LabWedge

Version 6.0K
6/2009 R1

LabWedge Manual
'Basic'

IMPORTANT NOTES:

PROWEDGE.CFG FILE: SAVE a copy of the original ProWedge.cfg file as ProWedge.cfg_bak so you can easily recover / go back to an original working configuration if needed. This file is found in C:\

PASSWORD: The default password is… **www.prowedge.com**

CHANGING PASSWORD: You may change this password in the configuration options.

ENDING PROGRAM: Type **END** in the password field.

MOVING DISPLAY: You can position the display anywhere you like on the screen… simply hold Right mouse button and drag it to a new position.

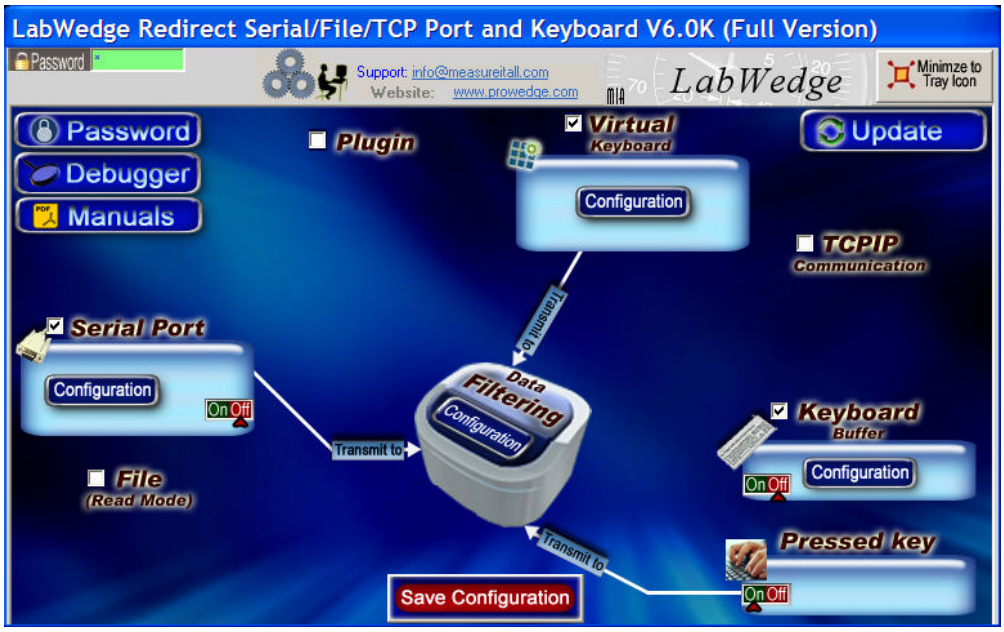
READ: Press F8, Button (image), or Start Button.

AUTOMATIC MODE: Set "Quantity" to 9999 for infinite number of readings.

For your convenience, we have highlighted some of the most relevant material in the manual.

SET-UP / START SCREEN:

Plug-in…	Unchecked
Virtual Keyboard…	Checked
TCPIP…	Unchecked
Keyboard Buffer…	Checked and 'OFF'
Pressed Key…	'ON'
File Read Mode…	Unchecked
Serial Port…	Checked and 'OFF'

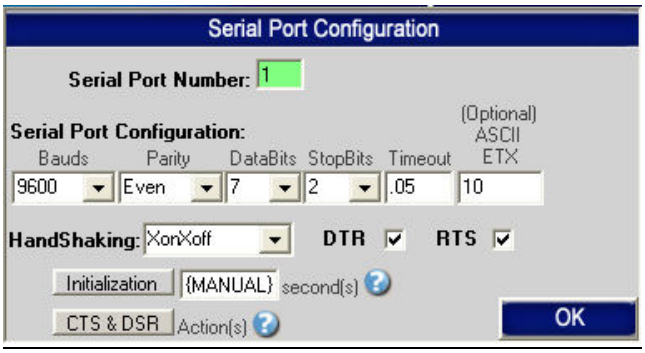


Don't forget to push the Save Configuration button to save your current settings.

Supported Operating Systems: Windows 98,Me,2000,3000,XP,NT4,Vista
Single Computer License with each purchase.

Simply the best RS232 data capture & virtual keyboard software available !

SERIAL PORT DESCRIPTION



Communicating through a Serial Port connection is very straightforward with the software. The software can send the data received by the Serial Port to the Keyboard buffer, TCP port, or a File, depending what switch is set to ON.
** See possible commands section*

The switch ON/OFF:

If position is ON, the data received by the peripherals: File, TCP, Keyboard and Virtual Keyboard are automatically redirect to this serial port.
If position is OFF, the data received by the peripherals are not sent to the serial port.

The Serial Port number:

Use this field to chose the serial port number *(where your peripheral is connected)*. If the field is GREEN, it's because the COM is present and available.
** To disable the serial port enter: 0*

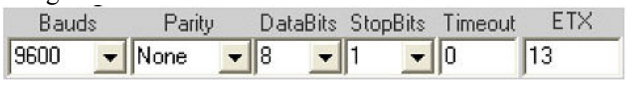
The Serial Port Configuration:

Use this section to set the baud rate, parity, data bit, stop bit, timeout *(in seconds) and ASCII ETX* parameters.

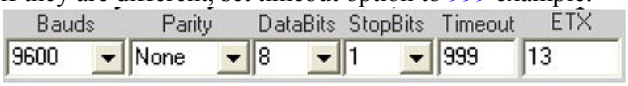
Timeout: The timeout feature is optional. It is used to received the packet in one shot.
* Example if the timeout is 0.2 then the serial port waits 0.2 second to send the string in one shot.

ASCII ETX: The ETX *(end of text (Terminating) character)* feature is optional.
It is used if your string always ends with the same character.
*ASCII characters from 0 to 255.

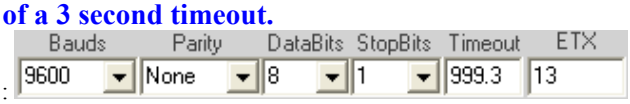
*For example: If your string ends with an ENTER/CR:



*To receive data only if they are different, set timeout option to 999 example:



- To receive the data only if they are different or if the **x** delay is exceed, set timeout option to **999.x Example of a 3 second timeout.**



HandShaking:

Use this field to chose the serial port hand shaking protocol configuration.
None: No handshaking
XonXoff: Software XON/XOFF handshaking
Rts: RTS/CTS (Request to send/ Clear to send) hardware handshaking
RtsXonXoff: Both request to send and XON/XOFF handshaking

DTR/RTS:

You can activate or deactivate the hardware line:
DTR, Enable the data terminal ready line.
RTS, Enable the request to send line.
Default is checked for standard operation

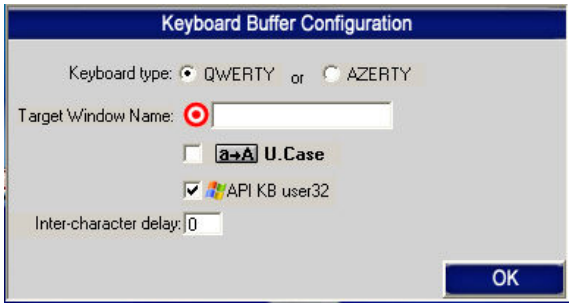
Initialisation button:

If the file: "C:\ProWedge.CFG\init_serial.txt" is present, then the contents of this file will be sent automatically:
- At startup.
- If you click on the button initialization.
- Default is {MANUAL} for standard operation

CTS & DSR button:

With the input pins CTS and DSR of your serial port you can generate action in your computer.
Example: If you put a hardware switch between the pin 7 and 8 of your serial port DB9 and you enter {F8} in the configuration CTS (action push) field}, then when you activate the hardware switch, an F8 key push is simulated in your computer.

KEYBOARD BUFFER CONFIGURATION



You can send the Serial port, File, TCP port and/or the Virtual Keyboard to the keyboard buffer if you set the Keyboard Switch to ON.

* *See possible commands section*

The QWERTY option (Default):

Select this option if you use a QWERTY keyboard.
**If your keyboard is QWERTY you can see just across from the letter Q the letters WERTY*

The AZERTY option:

Select this option if you use a AZERTY keyboard.
**If your keyboard is AZERTY you can see just across from the letter A the letters ZERTY*

The Target Windows Name:

With this option you can automatically activate a target window.
Enter the title of the window where you want to send the data
* If you enter nothing, the software sends the data to the current window.
* You can enter just a part of the title, for example if the title is "**my application software**" you can enter just a part of the title for example: **my application**

The U. Case option:

The software can switch all lower case characters automatically to upper case.

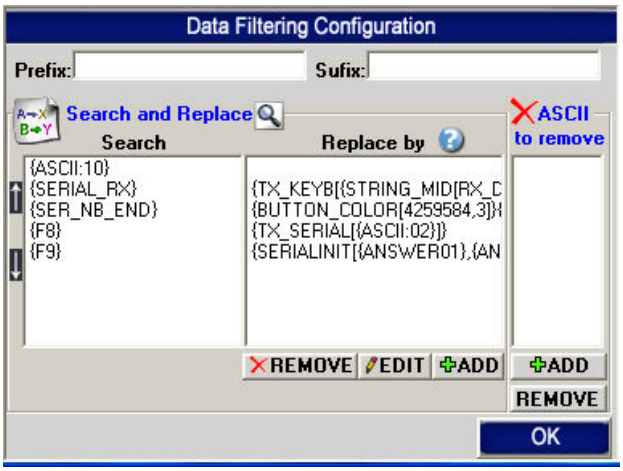
The API KB user 32 (ON by default):

If you activate this option, the data is sent directly to the Keyboard buffer, this method work's with all kinds of applications.
*Usually this option is set to ON.

The delay option:

With this option you can change the output inter character delay.
Default is 0 for standard operation.
*You likely don't need to change this option frequently.

DATA FILTERING CONFIGURATION



All options in this section apply to: Serial port, File, TCP port, Keyboard and Virtual Keyboard

The Prefix/Suffix:

When the software receives data from a peripheral, it adds the PREFIX at the beginning of data and the Suffix at the end.
[Prefix][data from a peripheral][Suffix]
The Prefix and the Suffix can contain any characters and or commands that you want.
**see possible commands section*

The Remove ASCII section:

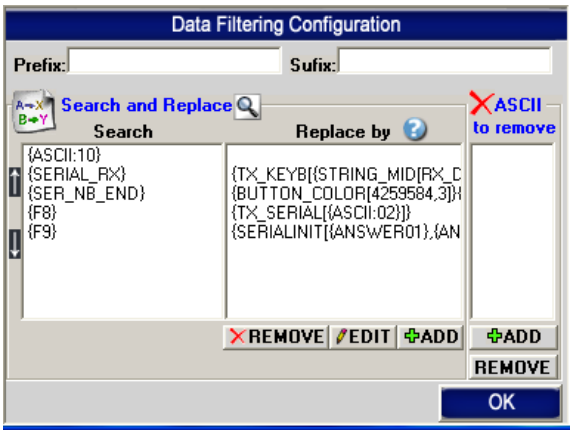
With this section you can remove ASCII characters directly from the received data. With the ADD button you can add new ASCII characters from 0 to 255.
*Example: To remove Enter, ADD: 13
*Example: To remove Line Feed, ADD: 10

The Search and Replace section (up and down arrows):

Use the arrows up and down to change the order of the lines present in this section.

NOTE: Be careful when Editing this section… Clicking on 'Cancel" will delete the entry during Editing. Clicking 'OK' will preserve/save the entry.

DATA FILTERING CONFIGURATION



The Search and Replace section:

With this section you can search and replace the string from any peripheral. In the field "Search" enter the search string and in the "Replace by" enter the replacement string. The "Replace by" string can contain whatever you want:
**see possible commands section*

To replace all that comes from the serial port use this syntax: {SERIAL_RX}

Default Data Filtering Set-up (for above, and most gauges)

Example / Explanation

Search (this color)
Replace by (this color)

First Rule: {ASCII:10}

Leaving *Blank removes extra LF from gauge output string,* (FYI, LF = {ASCII:10}).

Second Rule: {SERIAL_RX}

{TX_KEYB[{STRING_MID[RX_DATA,2,9]}{TAB}{CURDATE:MM-DD-YYYY}{TAB}{CURTIME}{ASCII:13}]}{TX_BUTTON_TXT[4,{STRING_MID[RX_DATA,2,9]}]}

Gets value: From the position 2, extract 9 characters from the gauge output string / sends characters to KB buffer.

Inserts TAB

Inserts Date

Inserts TAB

Inserts Time

Inserts CR

Inserts value into Green Button (#4). Gets value: From the position 2, extract 9 characters from the gauge output string / sends characters to Green Button #4

Third Rule: {SER_NB_END}
{BUTTON_COLOR[4259584,3]}{BUTTON_COLOR[16777215,2]}
Changes Start/Stop button color in the Timed Readings box.

Fourth Rule: {F8}
{TX_SERIAL[**{ASCII:02}**]}
Sends Call command **{ASCII:02}** *to gauge.* (**{ASCII:02}** = STX *command*)
Pressing F8 is for a Single Read… it sends the Call command to the gauge and the value (string) is returned.

Fifth Rule: {F9}
{SERIALINIT[{ANSWER01},{ANSWER02}]}{BUTTON_COLOR[4259584,2]}{BUTTON_COLOR[16777215,3]}
Pressing F9 Starts Timed Readings.

Adding the 'SEND/RECORD TO FILE' feature (works on most gauges)

If you want to record all the data you read/capture and send that data to a file in the background, follow these steps…

1. Create a file named FILE_IN_OUT.TXT on the C drive (C:\FILE_IN_OUT.TXT)
2. Replace the 'Second Rule' in the 'Data Filtering' with…

{SERIAL_RX}

{TX_KEYB[{STRING_MID[RX_DATA,2,9]}{TAB}{CURDATE:MM-DD-YYYY}{TAB}{CURTIME}{ASCII:13}]}{TX_BUTTON_TXT[4,{STRING_MID[RX_DATA,2,9]}{TX_FILE[C:\FILE_IN_OUT.TXT,{STRING_MID[ORI_DATA,2,9]}{ASCII:9}{CURDATE:MM-DD-YYYY}{ASCII:9}{CURTIME}{ASCII:13}{ASCII:10}]}]}]}

NOTE: To replace the rule,
click on 'Data Filtering',
highlight the second rule,
click on 'EDIT',
'COPY' the above rule and 'PASTE' it into the appropriate area, 'Search' or Replace';
click OK,
click 'Save Configuration',
click 'Minimize to Tray Icon,
End the program.

3. This feature will record each reading in a .TXT file (FILE_IN_OUT.TXT), in a column format and place the Date and Time next to each reading.
4. To remove the Date, simply delete {CURDATE:MM-DD-YYYY} and {ASCII:9}
5. To remove the Time, simply delete {CURTIME} and {ASCII:9}
6. To remove the spaces between reading, date, and time, simply delete them.

If you only want to log data to the file in the background without seeing the value on your screen…

Replace the Second Rule with…

{SERIAL_RX}
{TX_FILE[C:\FILE_IN_OUT.TXT,{STRING_MID[ORI_DATA,2,9]}{ASCII:13}{ASCII:10}]}

Misc. Filtering Information

TIP: To receive/display the ENTIRE string (including CR, LF) make the following change.
Example; in the LE1000-2 Default Filtering Set-up, replace this rule (number 2 in list)…

{SERIAL_RX}
{TX_KEYB[{STRING_MID[RX_DATA,2,9]}{TAB}{CURDATE:MM-DD-YYYY}{TAB}{CURTIME}{ASCII:13}]}{TX_BUTTON_TXT[0,{STRING_MID[RX_DATA,2,9]}]}{TX_BUTTON_TXT[4,{STRING_MID[RX_DATA,2,9]}]}

with this one…

{SERIAL_RX}
{TX_KEYB[RX_DATA]}{TX_BUTTON_TXT[0,RX_DATA]}

Search and Replace Examples:

Example 1: On all data received: From the position 2, extract 5 characters

Search	Replace by
{ALL_RX}	{STRING_MID[RX_DATA,2,5]}

POSSIBLE COMMANDS SECTION

Send data to KEYBOARD BUFFER:

You can send data directly to the KEYBOARD BUFFER
{TX_KEYB[String to Send]}
*The String to Send can contain "Special ASCII character": {ASCII:XXX}
Example: {TX_KEYB[RX_DATA and Hello{ASCII:13}{ASCII:10}]}

Send data to a VIRTUAL KEYBOARD BUTTON:

You can change the TEXT of a virtual button with this command
{TX_BUTTON_TXT[Button_Number, Text]}
Button_Number = The number of the button
Text = The new text for this button
*The Text can contain "Special ASCII character": {ASCII:XXX}
*To clear the text button use EMPTY in the Text
Example: {TX_BUTTON_TXT[0,Hello !!!]}

Send data to FILE:

With this command you can send data directly to a FILE,
{TX_FILE[FILE_NAME,STRING]}
FILE_NAME: Directory and file name
STRING : Sequence to save
*The String to Send can contain "Special ASCII character": {ASCII:XXX}
Example 1: {TX_FILE[C:\Barcode.txt,Hello !!!]}
Example 2: {TX_FILE[C:\Barcode.txt,RX_DATA]}

String manipulation MID:

With this command you can extract a specific number of characters,
{STRING_MID[RX_DATA,Start,Length]}
Start = Starting position.
Length = Number of character to return (use 0 to extract all character from the Start)
Example: {STRING_MID[RX_DATA,2,0]}
Example: {STRING_MID[RX_DATA,2,3]}

String manipulation RIGHT:

With this command you can extract from the right a specific number of characters,
{STRING_RIGHT[RX_DATA,Length]}

Length = Number of character to return

Example: {STRING_RIGHT[RX_DATA,4]}

Multiply by:

With this command you can multiply any numbers,
{MULTIPLY[TheNumber,Mutiply_by,Format]}

TheNumber = Number
Mutiply_by = Number
Format = *Optional: For example the format for 2 decimal is #.##

Example: {MULTIPLY[10,2]}
Example: {MULTIPLY[RX_DATA,10,#.##]}

Addition:

This command returns the sum of two numbers,
{SUM[TheNumber1, TheNumber2,Format]}

TheNumber1 = Number
TheNumber2 = Number
Format = *Optional: For example the format for 2 decimal is #.##

Example: {SUM[10,2]}
Example: {SUM[RX_DATA,10,#.##]}

Subtraction:

This command returns the subtraction of two numbers,
{SUB[TheNumber1, TheNumber2,Format]}

TheNumber1 = Number
TheNumber2 = Number
Format = *Optional: For example the format for 2 decimal is #.##

* Example: {SUB[10,2]}
* Example: {SUB[RX_DATA,10,#.##]}

POSSIBLE COMMANDS SECTION (continuation)

Keyboard buffer output, possible values

Each key is represented by one or more characters. To specify a single keyboard character, use the character itself. To represent the letters A, B, and C, use ABC for string. To specify characters that aren't displayed when you press a key, such as ENTER or TAB, and keys that represent actions rather than characters, use the codes shown below:

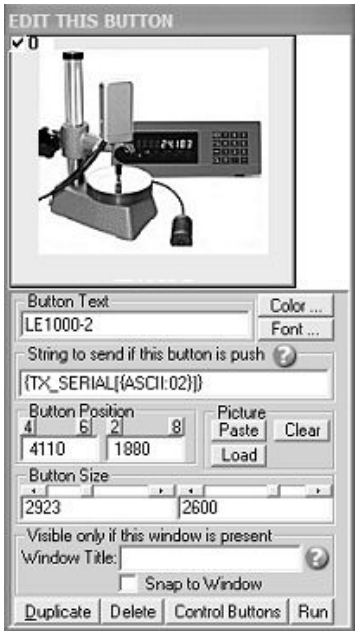
Description	Command
SEND TO WINDOWS DESKTOP	{WinDesk}
SEND TO CURRENT WINDOWS	{WinCur}
START MENU WINDOWS	{StartWin}
BACKSPACE	{BACKSPACE}, {BS}, or {BKSP}
BREAK	{BREAK}
CAPS LOCK	{CAPSLOCK}
DEL or DELETE	{DELETE} or {DEL}
DOWN ARROW	{DOWN}
END	{END}
ENTER	{ENTER}or ~
ESC	{ESC}
HELP	{HELP}
HOME	{HOME}
INS or INSERT	{INSERT} or {INS}
LEFT ARROW	{LEFT}
NUM LOCK	{NUMLOCK}
PAGE DOWN	{PGDN}
PAGE UP	{PGUP}
PRINT SCREEN	{PRTSC}
RIGHT ARROW	{RIGHT}
SCROLL LOCK	{SCROLLLOCK}
TAB	{TAB}
UP ARROW	{UP}
Current prefix	{CURPREFIX}
Current suffix	{CURSUFIX}
Current System Time	{CURTIME}
Current System Date	{CURDATE:DD-MM-YYYY}
Get clipboard text	{CLIPBOARD}
XX Button text (XX = The virtual button number)	{BUTTON_TXT[XX]}
Change Prefix	{PREFIX[TMP_PREFIX_HERE]}
Change Suffix	{SUFIX[TMP_SUFIX_HERE]}
F1 to F16	{F1} to {F16}
Activate a Target Windows Application	{APP[Window_Title_Name]}
{SHIFT_DOWN}	Key Shift press
{SHIFT_UP}	Key Shift release
{CTRL_DOWN}	Key Ctrl press
{CTRL_UP}	Key Ctrl release
{ALT_DOWN}	Key Alt press
{ALT_UP}	Key Alt release
Data received (example the Barcode)	RX_DATA
In TCP SERVER mode the IP address	{TCP_ADR}
{PAUSE.01SEC}	Do a pause of .01 second
{PAUSE.1SEC}	Do a pause of .1 second
{PAUSE.5SEC}	Do a pause of .5 second
{PAUSE1SEC}	Do a pause of 1 second
{PAUSE2SEC}	Do a pause of 2 seconds

To specify keys combined with any combination of the SHIFT, CTRL, and ALT keys, precede the key code with one or more of the following codes:

{SHIFT}	{CTRL}	{ALT}

Example: to send ALT and A use this syntax: {ALT}A

VIRTUAL KEYBOARD (All Buttons control)



Click on Button to open this window

Buttons list: Just under "Buttons" you can see the existing button list.

View all: Use this option to view all **Buttons list** on the screen.

Hide all: Use this option to hide all **Buttons list** on the screen.

Run: Use this option to try/run the Virtual Keyboard buttons.

Buttons Positions: Use this section to change the position of the selected button in the **Buttons list**. * Use the vertical scroll bars to change the moving speed.

Add: Use this option to add a new Virtual Keyboard Button.

Del: Use this option to delete the selected/checked buttons in the Buttons list. **Be very careful that the ONLY button checked is the button you want to delete. If other buttons are checked, they will also be deleted.**

View Config: Use this option to show or hide the configurations of the selected button in the **Buttons list**.

View Number: Use this option to show or hide the number of the selected button in the **Buttons list**.

Import: Use this option to import a virtual keyboard.

Export: Use this option to export the selected button(s) in the **Buttons list** to a FILE.

Color: Use this option to change the Color of the selected button in the **Buttons list**.

Font: Use this option to change the Font of the selected button in the **Buttons list**.

Save cfg: Use this option to save the current configuration.

Button Text: Use this field to set what you want to see on the button.

String to send if this button is pushed:

Use this field to set the string to send if the user pushes this button. The string is sent to the Serial port, TCP port, File and/or to the Keyboard buffer depending what switch is set to ON.

** The string is only send in **RUN** mode.*
The string can contain: **See possible commands section*

Button Position:

Use this section to change the position of the button.
** Tips: Activate the NumLock Click the button and use the number 4,6,2 and 8 to change the position.*

Button Size:

Use this section to change the size of the button.

Duplicate:

Use this option to duplicate this button.

Color:

Use this option to change the button's background color.

Font:

Use this option to change the button's text font.

Picture:

Use this section to:
Paste a picture from the ClipBoard to the button.
Clear a picture button.
Load a picture button.

Run:

Use this option to try the Virtual Keyboard buttons.