

4	RUNNING A PROGRAM (USER'S OR A SAMPLE).....	16
5	INPUT/OUTPUT.....	17
6	TROUBLESHOOTING.....	17
6.1	BUS SPEED.....	17
6.2	ABORTED CALIBRATION.....	17

Appendixes

APPENDIX A	INSTALLATION.....	18
A.1	QUICK INSTALLATION.....	18
A.2	STEP BY STEP INSTALLATION (MEMCHK4.EXE).....	19
A.2.1	BEFORE THE INSTALLATION PROCEDURE.....	19
A.2.1.1	CONFIG.SYS UPDATE.....	19
A.2.1.2	COPYING THE SOFTWARE PACKAGE.....	19
A.2.2	THE INSTALLATION PROCEDURE.....	20
APPENDIX B	HOW TO ACCESS THE CARD IN WINDOWS.....	23
APPENDIX C	THE BATCH MODE.....	24
C.1	GENERAL DESCRIPTION.....	24
C.2	CARD'S OPERATION IN BATCH MODE.....	24
C.2.1	DEFINITIONS OF AUTO CHARACTERISTICS.....	24
C.2.1.1	FILLING AMOUNT DEFINITION.....	24
C.2.1.2	CHECKS DEFINITION.....	25
C.2.1.3	COMPLETION DELAY DEFINITION.....	25
C.2.2	RUNNING INSTRUCTIONS.....	26
C.2.2.1	CARD'S MODES.....	26
C.2.2.1.1	FILL MODE.....	26
C.2.2.1.2	ZERO-OFFSET MODE.....	26
C.2.2.2	START/STOP FILLING.....	27
C.2.3	CURRENT STATUS.....	27
C.2.3.1	ZERO A/D.....	27
C.2.3.2	CURRENT A/D.....	27
C.2.3.3	CURRENT ZERO-OFFSET/TARE.....	27
C.2.3.4	CURRENT WEIGHT.....	28
C.2.3.5	CURRENT VALVES STATUS.....	28
C.2.4	AUTO FILLING RESULTS.....	28
C.2.4.1	FILLING TIME.....	28
C.2.4.2	STABILITY TIME.....	28
C.2.4.3	COMPLETION CODE.....	28
C.2.4.4	DATA READY FLAG.....	29
C.3	CARD'S BUFFERS.....	29
C.3.1	CARD'S SOFTWARE INPUTS.....	29
C.3.1.1	SETUP SOFTWARE INPUTS.....	30
C.3.1.1.1	UPDATE WEIGHT IN AUTO? (ADVANCED).....	30
C.3.1.1.2	TARGET MODE.....	30
C.3.1.1.3	THE "TARE" OPTION (ADVANCED).....	31
C.3.1.1.4	FAST/SLOW combinations.....	32
C.3.1.1.5	FAST TARGET.....	34
C.3.1.1.6	SLOW TARGET.....	34
C.3.1.1.7	TARE MIN. (ADVANCED).....	35
C.3.1.1.8	TARE MAX. (ADVANCED).....	35
C.3.1.1.9	TARE STAB. AMPLITUDE (ADVANCED).....	36
C.3.1.1.10	TARE STAB. TIME (ms) (ADVANCED).....	36
C.3.1.1.11	STOP STAB. AMPLITUDE (ADVANCED).....	37
C.3.1.1.12	STOP STAB. TIME (ms) (ADVANCED).....	37
C.3.1.1.13	AFTER STOP DELAY (ms).....	37
C.3.1.1.14	IMPACT TIME (ms) (ADVANCED).....	38
C.3.1.1.15	UPDATE WEIGHT UNTIL (ADVANCED).....	38
C.3.1.2	OPERATION SOFTWARE INPUTS.....	38
C.3.1.2.1	FILL MODE.....	38
C.3.1.2.2	FILL START/STOP.....	39
C.3.1.2.3	VALVE SELECTION.....	40
C.3.1.2.4	ZERO-OFFSET MODE.....	40

C.3.2	CARD'S SOFTWARE OUTPUTS.....	41
C.3.2.1	DATA READY.....	41
C.3.2.2	COMPLETION CODE.....	42
C.3.2.3	CURRENT VALVES STATUS.....	43
C.3.2.4	ZERO A/D (ADVANCED).....	43
C.3.2.5	CURRENT A/D (ADVANCED).....	43
C.3.2.6	CURRENT ZERO-OFFSET / TARE.....	44
C.3.2.7	CURRENT WEIGHT.....	44
C.3.2.8	FAST FILL TIME (ms).....	45
C.3.2.9	SLOW FILL TIME (ms).....	45
C.3.2.10	REAL TARE STAB. TIME (ms) (ADVANCED).....	45
C.3.2.11	REAL STOP STAB. TIME (ms) (ADVANCED).....	45
C.3.3	AUTO FILLING SEQUENCE.....	46
C.4	ACTIVATION THROUGH THE LCIC4.EXE UTILITY.....	46
C.4.1	MODIFYING CARD'S SOFTWARE INPUTS.....	46
C.4.2	DATA READY TURNING OFF.....	47
C.5	ACTIVATION THROUGH USER'S APPLICATION.....	47
APPENDIX D	LCIC-4HS VS LCIC-1106a.....	48
D.1	FOUR CHANNELS.....	48
D.2	HARDWARE INPUT/OUTPUT.....	48
D.3	RS232.....	48
D.4	CALIBRATION.....	48
D.4.1	CALIBRATION LIBRARY.....	48
D.4.2	MULTI CALIBRATION FEATURE.....	48
D.4.3	CALIBRATION PARAMETERS.....	49
D.4.4	INTEGRATION.....	49
D.4.5	FILTER ON/OFF.....	49
D.4.6	SAMPLING RATE FACTOR.....	49
D.4.7	CALIBRATION LOCK.....	49
D.4.8	CALIBRATION COUNTER.....	49
D.5	THE "SMART" PACKAGE.....	50
D.6	PROGRAMMER'S GUIDE.....	50
D.7	AVAILABLE SETUP MEMORY LOCATIONS.....	50
D.8	BATCH MODE.....	50
APPENDIX E	AVAILABLE SETUP MEMORY LOCATIONS (ABSOLUTE ADDRESSING)...	51
APPENDIX F	LOAD CELL CONNECTIONS.....	52
F.1	DIRECT CONNECTION.....	52
F.2	CONNECTION VIA AN IMS ADAPTOR (PC1112).....	53
F.2.1	SIX WIRES CONNECTION (VIA ADAPTOR).....	53
F.2.2	FOUR WIRES CONNECTION (VIA ADAPTOR).....	53
APPENDIX G	I/O CONNECTIONS.....	54
APPENDIX H	DRAWINGS.....	55
H.1	LCIC4HS - I/O CONNECTIONS.....	55
H.2	LCIC4HS - LOAD CELL CONNECTIONS (DIRECTLY TO CARD)....	56
H.3	LCIC4HS - LOAD CELL CONNECTIONS (VIA ADAPTOR).....	57
H.4	LCIC4HS - CARD TO ADAPTOR CABLE.....	58
H.5	LCIC4HS - ASSEMBLY FOR THE ADAPTOR BOARD.....	59

1. INTRODUCTION

1.1 GENERAL DESCRIPTION

The LCIC-4HS (Load Cell Interface Card, four channels, high-speed) is used to sample the output of up to four load cells (350 ohm or higher each, one per a channel) and to convert these analog signals into a digital format via a precision Analog to Digital (A/D) Converter. The A/D data can be either displayed directly (using the supplied software) or further manipulated by a user's program. The LCIC-4HS is designed for PC/AT and compatible computers. The card is accessible by software (either the provided, or user developed) running on the PC.

Software is provided to aid in the installation of the card(s) and to perform calibrations.

The user is not limited to the programs supplied - he may satisfy his specific requirements by developing his own software, using almost any programming language. The manual guides how to do that, and the source code of a basic sample program in C is supplied.

1.2 CARD INSTALLATION

The card is a memory mapped hardware device, utilizing a dual port RAM, through which it supplies quickly and reliably updated readings, usable by the software in the PC. The dual port RAM occupies 1K of the upper PC memory area (between 640K and 1M). To eliminate conflict with other software/hardware devices, the address of the dual port RAM is programmable - there are 32 potential memory locations. Due to the various locations available for the dual port RAM, more than one card may be installed. Theoretically, up to 32 cards may be installed on one PC platform. The actual quantity depends on user's PC configuration.

The manual guides the user by "step by step" instructions how to install his card(s).

The software package includes a utility to aid in the installation, detecting a free location available for the dual port RAM. Once the user has activated that utility and a free location has been found, each card should be set accordingly by a DIP switch. If the card's preset address is vacant, a faster single card installation procedure is given in section A.1.

1.3 CARD VERSION

Since recent versions of the software in LCIC-4HS card's ROM include various improvements, you might have to know your card's ROM version. Find out card's ROM version by either of these two ways:

- Watch the title of LCIC4.EXE utility (see section [2.4](#)).
- Watch ROM label on the LCIC-4HS card:
 1. Locate the label attached to one of the components on the card.
 2. Consider label's format:

I.M.S. LTD. HSnnVmmm date

Where:

nn is internal clock speed.

V stands for "Version".

mmm is version no.

E.g., "HS24V004" means internal clock speed=**24**, version=**004**.

2. SOFTWARE UTILITIES

The software package supplied includes four utilities:

MEMCHK4.EXE	An installation utility that identifies free space in the PC available as common memory for the card's dual port RAM. The card can then be set to one of the available locations.
INIT4.EXE	Activates card(s). Performs hardware and calibration test.
FINISH4.EXE	Terminates activity of the card(s).
LCIC4.EXE	Calibration, setup and I/O test.

2.1 MEMCHK4.EXE

MEMCHK4.EXE utility is used only during the installation process. Refer to Appendix A.

2.2 INIT4.EXE

FUNCTION

The INIT4.EXE utility performs the following functions:

1. Complete availability check of the installed card(s) - hardware and calibration diagnosis.
2. Setting the card(s) into active mode.

OPERATION

1. Activate INIT4.EXE using either of the following two forms:
 - Type "INIT4" and <Enter>.
 - Type "INIT4 n" and <Enter>, where n = the number of cards. E.g. "INIT4 2" for two cards.

The first form of the INIT4 command will prompt the user for a response.

The second form does not prompt the user and therefore may be used for unattended operation or in a batch file.

2. INIT4.EXE determines the number of cards which are expected to be installed, according to the SEGx file(s) (created during the installation).
3. The first form of the INIT4 command will ask the user to confirm the number of cards. E.g., "Data about 2 cards detected. Sure? (y/n)".
4. If the user answers "no" to the first form of the INIT4 command, or there's an error in the second form:
 - The INIT4.EXE program will NOT communicate the card(s).
 - The DOS ERROR_LEVEL variable will be set to 4.
 - The INIT4.EXE program will terminate. Otherwise, INIT4.EXE flows to next step.
5. INIT4.EXE performs the availability check.
6. INIT4.EXE sets the card(s) into active mode.

RESPONSE

INIT4.EXE findings are reported by three means:

1. The display: The display reports the results of INIT4.EXE operation. In case of failure, a corresponding message is given.
2. Card's LEDs: Each card includes three LEDs marked LD1, LD2 & LD3. BEFORE the INIT4.EXE process, LD3 should flash at a VARIABLE RATE, and the other ones (LD1 & LD2) should light up, indicating proper power supply. AFTER a successful INIT4.EXE process, LD3 should flash at a UNIFORM RATE, and the other ones should still light.
3. ERROR_LEVEL: The result of the INIT4 command is reported in ERROR_LEVEL of DOS. The user may employ this information in a program or batch file. ERROR_LEVEL codes are as follows:

CODE	I N T E R P R E T A T I O N
0	Successful operation.
1	Calibration error or missing parameters.
2	Critical hardware error (the specific error is reported on the display).
3	Missing SEGx file - either MEMCHK4.EXE was not run, or the file(s) were erased.
4	Mismatch in number of cards.

NOTES

1. INIT4.EXE operation takes some seconds to complete its tasks.
2. INIT4.EXE should only be run once as long as the PC is not turned off, and FINISH4.EXE has not been run.

2.3 FINISH4.EXE

FUNCTION

The FINISH4.EXE utility sets each of the installed cards into an inactive mode, as it was before INIT4.EXE was run.

OPERATION

1. Activate FINISH4.EXE by either of the following two forms:
 - Type "FINISH4" and <Enter>.
 - Type "FINISH4 n" and <Enter>, where n = number of cards. E.g. "FINISH4 2" for two cards.
2. FINISH4.EXE operation is similar to that of INIT4.EXE. See INIT4.EXE operation for details.

2.4 LCIC4.EXE (THE CALIBRATION UTILITY)

This is the main utility, enabling card's calibration and I/O test. Activate LCIC4.EXE by either of the following two forms:

- Type "LCIC4" and <Enter>.
- Type "LCIC4 i" and <Enter>, where i = card's serial no. E.g. "LCIC4 2" for card #2.

The second form is required only in case of more than one LCIC-4HS card. However, if there is more than one card and the `f i r s t` form is used, the user is asked to select the required one. Cards' sequencing is same as with the SEGx files - refer to note 2 at the end of section A.2.2.

2.4.1 LCIC4.EXE DISPLAY

LCIC4.EXE displays the following values per each calibrated channel:

ZERO	Current zero in A/D points. For Zero change refer to section 2.4.2.6.
FILTER	Always ON. Specifies that the 'Digital Filter' is active. The Digital Filter activates a mechanism that disregards unacceptable readings caused by random disturbances, reducing in this way the variations generated due to noise.
INTEGRATION	Integration Factor. See section 2.4.2.7.
A/D	Raw A/D points of current measurement.
WEIGHT	Current weight in user selected units (kg. etc.).
MAXIMUM LOAD	Maximum planned load (in user selected units (kg. etc.)).
RESOLUTION	Required resolution of the displayed weight (in user selected units (kg. etc.)).

2.4.2 LCIC4.EXE FUNCTIONS

The LCIC4.EXE uses function keys to select operating parameters or functions as follows:

Within the initial display, the following operations are available:

KEY	O P E R A T I O N	SECTION
F5	Select channel	2.4.2.1
F6	See constants	2.4.2.2
F8	I/O check	2.4.2.3
F10	Batch mode	2.4.2.4

Upon selecting some specific channel, the following operations are also available. Except the Batch mode (F10), the previous operations are still available also when some specific channel is selected.

KEY	O P E R A T I O N	SECTION
F2	Calibration	2.4.2.5
F3	Zero	2.4.2.6
F4	Integration	2.4.2.7

To quit the LCIC4.EXE utility, press Alt-X or F9.

2.4.2.1 F5 - SELECT CHANNEL

Use this function to specify the object of the following operations:

- * Calibration (F2, see section [2.4.2.5](#)).
- * Zero (F3, see section [2.4.2.6](#)).
- * Integration (F4, see section [2.4.2.7](#)).

Use this function also in order to "deactivate", i.e. cancel the selection of a channel (type '0' in this case). Note that the above three operations are available only after selecting the object channel. On the other hand, the Batch mode (F10) is available only when no channel is selected.

2.4.2.2 F6 - SEE CONSTANTS

This function displays some internal calibration constants in case some analysis is required. Use only in case of problems in calibration and pass the data to IMS.

2.4.2.3 F8 - I/O CHECK

The I/O Check permits the user to inspect the I/O states and to change the outputs.

Changes to the outputs can be accomplished in two ways:

1. SINGLE OUTPUT CHANGE

- Select the output to be changed by moving the displayed frame using the right and left arrow keys.
- Press the space bar to toggle the output state.

2. MULTIPLE OUTPUT CHANGE

- Press the F2 key to toggle **ALL** output states.

2.4.2.4 F10 - BATCH MODE

NOTE

This operation is NOT available if a channel has been selected by F5. To cancel channel's selection, use the F5 again - refer to section 2.4.2.1.

This function activates the Batch mode. Refer to appendix C.

2.4.2.5 F2 - CALIBRATION

NOTE

This operation applies to a specific channel hence is available only after selecting a channel (by F5, see section 2.4.2.1).

The calibration is a DESTRUCTIVE operation, since it erases the old information. In other words, the card loses the old calibration. If the channel is already calibrated, the program will verify that the user wants to proceed.

The following message appears...

WARNING - Actual calibration parameters will be lost.
Continue?

...preventing the user from accidentally starting the calibration process.

Answering "Y" (Yes) begins the calibration process.

2.4.2.5.1 STEP 1 - PARAMETER ENTRY

The user should insert the following parameters:

Parameter #1: Calibration points

There are two options:

- 'Normal' calibration.
- 'Quick' calibration.

The conventional procedure is 'Normal' calibration. It includes three weights that should be added or removed according to the instructions. This option assures the most accurate calibration results.

Sometimes such a calibration is impractical or unnecessary. For example, masses are very large, or an easier calibration method (using the other option) supplies satisfactory results.

The 'Quick' option supplies an easier calibration method. It includes two steps:

- First step: Start with no load.
- Second step: Add a known weight.

Calibration is complete - no need to remove the load!

NOTE - EXTREME CONDITIONS

The accuracy of the 'Quick' calibration may be reduced under extremely exceptional conditions. Such a situation may happen if the maximum planned load is 10% or less of load cell capacity, *and* the weight used during the calibration process is 10% or less of the maximum planned load. Note that in such a case calibration's weight is only 1% (or less) of the load cell capacity! In spite of the exceptional conditions, the calibration will still work, but possibly with slightly less accuracy.

The user should respond with "N" ('Normal') or "Q" ('Quick').

"Normal" Calibration vs. "Quick" calibration

The "Normal" calibration procedure includes the following steps:

1. Empty scale
2. Some weight
3. Empty scale again
4. 1st weight (not necessarily the same one as in step #2)
5. 2nd weight
6. 3rd weight

(This is the minimum sequence, but some steps may repeat themselves.)

The "Quick" calibration procedure involves only one weight:

1. Empty scale
2. Weight

Now, when comparing the two modes:

Obviously, the "Quick" mode is much easier.

However, the "Normal" mode has the following advantages:

- The results of the calibration are more accurate as the card is used more efficiently.
- The linearity of the weights is checked and in case of deviation larger than the "Accuracy" parameter the calibration will be rejected. This eliminates human mistakes while calibrating (for example, specifying a wrong value of the weight).
- Due to the averaging, the effect of some inaccuracy in a weight is reduced.

Finally, the user has to decide whether he prefers to have a more complicated calibration procedure but yielding better results, or an easier one yielding less accurate results.

Parameter #2: Units (Kg., Lbs. or Gr.)

The selected unit will be used in all weight displays.

Parameter #3: Load Cell Division

Load cell division according to manufacturer's specification. This is an "administrative" parameter serving for check whether the resolution that the user requires is beyond the stated specification of load cell's manufacturer. The user's required resolution is the quotient of the "Maximum Planned Load" and the "Display Resolution" parameters. For example, if "Maximum Planned Load" = 300 kg and "Display Resolution" = 0.1 kg, the implied resolution is $300/0.1 = 3000$. If load cell's resolution, as inserted by the user in the "Load Cell Division" parameter, is less than 3000 (say 2000), then the calibration program warns the user: "Display resolution too high". This warning is only a tip - the user may ignore it and confirm the parameters in spite of the conflict, being aware that his requirement is actually higher than the load cell - theoretically - supplies.

Parameter #4: Maximum Planned Load

The maximum net weight to be measured by the load cell.

Parameter #5: Accuracy

The maximum permitted deviation from linearity, which will still be acceptable. This parameter is relevant only in the case of 'Normal' calibration (parameter no. 1), where the various weights usually have some deviation. Should a larger deviation result during the calibration process, an error will be shown (see section 2.4.2.5.5.4).

Parameter #6: Display Resolution

Resolution of the displayed weight.

After all calibration values have been inserted, the following warning may be displayed:

Display resolution too high - accept?

This message indicates that there is a conflict within the specified Load Cell Division, the Maximum Planned Load and the Display Resolution (parameters no. 3, 4 and 6, respectively). However, as described in paragraph 3 above, the user is permitted to override the warning and proceed. A "Y" (Yes) answer overrides the warning, and an "N" (No) answer enables the user to specify new parameters.

Following a successful entry of parameters, the user is presented with the message:

Confirm calibration parameters?

Answering "Y" (Yes) confirms the parameters. An "N" (No) answer enables the user to specify new parameters.

2.4.2.5.2 STEP 2 - THE ACTUAL CALIBRATION (FOR 'NORMAL' OPTION)

1. Press the space bar when you are ready to sample tare (dead load).
BEWARE OF ACCIDENTALLY PRESSING THE SPACE BAR IF THE SCALE IS NOT EMPTY!
2. Place some weight on the load cell. The recommended value of the weight is around 70% of the maximum planned load. When ready, type in its value and press <Enter>.
3. Remove the weight and press the space bar when you are ready to sample again the tare (dead load) setting the zero point.
4. Place weight #1 on the load cell. Weight #1 is not necessarily the same one as used in step #2. When ready, type in its value and press <Enter>.
5. Place weight #2 on the load cell. When ready, type in its value and press <Enter>.
6. Place weight #3 on the load cell. When ready, type in its value and press <Enter>.

NOTES

1. This is the minimum sequence, but it may occur that more repetitions will be required.
2. The recommended value of the total weight is around 70% of the maximum planned load, in any case not more than 95%.

2.4.2.5.3 STEP 2 - THE ACTUAL CALIBRATION (FOR 'QUICK' OPTION)

1. Press the space bar when you are ready to sample tare (dead load).
BEWARE OF ACCIDENTALLY PRESSING THE SPACE BAR IF THE SCALE IS NOT EMPTY!
2. Place a weight on the load cell. The recommended value of the weight is around 70% of the maximum planned load. When ready, type in its value and press <Enter>.

2.4.2.5.4 CALIBRATION COMPLETION

After the calibration process has been completed, the user is asked:

Press any key to exit calibration check.

Before the user exits by pressing a key, the calibration may be checked by changing the weight applied. The response time in this mode is large, so that fast fluctuations are NOT displayed. Once a key has been pressed, the following message is displayed:

Please wait...

after which the main display appears again.

2.4.2.5.5 POSSIBLE CALIBRATION ERRORS

2.4.2.5.5.1 CHECK LOAD CELL WIRING

This error may occur during the calibration in two cases:

- Incorrect load cell wiring.
- Calibration procedure has not been followed correctly. For example, after tare (empty scale) sampling, a weight has been removed instead of being added.

2.4.2.5.5.2 ZERO FAILURE

Should this error occur during the tare operation, the A/D reading is too high or too low. This may occur if the load cell is not working properly. Alternatively, it may occur when using a very HIGH tare (around full load cell capacity), and a very LOW Maximum Planned Load (about 1-2% of load cell capacity). To resolve this problem, re-enter the Maximum Planned Load slightly higher or lower (refer to section 2.4.2.5.1, parameter #4).

For example:

Assumed full scale load cell capacity	10kg
Tare (dead load)	8kg
Maximum Planned Load	2kg
Resolution	0.001 kg

Repeat the calibration procedure with slightly different Maximum Planned Load, such as 1.8 or 2.2 kg.

2.4.2.5.5.3 WEIGHT IS TOO SMALL

Occurs during the calibration when the weight used is too small to assure accuracy. Select a heavier weight to resolve the problem.

2.4.2.5.5.4 NONLINEAR WEIGHT

Occurs during the calibration when the ratio between the two gains has exceeded the specified Accuracy (parameter no. 5 in section 2.4.2.5.1).

- Restart the process from the point of dead load sampling,
or:
- Re-enter the last weight.

The second option is useful when the value of the last (current) weight was misspelled.

2.4.2.5.5.5 WEIGHT NOT WELL ACCEPTED

Occurs in case of unstable load cell. The system returns to the 'Zero' stage prompting the user to press 'Z' so he'll surely notice the case.

2.4.2.6 F3 - ZERO

NOTE

This operation applies to a specific channel hence is available only after selecting a channel (by F5, see section 2.4.2.1).

The ZERO function is used to define a new zero weight. First, the following message appears:

WARNING - Actual calibration parameters will be lost.
Continue?

...preventing the user from accidentally starting the ZERO process.

The loss of the actual calibration parameters is not "certain". It will occur only if the new zero differs significantly from the original one. If the loss does occur, then a new full calibration is required.

When ready, the user should press the space bar to sample the new zero.

2.4.2.7 F4 - INTEGRATION

NOTE

This operation applies to a specific channel hence is available only after selecting a channel (by F5, see section 2.4.2.1).

The card includes an integrator that averages the raw readings of the A/D, thus supplying a much more stable value. The level of the integration is controllable by the Integration Factor (IF) so that the user may fit it to the application.

The IF may be set to one of eleven levels - from 0 to 10.

While a higher IF results in more stable readings, it increases exponentially card response time.

IF = 0 specifies the **lowest** level of integration.

Useful when the user prefers to process the raw A/D readings by his own program.

IF = 10 specifies the **highest** level of integration.

Useful when the user does not average the readings by his own program.

Obviously, the values of 1 through 9 are mean levels.

Answer in the INSTRUCTION window by the required IF.

3. PROGRAMMER'S GUIDE

The LCIC-4HS is memory-mapped in PC memory in a region which is normally not in use.

The user may communicate with the LCIC-4HS card by almost any programming language according to the following description.

MEMORY BLOCK ADDRESSING

The address of each card buffer is composed of two parts:

- SEGMENT - the address of the specific CARD.
- OFFSET - the address of the specific BUFFER.

The SEGMENT part is actually a Base Address assigned to each card by the MEMCHK4.EXE installation utility, allocating the card a unique address range within the upper memory area of the

PC (refer to appendixes A & E). The SEGMENT values (in hex.) reside in a series of "SEGx" files (SEG1., SEG2., etc.) created by the MEMCHK4.EXE utility during the installation. For example, if one card has been installed using the factory preset adjustment, the file "SEG1." contains the text "CC80" which represents the card's SEGMENT. The OFFSET part of the address assigns the location of each buffer, as described in the next section.

3.1 CARD'S BUFFERS

Card's buffers are described below. Each buffer description is followed by "programming information" required for user's application program coding. Section [3.2](#) explains how to use this "programming information".

Note Some of the buffers are only for an advanced usage. To start with, their description (marked "ADVANCED") may be skipped.

INPUTS	* Describes the status of the inputs.		
	PROGRAMMING INFORMATION		
	DATA TYPE	byte	
	OFFSET	0x00A	
	CODES	ON	0
OFF		1	
The lowest (rightmost) bit = input #1; The highest (leftmost) bit = input #8.			

OUTPUTS	* Describes the status of the outputs.		
	PROGRAMMING INFORMATION		
	DATA TYPE	byte	
	OFFSET	0x009	
	CODES	ON	0
OFF		1	
The lowest (rightmost) bit = output #1; The highest (leftmost) bit = output #8.			

Channel calibrated?	<ul style="list-style-type: none">* Indicates whether the channel is calibrated or not.* The LCIC4.EXE utility presents this information by the background colour of channel's data:<ul style="list-style-type: none">* Normal colour (green) indicates "calibrated".* Violet indicates "not calibrated".												
	<div>PROGRAMMING INFORMATION</div> <table><tr><td>DATA TYPE</td><td colspan="2">4 bits</td></tr><tr><td>OFFSET</td><td colspan="2">0x0B8-L</td></tr><tr><td rowspan="2">CODES</td><td>not calibrated</td><td>0</td></tr><tr><td>calibrated</td><td>1</td></tr></table>			DATA TYPE	4 bits		OFFSET	0x0B8-L		CODES	not calibrated	0	calibrated
DATA TYPE	4 bits												
OFFSET	0x0B8-L												
CODES	not calibrated	0											
	calibrated	1											

CURRENT WEIGHT	* Current weight in kg., lb., etc. (as was selected in calibration).	
	* CURRENT WEIGHT is irrelevant in case channel is not calibrated. (In this case it's not displayed by the LCIC4.EXE utility.)	
	PROGRAMMING INFORMATION	
	DATA TYPE	float: IEEE Standard, 32 bit; supports 24 bit of precision. The upper bit (MSB) of the mantissa is always a '1' and is therefore not stored.
	OFFSET	0x060, 0x064, 0x068, 0x06C
	DATA TYPE	ASCII: Null terminated string, 16 characters max.
	OFFSET	0x150, 0x160, 0x170, 0x180
	NOTES	
	1. FLOAT and ASCII	
	The float representation option is the faster and is therefore usually preferable. However, the advantage of the ASCII (string) representation is that the values it supplies use the same resolution with which the channel was calibrated.	
	2. READING PROCEDURE	
	Although current weights are calculated continuously inside the card, they are not available automatically on the PC buffers, unless explicitly requested by user's program. To get an up-to-date Current Weight, follow the next reading procedure: * Write (float or ASCII) INFO REQUEST code to the INFO FLAG buffer. * Delay 2 ms without accessing the card. * Wait until the INFO FLAG buffer equals INFO READY code. * Read the proper (float or ASCII) CURRENT WEIGHT buffer(s).	
	3. ALL CHANNELS ARE READ	
Writing an INFO REQUEST code to the INFO FLAG buffer requests CURRENT WEIGHTs for ALL channels. When the INFO FLAG buffer signals INFO READY, ALL the four CURRENT WEIGHTs are updated.		

INFO FLAG	* Used while reading CURRENT WEIGHT. * Refer to CURRENT WEIGHT square, note #2.			
	PROGRAMMING INFORMATION			
	DATA TYPE	byte		
	OFFSET	0x004		
	CODES	INFO REQUEST (Write to buffer)	ASCII float	0 5
CURRENT A/D *** ADVANCED ***		INFO READY	ASCII & float	1
		(Read from buffer)		
	* Current raw A/D value, before its conversion to weight units. * The current A/D is usually not useful for the user. * The LCIC4.EXE utility displays the current A/D only in the general mode, NOT in the batch mode. * The notes specified in the CURRENT WEIGHT square are relevant also for the CURRENT A/D: Upon completing the first three steps of the reading procedure, the CURRENT A/D buffers are updated as well (in addition to the (float or ASCII) CURRENT WEIGHT buffers).			
	PROGRAMMING INFORMATION			
	DATA TYPE	int		
ZERO A/D *** ADVANCED ***	OFFSET	0x0A0, 0x0A2, 0x0A4, 0x0A6		
	* Zero raw A/D value, before its conversion to weight units, as accepted during the calibration. * The zero A/D is usually not useful for the user. * The LCIC4.EXE utility displays the zero A/D only in the general mode, NOT in the batch mode.			
	PROGRAMMING INFORMATION			
	DATA TYPE	int		
	OFFSET	0x1C0, 0x1C4, 0x1C8, 0x1CC		
VERSION	* Describes card's software version.			
	PROGRAMMING INFORMATION			
	DATA TYPE	byte		
	OFFSET	0x00F		

3.2 PROGRAMMING INFORMATION

Each buffer's description terminates with "programming information" supplying the user with the required information how to access the buffer.

The "programming information" has three main items:

- Buffer's data type
- Buffer's offset
- Buffer's codes interpretation when the buffer contains a code (and not a quantitative value)

The interpretation of the data types and the offsets, both abbreviated in the "programming information", follows in the tables below. The buffer's codes, when applicable, are included in the individual description of the specific buffer.

DATA TYPE

ABBREVIATION	DESCRIPTION
4 bits	Four bits, each describing one channel's code (0 or 1)
byte	One byte
int	Two bytes unsigned integer numerical value
float	Four bytes float numerical value <div style="border: 1px solid black; padding: 5px; margin-top: 10px;"> The float format used conforms to the IEEE 32 bit standard and supports 24 bit of precision. The upper bit (MSB) of the mantissa is always a '1' and is therefore not stored. </div>
ASCII	Null terminated ASCII string

OFFSET

To make the description more clear, the table uses sample offsets.

DATA TYPE	ABBREVIATION	DESCRIPTION											
4 bits	0x009-L	LOW half (bits 0-3) of offset 0x009: <table><tr><th>Channel</th><th>Bit</th></tr><tr><td>1</td><td>0</td></tr><tr><td>2</td><td>1</td></tr><tr><td>3</td><td>2</td></tr><tr><td>4</td><td>3</td></tr></table> (Bit 0 is the lower)		Channel	Bit	1	0	2	1	3	2	4	3
	Channel	Bit											
1	0												
2	1												
3	2												
4	3												
	0x009-H	HIGH half (bits 4-7) of offset 0x009: <table><tr><th>Channel</th><th>Bit</th></tr><tr><td>1</td><td>4</td></tr><tr><td>2</td><td>5</td></tr><tr><td>3</td><td>6</td></tr><tr><td>4</td><td>7</td></tr></table> (Bit 7 is the upper)		Channel	Bit	1	4	2	5	3	6	4	7
Channel	Bit												
1	4												
2	5												
3	6												
4	7												
other	0x020, 0x024, 0x028, 0x02C	Channel	Offset										
		1	0x020										
		2	0x024										
		3	0x028										
		4	0x02C										

3.3 MORE PROGRAMMING TOPICS

READING MULTIPLE CARDS

The procedure of reading multiple cards is identical, except that the SEGMENT portion of the address changes for each card.

For example:

Suppose that...	...then to read FLOAT data of channel #1
SEG1. contains CC80	address of card #1 is CC80:0060
and...	and...
SEG2. contains CE80	address of card #2 is CE80:0060

PROGRAMMING EXAMPLES

(Suppose that SEG1. and SEG2. are still as above)

1. C Language, Load cell reading of card #1

```
float far* FLOATPTR ;
float RESULT ;
FLOATPTR = (float *far) MK_FP (0xCC80, 0x0060) ;
RESULT = *FLOATPTR ;
```

2. PASCAL Language, Load cell reading of card #2

```
var FLOATPTR : ^single ;
RESULT      : single ;
FLOATPTR := ptr ($CE80, $0060) ;
RESULT := FLOATPTR^ ;
```

3. BASIC Language, Load cell reading of card #2

```
DEF SEG = VAL("&H" + CE80)           ' Set segment
F$ = CHR$(PEEK(&H60)) + CHR$(PEEK(&H61))
    + CHR$(PEEK(&H62)) + CHR$(PEEK(&H63)) ' Read 4 bytes
RESULT$ = CVS(F$)                   ' Convert 4 bytes to single precision
```

PROGRAMMING IN WINDOWS

Refer to Appendix B for information how to program in Windows.

4. RUNNING A PROGRAM (USER'S OR A SAMPLE)

Using the Programmer's Guide in the previous section, the user may create his own application program. A sample C program may be run as well - refer to files test.c, test.prj & test.exe. In both cases, INIT4.EXE should be activated before running the program, and FINISH4.EXE afterwards (refer to sections [2.2](#) & [2.3](#)). However, to facilitate software development, only *o n e* INIT4.EXE is required as long as the PC is not turned off, and FINISH4.EXE has not been run. For ease and reliability, a batch file like the following may be used (two LCIC-4HS cards are assumed):

```
@ECHO OFF
INIT4 2
IF ERRORLEVEL 4 GOTO ERR4
IF ERRORLEVEL 3 GOTO ERR3
IF ERRORLEVEL 2 GOTO ERR2
IF ERRORLEVEL 1 GOTO ERR1
user's application program / sample program
FINISH4
GOTO EXIT
:ERR1
ECHO No calibration, or missing parameters.
GOTO EXIT
:ERR2
ECHO Critical hardware error.
GOTO EXIT
:ERR3
ECHO Missing SEGx file.
GOTO EXIT
:ERR4
ECHO Number of cards detected mismatch.
:EXIT
@ECHO ON
```

The LCIC4.EXE utility, however, is an exception: The INIT4.EXE & FINISH4.EXE are included integrally, so there is no need to run them separately.

5. INPUT/OUTPUT

DESCRIPTION

The I/O feature turns the LCIC-4HS into a control system. Due to its eight inputs and eight outputs (all opto-isolated), the option permits control of external or program events. The I/O's are accessible both by the software supplied and by user's application program.

- During the batch mode, two outputs are allocated per each channel (for the fast and the slow valves).
- When not in the batch mode, the outputs are general-purpose.
- An external 5 to 30 VDC supply is required.
- All the outputs are transistors in current sink mode.
- Maximum current per output: 500 mA.
- Maximum outputs power: 8 watt.
- For I/O check by the LCIC4.EXE utility refer to section [2.4.2.3](#).
- For input/output connections refer to drawing #1 (section H.1) and to appendix G.

PROGRAMMING - I/O ADDRESSING

- The eight inputs are accessible via the eight bits of the LCIC-4HS INPUTS buffer - bits 0 through 7 correspond to Inputs 1 through 8, respectively.
- The eight outputs are accessible via the eight bits of the LCIC-4HS OUTPUTS buffer - bits 0 through 7 correspond to Outputs 1 through 8, respectively.
- For details on the INPUTS and OUTPUTS buffers, see section [3.1](#).

6. TROUBLESHOOTING

6.1 BUS SPEED

If the PC bus speed is much higher than the ISA standards, it may occur that "BAD RAM" error message appears, or the screen is not updated during the calibration process. In such a case, the clock speed of the PC should be reduced using the BIOS SETUP PROGRAM, modifying a parameter such as "clock speed", "divide clock by", "ISA CLK Speed" etc. E.g., if "divide clock by" is 2, change it to 4.

6.2 ABORTED CALIBRATION

In case of POWER FAILURE or RESET while running the calibration utility (LCIC4.EXE), the card is liable to stay in a transitional state, being UNCALIBRATED: The old calibration was already wiped out, while the new one has been aborted. If there isn't even one channel calibrated, running the INIT4.EXE utility will lead a "BAD RAM" error message.

In order to avoid it, take the following steps:

- Turn the PC off.
- Turn the PC on.
- Re-calibrate the card using the LCIC4.EXE utility. NOTE: Do NOT run your application program, nor the INIT4.EXE utility, prior to the re-calibration! Otherwise, the "BAD RAM" message will re-appear.

APPENDIX A: INSTALLATION

The software package includes a utility named MEMCHK4.EXE which can detect a free location available for the LCIC-4HS dual port RAM (described in section [1.2](#)). Once such location has been found, the LCIC-4HS card should be adjusted accordingly by a DIP switch. The MEMCHK4.EXE utility automatically creates a file named "SEGx.", containing the segment address of the location detected (refer to Appendix E). If there are several LCIC-4HS cards, *some* "SEGx." files are created - SEG1., SEG2., SEG3., etc.

However, refer first to "Quick Installation", that might simplify the procedure for the first LCIC-4HS card.

A.1 QUICK INSTALLATION

Trying to save your time, the card has already been factory preset, and a corresponding SEG1 file is already supplied on LCIC-4HS software package. Both have been set to address CC80 (2nd selection in Appendix E). The DIP switch setting on the LCIC-4HS is as follows:

S1	S2	S3	S4	S5
OFF	ON	ON	ON	ON

If the default address is not occupied, you may skip the full "STEP BY STEP INSTALLATION" (section A.2) and carry out the following sequence:

1. Review the CONFIG.SYS file on your boot diskette/hard disk with any text editor. If there's a DEVICE=C:\DOS\EMM386.EXE statement, append "AUTO X=C000-CFFF" to that statement. (It will look like: DEVICE=C:\DOS\EMM386.EXE AUTO X=C000-CFFF)
2. There are the following installation folders:
Doc
DOS
Win98
WinNT-2000-XP
In order to operate the LCIC-4HS card under **DOS**, just copy the files in the DOS folder to your hard disk. For **Windows**, please refer to Appendix B.
Preferably, first backup the files.
3. Turn the PC OFF, remove its cover and locate a free ISA slot.
4. Install the LCIC-4HS card inside the free ISA slot. **MAKE SURE THAT THE BOARD CONNECTOR IS WELL INSERTED INTO THE SLOT.** Do NOT close the cover yet.
5. Turn the PC ON. Watch LCIC-4HS card's LEDs. The LED marked as LD3 should flash at a variable rate. The other LEDs (LD1 & LD2) should light up. If this is NOT the case, then:
 - Turn the PC OFF.
 - Try to re-install the LCIC-4HS card in the slot.
 - Turn the PC ON.
 - If LEDs' status is correct, proceed to step 6.
 - Otherwise (LEDs' status is still wrong), there might be a problem with the connector of the selected ISA slot:
 - Turn the PC OFF.
 - Locate another free ISA slot.
 - Return to step 4.
6. Use the subdirectory where you installed the LCIC-4HS software package in step 2.
7. Run the LCIC4.EXE program (the calibration utility).
8. If the main screen of the calibration utility appears, your card is properly installed and ready for use - close the PC's cover and proceed with the calibration.

9. If your computer "hangs up", or displays any message such as "Please wait" or "Please shut off your computer" - the preset address is conflicting with some other device in your system. Please refer to the "STEP BY STEP INSTALLATION" procedure.

A.2 STEP BY STEP INSTALLATION (MEMCHK4.EXE)

A.2.1 BEFORE THE INSTALLATION PROCEDURE...

A.2.1.1 CONFIG.SYS UPDATE

Review the CONFIG.SYS file on your boot diskette/hard disk with any text editor. If there's a DEVICE=C:\DOS\EMM386.EXE statement, some steps should be taken before carrying out the installation procedure to eliminate memory conflicts with the card(s). Either of the following two procedures will do the job. However, in case you already carried out the "quick installation", sequence "a" is ALMOST done - except the range of the "X" field that should be as specified in para. a.1.

a. Manual modification of the DEVICE=C:\DOS\EMM386.EXE statement

- a.1 Include in the beginning of your CONFIG.SYS file the statement:

```
DEVICE=C:\DOS\EMM386.EXE AUTO X=C000-EFFF
```

- a.2 Re-boot your PC.

- a.3 Apply the LCIC-4HS installation procedure and make sure that the INIT4.EXE operation completes successfully.

b. Automatic modification of the DEVICE=C:\DOS\EMM386.EXE statement

- b.1 Remove the EMM386.EXE statement from CONFIG.SYS (alternately, you may put REM in the beginning of this statement).

- b.2 Re-boot your PC.

- b.3 Apply the LCIC-4HS installation procedure and make sure that the INIT4.EXE operation completes successfully.

- b.4 Run MEMMAKER.EXE (available in DOS version 6 and up). This utility will recognize your installed card(s) and return to CONFIG.SYS a revised DEVICE statement with such parameters that consider the new card(s), so that no memory conflicts will occur.

A.2.1.2 COPYING THE SOFTWARE PACKAGE

There are the following installation folders:

Doc

DOS

Win98

WinNT-2000-XP

In order to operate the LCIC-4HS card under **DOS**, just copy the files in the DOS folder to your hard disk. For **Windows**, please refer to Appendix B.

Preferably, first backup the files.

A.2.2 THE INSTALLATION PROCEDURE

The step by step installation process includes three passes, each requiring execution of the MEMCHK4.EXE utility. The task of this utility is to locate free space in the PC memory available for the dual port RAM of the card(s).

- In pass one the utility starts its examination of the PC memory configuration.
- In pass two the utility completes the examination of the PC memory configuration and displays the available locations. Once this pass has been completed, the user selects an available address location by setting a DIP switch in the card.
- Pass three reports how many cards have been recognized.

Note

In passes one and two the card(s) are **out** of the PC. The card(s) are inserted during the third pass.

Pass one

(Card(s) NOT inserted in the PC.)

1. Use the subdirectory where you installed the software package in step A.2.1.2.
2. Run the MEMCHK4.EXE program. Utility responds with:

MEMCHK4 - Memory Check (4 channels) utility Ver 1.00 (Dec. '96)
3. Turn the PC OFF.

Pass two

(Card(s) still NOT inserted in the PC.)

1. Turn the PC ON.
2. Use the subdirectory where you installed the software package in step A.2.1.2.
3. Run the MEMCHK4.EXE program.

MEMCHK4.EXE responds with a report dividing the PC memory locations into three categories:

 1. Locations definitely available.
 2. Locations definitely not available.
 3. Locations that may be available.
4. Set card's DIP switch accordingly. To determine the actual memory addresses corresponding to DIP switch setting, consult the table of Appendix E.
5. Turn the PC OFF.

Pass three

(The PC is still turned OFF!)

1. Remove PC cover and locate a free ISA slot.
2. Install the LCIC-4HS card into the free ISA slot. **MAKE SURE THAT THE BOARD CONNECTOR IS INSERTED WELL INTO THE SLOT.** Do NOT close the cover yet.
3. Turn the PC ON. Watch LCIC-4HS card's LEDs. The LED marked as LD3 should flash at a variable rate. The other LEDs (LD1 & LD2) should light up. If this is NOT the case, then:
 - Turn the PC OFF.
 - Try to re-install the LCIC-4HS card in the slot.
 - Turn the PC ON.
 - If LEDs' status is correct, proceed to step 4.
 - Otherwise (LEDs' status is still wrong), there might be a problem with the connector of the selected ISA slot:
 - Turn the PC OFF.
 - Locate another free ISA slot.
 - Return to step 2.
4. Use the subdirectory where you installed the LCIC-4HS software package in step A.2.1.2.
5. Run the MEMCHK4.EXE program.
MEMCHK4.EXE responds with the number of cards that have been recognized.
6. Turn the PC OFF.
7. Turn the PC ON.
8. Use the subdirectory where you installed the software package in step A.2.1.2.
9. Run the LCIC4.EXE program (the calibration utility) - refer to section [2.4](#). The main screen of the calibration utility should appear, indicating that your card is properly installed, ready for use - proceed with the calibration.
10. Close PC's cover.

Notes

1. NO. OF CARDS DETECTED

The number of cards detected may be less than the actual number of cards inserted. If this occurs, check whether two cards have the same DIP switch settings. Alternatively, MEMCHK4.EXE may fail to recognize a card because of some other memory conflict. In this case, please consult note 3.

2. SEGx FILES

MEMCHK4.EXE creates one or more files called "SEGx", where x = 1, 2, 3, ... corresponding to the number of cards recognized. Each SEGx file contains a four-character ASCII text identifying the segment address allocated for the corresponding card. The sequencing of the files relates to the order in the list displayed in pass two.

Example

If three cards are installed and respectively set to the 3rd, 4th and 7th DIP switch settings (as shown in Appendix E), the SEGx files would contain the following:

FILE NAME	SEG3.	SEG4.	SEG7.
CONTENTS	CD40	CE40	CF00

The user is **not** involved in the creation and use of these files, but he should be aware that they are necessary for proper operation of the LCIC-4HS and they should not be deleted or modified.

3. MEMORY CONFLICTS

In most cases, one or more of the addresses that MEMCHK4.EXE identifies as available will work OK. However, there is the remote possibility, that none of those addresses are actually available. In this case:

1. Check the contents of your CMOS RAM, in particular the CACHE options. Turn off the cache for any locations that the card(s) might occupy.
2. Check the contents of your CONFIG.SYS to ensure that no memory address conflicts with a TSR (Terminate and Stay Resident) program or a driver.
3. Similarly, check the contents of your AUTOEXEC.BAT to eliminate TSR or driver conflicts.
4. Sometimes, other add-on cards made by various manufacturers might use the same address, so another location should be selected for the LCIC-4HS. Specifically, check your display controller.

SUMMARY OF STEP BY STEP INSTALLATION

1. Examine and modify if necessary the CONFIG.SYS file.
2. Copy files from the LCIC-4HS software package.
3. Do NOT install card(s).
4. Run MEMCHK4.EXE.
5. Turn PC OFF & ON.
6. Run MEMCHK4.EXE.
7. Select LCIC-4HS location(s) according to the data supplied.
8. Turn PC OFF.
9. Install card(s).
10. Turn PC on.
11. Run MEMCHK4.EXE.
12. Turn PC OFF & ON.
13. Run LCIC4.EXE.

APPENDIX B: HOW TO ACCESS THE CARD IN WINDOWS

The Windows driver packages contain the required files for the programmer in Windows 95/98 or in Windows NT/2000/XP, supplying the following functions:

- Initialize card's hardware (Windows NT/2000/XP only)
- Finalize card's hardware (Windows NT/2000/XP only)
- Get card version no.
- Get weight reading in float mode
- Get input status
- Set output status
- Get output status

The drivers support Visual C++ , Visual Basic and more.

Currently, the following functions are *not* supported in Windows:

Windows 95/98

- Initialize card's hardware
- Finalize card's hardware
- Get weight reading in ASCII mode

Windows NT/2000/XP

- Get weight reading in ASCII mode

For **Windows 95/98** refer to the Win98 folder.

For **Windows NT/2000/XP** refer to the WinNT-2000-XP folder.

APPENDIX C: THE BATCH MODE

C.1 GENERAL DESCRIPTION

The "batch mode" supplies mainly two functions:

- **MANUAL VALVE ACTIVATION:** Manual activation of the fast and/or slow valve(s).
- **AUTO FILLING PROCESS:** An automatic filling process, using one or two filling speeds. Process characteristics are defined by the user.

These functions are accessible either by a supplied utility (LCIC4.EXE), or by an application program that the user himself develops. The two alternatives are described below.

C.2 CARD'S OPERATION IN BATCH MODE

- The card is operated by flow of information inwards and outwards.
- This section tells the global concept "what" the card does, through introducing the inputs and the outputs. (By definition, the terms "inputs" and "outputs" relate to CARD's side. Obviously, card's input is PC's output, and vice versa.)
- Card's input includes two categories:
 - Definitions of auto characteristics (section C.2.1).
 - Running instructions (section C.2.2).
- Card's output includes two categories:
 - Current status (section C.2.3).
 - Auto filling results (section C.2.4).
- Section C.3 describes the details "how" to use these inputs and outputs.
- The inputs and outputs are accessible by two independent means:
 - The supplied LCIC4.EXE utility
 - A user developed application program
- Section C.4 describes card activation through LCIC4.EXE utility.
- Section C.5 describes card activation through user's application program.
- Getting familiar with the easy-to-use LCIC4.EXE utility will probably give the user a better feeling of what does the batch mode do and how his own program should look like. Therefore, it would be worth while to go over section C.4 even though user's final target is to code his own application program.

C.2.1 DEFINITIONS OF AUTO CHARACTERISTICS

The "definitions of auto characteristics" specify how the auto filling, when operated by the "running instructions", will look like. Only then, the definitions will take effect. The definitions include three elements:

- Filling amount definition
- Checks definition
- Completion delay definition

C.2.1.1 FILLING AMOUNT DEFINITION

The filling process controls two valves, one allocated for fast filling, and the other for slow. The user may select fast filling, slow filling, or both (fast and then slow). Two inputs specify the fast and slow amounts. The two amounts are specified either by weighing units (e.g., kg.), or by time. The "time" specification is essential to activate an uncalibrated channel, or when the load cell is faulty. A third input selects whether the amounts specify "weight" or "time".

C.2.1.2 CHECKS DEFINITION

There are some optional checks that may be defined in order to reject or abort the process on some conditions:

- The TARE RANGE CHECK tests if the actual tare weight is within a pre-defined permissible range. An attempt to start auto filling when the actual tare weight is beyond that range will be rejected. This check may detect a malfunction such as missing or already filled vessel, eliminating the damage of "filling" in such a case.
- The TARE STABILITY CHECK tests whether the actual tare readings, during a pre-defined duration, fluctuate within a pre-defined amplitude. If the actual amplitude is too large, the auto filling process will be rejected. This check eliminates the filling process that probably would have been meaningless due to the unstable system.
- The STOP STABILITY CHECK tests system's stability AFTER closing the valve. The user may postpone the check by some pre-defined time (see next section). Again, the check includes pre-defined duration and amplitude. This time the check is not operational in card's level, since the filling process has been already terminated. However, check's result is reported to the user, letting him act accordingly.

C.2.1.3 COMPLETION DELAY DEFINITION

At filling's end, the card closes the valve, but, of course, the mechanics does not become stable at once. The user has the option to define some delay that will postpone the continuation of the process - stop stability check (if there's such), final weight reading, and process completion.

C.2.2 RUNNING INSTRUCTIONS

Two functions are controlled by the "running instructions":

- Card's modes
- Start/stop filling

C.2.2.1 CARD'S MODES

The card has two types of modes:

- Fill mode (MANUAL or AUTO)
- Zero-offset mode (OFF or ON)

C.2.2.1.1 FILL MODE

- There are two filling modes: **MANUAL** and **AUTO**. The selection of the required mode is implemented through an input.
- In the **MANUAL** fill mode the user has direct access to the valves. The card just implements the input accepted from the PC, but does not decide when a valve will be opened or closed.
When **MANUAL** fill mode is selected:
 - The definitions of auto characteristics are irrelevant.
 - The "start/stop filling" input has a **MANUAL** effect.
- In the **AUTO** fill mode the user initiates an automatic filling process which is based on his pre-determined definitions. The user has no direct access to the valves. The card decides when a valve will be opened or closed.
When **AUTO** fill mode is selected:
 - The definitions of auto characteristics are validated.
 - The "start/stop filling" input has an **AUTO** effect.
- The **initial** fill mode is **MANUAL**.

C.2.2.1.2 ZERO-OFFSET MODE

- The weights that the card uses may be either the original values (zero-offset mode is **OFF**), or shifted by a "zero-offset" (zero-offset mode is **ON**). The switching between the two zero-offset modes, as well as the definition of the offset, are implemented through a card's input.
- Zero-offset mode = **OFF** means that the weights that the card uses are on the same scale as accepted after the calibration procedure.
- Zero-offset mode = **ON** means that the weights that the card uses are shifted so that the weight when zero-offset mode was changed from **OFF** to **ON** is 0. The original value of this weight (before making it zero) is actually the offset and is supplied by a card's output.
- Initially, the zero-offset mode is **OFF** and the offset is 0.

C.2.2.2 START/STOP FILLING

- The filling process - manual or auto - is started or stopped through the "start/stop filling" input.
- When "fill mode" is MANUAL:
Turning "start/stop filling" from OFF to ON opens the "selected" valve, leaving the other one unchanged (whether open or closed). The "selected" valve (fast or slow) is specified by another input. Turning "start/stop filling" from ON to OFF closes both valves. Note that this mechanism enables to have both valves open simultaneously.
- When "fill mode" is AUTO:
 - Turning "start/stop filling" from OFF to ON: Issues a "run request" to the auto filling process, as pre- determined by the "definitions of auto characteristics". At the end of the process, the card supplies the "auto filling results". These results are kept frozen in order to let the PC read them safely.
 - Turning "start/stop filling" from ON to OFF:
 - BEFORE process completion: Issues an "emergency stop request".
 - AFTER process completion: Directs the card to terminate current session of the auto filling process, being ready for a new activity.

C.2.3 CURRENT STATUS

The card supplies the current status of the system, composed of five items:

- Zero A/D
- Current A/D
- Current zero-offset/tare
- Current weight
- Current valves status

C.2.3.1 ZERO A/D

This output supplies the zero A/D reading, as it was accepted during the calibration procedure.

C.2.3.2 CURRENT A/D

In MANUAL fill mode, this output supplies the current A/D reading.

C.2.3.3 CURRENT ZERO-OFFSET/TARE

This output is dual-purpose:

- Usually it's current zero-offset, as determined by "zero-offset mode" manipulations (refer to section C.2.2.1.2).
- Upon completion of an auto filling process, it supplies the current tare, i.e., the weight before the filling started.

C.2.3.4 CURRENT WEIGHT

This output supplies the current weight of a calibrated channel.

NOTES

- The current weight is affected by "zero-offset mode" manipulations (refer to section C.2.2.1.2).
- This output is active only in the following modes:
 - In MANUAL fill mode.
 - In AUTO fill mode:
 - When NOT in filling process: If the user specified (in an appropriate input) that this output should be updated also in AUTO fill mode. The effect of this specification is detailed in section C.3.1.1.
 - When in filling process: As long as the filled weight is still less than 80% of the required filling amount.

C.2.3.5 CURRENT VALVES STATUS

This output reports the status of the two valves (fast and slow). Each of them may be either open or closed.

C.2.4 AUTO FILLING RESULTS

Upon completing an auto filling process, its results are supplied by some outputs. These results include four elements:

- Filling time
- Stability time
- Completion code
- Data ready flag

C.2.4.1 FILLING TIME

When the filling amount is specified by kg. etc. (i.e., weighing units and not time), two outputs tell how much time took the fast and/or slow filling. The value accepted may be used later to activate auto filling process based on time instead of weight, if, for some reason, the weight is not available.

C.2.4.2 STABILITY TIME

Two outputs report the actual durations of the TARE STABILITY CHECK and the STOP STABILITY CHECK. These durations were defined by the user (refer to section C.2.1.2), but the actual time, reported here, may be slightly different.

C.2.4.3 COMPLETION CODE

The way the auto filling process terminated is supplied in an output called "COMPLETION CODE", indicating success or failure. In case of failure, the code specifies its reason. The various codes and their meaning are detailed in section C.3.2.

C.2.4.4 DATA READY FLAG

This is a bi-directional flag, through which:

- The card informs the PC that the auto filling process has been completed, the result buffers being available.
- The PC acknowledges to the card that process completion has been recognized and the required result buffers have already been read.

Section C.3.2 describes the details of the "conversation" through this flag.

C.3 CARD'S BUFFERS

Card's buffers (software inputs and software outputs) are described below. The symbols used (written in CAPITAL LETTERS) are the same symbols used by the LCIC4.EXE utility. Each buffer description is followed by "programming information" required for user's application program coding. Section [3.2](#) explains how to use this "programming information".

Notes:

1. Some of the buffers are only for an advanced usage. To start with, their description (marked "ADVANCED") may be skipped.
2. Most of the buffers below are exclusive for the batch. However, a few of them are not exclusive for the batch mode but are general-purpose buffers described already in section [3.1](#). Though, for user's convenience, their description is repeated here once more with a note that this is the case. The CURRENT WEIGHT, however, is an exception as its description is too long to be repeated, therefore only a reference is supplied in this case.

C.3.1 CARD'S SOFTWARE INPUTS

Card's software inputs are divided into two types:

- * **SETUP** SOFTWARE INPUTS
- * **OPERATION** SOFTWARE INPUTS

The **SETUP** software inputs include the inputs whose modification is rather infrequent. Typically, once such input is set, it won't be modified any more. On the other hand, the **OPERATION** software inputs are intensively updated during the batch mode activity. Both types of inputs are accessed on the same way by the LCIC4.EXE utility, as well as by user's application program.

C.3.1.1 SETUP SOFTWARE INPUTS

<div>C.3.1.1.1</div> <div>UPDATE WEIGHT IN AUTO?</div> <div>*** ADVANCED ***</div>	<div><div><div>* Specifies whether the card software output CURRENT WEIGHT should be updated when FILL MODE is AUTO, and card is NOT within a filling process.</div><div>* YES = update, NO = don't update.</div><div>* Within the filling process, the updating of CURRENT WEIGHT does NOT depend on "UPDATE WEIGHT IN AUTO?":<div><div>* As long as the filled weight is less than 80% of the total target, CURRENT WEIGHT is updated.</div><div>* Upon reaching the 80% limit, the updating stops until the valve is closed.</div></div></div><div>* Selecting YES (=update) MIGHT slightly reduce the response speed of the card during an auto filling process in this or another channel. Therefore, to be on the safe side, it would be a good practice to have this input as NO, unless the updating is really needed.</div></div><div>PROGRAMMING INFORMATION</div><table><tr><td>DATA TYPE</td><td colspan="2">4 bits</td></tr><tr><td>OFFSET</td><td colspan="2">0x0F6-L</td></tr><tr><td rowspan="2">CODES</td><td>NO</td><td>0</td></tr><tr><td>YES</td><td>1</td></tr></table></div>	DATA TYPE	4 bits		OFFSET	0x0F6-L		CODES	NO	0	YES	1
DATA TYPE	4 bits											
OFFSET	0x0F6-L											
CODES	NO	0										
	YES	1										
<div>C.3.1.1.2</div> <div>TARGET MODE</div>	<div><div><div>* Determines the units of FAST TARGET and SLOW TARGET:<div><div>* When TARGET MODE = WEIGHT, the TARGETs specify weight in kg., lb., etc. (as was selected in calibration). Only a calibrated channel may be operated with this selection.</div><div>* When TARGET MODE = TIME, the TARGETs specify time in ms. With this selection, an uncalibrated channel may be operated, as well as a channel with a faulty load cell.</div></div></div></div><div>PROGRAMMING INFORMATION</div><table><tr><td>DATA TYPE</td><td colspan="2">4 bits</td></tr><tr><td>OFFSET</td><td colspan="2">0x0F8-H</td></tr><tr><td rowspan="2">CODES</td><td>WEIGHT</td><td>0</td></tr><tr><td>TIME</td><td>1</td></tr></table></div>	DATA TYPE	4 bits		OFFSET	0x0F8-H		CODES	WEIGHT	0	TIME	1
DATA TYPE	4 bits											
OFFSET	0x0F8-H											
CODES	WEIGHT	0										
	TIME	1										

C.3.1.1.3 THE "TARE" OPTION

*** ADVANCED ***

This section is an introduction to the following items:

- * FAST TARGET
- * SLOW TARGET
- * TARE MIN.
- * TARE MAX.
- * TARE STAB. AMPLITUDE
- * TARE STAB. TIME

User's selection whether the "tare" option is active or not determines whose responsibility is to consider the initial weight before the filling starts:

- * When the "tare" is NOT active (the default), it's USER'S responsibility to consider the initial weight. For example, if the required filling quantity is 100 g, and the initial weight is 15 g, the user should specify target of 115 g, NOT 100.
- * When the "tare" IS active, it's CARD'S responsibility to consider the initial weight. For example, if the required filling quantity is 100 g, the user should specify target of 100 g, REGARDLESS of the initial weight.

The interpretation of the two card's inputs FAST TARGET and SLOW TARGET depends on whether the "tare" option is active or not - refer to the "FAST/SLOW combinations" square, below.

The following two checks are in operation only when the "tare" option is active:

- * Tare min./max. check. This check requires that the initial weight before the filling starts will be in the specified range, otherwise the RUN request is rejected.
- * Tare stability check. This check requires that during the specified time, all weight readings lie within the specified amplitude. For example, if TARE STAB. AMPLITUDE = 1 gram and TARE STAB. TIME = 100 ms, the check requires that during continuous 100 ms (beginning when the user raised the FILL START/STOP to "on") the difference from the minimal to maximal weight reading will be not more than one gram.

NOTES

- * The "switch" that determines whether the "tare" option is active is the TARE STAB. AMPLITUDE.
 - * If it's 0 (the default), the "tare" option is inactive.
 - * Otherwise, the "tare" option is active.
- * The trigger for the "tare" operation is raising the FILL START/STOP to "on". THE USER SHOULD RAISE THE FILL START/STOP ONLY WHEN THE VESSEL HAS BEEN STABILIZED ON THE LOAD CELL. A too early FILL START signal will probably cause an unwanted rejection of the RUN request, or an inaccurate filling.

<div>C.3.1.1.4</div> <div>FAST/SLOW combinations</div>	<div>This square describes the interrelation between next two card's inputs (FAST TARGET and SLOW TARGET).</div> <div>* There are three FAST/SLOW combinations possible:<ul style="list-style-type: none">* FAST only* SLOW only* Both FAST and SLOW</div>																												
	<div>1. TARGET MODE = TIME</div> <div>When FAST TARGET and SLOW TARGET specify time, their meaning is straightforward, symmetric and independent of each other.</div>																												
	<div>2. TARGET MODE = WEIGHT and TARE FUNCTION INACTIVE</div> <div>When TARE STAB. AMPLITUDE = 0, the tare stability check is NOT active, as well as the "tare" function (considering the weight prior to filling). In this case, if FAST TARGET and SLOW TARGET specify weight, their meaning is asymmetric, and SLOW TARGET's interpretation depends on FAST TARGET's value. Let us denote: SWITCH POINT (SP): CURRENT WEIGHT at switching fast filling to slow. END POINT (EP): CURRENT WEIGHT at filling's end. Then consider the following table:</div> <table><tr><th colspan="5">USER'S SPECIFICATION</th></tr><tr><th rowspan="2">REQUIRED COMBINATION</th><th colspan="2">FAST TARGET</th><th colspan="2">SLOW TARGET</th></tr><tr><th>VALUE</th><th>MEANING</th><th>VALUE</th><th>MEANING</th></tr><tr><td>FAST only</td><td>weight</td><td>abs. EP</td><td>0</td><td>no slow</td></tr><tr><td>SLOW only</td><td>0</td><td>no fast</td><td>weight</td><td>abs. EP</td></tr><tr><td>FAST+SLOW</td><td>weight</td><td>abs. SP</td><td>weight</td><td>rel. EP</td></tr></table> <div>EXPLANATION</div> <div><div>* weight: The weight that the user specifies.</div><div>* abs. SP: Absolute SWITCH POINT: SWITCH POINT = weight</div><div>* abs. EP: Absolute END POINT: END POINT = weight</div><div>* rel. EP: Relative END POINT: END POINT = SWITCH POINT + weight</div><div>* no slow: No slow filling (only fast stage).</div><div>* no fast: No fast filling (only slow stage).</div><div>Note that in any event, both SWITCH POINT and END POINT do NOT relate to the CURRENT WEIGHT when the filling process started! In other words, the CURRENT WEIGHT when FILL START/STOP is turned ON does NOT affect the SWITCH POINT and the END POINT!</div></div>	USER'S SPECIFICATION					REQUIRED COMBINATION	FAST TARGET		SLOW TARGET		VALUE	MEANING	VALUE	MEANING	FAST only	weight	abs. EP	0	no slow	SLOW only	0	no fast	weight	abs. EP	FAST+SLOW	weight	abs. SP	weight
USER'S SPECIFICATION																													
REQUIRED COMBINATION	FAST TARGET		SLOW TARGET																										
	VALUE	MEANING	VALUE	MEANING																									
FAST only	weight	abs. EP	0	no slow																									
SLOW only	0	no fast	weight	abs. EP																									
FAST+SLOW	weight	abs. SP	weight	rel. EP																									

FAST/SLOW combinations
(cont'd)

EXAMPLES

Below are three examples illustrating the three cases above (respectively):

USER'S SPECIFICATION		CARD'S RESPONSE	
FAST TARGET	SLOW TARGET	SWITCH POINT	END POINT
3	0	-	3
0	3	-	3
3	2	3	5

3. TARGET MODE = WEIGHT and TARE FUNCTION ACTIVE

When TARE STAB. AMPLITUDE > 0, the tare stability check IS active, as well as the "tare" function (considering the weight prior to filling). In this case, if FAST TARGET and SLOW TARGET specify weight, their meaning is always relative:

- * In case of "FAST only" or "SLOW only":
FAST TARGET / SLOW TARGET is relative to "tare" ("tare" is the weight prior to filling).
- * In case of "FAST+SLOW":
 - * FAST TARGET is relative to tare.
 - * SLOW TARGET is relative to SWITCH POINT.

EXAMPLES

Below are the same three examples as in paragraph 2, assuming that the tare was 1. This time, card's response does depend on the tare:

USER'S SPECIFICATION		CARD'S RESPONSE	
FAST TARGET	SLOW TARGET	SWITCH POINT	END POINT
3	0	-	4
0	3	-	4
3	2	4	6

<u>C.3.1.1.5</u> FAST TARGET	<ul style="list-style-type: none">* Specifies the filling quantity required in fast filling speed when FILL MODE = AUTO.* The value specified is either weight or time (in ms) according to TARGET MODE selection.* When only slow speed is required, specify 0.* Refer to FAST/SLOW COMBINATIONS square, above. <table><tr><th colspan="3">PROGRAMMING INFORMATION</th></tr><tr><td rowspan="2">DATA TYPE</td><td>weight:</td><td>float</td></tr><tr><td>time:</td><td>int</td></tr><tr><td>OFFSET</td><td colspan="2">0x020, 0x024, 0x028, 0x02C</td></tr></table>	PROGRAMMING INFORMATION			DATA TYPE	weight:	float	time:	int	OFFSET	0x020, 0x024, 0x028, 0x02C	
PROGRAMMING INFORMATION												
DATA TYPE	weight:	float										
	time:	int										
OFFSET	0x020, 0x024, 0x028, 0x02C											
<u>C.3.1.1.6</u> SLOW TARGET	<ul style="list-style-type: none">* Specifies the filling quantity required in slow filling speed when FILL MODE = AUTO.* The value specified is either weight or time (in ms) according to TARGET MODE selection.* When only fast speed is required, specify 0.* Refer to FAST/SLOW COMBINATIONS square, above. <table><tr><th colspan="3">PROGRAMMING INFORMATION</th></tr><tr><td rowspan="2">DATA TYPE</td><td>weight:</td><td>float</td></tr><tr><td>time:</td><td>int</td></tr><tr><td>OFFSET</td><td colspan="2">0x130, 0x134, 0x138, 0x13C</td></tr></table>	PROGRAMMING INFORMATION			DATA TYPE	weight:	float	time:	int	OFFSET	0x130, 0x134, 0x138, 0x13C	
PROGRAMMING INFORMATION												
DATA TYPE	weight:	float										
	time:	int										
OFFSET	0x130, 0x134, 0x138, 0x13C											

<p><u>C.3.1.1.7</u> <u>TARE MIN.</u> *** ADVANCED ***</p>	<p>* Specifies the minimum acceptable tare weight during the tare range check of an auto filling process.</p> <p>* Whenever the CURRENT WEIGHT is less than TARE MIN., the card will reject a RUN request (turning FILL START/STOP to ON), supplying an appropriate error indication in COMPLETION CODE.</p> <p>* When -</p> <ul style="list-style-type: none"> * TARGET MODE = TIME, o r * TARE STAB. AMPLITUDE = 0, <p>the two checks:</p> <ul style="list-style-type: none"> * Tare range check * Tare stability check <p>are inactive, and this input is irrelevant.</p> <table border="1" data-bbox="568 546 1094 689"> <tr> <th colspan="2">PROGRAMMING INFORMATION</th></tr> <tr> <td>DATA TYPE</td><td>float</td></tr> <tr> <td>OFFSET</td><td>0x0C0, 0x0C4, 0x0C8, 0x0CC</td></tr> </table>	PROGRAMMING INFORMATION		DATA TYPE	float	OFFSET	0x0C0, 0x0C4, 0x0C8, 0x0CC
PROGRAMMING INFORMATION							
DATA TYPE	float						
OFFSET	0x0C0, 0x0C4, 0x0C8, 0x0CC						
<p><u>C.3.1.1.8</u> <u>TARE MAX.</u> *** ADVANCED ***</p>	<p>* Specifies the maximum acceptable tare weight during the tare range check of an auto filling process.</p> <p>* Whenever the CURRENT WEIGHT is more than TARE MAX., the card will reject a RUN request (turning FILL START/STOP to ON), supplying an appropriate error indication in COMPLETION CODE.</p> <p>* When -</p> <ul style="list-style-type: none"> * TARGET MODE = TIME, o r * TARE STAB. AMPLITUDE = 0, <p>the two checks:</p> <ul style="list-style-type: none"> * Tare range check * Tare stability check <p>are inactive, and this input is irrelevant.</p> <table border="1" data-bbox="568 1061 1094 1205"> <tr> <th colspan="2">PROGRAMMING INFORMATION</th></tr> <tr> <td>DATA TYPE</td><td>float</td></tr> <tr> <td>OFFSET</td><td>0x0D0, 0x0D4, 0x0D8, 0x0DC</td></tr> </table>	PROGRAMMING INFORMATION		DATA TYPE	float	OFFSET	0x0D0, 0x0D4, 0x0D8, 0x0DC
PROGRAMMING INFORMATION							
DATA TYPE	float						
OFFSET	0x0D0, 0x0D4, 0x0D8, 0x0DC						

<p>C.3.1.1.9 TARE STAB. AMPLITUDE *** ADVANCED ***</p>	<p>* Amplitude limit during tare stability check. * If the actual amplitude of the fluctuations in CURRENT WEIGHT values is smaller, the test passes. Otherwise - * The test fails. * The card rejects the RUN request (turning FILL START/STOP to ON), supplying an appropriate error indication in COMPLETION CODE. * The TARE STAB. AMPLITUDE is a switch of two checks: * The tare range check * The tare stability check If TARE STAB. AMPLITUDE = 0, the two checks are inactive, disregarding the three inputs: * TARE MIN. * TARE MAX. * TARE STAB. TIME Therefore, the two checks may be easily inactivated by zeroing TARE STAB. AMPLITUDE, even though the other three inputs may still be non-zero. * When TARGET MODE = TIME, this input is irrelevant.</p> <table border="1" data-bbox="544 712 1262 860"> <tr> <th colspan="2">PROGRAMMING INFORMATION</th></tr> <tr> <td>DATA TYPE</td><td>float</td></tr> <tr> <td>OFFSET</td><td>0x1B0, 0x1B4, 0x1B8, 0x1BC</td></tr> </table>	PROGRAMMING INFORMATION		DATA TYPE	float	OFFSET	0x1B0, 0x1B4, 0x1B8, 0x1BC
PROGRAMMING INFORMATION							
DATA TYPE	float						
OFFSET	0x1B0, 0x1B4, 0x1B8, 0x1BC						
<p>C.3.1.1.10 TARE STAB. TIME (ms) *** ADVANCED ***</p>	<p>* Duration of the tare stability check. * The real duration used: * might be slightly different. * will be reported by card's REAL TARE STAB. TIME output. * When - * TARGET MODE = TIME, o r * TARE STAB. AMPLITUDE = 0, the two checks: * Tare range check * Tare stability check are inactive, and this input is irrelevant.</p> <table border="1" data-bbox="544 1193 1262 1341"> <tr> <th colspan="2">PROGRAMMING INFORMATION</th></tr> <tr> <td>DATA TYPE</td><td>int</td></tr> <tr> <td>OFFSET</td><td>0x198, 0x19A, 0x19C, 0x19E</td></tr> </table>	PROGRAMMING INFORMATION		DATA TYPE	int	OFFSET	0x198, 0x19A, 0x19C, 0x19E
PROGRAMMING INFORMATION							
DATA TYPE	int						
OFFSET	0x198, 0x19A, 0x19C, 0x19E						

<p>C.3.1.1.11 STOP STAB. AMPLITUDE *** ADVANCED ***</p>	<ul style="list-style-type: none"> * Amplitude limit during stop stability check. * If the actual amplitude of the fluctuations in CURRENT WEIGHT values is smaller, the test passes. Otherwise - * The test fails. * The auto process filling terminates abnormally, i.e., an appropriate error indication is supplied in COMPLETION CODE. * The STOP STAB. AMPLITUDE is a switch of the stop stability check: If STOP STAB. AMPLITUDE = 0, the check is inactive, disregarding the STOP STAB. TIME input. Therefore, the check may be easily inactivated by zeroing STOP STAB. AMPLITUDE, even though the STOP STAB. TIME input may still be non-zero. * When TARGET MODE = TIME, this input is irrelevant. <table border="1" data-bbox="568 667 1094 801"> <tr> <th colspan="2">PROGRAMMING INFORMATION</th></tr> <tr> <td>DATA TYPE</td><td>float</td></tr> <tr> <td>OFFSET</td><td>0x1A0, 0x1A4, 0x1A8, 0x1AC</td></tr> </table>	PROGRAMMING INFORMATION		DATA TYPE	float	OFFSET	0x1A0, 0x1A4, 0x1A8, 0x1AC
PROGRAMMING INFORMATION							
DATA TYPE	float						
OFFSET	0x1A0, 0x1A4, 0x1A8, 0x1AC						
<p>C.3.1.1.12 STOP STAB. TIME (ms) *** ADVANCED ***</p>	<ul style="list-style-type: none"> * Duration of the stop stability check. * The real duration used: * might be slightly different. * will be reported by card's REAL STOP STAB. TIME output. * When - * TARGET MODE = TIME, o r * STOP STAB. AMPLITUDE = 0, the stop stability check is inactive, and this input is irrelevant. <table border="1" data-bbox="568 1102 1094 1236"> <tr> <th colspan="2">PROGRAMMING INFORMATION</th></tr> <tr> <td>DATA TYPE</td><td>int</td></tr> <tr> <td>OFFSET</td><td>0x190, 0x192, 0x194, 0x196</td></tr> </table>	PROGRAMMING INFORMATION		DATA TYPE	int	OFFSET	0x190, 0x192, 0x194, 0x196
PROGRAMMING INFORMATION							
DATA TYPE	int						
OFFSET	0x190, 0x192, 0x194, 0x196						
<p>C.3.1.1.13 AFTER STOP DELAY (ms)</p>	<ul style="list-style-type: none"> * Specifies a delay time while terminating an auto filling process, after closing the valve, in order to let the mechanics be stabilized. * Specify 0 to indicate that there's no delay. * During the delay, CURRENT WEIGHT is updated continuously. * Upon ending the delay - * The final stability check - if there's such - takes place. * CURRENT WEIGHT is fixed, kept frozen. * Other relevant card's outputs are updated. * DATA READY is turned ON. <table border="1" data-bbox="568 1576 1094 1711"> <tr> <th colspan="2">PROGRAMMING INFORMATION</th></tr> <tr> <td>DATA TYPE</td><td>int</td></tr> <tr> <td>OFFSET</td><td>0x0A8, 0x0AA, 0x0AC, 0x0AE</td></tr> </table>	PROGRAMMING INFORMATION		DATA TYPE	int	OFFSET	0x0A8, 0x0AA, 0x0AC, 0x0AE
PROGRAMMING INFORMATION							
DATA TYPE	int						
OFFSET	0x0A8, 0x0AA, 0x0AC, 0x0AE						

C.3.1.1.14 IMPACT TIME (ms) *** ADVANCED *** CARD VER. 002 and UP	<ul style="list-style-type: none"> * Sets period of time at auto filling start in which the readings are assumed to be meaningless. * During this period the card disregards the current readings. 			
	<p align="center">PROGRAMMING INFORMATION</p> <table border="1"> <tr> <td>DATA TYPE</td><td>int</td></tr> <tr> <td>OFFSET</td><td>0x1D0, 0x1D2, 0x1D4, 0x1D6</td></tr> </table>	DATA TYPE	int	OFFSET
DATA TYPE	int			
OFFSET	0x1D0, 0x1D2, 0x1D4, 0x1D6			
C.3.1.1.15 UPDATE WEIGHT UNTIL *** ADVANCED *** CARD VER. 004 and UP	<ul style="list-style-type: none"> * Sets weight percentage until which the card updates the PC bus with the current weight. * E.g., if the target weight is 100g and UPDATE WEIGHT UNTIL = 80% (the default), the card updates the PC bus until the weight reaches (approx.) 80g, then stops updating. * The PC bus will be updated again at filling's end. 			
	<p align="center">PROGRAMMING INFORMATION</p> <table border="1"> <tr> <td>DATA TYPE</td><td>byte</td></tr> <tr> <td>OFFSET</td><td>0x1D8, 0x1D9, 0x1DA, 0x1DB</td></tr> </table>	DATA TYPE	byte	OFFSET
DATA TYPE	byte			
OFFSET	0x1D8, 0x1D9, 0x1DA, 0x1DB			

C.3.1.2 OPERATION SOFTWARE INPUTS

C.3.1.2.1

FILL MODE

* Determines the filling mode:

* MANUAL specifies that the valves are activated manually, using the next two inputs.

* AUTO specifies that the valves are activated automatically, using the next input, according to the SETUP software inputs.

* Selecting MANUAL results in continuous updating of CURRENT A/D and CURRENT WEIGHT, that MIGHT slightly reduce the response speed of the card during an auto filling process in another channel. Therefore, to be on the safe side, it would be a good practice to have this input for a calibrated channel as AUTO, unless the MANUAL mode is really needed.

PROGRAMMING INFORMATION

DATA TYPE	4 bits	
OFFSET	0x0F2-H	
CODES	AUTO	0
	MANUAL	1

C.3.1.2.2
FILL START/STOP

* Activates the filling:
* When FILL MODE = MANUAL:
* ON / OFF turns the "active" valve on or off.
* The "active" valve (fast or slow) is selected by the next input (VALVE SELECTION).
* It is legitimate to turn on both valves:
* Select FAST or SLOW valve by the next input.
* Turn to ON.
* The selected valve is turned on.
* Select the other valve by the next input.
* Now both valves are turned on.
* Selecting OFF turns off both valves.
* When FILL MODE = AUTO:
* ON signals a RUN request to the card. If everything is o.k., the auto filling process starts.
* Within the auto filling process, OFF signals an EMERGENCY STOP request to the card, resulting in immediate termination of the process. Card's COMPLETION CODE output will indicate that the process ended due to an EMERGENCY STOP request.

NOTE

Dynamic valves' status is available in the CURRENT VALVES STATUS card's output.

PROGRAMMING INFORMATION

DATA TYPE	4 bits	
OFFSET	0x0F2-L	
CODES	OFF	0
	ON	1

<div>C.3.1.2.3 VALVE SELECTION</div>	<div><ul style="list-style-type: none">* Selects the "active" valve (FAST or SLOW) when FILL MODE = MANUAL.* The active valve is the valve that will be turned on when FILL START/STOP is turned to ON.* Upon turning FILL START/STOP to OFF, both valves will be off.* When FILL MODE = AUTO, this input is irrelevant (therefore not displayed in this case by the LCIC4.EXE utility).</div> <div>PROGRAMMING INFORMATION<table><tr><td>DATA TYPE</td><td colspan="2">4 bits</td></tr><tr><td>OFFSET</td><td colspan="2">0x1F0-L</td></tr><tr><td rowspan="2">CODES</td><td>FAST</td><td>0</td></tr><tr><td>SLOW</td><td>1</td></tr></table></div>	DATA TYPE	4 bits		OFFSET	0x1F0-L		CODES	FAST	0	SLOW	1			
DATA TYPE	4 bits														
OFFSET	0x1F0-L														
CODES	FAST	0													
	SLOW	1													
<div>C.3.1.2.4 ZERO-OFFSET MODE</div>	<div><ul style="list-style-type: none">* When OFF:<ul style="list-style-type: none">* CURRENT ZERO-OFFSET is 0.* The CURRENT WEIGHT readings are set according to the same scale accepted after the calibration procedure.* Upon turning to ON, the following sequence occurs:<ul style="list-style-type: none">* CURRENT ZERO-OFFSET gets the current value of CURRENT WEIGHT.* CURRENT WEIGHT is cleared (i.e., becomes 0) and is from now on the new zero reference, shifting further CURRENT WEIGHT readings, as long as ZERO-OFFSET MODE is ON.</div> <div>NOTES:<ul style="list-style-type: none">* The ZERO-OFFSET MODE supplies an easy way to re-define system's zero, without needing the calibration utility. Its effect is in both fill modes - MANUAL and AUTO.* The current value of the ZERO-OFFSET is supplied by the output CURRENT ZERO-OFFSET.</div> <div>PROGRAMMING INFORMATION<table><tr><td>DATA TYPE</td><td colspan="2">4 bits</td></tr><tr><td>OFFSET</td><td colspan="2">0x120-L FEEDBACK BUFFER: 0x121-L</td></tr><tr><td rowspan="2">CODES</td><td>OFF</td><td>0</td></tr><tr><td>ON</td><td>1</td></tr><tr><td>NOTE</td><td colspan="2">After writing to offset 0x120-L: 1. Delay 30 ms without accessing the card. 2. Wait until card's feedback buffer (0x121-L) equals the data written to 0x120-L.</td></tr></table></div>	DATA TYPE	4 bits		OFFSET	0x120-L FEEDBACK BUFFER: 0x121-L		CODES	OFF	0	ON	1	NOTE	After writing to offset 0x120-L: 1. Delay 30 ms without accessing the card. 2. Wait until card's feedback buffer (0x121-L) equals the data written to 0x120-L.	
DATA TYPE	4 bits														
OFFSET	0x120-L FEEDBACK BUFFER: 0x121-L														
CODES	OFF	0													
	ON	1													
NOTE	After writing to offset 0x120-L: 1. Delay 30 ms without accessing the card. 2. Wait until card's feedback buffer (0x121-L) equals the data written to 0x120-L.														

C.3.2 CARD'S SOFTWARE OUTPUTS

C.3.2.1

DATA READY

- * DATA READY is an exceptional buffer: Although it is described here within card's outputs, DATA READY is the only buffer which is bi-directional, being both card's output and card's input.
- * The "conversation" through this buffer occurs upon terminating an auto filling process:
 - * The card sets DATA READY to ON to indicate that the process terminated, as well as updating the relevant output buffers, keeping them frozen in order to supply the PC a steady reading of the final position of the process, until the PC acknowledges that it's not needed any more.
 - * Upon recognizing that DATA READY is ON, the PC knows that the auto filling process ended and that the buffers are available.
 - * The PC may read these buffers.
 - * The PC sets DATA READY to OFF signaling the card that:
 - * The PC recognized that DATA READY was ON;
 - * The required buffers have already been read by the PC, so their freeze is not needed any more.
 - * After the card recognizes that DATA READY is OFF, it waits until the PC turns FILL START/STOP to OFF and then it terminates the "auto filling session" and is ready for further operation.

PROGRAMMING INFORMATION

DATA TYPE	4 bits	
OFFSET	0x0F1-L	
CODES	OFF	0
	ON	1

C.3.2.2

COMPLETION CODE

* Describes how the auto filling process terminated.
* The various cases are described in the table below.

O.K.	Normal completion.
TARE UNST.	Tare unstable - RUN request rejected.
TARE RANGE	CURRENT WEIGHT is outside tare range - RUN request rejected.
EMERGENCY	Auto fiiling process aborted due to EMERGENCY STOP request from the PC.
STOP UNST.	Weight unstable - Filling terminated abnormally.
TARGET ILL.	Illegal target specified - RUN request rejected.
SLOW ILL.	Illegal slow target specified - RUN request rejected.
UN-CALIB	Channel uncalibrated - RUN request rejected.

PROGRAMMING INFORMATION

DATA TYPE	Byte		
OFFSET	channels 1 & 2: 0x041		
	channels 3 & 4: 0x042		
CODES		CHANNEL 1/3	CHANNEL 2/4
	O.K.	0xF0	0x0F
	TARE UNST.	0x01	0x10
	TARE RANGE	0x02	0x20
	EMERGENCY	0x03	0x30
	STOP UNST.	0x04	0x40
	TARGET ILL.	0x05	0x50
	SLOW ILL.	0x06	0x60
	UN-CALIB	0x07	0x70

C.3.2.3 CURRENT VALVES STATUS	<ul style="list-style-type: none"> * Describes the status of the valves. * There are four cases: <ul style="list-style-type: none"> * blank: No valve is open. * FAST: Only the fast valve is open. * SLOW: Only the slow valve is open. * BOTH: Both valves are open. <p>Note: The CURRENT VALVES STATUS utilizes card's eight hardware outputs.</p>													
	<p align="center">PROGRAMMING INFORMATION</p> <table border="1"> <tr> <td>DATA TYPE</td><td colspan="2">4 bits</td></tr> <tr> <td rowspan="2">OFFSET</td><td>fast:</td><td>0x009-H</td></tr> <tr> <td>slow:</td><td>0x009-L</td></tr> <tr> <td rowspan="2">CODES</td><td>open</td><td>0</td></tr> <tr> <td>closed</td><td>1</td></tr> </table>		DATA TYPE	4 bits		OFFSET	fast:	0x009-H	slow:	0x009-L	CODES	open	0	closed
DATA TYPE	4 bits													
OFFSET	fast:	0x009-H												
	slow:	0x009-L												
CODES	open	0												
	closed	1												

C.3.2.4 ZERO A/D *** ADVANCED ***	<ul style="list-style-type: none"> * Zero raw A/D value, before its conversion to weight units, as accepted during the calibration. * The zero A/D is usually not useful for the user. * The LCIC4.EXE utility displays the zero A/D only in the general mode, NOT in the batch mode. * This buffer is not exclusive for the batch mode; it's a general-purpose buffer, describes already in section 3.1. Though, for user's convenience, its description is repeated here once more. 						
	<p align="center">PROGRAMMING INFORMATION</p> <table border="1"> <tr> <td>DATA TYPE</td><td colspan="2">int</td></tr> <tr> <td>OFFSET</td><td colspan="2">0x1C0, 0x1C4, 0x1C8, 0x1CC</td></tr> </table>		DATA TYPE	int		OFFSET	0x1C0, 0x1C4, 0x1C8, 0x1CC
DATA TYPE	int						
OFFSET	0x1C0, 0x1C4, 0x1C8, 0x1CC						
C.3.2.5 CURRENT A/D *** ADVANCED ***	<ul style="list-style-type: none"> * Current raw A/D value, before its conversion to weight units. * The current A/D is usually not useful for the user. * The LCIC4.EXE utility displays the current A/D only in the general mode, NOT in the batch mode. * The notes specified in the CURRENT WEIGHT square in section 3.1 are relevant also for the CURRENT A/D: Upon completing the first three steps of the reading procedure, the CURRENT A/D buffers are updated as well (in addition to the (FLOAT or ASCII) CURRENT WEIGHT buffers). * NOT affected by the ZERO-OFFSET MODE input. * Updated by the card only when FILL MODE = MANUAL. * Current A/D is irrelevant in either of these cases: <ul style="list-style-type: none"> * FILL MODE = AUTO. * Channel is not calibrated. * This buffer is not exclusive for the batch mode; it's a general-purpose buffer, described already in section 3.1. Though, for user's convenience, its description is repeated here once more. 						
	<p align="center">PROGRAMMING INFORMATION</p> <table border="1"> <tr> <td>DATA TYPE</td><td colspan="2">int</td></tr> <tr> <td>OFFSET</td><td colspan="2">0x0A0, 0x0A2, 0x0A4, 0x0A6</td></tr> </table>		DATA TYPE	int		OFFSET	0x0A0, 0x0A2, 0x0A4, 0x0A6
DATA TYPE	int						
OFFSET	0x0A0, 0x0A2, 0x0A4, 0x0A6						

<p>C.3.2.6 CURRENT ZERO-OFFSET/ TARE</p>	<ul style="list-style-type: none"> * This output is dual-purpose: * Usually it's CURRENT ZERO-OFFSET. * Upon completion of an auto filling process, it supplies the CURRENT TARE. * CURRENT ZERO-OFFSET/TARE is irrelevant in case channel is not calibrated. In this case it's not displayed by the LCIC4.EXE utility. * CURRENT ZERO-OFFSET is derived from the ZERO-OFFSET MODE: <ul style="list-style-type: none"> * If ZERO-OFFSET MODE = OFF: CURRENT ZERO-OFFSET = 0. * If ZERO-OFFSET MODE = ON: CURRENT ZERO-OFFSET = CURRENT WEIGHT value before ZERO-OFFSET MODE was set to ON. * Refer also to section C.3.1.2.4 (ZERO-OFFSET MODE square). * CURRENT TARE = CURRENT WEIGHT value before starting the auto filling process. <table border="1" data-bbox="568 674 1094 808"> <tr> <th colspan="2">PROGRAMMING INFORMATION</th></tr> <tr> <td>DATA TYPE</td><td>float</td></tr> <tr> <td>OFFSET</td><td>0x0E0, 0x0E4, 0x0E8, 0x0EC</td></tr> </table>	PROGRAMMING INFORMATION		DATA TYPE	float	OFFSET	0x0E0, 0x0E4, 0x0E8, 0x0EC
PROGRAMMING INFORMATION							
DATA TYPE	float						
OFFSET	0x0E0, 0x0E4, 0x0E8, 0x0EC						
<p>C.3.2.7 CURRENT WEIGHT</p>	<ul style="list-style-type: none"> * Current weight (only in float representation). * This buffer is not exclusive for the batch mode; it's a general-purpose buffer, described already in section 3.1. Due to its length, the description is not repeated here. Please refer to section 3.1. * If the ZERO-OFFSET MODE input is turned ON, the further CURRENT WEIGHT readings will be shifted. For details refer to section C.3.1.2.4 (ZERO-OFFSET MODE square). * When FILL MODE = AUTO, CURRENT WEIGHT is NOT updated in either of the following cases: <ul style="list-style-type: none"> * TARGET MODE = TIME * System is NOT in a filling process, and UPDATE WEIGHT IN AUTO? = NO. * System IS in a filling process, but the filled weight is more than 80% of the filling amount. (Truly, the limit is not necessarily 80%; refer to section C.3.1.1.15.) 						

<p>C.3.2.8 FAST FILL TIME (ms)</p>	<ul style="list-style-type: none"> * The time that the FAST valve was open. * Relevant only when: FILL MODE = AUTO, a n d TARGET MODE = WEIGHT. * The value accepted may be used later to activate an auto filling process using TARGET MODE = TIME and the FAST TARGET input in order to utilize an uncalibrated channel, or a channel with a faulty load cell. To use this feature: * Average some values of FAST FILL TIME. * Record various averages as a function of the physical conditions (material, temperature, humidity etc.). * When running with TARGET MODE = TIME, write to FAST TARGET the average that fits current physical conditions. <table border="1" data-bbox="568 613 1096 759"> <tr> <th colspan="2">PROGRAMMING INFORMATION</th></tr> <tr> <td>DATA TYPE</td><td>int</td></tr> <tr> <td>OFFSET</td><td>0x070, 0x072, 0x074, 0x076</td></tr> </table>	PROGRAMMING INFORMATION		DATA TYPE	int	OFFSET	0x070, 0x072, 0x074, 0x076
PROGRAMMING INFORMATION							
DATA TYPE	int						
OFFSET	0x070, 0x072, 0x074, 0x076						
<p>C.3.2.9 SLOW FILL TIME (ms)</p>	<ul style="list-style-type: none"> * The time that the SLOW valve was open. * Relevant only when: FILL MODE = AUTO, a n d TARGET MODE = WEIGHT. * The value accepted may be used later to activate an auto filling process using TARGET MODE = TIME and the SLOW TARGET input in order to utilize an uncalibrated channel, or a channel with a faulty load cell. To use this feature: * Average some values of SLOW FILL TIME. * Record various averages as a function of the physical conditions (material, temperature, humidity etc.). * When running with TARGET MODE = TIME, write to SLOW TARGET the average that fits current physical conditions. <table border="1" data-bbox="568 1193 1096 1339"> <tr> <th colspan="2">PROGRAMMING INFORMATION</th></tr> <tr> <td>DATA TYPE</td><td>int</td></tr> <tr> <td>OFFSET</td><td>0x270, 0x272, 0x274, 0x276</td></tr> </table>	PROGRAMMING INFORMATION		DATA TYPE	int	OFFSET	0x270, 0x272, 0x274, 0x276
PROGRAMMING INFORMATION							
DATA TYPE	int						
OFFSET	0x270, 0x272, 0x274, 0x276						
<p>C.3.2.10 REAL TARE STAB. TIME (ms) *** ADVANCED ***</p>	<ul style="list-style-type: none"> * Actual duration of the TARE STABILITY CHECK. * The desired duration was actually specified by the TARE STAB. TIME input, but in fact it may be slightly different, as reported here. <table border="1" data-bbox="568 1482 1096 1628"> <tr> <th colspan="2">PROGRAMMING INFORMATION</th></tr> <tr> <td>DATA TYPE</td><td>int</td></tr> <tr> <td>OFFSET</td><td>0x078, 0x07A, 0x07C, 0x07E</td></tr> </table>	PROGRAMMING INFORMATION		DATA TYPE	int	OFFSET	0x078, 0x07A, 0x07C, 0x07E
PROGRAMMING INFORMATION							
DATA TYPE	int						
OFFSET	0x078, 0x07A, 0x07C, 0x07E						
<p>C.3.2.11 REAL STOP STAB. TIME (ms) *** ADVANCED ***</p>	<ul style="list-style-type: none"> * Actual duration of the STOP STABILITY CHECK. * The desired duration was actually specified by the STOP STAB. TIME input, but in fact it may be slightly different, as reported here. <table border="1" data-bbox="568 1774 1096 1919"> <tr> <th colspan="2">PROGRAMMING INFORMATION</th></tr> <tr> <td>DATA TYPE</td><td>int</td></tr> <tr> <td>OFFSET</td><td>0x088, 0x08A, 0x08C, 0x08E</td></tr> </table>	PROGRAMMING INFORMATION		DATA TYPE	int	OFFSET	0x088, 0x08A, 0x08C, 0x08E
PROGRAMMING INFORMATION							
DATA TYPE	int						
OFFSET	0x088, 0x08A, 0x08C, 0x08E						

C.3.3 AUTO FILLING SEQUENCE

Since the activation of the auto filling process is the most complex one, the steps that the user should take, as specified already in buffers' description, have been gathered below in a compact list:

- Set card's relevant **SETUP** software inputs.
- Set the following card's **OPERATION** software inputs:
 - Turn **FILL MODE** to **AUTO**
 - To **START** the filling, turn **FILL START/STOP** to **ON**
 - To **ABORT** the filling, turn **FILL START/STOP** to **OFF** (emergency stop)
- Wait until card's **DATA READY** becomes **ON**. All relevant card's software outputs are now available, kept frozen.
- Read the essential available card's software outputs.
- Turn card's **DATA READY** to **OFF**.
- Turn **FILL START/STOP** to **OFF**.

C.4 ACTIVATION THROUGH THE LCIC4.EXE UTILITY

The various batch mode functions are accessible through the **LCIC4.EXE** utility. Run the utility by typing **LCIC4** followed by <Enter> when your PC is in the same subdirectory where the installation procedure was done. Upon activating the batch mode (by F10), two displays become available:

- The **SETUP** display –
Giving access to card's **SETUP** software inputs.
- The **OPERATION** display -
 1. Giving access to card's **OPERATION** software inputs.
 2. Showing card's software outputs, which are sampled continuously.

Each display shows all at once the information about all the channels. An uncalibrated channel is marked by a violet background colour.

C.4.1 MODIFYING CARD'S SOFTWARE INPUTS

To modify a card's software input (**SETUP** or **OPERATION**) follow these steps:

- Select the appropriate (**SETUP** or **OPERATION**) display, using the <Pg Up> or <Pg Dn> keys.
- Use the arrow keys moving the "blinker" to the required card's input in the appropriate channel.
- Press <Enter>

If current input selects an OPTION (e.g., ON/OFF):

Next option is automatically selected. If necessary, press the <Enter> again until the required option appears.

If current input is NUMERICAL:

The old input value is displayed, blinking with a black background. The old value may be either **REPLACED** by a totally new one, or **EDITED**.

To REPLACE the old value by a totally new one:

Type the new value and press <Enter>.

To EDIT the old value:

Press the left or right arrow key, then edit the old value. The <Ins> key is available to switch between "insert" and "overstrike".

Upon completing the edit, press <Enter>.

C.4.2 DATA READY TURNING OFF

The DATA READY is a bi-directional buffer, as described in "DATA READY" square in section C.3.2.

To turn off the DATA READY of some channel:

- Select the OPERATION display.
- Using the left and right arrow keys, locate the "blinker" on any input belonging to the required channel.
- Press F2.

C.5 ACTIVATION THROUGH USER'S APPLICATION

The communication with the card is through memory-mapped registers, as described in section 3.

To activate the batch mode, follow these steps:

Enter the batch mode: 1. Write the code 12 to offset 0x110. 2. Write the code 34 to offset 0x111. 3. Delay 30 ms without accessing the card. 4. Wait until both locations are zero.
Operate the required functions using card's buffers (see section C.3).
Exit batch mode: 1. Write the code 56 to offset 0x112. 2. Write the code 78 to offset 0x113. 3. Delay 30 ms without accessing the card. 4. Wait until both locations are zero.

APPENDIX D: LCIC-4HS VS LCIC-1106a

For the sake of the user who is already familiar with the 1106a, this section describes the main differences between the 4HS and the 1106a. To recall the 1106a, its User's Guide is supplied too (LCIC.PDF).

D.1 FOUR CHANNELS

Up to four channels are supported, compared with a single one in the 1106a. Consequently, the LCIC4.EXE, which is the LCIC.EXE equivalent, looks different. It displays all the channels, enabling selecting one of them for calibration manipulations. Upon selecting a channel, the LCIC4.EXE is similar to LCIC.EXE operated on the specified channel.

D.2 HARDWARE INPUT/OUTPUT

(Refer to section 5.1 in the LCIC 1106a User's Guide.)

Unlike the 1106a, the hardware input/output feature is standard, not an option. The card contains the following hardware inputs and outputs:

- 8 general-purpose hardware inputs (vs. 4 in the 1106a).
- 8 hardware outputs (as in the 1106a).
 - During the batch mode, two outputs are allocated per each channel (for the fast and the slow valves).
 - When not in the batch mode, the outputs are general-purpose, as in the 1106a.
- For further details refer to section 5.

D.3 RS232

(Refer to section 5.2 in the LCIC 1106a User's Guide.)

No RS232 option in 4HS.

D.4 CALIBRATION

D.4.1 CALIBRATION LIBRARY

(Refer to sections 2.4.2.1, 2.4.2.1.6 and appendix D in the LCIC 1106a User's Guide.)

No calibration library in 4HS.

D.4.2 MULTI CALIBRATION FEATURE

(Refer to appendix E in the LCIC 1106a User's Guide.)

No 'Multi Calibration Feature' in 4HS.

D.4.3 CALIBRATION PARAMETERS

(Refer to section 2.4.2.1.1 in the LCIC 1106a User's Guide.)

1. Parameter #1 - Calibration name:
As there's no calibration library in the 4HS, this parameter has been dropped in the 4HS.
2. Parameter #2 - Calibration points:
The 4HS supports only two calibration methods:
 - "Normal" is equivalent to "Three point calibration" in 1106a.
 - "Quick" is same as "Quick" in 1106a.For further information refer to section 2.4.2.5.1.
3. Parameter #3 - Units:
Only the following units are supported: Kg., Lbs., Gr.

D.4.4 INTEGRATION

(Refer to section 2.4.2.3 in the LCIC 1106a User's Guide.)

The Integration Factor (adjustable by F4) has three additional degrees vs. the 1106a.
Its range is 0 - 10.

D.4.5 FILTER ON/OFF

(Refer to section 2.4.2.4 in the LCIC 1106a User's Guide.)

No 'Filter On/Off' selection in 4HS. The 'Filter' is constantly 'On'.

D.4.6 SAMPLING RATE FACTOR

(Refer to section 2.4.2.5 in the LCIC 1106a User's Guide.)

No 'Sampling Rate' adjustment in 4HS.

D.4.7 CALIBRATION LOCK

(Refer to section 2.4.2.6 in the LCIC 1106a User's Guide.)

No 'calibration lock' option in 4HS.

D.4.8 CALIBRATION COUNTER

(Refer to appendix C in the LCIC 1106a User's Guide.)

No 'Calibration Counter' in 4HS.

D.5 THE "SMART" PACKAGE

(Refer to appendix G in the LCIC 1106a User's Guide.)

No "Smart" package in 4HS.

D.6 PROGRAMMER'S GUIDE

(Refer to section 3 in the LCIC 1106a User's Guide.)

The programming with the 4HS is different than with the LCIC-1106a. Refer to section 3.

D.7 AVAILABLE SETUP MEMORY LOCATIONS

(Refer to appendix I in the LCIC 1106a User's Guide.)

The locations (addresses) in 4HS are different. Refer to appendix E.

D.8 BATCH MODE

(Refer to appendix F in the LCIC 1106a User's Guide.)

The "batch mode" in the 4HS is a significant revision of the "batch mode" of the 1106a.

Please refer to appendix C.

APPENDIX E: AVAILABLE SETUP MEMORY LOCATIONS **(ABSOLUTE ADDRESSING)**

	Base Address	S1	S2	S3	S4	S5
1	C800	ON	ON	ON	ON	ON
2	CC80	OFF	ON	ON	ON	ON
3	CD40	ON	OFF	ON	ON	ON
4	CE40	OFF	OFF	ON	ON	ON
5	CE80	ON	ON	OFF	ON	ON
6	CEC0	OFF	ON	OFF	ON	ON
7	CF00	ON	OFF	OFF	ON	ON
8	CF40	OFF	OFF	OFF	ON	ON
9	CF80	ON	ON	ON	OFF	ON
10	D080	OFF	ON	ON	OFF	ON
11	D200	ON	OFF	ON	OFF	ON
12	D380	OFF	OFF	ON	OFF	ON
13	D4C0	ON	ON	OFF	OFF	ON
14	D800	OFF	ON	OFF	OFF	ON
15	DA80	ON	OFF	OFF	OFF	ON
16	DD80	OFF	OFF	OFF	OFF	ON
17	DDC0	ON	ON	ON	ON	OFF
18	DE00	OFF	ON	ON	ON	OFF
19	DF80	ON	OFF	ON	ON	OFF
20	DFC0	OFF	OFF	ON	ON	OFF
21	E000	ON	ON	OFF	ON	OFF
22	E200	OFF	ON	OFF	ON	OFF
23	E240	ON	OFF	OFF	ON	OFF
24	E2C0	OFF	OFF	OFF	ON	OFF
25	E800	ON	ON	ON	OFF	OFF
26	E880	OFF	ON	ON	OFF	OFF
27	EB00	ON	OFF	ON	OFF	OFF
28	EB80	OFF	OFF	ON	OFF	OFF
29	EC00	ON	ON	OFF	OFF	OFF
30	ED40	OFF	ON	OFF	OFF	OFF
31	EEC0	ON	OFF	OFF	OFF	OFF
32	EF80	OFF	OFF	OFF	OFF	OFF

APPENDIX F: LOAD CELL CONNECTIONS

The card is able to power up to four load cells (350 ohm or higher each), one per channel. There are two ways to connect the load cells:

- Directly
- Via an IMS adaptor (PC1112)

The direct connection has some weak points:

- (Except for LCIC-4HS Ver. 02:) It requires soldering four resistors to the DB15 connector.
- In case of a problem in a load cell, you have to:
 - Unsolder the suspected load cell and test it.
 - Solder back the disconnected load cell (or another one).

Unless all load cells have been removed, you'll have to do the soldering/unsoldering of the crowded connector in the field, usually under difficult conditions.

In order to avoid these weak points there comes the PC1112 adaptor through which you may easily connect/disconnect the load cells to/from the card - no soldering/unsoldering is required.

F.1 DIRECT CONNECTION

- (Except for LCIC-4HS Ver. 02:) Solder in the connector four 500K resistors according to the following wiring list.
- Refer also to drawing #2 (section H.2).

CONN. DB15 PIN NO.	DESCRIPTION
1,9,2	+EXCITATION (+IN)
10	−SIGNAL (−OUT) #1
3	500K resistor between −OUT #1 and +OUT #1♣ +SIGNAL (+OUT) #1
11	−SIGNAL (−OUT) #2
4	500K resistor between −OUT #2 and +OUT #2♣ +SIGNAL (+OUT) #2
12	−SIGNAL (−OUT) #3
5	500K resistor between −OUT #3 and +OUT #3♣ +SIGNAL (+OUT) #3
13	−SIGNAL (−OUT) #4
6	500K resistor between −OUT #4 and +OUT #4♣ +SIGNAL (+OUT) #4
14, 7, 15	−EXCITATION (−IN)
8	SCREEN

♣ Not required in LCIC-4HS Ver. 02.

F.2 CONNECTION VIA AN IMS ADAPTOR (PC1112)

Connect each load cell to the adaptor (PC1112) according to one of the following two wiring lists.
Refer also to drawings 3, 4 & 5 (sections H.3, H.4 & H.5).

F.2.1 SIX WIRES CONNECTION (VIA ADAPTOR)

CONN. DB9 PIN NO.	DESCRIPTION
1	+EXCITATION (+IN)
6	+SIGNAL (+OUT)
2	−EXCITATION (−IN)
7	−SIGNAL (−OUT)
3,8	N.C.
4	−SENSE
9	SHIELDING
5	+SENSE

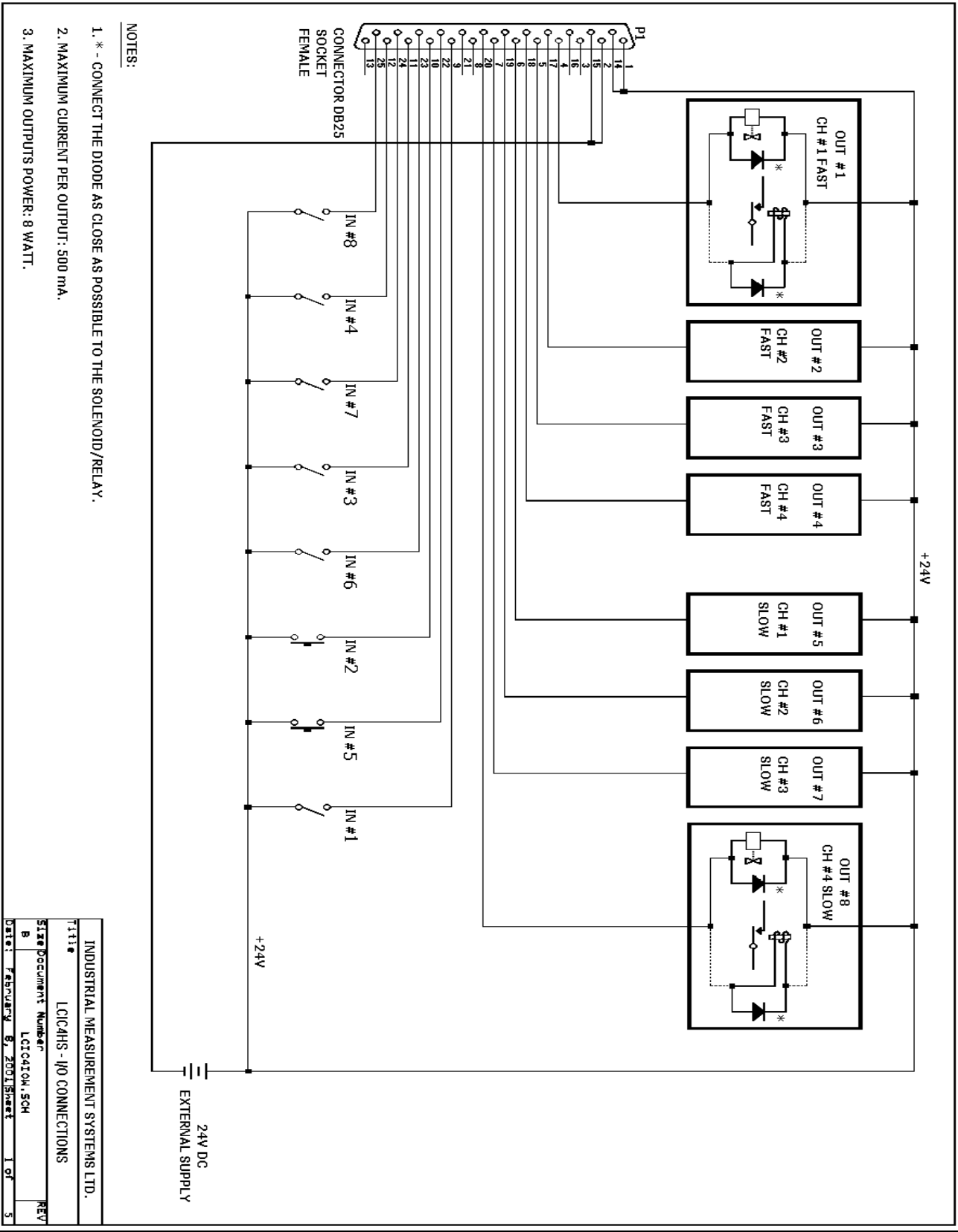
F.2.2 FOUR WIRES CONNECTION (VIA ADAPTOR)

CONN. DB9 PIN NO.	DESCRIPTION
1	+EXCITATION (+IN)
6	+SIGNAL (+OUT)
2	−EXCITATION (−IN)
7	−SIGNAL (−OUT)
3,8,4	N.C.
9	SHIELDING
5	N.C.

APPENDIX G: I/O CONNECTIONS

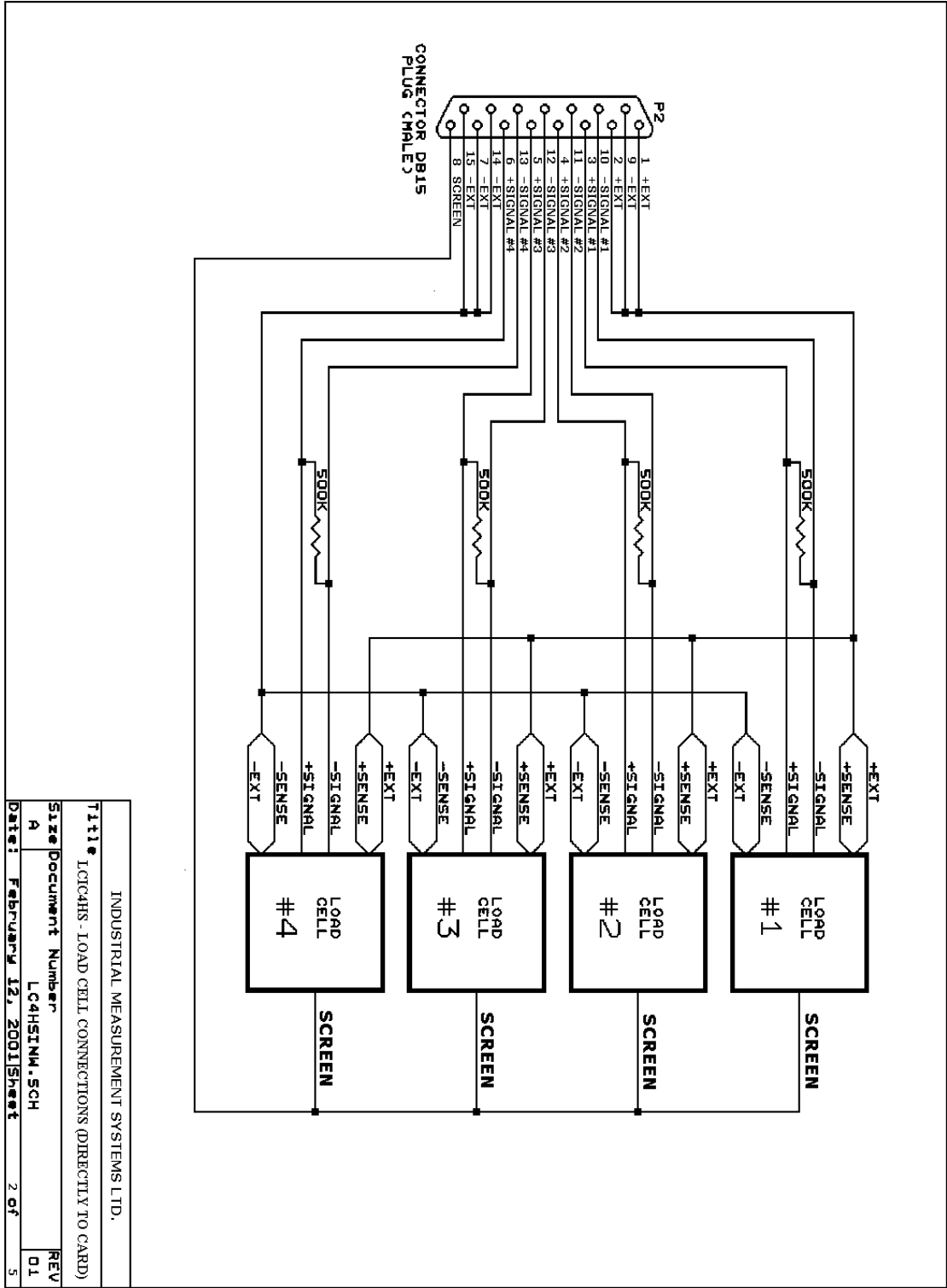
CONN. DB25 PIN NO.	SIGNAL	FUNCTION
1,14	+V EXT	
2,15	−V EXT	
3,16	N.C.	
4	OUT1	CH #1 FAST
17	OUT2	CH #2 FAST
5	OUT3	CH #3 FAST
18	OUT4	CH #4 FAST
6	OUT5	CH #1 SLOW
19	OUT6	CH #2 SLOW
7	OUT7	CH #3 SLOW
20	OUT8	CH #4 SLOW
8,21	N.C.	
9	IN1	
22	IN5	
10	IN2	
23	IN6	
11	IN3	
24	IN7	
12	IN4	
25	IN8	
13	N.C.	

APPENDIX H: DRAWINGS



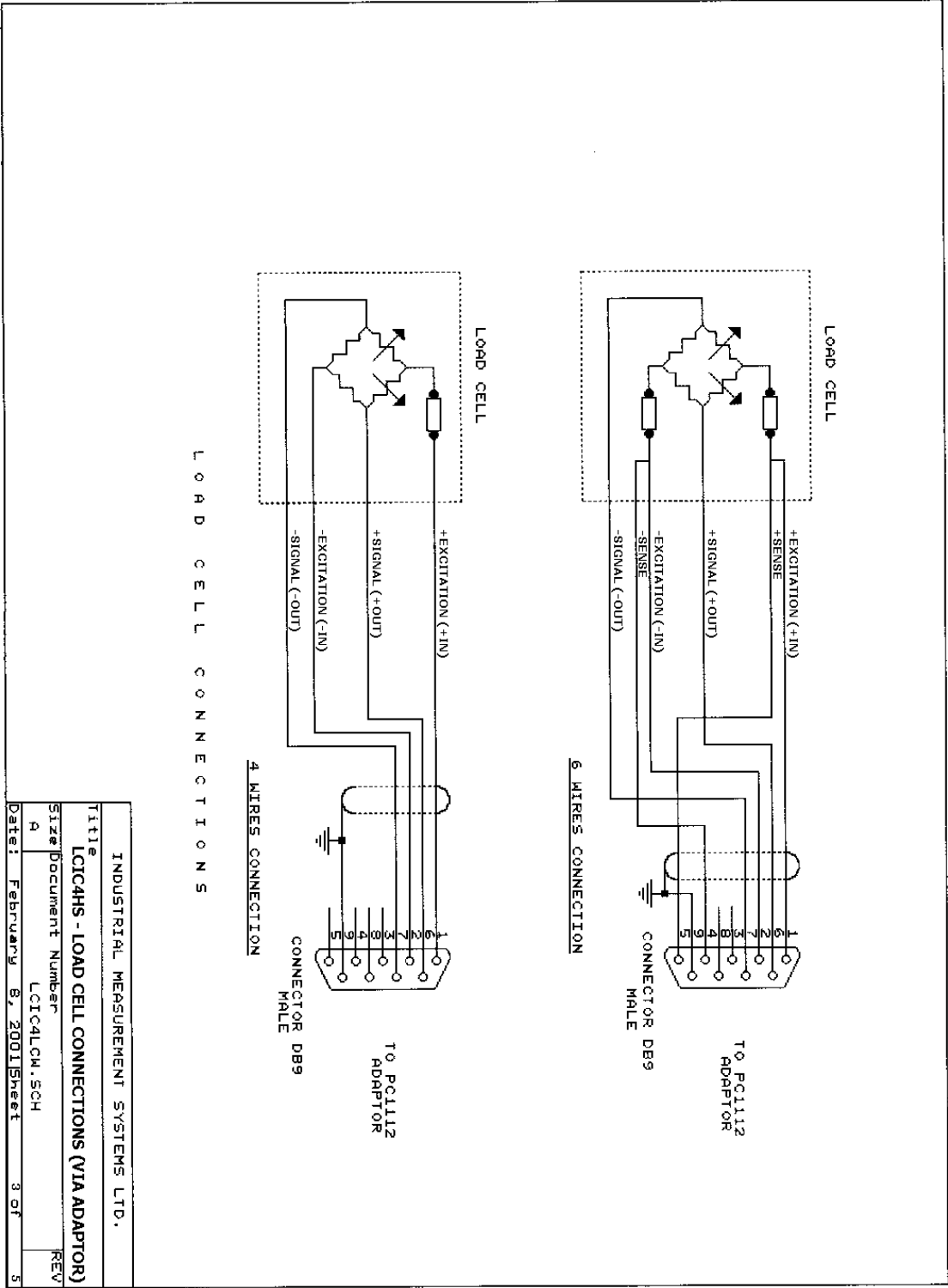
H.1: LCIC4HS – I/O CONNECTIONS

APPENDIX H: DRAWINGS



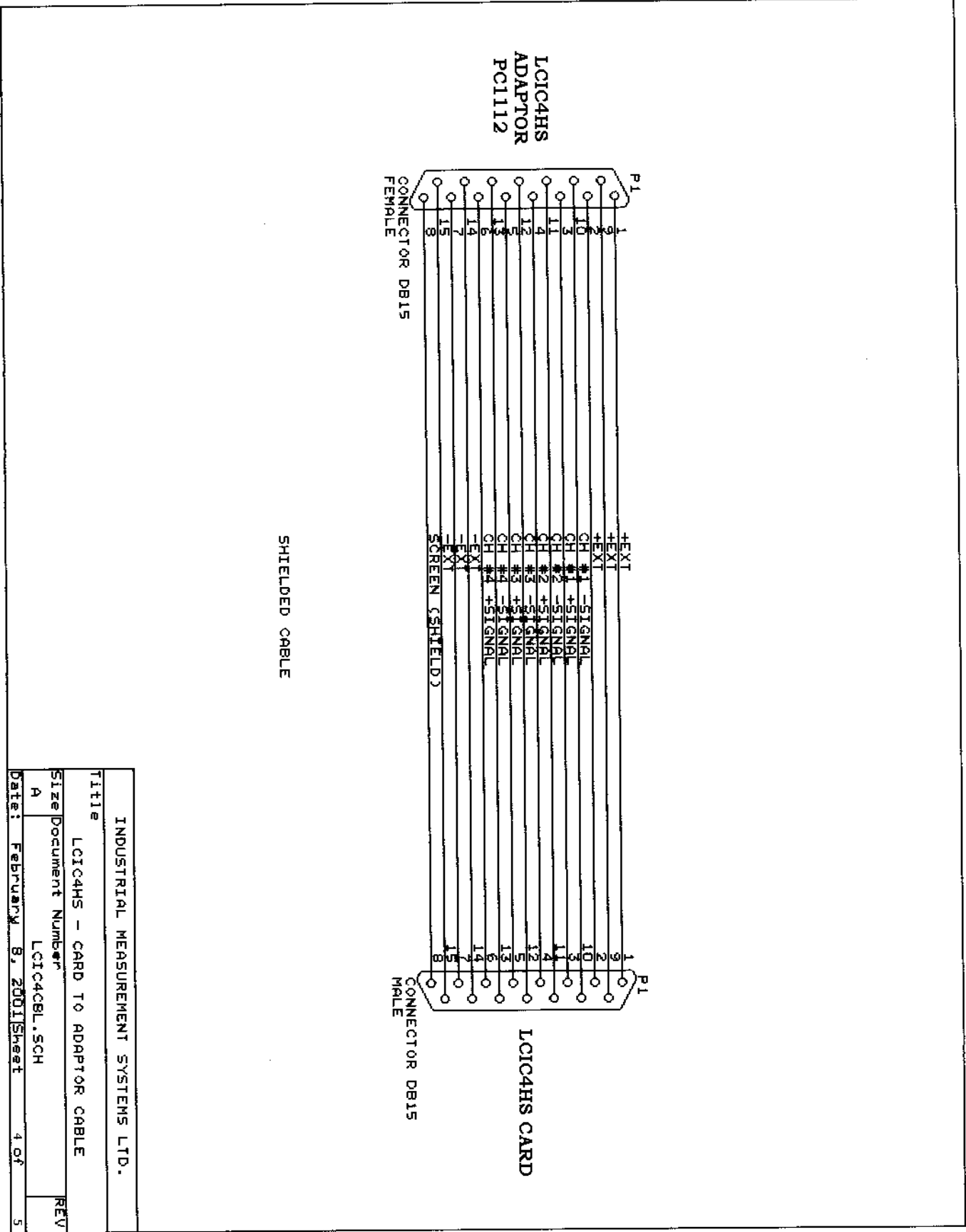
H.2: LCIC4HS – LOAD CELL CONNECTIONS (DIRECTLY TO CARD)

APPENDIX H: DRAWINGS



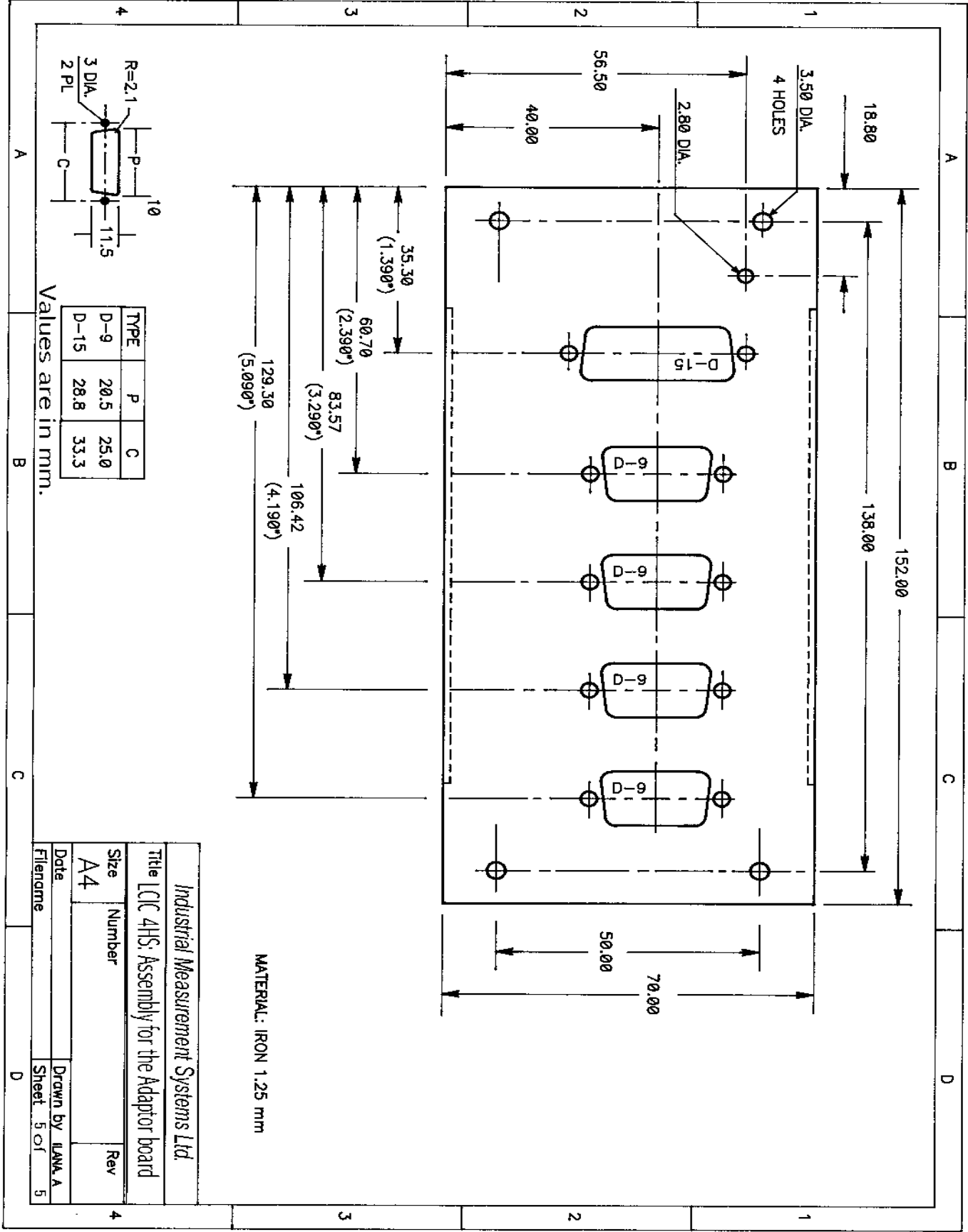
H.3: LCIC4HS – LOAD CELL CONNECTIONS (VIA ADAPTOR)

APPENDIX H: DRAWINGS



H.4: LCIC4HS - CARD TO ADAPTOR CABLE

APPENDIX H: DRAWINGS



H.5: LCIC4HS – ASSEMBLY FOR THE ADAPTOR BOARD