I.M.S INDUSTRIAL MEASUREMENT SYSTEMS LTD.
P.O. BOX 6305 Haifa, Israel 3106201
Tel.: 972-4-811-0877
Fax: 972-4-811-0875
email: sales1@ims.co.il
http: //www.ims.co.il

LCIC-WIM USER'S MANUAL

17 June '15

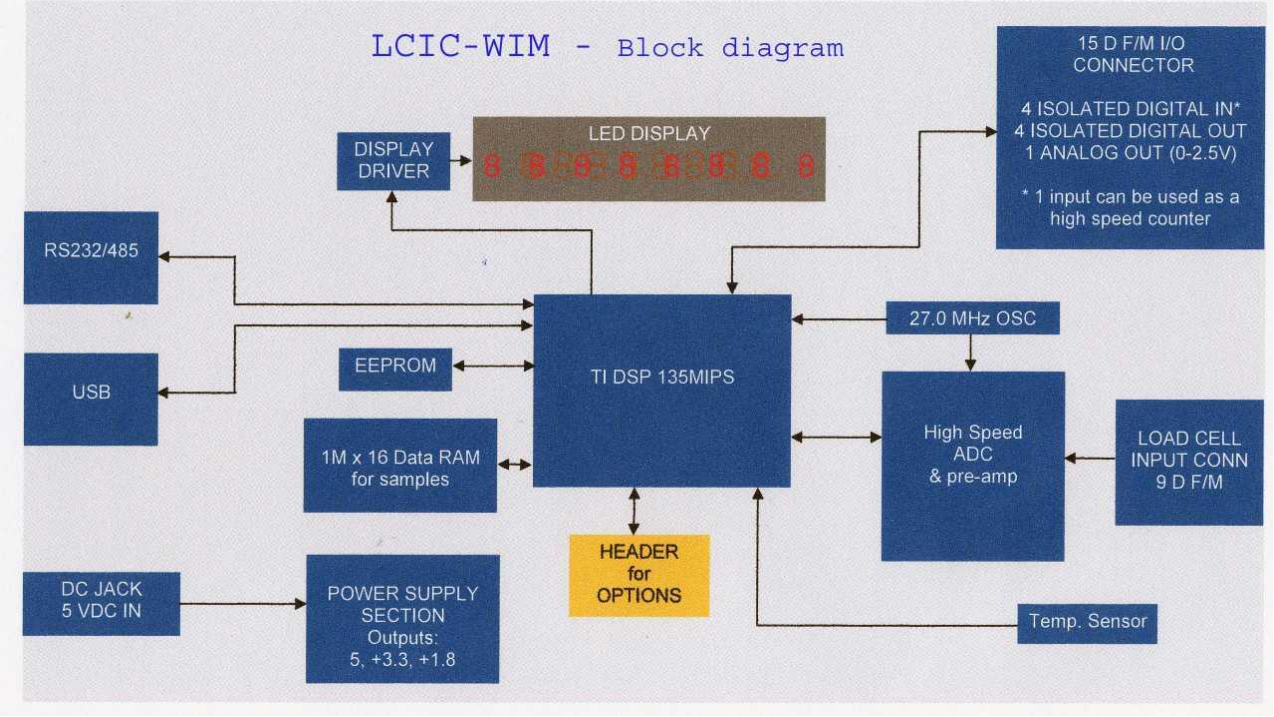


Table of Contents

1. Introduction
1.1 General Description
1.2 Typical Applications
1.3 Minimum PC Requirements

2. Installing the board in the PC

3. Utilities
3.1 Setup & Running
3.2 The Calibration Utility
3.2.1 General
3.2.2 The Calibrations Library
3.2.3 Parameters
3.2.4 Calibration Efficiency (CE)
3.2.5 Non-Linear Scale Adjustment (NLSA)
3.3 The Settings Utility
3.3.1 The Menu Bar
3.3.1.1 Tools / Analog Output
3.3.1.2 Tools / Baud Rate for SCI port
3.3.1.3 Tools / General Setpoints
3.3.1.4 Tools / Sampling Rate
3.3.1.5 Tools / Set to Default Parameters
3.3.2 Parameters
3.3.2.1 Communication
3.3.2.1.1 Communication type
3.3.2.1.2 Baud Rate
3.3.2.1.3 RS485 Address
3.3.2.1.4 Get results immediately
3.3.2.2 Auto Zero
3.3.2.2.1 Activate
3.3.2.2.2 Max Zero
3.3.2.2.3 Min Zero
3.3.2.2.4 Time limit
3.3.2.3 Start Fill-mode
3.3.2.3.1 Fill-mode starts automatically upon card reset
3.3.2.4 Filtering
3.3.2.5 Filling Definition

3.4 The Parameters Import/Export Utility
3.5 The Parameters Lock Utility

4. Programming your Application

4.1 Commands
4.2 Parameters
4.3 LCIC-WIM ActiveX
4.3.1 Start/Stop Communication
4.3.2 Variables
4.3.3 Filters
4.3.4 Fast Mode
4.3.5 Misc.
4.4 RS485
4.4.1 Introduction
4.4.2 Doubling Rules
4.4.2.1 Which character should be doubled?
4.4.2.1.1 In General Mode
4.4.2.1.2 Inside the inner mode (LiW-mode, CHW-mode, etc.)
4.4.2.2 Is a doubled character limited in time?
4.4.3 Tips
4.4.3.1 Baud Rate

Appendices

A. I/O & the LED Display

A.1 General Notes about the I/O
A.2 Connecting External Devices to the Inputs
A.3 LCIC-WIM I/O & Analog Output
A.4 The LED Display

B. Scaling the Load Cell Input

C. Load Cell Wiring

D. USB, RS232 & RS485

D.1 Communication Parameters
D.2 Baud Rate
D.3 General RS232 Terminal
D.4 Serial Communication & PC Power On/Off
D.5 RS485
D.6 Automatic Refresh of the Serial Communication

E. Fill Mode

E.1 Introduction
E.2 Hardware Inputs
E.3 Hardware Outputs
E.4 Filling Parameters
E.4.1 Filling By = Weight
E.4.1.1 Filling by Weight Parameters
E.4.1.2 Auto Tare
E.4.1.3 Valid Results Limits
E.4.1.4 Stabilization Criterion (Tare & Stop)
E.4.1.5 Lazy Filling
E.4.2 Filling By = Time
E.4.2.1 Filling by Time Parameters
E.4.2.2 Stabilization Criterion (Stop)
E.4.3 Fast Speed Config
E.4.4 Advanced Parameters
E.4.5 The Filling Configurations Library
E.4.6 Addresses of the Filling Parameters
E.5 LED Display Notations

E.6 Commands
E.7 Error Codes

F. Specifications
F.1 Load Cell Input
F.2 A/D
F.3 Digital Inputs
F.4 Digital & Analog Outputs
F.5 Standard Interfaces
F.6 Software
F.7 Dimensions (mm)
F.8 Misc.

G. Trouble-shooting
G.1 Card does not respond after PC power-on
G.2 Loss of communication after calibration with RS485

H. Zero & Tare
H.1 The *Zero* function
H.2 The *Auto-Tare* function

I. Board Reset by Input #4

J. Overflow & Underflow

1. Introduction

1.1 General Description

The model LCIC-WIM is a very High Speed, Intelligent Load Cell Interface Card with USB/RS232/RS485. Besides its basic mode – named below as the **general mode** – the board includes an integral Fill Mode supplying an independent filling control. The board is intelligent and powerful enough for OEM customers – it is ready to accept piggy-back modules and/or embedded applications for OEM special requirements.

1.2 Typical Applications

- Dynamic weighing – vehicles, livestock
- Dynamic force measurement
- High speed checkweighing
- High speed filling / batching
- Beltweighing
- Force measurement / Press Machines
- WIM-Monitor for analysis of dynamic systems

1.3 Minimum PC Requirements

CPU: Pentium 4, 2.0 GHz

RAM: 512 MB

Hard Disk: 60 GB

Free port of USB or RS232.

OS: Microsoft Windows XP Professional, Service Pack 2
(Note: Vista is not supported)

2. Installing the board in the PC

(The following description refers to Windows XP. Obviously, on another operating system it might be different.)

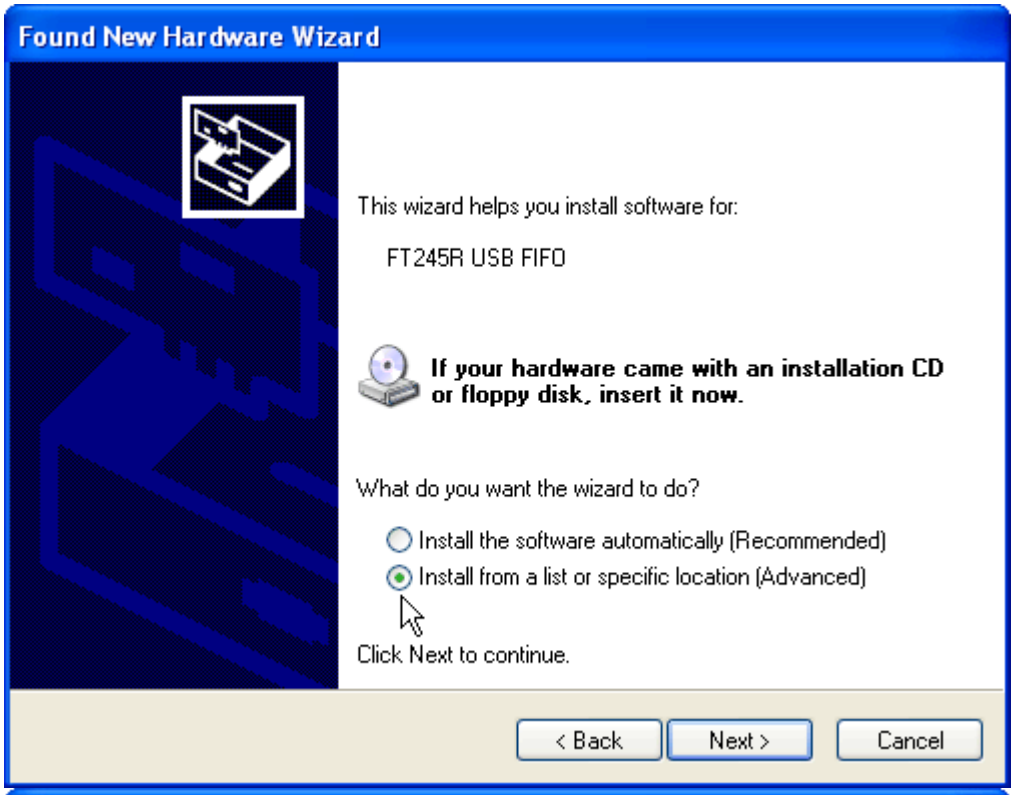
1. Make sure that all installation files have been copied to your hard disk to a new folder, say, LCIC-WIM.

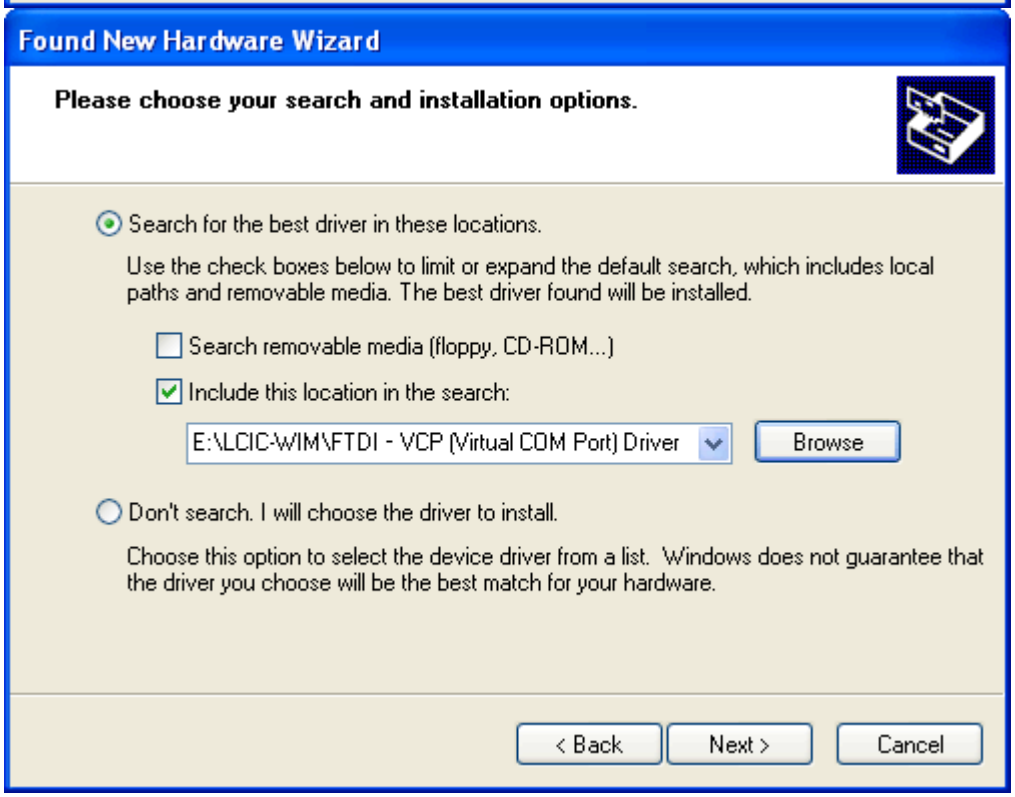
2. Connect the LCIC-WIM board to your PC.

3. The 'Found New Hardware Wizard' appears.
   Select the last option like this:

Click 'Next'.

4. Select the second option, click 'Next' and browse to the "FTDI - VCP (Virtual COM Port) Driver" folder (under the folder where you copied the installation files in step 1).





Click 'Next'.

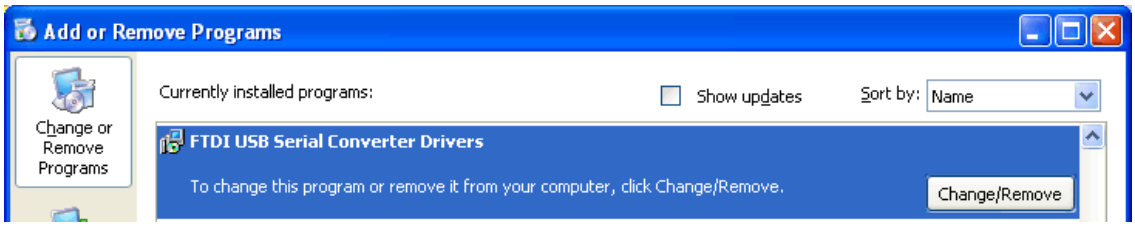
5. After a while, you'll have this display:

Click 'Finish'.

Notes

1. *It might occur that the wizard will return to step 3, requiring to repeat the process. This is normal, just repeat steps 3-5.*

2. *You may watch the new driver in 'Add/Remove programs':*



3. Utilities

3.1 Setup & Running

1. Run the setup(s) of the LCIC-WIM utilities in the folders:

* LCIC-WIM-CALIBRATION
* LCIC-WIM-SETTINGS
* LCIC-WIM-PARAM
* LCIC-WIM-PARAM-LOCK

2. Run a utility:

* If the utility reports that **.Net Framework** is not installed, then run "dotnetfxV1.1.4322.exe" in the "Dot_Net_Framework" folder on your CD. This is a free public file released by Microsoft.
* If the utility reports **"LCIC driver is not installed"**, then refer to the previous section ("Installing the board in the PC").
* The **interface** of the 1st & 2nd utilities (LCIC-WIM-CALIBRATION (V3.06 and above) and LCIC-WIM-SETTINGS (V2.35 and above)) may be **self-translated** (by the user) to **another language**. For details refer to: *How to Switch Interface Language.pdf*.

3.2 The Calibration Utility

3.2.1 General

The calibration utility (LCIC-WIM-CALIBRATION) enables to calibrate the LCIC-WIM board adjusting it to your own system.
The utility is straightforward and is in the form of a Windows wizard. It includes three main stages carried out in five steps.

The three stages are:

1. **Show Data (step 1)**
   (Described below under 'Step 1'.)
2. **Trial Calibration (steps 2-3-4)**
   The calibration is called 'trial' as it won't be saved in the board unless the user confirms it in the next stage. As long as the user did not confirm the new (trial) calibration, the previous calibration remains in effect.
   (The details are described below under 'Step 2-3-4'.)
3. **Save or Quit (step 5)**
   (Described below under 'Step 5'.)

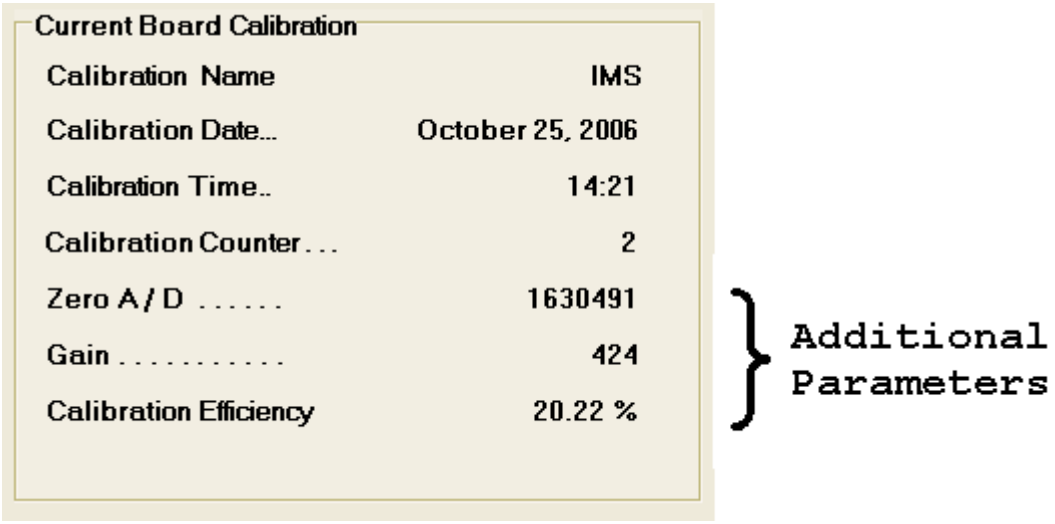
Shortcuts

There are two shortcuts to facilitate the calibration procedure:
* Using the **'Calibrations Library**' you may restore a previous calibration quickly and reliably. Refer to section 3.2.2.
* The '**Fast Digital Calibration**' (available only for boards that were supplied with that option) enables a fast and easy calibration without any physical operation – provided that load cell's exact characteristics are available and that calibration's accuracy is not critical.
  For details refer to section 3.2.3.

The five calibration steps are:

Step 1 – Show Data

This step introduces both the **parameters** and the **current readings**, as received from the board. The step is passive in the sense that it only *shows* data passed by the board, but it does not make any *change* in the board. Some **additional parameters** may be displayed in the bottom of the "Current Board Calibration" box. Type <Ctrl>/<I> and – as shown in the screenshot below – the additional parameters will appear, hiding the lower original parameters (Load Cell Output, Full Load Cell(s) Capacity, Maximum Applied Capacity and Display Resolution). In order to hide the additional parameters and return to all the original ones, type <Ctrl>/<I> again.



Step 2 – Parameters

This step starts the calibration procedure. It enables to change calibration parameters. Whether you changed the parameters or not, you may proceed to the next step by pressing the 'Next' button.

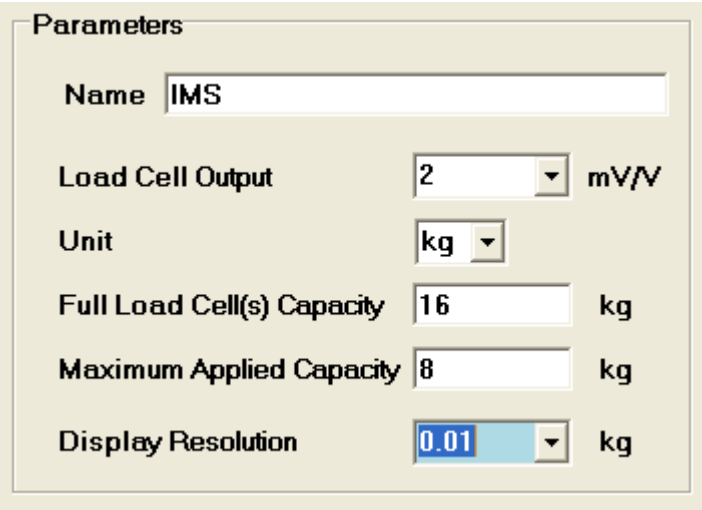
Calibrations Library

1. Alternatively, you may click 'Library' in order to access the library, as described in section 3.2.2.
   (If there are still no calibrations in the library, the 'Library' button will be inactive.)
2. Normally, upon confirming a new calibration it will be saved also to the calibrations library. However, if you want to save it only in the board, uncheck the 'Save to Library' box.

Fast Digital Calibration

Another alternative is the "Fast Digital Calibration", which is calibration without weights. Refer to 3.2.3.

In the example below the user changed the Display Resolution parameter:



Notes

1. The changed parameter (Display Resolution) is displayed with blue background.
2. The options list of the Display Resolution depends on the value of Maximum Applied Capacity.

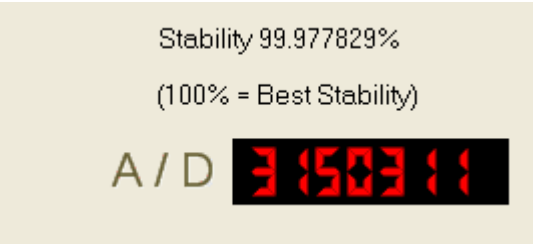
For details on each parameter refer to section 3.2.3.

Step 3 – Zero

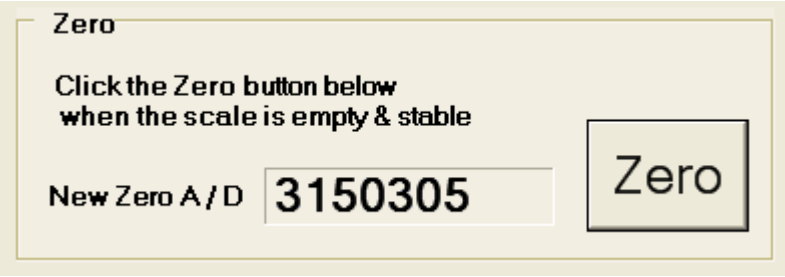
This step enables to redefine the 'zero' level. Click 'Skip' if you are satisfied with the previous definition of the 'zero' level. Otherwise, when the scale is empty and stable (see note), click 'Zero' to sample another 'zero' level. Once you clicked 'Zero', you may either confirm the new 'zero' level by pressing 'Next', or redefine it by clicking again the 'Zero' button, or leave it out by clicking 'Skip'. Please note that even if you do confirm the new 'zero' level by pressing 'Next', its effect is limited to the 'trial calibration' stage. It will be stored in the board only if the new (trial) calibration is confirmed in step 5.

Note

In order to know the **stability**, watch the Stability indicator above the A/D reading (e.g., 99.977829% in the screenshot below).



In the example below the user clicked 'Zero':



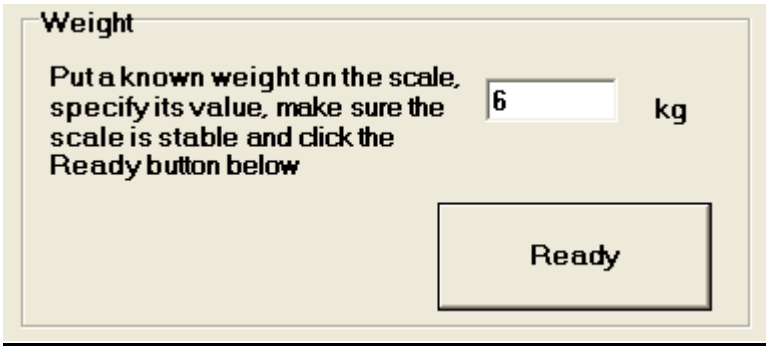
Now the user has three options:

1. Confirm this 'zero' level (click 'Next');
2. Redefine the 'zero' level (click 'Zero' once more);
3. Leave out this new 'zero' level staying with the previous 'zero' adjustment (click 'Skip').

Step 4 – Weight

This step enables to redefine the 'weight' level. Click 'Skip' if the previous 'weight' level was OK. Even if there was a fixed shift in the weight (which you probably corrected in step 3), you *don't have* to redefine the 'weight' level – just click 'Skip'. Otherwise, specify the value of the weight, and when the weight is stable on the scale, click 'Ready' to sample another 'weight' level. Once you clicked 'Ready', you may either confirm the new 'weight' level by pressing 'Next', or redefine it by clicking again the 'Ready' button, or leave it out by clicking 'Skip'. Please note that even if you do confirm the new 'weight' level by pressing 'Next', its effect is limited to the 'trial calibration' stage. It will be stored in the board only if the new (trial) calibration is confirmed in step 5.

In the example below the user specified the value of the weight (6 kg) and clicked 'Ready':



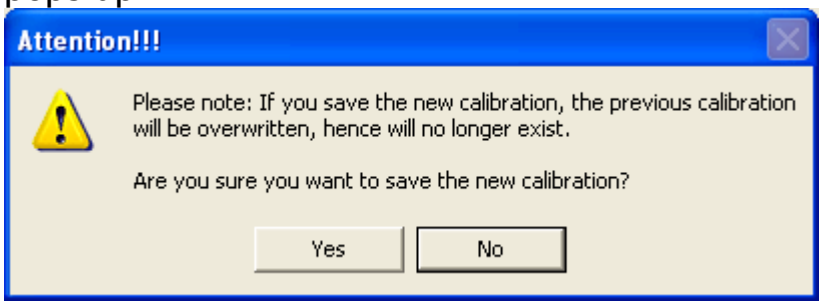
Now the user has three options:

1. Confirm this 'weight' level (click 'Next');
2. Redefine the 'weight' level (click 'Ready' once more);
3. Leave out this new 'weight' level staying with the previous 'weight' adjustment (click 'Skip').

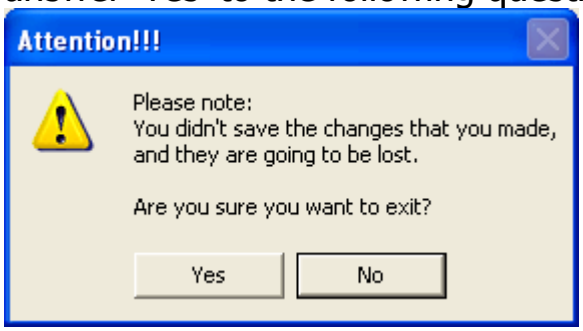
Step 5 – Save or Quit

This is the final step – here you decide whether to confirm the trial calibration, or leave it out. Before you decide, you may watch the current weight values, which are displayed without black background – in order to remind you that these are 'preview' weight values. That is, these values are supplied by the PC application (not by the board), showing what the board would supply if you confirm the trial calibration. Examine these preview weight values in order to decide whether to confirm the trial calibration or not.

- In case you do want to confirm the new (trial) calibration (overwriting the previous calibration), press the 'Save to Board' button and answer 'Yes' to the following question that pops up:



- Otherwise – that is, you want to stay with the previous calibration leaving out the 'trial calibration' – click 'Exit' and answer 'Yes' to the following question that pops up:



Please note that after saving the calibration to the board you still may remain in the utility, which will show now board's response after the calibration, which is now 'real' and not 'trial' any more.

3.2.2 The Calibrations Library

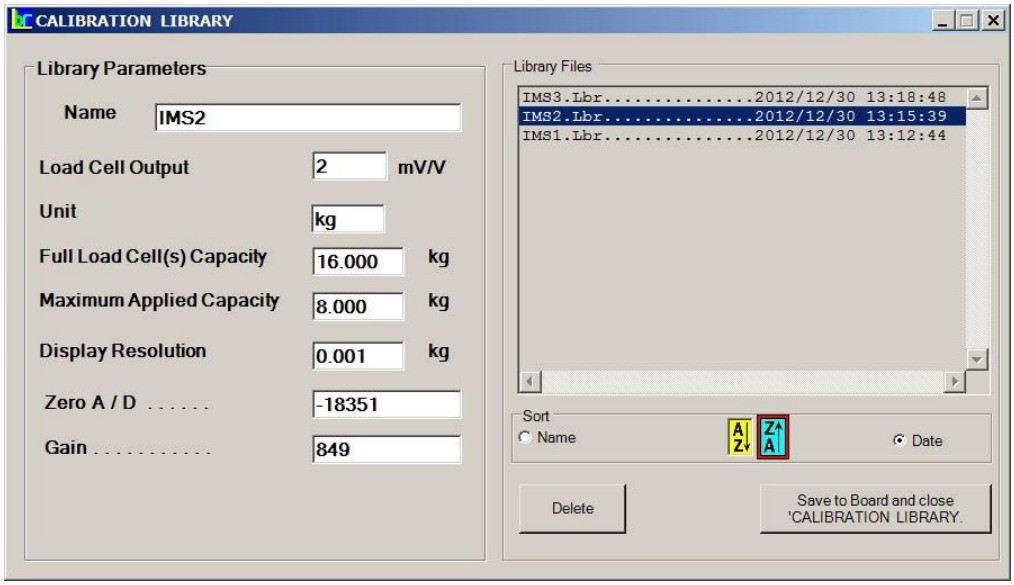
Each calibration that the user applies may be **saved** in the 'Calibrations Library'. Later on, the user may use that library as a shortcut in order to **restore** a previous calibration quickly and reliably.
The procedure is very simple:

Save

Unless the 'Save to Library' box is unchecked, each calibration is automatically saved to the library upon its saving to the board in Step 5. Its name is the 'Calibration Name' parameter.

Restore

In order to restore a calibration from the library to the board, press 'Library' in Step 2, and select the required calibration file (e.g., IMS2.Lbr in the example below). You may watch (but not change) the selected calibration's parameters. Clicking 'Save to Board and Exit' will restore the selected calibration to the board. Unnecessary calibration may be erased by clicking the 'Delete' button. You may sort the files list by name or by date.



The Calibration Library option supplies a fast calibration method in order to restore a previous calibration which worked correctly in the past and was saved in the library. However, it is designed to save and restore calibrations in **a same board**. That means that if you duplicate calibration parameters from one board to another, it apparently works, but the 'planted' calibration in the other board will **not** be accurate, as the Calibration Library is not designed for such an operation.

3.2.3 Parameters

This section describes the parameters to be specified during the calibration procedure.

Fast Digital Calibration

Introduction

LCIC-WIM boards that were supplied with the "Fast Digital Calibration" option may be calibrated without weights, based on three values provided by manufacturer's data sheet about the specific load cell being calibrated:
a. Load Cell Output (e.g., 1.984 mV/V).
b. Full Load Cell Capacity (e.g., 100 kg).
c. Zero Balance (e.g., 1%).
Zero Balance = Zero_Voltage / (Full_Scale_Voltage – Zero_Voltage)

The "Fast Digital Calibration" is very easy and fast and does not require any physical operation. Its two disadvantages are that (a) it requires that the information about the 3 above values is available and (b) it is not as accurate as a physical calibration.

How to carry out the "Fast Digital Calibration"

- In step #2 check the "Fast Digital Calibration" option.
- Specify the exact values of the 3 items above:
  Item a → parameter #1.
  Item b → parameters #2 & 3.
  Item c → parameter #6.
- Specify also parameters #4 & 5.
- Click 'Save to Board'.

Parameter #1: Load Cell Output

The mV/V output of your load cell.
In case of a physical calibration: Select 1, 2, 3 or "Unknown".
In case of "Fast Digital Calibration": Specify the exact value according to your data sheet (e.g., 1.984).
Refer also to Appendix B.

Parameter #2: Unit

The desired weight unit (g, kg, ton, oz or lb).

Parameter #3: Full Load Cell(s) Capacity

The maximum capacity of all the load cell(s) in the scale system, according to manufacturer's specifications. If the scale system has more than one load cell, e.g. four load cells of 10 kg each, then type 40.

Parameter #4: Maximum Applied Capacity

Fill in the actual full scale, i.e. the maximum load you plan to put on the scale.

Note
Keep the following condition true:
Maximum Applied Capacity + Dead Load <= Load Cell(s) Capacity.

Parameter #5: Display Resolution

Select the Resolution value of the displayed weight in the selected unit (1g, 0.05kg, etc.) that fits your application.

Parameter #6: Zero Balance

(Only in case of "Fast Digital Calibration".)

Specify the value of the Zero Balance according to your data sheet. For example: 0.5%.
Zero Balance = Zero_Voltage / (Full_Scale_Voltage – Zero_Voltage)

3.2.4 Calibration Efficiency (CE)

The potential range of A/D points is between 0 and near ±8,400,000. The 'Calibration Efficiency' specifies what portion
of this potential range is in use. The closer it is to 100%, the
better accuracy / stability you have. However, in practice,
100% is a theoretical number and almost not reachable.
The accuracy and stability will still be excellent even if
'Calibration Efficiency' is far lower than 100%.

3.2.5 Non-Linear Scale Adjustment (NLSA)

Availability: LCIC-WIM-CALIBRATION: V3.05 and up.
Board firmwares: Currently only in WiM-mode board, V3.124 and up.

Introduction

The basic calibration procedure assumes that the scale is **fully linear**. However, in some cases, like when bending plates are being used, the scale may be nonlinear, which is likely to affect the final accuracy. In order to satisfy even the case of the nonlinear scale, the LCIC-WIM-CALIBRATION application is now equipped with the 'Non-Linear Scale Adjustment' (NLSA) option. The NLSA option divides the full range into some **sub-ranges**, each one of them – by itself – being linear. You may 'teach' the board up to 8 points that you measured and know their real weights and their displayed weights (what the **card** shows on its LED display before applying the NLSA option). The NLSA option lets you feed the board with these data, so the board will implement the division of the full range to some sub-ranges. Coming readings will be matched with the proper sub-range and then adjusted by linear interpolation within this sub-range.

Operation

First, the standard calibration with one known weight must be completed. Once calibration was finished, it is recommended to check the scale linearity from zero to the maximum applied load with few reference weights. If the results are linear and acceptable – no need to do the Non-Linear Scale Adjustment.

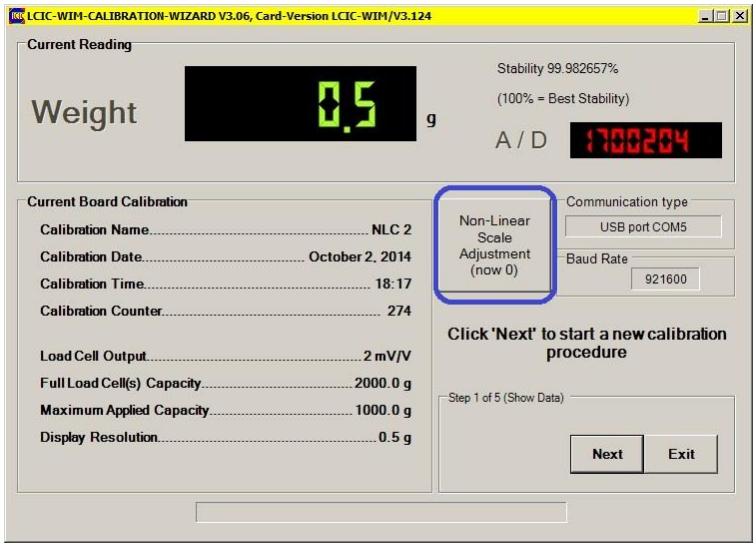
If the results show that the scale is nonlinear, make sure to note down a table of up to 8 weights that were applied: Note, side by side, the **real** (known) weight and the **displayed** weight – the readings that you read directly from board's LED.

Run the calibration procedure (LCIC-WIM-CALIBRATION).

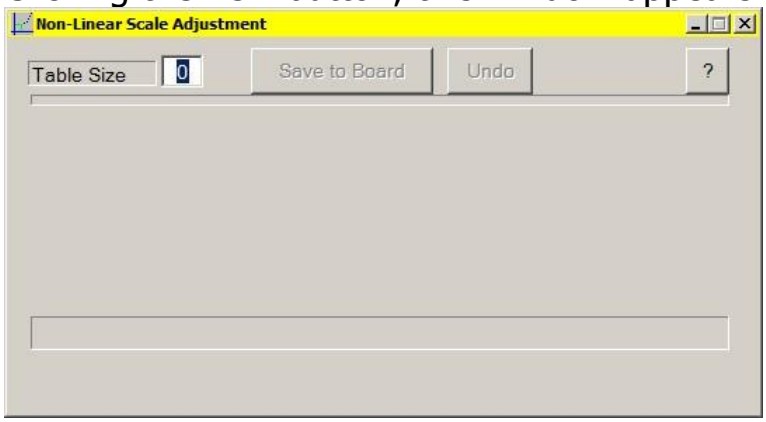
Make sure that when the scale is empty, the board shows zero. If this is not the case, correct it by clicking **Zero** in Step 3 while the scale is empty, then **Next**, **Skip** in Step 4, **Save to Board** in Step 5, **Yes** and **Yes**. Next, run again the Calibration application.

In Step 1 there is a new button:

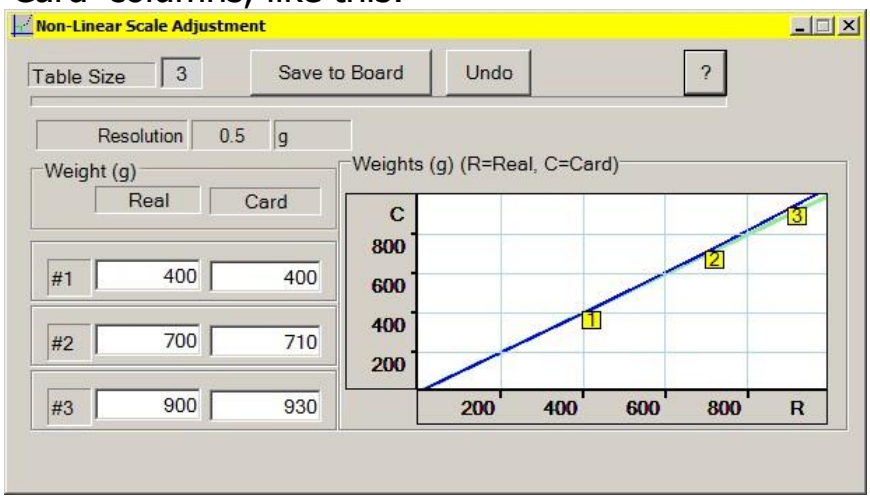
"Non-Linear Scale Adjustment". Here it is, marked (in the picture) by a blue frame:



Clicking the new button, this window appears:



Suppose your list contains 3 values: Your scale was linear up to 400 kg, but when you applied 700 kg the board showed 710 kg, and upon 900 kg the board showed 930 kg. So, all that you have to do is note down these values and "teach" the board these three points. Specify '3' in the 'Table Size' box and fill the values you noted down in the 'Real' and 'Card' columns, like this:

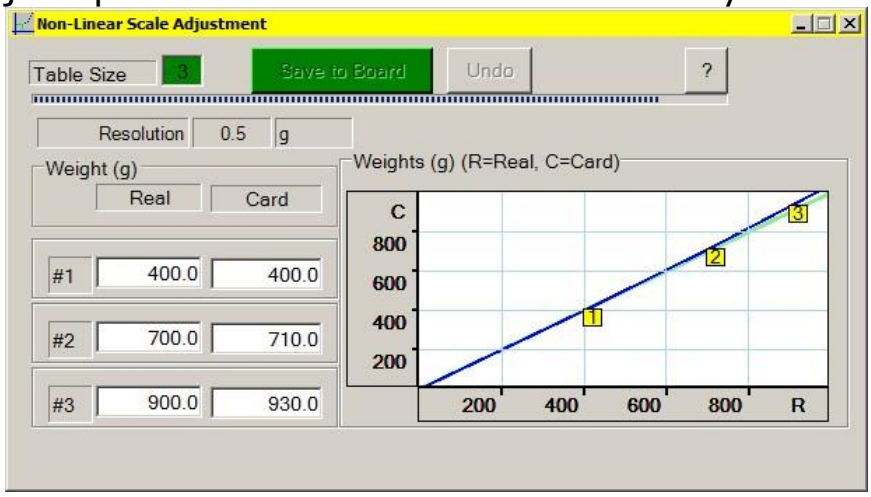


Note that the application gives you three levels of protection against human mistakes:
1. The application won't accepts weight values (neither 'Real' nor 'Card') beyond the range [0 → 'Maximum Applied Capacity'].
(Recall that 'Maximum Applied Capacity' is a parameter defined by the user during the calibration procedure.)
2. The application requires that the user specifies each column ('Real' & 'Card') in ascending order.

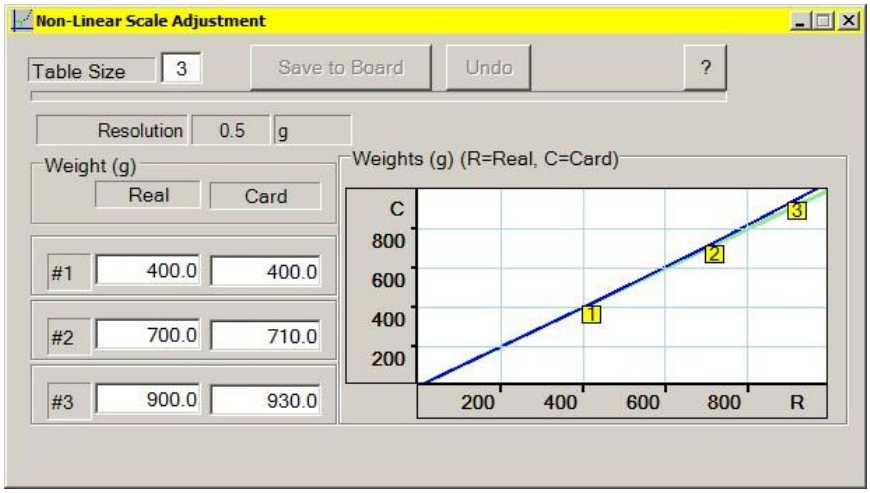
3. The application shows a graph of the **Real** weight vs. the **Card** weight. The ideal (linear) situation (always Real = Card) is marked by a green line. In the example above, as the first two segments ((origin→1 and 1→2) are the same (or almost the same) as the ideal situation, the green line is 'covered' by the blue (actual) line, so it's invisible. However, the third segment (2→3) shows the actual (blue) line and the ideal (green) line side by side.
Normally, the green line (if visible) will be:
* rather close to the blue line
* always in the same side (right or left) of the blue line

However, there is no protection beyond these three levels, so it is **<u>your</u>** responsibility to specify the weights correctly, as the application can't 'guess' whether a value is true or not.

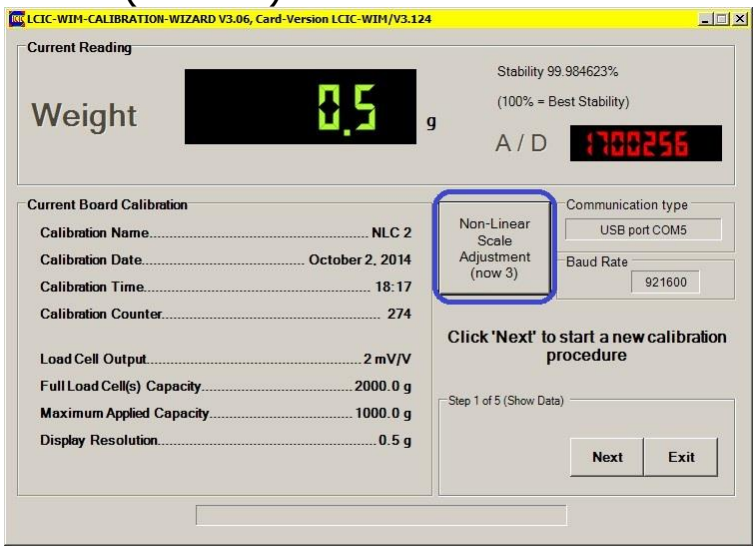
Click the 'Save to Board' button. The saving of the "points table" you just specified takes some seconds in which you'll see this:



Finally, the 'save' completes and the display shows:

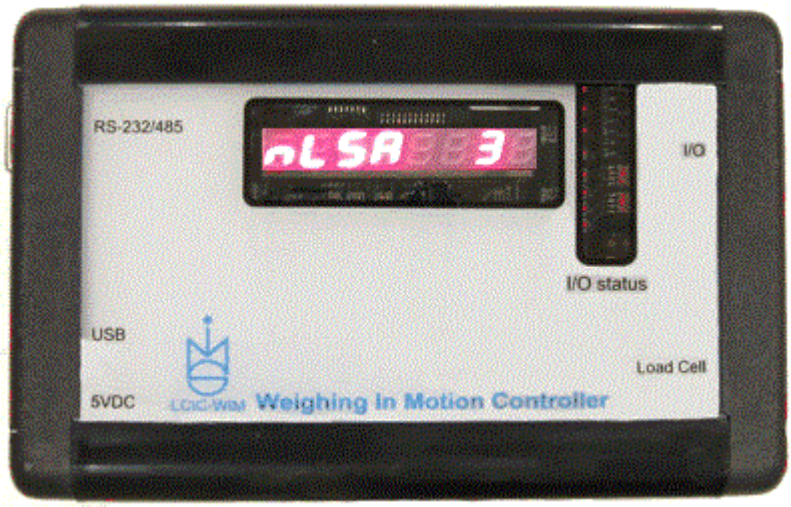


Now close this form. Note that the value of 'Table Size' (3 in our case) is specified on the bottom of the new "Non-Linear Scale Adjustment" button ("now 3"):



This '(now x)' line gives an instant indication about the current status of board's non-linear adjustment: '(now 0)', as it was in the beginning (see the first screenshot) indicates that the non-linear adjustment is inactive. '(now 3)', as it is now, indicates that a 3 points non-linear adjustment is currently active.

Current status of board's non-linear adjustment is shown also on board's LED: In our case, upon board restart, the last opening message is "nLSA 3" ("nLSA" stands for "**n**on-**L**inear **S**cale **A**djustment"), like this:



For compatibility, when the non-linear adjustment is inactive (the default), this opening message on the LED of the LCIC-WIM is skipped – there won't be a "nLSA 0" message.

In future:
1. If you'd like to repeat the "Non-Linear Scale Adjustment", you'll have to re-define the table. However, in the 'Card' column you should specify the weights that the LCIC-WIM board shows in **linear status** (without the "Non-Linear Scale Adjustment"). Therefore, in this case you must set 'Table Size' to 0, save it in the board, apply again your weights and note down again their 'Real' and 'Card' values. Then continue the same way as you did before.
2. If you re-calibrate specifying a weight in Step 4, the previous table becomes irrelevant, so 'Table Size' is automatically set to 0.

3.3 The Settings Utility

The Settings utility (LCIC-WIM-SETTINGS) gives control to card's filters, analog output, fill mode parameters and more. The utility has three items:

- The Menu Bar
- Current Weight Display
- Parameters

The 'Current Weight Display' is rather obvious – it continuously shows the actual weight. The other two items are detailed below.

3.3.1 The Menu Bar

The Menu Bar supplies some functions:

- Exit
  An alternative way to quit the utility.
- Tools / Analog Output
  Described below (section 3.3.1.1).
- Tools / Baud Rate for SCI port
  Described below (section 3.3.1.2).
- Tools / General Setpoints
  Described below (section 3.3.1.3).

3.3.1.1 Tools / Analog Output

(Relevant only for boards that include the 'analog output' option.)

The 'Settings' utility gives access to the analog output mechanism:
Click 'Tools' / 'Analog Output'.
For 'manual' mode uncheck the 'Activate Auto Mode' box.
For 'auto' mode check the 'Activate Auto Mode' box.

Manual Mode
On the top of the display, specify a desired voltage in the 'Manual Mode' square and click 'Send'.

Auto Mode
Specify the following parameters:
1. Voltage Max.
2. Weight Max.
3. Weight Min.
4. Update Frequency

The Auto Mode works like this:
* When the current weight = **Weight Min.** or less, the analog output is set to 0.
* When the current weight = **Weight Max.** or more, the analog output is set to **Voltage Max.**
* When the current weight is between **Weight Min.** and **Weight Max.**, the analog output is set between 0 and **Voltage Max.**, in the same ratio.
* The rate of updating the analog output depends on the **Update Frequency** parameter. For example, when **Update Frequency** is 4, the analog output is updated 4 times a second.

3.3.1.2 Tools / Baud Rate for SCI port

Click 'Tools' / 'Baud Rate for SCI port' to see the current baud rate for the RS232/RS485 serial port. It may be changed to some values between 19,200 and 115,200. (The baud rate for the USB need not be defined – usually it is 921,600.) **The change will take effect only after card reset.** The current b/r used by the board for the serial communication is shown for a while on the LED display upon card reset, prefixed by 'Sb' (=Serial baud rate). Do not mix the serial b/r with the b/r shown in the Communication box – they are not necessarily the same: The b/r in the Communication box specifies the ***actual*** b/r in which the Settings utility is communicating with the card. If the Settings utility communicates with the card through a serial port, the two b/r values will indeed be the same. However, in case the Settings utility communicates with the card through a USB port, the b/r displayed in the Communication box will be usually 921,600, regardless of the serial b/r.

Note: Normally, the b/r in the board side will be the same as the b/r in the PC side. However, it does happen that the communication is successful only when the PC sets the b/r to some other value. (This anomaly might occur only with the serial communication, not when using USB.) You may find out the required b/r in the PC side by watching any of the supplied utilities (Calibration, Settings or Monitor): upon the initialization they try to communicate with the board with various values of b/r.

3.3.1.3 Tools / General Setpoints

The LCIC-WIM board has four digital outputs. Each of them may be defined – through the *Tools / General Setpoints* – either as a **manual** output, or as a **general setpoint** output:

- A **manual** output is controlled by a user's command sent from the PC (or another computer). That is, the user sends – either by his own application or by a general RS232 terminal (see section D.3) – commands to turn an output on or off. The user – and not the card – has the initiative to turn an output on or off.
- A **general setpoint** output is controlled by the card according to user pre-defined criterion. Say, initially the user defined the setpoint as 10 kg, then the card automatically turns the output off/on when the current weight is less/more than 10 kg. The user cannot explicitly turn an output on or off by a command.
  Note – these general setpoints are absolutely different from the setpoints in the Fill mode (section E.4). The general setpoints are inactive in the Fill mode.

In order to configure an output:

- Select output's Mode as **Manual** or **Setpoint**.
- In case you selected **'Mode: Setpoint'**, specify the setpoint value (e.g., 10).
- Click 'Save to Board' to validate your new configuration.
- You may turn **on** or **off** each output that was defined as **manual**.

3.3.1.4 Tools / Sampling Rate

Here you may access the Sampling Rate for Fast Mode (see section 4.3.4). Please note that there are **two** sampling rates – one for a USB port and one for a SCI (serial) port.

3.3.1.5 Tools / Set to Default Parameters

Use this function to set all parameters to their default values, **overriding their previous values**.

3.3.2 Parameters

The following sections describe the various parameters. After changing parameter(s), click the 'Save to Board' button and wait a while until the new value(s) are accepted by the board.

3.3.2.1 Communication

The Communication box refers to card's communication port – either serial or USB. The current Port and Baud Rate are shown. They are 'read only', that is, not changeable. (About changing the Baud Rate for a serial port, refer to section 3.3.1.2.) A third item ('Get results immediately') is a parameter controlling card's response in communication during the Fill mode.

3.3.2.1.1 Communication type

Shows port's type and number, e.g., "USB port COM4" or "SCI port COM1" (SCI stands for 'Serial Communications Interface", that is, RS232 or RS485).

3.3.2.1.2 Baud Rate

Specifies the actual Baud Rate in which the Settings utility is communicating with the card. Refer also to section 3.3.1.2.

3.3.2.1.3 RS485 Address

Up to 64 LCIC-WIM boards may reside on one RS485 bus consuming only one PC port. In case you do not need this feature, specify RS485 Address = 0; this will simplify the coding of your application. If you do like to utilize this feature, specify the required RS485 Address – between 1 to 64.

→ The address setting takes effect only upon board reset.

3.3.2.1.4 Get results immediately

Controls card's response in communication during an inner mode, such as Fill-mode, CheckWeigher-mode, CatchWeigher-mode & WiM-mode:
When the 'Get results immediately' option is ON (checked), the card assumes that the PC (or another remote computer) is continuously connected and listening to card's communication port. Therefore, the card takes the initiative and sends messages to the PC, reporting the weighing results – immediately when they are available – in a format similar to the format of the report sent by the 'r' command. However, you might not need these messages, as the card functions as a 'stand-alone' device, and/or the PC draws card's results from time to time by the 'r' command. In the latter case, make the 'Get results immediately' option OFF (unchecked). Please note that when no remote computer (such as a PC) is connected and listening to the card, the 'Get results immediately' option **should** be unchecked, otherwise there might arise problems when the card sends messages to 'nowhere' as no remote computer is listening.

3.3.2.2 Auto Zero

The 'Auto Zero' optional feature supplies an automatic correction to creeps in the zero level during an inner mode (such as the fill-mode or the WIM-mode), caused by dust, temperature etc.
When this feature is activated and the card is inside the inner mode, the board automatically clears the gross weight if some pre-defined condition is satisfied: All readings within some 'continuous duration' are inside the 'zero range'.
Say, all readings during 3 consecutive seconds are between –1g and +2g.
These 'continuous duration' & 'zero range' are user-defined by the following parameters.
The auto Zero effect is temporary – it expires upon the exit from the inner mode, or card reset.

3.3.2.2.1 Activate
Activates the 'Auto Zero' feature. When 'Activate' is deselected (unchecked), an "Auto Zero' operation will never take place.

3.3.2.2.2 Max Zero
The *upper* bound of the 'zero range' (+2g in the example).
Note: Max. Zero refers to the original zero level as was defined during the calibration procedure.

3.3.2.2.3 Min Zero
The *lower* bound of the 'zero range' (-1g in the example).
Note: Min. Zero refers to the original zero level as was defined during the calibration procedure.

3.3.2.2.4 Time limit
The size of the 'continuous duration' (3s in the example).

If Time limit < 3 second, the 'auto zero' operation is not shown on the LED, but it does take place.

3.3.2.3 Start Fill-mode

3.3.2.3.1 Fill-mode starts automatically upon card reset

When this option is activated (checked), the card starts automatically the Fill-mode upon reset. Otherwise (the option is unchecked), the cards 'awakes' in the upper level, referred to in this document as the 'general mode'.

Notes

1. In order to switch the card from Fill mode to General mode, use the 'x' command (small 'x').
2. In order to switch the card from General mode to Fill mode, use the 'F' command.
3. About the 'x', 'F' and other commands in Fill mode, refer to section E.6.

3.3.2.4 Filtering

Board's **digital filtering** is used to "smooth" the read samplings by averaging a pre-set number of the internal readings. It's especially essential on a noisy environment, as this mechanism reduces system's susceptibility to short interferences. The 'noise' may be either mechanical (e.g., load cell vibrations), or electrical.
The digital filter averages the raw internal readings of the A/D, whose rate is 52,734 A/D readings per second. The behavior of the filtering mechanism is controlled by three parameters:

- **Filter1**
- **Filter2**
- **Decimator**

These three parameters, described below, are adjustable in order to make them suit best your application.

The filtering mechanism includes **two** levels:
Level 1: Each N_1 successive A/D readings are averaged forming the **first level average**.
Level 2: Each N_2 first level averages are re-averaged forming the **second level average**. Unlike the N_1 readings of level 1, the N_2 readings of level 2 are not successive: Only each N_3-th 'first level average' participates in the second level averaging procedure, the other N_3-1 ones being ignored.
As you might guess, N_1 is Filter1, N_2 is Filter2 and N_3 is Decimator.
The range of Filter1 & Filter2 is selectable from 2 to 256, where 2 is the **lowest** filtering and 256 is the **highest** filtering.
The range of Decimator is from 2 to 1000.

Note: As described in section 4.1, 4.2 & 4.3.2, user's application may select the filtering level it requests – either only Filter1, or Filter2 (=both Filter1 & Filter2). **However, the weight on board's numeric LED display is always after Filter2.**

Auxiliary Application: How to Select the Filters

The following board versions support an auxiliary application that facilitates the process of adjusting your filtering parameters:
Wim mode: V3.121 and up.
CatchWeigher: V4.274 and up.
CheckWeigher: V5.22 and up.

Refer to "LCIC-WIM-FILTERS-SIMULATOR.PDF".

3.3.2.5 Filling Definition

The '**Filling Definition**' box includes the parameters used upon a filling operation in board's **Fill Mode**, as described in section E.4.

3.4 The Parameters Import/Export Utility

Introduction

The Parameters Import/Export Utility (LCIC-WIM-PARAM) lets you 'export' your set of parameters to a file on your PC. Later you'll be able to 'import' this set of parameters to another board (or to the same board again) quickly and reliably.

How it Works?

The utility supplies two functions:
* **Upload** parameters ('export', board → file)
* **Download** parameters ('import', file → board)

Suppose you uploaded your set of parameters to file x. Now you may download this set from file x to another board, or to the same board again.

Notes

1. Excluded parameters

The Calibration data are **not** included in the upload file, as they may be stored by the 'Library' option in the calibration utility. So, when you download a file into the board, you may be sure that the calibration data will never be overridden.

2. Folder where the files exist

Uploaded files are always produced in the folder PARAM-LIBRARY under the application's folder; for example:
'C:\Program Files\IMS\LCIC-WIM-PARAM-V1.02\PARAM-LIBRARY'.
However, when **downloading** a file, PARAM-LIBRARY is merely the **default** folder – the user has the option to select a file in another folder.

3. File Name

The uploaded **file name** is according to:

* Board's firmware version (e.g., V1.21)
* Board's serial number (e.g., LC1-10025)
* Upload's date/time (e.g., 30_Mar_2011_15_23_22)

For example:

PARAM_V1.21_LC1-10025_30_Mar_2011_15_23_22.txt.

4. What happens when a set downloaded from a board with firmware version x is downloaded to a board with firmware version y?

If the user requests to download parameters that do not correspond to board's firmware version and/or authorization, the utility displays a warning message, but the user is still allowed to do the download.

Example:

*** You asked to download to V4.25 board version,**
but this set of parameters was uploaded from V3.108 board version.
Are you sure you want to download?

Instructions

1. To **upload** parameters (Board → File):
* Optionally, insert your remarks in the 'Remarks' box.
* Click 'Upload Parameters'.
2. To **download** parameters (File → Board):
* Select the required file:
  If it is in the **PARAM-LIBRARY** folder:
    Just click the name of the required file in the 'Library' frame.
  If it is in **another folder**:
    Click 'Browse', select the required file and click 'Open'.
    The 'Library' frame becomes blank. Click it to return to the PARAM-LIBRARY folder.
* Click 'Download Parameters'.

3.5 The Parameters Lock Utility

Introduction

The Parameters Lock Utility (LCIC-WIM-PARAM-LOCK) lets you ‘lock’ or 'unlock' board's parameters. Locking the board assures that only an authorized user – who knows the password – will have the ability to change parameters.

How it Works?

The initial state of the board – as supplied by the manufacturer – is **unlocked**. In this state you may lock the board by clicking 'Lock' and follow the displayed instructions.
When the board is locked:

* Applications that might change parameters – such as Settings & Calibration – ask the user to specify the password. The password may be specified either at application start or at any stage by typing <Ctrl>+P. If the user **does not** supply the password, the parameters will be displayed in 'view only' mode. If the user **does supply** the password, he is authorized to change the parameters. In order to make sure that the user does not forget the board in this mode, the authorization is limited by a countdown timer. The initial value of the timer is set (in seconds) by the PasswordTimeout.txt file in the application folder. You may manually edit this file and replace the default 1000 seconds value in the 'LockPasswordTimeout,sec.: 1000' string by any integer value between 30 to 5000. As long as the user works with the application, the countdown timer is renewed, so it becomes significant only after the user stops 'talking' with the application.

* The Parameters Lock Utility lets the user (who knows the password) to **unlock** the board, or change the password (set a **new password**). Click the appropriate button and continue according to the displayed instructions.

Using user's terminal or application

The command '**{**' (responses '{' + C/R) permits temporarily (10 sec.) to change parameters (by the command 'W') even though the board is locked.
The command '**}**' (responses '}' + C/R) terminates the above permission earlier (that is, before 10 sec.).

If – while the board is locked – the user tries to change any parameter by the 'W' command using his terminal (such as Termite) or application **without** preceding it by the '{' command,
board reaction will be as described in the "Board's response upon attempt to violate the locking" section, below.

Old Versions of IMS Applications

As specified in the 'Availability' section, using the 'board locking' option requires **new versions** of IMS applications. However, if, by mistake, a user still uses old versions of IMS applications (LCIC-WIM-CALIBRATION and/or LCIC-WIM-SETTINGS), then:
* When the board is unlocked: The applications will still work properly (the same as before).
* When the board is locked and the user tries to change parameter(s): The old IMS applications do not 'know' the 'locking board' option, so they **won't block** the user, and will send his (illegal) request to the board. Board reaction will be as described in the "Board's response upon attempt to violate the locking" section.

Board's response upon attempt to violate the locking

An application might send a 'write' command to a locked board. In this case the board will recognize the illegal request, so it will supply an indication that some severe disorder occurred by hanging up and displaying cyclically: "Err PWrd" and "do rESEt" (which stand for "Error Password" and "Do Reset"). The board will stay in this state until it is manually reset, in order to draw user's attention that something is wrong.

Availability

The 'Lock/Unlock' feature is supported only by the following board & applications versions:

Board versions

V3.114 and up.

Applications versions

LCIC-WIM-CALIBRATION: V2.21 and up.
LCIC-WIM-SETTINGS: V2.25 and up.

4. Programming your Application

The control of the board is by **commands** and **parameters**, described below. You may either use them directly (see also section D.3), or call an ActiveX (see section 4.3) that does the work.

4.1 Commands

Notes

►1. <c/r> signifies a carriage return.

►2. High Speed Commands
Single character commands (where no <c/r> is required) are used for quick direct access/control of the card.

►3. Commands with Arguments
Some commands include argument(s). In this cases wait 10 ms before sending the argument.
For example:
1. In the 'R*nn*<c/r>' command:
After sending the 'R', wait 10 ms before sending the '*nn*<c/r>'.
2. In the '**W***nn*<c/r>*argument*<c/r>' command:
Wait 10 ms:
* before sending the '*nn*<c/r>'
* before sending the '*argument*<c/r>'

Command	Action
a. Parameters: Read & write:	
R*nn*<c/r>	Read parameter #*nn*. All values are returned in floating point scientific format, preceded by *nn*<c/r> echo and appended by a c/r. E.g., the answer to 'R23<c/r>' is '23<c/r>2.560000e+02<c/r>' indicating that the value of Filter1 is 256. See note #3, above.
W*nn*<c/r>*argument*<c/r> **(upper case W)**	Write the value *argument* to parameter #*nn*. Floating point values can either be normal or scientific representation. For example 0.003 could be sent as it is or as 3.0E-03.After the '**W***nn*<c/r>' the board echoes '*nn*<c/r>' and after the '*argument*<c/r>' the board responds '*argument*<c/r>OK<c/r>'. For example, PC sends 'W23<c/r>', the board answers '23<c/r>, the PC then sends '123<c/r>' and the board answers '123<c/r>OK<c/r>' changing Filter1 to 123. See note #3, above.
b. Fast Mode: Start & stop (the Fast Mode is not available with RS485):	
u **(lower case u)**	Start a 'Fast Mode' session using Filter1 (no LED update). Terminated by the 'x' command. A timer stamp is appended. Its value is the time elapsed from start of transmission until end of transmission, in ms.
U **(upper case U)**	Start a 'Fast Mode' session using Filter2 (no LED update). Terminated by the 'x' command. A timer stamp is appended. Its value is the time elapsed from start of transmission until end of transmission, in ms.
x **(lower case x)**	Exit Fast Mode and return to general mode. In order to exit the fast mode, the 'x' command should be synchronised, that is, issued upon receiving a block.

c. Get a single reading of: weight, A/D or temperature:	
.	Get weight (after Filter2, not rounded to resolution).
?	Get weight (after Filter2, rounded to resolution).
>	Get A/D reading after Filter1.
<	Get A/D reading after Filter2.
T	Get temperature
d. Analog output: Read & write voltage: (Relevant only for boards that include the 'analog output' option.)	
@	Gets the analog output voltage (in Volts).
v**x**<c/r> **(lower case v)**	Sets the analog output voltage to **x** Volts. The analog output voltage is measured at pin 12 of CONN6 with respect to pin 13 which is ground. See note #3, above.
e. Digital Outputs: Write outputs:	
a	Turn on Output1 if Manual
A	Turn off Output1 if Manual
b	Turn on Output2 if Manual
B	Turn off Output2 if Manual
c	Turn on Output3 if Manual
C	Turn off Output3 if Manual
d	Turn on Output4 if Manual
D	Turn off Output4 if Manual
f. Digital Outputs: Read outputs:	
O **(upper case o)**	Card returns a string of the form '**xxxx**<c/r>', where **x** is either "1" or "0" representing the status of the 4 output opto relays OUT4,OUT3,OUT2,OUT1 respectively.
g. Digital Inputs: Read inputs & toggling counter:	
I **(upper case i)**	Card returns a string of the form '**xxxx**<c/r>', where **x** is either "1" or "0" representing the digital input status of IN4,IN3,IN2,IN1 respectively.
i	Gets a 16 bit 'toggling counter'. The 'toggling counter' increments each time input #2 goes from **high** to **low**.

h. RS485: Address selection, setting & reading (for more details and examples refer to section D.5; except 'Nx<c/r>', these commands are available also in the fill mode):	
:**x**<c/r>	Activate the board addressed **x**. **x** is between 1 and 64. Notes 1. Wait 10 ms after sending the colon (':') before sending the rest of the command ('**x**<c/r>'). 2. Wait 30 ms after sending the ':**x**<c/r>' command before analyzing the response(s).
:*999*<c/r>	"Hello" command: prompt all existing boards to respond. Notes 1. Wait 10 ms after sending the colon (':') before sending the rest of the command ('999<c/r>'). 2. Wait 1000 ms after sending the ':999<c/r>' command before analyzing the response(s).
n	Read the RS485 address of the active board. Board's response: '**#x**<c/r>', where **x** is the address of the active board. **x**=0 means that the board has been configured as a non-RS485 device. Other value (between 1 & 64) specifies the RS485 address of the active board. Notes 1. Wait 1 second after sending the 'n' command. 2. If no board is active, there will be no response. 3. In case of malfunction, more than one '**#x**<c/r>' will be responded in sequence. This is theoretically impossible but should be checked in order to be on the safe side.
N**x**<c/r>	Set the required RS485 address. **x** is between 0 and 64. Board's response: 'OK<c/r>'. (**x**=0 sets the board as a non-RS485 device; this will facilitate your coding, as no '**:x**<c/r>' command will be needed.) Notes 1. **The address setting takes effect only upon board reset.** 2. When the RS485 address is not 0, its value is shown for a while on the LED display upon board reset. 3. See the above note #3.

i. Misc.:	
z **(lower case z)**	Manually zero the gross weight. The effect of this function is temporary — it expires upon card reset. Response (versions 1.12, 3.09, 6.01, 7.00 and up): 'z'<c/r>.
Z **(Upper case Z)** (versions 1.12, 3.09, 6.01, 7.00 and up)	Cancel the manual zero operation (the lower case 'z'). That is, return to the original calibration zero. Response: 'Z'<c/r>.
S **(upper case S)**	Board reset (software reset to perform a restart as if board is turned off/on). Response (versions 1.12, 3.09, 6.01, 7.00 and up): 'S'<c/r> - possibly followed by additional binary character(s). **Wait 6 seconds before accessing the board again.** Issue this command after changing any of the following parameters: * Auto transmit interval (parameter #20). * Filters & Decimator (parameters #23-25). * General Setpoints (parameters #101-104). * Output modes (parameters #111-114). Note A board reset is available also by a **hardware input**. Refer to appendix I.
F **(upper case F)**	Start Fill-mode.
V **(upper case V)**	Get version no. of DSP software. Board's response: "LCIC-WIM/V#.##", where #.## = Version Number.
/ **(slash)**	In board versions: 1.22 and up 3.116 and up 4.272 and up (only in general mode): Shows on board's LED "APPL----" and "----APPL" (alternately) during 10 seconds. Response: '/' + C/R. The '/' command is useful when one PC is connected to **some** LCIC-WIM boards, and the user needs immediate identification to which of his boards the application's port is connected. For example, the user is connected to port 3 by a terminal. He sends the '/' command and sees which board shows the "APPL----"/"----APPL" indication on its LED. ('APPL' stands for 'APPLication'.)
* **(star)**	Resumes RS232/RS485 communication: As specified in section D.4, after PC power on or off the serial communication (RS232/RS485) is likely to drop. One solution is card reset. Alternatively, you may utilize the fact that the collapse is only in RS232/RS485 while the USB remains unharmed: Temporarily communicate the card through a USB port and send the '*' command, either in the general mode or in the inner mode (e.g., WiM-mode). The advantage of this option is that you remain inside the mode (e.g., WiM-mode): You do not lose board's log and the continuity is kept. Response: '*' + C/R.

Summary of Weight & A/D Reading Commands

		Reading Type			Filtering Level	
			Weight			
		A/D	Rounded to Resolution	Not Rounded	Filter1	Filter2
Single Reading	.			v		v
	?		v			v
	>	v			v	
	<	v				v
Fast Mode	u			v	v	
	U			v		v

4.2 Parameters

Parameter number	Parameter Description	Data Type
1	Load Cell mV/V (1, 2 or 3. 0 = unknown).	Float
2	Units: 0=g, 1=kg, 2=ton, 3=oz, 4=lb.	Float
3	Full Load Cell(s) Capacity	Float
4	Maximum Applied Capacity	Float
5	Resolution Index (0-17) Resolution Index is actually the index to an array of 18 defined values(0-17) like that: 0.0001, 0.0002, 0.0005, 0.001, 0.002, 0.005, 0.01, 0.02, 0.05, 0.1, 0.2, 0.5, 1, 2, 5, 10, 20, 50. E.g., if Resolution Index=8, then system resolution = 0.05.	Float
8	Calibration Date formatted as "MMDDYY". So, 10107 <= Calibration Date <= 123199.	Float
11	Calibration Time formatted as "HHMM". So, 0 <= Calibration Time <=2359	Float
20	Auto transmit interval (3-52734, integer). **How many internal adc updates (52734 Hz) there are between auto transmissions.** This gives a theoretical reading rate from 17578 per sec to 1 per sec. (i.e. rate=52734/P20). Practically, the actual rate for low values of P20 is usually less than the theoretical rate. (“P20” stands for “Parameter #20”.) Note that this rate relates to USB communication; upon using a serial port, the rate is far smaller. **Becomes effective only after a board reset (either power off/on or using the ‘S’ command).**	Float

23	Filter1 value: 2-256. **Becomes effective only after a board reset (either power off/on or using the 'S' command).**	Float
24	Filter2 value: 2-256. **Becomes effective only after a board reset (either power off/on or using the 'S' command).**	Float
25	Decimator: 2-1000. **Becomes effective only after a board reset (either power off/on or using the 'S' command).**	Float
101	General Setpoint1. **Becomes effective only after a board reset (either power off/on or using the 'S' command).**	Float
102	General Setpoint2. **Becomes effective only after a board reset (either power off/on or using the 'S' command).**	Float
103	General Setpoint3. **Becomes effective only after a board reset (either power off/on or using the 'S' command).**	Float
104	General Setpoint4. **Becomes effective only after a board reset (either power off/on or using the 'S' command).**	Float
111	Output1 Mode 0=manual, 1=general setpoint. **Becomes effective only after a board reset (either power off/on or using the 'S' command).**	Float
112	Output2 Mode 0=manual, 1=general setpoint. **Becomes effective only after a board reset (either power off/on or using the 'S' command).**	Float
113	Output3 Mode 0=manual, 1=general setpoint. **Becomes effective only after a board reset (either power off/on or using the 'S' command).**	Float
114	Output4 Mode 0=manual, 1=general setpoint. **Becomes effective only after a board reset (either power off/on or using the 'S' command).**	Float
115	Baud rate for the RS232/RS485 port. Possible values: 19200, 28800, 38400, 57600, 115200. Note The baud rate for the USB need not be defined – its upper limit is 921,600.	Float
301	RS485 Address. Possible values: 0-64. → **The address setting takes effect only upon board reset.** Refer to section 3.3.2.1.3.	Float

304	Duration of long activation of digital input #4. Refer to appendix I.	Float
305	Duration of long activation of digital input #3. Refer to section H.1.	Float
311	Timeout (in ms) to the interval between two 'doubled' characters in RS485 communication. Refer to section 4.4.2.2.	Float
321	SCI Refresh Rate. Refer to section D.6.	Float
1024-1034	Calibration Name (32 characters max.) (Organised 3 characters per location; in case the length is less than 32, the last character is followed by a binary zero byte.)	Float
1040	Get Results Immediately (GRI). (See section 3.3.2.1.4). Read the parameter (let's call it P). If you want to turn GRI **ON**: If P >= 4 then GRI is already on - nothing to do. Otherwise, add 4 to P (that is, P = P + 4). If you want to turn GRI **OFF**: If P < 4 then GRI is already off - nothing to do. Otherwise, subtract 4 from P (that is, P = P - 4). **The change takes effect only upon board reset.**	Float
1047	Auto Zero / Activate: 0 = Not active, 1 = Active. Default: 0. See section 3.3.2.2.1.	Float
1049	Auto Zero / Max. Zero: Min. = 0, Max. = 1000. Units: Weight unit (e.g., kg). Default: 10*Resolution. See section 3.3.2.2.2.	Float
1048	Auto Zero / Min. Zero: Min. = -10000, Max. = 0. Units: Weight unit (e.g., kg). Default: -10*Resolution. See section 3.3.2.2.3.	Float
1050	Auto Zero / Time limit: Min. = 1, Max. = 1000. Units: Sec. Default: 5. See section 3.3.2.2.4. If Time limit < 3 second, the 'auto zero' operation is **not** shown on the LED, but it does take place	Float
1053	Analog Output Max Voltage	Float
1054	Analog Output Min Weight	Float
1055	Analog Output Max Weight	Float
1059	Analog Output Mode (0 = Manual, 1 = Auto)	Float
1066-1069	Card Serial Number (12 characters max.) (Organised 3 characters per location, in case the length is less than 12, the last character is followed by a binary zero byte.)	Float

4.3 LCIC-WIM ActiveX

Unless otherwise specified, a function returns a Boolean: True for success, or False for failure.

4.3.1 Start/Stop Communication

Is_LCIC_WIM_Port(***CommPortNumber***)

Returns:
0 if the port does not respond.
1 if the port responds but not as an LCIC-WIM.
2 if the port responds as an LCIC-WIM.
CommPortNumber (Integer): Number of communication port.

OpenLCIC_WIM(***CommPortNumber, Baud_Rate***)

Opens the specified port.
CommPortNumber (Integer): Number of communication port.
Baud_Rate (Integer): Required Baud Rate (110, 300, 1200, 2400, 4800, 9600, 14400, 19200, 28800, 38400, 57600, 115200, 230400, 460800 or 921600).
(Refer also to sections 3.3.1.2 & D.2.)
Note: Using USB, all these values of *Baud_Rate* are relevant (even though usually 921600 is used). However, upon using serial communication, board's possible b/r values are limited to the range 19200-115200; please refer to section 3.3.1.2.

CloseLCIC_WIM()

Closes the open port.

Get_First_Free_LCIC_WIM_PortNumber()

Returns the number of the first port (from COM1 to COM16) that responds as an LCIC-WIM board.
If none is detected, 0 is returned.

4.3.2 Variables

The system has **variables** with which the user may adjust the system to his needs and communicate with the I/O. Actually these variables consist of parameters, inputs and outputs. A variable may be read and sometimes also may be written. The table below lists the variables, describes them and specifies which of them may be also written.

The methods to read and write a variable are:

Read:

Get_Variable(**r*<Variable Name>***)
Returns a string with the value of the variable.
r*<Variable Name>* is the variable name as it appears in the table below, prefixed by 'r' for **r**ead, e.g., rOutput_1_Mode.

Write:

Set_Variable(***w<Variable Name>***, ***Value***)
w<Variable Name> is the variable name as it appears in the table below, prefixed by 'w' for **w**rite, e.g., wOutput_1_Mode.
Value is a string with the value of the value to be written to the variable.

Variables Table

Category	Variable Name	Description	Get	Set
Calibration Info	Calibration_Name	Name of calibration	V	
	Calibration_Date	Calibration date (MMDDYY).	V	
	Calibration_Time	Calibration time (HHMM, e.g., 1545).	V	
	Unit	Weighing unit: ton, kg, g, lb or oz.	V	
	Resolution	Weighing resolution: 0.0001, 0.0002, 0.0005, 0.001, 0.002, 0.005, 0.01, 0.02, 0.05, 0.1, 0.2, 0.5, 1, 2, 5, 10, 20 or 50.	V	
	Full_Capacity	Full capacity of the load cell.	V	
	Maximum_Load	Maximum applied load on the load cell.	V	
	Load_Cell_Output	Output of the load cell: 0 = Unknown 1, 2 or 3 = 1, 2 or 3 mV/V.	V	
Filtering	Filter1	There are **two** filters. The ***first filter*** is basically a 1st level moving average filter of size **Filter1** (2-256). Then depending on **Decimator** (2-1000), every Decimator-th result from the 1st level filter is put through another moving average filter of size **Filter2** (2-256), which is the ***second filter***. * **Setting** these three variables causes a **board reset**, so it is a **slow** function. Therefore, in order to set **all of them** there is one more option: the method Set_Filtering , which works faster than three individual activations of the Set_Variable method.	V	*
	Filter2		V	*
	Decimator		V	*

Variables Table (cont'd)

Category	Variable Name	Description	Get	Set
Analog Inputs	Weight_Native	Current weight after Filter2, not rounded.	V	
	Weight_Rounded	Current weight after Filter2, rounded to resolution.	V	
	A2D_F1	Current A/D after Filter1.	V	
	A2D_F2	Current A/D after Filter2.	V	
	Temperature	The temperature measured on the board (°C).	V	
Digital Outputs	Output_1_Mode	0 = Manual, 1 = General Setpoint.	V	V
	Output_2_Mode		V	V
	Output_3_Mode		V	V
	Output_4_Mode		V	V
	Output_1_Status	0 = off, 1 = on. * The 'Set' is relevant only if the corresponding Output_**x**_Mode is 'Manual'.	V	*
	Output_2_Status		V	*
	Output_3_Status		V	*
	Output_4_Status		V	*
	Output_A_Status	Status of **all** outputs at once: $\#_4\#_3\#_2\#_1$ $\#_x$ = Status of output x: 0 = off, 1 = on (e.g., 0101).	V	
General Setpoints	Setpoint_1	The weight limit for output x. Relevant only if the corresponding Output_**x**_Mode is 'General Setpoint'. Irrelevant in Fill mode.	V	V
	Setpoint_2		V	V
	Setpoint_3		V	V
	Setpoint_4		V	V

Variables Table (cont'd)

Category	Variable Name	Description	Get	Set
Analog Output Relevant only for boards that include the 'analog output' option. ***(The analog output voltage is measured at pin 12 of CONN6 with respect to pin 13 which is ground.)***	Analog_Output_ Mode	0 = Manual, 1 = Auto.	V	V
	Analog_Output_ Level	Voltage in the analog output, in volts (0 – 2.5). * The 'Set' is relevant only if Analog_Output_Mode is 'Manual'.	V	*
	Auto_Hi_Voltage	When Analog_Output_Mode = 'Auto': The voltage to be supplied for weight=Auto_Hi_Weight (in volts, up to 2.5).	V	V
	Auto_Lo_Weight	When Analog_Output_Mode = 'Auto': The weight for which 0V should be supplied.	V	V
	Auto_Hi_Weight	When Analog_Output_Mode = 'Auto': The weight for which maximal voltage (=Auto_Hi_Voltage) should be supplied.	V	V
Digital Inputs	Input_1_Status	0 = off, 1 = on.	V	
	Input_2_Status		V	
	Input_3_Status		V	
	Input_4_Status		V	
	Input_A_Status	Status of **all** inputs at once: $\#_4\#_3\#_2\#_1$ $\#_x$ = Status of input x: 0 = off, 1 = on (e.g., 0101).	V	
	Toggling_Counter	A 16 bit counter that increments when opto input #2 is toggled.	V	

Variables Table (cont'd)

Category	Variable Name	Description	Get	Set
Fast Mode	FM_Updates	FM_Updates = Each how many internal updates there will be a Fast Mode transmission (3 – 52,734, integer). The frequency of the internal updates is 52,734 Hz. So, the theoretical reading rate is from 17578 per sec to 1 per sec. (52734/FM_Updates). Practically, the actual rate for low values of FM_Updates is usually less than the theoretical rate. Notes: 1. This rate relates to USB communication; upon using RS232, the rate is far smaller. 2. The Fast Mode is not available with RS485.	V	V
Misc.	Version_ID	"LCIC-WIM/V#.##" #.## = Version Number.	V	
	Serial_Number	Card's serial number.	V	

4.3.3 Filters

Set_Filtering(***Filter1, Filter2, Decimator***)
Filter1 (Integer): 2 – 256 or 0.
Filter2 (Integer): 2 – 256 or 0.
Decimator (long): 2 – 1000 or 0.

(Refer to the 'Filtering' square in the variables table above.)

Set_Filtering supplies a faster way to change the filtering parameters when more that one of them has to be changed, as the change operation causes **board reset** which is time consuming; individual activations of Set_Variable would require this time more than once. **Specify '0' for a parameter that needs no change.**
For example, in order to set Filter1, Filter2 & Decimator to 11, 22 & 33, respectively, apply Set_Filtering(11, 22, 33). Now in order to change Filter1 & Filter2 to 10 & 20, respectively, and leave Decimator unchanged, apply Set_Filtering(10, 20, 0). Finally, in order to leave both Filter1 & Decimator unchanged and set Filter2 to 2, apply either Set_Filtering(0, 2, 0), or Set_Variable(wFilter2, 2).

Get_Filtering(***Filter1, Filter2, Decimator***)
Filter1 (Integer): 2 – 256.
Filter2 (Integer): 2 – 256.
Decimator (long): 2 – 1000.

(Refer to the 'Filtering' square in the variables table above.)

The Get_Filtering is functionally equivalent to
Filter1 = Get_Variable(rFilter1)
Filter2 = Get_Variable(rFilter2)
Decimator = Get_Variable(rDecimator)
and has no time advantage, as the get operation does *not* cause a board reset; the Get_Filtering function has been supplied just for symmetry with Set_Filtering.

4.3.4 Fast Mode

(The Fast Mode is not available with RS485.)
During the Fast Mode there is auto high speed transmission of weight readings to the communication. About the transmission rate, refer to the 'Fast Mode' square in the variables table above.
At the end, a 'timer stamp' is appended. Its value is the time elapsed from start of transmission until end of transmission, in ms.
The readings that are returned in this mode are always integers and they equal the actual weight multiplied by 1, 10, 100, 1000 or 10000, depending upon the resolution setting. So, although the readings are integers, due to that multiplication the original precision is maintained. The readings are *not* rounded to the resolution. For example, if the resolution is 0.05, then the readings transmitted by the board will be multiplied by 100, so that '123' will represent '1.23'.

Start_Fast_Mode(***Filtering_Level***)

Starts the Fast Mode.
Note: During the Fast Mode the LED Display is *not* updated.

Filtering_Level (Integer):
1 = Supply readings after Filter1.
2 = Supply readings after Filter2.

Stop_Fast_Mode()

Stops the Fast Mode.

(A timer stamp is appended. Its value is the time elapsed from start of transmission until end of transmission, in ms.)

The mechanism to receive the data uses events and methods as described below:

The transmission sends blocks of information.

Stage 1

Except the last one, each block generates the event **DataArrivalInFastMode**. When the event occurs, run the method **Get_CurrentBlock** to read the current block. The block consists of *integer weights* separated by a Carriage Return. At this stage, just store the blocks into a string array. This stage repeats until the last block arrives. That is, if there are 10 blocks, then 'stage 1' occurs 9 times.

Stage 2

The last block generates the event **DataArrivalLastInFastMode**. Run the method **Get_LastBlock** in order to read the last portion of the *integer weights* and store them too in the string array used in stage 1. Run the method **Get_Time_ms** in order to get the time stamp.

Stage 3

After the last block was received and stored, the weights may be processed:
1. Recalling that the values are separated by Carriage Return, parse the string array and keep the individual values in a numeric array. One clean way to do that is write the array to file by Print and read back the file using Input. Note that a value may be split between two blocks, e.g., the value '123' may appear as '12' in the end of one block and '3' in the beginning of the next block. The above way using a file handles the parsing well.
2. As described in the beginning of this section, the values are integers that were accepted by multiplying the actual weight by 1, 10, 100, 1000 or 10000. You may find the actual weight by multiplying the integer weight by a "Resolution_Factor" which is 1, 0.1, 0.01, 0.001 or 0.0001, respectively. You may find the Resolution_Factor yourself, but for your convenience there is the method Get_Resolution_Factor which returns the proper value.

How to work with the Fast Mode in VB using the ActiveX

During the Fast Mode process the board transmits mass data to the PC. Therefore, in order to avoid data loss, all the actions on your PC should be minimized.

1. Define string Array
   Dim Fast_mode_Data(1 to SizeOfArray) as String
   Dim fmCounter as long ' Current counter (index)

2. Select Filter :
   Filter=Filter2

3. To start the Fast Mode:
   Call LCICwim_commands1.Start_Fast_Mode(Filter)

4. In the event DataArrivalInFastMode:
   ' Get current block:
   fmCounter = fmCounter + 1
   Fast_mode_Data(fmCounter)=
   LCICwim_commands1.Get_CurrentBlock

   So, all arrived data are stored in the array Fast_mode_Data

5. To terminate the Fast Mode:
   Call LCICwim_commands1.Stop_Fast_Mode
   Except the last one, each block will still cause a
   DataArrivalInFastMode event, as described in para. 3.

   In the DataArrivalLastInFastMode event that the last block
   will cause:
   fmCounter = fmCounter + 1

   Fast_mode_Data(fmCounter) =
   LCICwim_commands1.Get_LastBlock
   TotalTimeInFastMode =
   LCICwim_commands1.Get_Time_ms

Interpreting the data in a block:

Each block includes integer weight values separated by a c/r .
In order to get the real weight values, the integer values should be multiplied by the current **Resolution Factor** (for details refer to the end of stage 3, above). You may get the current Resolution Factor using the method:
LCICwim_commands1.Get_Resolution_Factor

4.3.5 Misc.

Apply_Temporary_Zero()
Manually zero the gross weight. The effect of this function is temporary — it expires upon card reset.

Reset_Board()
Resets the board. Usually this function is not required.

4.4 RS485

4.4.1 Introduction

Using RS485 there is a higher probability for electronic noise. Therefore, there is the 'double' mechanism which supplies a protection against that noise, yielding a higher reliability in the communication. The principle of the mechanism is that part of the characters sent from the PC to the board are 'doubled'; that is, repeated at least once (the exact quantity of repetitions varies from one PC to another and should be found by 'Trial & Error'). For example, the original command 'R25' might be 'doubled' to 'RR25' or 'RRR25'. '?' might look like '??' or '???', etc. The exact 'doubling rules' are described below.

4.4.2 Doubling Rules

4.4.2.1 Which character should be doubled?

Only **part** of the characters should be doubled, depending on board's mode:

4.4.2.1.1 In General Mode:

Only the ':' character of the RS485 address handling commands should be doubled (refer to section 4.1, square 'h' in the table).

4.4.2.1.2 Inside the inner mode (LiW-mode, CHW-mode, etc.):

Only the first character of a command should be doubled.
To make it clear – the definition '**first character of a command**' includes:
* The ':' character as in General Mode.
* One character commands.

The 'V' command is an exception – no doubling is required.
Thanks to this exception, the syntax of the 'V' command is the same both in the general mode and in the inner mode. That enables you any time send 'V' in order to find out board's mode.
For example:
Response = "LCIC-WIM/V5.20" indicates the board is in general mode.
Response = "CHW-mode" indicates the board is in CheckWeigher mode.

Examples

Description	Command Syntax	Mode	
		General	Inner
Doubled in both modes	::1	+	+
Not doubled in general mode	R25	+	
Doubled in inner mode	RR25		+
Not doubled in general mode	?	+	
Doubled in inner mode	??		+
Not doubled in both modes	V	+	+

4.4.2.2 Is a doubled character limited in time?

Yes, there is a timeout to the interval between two characters:
The doubled character should be received within, say, 1 second after the previous character arrived at the board. The size of this timeout (in ms) is determined by Parameter #311. For example, in order to define the timeout as one second, set Parameter #311 to 1000.
→ Parameter #311 is accessible by the 'R' and 'W' commands (refer to section 4.1, square 'a' in the table).

4.4.3 Tips

4.4.3.1 Baud Rate

Best reliability is achieved with B/R=19200.
However, a higher B/R is possible, though not recommended.

Appendix A: I/O & the LED Display

A.1 General Notes about the I/O

*** The digital I/O is available on CONN6 (15 pin Dsub).**

*** Digital Outputs**

The outputs are opto-isolated 300mA 50V solid state relays. When activated (status LED is on), they switch the OUTPUT x (x=1,2,3, or 4) to I/O VOLTAGE OV. Hence the load would normally be connected between OUTPUT x and the I/O VOLTAGE+.

*** Digital Inputs**

The digital inputs are designed to work with either npn or contact input devices. They are activated by an external device pulling INPUT x (x=1,2,3,or 4) down to I/O VOLTAGE OV.
To work in this way an external IO VOLTAGE+ (in the range 10-30V) must be present.

*** Analog Output (option)**

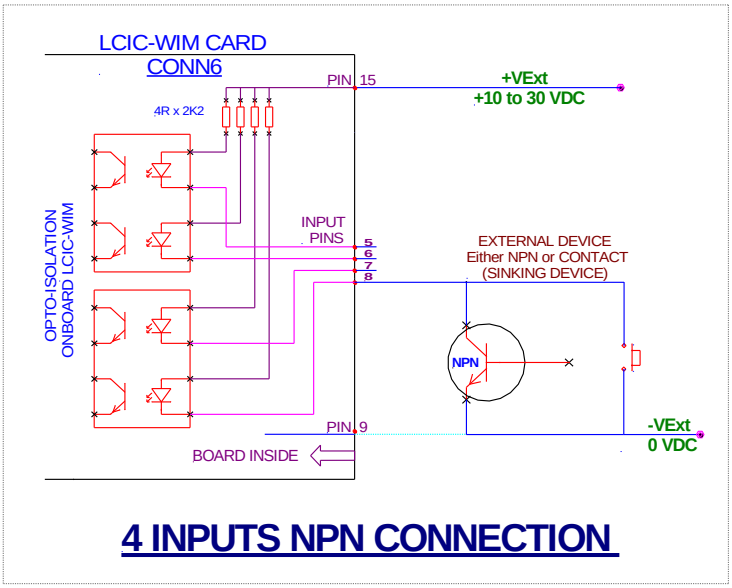
The analog output signal is set by a 16 bit DAC and appears on pin12 of CONN6. The output is with respect to the LCIC_WIM board **ground** (**not** the I/O 0V). The board ground appears at pin13 of CONN6. The analog output is application accessible.

*** Connections**

The following table shows the I/O pinout:

Pin	Function
1	Output 1
2	Output 2
3	Output 3
4	Output 4
5	Input 1
6	Input 2
7	Input 3
8	Input 4
9	I/O Voltage 0V
10	NC
11	NC
12	Analog Out Signal (Option)
13	Analog Out Gnd
14	NC
15	I/O Voltage+ (10 to 30V)

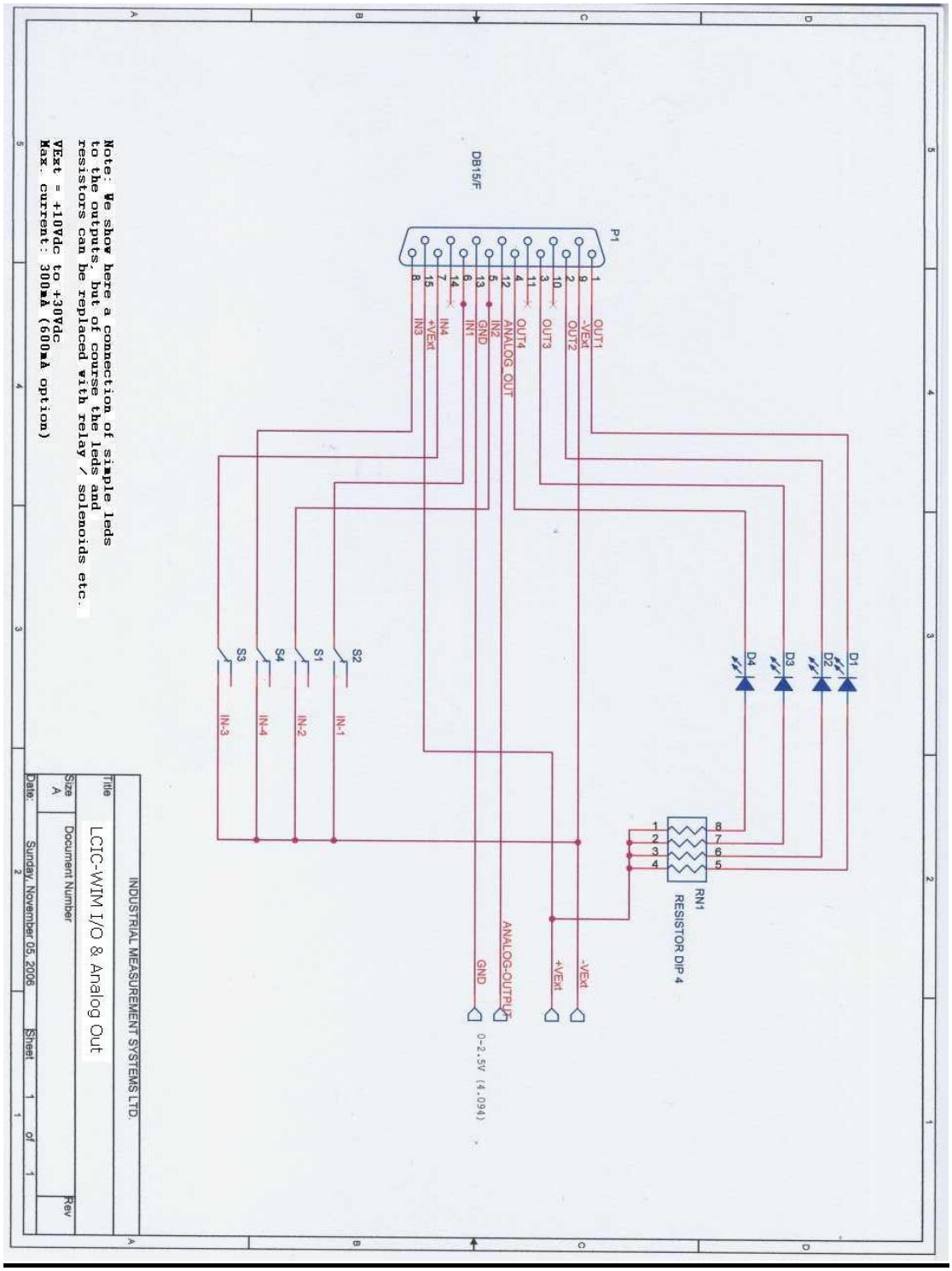
A.2 Connecting External Devices to the Inputs



NOTE: PNP DEVICES ARE NOT SUPPORTED

Title: Connecting External Devices to the LCIC-WIM Inputs (Only one shown)		Industrial Measurement Systems Ltd. P.O. BOX 6305 HAIFA 31062 ISRAEL TEL: 972-4-8110877 FAX: 972-4-8110875 www.ims.co.il E-mail: sales1@ims.co.il
Client		
Filename: Input Devices conn	Drawing No.	Sheet of

A.3 LCIC-WIM I/O & Analog Output



A.4 The LED Display

Upon **board restart**, the two following messages are shown on the LED display – each for a while:

LCIC **x.xx**	**x.xx** is board's DSP version.
Sb **yyy**	**yyy** is current board's Serial baud-rate (refer to sections 3.3.1.2 & D.2).

Then the display shows the current data.

Notes

1. The weight on the LED display is always after Filter2 (refer to section 3.3.2.4).

2. In **Fill-mode**, the LED display shows additional information – refer to section E.5.

3. During the Fast Mode (section 4.3.4) the LED display is *not* updated.

Appendix B: Scaling the Load Cell Input

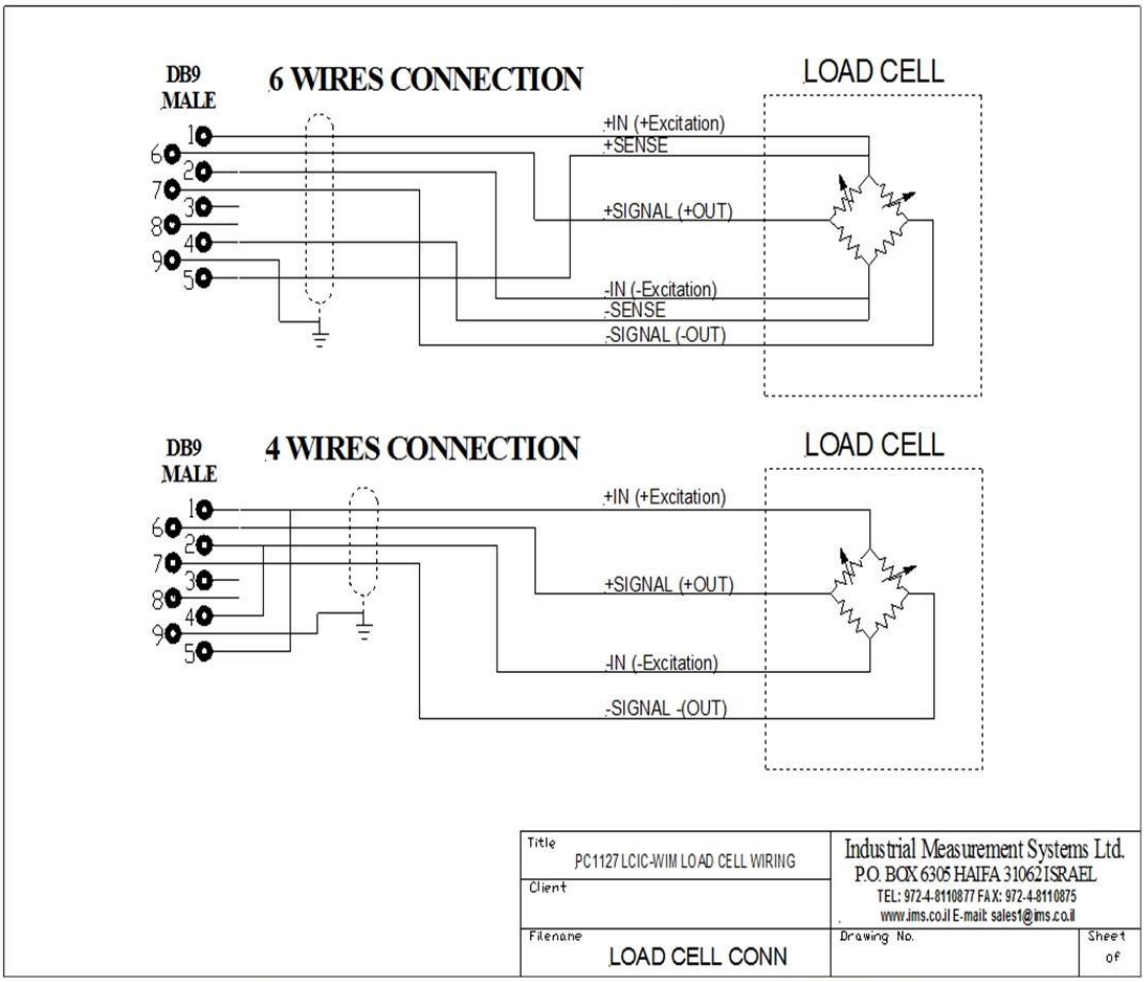
The full scale of the input coming from the load cell may be adjusted by the LK4 jumper (which is next to load cell connector):

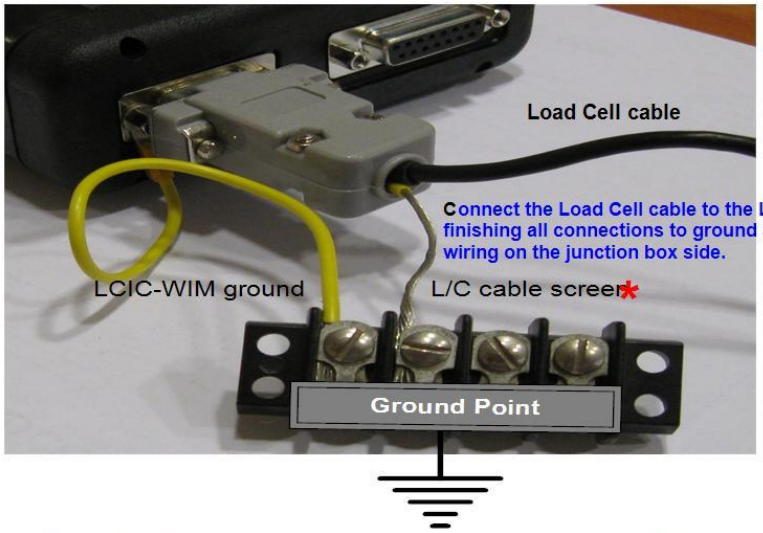
- Across the two **leftmost** pins (default):
  Load cell output is 1-2mV/V.

- Across the two **rightmost** pins:
  Load cell output is 3mV/V.

Appendix C: Load Cell Wiring

LCIC-WIM Load Cell wiring





*** Important:** Cable's screen must be connected to earth/ground only at **one** side of the cable - next to the LCIC-WIM, and never on the Load cell side.

L/C Connector with cable - color code: +Ext=Green -Ext=Black +Signal=Red -Signal=White +Sense=Blue -Sense=Brown

Appendix D: USB, RS232 & RS485

In addition to USB, The LCIC has an option for both full-duplex RS232 and half-duplex RS485 interfaces. These are brought out on CONN3, a 9 way 'D' type connector. The pin-out is as follows:

CONN3 PIN	FUNCTION
1	RS485-
2	RS232 TX (out)
3	RS232 RX (in)
4	NC
5	SIGNAL GROUND
6	RS485+
7	NC
8	NC
9	NC

i.e., for RS232 use pins 2,3 & 5 and for RS485 use pins 1,5 & 6
For RS232 connection, a standard direct (straight through) wired 9D M-F cable can be used for direct connection to a standard 9 way 'D' type PC COM port.

RS485 line termination – placing a jumper across the two rightmost pins of LK1 puts a 120 Ohm (ac coupled) impedance across the RS485 data lines. This should be used in single LCIC RS485 applications. In applications where multiple LCIC boards reside on one RS485 bus, the termination impedance should only be added on the last board on the bus.

D.1 Communication Parameters

For both the USB, the full-duplex RS232 and half-duplex RS485 interfaces, the communications parameters are fixed as follows:

Data 8 bits
Parity None
Stop Bits 1
► This is generally referred to as "8,N,1".

D.2 Baud Rate

For the **USB**, the maximal baud rate is 921,600. The board responds well without a need to pre-define the used b/r.
For **serial communication**, the required b/r should be pre-defined by the user via the Settings utility (section 3.3). The available baud rates are between 19,200 and 115,200. The current b/r used by the board for the serial communication is shown for a while on the LED display upon card reset, prefixed by 'Sb' (=Serial baud rate). Refer also to sections 3.3.1.2 & 3.3.2.1.2.

D.3 General RS232 Terminal

You may talk with the card either by your own application or by a general RS232 terminal. One simple one called Termite is available for free at http://www.compuphase.com/software_termite.htm

D.4 Serial Communication & PC Power On/Off

After PC power on or off the serial communication (RS232/RS485) is likely to drop. There are two options to resume the communication:
1. Restart the card.
2. Send the '*' command through a USB port
   (refer to section 4.1, square 'i' in the table).

D.5 RS485

Up to 64 LCIC-WIM boards may be connected to one PC port.
In the **PC side**, use a converter either from the RS232 port, or from the USB port (that is, RS232 to RS485 converter, or USB to RS485 converter). In the **board side**, use the serial port (CONN3) – refer to the table in the beginning of this appendix.
Using the LCIC-WIM-SETTINGS utility, each board is assigned a unique address between 1 and 64 – refer to section 3.3.2.1.3 (**never give the same address to more than one board**).
An RS485 board may be either 'active' or 'inactive'. An 'active' board honours all commands, while an 'inactive' board honours only the RS485 address handling commands. Upon board reset it 'wakes up' inactive. A board **starts** being active when it receives **its** 'activate board' command and **stops** being active when an 'activate board' command to any other board is sent. Hence, only one board (at most) may be active at the same time.
About the handling of the RS485 address while coding your application please refer to the RS485 Commands and RS485 Responses sections below, which expand the summary given in section 4.1/h ('RS485: Address selection, setting & reading').
→ When a board is configured as an RS485 device, its address is shown for a while on the LED display upon board reset, for example "rS.485-18" for the board addressed 18.

RS485 Commands

Except 'Nx<c/r>' (paragraph d), these commands are available also in the fill mode.

a. Activate address x (x=1, 2, 3, …, 62, 63, 64):

:**x**<c/r>

Board(s) response:

1. If address **x** is already active:
   !**x**<c/r>
   (The board reports that it is already active and has nothing to do.)

2. If address **x** is not active:
   2.1 If another address is currently active:
   ^**x**<c/r> *(I'm going to sleep)*
   (The 'falling' board reports that it received an activation command to another board, then it makes itself inactive.)
   2.2 If address **x** exists:
   A**x**<c/r> *(I'm becoming active)*
   (The 'rising' board reports that it makes itself active.)

(Please refer to the notes on the next page.)

Notes

1. Wait 10 ms after sending the colon (':') before sending the rest of the command ('**x**<c/r>').

2. Wait 30 ms after sending the ':**x**<c/r>' command before analyzing the response(s).

3. There might be four cases with the conditions of 2.1 & 2.2:
   #1: Both conditions are true:
   Both responses will be transmitted –
   first '^**x**<c/r>' and then 'A**x**<c/r>'.
   (Old address retired and address **x** became active.)
   #2: None of the conditions is true:
   There will be no response.
   (No address was active before, none is active now.)
   #3: Only 2.1 is true:
   '^**x**<c/r>' will be transmitted.
   (No address is active, as old address retired and the new one does not exist.)
   #4: Only 2.2 is true:
   'A**x**<c/r>' will be transmitted.
   (Address **x** became active.)

4. Verify normal address switching by the 'n' command. In case of unexpected response to the 'n' command, repeat the ':**x**<c/r>' command. In case #1 of note 3 (which is the most frequent) this verification is not needed, simplifying the switching procedure.

b. Hello:
:999<c/r>

The 'hello' command is like a 'who is alive?' question.
This is useful in order to detect which addresses exist in the system.
Each existing board, whether active or not, responds '*x<c/r>', where **x** is its address.

Notes

1. Wait 10 ms after sending the colon (':') before sending the rest of the command ('999<c/r>').

2. The '*x<c/r>' responses will be transmitted in sequence.
   That is, if all 64 addresses exist, then first '*1<c/r>' will be transmitted, then '*2<c/r>' and so on, and finally '*64<c/r>'.

3. The 'hello' command does not change the 'active' mode of the address – it will remain the same as before.

4. After sending the 'hello' command, wait 1 second to give chance to all 64 potential addresses to respond.

c. Read the RS485 address of the active board:
n

Board's response:
'#**x**<c/r>', where **x** is the address of the active board.
x=0 means that the board has been configured as a non-RS485 device.
Other value (between 1 & 64) specifies the RS485 address of the active board.

Notes

1. After sending the 'n' command, wait 1 second to give chance to all 64 potential addresses to respond.

2. If no board is active, there will be no response.

3. In case of malfunction, more than one '#**x**<c/r>' will be responded, in sequence. This is theoretically impossible but should be checked in order to be on the safe side. In this case re-activate the required address by the 'activate address' command (paragraph a).

d. Set the specified RS485 address:
N**x**<c/r>

This command sets the board's RS485 address. **x** is between 0 and 64. (**x**=0 sets the board as a non-RS485 device; this will facilitate your coding, as no ':**x**<c/r>' command will be needed.) Note: When the RS485 address is not 0, its value is shown for a while on the LED display upon board reset.
Board's response: 'OK<c/r>'.
Normally a user application will never use this command, as the address setting is carried out using the LCIC-WIM-SETTINGS utility.

RS485 Responses

(Most responses are already described in the 'Commands' section.)

!**x**<c/r>	Address **x** is already active. See Commands/a/1.
^**x**<c/r>	Becoming inactive on behalf of address **x**. See Commands/a/2.1.
A**x**<c/r>	Address **x** becomes active. See Commands/a/2.2.
*__x__<c/r>	Address **x** is alive. See Commands/b.
?**x**<c/r>	A ':**x**<c/r>' command was received but **x** is illegal. That is, **x** is neither in the range (1,…,64), nor 999.
#**x**<c/r>	Address **x** is active. See Commands/c.

Examples

Suppose there are 3 boards in the system, addressed 1, 2 & 3.
(The blue text is the PC side, the red text is the response from the board(s), and the black text is our comments)

Example #1: Everything goes fine

:999
*1
*2
*3
n
(no response as no board is active)
:1
A1 Board #1 becomes active
n
#1
:2
^2 Responded by board #1. Means: I (board #1) am becoming inactive in favor of board #2, even though it's unknown for me whether board #2 exists or not.
A2 Responded by board #2. Means: I (board #2) am becoming active, assuming that no other board is active.

(Note: Comments analogue to the above two ones (about ^2 and A2) are relevant also for the other ^*x* and A*x* commands; however, for the sake of readability, the following comments are shorter.)

n
#2
:3
^3 Board #2 becomes inactive in favor of board #3
A3 Board #3 becomes active
n
#3
:1
^1 Board #3 becomes inactive in favor of board #1
A1 Board #1 becomes active
n
#1

So far everything was smooth; however, the quality of the communication depends – besides the board and the PC – also on the environment. Hence, there might be irregular situations that the user should know to handle – this is what the following two examples (#2 and #3) explain.

Example #2: No response from the **new** board

n
#1
:2
^2 Board #1 becomes inactive in favor of board #2

But the 'A2' response, telling that board #2 became active, did not arrive! Where is the disorder? Maybe board #2 did receive the command and is indeed active, just the 'A2' response was lost, and everything is OK (case #1); but maybe board #2 did not receive the command and no board is active (case #2). In both cases sending the ':2' command again will throw light on the situation: In case #1 '!2' will be responded telling that board #2 was already active; in case #2 'A2' will be responded telling that board #2 became now active. In both cases everything is ok and we may proceed. However, if there is no response at all on the ':2' command, repeat it say, 3 or 4 times and if there is still no response, then there is some severe problem requiring a human action, maybe a board reset should be applied.

Note that theoretically the 'n' command could be used as well, but practically it is recommended to avoid using the 'n' command – where possible – for some reasons:

1. The 'n' command is time consuming – it requires waiting 1 second in order to let all potential boards respond. (This is unavoidable as – by definition – the mechanism of the 'n' command takes into consideration also a faulty situation in which two (or even more) boards are active at the same time. This mechanism ensures that the responses will arrive in sequence and not simultaneously, therefore it consumes so much time.)
2. The conclusion derived according to the response on the 'n' command is not always clear. That might require sending the 'n' command again, requiring another 1 second.
3. The 'n' command is only informative, it does not fix anything. If it's possible – as in our case – to both fix a problem and get information at the same time, then it is preferable.

Example #3: No response from the **old** board

n

#1

:2

A2 Board #2 becomes active

But the '^2' response, telling that board #1 became inactive, did not arrive! Where is the disorder? Maybe board #1 did receive the command and is indeed inactive, just the '^2' response was lost, and everything is OK (case #1); but maybe board #1 did not receive the command and both boards are active (case #2). This is a dangerous situation and definitely should be avoided. This problem is more complicated than the previous one (in example #2) – sometimes re-sending the ':2' command won't help: In case #2 re-sending the ':2' command should be responded by '^2' which indicates that everything is OK. ('!2' will be responded as well, but this does not add us any new information.) However, if the '^2' response does not arrive, we have to use the 'n' command in order to make sure that we are not in the situation of two boards active at the same time, which should be avoided. Therefore, we should send the 'n' command and expect to get only '*2' (and not '*1').

Although it's time consuming, we have to repeat the 'n' command at least once in order to have no doubt that board #1 is really inactive.

Board selection by the supplied utilities

The three supplied utilities – LCIC-WIM-CALIBRATION, LCIC-WIM-SETTINGS & LCIC-WIM-MONITOR – enable easy selection of the required board:

- Upon program start, all detected addresses are reported. Verify that the total number of boards detected (reported at the bottom of the display) corresponds the real number. It might occur that the automatic detection fails to detect a board. If you know that some board does exist although it was not detected, click its address manually. Finally, select the board that will be activated first by right-clicking its address (or leave the default selection) and click 'Continue' (or just wait).
- When you are accessing some board, usually you may switch to another one. (In the LCIC-WIM-CALIBRATION this option is available only in step 1.) In order to switch the active board, right click the mouse. You'll get a list of all existing boards, with the active one dimmed and checked. You may select another board to be activated by clicking its address.

D.6 Automatic Refresh of the Serial Communication

(Availability: board version V3.114 and up.)

This feature enables automatic refresh of the serial communication in case the board 'feels' it dropped. The feature works both in general mode and in the inner mode (like WiM-mode). The condition according which the board decides that the communication dropped is described below:

If
last '*such and such*' successive seconds there was no communication input from the PC,
and
last communication input from the PC was **not** via the USB port,
then
the board refreshes the serial communication automatically.

'*such and such*' is user programmable by the 'SCI Refresh Rate' parameter.
There are two options to access this parameter:
* By LCIC-WIM-SETTINGS application V2.24 and up:
  In the 'Communication' frame, click on the 'Baud Rate' sub-frame, switching to the 'SCI Refresh Rate' sub-frame.
* By a terminal or a user application: Refer to address 321.
  If you change the parameter by this option, restart the board in order to validate the new value.

Parameter values:
parameter = 0: feature disabled.
1 <= parameter <= 100000: '*such and such*' = parameter (note that the units are seconds).
Default: parameter = 100; that is, the feature is enabled and '*such and such*' = 100 (seconds).

Notes:
1. Except the option described in next paragraph, the parameter should be an integer.
2. Optionally, you may get an **indication** on the LED display when the 'refresh' works.
This option is good only for testing, as during the indication time (one second) the board is idle.
Therefore, it should be used only temporarily and by an advanced user – it is **not** available via the LCIC-WIM-SETTINGS application.

To activate the 'indication' option:

* Use a terminal or a user application and refer to address 321.
* Set the parameter to a value more than 6, plus 0.5. For example, 10.5. (The '.5' is just a **mark** that you want to get the indication; the effective refresh rate will be 10 seconds, **not** 10.5.)
* Restart the board.
* Connect to board's serial port by an RS232 terminal but send nothing to the board.
* Each 10 sec. you'll get a 'refresh' indication: 'rEF-SCI' (**ref**reshing the **s**erial **c**ommunications **i**nterface), on board's LED display, during one second.
* If (for example) 6 seconds after a 'rEF-SCI' indication you send some command, say 'V', the next refresh will take place only 10 (**not 4**) seconds later. In other words, a refresh occurs only after there is a 10 successive seconds pause in communication input from the PC; otherwise, there will be no refresh.

Appendix E: Fill Mode

E.1 Introduction

The LCIC-WIM supplies a ***Fill Mode*** in which it may control a filling operation, using the *hardware inputs* (section E.2) and *hardware outputs* (section E.3). The character of the filling operation is determined by *parameters* (section E.4) set via the *Settings utility* (section 3.3). The filling results are shown on the *LED display* (section E.5). The Fill Mode supports also *commands* (section E.6) sent through the communication line (USB, RS232 or RS485). These commands may draw the filling results as well as send operational instructions, such as start a filling operation. Both the *LED display* and the communication line may indicate an *error code* (section E.7).
During the filling cycles the board learns the system in order to improve the results of the following cycles by compensating system's unavoidable inaccuracy. The *first* cycle is split into two parts (start-stop-start-stop) enabling the board to learn the system, thus apply already the above compensation, trying to achieve a correct result even in the first cycle.

E.2 Hardware Inputs

Preface

The filling mode has two operation sub-modes: **Manual** & **Auto**.
* In the **Manual** mode you may temporarily activate a valve (slow of fast) for your tests, adjustments etc.
* Use the **Auto** mode to (1) start/stop a controlled filling process, (2) switch the active setpoint (recall that there are 3 setpoint parameters), or (3) switch the board to the general mode.

<table>
<tr><th>Input #1</th><th>Input #2</th><th colspan="2">Input #3</th><th>Input #4</th></tr>
<tr><td rowspan="2">OFF = Manual Mode</td><td colspan="3">Manual activation of outputs:</td><td rowspan="2">N/A</td></tr>
<tr><td>ON/OFF = Turn output #1 on/off</td><td colspan="2">ON/OFF = Turn output #2 on/off</td></tr>
<tr><td rowspan="2">ON = Auto Mode</td><td rowspan="2">Momentary ON = Start</td><td>When in error status:</td><td>Momentary ON = Terminate the error status</td><td rowspan="2">Momentary ON = Emergency Stop</td></tr>
<tr><td>When <u>not</u> in error status and <u>not</u> during filling:</td><td>Long ON = Switching Mode:
* Switch to the visible setpoint by releasing input #3.
* Switch to the general mode by turning input #2 on for ~2 sec.
(See details on the next page.)</td></tr>
</table>

The Switching Mode

In this mode:
* 'SP' flashes on the left side of the LED display.
* The 3 setpoint parameters are shown, in turn, on the LED display.

In this state (while input #3 is still on) you have two options:

1. Switch a setpoint

You may **switch** the active setpoint:
When the desired setpoint is shown on the LED display, select it by releasing (turning off) input #3. The LED display will show 'SUCCESS' and then the board will return to regular auto-mode. You will be able to see the new active setpoint on the LED display (prefixed by 'SP'). If the 'get results immediately' option is selected, the board reports to the PC: 'Setpoint selected as…'.

2. Switch to general (non-fill) mode

Turn input #2 on for ~2 seconds.
The LED display will show 'Gen.ModE' and then the board will switch to the general mode.
Note: When the board switches to the general mode, make sure to turn input #3 off, otherwise the manual zero operation will take place – refer to section H.1 / Manual Zero / Hardware input.

E.3 Hardware Outputs

<table>
<tr><th>Output #1</th><th colspan="2">Output #2</th><th>Output #3</th><th>Output #4</th></tr>
<tr><td colspan="3">Option #1 (*)
Fast Valve = Output #1
Slow Valve = Output #2</td><td rowspan="7">Error</td><td rowspan="7">Continuous:
Filling Complete
(With board version 1.11 or higher.)

Flashing:
Scale is empty
(With board version 1.14 or higher.)

Refer to section E.4.4.</td></tr>
<tr><td>ON/OFF = Fast Valve is ON/OFF</td><td colspan="2">ON/OFF = Slow Valve is ON/OFF</td></tr>
<tr><td colspan="3">Option #2 (*)
Fast Valve = Output #1 + Output #2
Slow Valve = Output #2 only</td></tr>
<tr><td>OFF</td><td rowspan="2">ON=</td><td>Slow valve is ON</td></tr>
<tr><td>ON</td><td>Fast valve is ON</td></tr>
<tr><td>OFF</td><td rowspan="2">OFF=</td><td rowspan="2">Both valves are OFF</td></tr>
<tr><td>ON</td></tr>
</table>

(*) Options #1 and Option #2

Options #1 or **Option #2** is selected by the user when he sets the Filling Parameters; refer to section E.4.3 ('Fast Speed Config').

E.4 Filling Parameters

Notes
1. There is a set of three setpoints. Once they are specified (using the Settings utility), the user may switch to another setpoint without needing a PC. This gives more flexibility when several setpoints are needed. For details about the switching procedure refer to section E.2.
2. These setpoints are absolutely different from the four general setpoints (section 3.3.1.3). The general setpoints are inactive in the Fill mode.

Filling By

Weight
Means that **Setpoint** is defined in *weighing units*.

Time
Means that **Setpoint** is defined in *time units*.

E.4.1 Filling By = Weight

E.4.1.1 Filling by Weight Parameters

Setpoint #x
The required total filling weight when the user selects setpoint #**x** (**x** = 1, 2 or 3).

Slow Amount
The required slow filling weight, in % of the actual setpoint (a tip shows the value of **Slow Amount** in weight units).
When only one speed is required, specify '0' and use the 'Fast' output.

Filling Timeout
Time limitation for the filling process (in ms).
Note: The 'Filling Timeout' parameter is common for all the three setpoints. Therefore, specify a 'Filling Timeout' value large enough to cover all the setpoints used.

Auto Correction &
Averaging x last fillings
When 'Auto Correction' is checked, the board tries to correct the filling amount, based on the results of the last **x** fillings.

E.4.1.2 Auto Tare

Activate

- When *not checked*, the Setpoint defines the requested **final gross weight**.
  That is, if the setpoint is 100 kg and the starting gross weight is 90 kg, the filling amount will be 10 kg.
  The next two parameters (Hi/Lo Limits) are irrelevant.
- When *checked*, the Setpoint defines the requested **filling amount**.
  That is, if the setpoint is 100 kg, the filling amount will also be 100 kg, regardless of the starting gross weight.
  However, the filling operation will take place only if the starting gross weight, as found in accordance with the Stabilization Criterion (section E.4.1.4), is within the range defined by the following two parameters (Hi/Lo Limits).
  Otherwise, the filling operation will be rejected and an error will be reported.

Hi Limit
When 'Auto Tare' is activated, specifies the *high* allowed tare limit.

Lo Limit
When 'Auto Tare' is activated, specifies the *low* allowed tare limit.

E.4.1.3 Valid Results Limits

The resulting filling weight should normally be inside a user pre-defined 'valid range'. In case the weight exceeds that range, an error situation will occur. Specify 'Valid Limits (±)'= 0 if you don't need this check.

Start from filling # ...
Specifies the first filling to be checked, letting you disregard some filling until the board learns the system thus gives good results. For example, if 7 is specified, then no error situation will occur upon the first 6 fillings, whatever the results will be.

Valid Limits (±) ... %
Specifies the acceptable deviation for a valid result, as a percent of the actual setpoint (a tip shows the value of that acceptable deviation in weight units).
For example, if the actual setpoint = 100 kg and 'Valid Limits (±)' = 1%, then the result will be considered as 'valid' when it is in the range [99,101] kg. Otherwise, error #101 or #102 will occur. Specify 'Valid Limits (±)'= 0 if you don't need the 'Valid Results Limits' check.

E.4.1.4 Stabilization Criterion (Tare & Stop)

At the beginning and at the end of a filling cycle the board waits for the scale to stabilize in order to read its weight. Hence, some *stability criterion* is required. The board requires that all readings within 'Delta Time' will be inside a range whose width is 'Delta Weight', both at the beginning (Tare) and at the end (Stop) of the filling cycle. The waiting for that stabilization is limited by 'Timeout', causing an error situation in case of failure.

Delta Weight
See the description above.
The units of Delta Weight are **percent of the setpoint**.
When you put the cursor above the '%' sign, you get in a tip the Actual Delta Weight (in weighing units) calculated for the selected setpoint.

Delta Time
See the description above.

Stabilization Timeout
See the description above.

Impacts
The system copes with impacts and automatically resumes the filling in case the current weight – after stabilization – is less than the actual setpoint.

E.4.1.5 Lazy Filling

Activate
When checked, the board will identify too slow filling, thus reporting an error earlier than the 'Timeout' check would do.

E.4.2 Filling By = Time

E.4.2.1 Filling by Time Parameters

Setpoint #x
The required total filling time when the user selects setpoint #**x** (**x** = 1, 2 or 3).

Slow Amount
The required slow filling time, in % of the actual setpoint (a tip shows the value of **Slow Amount** in ms).
When only one speed is required, specify '0' and use the 'Fast' output.

E.4.2.2 Stabilization Criterion (Stop)

At the end of a filling cycle the board waits for the scale to stabilize before proceeding to the next cycle. The size of this waiting delay is specified by the "**Delta Time**" parameter.

Delta Time
See the description above.

E.4.3 Fast Speed Config

Option1
Fast Speed = Output #1
Slow Speed = Output #2

Option2
Fast Speed = Output #1 + Output #2
Slow Speed = Output #2

E.4.4 Advanced Parameters
(in LCIC-WIM-SETTINGS Version V2.17 and above)

Click the 'Adv.' button (near the low-right corner) to access the following **Advanced** parameters.

#1: Delay XXX ms before picking up the Final Result
After filling is over, wait XXX ms before picking up the Final Result.

For operator's convenience, Output4 supplies indications when these two events occur:
1. ***The filling ended*** *(Output4 is ON) –*
   the operator may remove the full bag.
2. ***The scale is empty and stable*** *(Output4 is flashing) –*
   the operator may put a new bag on the scale.

Both Output4 indications (ON or flashing) ***stop*** *when there is a request for a new filling – either by Input2 or by a 'g' command.*
The 1st parameter below lets you adjust the 1st indication.
The 2nd parameter lets you adjust the 2nd indication.

#2: After End of Filling Output#4 is ON for XXX ms
If XXX=0: Output4 is CONSTANTLY ON upon End of Filling
If XXX>0: Output4 is ON during XXX ms upon End of Filling

#3: Output#4 Flashes if for XXX ms the Scale is Empty
The flashing takes place only when Auto Zero / Activate is **not** checked.

E.4.5 The Filling Configurations Library

There is a 'Filling Configurations Library' in which you may save sets of filling configuration parameters. This is useful in case you have more than one type of filling, letting you switch easily and reliably from one configuration to another.
(Note: This library has nothing to do with the *calibration* library mentioned in section 3.2.2.)

Refer to the 'Library' box.

Save
If you like your configuration to be saved in the library, specify a Library Name and check the option box. (It is recommended to specify a meaningful Library Name so that later you'll recognize the various files you created in the library.) Your filling configuration will be saved in the library only upon clicking the 'Save to Board' button. In case a library file having the name you specified already exists, you'll have to select either to overwrite it, or to use another name.

Recall or Delete a File in the Library
Click the Select button in order to select a file to be recalled or deleted. You may watch the various files and their contents. In order to **delete** a file, click the Delete button. The procedure to **recall** a file has two or three steps:
1. Click the Confirm button. This will insert file's parameters to the 'Filling Definition' box.
2. Optionally, you may modify some of the parameters, thus using the library file as a draft to make your changes easier. In case you want the new values to be saved in a file (either the original one or another), make sure to check the option in the 'Library' box.
3. Click the 'Save to Board' button.

E.4.6 Addresses of the Filling Parameters

Below are the addresses of the various filling parameters so you may access them directly from your application by the 'R' and 'W' commands. Note that if you change a parameter, you should exit to the general mode (by 'x') and reactivate the Fill mode (by 'F') in order to validate the change.

Filling Parameters

Filling By: 1081 (Weight = 1, Time = 2)

Setpoint #1: 1101
Setpoint #2: 1102
Setpoint #3: 1103
Active Setpoint: 1096 (1, 2 or 3)

Slow Amount: 1091 (%)

Filling Timeout: 1087 (**seconds!**)

Auto Correction: 1083 (Checked = 1, Unchecked = 0)

Averaging x last fillings: 1088

Auto Tare

Activate: 1084 (Checked = 1, Unchecked = 0)

Hi Limit: 1086

Lo Limit: 1085

Valid Results Limits

Start from filling # ...: 1094

Valid Limits (±) ...: 1097 (%)

Stabilization Criterion

Delta Weight: 1098 (%)

Delta Time: 1099 (ms)

Stabilization Timeout: 1100 (ms)

Lazy Filling

Activate: 1095 (Checked = 1, Unchecked = 0)

Fast Speed Config: 1082 (Option1 = 1, Option2 = 2)

Advanced

Delay XXX ms before picking up the Final Result: 309 (ms)

After End of Filling Output#4 is ON for XXX ms: 1080 (ms)
(Specify 0 for constantly ON)

Output#4 Flashes if for XXX ms the Scale is Empty: 1116 (ms)

E.5 LED Display Notations

In **Fill-mode**, the LED display shows the current sub-mode:

m Fill Manual Fill-mode (input #1 is off)
AutoFill Auto Fill-mode (input #1 is on)

Likewise, the value shown on the LED display is prefixed by one or two letters:

C Current weight
F Weight during a filling cycle
A Actual weight (the resulting final filling weight)
SP SetPoint

The scale of the displayed weight

The displayed weight is sometimes **gross** and sometimes **net**:
* During a Filling Cycle, **and** the user selected Auto Tare:
  The displayed weight is **net**.
* In all other cases (that is, not during a Filling Cycle, or even during a Filling Cycle but the user did **NOT** select Auto Tare):
  The displayed weight is **gross**.

E.6 Commands

Enter & Exit Fill Mode

F Enter Fill Mode (from General Mode)
x Exit Fill Mode (to General Mode) (small 'x')

Inside the Fill Mode

g Start filling ('g' stands for 'go') (like input #2 does).

t Terminate an error status (like input #3 does).

e Emergency stop (like input #4 does).

r Get filling(s) report
Response Example:
 # 1 A= 40.99 Tr= 6.65 Cv= 0.00 Ft= 7527 ms St= 2554 ms Cc= 0
Legend: Filling #1, Actual (=final) weight=40.99,
Tare=6.65, Correction Value=0.00,
Fast time=7527 ms, Slow time=2554 ms,
Completion Code=0 (0 is normal, otherwise it's an error code).
Notes:
1. The actual (final) weight (40.99 in the example) is:
    * *net weight* if Auto Tare is selected
    * *gross weight* if Auto Tare is **not** selected.
2. The reports are accumulated in a FIFO whose size is 30 fillings. That is, only up to last 30 reports are available.
   Note: The FIFO size (30) is subject to change in the various versions.
3. Please refer also to section 3.3.2.1.4.

F Get fillings summary:
Number of normal fillings.
Number of each error type cases for each error type that occurred at least once.

s Get current status (small ‘s’)
Response Example:
Current_Status: W= 17.14 Tr= 6.65 Cv= 0.00 M=F A=I S= 0
Legend: Current (gross) weight=17.14, Last Tare=6.65,
Next Correction Value=0.00, Mode=Fill mode,
Activity=Idle (or: Tare, Fast, Slow)
System staus=0 (0 is normal, otherwise it’s an error code).

p Get parameters list (small ‘p’).

i Turn **off** the “Get results immediately” feature.

I Turn **on** the “Get results immediately” feature.

V Get current mode (upper case V).
Board’s response: ‘Fill-mode’.
Note: This command is useful in order to find out board’s mode. In the general mode the response to the ‘V’ command is ‘LCIC-WIM/V#.##’ (refer to section 4.1/i), as opposed to the ‘Fill-mode’ response in the fill mode.

E.7 Error Codes

101 Actual Filling Weight < Low Limit of 'Valid Result Limits.'
102 Actual Filling Weight > High Limit of 'Valid Result Limits.'
103 SetPoint < Low Limit of 'Valid Result Limits.'
104 SetPoint > High Limit of 'Valid Result Limits.'
105 High Tare Limit < Low Tare Limit
106 Actual Tare < Low Tare Limit
107 Actual Tare > High Tare Limit
108 Valid High Limit < Valid Low Limit
109 Lazy Filling
111 Filling Timeout
112 Low Tare Limit > High Tare Limit
113 SetPoint < Stabilization Delta Weight
115 Tare Stabilization Timeout
116 Initial weight is too large
117 Slow amount is too large
118 End Stabilization Timeout
119 Stabilization Delta Weight < Resolution
120 Memory Failure
121 User Emergency Stop
122 Stabilization Delta Time > Stabilization Timeout
123 Filling Timeout > Stabilization Timeout

Appendix F: Specifications

F.1 Load Cell Input

- 5 Volt excitation for upto 10 load cells (350 Ohm)
- Compatible with 1, 2 & 3 mV/V load cells
- Low noise wide bandwidth amplifier & 24 bit ADC

F.2 A/D

- Very high speed A/D: upto 52,000 samples per second
- 24 Bit A/D with ± 8 million counts for tension and compression applications

F.3 Digital Inputs

- 4 opto-isolated inputs with 10-30 VDC range, each with status LED
- Input #2 configurable as high speed counter

F.4 Digital & Analog Outputs

- 4 opto-isolated solid state relays rated at 50V, 300mA, each with status LED. Configurable as setpoints or manual outputs.
- Analog output of 0 to 2.5V with 16 bit resolution (option).

F.5 Standard Interfaces

- USB 2.0 Full Speed compatible
- Combined RS232/RS485
- Multiple boards may be connected via USB or RS232/RS485
- Ideal for PLC based applications

F.6 Software

- LCIC-WIM Calibration Wizard software is included.
- Included is the LCIC-WIM Monitor utility which is a vital tool for analyzing dynamic load/force systems. It takes full advantage of the board's speed.
- Also included the LCIC-WIM-SETTINGS utility which gives control to card's filters, analog output (option), filling parameters and other settings.
- An ActiveX interface is supplied for easier programming of user's application; however, direct conversation with the board is available either.

F.7 Dimensions (mm)

- Standard OEM model 160 x 100pcb (Eurocard)
- ABS cased option

F.8 Misc.

- Powerful 32 bit / 135 MIPS DSP for high speed onboard processing.
- 8 digit LED display
- On board temperature sensor
- Card includes an integral Fill Mode supplying an independent filling control.

Appendix G: Trouble-shooting

G.1 Card does not respond after PC power-on

Q. Everything was OK, but after PC restart the card suddenly stopped responding.
A. As specified in section D.4, after PC power on or off the serial communication (RS232/RS485) is likely to drop. There are two options to resume the connection:
1. Restart the card.
2. Send the '*' command through a USB port
   (refer to section 4.1, square 'i' in the table).

G.2 Loss of communication after calibration with RS485

To prevent loss of communication after calibration with RS485, act according to the following steps:

1. When the calibration process is finished, appears the message:

Select [No]

2. Wait some seconds until the current weight appears.

3. Now you can exit the Calibration Utility.

The RS485 communication will be restored.

Appendix H: Zero & Tare

There are two functions which are similar, yet actually different: Zero & Tare:

- The ***Zero*** function supplies both manual and automatic ways to clear the ***gross weight***.
- The ***Auto-Tare*** function supplies a way to define the meaning of the ***setpoint parameter***.

H.1 The *Zero* function

This function supplies both manual and automatic ways to clear the ***gross weight:***

Manual Zero

The manual zero is available only in the general mode, not in the inner mode (such as Fill-mode, CheckWeigher-mode, CatchWeigher-mode & WiM-mode). Its effect is temporary – it expires upon card reset. (Yet, some versions of WiM-mode board do include an inner manual zero function.)

The manual zero is accessible via two ways:

a. Hardware input: **Long activation** of digital input #3:

Long Activation

Until versions 1.13 and 3.102, that 'long activation' was fixed:
two seconds.

Starting from versions 1.14 & 3.104, the duration of the 'long activation' is user-defined by parameter #305 as follows:
Suppose that parameter #305 = x, then:
If x=0 (the default), the 'Manual Zero' option by input #3 is **disabled**.
Otherwise, the 'zero' operation will take place after a **continuous** activation of input #3 during x seconds.
The minimal valid x value is 0.
The maximal valid x value is 126.

Notes

1. Parameter #305 is accessible for the user by the 'R' and 'W' commands (refer to section 4.1, square 'a' in the table).
2. After changing parameter #305 the board **should be reset** in order that the new value will be in effect. You may either turn the power off & on, or send the 'S' command.
3. The default value (that is, the parameter is not defined or out of the valid range) is 0 (option disabled).
4. Board's LEDs will show ZEro On / ZEroOFF alternately. Releasing input #3 during "**Zero ON**" zeroes the scale. Releasing input #3 during "**ZEroOFF**" returns the calibration zero.

b. Communication:
   * When accessing the communication directly:
     Send the 'z' command (section 4.1/i).
   * When using the LCIC-WIM ActiveX:
     Call the Apply_Temporary_Zero function (section 4.3.5).

The manual zero – both by a hardware input or by communication – is honored <u>at any weight value</u> with no restriction, even if it is close to the Maximum Applied Load. That is, as this operation is manual, the operator is assumed to be responsible and has no restrictions.

<u>Auto Zero</u>

The auto zero is available only in the fill-mode, **not** in the general mode. Its effect is temporary – it expires upon card reset or upon exiting the fill-mode.
The auto zero operation occurs when some user's pre-defined condition is satisfied. For more details refer to section 3.3.2.2.

H.2 The *Auto-Tare* function

This function supplies a way to define the meaning of the ***setpoint parameter:***

* When the 'AutoTare' option is **not** activated, the setpoint defines the requested **final gross weight**.
  That is, if the setpoint is 100 kg and the starting gross weight is 90 kg, the filling amount will be 10 kg.
* When the 'AutoTare' option is activated, the setpoint defines the requested **filling amount**.
  That is, if the setpoint is 100 kg, the filling amount will also be 100 kg, regardless of the starting gross weight.
  However, the filling operation will take place only if the starting gross weight is within a user pre-defined range; otherwise, the filling operation will be rejected and an error will be indicated.

The activation of the 'Auto-Tare' option as well as its parameters are described in section E.4.1.2.

Appendix I: Board Reset by Input #4

There is an option to reset (restart) the board by hardware input #4.
This option **is** available in the –

- General Mode
- WiM Mode

The option is **not** available in the –

- CheckWeigher Mode
- Filling Mode.
- LiW Mode

The board reset is applied by **Long activation** of digital input #4.

Long Activation

The duration of the 'long activation' is user-defined by parameter #304 as follows:
Suppose that parameter #304 = x, then:
If x=0 (the default), the 'Board Reset' option by input #4 is **disabled**.
Otherwise, the 'Board Reset' operation will take place after a **continuous** activation of input #4 during x seconds.
The minimal valid x value is 0.
The maximal valid x value is 126.

Notes

1. Parameter #304 is accessible for the user by the 'R' and 'W' commands (refer to section 4.1, square 'a' in the table).
2. After changing parameter #304 the board **should be reset** in order that the new value will be in effect. You may either turn the power off & on, or send the 'S' command.
3. The default value (that is, the parameter is not defined or out of the valid range) is 0 (option disabled).

Note

A board reset is available also by a **software command**.
Refer to section 4.1, square 'i' in the table, command 'S'.

Appendix J: Overflow & Underflow

Availability:

Board version 3.115 and up: Both in general-mode and in WiM-mode.

If load cell is in **underflow** or **overflow**, then:

* The LED Display shows **LoAd.Err.O** (Overflow) or **LoAd.Err.U** (Underflow).
* A message is sent to the communication: '**Load Cell Err.Overflow**' or '**Load Cell Err.Underflow**' (optional).

The communication message is optional. It is controlled by an internal parameter in address 322:
170 = Don't send (default),
85 = Send
(The parameter takes effect only after a board reset).

Note

The original purpose of the Overflow/Underflow message is to supply an immediate indication on a **real** overflow or underflow condition.
As a by-product, it **may** supply an indication also on disconnected load cell.
However, such indication is neither for certain, nor immediate.
The reason is that in this case board's load cell input is **undefined**. **Usually**, within some seconds it will reach the Overflow or Underflow level, but sometimes it will **not** occur, so there will be no Overflow/Underflow indication.