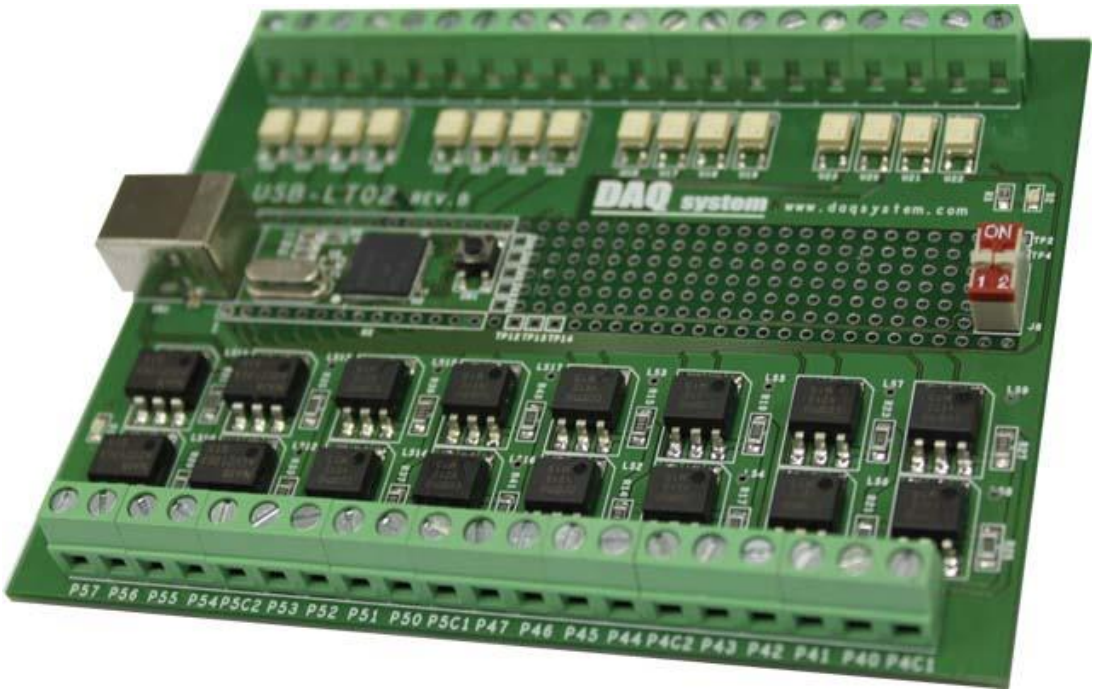
USB-LT02_B

API Manual

Version 1.3



ⓒ **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.

DAQ SYSTEM
Measurement & Automation

Contents

Board Level API Functions

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

Port API Functions

PortConfigure ---------- 3
PortRead ---------- 4
PortWrite ---------- 4
Port1Write ---------- 4
PortConfigureEx ---------- 5
PortReadEx ---------- 5
PortWriteEx ---------- 6
Port1WriteEx ---------- 6

Board Level API Functions

Overview

Int **InitBoard (void))**

InitBoard

This function returns the number of recognized devices.

int **InitBoard (void)**

Parameters:

Return Value:

Number of recognized devices.

Port API Functions

Overview

int **PortConfigure (WORD P4, WORD P5, WORD P6, WORD P7, WORD P1)**

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

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

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

int **PortConfigureEx (int nDevice, WORD P4, WORD P5, WORD P6, WORD P7, WORD P1)**

Int **PortReadEx (int nDevice, BYTE *P4, BYTE *P5, BYTE *P6, BYTE *P7, BYTE *P1)**

Int **PortWriteEx (int nDevice, BYTE P4, BYTE P5, BYTE P6, BYTE P7)**

int **Port1WriteEx (int nDevice, BYTE act, BYTE pos)**

PortConfigure

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

int **PortConfigure (WORD P4, WORD P5, WORD P6, WORD P7, WORD P1)**

Parameters:

P4: Delivers 16-bit values of upper Port 5 (7..0) and lower Port 4 (7..0).

P5 : Higher Port 1 (7..2), P7 (1..0) and lower Port 6 (7..0) 16-bit values are transmitted.

P6, P7, P1 : Not used.

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.

PortConfigureEx

This function decides whether to use each port as input or output. (For Multi-board)

int **PortConfigureEx (int nDevice, WORD P4, WORD P5, WORD P6, WORD P7, WORD P1)**

Parameters:

nDevice : Number of USB-LT02 devices detected

P4: Delivers 16-bit values of upper Port 5 (7..0) and lower Port 4 (7..0).

P5 : Higher Port 1 (7..2), P7 (1..0) and lower Port 6 (7..0) 16-bit values are transmitted.

P6, P7, P1 : Not used.

Return Value:

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

PortReadEx

This function returns the current input state of each port. (For Multi-board)

int **PortReadEx (int nDevice, BYTE *P4, BYTE *P5, BYTE *P6, BYTE *P7, BYTE *P1)**

Parameters:

nDevice : Number of USB-LT02 devices detected

*P4, *P5, *P6, *P7, *P1 : A pointer to the value of each port.

Return Value:

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

PortWriteEx

This function outputs each port as a specified value (except for port 1). (For Multi-board)

int PortWriteEx (int nDevice, BYTE P4, BYTE P5, BYTE P6, BYTE P7)

Parameters:

nDevice : Number of USB-LT02 devices detected

P4, P5, P6, P7 : The value of the port to output.

Return Value:

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

Port1WriteEx

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. (For Multi-board)

int Port1WriteEx (int nDevice, BYTE act, BYTE pos)

Parameters:

nDevice : Number of USB-LT02 devices detected

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.daqsystem.com

Email : postmaster@daqsystem.com

DAQ SYSTEM
Measurement & Automation