sneak current results when more than three keys are pressed simultaneously, an error code is sent for all further keyboard functions.

e) If two or more keys are pressed simultaneously while two or more other keys are already being pressed, an error code is sent for all further keyboard functions, even if the combination of keys pressed does not produce a sneak current.

(3) Restrictions

Key repeat (intermittent output of a key scan code when a key is held down) cannot be set by keyboard function.

When key repeat is required, the repeat rate and initial repeat delay must be set by the host in software.

1.3 Keyboard Language, Keyboard Type

The keyboard applicable to these specifications is “Italy AT 105 Key Keyboard.”

1.4 Function Outline

See Fig. 4-1, Keyboard layout.

2 Device ID

The device ID corresponds to the Maple Bus 1.0 Standard Specifications. The table below shows the memory image on the host.

2.1 Italy AT 105 Key Keyboard Device ID Configuration

The configuration uses 16 bytes (128 bit).

bit	7	6	5	4	3	2	1	0
1st Data	0	0	0	0	0	0	0	0
2nd Data	0	0	0	0	0	0	0	0
3rd Data	0	0	0	0	0	0	0	0
4th Data	0	1	0	0	0	0	0	0
5th Data	0	0	0	0	0	1	1	0
6th Data	0	0	0	0	0	1	1	0
7th Data	0	0	0	0	0	0	0	0
8th Data	1	0	0	0	0	0	0	0
9th Data	0	0	0	0	0	0	0	0
10th Data	0	0	0	0	0	0	0	0
11th Data	0	0	0	0	0	0	0	0
12th Data	0	0	0	0	0	0	0	0
13th Data	0	0	0	0	0	0	0	0
14th Data	0	0	0	0	0	0	0	0
15th Data	0	0	0	0	0	0	0	0
16th Data	0	0	0	0	0	0	0	0

Fig. 2-1 Device ID

FT : Peripheral function type
FD1 : Function definition block for 1st function
FD2 : Function definition block for 2nd function
FD3 : Function definition block for 3rd function

(1) FT_0 - FT_{31}: Function type
Indicates the function type implemented by the peripheral. There are a total of 32 function types.

(2) FD_{31} - FD_0: Function definition block
These blocks define the various elements that make up a function.

3 Data Format

This section describes the Keyboard function data format.
The notation uses the memory image on the host.

3.1 Write Format

The format for writing Keyboard function data is shown below. When the host sends write data with "Set Condition", the keyboard returns a "Device Reply" for normal end or an error code for abnormal end.

bit	7	6	5	4	3	2	1	0
1st Data	LD_7	LD_6	LD_5	LD_4	LD_3	LD_2	LD_1	LD_0
2nd Data	0	0	0	0	0	0	0	0
3rd Data	0	0	0	0	0	0	0	0
4th Data	0	0	0	0	0	0	0	0

Fig. 3-1 Write format

Write format description

1st : LD LED setting

2nd : W1 Reserved = 00h

3rd : W2 Reserved = 00h

4th : W3 Reserved = 00h

LED setting

LD	Mapping	Lit	Out
0	Num Lock	1	0
1	Caps Lock	1	0
2	Scroll Lock	1	0
3	Reserved	1	0
4	Reserved	1	0
5	Kana	1	0
6	Power	1	0
7	Shift	1	0

Fig. 3-2 LED information bit

When set to ON (lit), the state is maintained until set to OFF (out).

With keyboard specifications, LD_7-LD_0 is normally “00h” because all LEDs are unavailable.

3.2 Read Format

The format for reading data from the Keyboard function is shown below. When the host sends a "Get Condition", the keyboard returns the "Data Transfer".
The data format size is 8 bytes.

bit	7	6	5	4	3	2	1	0
1st Data	M_7	M_6	M_5	M_4	M_3	M_2	M_1	M_0
2nd Data	LD_7	LD_6	LD_5	LD_4	LD_3	LD_2	LD_1	LD_0
3rd Data	$KC1_7$	$KC1_6$	$KC1_5$	$KC1_4$	$KC1_3$	$KC1_2$	$KC1_1$	$KC1_0$
4th Data	$KC2_7$	$KC2_6$	$KC2_5$	$KC2_4$	$KC2_3$	$KC2_2$	$KC2_1$	$KC2_0$
5th Data	$KC3_7$	$KC3_6$	$KC3_5$	$KC3_4$	$KC3_3$	$KC3_2$	$KC3_1$	$KC3_0$
6th Data	$KC4_7$	$KC4_6$	$KC4_5$	$KC4_4$	$KC4_3$	$KC4_2$	$KC4_1$	$KC4_0$
7th Data	$KC5_7$	$KC5_6$	$KC5_5$	$KC5_4$	$KC5_3$	$KC5_2$	$KC5_1$	$KC5_0$
8th Data	$KC6_7$	$KC6_6$	$KC6_5$	$KC6_4$	$KC6_3$	$KC6_2$	$KC6_1$	$KC6_0$

Fig. 3-3 Read format

Read format description

1st : Change key bit
2nd : LED information
3rd : Key scan code array #1
4th : Key scan code array #2
5th : Key scan code array #3
6th : Key scan code array #4
7th : Key scan code array #5
8th : Key scan code array #6

Change key bit

M	Mapping	Down	Up
0	Left Control	1	0
1	Left Shift	1	0
2	Left Alt	1	0
3	S1	1	0
4	Right Control	1	0
5	Right Shift	1	0
6	Right Alt	1	0
7	S2	1	0

Fig. 3-4 Change key bit

Example: [Control] + [Alt]

Normal '00000000' (00h)

Left「Alt」Down '00000100' (04h)

Right「Control」Down '00010100' (14h)

LED information

LD	Mapping	Lit	Out	Default
0	Num Lock	1	0	0
1	Caps Lock	1	0	0
2	Scroll Lock	1	0	0
3	Reserved	1	0	0
4	Reserved	1	0	0
5	Kana	1	0	0
6	Power	1	0	0
7	Shift	1	0	0

Fig. 3-5 LED information bit

Key scan code array for simultaneous key depression

When a key on the keyboard is pressed, the key scan code (see "5. Key code") is placed in the array. The array holds up to six key scan codes. When seven or more keys are pressed, a rollover error is generated.

Key input	Change key	Array #1	Array #2	Array #3	Array #4	Array #5	Array #6	Description
No input	00h	00h	00h	00h	00h	00h	00h	
Changed key input Right [Alt] “down”	40h	00h	00h	00h	00h	00h	00h	Changed key only is updated
First key input [Q] “down”	40h	04h	00h	00h	00h	00h	00h	Array is left-aligned
Second key input [H] “down”	40h	04h	0Bh	00h	00h	00h	00h	
Third key input [I] “down”	40h	04h	0Bh	0Ch	00h	00h	00h	
Fourth key input [M] “down”	40h	04h	0Bh	0Ch	10h	00h	00h	
Change key input Left [Shift] “down”	42h	04h	0Bh	0Ch	10h	00h	00h	Changed key only is updated
Fifth key input [O] “down”	42h	04h	0Bh	0Ch	10h	12h	00h	
Sixth key input [T] “down”	42h	04h	0Bh	0Ch	10h	12h	17h	
Seventh key input [A] “down”	42h	01h	01h	01h	01h	01h	01h	Seventh key was pressed. Rollover error generated.
First key input released [Q] “up”	42h	0Bh	0Ch	10h	12h	17h	00h	Array is left-aligned
Second key input released [H] + [M] “up”	42h	0Ch	12h	17h	00h	00h	00h	
Changed key input released Right [Alt] “up”	02h	0Ch	12h	17h	00h	00h	00h	Changed key only is updated
First key input released [I] “up”	02h	12h	17h	00h	00h	00h	00h	
First key input released Second key input [O] “up” + [Y] “down”	02h	17h	1Ch	00h	00h	00h	00h	Array is aligned with priority to "up"
First key input released [T] “up”	02h	1Ch	00h	00h	00h	00h	00h	
Changed key input released Left [Shift] “up”	00h	1Ch	00h	00h	00h	00h	00h	Changed key only is updated
First key input released [Y] “up”	00h	00h	00h	00h	00h	00h	00h	No key pressed

Fig. 3-6 Operation example (no sneak current is produced)

Sneak current phenomenon

Depending on the key position and timing, pressing three or more keys can lead to a condition where the keyboard circuitry cannot respond accurately, producing a false key scan code output. This is called sneak current. When it occurs, the keyboard products a rollover error.

Key status	Change key	Array #1	Array #2	Array #3	Array #4	Array #5	Array #6	Description
Normal	00h	00h	00h	00h	00h	00h	00h	
Right [Alt] down	40h	00h	00h	00h	00h	00h	00h	Changed key only is updated
[Q] down	40h	04h	00h	00h	00h	00h	00h	Array is left-aligned
[W] down	40h	04h	1Dh	00h	00h	00h	00h	
[P] down	40h	04h	1Dh	01h	01h	01h	01h	Sneak current occurs. Rollover error is generated.

Fig. 3-7 Sneak current condition

4 Key Scan Codes

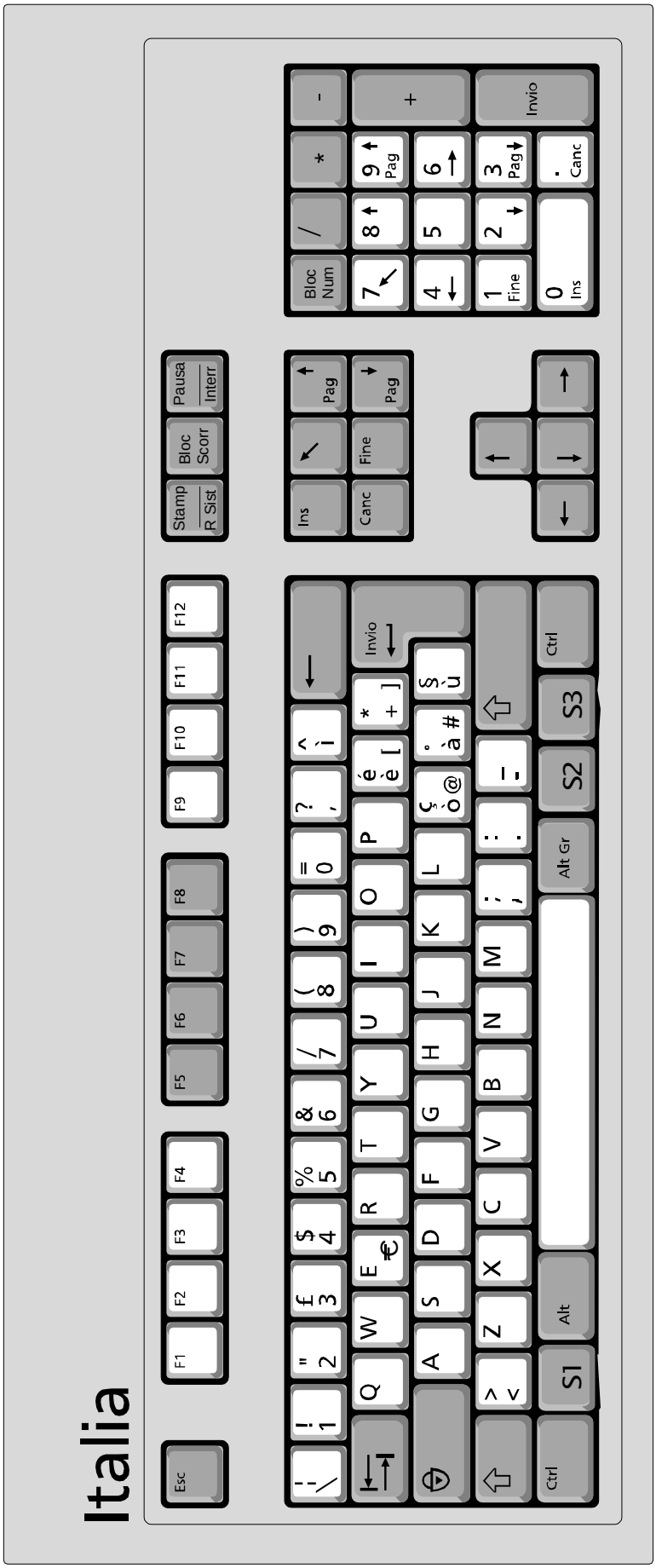

Fig. 4-1 Keyboard layout

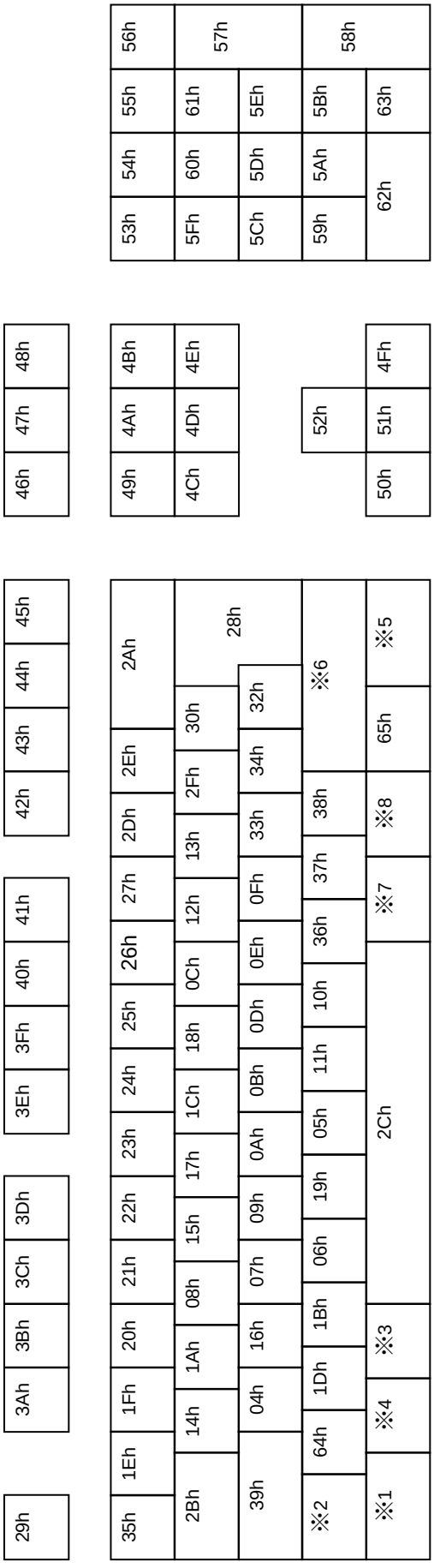

Fig. 4-2 Key scan code

code DEC	code HEX	Italian 105K	code DEC	code HEX	Italian 105K
0	00h	No Event Indicated	56	38h	Keyboard [-], [_]
1	01h	Error Roll Over	57	39h	Keyboard [Caps Lock]
4	04h	Keyboard [a], [A]	58	3Ah	Keyboard [F1]
5	05h	Keyboard [b], [B]	59	3Bh	Keyboard [F2]
6	06h	Keyboard [c], [C]	60	3Ch	Keyboard [F3]
7	07h	Keyboard [d], [D]	61	3Dh	Keyboard [F4]
8	08h	Keyboard [e], [E], [€](Euro)	62	3Eh	Keyboard [F5]
9	09h	Keyboard [f], [F]	63	3Fh	Keyboard [F6]
10	0Ah	Keyboard [g], [G]	64	40h	Keyboard [F7]
11	0Bh	Keyboard [h], [H]	65	41h	Keyboard [F8]
12	0Ch	Keyboard [i], [I]	66	42h	Keyboard [F9]
13	0Dh	Keyboard [j], [J]	67	43h	Keyboard [F10]
14	0Eh	Keyboard [k], [K]	68	44h	Keyboard [F11]
15	0Fh	Keyboard [l], [L]	69	45h	Keyboard [F12]
16	10h	Keyboard [m], [M]	70	46h	Keyboard [Stamp]
17	11h	Keyboard [n], [N]	71	47h	Keyboard [Bloc Scorr]
18	12h	Keyboard [o], [O]	72	48h	Keyboard [Pausa]
19	13h	Keyboard [p], [P]	73	49h	Keyboard [Ins]
20	14h	Keyboard [q], [Q]	74	4Ah	Keyboard [?]
21	15h	Keyboard [r], [R]	75	4Bh	Keyboard [Pag↑]
22	16h	Keyboard [s], [S]	76	4Ch	Keyboard [Canc]
23	17h	Keyboard [t], [T]	77	4Dh	Keyboard [Fine]
24	18h	Keyboard [u], [U]	78	4Eh	Keyboard [Pag↓]
25	19h	Keyboard [v], [V]	79	4Fh	Keyboard [→]
26	1Ah	Keyboard [w], [W]	80	50h	Keyboard [←]
27	1Bh	Keyboard [x], [X]	81	51h	Keyboard [↓]
28	1Ch	Keyboard [y], [Y]	82	52h	Keyboard [↑]
29	1Dh	Keyboard [z], [Z]	83	53h	Keypad [Broc Num]
30	1Eh	Keyboard [1], [!]	84	54h	Keypad [/]
31	1Fh	Keyboard [2], ["]	85	55h	Keypad [*]
32	20h	Keyboard [3], [£]	86	56h	Keypad [-]
33	21h	Keyboard [4], [$]	87	57h	Keypad [+]
34	22h	Keyboard [5], [%] , ~~[€](Euro)~~	88	58h	Keypad [Invio]
35	23h	Keyboard [6], [&]	89	59h	Keypad [1], [Fine]
36	24h	Keyboard [7], [/]	90	5Ah	Keypad [2], [↓]
37	25h	Keyboard [8], [(]	91	5Bh	Keypad [3], [Pag↓]
38	26h	Keyboard [9], [)]	92	5Ch	Keypad [4], [←]
39	27h	Keyboard [0], [=]	93	5Dh	Keypad [5]
40	28h	Keyboard [Invio](Enter)	94	5Eh	Keypad [6], [→]
41	29h	Keyboard [ESC]	95	5Fh	Keypad [7], [?]
42	2Ah	Keyboard [Delete](BS)	96	60h	Keypad [8], [↑]
43	2Bh	Keyboard [Tab]	97	61h	Keypad [9], [Pag↑]
44	2Ch	Keyboard [Spacebar]	98	62h	Keypad [0], [Ins]
45	2Dh	Keyboard ['], [?]	99	63h	Keypad [.], [Canc]
46	2Eh	Keyboard [ì], [^]	100	64h	keyboard [<], [>]
47	2Fh	Keyboard [è], [é], [[]	101	65h	keyboard [S3]
48	30h	Keyboard [+], [*], []]	*224	E0h	Keyboard L[Control]
49	31h	Not Use	*225	E1h	Keyboard L[Shift]
50	32h	keyboard [ù], [§]	*226	E2h	Keyboard L[Alt]
51	33h	Keyboard [ò], [ç], [@]	*227	E3h	Keyboard [S1]
52	34h	Keyboard [à], [°], [#]	*228	E4h	Keyboard R[Control]
53	35h	Keyboard [\], [|]	*229	E5h	Keyboard R[Shift]
54	36h	Keyboard [,], [;]	*230	E6h	Keyboard R[Alt]
55	37h	Keyboard [.], [:]	*231	E7h	Keyboard [S2]

Fig. 4-3 Keyboard List

The following key codes exist in addition to those shown in Fig. 4-2.

00h: No operation

01h: Rollover error

The 2nd data of the keys marked *1 (Left Control), *2 (Left Shift), *3 (Left Alt), *4 (S1), *5 (Right Control), *6 (Right Shift), *7 (Right Alt), and *8 (S2) is handled as the change key bit, so no key codes are returned for the 3rd to 8th data.

For a detailed explanation of character input methods, please refer to the software (IME) specifications.

5 Keyboard Function Information

This section describes device-specific information (device status). The device status must be stored in such a way that device status data cannot be changed or erased.

5.1 Type

Fixed Device Status

This refers to 112 bytes of device status information data with a fixed format, comprising required information. Correct connection and operation are only assured if all items are properly recorded.

Free Device Status

This refers to a maximum of 908 bytes of device-specific status information that can be allocated freely.

5.2 Fixed Device Status

The Fixed Device Status area must include all the items listed below.

(1) Device ID

Size : 16 bytes (00000040-06060080-00000000-00000000)

Description : Specify peripheral device ID.

(2) Country specification

Size : 1Byte (08h)

Description : Only available in the Europe region.

(3) Connection method

Size : 1byte(00h)

Description : Connects to the expansion device and records the direction of the expansion socket. Becomes “00h” when the expansion device is not connected to the keyboard.

(4) Model name

Size : 30 bytes

Description : “Keyboard” in hankaku characters. Remaining slots to be padded with spaces (20h).

(5) License

Size : 60 bytes

Description : “Produced By or Under License From SEGA ENTERPRISES,LTD.” in hankaku characters. Remaining slots to be padded with spaces (20h).

(6) Standby current consumption

Size : 2 bytes

Description : Indicates the current consumption of the unit in paused condition (During minimum power consumption), in 0.1 mA units (hexadecimal notation).
Contains 01-2Ch since power consumption for the 105 Key Keyboard for Italy w is 30 mA.

(7) Maximum current consumption

Size : 2Byte

Description : Indicates the maximum current consumption of the unit, in 0.1 mA units (hexadecimal notation).
Contains 01-90h since power consumption for the 105 Key Keyboard for Italy w is 40 mA

5.3 Free Device Status

The Free Device Status area can include information about developers, designers, and programmers or any other information. The host can obtain this information by issuing the "All Device Request" command. If it is to be used by an application, the data orering sequence must be taken into consideration.

"Version 1.010,1999/04/27,315-6211-AM ," in hankaku characters.

6 Remarks