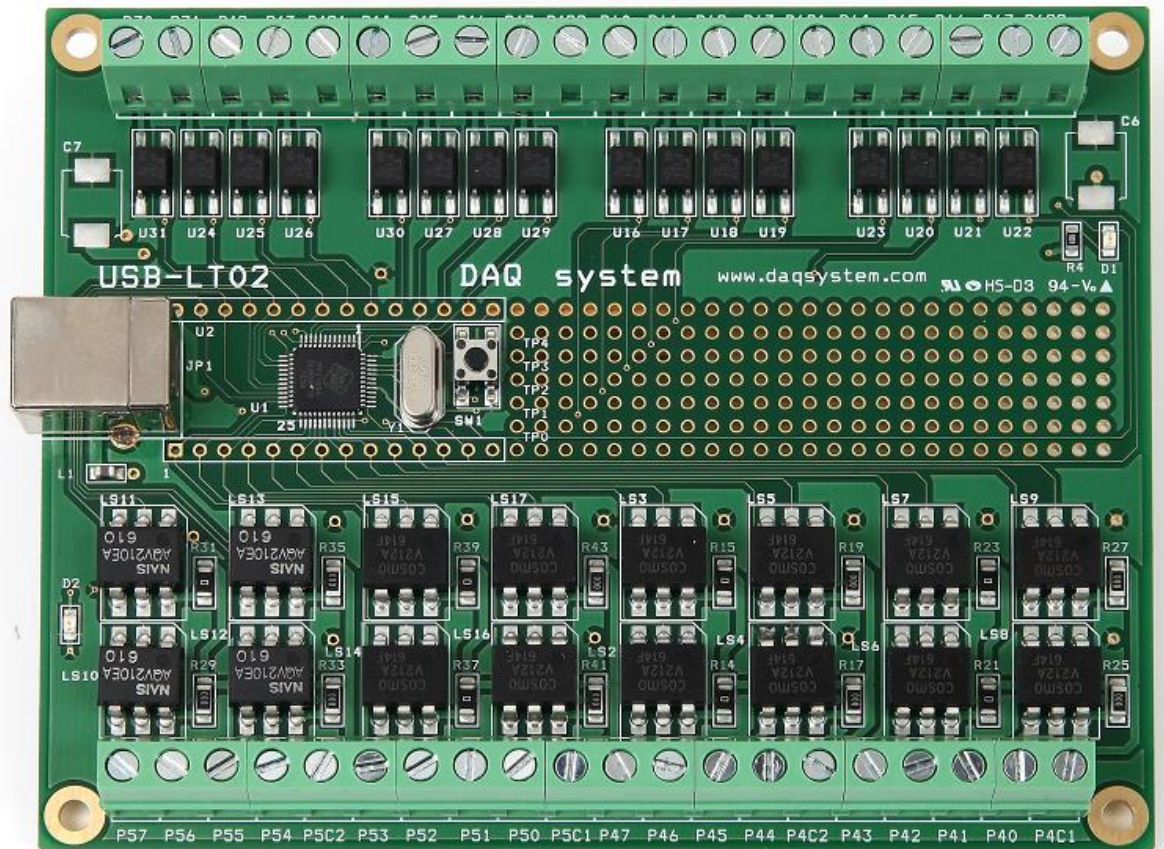


# USB-LT02

## API Manual

Version 1.0



© 2005 DAQ SYSTEM Co., Ltd. All rights reserved.

Microsoft® is a registered trademark; Windows®, Windows NT®, Windows XP®, Windows 7®, Windows 8®, Windows 10®  
All other trademarks or intellectual property mentioned herein belongs to their respective owners.

Information furnished by DAQ SYSTEM is believed to be accurate and reliable, However, no responsibility is assumed by DAQ SYSTEM for its use, nor for any infringements of patents or other rights of third parties which may result from its use. No license is granted by implication or otherwise under any patent or copyrights of DAQ SYSTEM.

The information in this document is subject to change without notice and no part of this document may be copied or reproduced without the prior written consent.

# Contents

## Board Level API Functions

InitBoard	-----	2
-----------	-------	---

## Port API Functions

PortConfigure	-----	3
PortRead	-----	5
PortWrite	-----	5
Port1Write	-----	6

## Board Level API Functions

### *Overview*

**Int                      InitBoard (void)**

### **InitBoard**

This function initializes the USB-LT02 board.

After initial power on, initialize the board before use.

**int                      InitBoard (void)**

**Parameters:**

**Return Value:**

If initialization fails, "0" is returned. If successful, "1" is returned.

## Port API Functions

### *Overview*

int	<b>PortConfigure</b> (WORD P4, WORD P5, WORD P6, WORD P7, WORD P1)
int	<b>PortRead</b> (BYTE *P4, BYTE *P5, BYTE *P6, BYTE *P7, BYTE *P1)
int	<b>PortWrite</b> (BYTE P4, BYTE P5, BYTE P6, BYTE P7)
int	<b>Port1Write</b> (BYTE act, BYTE pos)

## PortConfigure

This function decides whether to use each port as input or output.

int	<b>PortConfigure</b> (WORD P4, WORD P5, WORD P6, WORD P7, WORD P1)
-----	--

### **Parameters:**

Input/output is determined by each bit value of P4, P5, P6, P7, and P1. Below is a description with an example of the P4 value.

비트 위치

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
----	----	----	----	----	----	---	---	---	---	---	---	---	---	---	---

## Port4.7

15/14

0	0	Schmitt trigger input
0	1	Schmitt trigger input with pull-up
1	0	N-ch open drain output
1	1	N-ch open drain output with pull-up

## Port4.6

13/12

0	0	Schmitt trigger input
0	1	Schmitt trigger input with pull-up
1	0	N-ch open drain output
1	1	N-ch open drain output with pull-up



## Port4.1

3/2

0	0	Schmitt trigger input
0	1	Schmitt trigger input with pull-up
1	0	N-ch open drain output
1	1	N-ch open drain output with pull-up

## Port4.0

1/0

0	0	Schmitt trigger input
0	1	Schmitt trigger input with pull-up
1	0	N-ch open drain output
1	1	N-ch open drain output with pull-up

P4, P5, P6, and P7 have the same format. For reference, only ports 0 and 1 are available for P7. Therefore, only the values of bits 0,1,2,3 are meaningful.

The case of P1 is as follows.

Bit Position

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
----	----	----	----	----	----	---	---	---	---	---	---	---	---	---	---

Port1.7 – Port1.0

7 -- 0

0	Pull-up disable
1	Pull-up enable

In case of P1, it is processed the same as standard 8051 Port1 and there is no need to specify input/output separately, but it is decided whether to use internal pull-up resistor. In parameter P1, only bits 7 to 0 are meaningful, and bits 15 to 8 are meaningless.

**Return Value:**

If initialization fails, "0" is returned. If successful, "1" is returned.

## PortRead

This function returns the current input state of each port.

**int                    PortRead (BYTE \*P4, BYTE \*P5, BYTE \*P6, BYTE \*P7, BYTE \*P1)**

**Parameters:**

Pointer that can receive the value of each port (\*P4, \*P5, \*P6, \*P7, \*P1)

**Return Value:**

If the read fails, "0" is returned. In the case of success, "1" is returned.

## PortWrite

This function outputs each port as a specified value (except for port 1).

**int                    PortWrite (BYTE P4, BYTE P5, BYTE P6, BYTE P7)**

**Parameters:**

Port value to be output (P4, P5, P6, P7)

**Return Value:**

If writing fails, "0" is returned. In case of success, "1" is returned.

## Port1Write

This function outputs each bit of port 1 as a specified value. Port 1's 0 and 1 are used as USB data lines, so the entire value cannot be written, and each bit (except bits 0 and 1) must be written separately.

**int                    Port1Write (BYTE act, BYTE pos)**

**Parameters:**

act : If it is '0', '0' is recorded, and if it is not '0', '1' is recorded.

pos : indicates the bit position (bit positions 2 to 7). The rest of the values are meaningless.

**Return Value:**

If the command fails, "0" is returned. In case of success, "1" is returned.

# Memo

## Contact Point

Web sit : <https://www.daqsysteem.com>

Email : [postmaster@daqsysteem.com](mailto:postmaster@daqsysteem.com)

