

# SPECIFICATION

---

**MX83093**

**U+P Keyboard Encoder**

**Version 1.0**



## TABLE OF CONTENTS

<b>1. INTRODUCTION.....</b>	<b>3</b>
<b>2. FEATURE.....</b>	<b>3</b>
<b>3. APPLICATIONS.....</b>	<b>3</b>
<b>4 . PIN ASSIGNMENT.....</b>	<b>4</b>
<b>5. FUNCTION DESCRIPTION.....</b>	<b>4</b>
5.1 PS/2 Keyboard Function Description.....	5
5.1.1 Keyboard Buffer.....	5
5.1.2 Power-on Reset and Self Test.....	5
5.1.3 Low voltage reset (LVR).....	5
5.1.4 Keyboard Data Output.....	6
5.1.5 Keyboard Data Input.....	6
5.1.6 Command from the System.....	6
5.1.7 System Commands.....	9
5.1.8 Keyboard Scan Code Table.....	10
5.1.9 Turbo Function.....	18
5.2 USB Keyboard Function Description.....	18
5.2.1 USB Keyboard Description.....	19
5.2.2 LED Report for USB Keyboard.....	22
5.2.3 USB Keyboard Scan Code.....	23
<b>6. ABSOLUTE MAXIMUM RATINGS.....</b>	<b>257</b>
<b>7. DC CHARACTERISTICS.....</b>	<b>28</b>
<b>8. AC ELECTRICAL CHARACTERISTICS.....</b>	<b>28</b>
<b>9. PS/2 TIMING DIAGRAM.....</b>	<b>29</b>
<b>10. APPLICATION CIRCUIT.....</b>	<b>29</b>
<b>11. APPLICATION MATRIX.....</b>	<b>300</b>
<b>12. PAD DIAGRAM.....</b>	<b>31</b>
<b>13. REVISION HISTORY.....</b>	<b>32</b>

## 1. INTRODUCTION

The MX83093 is a single chip keyboard encoder, these devices detect the key press and release activity, sends the scan code to system, and accepts the command from system. The MX83093 function both in USB and PS/2 interface environment. For PS/2 interface, it supports IBM PC AT and all compatible machines. For USB interface, it supports the USB standard request version 1.1 as well as HID class request version 1.1.

The MX83093 is the most powerful USB+PS2 keyboard encoder. It is a crystal-less and low cost BOM solution with the minimal external components.

## 2. FEATURE

- Operating voltage 4.4V~5.5V
- Universal Serial Bus Specification Version 1.1
- USB Device Class Definition for Human Interface Device (HID), Firmware Specification Version 1.1
- Support scan code set 1, 2 and 3 for PS/2
- Support HID Usage ID for USB
- Support PC PS/2 and USB keyboard
- Support USB Sleep Mode
- 104/107 keys with multi-media or other special application keyboard encoder
- Compatible with Windows7, Windows Vista, WindowsXP, Windows2000, Windows ME, Windows98, MAC OS, Cent OS and Linux
- Support multi-media or other special application keyboard encoder
- Phantom key detects
- A watch dog timer with its own on-chip RC oscillator to prevent system stand still
- Built-in power on reset (POR)
- Power-up timer (PWRT)
- Oscillator Startup timer (OST)
- Built in low voltage reset (LVR)
- Low power CMOS device technology
- **Compatible Silver&carbon Membrane {Impedance Max(250kohm)}**

## 3. APPLICATIONS

- USB and PS/2 both compatible with Keyboard.
- IBM PC AT or compatible machine keyboard.
- IBM PS/2 model 30,50,60,80 or compatible keyboard.
- Apple PC USB keyboard.
- Japanese keyboard.
- Korean keyboard.

- Brazilian keyboard.

## 4 . PIN ASSIGNMENT

Symbol	I/O	Function Description
PA0~PA7	I/O	GPIO pins. These pins can be pulled-high internally through software control.
PB0~PB7	I/O	GPIO pins. These pins can be pulled-high internally through software control.
PD0~PD7	I/O	GPIO pins. These pins can be pulled-high internally through software control.
PE0~PE5	I/O	GPIO pins. These pins can be pulled-high internally through software control or LED sink pins.
PE6	I/O	GPIO pins.
PC0~PC2 PC6~PC7	I/O	LED sink pins
DP_CLK ( PC4 )	I/O	USB plus data line interface or CLK for PS/2 keyboard When the MX83093 is running under PS/2 mode, this pin will have an internal pulled-high resistor, with $V_{dd}=5.0V$
DM_DATA ( PC5 )	I/O	USB minus data line interface or DATA for PS/2 keyboard. When the MX83093 is running under PS/2 mode, this pin will have an internal pulled-high resistor, with $V_{dd}=5.0V$ When the MX83093 is running under USB mode, this pin will have an internal pulled-high resistor, 1.5k Ohm, with $V_{3.3}=3.3V$
OSCO	O	Return path for 6MHz crystal. The RC test clock
OSCI	I	6MHz crystal input.
VDD	PWR	Power supply pin.
GND	PWR	Ground pin.
V33	PWR	3.3V regulator output.

## 5. FUNCTION DESCRIPTION

This section is divided into subsections for user convenience and clearer distinction. Detailed PS/2 keyboard function descriptions are provided in Section 5.1. The USB keyboard function descriptions are discussed in details in Section 5.2.

## 5.1 PS/2 KEYBOARD FUNCTION DESCRIPTION

The keyboard buffer, power-on reset and self test, input, output, commands, and scan codes are described in the following sections.

### 5.1.1 KEYBOARD BUFFER

The keyboard will buffer 16 bytes in a first-in-first-out order when the system receives scan codes from the keyboard. The response codes and repeated codes will not be buffered. If the keystrokes generate a multiple-byte sequence, the entire sequence must fit into the buffer or the keystroke is discarded and a buffer-overflow condition occurs.

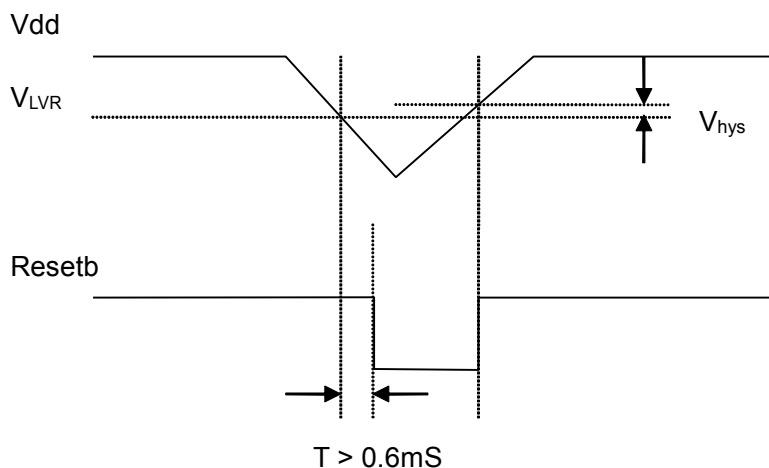
### 5.1.2 POWER-ON RESET AND SELF TEST

The duration of the keyboard Power-on-Reset (POR) should be within 150 milliseconds and 2 seconds after the power is applied to the keyboard.

After executing POR, the keyboard executes a self test. The LEDs are turned on at the beginning and off at the end of the self test. The self test takes a minimum of 300 milliseconds and a maximum of 500 milliseconds. If the self test is successful, a completion code AA hex is sent to the system and the keyboard starts scanning. If the self test fails, an error code is sent, the keyboard is disabled and waits for a command from the system. The completion codes are sent between 450 ms to 2.5 s after POR, and between 300 ms to 500 ms after a reset command is acknowledged.

### 5.1.3 LOW VOLTAGE RESET (LVR)

MX83093 has a low voltage reset function, which is used to monitor the voltage of power supply or external source. The low voltage reset voltage is 3.0V–3.2V.  $V_{hys} > 0.1V$ . Low voltage reset will not happen if the period  $V_{dd}$  stays in low voltage is less than  $T$  (0.6ms).



#### **5.1.4 KEYBOARD DATA OUTPUT**

When the keyboard is ready to send data to the system, it first checks the clock and data lines. If either one is in the low state, data is stored in the keyboard buffer. If both are in the high state, keyboard starts clocking data out. Data is valid before the trailing edge and after the leading edge of the clock pulse. During the transmission, the keyboard checks the clock line at least every 60 microseconds. If the system lowers the clock lines before the leading edge of the 10<sup>th</sup> clock, the keyboard should stop sending, then buffers the data and return the clock and data lines to a high state.

#### **5.1.5 KEYBOARD DATA INPUT**

When the system is ready to send data to the keyboard, it first checks the clock line to see if keyboard is sending data. If keyboard is not sending data or it is sending data but has not reached the 10<sup>th</sup> clock, the system can inhibit the interface by forcing the clock line low for more than 60 microseconds and prepares to send data. The keyboard checks the clock line status at least every 5 milliseconds. If a system Request to Send (RTS) is detected, the keyboard clocks 11 bits in. After the 10<sup>th</sup> bit, the keyboard checks for a high state in the data line then pulls it low and clocks one more bit to signal the system that data has been received. If data is low after the 10<sup>th</sup> bit, it indicates a frame error. The keyboard should continue to count until the data line goes high, then pulls it low and issues a RESEND to the system.

#### **5.1.6 COMMAND FROM THE SYSTEM**

The system can send commands to the keyboard at any time. The keyboard needs to respond within 20 milliseconds, unless the system prevents keyboard output, when doing self test or executing a reset. The following are the keyboard input commands and the actions that the keyboard needs to take.

##### **Set /Reset Status Indicators (Hex ED)**

-Responds with ACK –Receives option byte –Responds with ACK –Updates status indicators  
–Returns to previous scanning state

##### **Echo (Hex EEH)**

-Responds with EE hex –Returns to previous scanning state

##### **Invalid commands (Hex EF and F1)**

-Returns a RESEND command –Returns to previous scanning state

### Select Alternate Scan Codes (Hex F0)

-Responds ACK –Clears output buffer –Sets the default Typematic rate/delay –Clears last Typematic key –Receives option byte –Responds ACK –Option byte : 01 : selects scan code Set 1  
 02 : selects scan code Set 2

### Read ID (F2)

-Responds with ACK –Discontinues scanning –Sends two ID bytes. The second byte will be sent within 500 μS after the first byte. –Resumes scanning

### Set Typematic Rate/Delay (Hex F3)

-Responds with ACK –Receives rate/delay value byte –Responds with ACK –Set rate/delay (see Note 1) –Returns to previous scanning state.

Note:

- 1.Repeat period =  $(8+A) \times (2^B) \times 0.00417 \text{ s}$   
 A – Binary value of bits 2, 1, and 0  
 B – Binary value of bits 4 and 3
2. Delay =  $(C + 1) \times 250 \text{ ms}$   
 C – Binary value of bits 6 and 5.Bit 7 is always 0.
3. Default rate = 10.9 chars/sec + 20%  
 Default delay = 500ms +/- 20%

Bit 4 Bit 0	Typematic Rate ±20%	Bit 4 Bit 0	Typematic Rate ±20%
00000	30.0	10000	7.5
00001	26.7	10001	6.7
00010	24.0	10010	6.0
00011	21.8	10011	5.5
00100	20.0	10100	5.0
00101	18.5	10101	4.6
00110	17.1	10110	4.3
00111	16.0	10111	4.0
01000	15.0	11000	3.7

Bit 4 Bit 0	Typematic Rate ±20%	Bit 4 Bit 0	Typematic Rate ±20%
01001	13.3	11001	3.3
01010	12.0	11010	3.0
01011	10.9	11011	2.7
01100	10.0	11100	2.5
01101	9.2	11101	2.3
01110	8.6	11110	2.1
01111	8.0	11111	2.0

#### **Enable (Hex F4)**

-Responds with ACK –Clears output buffer –Clears last typematic key –Start scanning

#### **Default Disable (Hex F5)**

-Responds with ACK –Resets all conditions to power-on state –Clears output buffer –Sets the default key type (Scan Code Set 3 only) –Set the default typematic rate/delay –Clears the last typematic key –Stops scanning

#### **Set Default (Hex F6 )**

-Responds with ACK –Resets all conditions to power-on state –Clears output buffer –Sets the default key type (Scan Code Set 3 only) –Set the default typematic rate/delay –Clears last typematic key –Continues scanning

#### **Set All Keys (Hex F7, F8, F9, FA)**

-Responds ACK –Clears output buffer –Sets all key type (affect only Scan Code Set 3 operation)  
F7 : Typematic F8 : Make/Break F9 : Make FA : Typematic/Make/Break –Returns to previous scanning rate

#### **Set Key Type (Hex FB, FC, FD)**

-Responds with ACK –Clears output buffer –Receives key ID byte –Responds ACK –Sets key ID type (affect only Scan Code Set 3 operation)

FB : Typematic



FC : Make/Break FD : Make –Returns to previous scanning rate

#### **Resend (Hex FE)**

-Sends the previous output again. If the previous byte is RESEND, the last byte before RESEND shall be sent.

#### **Reset (Hex FF)**

-Responds with ACK-Checks clock and data lines. Reset in the high state for at least 500  $\mu$ s or receives another command.-Default to scan code set 2.

### **5.1.7 SYSTEM COMMANDS**

The following are the keyboard output commands.

#### **Acknowledge (Hex FA)**

The keyboard sends an acknowledge in response to any valid command from the system except for RESEND and ECHO.

#### **BAT Completion Code (Hex AA)**

This command is issued after a successful completion of the keyboard self test.

#### **Echo (Hex EE)**

The keyboard sends this code in response to an ECHO command.

#### **Keyboards ID (Hex 83AB)**

The 2-byte ID is issued to respond to READ ID command. The low byte is sent first, followed by the high byte.

#### **Keyboard Buffer Overrun (Hex 00 or FF)**

If the keyboard buffer overflows, the overrun code will replace the last byte in the buffer. If the keyboard is using scan code Set 1, the code is Hex FF. For Sets 2 and 3, the code is Hex 00.

#### **Resend (Hex FE)**

When the keyboard receives an invalid input or any input with incorrect parity, the RESEND

command is sent.

### 5.1.8 KEYBOARD SCAN CODE TABLE

The keyboard contains three scan code sets. The system defaults to scan code Set 2, but can be switched to Set 1 or Set 3. The following tables show the key numbers and three scan code sets in hexadecimal values. All the keys are typematics except for the pause key. When a key is pressed down, the make scan code is sent to the system. When the key is released, its break code is sent. If two or more keys are held down, only the last key pressed repeats at the typematic rate.

Typematic operation stops when the last pressed key is released, even if other keys are being released or some other keys are still held down.

The following table shows three scan code sets used in the keyboard.

#### Scan code set 1

Key	Make Code	Break Code	Key	Make Code	Break Code
`(~)	29	A9	X	2D	AD
1	02	82	C	2E	AE
2	03	83	V	2F	AF
3	04	84	B	30	B0
4	05	85	N	31	B1
5	06	86	M	32	B2
6	07	87	, (<)	33	B3
7	08	88	. (>)	34	B4
8	09	89	/ (?)	35	B5
9	0A	8A	SHIFT_R	36	B6
0	0B	8B	CTRL_L	1D	9D
-(-)	0C	8C	ALT_L	38	B8
=(+)	0D	8D	SPACE	39	B9
BACKSPACE	0E	8E	ALT_R	E038	E0B8
TAB	0F	8F	CTRL_R	E01D	E09D
Q	10	90	NUM	45	C5
W	11	91	(7)	47	C7
E	12	92	(4)	4B	CB
R	13	93	(1)	4F	CF
T	14	94	(8)	48	C8
Y	15	95	(5)	4C	CC



Key	Make Code	Break Code	Key	Make Code	Break Code
U	16	96	(2)	50	D0
I	17	97	(0)	52	D2
O	18	98	(*)	37	B7
P	19	99	(9)	49	C9
[{	1A	9A	(6)	4D	CD
]}	1B	9B	(3)	51	D1
\( ) (101 key keyboard only)	2B	AB	(DEL)	53	D3
CAPS	3A	BA	(-)	4A	CA
A	1E	9E	(+)	4E	CE
S	1F	9F	(ENTER)	E01C	E09C
D	20	A0	ESC	01	81
F	21	A1	F1	3B	BB
G	22	A2	F2	3C	BC
H	23	A3	F3	3D	BD
J	24	A4	F4	3E	BE
K	25	A5	F5	3F	BF
L	26	A6	F6	40	C0
;(,)	27	A7	F7	41	C1
'(")	28	A8	F8	42	C2
K42 (102 key keyboard only)	2B	AB	F9	43	C3
ENTER	1C	9C	F10	44	C4
SHIFT_L	2A	AA	F11	57	D7
K45 (102 key keyboard only)	56	D6	F12	58	D8
Z	2C	AC	SCROLL	46	C6



Key	Base Case, or Shift + Nun		Shift Case * Make/Break	Nun Lock Make/Break
	Lock	Make/Break		
INS		E0 52/E0 D2	E0 AA E0 52 /E0 D2 E0 2A	E0 2A E0 52 /E0 D2 E0 AA
DEL		E0 53/E0 D3	E0 AA E0 53 /E0 D3 E0 2A	E0 2A E0 53 /E0 D3 E0 AA
LEFT		E0 4B/E0 CB	E0 AA E0 4B /E0 CB E0 2A	E0 2A E0 4B /E0 CB E0 AA
HOME		E0 47/E0 C7	E0 AA E0 47 /E0 C7 E0 2A	E0 2A E0 47 /E0 C7 E0 AA
END		E0 4F/E0 CF	E0 AA E0 4F /E0 CF E0 2A	E0 2A E0 4F /E0 CF E0 AA
UP		E0 48/E0 C8	E0 AA E0 48 /E0 C8 E0 2A	E0 2A E0 48 /E0 C8 E0 AA
DOWN		E0 50/E0 D0	E0 AA E0 50 /E0 D0 E0 2A	E0 2A E0 50 /E0 D0 E0 AA
PAGE UP		E0 49/E0 C9	E0 AA E0 49 /E0 C9 E0 2A	E0 2A E0 49 /E0 C9 E0 AA
PAGE DOWN		E0 51/E0 D1	E0 AA E0 51 /E0 D1 E0 2A	E0 2A E0 51 /E0 D1 E0 AA
RIGHT		E0 4D/ED CD	E0 AA E0 4D /E0 CD E0 2A	E0 2A E0 4D /E0 CD E0 AA

Key (/)	Scan Code Make/Break E0 35 / E0 B5	Shift Case Make/Break E0 AA E0 35 / E0 B5 E0 2A
------------	---------------------------------------	--

Key PRINT	Scan Code Make/Break E0 2A E0 37 /E0 B7 E0 AA	Ctrl Case, Shift Case Make/Break E0 37 / E0 B7	Alt Case Make/Break 54 / D4
--------------	--	--	-----------------------------------

Key Pause (not	Make Code E1 1D 45 E1 9D C5	Ctrl Key Pressed E0 46 E0 C6
-------------------	--------------------------------	---------------------------------



typematic)		
------------	--	--

Key ¥	38 52 D2 4F CF 4D CD 4C CC B8	Num lock 45 C5 38 52 D2 4F CF 4D CD 4C CC B8 45 C5
----------	-------------------------------	--

Note \* : If the left Shift key is held down, the AA/2A shift make and break is sent with the other scan codes. If the right Shift key is held down, B6/36 is sent. If both Shift keys are down, both sets of codes are sent with the other scan code.

**Scan code set 2**

Key	Make code	Break code	Key	Make code	Break code
`(~)	0E	F00E	X	22	F022
1	16	F016	C	21	F021
2	1E	F01E	V	2A	F02A
3	26	F026	B	32	F032
4	25	F025	N	31	F031
5	2E	F02E	M	3A	F03A
6	36	F036	,(<)	41	F041
7	3D	F03D	,(>)	49	F049
8	3E	F03E	/(&?)	4A	F04A
9	46	F046	SHIFT_R	59	F059
0	45	F045	CTRL_L	14	F014
-(-)	4E	F04E	ALT_L	11	F011
=(+)	55	F055	SPACE	29	F029
BACKSPACE	66	F066	ALT_R	E011	E0F011
TAB	0D	F00D	CTRL_R	E014	E0F014
Q	15	F015	NUM	77	F077
W	1D	F01D	(7)	6C	F06C
E	24	F024	(4)	6B	F06B
R	2D	F02D	(1)	69	F069
T	2C	F02C	(8)	75	F075
Y	35	F035	(5)	73	F073
U	3C	F03C	(2)	72	F072
I	43	F043	(0)	70	F070



Key	Make code	Break code	Key	Make code	Break code
O	44	F044	(*)	7C	F07C
P	4D	F04D	(9)	7D	F07D
[{	54	F054	(6)	74	F074
]}	5B	F05B	(3)	7A	F07A
\( ) (101 key keyboard only)	5D	F05D	(DEL)	71	F071
CAPS	58	F058	(-)	7B	F07B
A	1C	F01C	(+)	79	F079
S	1B	F01B	(ENTER)	E05A	E0F05A
D	23	F023	ESC	76	F076
F	2B	F02B	F1	05	F005
G	34	F034	F2	06	F006
H	33	F033	F3	04	F004
J	3B	F03B	F4	0C	F00C
K	42	F042	F5	03	F003
L	4B	F04B	F6	0B	F00B
;(:	4C	F04C	F7	83	F083
'(")	52	F052	F8	0A	F00A
K42 (102 key keyboard only)	5D	F05D	F9	01	F001
ENTER	5A	F05A	F10	09	F009
SHIFFT_L	12	F012	F11	78	F078
K45 (102 key keyboard only)	61	F061	F12	07	F007
Z	1A	F01A	SCROLL	7E	F07E

Key	Base Case, or Shift + Nun Lock Make/Break	Shift Case * Make/Break	Num Lock Make/Break
-----	---	----------------------------	---------------------



Key	Base Case, or Shift + Nun Lock Make/Break	Shift Case * Make/Break	Num Lock Make/Break
INS	E0 70 / E0 F0 70	E0 F0 12 E0 70 /E0 F0 70 E0 12	E0 12 E0 70 /E0 F0 70 E0 F0 12
DEL	E0 71 / E0 F0 71	E0 F0 12 E0 71 /E0 F0 71 E0 12	E0 12 E0 71 /E0 F0 71 E0 F0 12
LEFT	E0 6B / E0 F0 6B	E0 F0 12 E0 6B /E0 F0 6B E0 12	E0 12 E0 6B /E0 F0 6B E0 F0 12
HOME	E0 6C /E0 F0 6C	E0 F0 12 E0 6C /E0 F0 6C E0 12	E0 12 E0 6C /E0 F0 6C E0 F0 12
END	E0 69 / E0 F0 69	E0 F0 12 E0 69 /E0 F0 69 E0 12	E0 12 E0 69 /E0 F0 69 E0 F0 12
UP	E0 75 / E0 F0 75	E0 F0 12 E0 75 /E0 F0 75 E0 12	E0 12 E0 75 /E0 F0 75 E0 F0 12
DOWN	E0 72 / E0 F0 72	E0 F0 12 E0 72 /E0 F0 72 E0 12	E0 12 E0 72 /E0 F0 72 E0 F0 12
PAGE UP	E0 7D / E0 F0 7D	E0 F0 12 E0 7D /E0 F0 7D E0 12	E0 12 E0 7D /E0 F0 7D E0 F0 12
PAGE DOWN	E0 7A / E0 F0 7A	E0 F0 12 E0 7A /E0 F0 7A E0 12	E0 12 E0 7A /E0 F0 7A E0 F0 12
RIGHT	E0 74 E0 F0 74	E0 F0 12 E0 74 /E0 F0 74 E0 12	E0 12 E0 74 /E0 F0 74 E0 F0 12

Key	Scan Code Make/Break	Shift Case Make/Break *
(/)	E0 4A / E0 F0 4A	E0 F0 12 E0 4A / E0 F0 4A E0 12

Key	Scan Code Make/Break	Ctrl Case, Shift Case Make/Break	Alt Case Make/Break
PRINT	E0 12 E0 7C /E0 F0 7C E0 F0 12	E0 7C / E0 F0 7C	84 / F0 84

Key	Make Code	Ctrl Key Pressed
PAUSE	E1 14 77 E1 F0 14 F0 77	E0 7E E0 F0 7E

(Not typematic)		
-----------------	--	--

Key ¥	11 70 F0 70 69 F0 69 74 F0 74	Num lock 77 F0 77 11 70 F0 70 69 F0 69 74 F0 74
	73 F0 73 F0 11	73 F0 73 F0 11 77 F0 77

Note \* : If the left Shift key is held down, the F0 12/12 shift make and break is sent with the other scan codes. If the right Shift key is held down, F0 59/59 is sent. If both Shift keys are down, both sets of codes are sent with the other scan code.

**New key codes for scan set 1:**

New key	Make	Break
LWIN	E05B	E0DB
RWIN	E05C	E0DC
APP	E05D	E0DD
N-CHG (131)	7B	FB
CHG (132)	79	F9
ROMA (133)	70	F0
K14	7D	FD
K56	73	F3
K107	7E	FE
KL	F1	X
KR	F0	X
POWER	E05E	E0DE
SLEEP	E05F	E0DF
WAKE UP	E063	E0E3

**New key codes for scan set 2:**

New key	Make	Break
LWIN	E0 1F	E0 F0 1F
RWIN	E0 27	E0 F0 27
APP	E0 2F	E0 F0 2F
N-CHG (131)	67	F0 67
CHG (132)	64	F0 64





New key	Make	Break
ROMA (133)	13	F0 13
K14	6A	F0 6A
K56	51	F0 51
K107	6D	F0 6D
KL	F1	X
KR	F2	X
POWER	E0 37	E0 F0 37
SLEEP	E0 3F	E0 F0 3F
WAKE UP	E0 5E	E0 F0 5E

**Windows 2000 Multi-Media key Code.**

		SET1		SET2	
		MAKE	BREAK	MAKE	BREAK
K130	WWW Back	E06A	E0EA	E038	E0F038
K131	WWWForward	E069	E0E9	E030	E0F030
K132	WWWstop	E068	E0E8	E028	E0F028
K133	WWWRefresh	E067	E0E7	E020	E0F020
K134	WWWSearch	E065	E0E5	E010	E0F010
K135	WWWFavorites	E066	E0E6	E018	E0F018
K136	WWWWeb/Home	E032	E0B2	E03A	E0F03A
K137	Mail	E06C	E0EC	E048	E0F048
K138	Mute	E020	E0A0	E023	E0F023
K139	Volume-	E02E	E0AE	E021	E0F021
K140	Volume+	E030	E0B0	E032	E0F032
K141	Play/Pause	E022	E0A2	E034	E0F034
K142	Stop	E024	E0A4	E03B	E0F03B
K143	PreTrack	E010	E090	E015	E0F015
K144	Nexttrack	E019	E099	E04D	E0F04D
K145	Media Select	E06D	E0ED	E050	E0F050
K146	MyComputer	E06B	E0EB	E040	E0F040
K147	Calculator	E021	E0A1	E02B	E0F02B

### 5.1.9 TURBO FUNCTION

Fn+	F1	2.1hz	F6	30hz
	F2	4.2hz	F7	46hz
	F3	7.2hz	F11	keyLock
	F4	16.3hz		
	F5	21hz		

## 5.2 USB KEYBOARD FUNCTION DESCRIPTION

We describe the USB keyboard function descriptions and data transfer in the following section.

### 5.2.1 USB KEYBOARD DESCRIPTION

#### Device Descriptor:

Descriptor size: 0X12

Descriptor type: 0X01

USB Specification release number: 0X0110

Device class: 0X00 Device subclass:

0X00 Device Protocol: 0X00

Max package for Endpoint 0 0X08

Vendor ID: 0X1C4F

Product ID: 0X0002

Device Release number: 0X0104

Index of string descriptor describing manufacturer: 0X01

Index of string descriptor describing product: 0X02

Index of string descriptor describing the device's serial number: 0X00

Number of possible configurations: 0X01

#### Configuration Descriptor:

Descriptor size: 0X09

Descriptor type: 0X02

Total length of data return for this configuration: 0X003B

Number of interfaces: 0X02

Value used to address this configuration: 0X01

Index of string descriptor describing this configuration: 0X00

Configuration characteristic: 0XA0

Maximum power consumption: 0X31



### **Interface Descriptor (Interface 0)**

Descriptor size: 0X09  
Descriptor type: 0X04  
Number of this interface: 0X00  
Alternate setting: 0X00  
Number of endpoints used by this interface: 0X01  
Class code: 0X03  
Subclass code: 0X01  
Protocol code : 0X01  
Index of string descriptor describing this interface: 0X00

### **HID Descriptor**

Descriptor size: 0X09  
Descriptor type: 0X21  
HID Specification release number: 0X0110  
Country code: 0X00  
Number of class descriptors: 0X01  
Type of class descriptor: 0X22  
Length of report descriptor: 0X0036

### **Endpoint Descriptor**

Descriptor size: 0X07  
Descriptor type: 0X05  
Endpoint's status: 0X81  
Endpoint's attributes: 0X03  
Maximum package size of this endpoint: 0X08  
Interval for polling endpoint for data transfers: 0X000A

### **Report Descriptor (Endpoint 1)**

Using page (generic desktop): 0x05 0x01  
Using (keyboard): 0x09 0x06  
Collection (application): 0xa1 0x01  
Using page (Page#for LEDs): 0x05 0x08  
Using minimum (1): 0x19 0x01  
Using maximum (3): 0x29 0x03  
Logical minimum (0): 0x15 0x00  
Logical maximum (1): 0x25 0x01



Report size (1): 0x75 0x01  
Report count (3): 0x95 0x03  
Output (data, variable, absolute): 0x91 0x02  
Report count (5): 0x95 0x05  
Output (constant): 0x91 0x01  
Using page (key codes): 0x05 0x07  
Using minimum (224): 0x19 0xe0  
Using maximum (231): 0x29 0xe7  
Report count (8): 0x95 0x08  
Input (data, variable, absolute): 0x81 0x02  
Report size (8): 0x75 0x08  
Report count (1): 0x95 0x01  
Input (constant): 0x81 0x01  
Using minimum (0): 0x19 0x00  
Using maximum (91): 0x29 0x91  
Logical maximum (ff): 0x26 0xff 0x00  
Report count (6): 0x95 0x06 Input (data, array): 0x81 0x00  
End collection: 0xc0

### **Interface Descriptor (Interface 1)**

Descriptor size: 0x09  
Descriptor type: 0x04  
Number of this interface: 0x01  
Alternate setting: 0x00  
Number of endpoints used by this interface: 0x01  
Class code: 0x03  
Subclass code: 0x00  
Protocol code: 0x00  
Index of string descriptor describing this interface: 0x00

### **HID Descriptor**

Descriptor size: 0x09  
Descriptor type: 0x21  
HID Specification release number: 0x0110  
Country code: 0x00  
Number of class descriptors: 0x01  
Type of class descriptor: 0x22

Length of report descriptor: 0X0034

### **Endpoint Descriptor**

Descriptor size: 0X07

Descriptor type: 0X05

Endpoint's status: 0X82

Endpoint's attributes: 0X03

Maximum package size of this endpoint: 0X03

Interval for polling endpoint for data transfers: 0X000A

### **Report Descriptor (Endpoint 2)**

Using page (consumer): 0x05 0x0c

Using (consumer): 0x09 0x01

Collection (application): 0xa1 0x01

Report id (01): 0x85 0x01

Using minimum (00): 0x19 0x00

Using maximum (023c): 0x2a 0x3c 0x02

Logical minimum (0): 0x15 0x00

Logical maximum (023c): 0x26 0x3c 0x02

Report count (1): 0x95 0x01

Report size (10): 0x75 0x10

Input (data, array, absolute): 0x81 0x00

End collection: 0xc0

Using (system control): 0x09 0x80

Collection (application): 0xa1 0x01

Report id (2): 0x85 0x02

Using minimum (81): 0x19 0x81

Using maximum (83): 0x29 0x83

Logical minimum (0): 0x15 0x00

Logic maximum (1): 0x25 0x01

Report size (1): 0x75 0x01

Report count (3): 0x95 0x03

Input (data, variable, absolute): 0x81 0x02

Report count (5): 0x95 0x05

Input (constant): 0x81 0x01End collection: 0xc0

### 5.2.2 LED REPORT FOR USB KEYBOARD

When one of the LED buttons (Num, Caps, Scroll) is pressed, the host issues a Setup package with a Set\_Report request to Endpoint 0 and an Out package with 1 byte data, which indicates the LED status. The format of the 1 byte data is:

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Constant	Constant	Constant	Kana	Compose	Scroll Lock	Caps Lock	Num Lock

### 5.2.3 USB KEYBOARD SCAN CODE

When the keys of HID Using Page 07 are pressed, the scan codes will be sent using Endpoint 1. If the keys of HID Using Page 01 and 0C are pressed, the scan codes will be sent using Endpoint 2.

When the host issues an IN packet to Endpoint 1 to receive data from the device and the keys of HID Using Page 07 are pressed, the device return 8 bytes to the host. The 8 bytes as follows:

Byte 7	Byte 6	Byte 5	Byte 4	Byte 3	Byte 2	Byte 1	Byte 0
Key 6	Key 5	Key 4	Key 3	Key 2	Key 1	Reserved	Modifier

The modifier bytes are as follows:

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Right GUI	Right Alt	Right Shift	Right Ctrl	Left GUI	Left Alt	Left Shift	Left Ctrl

When the host issues an IN packet to Endpoint 2 to receive data from the device and the keys of HID Using Page 0C (or 01) are pressed, the device return 3 (or 2) bytes to the host. The 3 (or 2) bytes are as follows:

HID Using Page 0C:

<b>Byte 2</b>	<b>Byte 1</b>	<b>Byte 0</b>
Using ID 2nd Byte	Using ID 1st Byte	Report ID 0X01

HID Using Page 01:

<b>Byte 1</b>	<b>Byte 0</b>
Using ID	Report ID 0X02



**HID Using ID:**

Key	HID Using Page	HID Using ID	Key	HID Using Page	HID Using ID
~(`)	07	35	X	07	1B
1	07	1E	C	07	06
2	07	1F	V	07	19
3	07	20	B	07	05
4	07	21	N	07	11
5	07	22	M	07	10
6	07	23	,(<)	07	36
7	07	24	.(>)	07	37
8	07	25	/(!)	07	38
9	07	26	Shift_R	07	E5
0	07	27	Ctrl_L	07	E0
-(-)	07	2D	Alt_L	07	E2
=(+)	07	2E	Space	07	2C
Backspace	07	2A	Alt_R	07	E6
Tab	07	2B	Ctrl_R	07	E4
Q	07	14	Num	07	53
W	07	1A	(7)	07	5F
E	07	08	(4)	07	5C
R	07	15	(1)	07	59
T	07	17	(8)	07	60
Y	07	1C	(5)	07	5D
U	07	18	(2)	07	5A
I	07	0C	(0)	07	62
O	07	12	(*)	07	55
P	07	13	(9)	07	61
[({	07	2F	(6)	07	5E
]})	07	30	(3)	07	5B
\( )@1	07	31	(DEL)	07	63
CAPS	07	39	(-)	07	56
A	07	04	(+)	07	57
S	07	16	(Enter)	07	58
D	07	07	ESC	07	29



F	07	09	F1	07	3A
G	07	0A	F2	07	3B
H	07	0B	F3	07	3C
J	07	0D	F4	07	3D
K	07	0E	F5	07	3E
L	07	0F	F6	07	3F
;( )	07	33	F7	07	40
'( )	07	34	F8	07	41
K42	07	32	F9	07	42
Enter	07	28	F10	07	43
Shift_L	07	E1	F11	07	44
K45	07	64	F12	07	45
Z	07	1D	Scroll	07	47
Ins	07	49	(/)	07	54
Del	07	4C	Print	07	46
Left	07	50	Pause	07	48
Home	07	4A	Break (Pause+Ctrl)	07	48
Up	07	52	www Back	0C	0224
Down	07	51	www Forward	0C	0225
Page Up	07	4B	www stop	0C	0226
Page Down	07	4E	www Refresh	0C	0227
Right	07	4F	www Search	0C	0221
WIN_L	07	E3	www Favorite	0C	022A
WIN_R	07	E7	www Home	0C	0223
APP	07	65	Mail	0C	018A
N-CHG (131)	07	8B	Mute	0C	00E2
CHG (132)	07	8A	Volume +	0C	00E9
ROMA (133)	07	88	Volume	0C	00EA
K14	07	89	Play/Pause	0C	00CD
K56	07	87	Stop	0C	00B7
K107	07	85	PreTrack	0C	00B6
KL	07	91	Nexttrack	0C	00B5



KR	07	90	Media Select	0C	0183
Power	01	81	My Computer	0C	0194
Sleep	01	82	Calculator	0C	0192
Wake up	01	83			

### 5.3 TURBO FUNCTION (PE2), IF PE2(14 PIN)=0:

USB:

Fn+	F11	All Key Lock
-----	-----	--------------

### 5.4 SMALL KEYBOARD FOR 88 KEY (PE3) ,IF PE3 (15 PIN) =0:

**Only USB 88KEY** 功能，即将数字小键盘功能映射到字母区：

5.4.1 当 NUM 灯亮时，字母键转为数字键，FN+字母则仍为字母键；

5.4.2 当 NUM 灯灭时，FN+字母则为数字键功能，当独按字母键则仍为字母键。

同时，有以下 FN 功能：

Fn+	F1	F11
	F2	F12
	F3	Media
	F4	Pre Track
	F5	Play/Pause
	F6	Next track
	F7	Mute
	F8	Volume-
	F9	Volume+
	F10	Scroll Lock

Function Key Usage

Key Location	Status							
	Fn	Num Lock	Fn	Num Lock	Fn	Num Lock	Fn	Num Lock
	OFF	OFF	ON	OFF	OFF	ON	ON	ON
7	&7		Home		7		7	
8	*8		↑		8		8	
9	(9		PgUp		9		9	
0	)0		*		*		0	
U	U		←		4		U	
I	I				5		I	
O	O		→		6		O	
P	P		-		-		P	
J	J		End		1		J	
K	K		↓		2		K	
L	L		PgDn		3		L	
::	::		+		+		::	
M	M		Ins		0		M	
>.	>.		Del		.		>.	
?/	?/		/		/		?/	
Enter	Enter (L)		Enter(R)		Enter(R)		Enter(L)	
F1/F11	F1		F11		F1		F11	
F2/F12	F2		F12		F2		F12	
F10/ ScrLock	F10		ScrLock		F10		ScrLock	
↑/PgUp	↑		Page Up		↑		Page Up	
↓/PgDn	↓		Page Down		↓		Page Down	
←/Home	←		Home		←		Home	
→/End	→		End		→		End	

## 6. ABSOLUTE MAXIMUM RATINGS

Parameter	Sym.	Ratings
Supply Voltage	VCC	-0.5 ~ +6V
Input Voltage	VIN	-0.3V ~ +6V
Output Voltage	VOUT	-0.3V ~ +6V
Operating temperature	TOP	-20℃ ~ 70℃
Storage temperature	TS	-50℃ ~ 150℃

## 7. DC CHARACTERISTICS

(VDD = 5V, Temperature = 27°C)

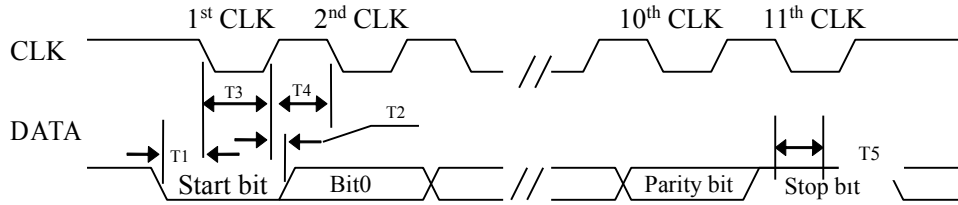
Parameter	Sym.	Min.	Typ.	Max.	Unit	Condition
Operating voltage	VCC	4.5	5	5.5	V	
Operating supply current	ICC	-	-	7	mA	Fosc=6MHz
Input leakage	IIN	-	-	2	μA	VIN=VCC,VSS
Input high voltage	VIH	3.0	-	-	V	PortA,B,D,E
Input low voltage	VIL	-	-	2.3	V	
Output low voltage	VOL1	-	-	0.4	V	IOL1=4.4mA
Output low voltage for LEDS	VOL2	-	-	3.2	V	IOL2=10mA
Internal Pull-high resistance (PORT B, E)	R <sub>PH1</sub>	8	-	16	KΩ	V <sub>PH1</sub> =0
Internal Pull-high resistance for DATA CLK(PS2)	R <sub>PH2</sub>	3.5	4.2	5	KΩ	V <sub>PH2</sub> =0
Internal Pull-high resistance for D-(USB)	R <sub>PH3</sub>	-	1.5	-	KΩ	V <sub>PH3</sub> =0
Output low voltage for DATA CLK	VOL3	-	-	0.4	V	IOL3=5mA

## 8. AC ELECTRICAL CHARACTERISTICS

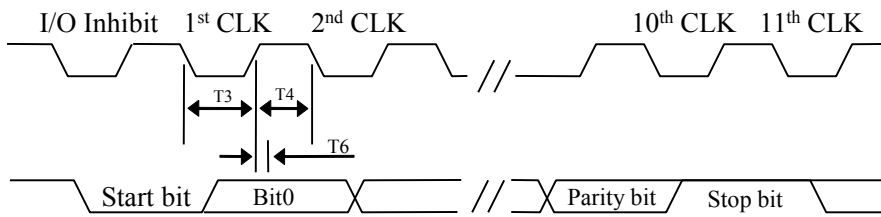
Parameter	Sym.	Min.	Typ.	Max.	Unit
Time from Data transition to falling edge of CLK	T1	5	-	25	us
Time from rising edge of CLK to Data transition	T2	5	-	25	us
Duration of CLK inactive	T3	30	40	50	us
Duration of CLK active	T4	30	40	50	us
Time for auxiliary device inhibit after Clock 11 to ensure the auxiliary device does not start another transmission	T5	-	-	50	us
Time from inactive to active CLK transition, used to time when auxiliary device samples Data	T6	5	-	25	us
System clock	FCLK	-	6	-	MHz

## 9. PS/2 TIMING DIAGRAM

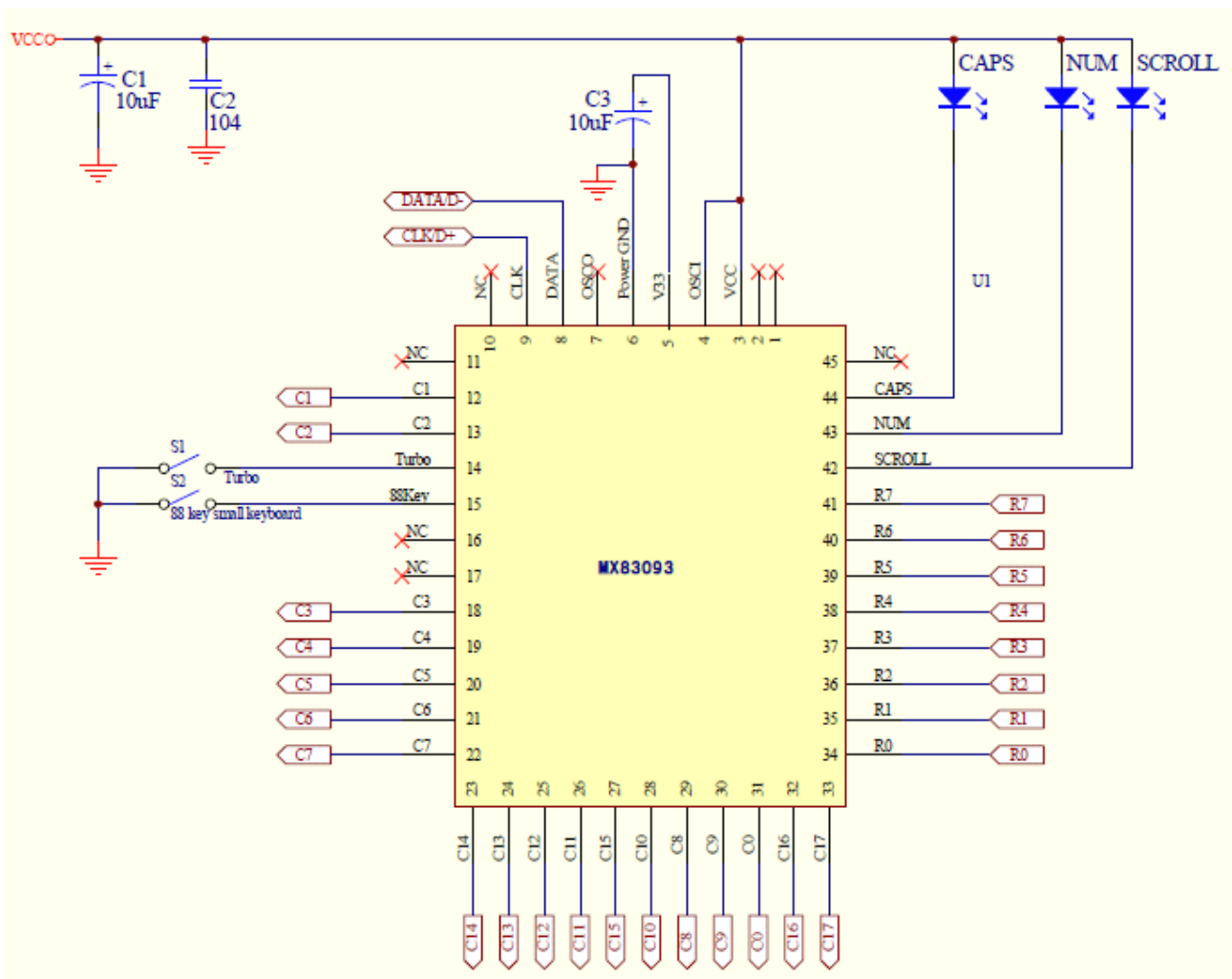
Data output



Data input



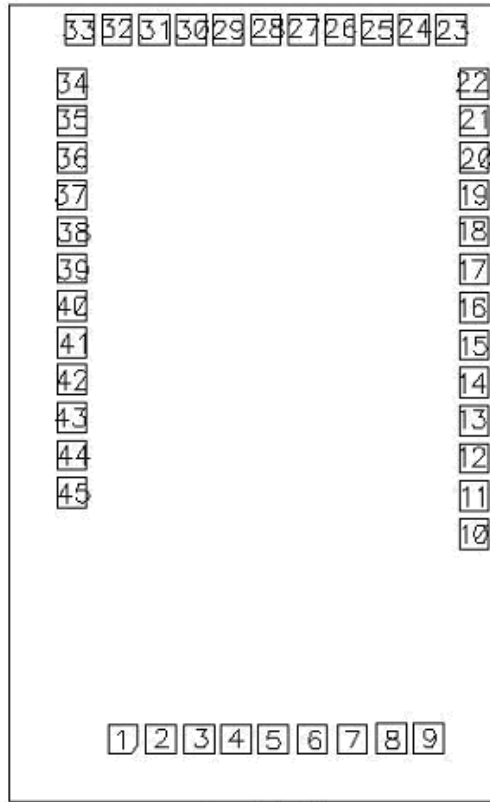
## 10. APPLICATION CIRCUIT



## 11. APPLICATION MATRIX

	<b>R0</b>	<b>R1</b>	<b>R2</b>	<b>R3</b>	<b>R4</b>	<b>R5</b>	<b>R6</b>	<b>R7</b>
<b>C0</b>	A	Z	WIN_L	,(<)	'(")	ROMA	WIN_R	*(K)
<b>C1</b>	PlayPause	MyComputer	N_CHG	9('')	_(-)	←	wwwBACK	↓
<b>C2</b>	Stop	WK_UP	SLEEP	0('')	F11	Mute	Refresh	Search
<b>C3</b>	Previous	V+	POWER	F7	F10	Mail	Forward	webHome
<b>C4</b>	Media	Calculator	F6	F8	F9	INS	wwwStop	Favorite
<b>C5</b>	V-	F3	F5	8(*)	PRINT	HOME	END	Next
<b>C6</b>	F1	F2	F4	I	F12	SCROLL	PAGE_DN	
<b>C7</b>	ESC	4(\$)	7(&)	U	+(&=)	DEL	4(K)	
<b>C8</b>	W	C	SPACE	.(>)	PrevTrack	→	/(K)	+(K)
<b>C9</b>	S	X	V	ALT_L	ALT_R	K42	NUM	ENTER(K)
<b>C10</b>	CAP	D	B	;(:	ENTER	↑	0(K)	.(K)
<b>C11</b>	TAB	E	H	P	[({	PAGE_UP	2(K)	3(K)
<b>C12</b>	`(~)	R	T	K	BACK	Volume-	5(K)	6(K)
<b>C13</b>	2(@)	3(#)	Y	J	K14	Volume+	7(K)	K107
<b>C14</b>	1(!)	5(%)	6(^)	O	\()	NextTrack	1(K)	
<b>C15</b>	Q	F	G	L	]})	PAUSE	8(K)	9(K)
<b>C16</b>	K45	CTR_L	N	M	\()	CHG	APP	CTR_R
<b>C17</b>	SHIFT_L	FN	K150	K151	/(?)	SHIFT_R	K56	-(K)

## 12. PAD DIAGRAM



Substrate Size:1800μm×2600μm

Substrate Connect GND

PIN NO	NAME	X	Y	PIN NO	NAME	X	Y
1	PC7	286.41	148.06	24	PA6	1070.01	2066.96
2	PC6	388.66	148.06	25	PA7	970.01	2066.96
3	vdd!	490.91	148.06	26	PD0	870.01	2066.96
4	OSCI	590.91	148.06	27	PD1	770.01	2066.96
5	V33	690.91	148.06	28	PD2	670.01	2066.96
6	gnd!	795.42	148.06	29	PD3	570.01	2066.96
7	OSCO	903.86	148.06	30	PD4	470.01	2066.96
8	PC5	1008.86	149.21	31	PD5	370.01	2066.96
9	PC4	1108.86	149.21	32	PD6	270.01	2066.96
10	PE6	1231.96	702.79	33	PD7	170.01	2066.96
11	PE7	1231.96	805.04	34	PB0	148.06	1920.79
12	PE0	1231.96	907.29	35	PB1	148.06	1820.79
13	PE1	1231.96	1009.54	36	PB2	148.06	1720.79



PIN NO	NAME	X	Y	PIN NO	NAME	X	Y
14	PE2	1231.96	1111.79	37	PB3	148.06	1620.79
15	PE3	1231.96	1214.04	38	PB4	148.06	1520.79
16	PE4	1231.96	1316.29	39	PB5	148.06	1420.79
17	PE5	1231.96	1418.54	40	PB6	148.06	1320.79
18	PA0	1231.96	1520.79	41	PB7	148.06	1220.79
19	PA1	1231.96	1620.79	42	PC2	148.06	1120.79
20	PA2	1231.96	1720.79	43	PC1	148.06	1018.54
21	PA3	1231.96	1820.79	44	PC0	148.06	916.29
22	PA4	1231.96	1920.79	45	RST_EXTb	148.06	814.04
23	PA5	1170.01	2066.96				

### 13. REVISION HISTORY

Version	Update date	Revised Content	Version by	Confirmed by
V1.0	2012-09-14		jacke	jacke