C500 Series
Digital Indicator Reference Manual

rinstrum

SMART WEIGHING SOLUTIONS

Copyright

Disclaimer

Rinstrum Pty Ltd reserves the right to make changes to the products contained in this manual in order to improve design, performance or reliability. The information in this manual is believed to be accurate in all respects at the time of publication, but is subject to change without notice. Rinstrum Pty Ltd assumes no responsibility for any errors or omissions and disclaims responsibility for any consequences resulting from the use of the information provided herein.

Warning

This is a class A product. In a domestic environment this product may cause radio interference in which case the user may be required to take adequate measures.

Contents

1 Introduction

The C520 is a precision digital indicator. The C530 is a precision digital weight transmitter. Both use a sigma-delta analog-to-digital (A/D) converter to ensure extremely fast and accurate weight readings. This advanced technology allows the C520 and C530 to be configured for up to 100,000 divisions with up to 100 A/D conversions per second. The units have extended sensitivity adjustment which can handle scales with outputs 0.2-5.0mV/V for full range. The design is optimized to deliver precision performance on scale bases delivering above 0.5mV/V at display resolutions of up to 10000 divisions in single range mode, or 5000 divisions in dual range mode.

Throughout this document the two devices will be referred to as the C500 series.

The C520 digital weight indicator has a green six digit 14 segment LED display with additional indication of weighing status. Each digit is 14mm high. The instrument has 8 setpoints with status display on the front panel. The C530 digital weight transmitter has no display, but can be accessed by the C500 series viewer PC software.

The C500 series can either operate from mains AC supply (86-260VAC 48-62Hz) or from stable DC supplies (12-24VDC).

The setup and calibration are digital, with a non-volatile security store for all setup parameters. The built-in clock can be used to date-stamp printouts. There is an NVRAM store to ensure day to day operating settings (zero, tare, date/time, etc) are retained when power is removed.

Infrared communications ports for easy configuration are available. The C520 provides two (front and rear), whilst the C530 provides one.

Two optional accessory modules can be fitted to the indicator. These provide additional hardware interface features such as:

- AC power input
- Additional serial ports for RS232 and RS485
- Inputs and outputs
- Analog outputs

2 Installation

2.1 Introduction

The C520 can be used as either a desk-top or panel-mount instrument. The C530 is intended for installation in electronics cabinets. The C500 series contains precision electronics and must not be subject to shock, excessive vibration, or extremes of temperature, either before or after installation. The operating environment must fall within the allowed temperature range and humidity.

The inputs of the C500 series are protected against electrical interference, but excessive levels of electromagnetic radiation and RFI may affect the accuracy and stability of the instrument. The C500 series should be installed away from any sources of electrical noise. The loadcell cable is particularly sensitive to electrical noise, and should be located well away from any power or switching circuits. Termination of the loadcell shield at the indicator end (with a sound connection to the indicator case via the DB9 plug or screw terminal shield connection) is important for EMC immunity.

The C500 series must be installed in a manner and location that is protected from impacts.

Warning!

If the equipment is not installed and used as specified by the manufacturer, the protection provided by the equipment may be impaired.

2.2 Environmental Conditions

The C500 series is designed for use in the following environmental conditions:

- C520: Indoor use only. Although the front of the device can be washed down (as per IP65), the rear of the device, along with cable attachments and optional accessories, must be protected from liquids and small objects (as per IP30).
- C530: Indoor use only. The device cannot be washed down. The device, along with cable attachments and optional accessories, must be protected from liquids and small objects (as per IP30).
- Altitude: up to 2000m
- Operating temperature range: -10 °C to 40 °C
- Storage temperature range: -20 °C to 50 °C
- Humidity: Maximum relative humidity of 80% for temperatures up to 31 °C decreasing linearly to 50% at 40 °C
- Mains supply voltage fluctuations: up to ± 10%.
- Pollution degree: 2

2.3 Mounting

2.3.1 C520 Panel Mounting

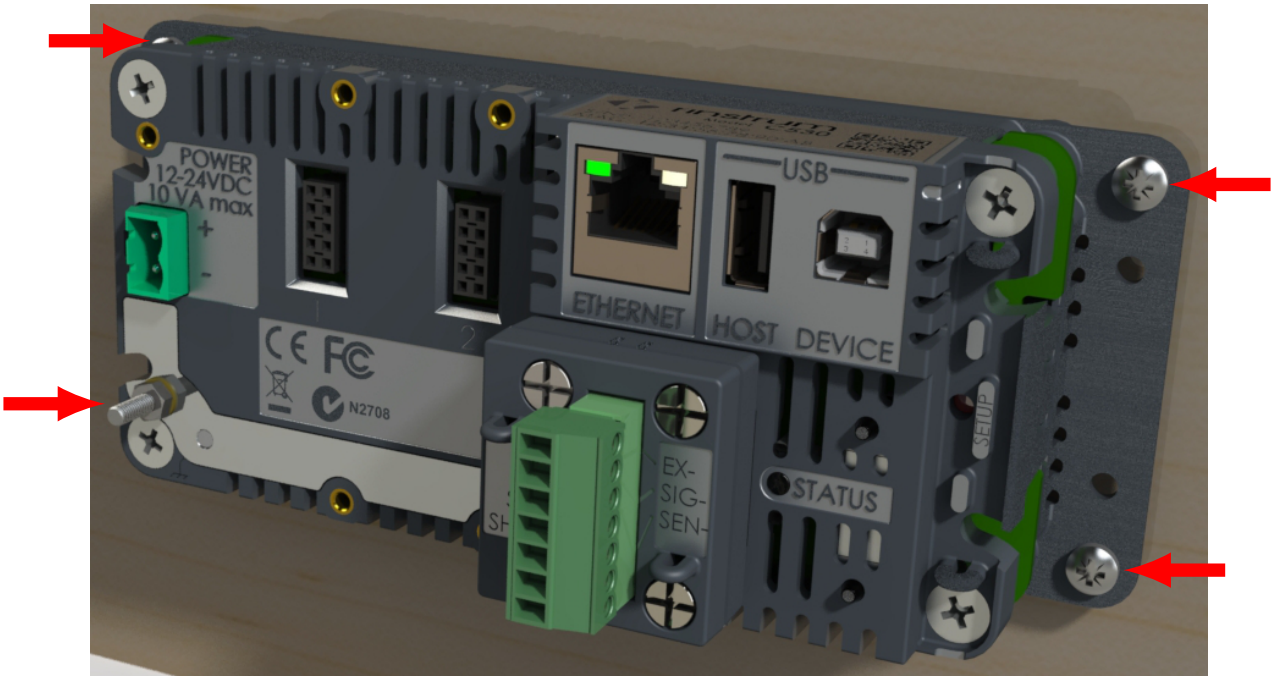
The C520 fits a DIN 43 700 standard 138(-0/+1)mm x 67(-0/+1)mm panel cut-out. Panel mounting is included standard. To panel mount:

1. Insert the 4 supplied M4 screws in the mounting tabs in the corners.
2. Insert the indicator into the panel and rotate the 4 tabs out from the indicator.
3. Tighten the screws to clamp the indicator into the panel. Do not over tighten.

2.3.2 C530 Surface Mount

The C530 can be attached to a surface using the four mounting holes.

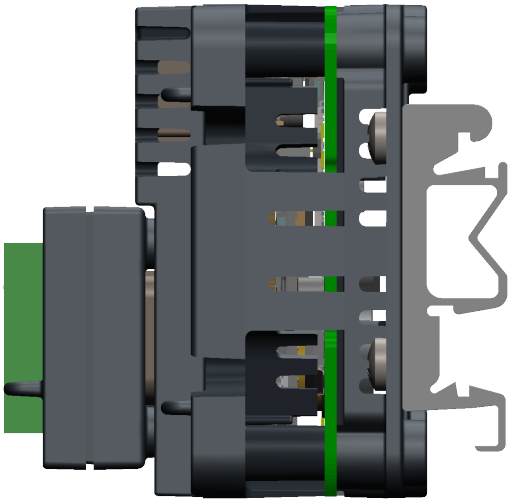
1. Use four M4 bolts, or four #6 screws to attach the C530 to the mounting surface as shown below.

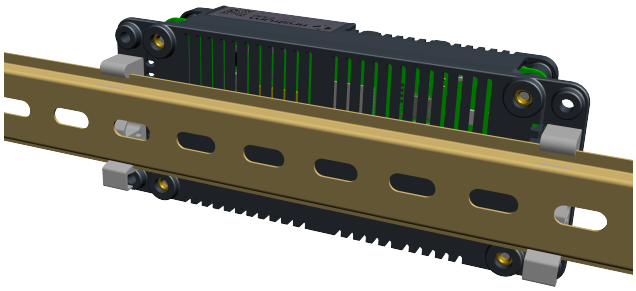


2.3.3 C530 DIN Rail Mounting

The C530 can be attached to a DIN rail using the optional DIN rail mounting kit:

1. Attach the two DIN rail mounts to the C530 using the four supplied screws
2. Clip the C530 to the DIN rail





2.4 Power Connection

The C500 series can be powered from DC or AC supplies.

2.4.1 DC Power

The C500 series requires a 12-24VDC power supply. The supply need not be regulated, provided that it is free of excessive electrical noise and sudden transients. The C500 series can operate from good quality plug-packs of sufficient capacity to drive both the indicator and the loadcells.

The case ground connection is available via the earth stake on the rear of the unit. The resistance measured between the case of the indicator and the nearest earth point should be less than 2 ohms.

2.4.2 AC Power Module

There is an optional AC supply module which requires an AC input of 100-240VAC, 50-60Hz, ≈0.6A. Maximum supply voltage variations should not exceed ±10% of the voltage range specified.

The AC supply module also provides a 12DC 0.5A(max) output. Attachment of the AC module is shown below.

2.5 Loadcell Connection

2.5.1 Overview

The C500 series can drive any number of full bridge strain gauge loadcells up to the equivalent of 16 x 350 ohm cells (21 ohm load).

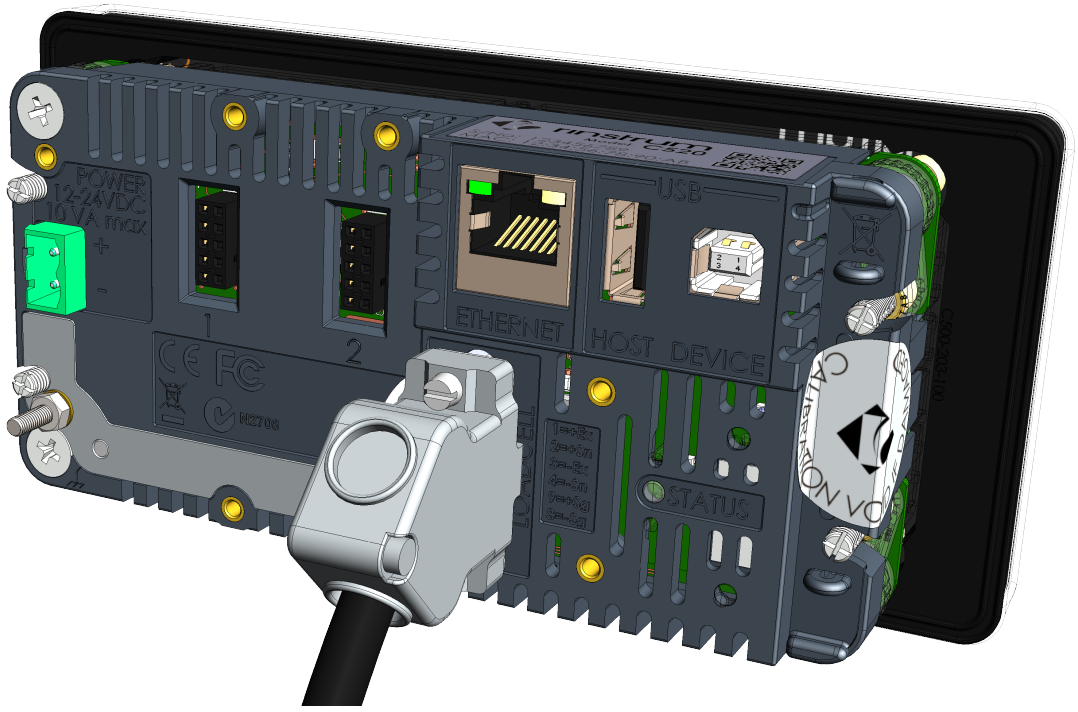
The span range of the loadcell outputs (the change of signal from the loadcells between zero load and full gross load) must be within the range of 0.2 to 5.0 mV/V. Very low output scale bases can be used with the C500 series, but may induce some instability in the weight readings when used with higher resolutions. Generally speaking, the higher the output, or the lower the number of divisions, the greater the display stability and accuracy.

When shunting loadcells, use only good quality metal film resisters with high temperature stability ratings. Typical values for zero adjustment would fall within the range of 500k ohms (small effect) to 50k ohms (larger effect).

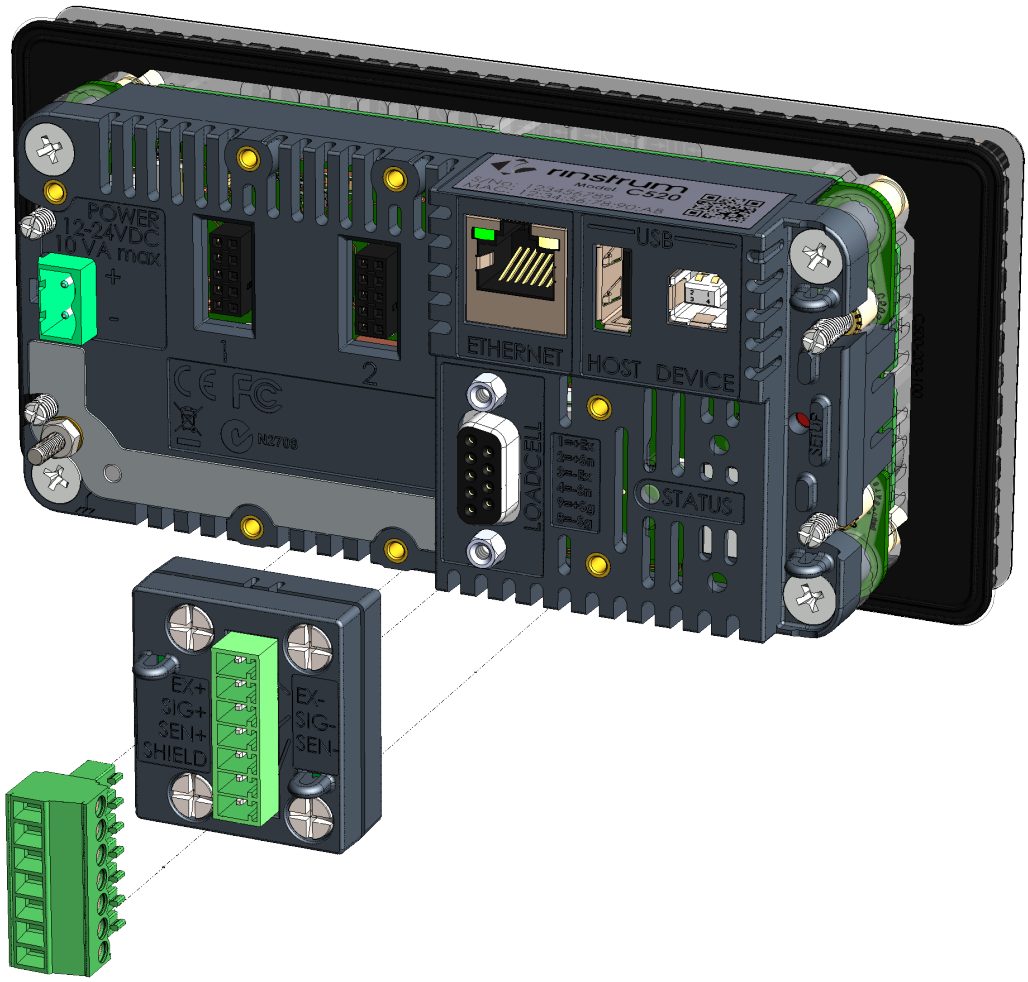
The C500 series has a mV/V meter test mode which can be used to check scale base signal output levels. Refer to Section 5.8.

> **Warning!**
> Sense lines must be connected. Failure to do this will result in the C500 series displaying an error message (E00040, E00080 or E000C0).

The C500 series offers two loadcell connection options. For replacement of 5000 indicators, the DB9 connection can be used, as shown below:



For new installations, the loadcell adaptor can be attached to the rear of the indicator. This permits easier connection of the loadcell using screw terminals.



2.5.2 Cable

When wiring loadcells use only high quality shielded multi-core cable. The cable should be run as far away from any other cabling as possible (minimum separation distance 150mm). Do not bundle loadcell cables with power or control switching cables as interference can trigger display instability, and cause unreliable operation.

The loadcell shield must be installed so as to connect electrically with the metal shell of the DB9 plug or screw terminal shield connection in order for the C500 series to provide its full EMC resistance. Any noise absorbed by the cable shield must be conducted as quickly as possible to the indicator case via the DB9 plug shell, then direct to a solid earthing point via the earth terminal in the power input socket (or earthing lug on DC models).

2.5.3 Six Wire Connection

The connection is made using a standard DB9 male plug. The loadcell socket is wired for six wire systems as follows:

DB9 Pin	Screw Terminal Pin	Function
1	1	Positive excitation
2	5	Positive sense
3	2	Negative excitation
4	6	Negative sense
9	3	Positive signal
8	4	Negative signal
5	7	Cable shield

2.5.4 Four Wire Connection

When a four wire loadcell system is connected some solder or wire bridges are used to ensure that the excitation voltages are fed into the sense inputs. For DB9 connections, short the following:

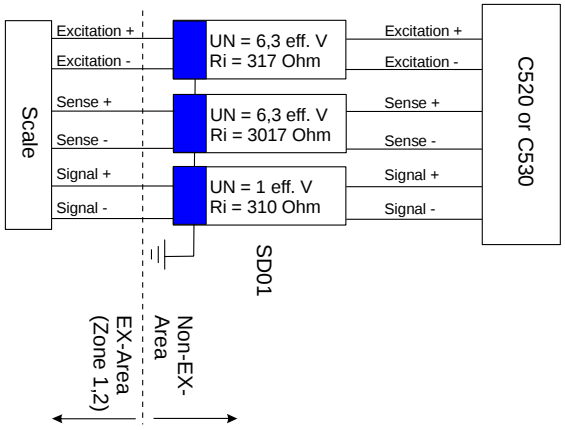
EX pin	Short to
1	2
3	4

For screw terminal connections, short the following:

EX pin	Short to
1	5
2	6

2.6 EX-I Loadcell Connection

The C500 series can be installed with barriers for EX-i applications. Six-wire connection is necessary in this case to achieve an acceptable performance. The C500 series is not intrinsic safe and must be installed outside the hazardous area. Connections for an EX-I application are:



If errors E00040, E00080 or E000C0 occur in an EX-I application the check of the voltage drop over the sense lines can be turned off with the `OPTION:SENS.CH` setting. Thus the C500 series will accept a higher voltage drop over the sense lines, but will no longer detect problems with the sense connections.

2.6.1 Non-trade Limits

C520/C530 with zener barrier SD01 (Input signal $\geq$ 0,2 μV/e, divisions 3000d, loadcells 2mV/V)

No of loadcells	Minimum yield of the loadcells 350 Ω	Minimum yield of the loadcells 700 Ω
1	7 %	6 %
2	11 %	7 %
3	14 %	9 %
4	17 %	11 %
6	24 %	14 %
8	31 %	18 %

2.6.2 Trade Limits

C520/C530 with zener barrier SD01 (Input signal $\geq$ 1,0 μV/e, divisions 3000d, loadcells 2mV/V)

No of loadcells	Minimum yield of the loadcells 350 Ω	Minimum yield of the loadcells 700 Ω
1	36 %	28 %
2	53 %	36 %
3	70 %	45 %
4	87 %	53 %
6	-	70 %
8	-	87 %

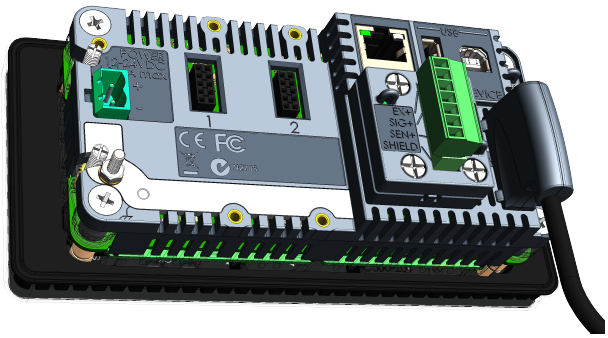
The minimum yield of the loadcells is proportional to the divisions. E.g. at 2000d and 4 loadcells 350 ohm the minimum yield is 58%. These calculations are valid for an overall maximum cable resistance of 3.5 ohm per lead. Larger cable resistances increase the minimum yield.

2.7 Optical Communications

A temporary infrared communications link can be established between the instrument and a PC using an optional cable. This connection can be used to transfer setup and calibration information from a PC.

The PC end of the cable is a standard USB connector. The instrument end of the cable attaches to the left side of the indicator display, or the rear of the indicator as shown below.

Warning!
The optical coupling head contains a strong magnet and should not be placed near any magnetic storage media (eg. credit cards, floppy disks etc.)



2.8 Optional Module Connections

Two optional modules can be connected. These provide a range of external drivers and features. See Section 15 and the module datasheets (for module specific connection details).

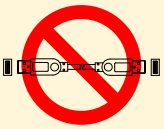
2.9 Ethernet Connection

A standard 10/100 Ethernet port is provided. See Section 14.

2.10 USB Host and Device Connections

A USB host and USB device interface are provided. See Section 10.

Warning!
Do not connect the USB host port to another USB host port. It is possible to purchase USB A Male to USB A Male cables as shown in the picture below that permit such a connection. Making such a connection will damage the indicator and PC, voiding the warranty on both devices.



3 User Interface

3.1 General

The front panel of the C520 has a six digit LED display and a 6 key keypad. A cover on the rear allows access to a hidden key which can be used to enter full digital setup and calibration. The diagram below shows the main elements of the front panel.

The C520 user interface includes:

1. 6 digit 14 segment LED display
2. Units indicator
3. Weighing status
4. Setpoint status
5. 6 key keypad
6. Multiple range/interval status
7. Check weigh status
8. Rinlink attachment
9. Full setup key (on the rear of the indicator)

3.2 Display Functions

- Display (1): weight readings, errors, information and setup.
- Units indicator (2): units for the weight reading (pounds (lb), kilograms (kg), grams (g), ounces (oz), and tons (t)).
- Status indicators (3), (4), (6) and (7): The indicator bank made up of 4 groups of LEDs. The weighing status (3) shows the status of the displayed reading. The multirange status (6) shows the current multirange operation. The setpoint status set (4) shows the status of the 8 setpoints (IO).

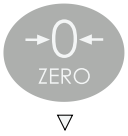
→0←	Lit when the displayed reading is within $\pm\frac{1}{4}$ of a division of true zero.
~	Lit when the displayed reading is in motion.
NET	Lit when the displayed reading represents net weight.
⌊⌋	Lit when the displayed reading is within the zero range.
H	Lit when the display reading has been held.
→|1|← →|2|← →|3|←	Ranges 1 and 2 (multiple range/interval modes only). The three annunciators are flashed within the setup menus to indicate the current menu level.
1 2 3 4 5 6 7 8	Lit to indicate when a setpoint output is active.
	Lit to indicate various states during checkweigh.

3.3 Keys

The C520 has 6 front panel keys that control the operation of the instrument. The 7th key (SETUP) is on the rear of the instrument. The setup key can be sealed to prevent unauthorized tampering of trade critical settings and calibration. Each of the front panel keys has two separate functions:

- A normal function that is available during normal weighing (as printed on the key). These are described below.
- A setup function which is available during setup and calibration (as printed beneath the key). Refer to Section 5.1.4.

3.3.1 Zero Key



▽

Normal function	Zero	Zero the scale.
Long press function	Cancel zero	Cancel all previous zero operations (not available in trade mode).

The range of the zero setting is limited according to settings `OPTION:USE` and `OPTION:Z.RANGE`.

3.3.2 Tare Key

◁

Normal function	Tare	Tare the scale.
Long press function	Preset tare	Enter a preset tare value to apply to the scale.

The Tare key can operate over the entire weight range. In trade mode the Tare key will not operate if the gross weight is negative.

3.3.3 Gross/Net Key

▷

Normal function	Gross / Net	Toggle between gross and net weight.
Long press function	Operator and setup menus	Access to the operator and setup menus and alibi mode.

3.3.4 Function 1, 2 & 3 Keys

Normal function	Configurable	Configured in setup.
Long press function		Depends on configured normal function.

3.3.5 Rear Setup Key

Normal function	Setup	Enter and exit full setup. See Section 5.1.4.

4 Licensing

The C500 series uses licence codes to activate software packages installed on the indicator. All packages with names in the C500-K5xx and L9xx-5xx range will require a license to be entered.

Licence codes are unique for each indicator and can be acquired from the nearest Rinstrum office.

4.1 Unlicensed Packages

If a package requiring a license is installed but not yet licensed, a message like “C500” “K501” “NOT” “LCNSED” will be displayed when the indicator has finished startup. In this example the package C500-K501 requires a license. Care must be taken to ensure the license entered matches the package name the indicator is displaying.

When an unlicensed package is installed, the indicator can not be used until the package is licensed or uninstalled.

To allow temporary use of the indicator, license entry can be skipped by pressing the cancel key. The indicator will allow 5 minutes of normal operation before returning to a license entry prompt. If the license is not successfully entered at this stage the indicator will restart.

4.2 Licence Code Entry

To license a package on the C500 series, first install the required firmware (see the Install Firmware section for details). After successful startup, the indicator will prompt the user with the text ‘C500”, “K501”, “NOT”, “LCNSED”. Pressing the OK button will allow the user to enter the licence code.

Once the license code has been entered, press OK. If successful the indicator will prompt “LCNSE” “OK” and the indicator will continue to normal operation.

Note

The ◄► keys are used to move to the next letter and the ▼▲ keys are used to select the appropriate letter.

Alternatively, the network communication command `LRP` can be used to license pacakges (see Section 21.3.32).

5 Setup

5.1 General Information

5.1.1 Introduction

Setup and calibration is carried out entirely from the front panel using the setup functions on the control keys. There are 3 types of setup:

- Full setup: Allows access to all setup parameters, including calibration.
- Safe setup: Allows access to setup parameters which are not critical for trade operation of the scale.
- Operator setup: Allows access setpoint targets and inflights. See Section 6.

The setup menus are a menu tree of parameters. All parameters can be edited using the C500 series interface. They can also be edited using the PC configuration tool.

Note

There are several layers of menu settings. This document uses a colon to show these layers. For example, `BUILD:DP` refers to the DP setting (decimal point) in the BUILD menu.

5.1.2 Setup security

There are 2 types of security for unauthorised setup access:

- Passcodes:
  - Access to the setup can be passcode protected to prevent unauthorized tampering.
  - Safe and full setup have separate passcodes. These are set in `SPEC:SAFE.PC` and `SPEC:FULL.PC`
  - The full setup passcode can be used to access safe setup.
  - The safe passcode cannot be used to access full setup.
  - Set these passcodes to 0 to disable passcode protection. This is the default.
- Physical seals: It is possible to disable access to the full setup via the gross/net key (using `SPEC:R.ENTRY`). The rear key must then be used to access full setup. The rear key can be sealed via physical seals such as destructible labels, lead seals, etc.

5.1.3 Entry

There are 2 methods of entering the setup program.

- Safe/full setup (front keys):
  1. Long press the Gross/net key.
  2. Press the Gross/net key to choose which level of setup is required.

Note

Full setup will not be available if this has been disabled in the setup

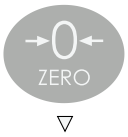
  3. Press the OK key.
  4. If a passcode has been set: Use the Gross/net and Print keys to enter the passcode. Press the OK key to confirm the passcode.

- Full setup (rear key):
  1. Press the key on the rear of the indicator. The security cover may have to be removed.
  2. If a passcode has been set: Use the Gross/net and Print keys to enter the passcode. Press the OK key to confirm the passcode.

5.1.4 Navigation

The setup menus are organised in a tree structure. Main menus are called groups. Groups contain sub-groups and items. Items are settings which can be edited. All items in a groups or sub-groups have related functions.

Zero Key



Setup menu function	Navigate first level	Step through the list of top level menu items. Long press to step in reverse direction.
Setup editor function	Down	Decrement selected digit in number editor. Previous option in bit and list editor.

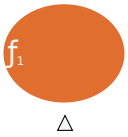
Tare Key

Setup menu function	Navigate first level	Step through the list of top level menu items. Long press to step in reverse direction.
Setup editor function	Down	Decrement selected digit in number editor. Previous option in bit and list editor.

Gross/Net Key

Setup menu function	Navigate third level	Step through the list of third level menu items. Long press to step in reverse direction.
Setup editor function	Right	Navigate right when editing numbers, string and bit editors.

Function 1 Key

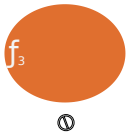


Setup menu function	Navigate fourth level	Step through the list of fourth level menu items. Long press to step in reverse direction.
Setup editor function	Up	Increment selected digit in number editor. Next option in bit and list editor.

Function 2 Key

Setup menu function	Descend menus/start edit	Descend to the next level down of menus or start editing the current menu item. Long press to save changes and exit setup.
Setup editor function	OK	Accept the current change. Long press to save changes and exit setup.

Function 3 Key



Setup menu function	Ascend menus	Ascend to level above of menus.
Setup editor function	Cancel	Cancel the current changes.

Rear Setup Key

Setup menu function	Save and exit setup	Save changes and exit setup.
Setup editor function	Save and exit setup	Exit current editor, save changes and exit setup.

5.1.5 Editing Option Items

Some settings allow the choice of an option from a predefined list of options. Examples are `BUILD:DP` or `OPTION:USE`. To show/edit:

- Press the OK key to show the current setting
- Press the up or down keys until the correct setting is shown
- Press the OK key to exit the editor

5.1.6 Editing Weight and Number Items

Some settings require the entry of a weight or other number. Examples are `BUILD:CAP1` or `OPTION:Z.BAND`. The correct decimal point and units (if applicable) are shown while editing. To show/edit:

- Press the OK key to show the current setting
- Use the left and right keys (change digit) and the up and down keys (increment/decrement digit) to show the correct value
- Press the OK key to exit the editor
  - If the setting is not possible (for example if the value is greater than the allowed maximum), —— is shown and the editor will not exit.

5.1.7 Editing String Items

Some settings require the entry of a text or token string. Examples are `SERIAL:AUT.OPT:AUT.FMT` or `SERIAL:PRN.OPT:HEADER`. To show/edit:

- Press the OK key to show the first character in the string
  - The character is shown as xxx.yyy.
  - xxx is the character position in the string (e.g: 001, 002, 003, etc).
  - yyy is the ASCII code for the character (e.g: 065=A, 002=STX). See the ASCII table in Section 9.5.
- To enter the ASCII value, use the left and right keys to change the selected digit and the up and down keys to change that digit
- Use the OK key to advance to the next character in the string
- Press the cancel key to exit the editor (changes will be saved)

5.1.8 Editing With Functions

Some settings have a special function to control their use. These are not simple settings but are more complex routines. Examples are `CAL:ZERO` or `TEST:SCALE`. To use:

- Press the OK key to start the function
- All functions are different and there are no standard keys. The user will be prompted for what is required.
- Use the cancel key to exit the function

5.1.9 Exit

There are several methods of exiting the setup menu.

- Method 1: Save and exit
  - Long press the OK key.
- Method 2: Save and exit
  1. Press the Zero key until -END- is shown.
  2. Press the OK key
- Method 3: Save and exit
  - Press the rear key
- Method 4: Exit without saving
  - Remove the power from the instrument.

5.2 BUILD: Scale Build Menu

Items in this group configure the scale build. It is important to set the build options before calibrating. Changes after calibration may invalidate the calibration.

Items		Name	Description
Scale type	⊕	TYPE	Selects the number of ranges used. Options are: • SINGLE(def): Single range • DUAL I: Dual interval • DUAL R: Dual range
Decimal point position	⊕	DP	Sets the decimal point position on the display. Options are: • 000000(def) • 00000.0 • 0000.00 • 000.000 • 00.0000 • 0.00000
Range 1 scale capacity	⊕	CAP1	Sets the maximum capacity of the scale. If `BUILD:TYPE=DUAL_I` or `BUILD:TYPE=DUAL_R`, it sets the capacity of the 1st range. Default: 3000.

Range 1 verification interval	⊕	E1	Sets the minimum verification interval of the scale (as displayed). If `BUILD:TYPE=DUAL_I` or `BUILD:TYPE=DUAL_R`, it sets minimum verification interval of the 1st range. Options are: • 1(def) • 2 • 5 • 10 • 20 • 50 • 100
Range 2 scale capacity	⊕	CAP2	Sets the maximum capacity of range 2. Only used if `BUILD:TYPE=DUAL_I` or `BUILD:TYPE=DUAL_R`. Default: 6000.
Range 2 verification interval	⊕	E2	Sets the minimum verification interval of range 2. Only used if `BUILD:TYPE=DUAL_I` or `BUILD:TYPE=DUAL_R`. Options are: • 1(def) • 2 • 5 • 10 • 20 • 50 • 100
Additive tare limit	⊕	AD.TARE	Sets the additive tare limit. Default: 0. See Section 7.2.6 for more details.
Weighed units	⊕	UNITS	Sets the weighing units for the scale. Options are: • NONE • G • KG (default) • LB • T • OZ • USER
A/D synchronisation frequency (effects calibration)	⊕	SYNC	Sets the frequency of the primary anti-noise filter in Hz. This sets the rate that data is processed. It will deliver optimum noise performance in a 50Hz environment when set to 12.5, 25 or 50. Options are: 10, 12.5, 15, 20, 25, 30, 50(def), 60, 100
High resolution display	⊕	HI.RES	In full setup, this setting enables x10 high resolution display. Weights will be shown with 10x resolution. The options are: • OFF(def) • ON In safe setup, this setting will display the weight in x10 mode only while the menu is active. Upon exit the x10 mode is disabled. In trade mode this test is only available for 5 seconds.

⊕ = Change only possible in FULL Setup

5.3 OPTION: Scale Options Menu

Items within this group set various weighing options for the scale. Changes of some items will affect certification.

Items		Name	Description
Scale trade use	⊕	USE	Set whether the scale is for trade use. Options are: • OIML: Scale operation is not restricted by OIML or NTEP • INDUST: Scale operation will comply with OIML (default) • NTEP: Scale operation will comply with NTEP
Weight averaging	⊕	FILTER	Sets the number of weight readings to average before the weight is used. Options are: • 1 • 2 • 3 • 4 • 5 • 6 • 7 • 8 • 9 • 10(def) • 25 • 50 • 75 • 100 • 200
Anti-jitter average reset	⊕	JITTER	Sets the reset threshold for the anti-jitter filter. Options are: • OFF: Anti-jitter average disabled. • FINE(def): Small changes will reset the filter. • COARSE: Larger changes are required to reset the filter.
Motion Detection	⊕	MOTION	Sets how trigger level for motion detection. Set in divisions per time period. For example, 0.5-1.0 means that more than 0.5 divisions in 1 second is considered motion. Options are: • NONE: Disable motion detection • 0.5-1.0(def) • 1.0-1.0 • 2.0-1.0 • 5.0-1.0 • 0.5-0.5 • 1.0-0.5 • 2.0-0.5 • 5.0-0.5 • 0.5-0.2 • 1.0-0.2 • 2.0-0.2 • 5.0-0.2 • 3.0-1.0 • 3.0-0.5 • 3.0-0.2

Auto-zero on start-up	⊕	AUTO.Z	Enable and initial zero on start. The amount of weight that can be zeroed is limited to ±10% of scale capacity. Options are: • OFF(def) • ON
Zero tracking	⊕	Z.TRAC	Enable zero tracking. Set in divisions per time period. For example, 0.5-1.0 means that zero tracking will cancel no more than 0.5 divisions in 1 second whilst in the zero band. Options are: • NONE(def): Disable zero tracking • 0.5-1.0 • 1.0-1.0 • 2.0-1.0 • 5.0-1.0 • 0.5-0.5 • 1.0-0.5 • 2.0-0.5 • 5.0-0.5 • 0.5-0.2 • 1.0-0.2 • 2.0-0.2 • 5.0-0.2 • 3.0-1.0 • 3.0-0.5 • 3.0-0.2
Allowable range of zero setting	⊕	Z.RANGE	Range around 0 where a scale zero is possible. Options are: • 02-02(def): -2% to +2% of capacity • 01-03: -1% to +3% of capacity • 20-20: -20% to +20% of capacity • 100.100: -100% to +100% of capacity
Zero Dead Band	⊕	Z.BAND	Range around zero which is considered zero for zero tracking, multiranging, auto-printing, etc. Default: 0.
Sense line check		SENS.CH	Enable sense checking. Useful with EX applications, see Section 2.6 Options are: • OFF • ON(def)

⊕ = Change only possible in FULL Setup

5.4 CAL: Scale Calibration Menu

Items in this group are used for scale calibration. See Section 7.

Items		Name	Description
Zero calibration	⊕	ZERO	Perform a zero calibration
Span calibration	⊕	SPAN	Perform a span calibration
Linearity calibration	⊕	ED.LIN	Perform a linearity calibration
Clear linearity points	⊕	CLR.LIN	View/clear linearity points
Direct zero calibration	⊕	DIR.ZER	Perform a direct zero calibration
Direct span calibration	⊕	DIR.SPN	Perform a direct span calibration
Restore factory calibration	⊕	FAC.CAL	Set the calibration and BUILD menu settings to factory default values

⊕ = Change only possible in FULL Setup

5.5 SERIAL: Serial Communications Menu

Items within this group set the serial and printing outputs.

Items	Name	Description
Serial Ports	SER.M.1A SER.M.1B SER.M.2A SER.M.2B SER.SLV SER.HST	Options for module 1 port A Options for module 1 port B (transmit only) Options for module 2 port A Options for module 2 port B (transmit only) Options for USB slave serial port Options for USB host serial port
Function type	TYPE	Sets the function of the serial port. Options are: • OFF: Disable the port • AUTO.LO: Auto transmit at 10Hz • AUTO.HI: Auto transmit at the sync frequency • PRINT: Enable printing • SINGLE: Transmit once (see also Section 17.7) • NET: Rinstrum network protocol (see Sections 18 and 21) • MB.LGCY: 5000 Modbus ASCII legacy protocol (see Section 22) • MB.ASCI: Fully featured Modbus ASCII protocol (see Section 19) • MB.RTU: Fully featured Modbus RTU protocol (see Section 19) Availability: • SER.M.xA: OFF - MB.LGCY • SER.M.xB: OFF - SINGLE • SER.SLV: OFF - NET • SER.HST: OFF - MB.RTU
Baudrate	BAUD	Sets the baudrate. Options are: • 300 • 600 • 1200 • 2400 • 4800 • 9600 (default) • 19200 • 38400 • 57600 • 115200
Parity, data bits, stop bits, termination, RS232/RS485 and duplex	BITS	Sets port transmission options. Options for each position are: 1. N/O/E: No parity, odd parity, even parity 2. 8/7: Number of data bits 3. 1/2: Number of stop bits. 4. -/T: RS485 termination resistor enable 5. 2/4: RS232 or RS485 6. -/D: DTR handshake enable Default: N81-2- Note: Not all options apply to all ports.
TCP/IP Network Communication Port	TCP.IN.1	Options for incoming TCP/IP network communication port. Note that this port's type is always fixed to NET to allow for indicator configuration, regardless of the configuration of other ports.
TCP Port	PORT	Sets the TCP port used for the socket. Selectable from 1024 to 65535. Default: 2222.

Timeout	T.OUT	Sets the timeout period in seconds for the TCP port. A value of 0 will disable the timeout. When set, if the TCP output buffer does not empty within the timeout period, the connection will be forceably closed. On network ports, if no data is read from the TCP socket within the timeout period, the connection will be forceably closed. Selectable from 0 to 86400. Default: 0.
TCP/IP Automatic Output Port	TCP.IN.2	Options for incoming TCP/IP automatic output port
Function type	TYPE	Sets the function of the incoming TCP/IP automatic output port. See serial port types above for descriptions. Availability: AUTO.LO - AUTO.HI
TCP Port	PORT	Sets the TCP port used for the socket. Selectable from 1024 to 65535. Default: 2223.
Timeout	T.OUT	Sets the timeout period in seconds for the TCP port. A value of 0 will disable the timeout. When set, if the TCP output buffer does not empty within the timeout period, the connection will be forceably closed. On network ports, if no data is read from the TCP socket within the timeout period, the connection will be forceably closed. Selectable from 0 to 86400. Default: 0.
Outgoing UDP/IP Serial Ports	UDP.O.1 UDP.O.2	Options for outgoing UDP/IP port 1 Options for outgoing UDP/IP port 2
Function type	TYPE	Sets the function of the UDP port. Options are: • OFF: Disable the port (default) • AUTO.LO: Auto transmit at 10Hz • AUTO.HI: Auto transmit at the sync frequency • PRINT: Enable printing • SINGLE: Transmit once (see also Section 17.7) • NET: Rinstrum network protocol (see Sections 18 and 21) • MB.LGCY: 5000 Modbus ASCII legacy protocol (see Section 22)
Destination IP	DST.IP	Set the destination IP address from which to accept commands and to which data should be sent. Displayed as `nnn.nnn.nnn.nnn` (eg `192.168.1.254`). Each octet has the range of 0 to 255. Default is `0.0.0.0` (disabled).
Destination UDP port	DST.PRT	Sets the destination UDP port from which to accept commands and to which data should be sent. Selectable from 0 to 65535. Default is `0` (disabled).
Incoming UDP/IP Serial Ports	UDP.IN.1 UDP.IN.2	Options for incoming UDP/IP port 1 Options for incoming UDP/IP port 2
Function type	TYPE	Sets the function of the UDP port. Options are: • OFF: Disable the port (default) • AUTO.LO: Auto transmit at 10Hz • AUTO.HI: Auto transmit at the sync frequency • PRINT: Enable printing • SINGLE: Transmit once (see also Section 17.7) • NET: Rinstrum network protocol (see Sections 18 and 21) • MB.LGCY: 5000 Modbus ASCII legacy protocol (see Section 22)

UDP listen port	PORT	Sets the port on which the indicator listens for commands. Replies are sent to the last IP:port that a command was received from. On non-network ports (AUTO.LO, etc), the receving end should send an empty UDP packet to the indicator to initiate sending. Selectable from 0 to 65535. Default is `0` (disabled).
Network options	NET.OPT	Options for serial networking
Serial Address	ADDRESS	The serial network address. Selectable from 0 to 31. Default: 31
Auto Transmit Options	AUT.OPT	This is a list of all items concerned with automatic and single serial transmission of weight data.
Auto output	TYPE	The auto transmit format (see Section 8). Options are: • AUTO.A: Format A (default) • AUTO.B: Format B • AUTO.C: Format C • AUTO.D: Format D • CUSTOM: Custom format as set in `SERIAL:AUT.OPT:AUT.FMT` • AUTO.F: Format F
Auto Output Source	SOURCE	The weight source for the output data. Options are: • DISP(def): Displayed weight • GROSS: Gross weight • NET: Net weight • TOTAL: Total weight
Custom format	AUT.FMT	Custom format (see Section 8.2.6 for more information).
Start Character	ST.CHR	Sets the ASCII character sent at the start of the automatic message. If set to 0, no character is sent. Default: 2 (STX)
End Character 1	END.CH1	Sets the 1st of 2 ASCII characters sent at the end of the automatic message. If set to 0, no character is sent. Default: 3 (ETX)
End Character 2	END.CH2	Sets the 2nd of 2 ASCII characters sent at the end of the automatic message. If set to 0, no character is sent. Default: 0 (none)
Printing Options	PRN.OPT	This is a list of all items concerned with printing.
Printout Type	PRNT.TP	Set the printout type (see Section 9.1). Options are: • SINGLE(def): Print a single line with no extra line feed • DOUBLE: Print single line output but double spaced • TICKET: Print full weight ticket • A.SING: Auto print a single line with no extra line feed • A.DOUB: Auto print single line output but double spaced • A.TICK: Auto print full weight ticket • TOTAL: Totalising printer mode • A.TOTAL: Automatic Totalising printer mode
Ticket Header	HEADER	Ticket header (see Section 9.2).
Ticket Format	TIC.FMT	Ticket format (see Section 9.3).
Space Columns and rows for print	SPACE	Position of the printout. The 1st parameter sets the horizontal spacing. The 2nd parameter sets the vertical spacing (see Section 21.3.40). Default: 00.00
Print Interlock	I.LOCK	Print interlock type. Options are: • NONE: No interlock (default) • MOTION: Motion must occur between printouts • RET.Z: Scale must return to the zero band between printouts
Modbus Options	MOD.OPT	This is a list of all items concerned with Modbus.
Modbus TCP Port	PORT	Set the TCP port that listens for Modbus connections. Default: 502

32bit Endian	ENDIAN	This sets the endianness for 32bit Modbus registers. The Modbus specification does not specify endian for values larger than 16bits. This setting does not swap the endian for 16bit values. These are fixed. Options are: • BIG(def): Big endian, the Modbus defacto standard. Low register = data high 16bits, high register = data low 16bits. • LITTLE: Little endian. Low register = data low 16bits, high register = data high 16bits.

5.6 ETH.NET: Ethernet Menu

Items within this group set the Ethernet and IP parameters.

Items	Name	Description
DHCP enable	DHCP	Set whether IP setup is obtained via DHCP. Options are: • OFF • ON (default)
IP address	IP	View or set the IP address. Displayed as `nnn.nnn.nnn.nnn` (eg `192.168.1.254`). Each octet has the range of 0 to 255. Read only when DHCP is enabled, displays the DHCP assigned address. Default is `192.168.1.254`.
Subnet mask	MASK	View or set the subnet mask. Displayed as `nnn.nnn.nnn.nnn` (eg `255.255.255.000`). Each octet has the range of 0 to 255. Read only when DHCP is enabled, displays the DHCP assigned mask. Default is `255.255.255.0`.
Gateway address	GATE.W	View or set the default gateway address. Displayed as `nnn.nnn.nnn.nnn` (eg `192.168.1.1`). Each octet has the range of 0 to 255. Read only when DHCP is enabled, displays the DHCP assigned address. Default is `192.168.1.1`.
DNS servers	DNS.1, DNS.2, DNS.3	View or set the DNS server addresses. Displayed as `nnn.nnn.nnn.nnn` (eg `192.168.1.1`). Each octet has the range of 0 to 255. Read only when DHCP is enabled, displays the DHCP assigned address. Defaults are `192.168.1.1`, `0.0.0.0`, `0.0.0.0`.
Host name	HST.NAM	The indicator hostname. Defaults to <`model`>-<`serial no`>, e.g. `c520-123456`.
Search domain	SEARCH	Search domain used in addition to domain name. Read only when DHCP is enabled, displays the DHCP assigned search domain. Default is “”.
MAC address	MAC	Ethernet MAC address in hexadecimal. Read only.

5.7 SPEC: Special Settings Menu

Items within this Group set the security codes, key locking and special modes.

Items		Name	Description
Safe setup passcode	⊕	SAFE.PC	Passcode required to enter safe setup. Set to 0 to allow free access. Default is 0.
Full setup passcode	⊕	FULL.PC	Passcode required to enter full setup. Set to 0 to allow free access. Default is 0.
Rear entry mode	⊕	R.ENTRY	Rear entry mode for full setup. This option can only be changed when the rear key is used to enter the menus. Options are: • OFF(def): Full setup is available via the Gross/net key or rear key. • ON: Full setup access is only available via the rear key.
Key enable	⊕	BUTTON	Enable each front key (in the same order as the keys). Options are: • Y(def): enabled • N: disabled • I: immediate operation (dont wait for motion, not valid for trade) Immediate operation is not available on the gross/net key.
Function Key 1 Function Key 2 Function Key 3		KEY.FN KEY.FN.1 KEY.FN.2 KEY.FN.3	Set the function of the front function keys (see Section 17). Options are: • NONE • PRINT • SHW.TOT • CLR.TOT • UNDO • SIN.M1A • SIN.M1B • M.HOLD • P.HOLD • L.HOLD • SIN.M2A • SIN.M2B • UNITS • SIN.SLV • SIN.HST Defaults: KEY.FN.1 = PRINT, KEY.FN.2 = NONE, KEY.FN.3 = NONE

Operation of the external inputs Remote Key 1 Remote Key 2 Remote Key 3 Remote Key 4 Remote Key 5 Remote Key 6 Remote Key 7 Remote Key 8		IN.FN IN.FN.1 IN.FN.2 IN.FN.3 IN.FN.4 IN.FN.5 IN.FN.6 IN.FN.7 IN.FN.8	Set the function of the remote inputs (see Section 17). Options are: • NONE • ZERO • TARE • GROSS • PRINT • BLANK • LOCK • SHW.TOT • CLR.TOT • UNDO • SIN.M1A • SIN.M1B • M.HOLD • P.HOLD • L.HOLD • TXT.PRN • SIN.M2A • SIN.M2B • UNITS • SIN.SLV • SIN.HST • SP.RST.A • SP.RST.B Defaults: All inputs = NONE
Buzzer Enable		BUZZER	Enables and disables the buzzer operation. Options are: • OFF • ON (default)
Display Brightness		BRIGHT	Set the display brightness (0..100%). Default: 100%
Secondary Units		UNIT.2	Set the secondary units. Options are: • OFF (default) • G • KG • LB • OZ • CUSTOM
Tertiary Units		UNIT.3	Set the tertiary units. Options are: • OFF (default) • G • KG • LB • OZ
Custom Unit Factor		FACTOR	Set the conversion factor for custom secondary units (0.001..999.999) Default is 1.000.
LB-OZ display		LB.OZ	Display oz weights as lb-oz where possible. Options are: • OFF (default) • ON

⊕ = Change only possible in FULL Setup

5.8 TEST: Test Menu

Items within this group are used for testing.

Items		Name	Description
mV/V test mode		SCALE	Show the loadcell signal in mV/V. In trade mode this test is only available for 5 seconds.
Test IO: 1 to 4 Test IO: 5 to 8		IO-1.4 IO-5.8	Test IO by checking the value of inputs and setting outputs on/off. If no IO is attached to the indicator, "NO I.O." error will be displayed. The input status is: • 1..8: IO is on • -: IO off The output status is: • 1. to 8.: output is on • -.: output is off (despite being turned on) To drive an output, use the left and right keys to select the desired IO, and up and down keys to toggle the state of the key. The decimal point indicates the IO is being driven as an output. Press OK or cancel to exit the editor.
Overload Counter	⊕	O-LOAD	Shows the number of times that the instrument has been overloaded to above 135% of fullscale. Use the OK key to clear the overload count in full setup.

⊕ = Change only possible in FULL Setup

5.9 SET.PTS: Setpoints Menu

This section is used to set all of the operational logic of the setpoint system, as well as provide for the entry of secure setpoint target and inflight values.

Items	Name	Description
General	GEN	General setpoint settings
Jog on time	JOG ON	Job on time in milliseconds. Only used for pulse setpoint timing.
Jog off time	JOG.OFF	Job off time in milliseconds. Only used for pulse setpoint timing.
Jog set	JOG.SET	Number of jogs in a set. Only used for pulse setpoint timing.
Maximum sets	MAX.SET	Maximum number of jog sets. Only used for pulse setpoint timing.
Feeder	FEEDER	Feeder type for weigh in and weigh out setpoints. Options are: • MULT.: Outputs will run in parallel (default) • SINGLE: Outputs will run serially
Delay check	DLY.CHK	Delay time in milliseconds after weigh in or weigh out setpoint becomes active before weight is checked again.
Setpoint 1 Setpoint 2 Setpoint 3 Setpoint 4 Setpoint 5 Setpoint 6 Setpoint 7 Setpoint 8	SET.PT.1 SET.PT.2 SET.PT.3 SET.PT.4 SET.PT.5 SET.PT.6 SET.PT.7 SET.PT.8	Setpoint settings
Setpoint type	TYPE	Set setpoint type. Options are: • OFF: Disabled (default) • ON: Always active • OVER: Weight over setpoint • UNDER: Weight under setpoint • COZ: Centre of zero status • ZERO: Zero band status • NET: Gross/net status • MOTION: Motion status • ERROR: Error status • BUZZER: Buzzer status • W. OUT: Weigh out setpoint • W. IN: Weigh in setpoint
Target	TARGET	Setpoint target. Only used for weight setpoints.
Hysteresis	HYS	Setpoint hysteresis. Only used for weight setpoints.
Inflight	FLIGHT	Setpoint inflight. Only used for weight setpoints.
Lock	LOCK	Disable operator access (via operator menus). Options are: • OFF(def): menu access available • ON: menu access locked
Source	SOURCE	Select source for weight setpoints. Options are: • GROSS(def): use gross weight. • NET: use net weight.
Output logic	LOGIC	Output logic. Options are: • HIGH(def): output is high when setpoint is active. • LOW: output is low when setpoint is active.

Alarm	ALARM	Setpoint alarm. Options are: • OFF(def): no alarm. • SINGLE: Single beep alarm. • DOUBLE: Double beep alarm. • CONT.: Continuous beeping alarm. • FLASH: Flash display alarm.
Output timing	TIMING	Output timing. Options are: • LEVEL: Output is active when setpoint is active and reset is not active (default) • EDGE: Output is active when setpoint is active • LATCH: will stay inactive after reset until the next setpoint transition from inactive to active • PULSE: Setpoint will stay active once activated until the reset becomes active
Reset input	RESET	Reset input. Options are: • NONE: No reset input (default) • SP.RST.A: Setpoint reset A • SP.RST.B: Setpoint reset B

5.10 ANALOG: Analog Output Menu

Items within this menu set the options for the optional analog output module.

Items	Name	Description
Source	SOURCE	Select the reading source. Options are: • GROSS: Gross weight • NET: Net weight • DISP(def): Displayed weight (gross or net) • COMMS: Use value written to comms register.
Type	TYPE	Select either voltage or current output. Options are: • CURR(def): 4-20mA output • VOLT: 0V-10V voltage output
Absolute value	ABS	Select absolute value output. Options are: • OFF (def) • ON
Clip ouput value	CLIP	Select if output value is clipped. Options are: • OFF (def) • ON
Calibrate Zero output	CAL.LO	Calibrate analog output at 0 weight (either 4 mA or 0V). The analog output cards are factory calibrated but this allows for any fine adjustment on site.
Calibrate Fullscale output	CAL.HI	Calibrate analog output at fullscale weight (either 4 mA or 0V). The analog output cards are factory calibrated but this allows for any fine adjustment on site.
Test Analog Output	FRC.ANL	Test analogue output. Set the analogue output low (0V or 4mA) or high (10V or 20mA).

5.11 CLOCK: Clock Menu

This menu is used to set the clock/calendar and QA options.

Items		Name	Description
Time set		TIME	Set the time in the format HH.MM, where: • HH: Hours (00 - 23) • MM: Minutes (00 - 59)
Date set		DATE	Set the current date in EU format (DD.MM.YY), where: • DD: Day (01 - 31) • MM: Month (01 - 12) • YY: Year (1997 - 2200)
QA enable	⊕	QA.OPT	Enable QA calibration due check. Options are: • OFF(def) • ON
QA date	⊕	QA.DATE	Set the QA check date. Set in EU format (DD.MM.YYYY), where: • DD: Day (01 - 31) • MM: Month (01 - 12) • YY: Year (1997 - 2200)
Time format		T.FMT	Set the time format for printing. Options are: • 24 HR (default) • 12 HR
Date format		D.FMT	Set the date format for printing. Options are: • DD.MM.Y4 (default) • MM.DD.Y4 • Y4.MM.DD • DD.MM.Y2 • MM.DD.Y2 • Y2.MM.DD

⊕ = Change only possible in FULL Setup

5.12 FILE: File Menu

This group is used for saving and loading files to and from a USB disk attached to the indicator. The indicator only supports USB disks formatted with the FAT32 file system.

Items		Name	Description
Upgrade firmware	⊕	FW.UPD	Upgrade the firmware from a USB mass storage device. The upgrade file must be in the root directory of the disk, and have an extension of .rpk. There must only be one .rpk file on the disk, as there is no ability to select which file to install. The menu will prompt to continue before installing the firmware. The following errors may be displayed: • "NO DISK": No USB disk is present, or formatted in an unsupported filesystem • "NO RPK": There are no .rpk files in the root directory of the USB disk • "MANY RPK": There is more than one .rpk file in the root directory
Save settings to USB disk		SAV.SET	Save settings to a USB mass storage device. The file will be saved in the root directory and be called <model>_<serial number>_<date>_<time>_settings.rdb. For example: C520_3382100_20120608_145951_settings.rdb. See Section 5.15 `FACTRY:PRN.CFG` for printing this information to an attached printer.
Save change log to USB disk		SAV.CL	Save the change log file to a USB mass storage device. The file will be saved in the root directory and be named as follows: <model>_<serial number>_<date>_<time>_changelog.csv. For example: C520_3382100_20120608_145951_changelog.csv
Save DSD data to USB disk		SAV.DSD	Export DSD data to a USB mass storage device. The file will be saved in the root directory and be named as follows: <model>_<serial number>_<date>_<time>_dsd.csv. For example: C520_3382100_20120609_011556_dsd.csv
Save debug information to USB disk	⊕	SAV.DBG	Export debug information to disk. This file provides useful information to Rinstrum when trying to solve installation and setup problems. The file will be saved in the root directory and be named as follows: <model>_<serial number>_<date>_<time>_debug_info.txt. For example: C520_3382100_20120609_011556_debug_info.txt
Eject USB disk		EJECT	Eject the USB disk.

⊕ = Change only possible in FULL Setup

5.13 DSD: Digital Storage Device Menu

This group is used for digital storage device options.

Items		Name	Description
DSD Enable	⊕	ENABLE	Enables and disables the DSD. Options are: • OFF(def): Traceable weights are not stored in the DSD • ON: Traceable weights are stored in the DSD **Note** After enabling the DSD for the first time, it is necessary to initialise the DSD before use using the `DSD:INIT` menu.
Initialise DSD	⊕	INIT	Initialise the DSD. This option will create or re-create the DSD. All records will be lost. The user will be prompted to continue before this action will occur.
DSD Status		STATUS	Displays the current number of records in the DSD, and the total capacity of the DSD.
Auto purge	⊕	A.PURGE	Enables and disables automatic purging of the DSD. When the DSD is purged, 10% of the records are removed. Options are: • OFF(def): During printing, if the DSD becomes full, the indicator will prompt the user to purge the DSD • ON: During printing, if the DSD becomes full, the indicator will purge the DSD without user interaction
Manual Purge		PURGE	Manually purge 10% of the DSD records.

⊕ = Change only possible in FULL Setup

5.14 CHG.LOG: Changelog Menu

This group is used for change log options.

Items		Name	Description
Changelog Status		STATUS	Displays the percent full of the change log.
Clear Changelog	⊕	CLEAR	Clear the changelog. All records will be lost. An entry will be made in the changelog to indicate it has been cleared.

⊕ = Change only possible in FULL Setup

5.15 FACTRY: Factory menu

This group is used for factory options.

Items		Name	Description
Restore Factory default	⊕	DEFLT	Restores settings to factory defaults (except calibration settings).
Printout of the indicator settings		PRN.CFG	Print all settings to the connected printer. See Section 5.12 `FILE:SAV.SET` for saving this information to a disk.

⊕ = Change only possible in FULL Setup

6 Operator Menus

The operator menus provide access to some settings typically used by operators. These do not require access to safe or full setup.

Items	Name	Description
Setpoint Targets	TARGET	The target menu displays the sepoint targets for active setpoints only. Inactive and other setpoint types are hidden.
Setpoint 1 Target	TARG1	
Setpoint 2 Target	TARG2	
Setpoint 3 Target	TARG3	
Setpoint 4 Target	TARG4	
Setpoint 5 Target	TARG5	
Setpoint 6 Target	TARG6	
Setpoint 7 Target	TARG7	
Setpoint 8 Target	TARG8	
Setpoint Inflights	FLIGHT	The inflight menu displays the sepoint inflight values for active setpoints only. Inactive and other setpoint types are hidden.
Setpoint 1 Inflight	FLT 1	
Setpoint 2 Inflight	FLT 2	
Setpoint 3 Inflight	FLT 3	
Setpoint 4 Inflight	FLT 4	
Setpoint 5 Inflight	FLT 5	
Setpoint 6 Inflight	FLT 6	
Setpoint 7 Inflight	FLT 7	
Setpoint 8 Inflight	FLT 8	
USB	USB	The USB menu allows you to quickly eject a USB disk. This menu is only available when a USB disk is mounted.
Eject USB disk	EJECT	Eject the USB disk.
Current IP Settings	IP.INFO	This menu allows you to quickly view the IP settings of the indicator. These settings can only be changed from the Ethernet setup menu. See Section 5.6.
DHCP enable	DHCP	View whether IP setup is obtained via DHCP.
IP address	IP	View the currenly assigned IP address.
Subnet mask	MASK	View the currently assigned subnet mask.
Gateway address	GATE.W	View the currently assigned default gateway address.
DNS address	DNS	View the currently assigned DNS server address.
Host name	HST.NAM	View the indicator hostname.
MAC address	MAC	View the indicator Ethernet MAC address in hexadecimal.
Module info	MODULE	This menu allows you to view module status and information.
Model type	SLOT n:TYPE	The type of module fitted to this slot.
Serial number	SLOT n:SER.NO	The serial number of the module fitted to this slot.
Software version	SLOT n:SW.VER	The software version of the module fitted to this slot.
Exit	-END-	Exit the operator menus

7 Calibration and Trade

7.1 Introduction and Warnings

The calibration of the C500 series indicators are fully digital. The calibration results are stored in permanent memory for use each time the unit is powered up.

Some of the menu settings affect calibration. The BUILD settings must be set before calibration. Changing these settings after calibration may alter the calibration.

All calibration operations are in the CAL menu. Calibration routines are available in full setup only (not safe setup).

The calibration program will automatically prevent the indicator from being calibrated into an application outside its specification. If an attempt is made to calibrate the indicator outside of the permitted range, an error message will show and the routine will be abandoned. Refer to Section 21.

The indicator has a wide-range amplifier section. The non-trade calibration range of the instrument extends well beyond the trade approved range. It should not be assumed that just because the indicator has successfully calibrated a scale, that the scale is correct for trade use. Always check the scale build against the approval specification.

7.2 General Information

7.2.1 Terminology

A knowledge of basic weighing terms is useful in setting up and calibrating the indicator. These terms are used throughout the setting procedure and are defined as follows:

- Weighing range: This is the range of weights which can be measured. The indicator can be setup with 1 or 2 ranges.
- Division / count-by: The smallest unit of weight change which is displayed. This will be different for each range. This is setup in `BUILD:E1` and `BUILD:E2`.
- Fullscale / full capacity: The maximum amount of weight used on the scale. This will be different for each range. This is setup in `SCALE:CAP1` and `SCALE:CAP2`.
- Number of divisions: The number of divisions between 0 and full capacity. It is equal to full capacity / count-by. This might be different for each range. For example, 20000kg / 10kg = 2000 divisions.
- Units: Describes the unit of measurement used for the weight (kg, t, lb, etc).
- Loadcell signal at full capacity: This is the loadcell signal at full capacity.
- Signal resolution: This is the change in loadcell signal for each division.

Example:

- A 10000kg, 2.0mV/V loadcell is used in an application with a 5000kg capacity, displaying in 5kg steps.
- The values of each of the above terms is:
  - Units = kg
  - Capacity = 5000kg
  - Count-by = 5kg
- Number of divisions = 1000 divisions

- The loadcell signal at full capacity = (5000kg / 10000kg) x 2.0mV/V = 1.0mV/V
- The signal at capacity = 5V excitation * 1.0mV/V = 5mV
- The signal resolution = 5mV / 1000 divisions = 5 μV/division

7.2.2 Dual Interval and Dual Range Operation

The indicator provides single range, dual interval and dual range modes. In non-trade operation, up to 100,000 divisions are available so it is rare for the precision of the displayed reading to be a problem. However in trade applications where the number of divisions that can be legally displayed is limited, the use of dual interval or dual range operation allows greater precision in the displayed readings without exceeding the maximum number of graduations available in the certification of the loadcell.

Both of these modes of operation allow for the indicator to operate with 2 count-by settings so that it is possible to weigh for example up to 2 kg in 1g increments and then up to 5kg in 2g increments.

Dual interval and dual range are identical in many respects and can be treated the same for the purposes of setup and calibration. The difference in the two comes about in the operation of the scale. With dual range operation the range is determined based on the gross weight. Once the scale changes from low range to high range it will not change back to low range until the scale is returned to a stable zero reading. Dual interval operation however is based on the net weight and no restrictions are placed on the change from the high interval to the low interval. With dual interval operation it is therefore possible to weigh in the low interval with high tare weights.

The indicator is equally accurate in either mode but due to hysteresis effects with many loadcells it may not be possible to operate accurately in dual interval mode. In these cases dual range mode ensures that the weight readings taken from the loadcell are accurately displayed during loading and unloading operations.

7.2.3 Direct mV/V Operation

It is possible to calibrate the indicator without test weights if the output capacity of the loadcell is known. For applications like silo weighing etc where it is impractical to use test weights this mode of operation allows the mV/V signal strength at no load, as well as the mV/V signal strength of the span to be entered directly. This type of calibration is only as accurate as the loadcell output figures but for many applications this is more than adequate. For more details see Section 7.6.

7.2.4 Maintenance Date

The indicator is able to indicate when it is due for recalibration or regular maintenance. This is set in `CLOCK:QA.OPT` and `CLOCK:QA.DATE`. When the due date is reached, the indicator displays (CAL) (DUE). Pressing any key will hide this message temporarily. This message can only be disabled by changing the QA settings (which requires full setup).

7.2.5 Filtering Techniques

The indicator has a number of advanced filtering options which allow it to be optimized to produce the most accurate readings possible in the shortest time. There is a trade-off between noise reduction and step response time.

Note

The step response is the time between placing a weight on the scale and the correct weight reading being displayed.

FIR Filter:

- The first level of filtering provided is a FIR filter that is linked to the measurement rate. The measurement rate is set in `BUILD:SYNC`. This filter is a very high performance 'tuned' filter that provides up to 180 dB of attenuation at multiples of the SYNC frequency and broad band filtering of between 40 and 80 dB generally. For example, setting the SYNC frequency at 25Hz would provide 180 dB of noise rejection at 25, 50, 75 ... Hz.

- The primary noise source is the mains power so the SYNC is usually set to half the mains frequency or the mains frequency. For example, with 50Hz mains power, use 12.5, 25 or 50Hz SYNC frequency. The FIR filter introduces a delay of 3 samples to the step response. So for a SYNC frequency of 50Hz (i.e. readings every 20 milliseconds) there is a delay of 60 milliseconds between a weight change and the final weight reading (before averaging is applied).

Digital Averaging:

- In addition to the FIR filter the indicator has two levels of digital averaging.

- The first is a fixed length sliding window average where the average of the last 'n' readings is calculated. As each new reading is taken the oldest reading is discarded and a new average calculated. The length of the window is set in `OPTION:FILTR` from 1 reading to 200 readings. Increasing the average will increase the step response. For example an average of 10 readings with a SYNC frequency of 50Hz gives a step response = (10 + 3) / 50Hz = 0.26 seconds.

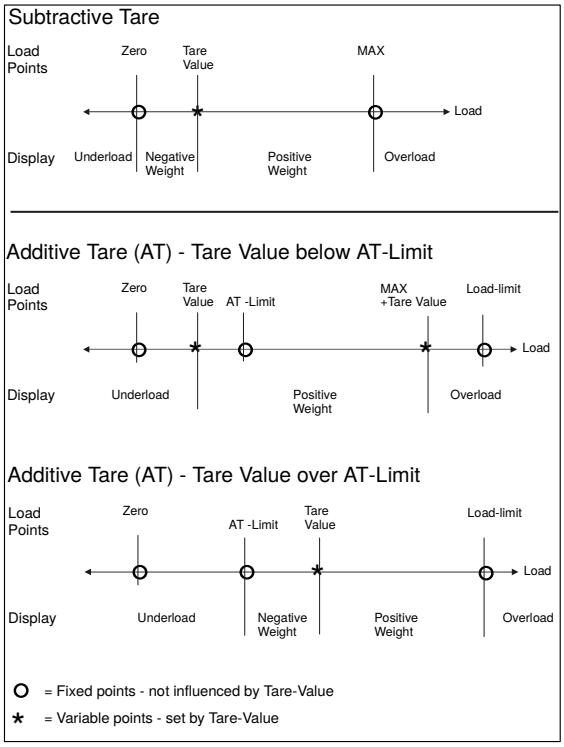
- The second average is similar to the first but has a variable length which grows from 1 reading up to a maximum of 10 readings. If a disturbance on the scale is detected the old readings are discarded and a new average starts all over again. Each of the 10 readings is calculated over the window length of the fixed average. In this way very long term averages are calculated without causing any delays. The amount of fluctuation that causes the average to be restarted can be selected as FINE or COARSE in `OPTION:JITTER`. The COARSE setting is more tolerant of weight change than the FINE.

7.2.6 Subtractive and Additive Tare

The indicator can operate subtractive and additive tare:

- Subtractive tare: The weight tared is subtracted from the total weight that can be measured. For example, if the scale capacity is 100kg and the tare is 40kg, then the scale can only measure a maximum 60kg NET.

- Additive tare: The weight tared is not subtracted from the total weight that can be measured. For example, if the scale capacity is 100kg and the tare is 40kg, then the scale can still measure to a maximum 100kg NET. The maximum amount of additive tare is setup in `BUILD:AD.TARE`.



Warning!

When using additive tare, ensure that the scale is rated to the additional load.

7.2.7 Trade versus Industrial Mode

The indicator may be operated in trade or industrial mode. The differences in operation are:

Operation	Trade mode	Industrial mode
Underload	Below zero range setting	-105% of fullscale
Overload	Fullscale + 9 divisions	105% of fullscale
Tare	Tare values must be > 0	No restriction
Preset tare	Available via tare key long press	Available via tare key long press
De-zero	Not available	Available via zero key long press
Test modes	Limited to 5 seconds	Unlimited time
Hold, peak, livestock	Not available	Available

7.2.8 Setup Counter

Within the setup program there are a number of parameters that can affect trade performance. If any of these steps are altered, the trade certification of the scale could be voided. These parameters are automatically blocked in safe setup.

There is a built-in setup counter to monitor the number of times the trade parameters are altered. The value of this counter is stored within the unit, and can only be reset at the factory. Each time a trade parameter is altered, the counter will increase by one. Whenever the indicator is powered up or setup menus entered/exited, the current value in the counter is displayed.

7.2.9 Security

See Section 5.1.2.

7.3 CAL:ZERO Zero Calibration Routine

1. Go to `CAL:ZERO` in the setup menus: Enter full setup (see Section 5.1.2); press the zero key until CAL is shown; press the tare key to show the `CAL:ZERO` setting (displays ZERO).
2. Press the OK key. The display will show the current weight. Remove all weight from the scale.
3. Press the OK key to start the zero calibration. The display will show (Z IN P) to show that calibration is in progress. When the process is complete the display will show the weight to allow the zero to be checked.
4. Press the cancel key to exit zero calibration and return to the menu.

7.4 CAL:SPAN Span Calibration Routine

1. A zero calibration should always be performed before a span calibration.
2. Go to `CAL:SPAN` in the setup menus: Enter full setup (see Section 5.1.2); press the zero key until CAL is shown; press the tare key until the `CAL:SPAN` setting is shown (displays SPAN).
3. Press the OK key. The display will show the current weight. Add the test weight to the scale.
4. Press the OK key to enter the actual weight on the scale. Use the left/right/up/down keys to edit the weight if required.
5. Press the OK key to start the span calibration. The display will show (S IN P) to show that calibration is in progress. When the process is complete the display will show the weight to allow the span to be checked.
6. Press the cancel key to exit span calibration and return to the menu.

7.5 Linearisation

Up to 10 linearisation points can be set independently anywhere in the operating range of the scale. Unused points may be cleared.

Warning!
Linearisation changes the signal resolution. If this is close to the limit, it should be checked.

7.5.1 CAL:ED.LIN Add Linearity Point

1. A zero and span calibration must be done before linearisation. Zero and span are assumed to be accurately set and have no linearisation error.
2. Go to `CAL:ED.LIN` in the setup menus.
3. Press the gross/net key. The list of linearisation points will be shown.
4. Select a linearity point. Press the gross/net key until the correct linearity point is shown, then press the OK key. The display will show the current weight. Add the test weight to the scale.
5. Press the OK key to enter the actual weight on the scale. Use the left/right/up/down keys to edit the weight if required.

6. Press the OK key to start the calibration. The display will show (L IN P) to show that calibration is in progress. When the process is complete the display will show the weight to allow the calibration to be checked.

7. Press the cancel key to exit calibration and return to the menu.

7.5.2 CAL:CLR.LIN Clear Linearity Point

1. Go to `CAL:CLR.LIN` in the setup menus.

2. Press the gross/net key to view the linearity points. The percentage of fullscale where the point was entered is shown.

3. Press the gross/net key until the unwanted linearity point is shown.

4. Press the OK key to clear the linearity point. The indicator will prompt with "CONT N". Use the up and down keys to select "Y", followed by the OK key to clear the linearization point and return to the menu. To exit without clearing, select "N".

7.6 Direct mV/V Calibration

It is possible to enter the mV/V values of zero and span directly. The internal mV/V accuracy of the indicator has a tolerance of 0.1

Calibration procedure:

1. Go to `CAL:DIR.ZER` in the setup menus: enter full setup (see Section 5.1.2); press the zero key until CAL is shown; press the tare key to show the `CAL:DIR.ZER` setting (displays DIR.ZER).

2. Press the OK key. The display will show the current weight.

3. Press the OK key to enter the mV/V value of the scale with no load. Use the left/right/up/down keys to edit the mV/V value.

4. Press the OK key to enter the calibration. The indicator will perform the calibration and then display the weight to allow the calibration to be checked.

5. Press the cancel key to exit the direct zero calibration and return to the menu.

6. Press the tare key to show the `CAL:DIR.SPN` setting (displays DIR.SPN).

7. Press the OK key. The display will show the current weight.

8. Press the OK key to enter the mV/V value of span at fullscale. Use the left/right/up/down keys to edit the mV/V value.

9. Press the OK key to enter the calibration. The display will perform the calibration and then display the weight to allow the calibration to be checked.

10. Press the cancel key to exit span calibration and return to the menu.

7.7 CAL:FAC.CAL Factory Calibration

The calibration can be reset to the factory calibration using the `CAL:FAC.CAL` menu item.

8 Automatic Weight Output

8.1 Introduction

The automatic output is normally used to drive remote displays, dedicated computer connections or PLCs. The output generates a simple weight message at predefined intervals.

8.2 Formats

There are 6 automatic output formats, including one custom format. The start and end characters of the format strings can be set (even for the standard formats) in the `SERIAL:AUT.OPT` menu.

Basic format:

```
Start Format End1 End2
```

where:

- Start: Start character (set in `SERIAL:AUT.OPT:ST.CHR`)
- Format: Format string (set in `SERIAL:AUT.OPT:TYPE`)
- End1: 1st termination character (set in `SERIAL:AUT.OPT:END.CH1`)
- End2: 2nd termination character (set in `SERIAL:AUT.OPT:END.CH2`)

8.2.1 Format A

Format:

```
Sign Weight Status
```

where:

- Sign (1 character): SPACE or '-'
- Weight (7 characters): The weight in 7 characters with decimal point and leading spaces.
- Status (1 character) = 'G', 'N', 'U', 'O', 'M' or 'E' for Gross, Net, Underload, Overload, Motion and Error.

8.2.2 Format B

Format:

```
Status Sign Weight Units
```

where:

- Status (1 character) = 'G', 'N', 'U', 'O', 'M' or 'E' for Gross, Net, Underload, Overload, Motion and Error.
- Sign (1 character): SPACE or '-'
- Weight (7 characters): The weight in 7 characters with decimal point and leading spaces.
- Units (3 characters): Weight units with leading spaces, e.g. ' kg' or ' t'. When the weight is unstable ' ' is sent.

8.2.3 Format C

Format:

```
Sign Weight S1 S2 S3 S4 Units
```

where:

- Sign (1 character): SPACE or '-'
- Weight (7 characters): The weight in 7 characters with decimal point and leading spaces.
- S1 (1 character) = 'G', 'N', 'U', 'O' or 'E' for Gross, Net, Underload, Overload and Error.
- S2 (1 character) = 'M' if motion, otherwise SPACE.
- S3 (1 character) = 'Z' if centre-of-zero, otherwise SPACE.
- S4 (1 character) = If in single range mode '-', otherwise, '1' for range 1, and '2' for range 2.
- Units (3 characters): Weight units with leading spaces, e.g. ' kg' or ' t'.

8.2.4 Format D

Format:

```
Sign Weight
```

where:

- Sign (1 character): SPACE or '-'
- Weight (7 characters): The weight in 7 characters with decimal point and leading spaces.

8.2.5 Format F

Format:

```
Sign Weight Units S1 S2
```

where:

- Sign (1 character): SPACE or '-'
- Weight (7 characters): The weight in 7 characters with decimal point and leading spaces.
- Units (1 character) = 'G', 'K', 'L' or 'T' for units of Grams, Kilograms, Pounds, Tonnes, otherwise SPACE.
- S1 (1 character) = 'G' or 'N' for Gross and Net.
- S2 (1 character) = 'O', 'I' or 'M' for Overload/underload, Error, Motion, otherwise SPACE.

8.2.6 Custom Format

Custom formats are specified in `SERIAL:AUT.OPT:AUT.FMT`. The format is entered character by character in ASCII codes and format tokens:

- ASCII codes: See the ASCII table in Section 9.5.
- Format tokens: See the codes in Section 8.3.

Up to 50 characters can be entered. If ASCII 0 needs to be sent, enter the token 128.

For example, the following format string would transmit the weight reading in a fixed 7 character field with leading zero suppression and no decimal point:

```
172 184 188 200 000
```

8.3 Custom Format Tokens

8.3.1 Formatting

Token	Description	Default
170	5 character weight	
171	6 character weight	
172	7 character weight	
173	8 character weight	✓
174	9 character weight	
179	No fixed length field for weight data	
180	No sign character sent	
181	Sign character send as ‘ ’ for positive and ‘-’ for negative	✓
182	Sign character send as ‘+’ for positive and ‘-’ for negative	
183	Sign character send as ‘0’ for positive and ‘-’ for negative	
184	No decimal point sent	
185	Decimal point sent as ‘.’	✓
186	Decimal point sent as ‘,’	
187	Weight sent with leading zeros, eg ‘000123’	
188	Weight sent without leading zeros eg ‘ 123’	✓
189	Weight readings sent regardless of overload or error status	✓
190	Weight data blanked on error	
191	Weight data send as ‘—’ on error	
192	Status characters are upper case	✓
193	Status characters are lower case	

8.3.2 Weight

Token	Description
200	Selected Weight (SRC)
201	Displayed Weight
202	Gross Weight
203	Net Weight
204	Tare Weight
205	Total

8.3.3 Status

Token	Options	Description
210	‘kg’, ‘lb’, ‘ t’, ‘ g’	Weight units
211	G, N, E, O, U, M	Standard HBM status
212	G, N, E, O, U	HBM status without motion
213	G, N	Gross/net status

214	M, ‘ ’	Motion status
215	M, S	Motion/Stable status
216	‘ ’, ‘kg’, ‘lb’, ‘ t’, ‘ g’	Weight units with ‘ ’ for motion
217	M, C, ‘ ’	Motion, over/under capacity or valid weight (‘ ’)
218	M, I, O, ‘ ’	Motion, invalid, over/under capacity, or valid weight (‘ ’)
219	I, O, U	Inscale, overload, underload
220	Z, ‘ ’	Centre of zero status
221	‘ ’, 1, 2	Single range (‘ ’) or range/interval 1 or 2. In Dual interval, the range printed is that of the last weight printed.
222	‘ST’, ‘US’, ‘OL’	Stable, unstable, overload
230		Send time in format: hh:mm:ss
231		Send Date in format: dd/mm/yyyy

9 Printing

9.1 Overview

Four print formats are available. They are based on a 20 or 40 column width printer and may be printed on a 20 or 40 column tally roll serial printer, or a 80 column dot matrix serial printer.

Printing may be triggered manually by a configured print key or automatically (auto-printing). Auto-printing prints when the scale reaches no motion with a weight above the zero band.

The printout will be sent to one printer only. If a USB printer is connected, this is used. Otherwise, the lowest serial port number which is configured as a printer is used. For example, if `SERIAL:SER.M.1A:TYPE=PRINT` and `SERIAL:SER.M.2A:TYPE=PRINT`, the printout will be sent via module 1 A port only.

The print format and manual/auto-printing are selected using `SERIAL:PRN.OPT:PRNT.TP`. The following table shows the options:

Format	Print trigger Print Key / Auto-print
Single line printout	
`SERIAL:PRN.OPT:PRNT.TP=`	`SINGLE / A.SING`
Example:	`0005 05/10/94 16:47 3654 kg G`
Double line printout	
`SERIAL:PRN.OPT:PRNT.TP=`	`DOUBLE / A.DOUB`
Example:	`0005 05/10/94 16:47` `3654 kg G`
Ticket printout	
`SERIAL:PRN.OPT:PRNT.TP=`	`TICKET / A.TICK`

Example:	`WEIGHT`
	`TICKET`
	`05/10/94 16:50:12`
	`ID: 0008`
	`T: 654 kg`
	`G: 3654 kg`
	`N: 3000 kg`
	`----------`

Total printout	
`SERIAL:PRN.OPT:PRNT.TP=`	`TOTAL / A.TOTAL`

Example:	`000491 01/01/1999 10:35:08 100.2 kg G`
	`000492 01/01/1999 10:35:08 105.7 kg G`
	`000493 01/01/1999 10:35:08 124.9 kg G`
	`ITEMS: 3 TOTAL: 330.8 kg`

9.2 Custom Ticket Headers

A custom header is available for ticket printouts (`SERIAL:PRN.OPT:PRNT.TP=TICKET or A.TICK`) in the `SERIAL:PRN.OPT:HEADER` setting. Up to 6 lines of 30 characters can be entered.

Within this setting item, the display shows each of the heading characters in turn using the following format:

`L.CC.XXX`

where:

- L: the line number (1..6)
- CC: the position of the character in that line (01..30)
- XXX: the ASCII code for the printed character

Use the left/right/up/down keys to edit the ASCII code for the character. Use the OK key to select the next character. Use the cancel key to exit the editor and return to the menu. Any printable ASCII codes can be used.

9.2.1 Example

The following table shows the coded entry for "JOE'S FRUIT & VEG". The 'J' would be entered as 1.08.074 for line 1, column 8, ASCII Code 74.

#	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20
1								J	O	E	'	S								
	32	32	32	32	32	32	32	74	79	69	39	83	32	32	32	32	32	32	32	32
2				F	R	U	I	T		&		V	E	G						
	32	32	32	70	82	85	73	84	32	38	32	86	69	71	32	32	32	32	32	32

9.3 Custom Ticket Format

Custom ticket formats can be entered in the `SERIAL:PRN.OPT:TIC.FMT` setting. Up to 250 characters can be entered to define the exact style of ticket printout.

Within this setting item, the display shows each of the characters in turn using the following format:

```
CC.XXX
```

where:

- CC: the character number (01 to 250)
- XXX: the ASCII code for the format character

Use the left/right/up/down keys to edit the ASCII code for the character or token. Use the OK key to select the next character. Use the cancel key to exit the editor and return to the menu. Any printable ASCII codes (see Section 9.5) or token (see Section 9.4) can be used.

9.3.1 Example

To print a ticket which looks like:

```
       JOE'S
   FRUIT & VEG
ID:000005
10/10/2012 10:25:30
        25.5 kg N
```

use the following characters/tokens:

Character No.	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16
Code	\H	I	D	:	\I	\E	\D	\E					\N	\E	\E	End
ASCII	136	073	068	058	137	133	132	133	32	32	32	32	142	133	133	00

Note

This uses the header defined above.

9.4 Custom Format Tokens

These format characters may be sent in a temporary weight string via the network communications to define the format of a single printout.

ASCII Code	Escape Sequence	Description
128	\.	Print a literal NULL character (NULL can't be entered as 00 as this is used to identify the end of the format string)
129	\A	Print lines 1-3 of the header w/o CRLF
130	\B	Print lines 4-6 of the header w/o CRLF
131	\C	Print the number of Columns of SPACE specified by the Printer Space settings
132	\D	Print Date Time field: hh:mm:ss dd:mm:yyy
133	\E	Print End of Line : literally prints CRLF (ASCII 013, 010)
134	\F	Print End of Line (CRLF) followed by 131 (print number of columns of space)
135	\G	Print Gross Weight: "weight(7) units(3) G"
136	\H	Print Custom Ticket Header
137	\I	Print ID number
142	\N	Print Net Weight: "weight(7) units(3) N"
143	\O	Print Total Weight: "weight(7) units(3) G"
146	\R	Print the number of Rows of SPACE specified by the Printer Space settings.
148	\T	Print Tare weight: "weight(7) units(3) T" (or PT or numeric Tare weight)
149	\U	Print units
151	\W	Print displayed weight: "weight(7) units(3) G" (or N)
152	\X	Print the thumbwheel IO selected text line. See Section 17.7.
155	\+	Print displayed weight as with \W but add weight to total.
156	\-	Undo last \+ operation.

9.5 ASCII codes

Code	Char	Code	Char	Code	Char	Code	Char	Code	Char	Code	Char
000	NULL	022	SYN	044	,	066	B	088	X	110	n
001	SOH	023	ETB	045	-	067	C	089	Y	111	o
002	STX	024	CAN	046	.	068	D	090	Z	112	p
003	ETX	025	EM	047	/	069	E	091	[	113	q
004	EOT	026	SUB	048	0	070	F	092	\	114	r
005	ENQ	027	ESC	049	1	071	G	093	]	115	s
006	ACK	028	FS	050	2	072	H	094	_	116	t
007	BEL	029	GS	051	3	073	I	095	-	117	u
008	BS	030	RS	052	4	074	J	096	‘	118	v
009	HT	031	US	053	5	075	K	097	a	119	w
010	LF	032	Space	054	6	076	L	098	b	120	x
011	VT	033	!	055	7	077	M	099	c	121	y
012	FF	034	”	056	8	078	N	100	d	122	z
013	CR	035	#	057	9	079	O	101	e	123	{
014	SO	036	$	058	:	080	P	102	f	124	|
015	SI	037	%	059	;	081	O	103	g	125	}
016	DLE	038	&	060	<	082	R	104	h	126	~
017	DC1	039	’	061	=	083	S	105	I	127	DEL
018	DC2	040	(	062	>	084	T	106	j		
019	DC3	041	)	063	?	085	U	107	k		
020	DC4	042	*	064	@	086	V	108	l		
021	NAK	043	+	065	A	087	W	109	m		

10 USB Interface

10.1 Device Interface

The USB device interface allows the indicator to be connected to a host PC as a slave device. The indicator will appear as a virtual serial port device to the PC. This requires a Windows driver to be installed which is supplied with the C500 viewer software. Windows versions 7, 8.1 and 10 are supported.

Once the drivers have been installed on the host PC, the serial port can be accessed in the same manner as a traditional PC serial port. This port is referred to as SER.SLV within the indicator.

10.2 Host Interface

The USB host interface allows slave devices to be connected to the indicator.

Warning!
Due to hardware limitations it is only possible to reliably connect one USB slave device to the USB host port.

Supported devices include:

- Keyboard: Connecting a keyboard will permit operation of the indicator via the keyboard. Keys are mapped as shown in the table below. The CTRL key in combination with F1-F6 will emulate a long press.

USB Keyboard	C500 Front Keys
F1	Zero
F2	Tare
F3	Gross/Net
F4	F1
F5	F2
F6	F3

- USB Mass Storage Device: Connecting a USB mass storage device will permit reading and writing files to the mass storage device. Firmware can be updated from an update file stored on a USB disk. Settings, DSD and changelog contents can be written to a USB disk. These settings are accessed via the `FILE` menu in safe and full setup menu (see Section 5.12). When a USB disk is connected, "USB.DSK ADDED" will appear on the display after a 5 second delay. If a USB disk is connected while the menus are active, this message will not be displayed until exiting the menus.

Note
Only FAT32 files systems are supported.

Warning!
It is recommended that an attached USB disk is ejected before removal to ensure all files have been written to the disk. See `FILE:EJECT` in the setup menus (Section 5.12).

- USB Printing: A text based USB printer can be connected to the indicator for printing. An example of a text based USB printer is the Custom Q1 printer (http://www.custom.it).
- USB Serial Ports: A USB serial port may be connected to provide the serial communication functions listed in `SERIAL:SER.HST:TYPE` (see Section 5.5). Supported chipsets include FTDI FT232R, Prolific PL2303 and Silicon Labs CP210x.

Warning!

Power off the indicator before disconnecting a USB serial port from the USB host port.

11 Change Log

11.1 Introduction

The indicator contains a change log that records trade significant events. These include changes to trade relevant settings, creation and clearing of the change log and trade relevant firmware upgrades. Only trade relevant settings are stored in the change log.

The change log records the following information about each change:

- The calibration counter at the time of the change
- The date and time of the change
- The system database name
- setting name that has changed
- The menu name of the setting that has changed
- The index of the setting that has changed. This is only used for array settings, such as resolution, where there is a resolution for each range
- Previous value of the setting
- New value of the setting
- The log mask

11.2 Format and Capacity

The change log is stored as a comma separated value (CSV) file. CSV files can be easily viewed in most spreadsheet programs.

The change log has a maximum size of 512 kilobytes. Records vary in length, but are generally in the order of 80bytes, giving approximately a 6500 record maximum of the change log. Once the change log is full, the indicator will refuse to save any setting changes until the change log has been cleared. This can be performed from the change log setup menu.

11.3 Changelog Security

The change log is stored on the internal indicator filesystem which is not accessible, and hence cannot be tampered with. The indicator application is the only means by which to access the change log. The indicator contains Alibi software which allows the change log to be viewed. The Alibi change log viewer is part of the trade approval.

When a USB disk is attached to the indicator, a copy of the change log can be made to the USB disk. This copy of the change log can always be verified against internal change log, via the Alibi change log viewer.

12 Digital Storage Device (DSD)

12.1 Introduction

The indicator contains a Digital Storage Device (DSD) to record traceable readings. The DSD records the following information about each entry:

- The print ID
- The weight reading
- The tare weight
- The date and time

12.2 Format and Capacity

The DSD is stored in a binary format internally. The DSD has a maximum size of 512 kilobytes, and can contain approximately 11900 records. The indicator can be configured to either prompt the user to purge 10% of records from the DSD, or automatically purge 10% of records, once the DSD is full. Records can also be manually purged from the DSD from the DSD:PURGE setup menu.

12.3 Changelog Security

The DSD is stored on the internal indicator filesystem which is not accessible, and hence cannot be tampered with. The indicator application is the only means by which to access the DSD. The indicator contains Alibi software which allows the DSD records to be viewed. The Alibi DSD viewer is part of the trade approval.

When a USB disk is attached to the indicator, the DSD can be exported to the USB disk in a comma separated value (CSV) file. CSV files can be easily viewed in most spreadsheet programs. This copy of the DSD can always be verified against internal DSD, via the Alibi DSD viewer.

12.4 Configuring the DSD

The DSD is not enabled by default. To prepare the DSD for use:

- Enter the full setup menus
- Set `DSD:ENABLE` to On
- Initialise the DSD with `DSD:INIT`
- Configure DSD purging with `A.PURGE`
- After this all traceable printouts will be recorded in the DSD

Note

Performing `DSD:INIT` will delete all existing DSD records.

12.5 Writing Records

A record will be stored in the DSD whenever a traceable weight is generated. A traceable weight is only generated for a single, double or ticket printout. For more information on print setup see Section 5.5 and Section 9. The DSD will also be written when a print occurs in alibi mode.

12.6 Reading Records

The records can be viewed in Alibi mode or exported to a USB disk in comma separated value (CSV) format from the File menu (see Section 10). CSV files can be easily viewed in most spreadsheet programs. For more information on viewing DSD records in Alibi mode see Section 13. DSD records exported in CSV format are formatted as follows:

```
Print ID, Reading, Units, Status, Tare Weight, Units, Status, Valid, Timestamp
3, 223.5, kg, G, 223.5, kg, T, Yes, 2012-06-13 05:43:47
```

13 Alibi Application

13.1 Introduction

The Alibi application is part of the trade approval and makes it possible to verify scale readings, view DSD and change log entries. It can be accessed from the main application by a long press of the Select key, and then selecting Alibi from the list.

On starting, it displays the following information:

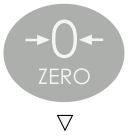
- "Alibi" to identify that the Alibi application is running
- ADC library version (this is listed on the trade approval)
- ADC library checksum (this is listed on the trade approval)
- Current calibration counter value

13.2 Changing Modes

In all Alibi modes, a long press of the F2 key will access the mode menu. Pressing the up or down keys will advance through each mode, and the F2 key will enter that mode. The "-END-" entry will exit Alibi mode and return to the main application.

13.3 Weighing and Counting Modes

The following keys apply in Alibi Weighing and Counting modes:



▽

Short Press	Zero	Zero the scale.
Long press	(none)	

◁

Short Press	Tare	Tare the scale
Long press	Preset Tare	Enter a preset tare

▷

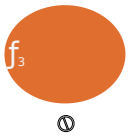
Short Press	Select	Toggle between gross and net weight.
Long press	Alibi Mode	

△

Short Press	Print	Print a simple traceable printout:
		`00000039 2012\06\16 07:04:08`
		`50.0 kg G`
		`50.0 kg N`
		`0.0 kg T`

Long Press	(none)	

Short Press	(None)	
Long press	Mode menu	Long press to access the Alibi mode menu



Short Press	Enter Pieces	Enter pieces (only in counting mode)
Long press	(None)	Cancel pieces display and return to weight display (only in counting mode)

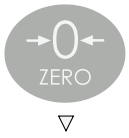
13.4 DSD and Change log Viewer Modes

The following keys apply in DSD and Change log Viewer modes:

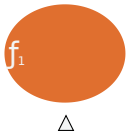
Short Press	Prev record	Go to the previous record
Long press	First record	Go to the first (oldest) record

Short Press	Next record	Go to the next record
Long press	Last record	Go to the last (newest) record

Short Press	Search	Search for print ID (DSD viewer), or calibration counter value (Change log viewer)
Long press	Alibi Mode	Long press to access the Alibi mode menu



Short Press	Prev item	Go to the previous item within the current record
Long press	First item	Go to the first item within the current record



Short Press	Next item	Go to the next item within the current record
Long press	Last item	Go to the last item within the current record

In the DSD viewer, the following items are shown for each record:

- Prn.Id: Print identifier
- Weight: Print weight
- Tar.Wgt: Tare weight
- Date: Date stamp of print (yy.mm.dd)
- Time: Time stamp of print (hh.mm.ss)

In the Change log viewer, the following items are shown for each record:

- Name: Name of parameter that has changed (will scroll if longer than 6 digits)
- Index: Index of parameter that has changed (only shown for parameters with indexes such as fullscale and resolution)
- Old.Val: The old value prior to the change (will scroll if longer than 6 digits)
- New.Val: The new value after the change (will scroll if longer than 6 digits)
- Date: Date stamp of parameter change (yy.mm.dd)

- Time: Time stamp of parameter change (hh.mm.ss)
- Cal.Cnt: The calibration counter value at the parameter change

14 Ethernet Interface

14.1 Overview

The Ethernet interface provides connectivity via a standard 802.3 network. The indicator contains a 100MBit Ethernet interface. This interface allows connectivity to other networked devices, such as PCs. The indicator IP network settings can be configured from the menus under `ETH.NET`. See Section 5.6. Fixed or DHCP assigned addresses can be used. When DHCP is used, the IP settings are read-only so the device's IP address can be determined.

In addition, the operator menu provides the `IP.INFO` menu to quickly view the device's IP settings. See Section 6.

14.1.1 Web Interface

The indicator provides a simple web interface that can be accessed from a web browser on a PC. The web interface allows new firmware to be uploaded to the indicator.

14.1.2 TCP Socket Interfaces

The indicator provides two TCP socket interfaces for extracting data from the indicator.

1. Referred to as `TCP.IN.1` within the indicator, this interface provides a bi-directional TCP socket connection over which the standard indicator serial command set can be used. The indicator viewer (C500-510) uses this interface when using a TCP connection.

2. Referred to as `TCP.IN.2` within the indicator, this interface provides a uni-directional TCP socket connection over which the indicator can transmit Automatic Output data.

A maximum of 20 concurrent connections shared between `TCP.IN.1` and `TCP.IN.2` are permitted.

Note

If the connection limit is being reached, it may be because connections are not being closed by clients, and left hanging. Setting a timeout value may solve this issue. See `T.OUT` in Section 5.5.

14.1.3 UDP Socket Interfaces

The indicator provides four UDP socket interfaces for extracting data from the indicator.

1. Two outgoing UDP sockets (`UDP.O.1` and `UDP.O.2`):

   (a) These sockets will read commands from and send data to a fixed IP and UDP port combination specified by `SERIAL:UDP.O.x:DST.IP` and `SERIAL:UDP.O.x:DST.PRT`.

   (b) The port type (auto output, network, etc) can be configured independently for each port.

   (c) Setting the type to off, or setting the destination IP address to 0.0.0.0 or destination port to 0 disables the socket.

   (d) The indicator source port will be the same as the destination port (`SERIAL:UDP.O.x:DST.PRT`).

2. Two incoming UDP sockets (`UDP.IN.1` and `UDP.IN.2`):

   (a) These sockets listen for data on a specified UDP port (`SERIAL:UDP.IN.x:PORT`) with in the indicator.

(b) They read commands from any source IP and port and send reply data to the last source IP and port that sent data.

(c) The port type (auto output, network, etc) can be configured independently for each port.

(d) Setting the type to off, or setting the source port to 0 disables the socket.

Warning!

Note that UDP is an un-reliable protocol. This means that there is no guarantee of packet delivery or packet arrival order. Your client will need to deal with this. Alternatively, use a TCP connection.

Warning!

The C500 UDP ports allow the use of broadcast destination IP addresses. These should be used with care, as too much broadcast traffic will degrade network performance.

14.1.4 Modbus TCP Interface

The indicator provides Modbus TCP. See Section 19.

15 Accessory Modules

15.1 Introduction

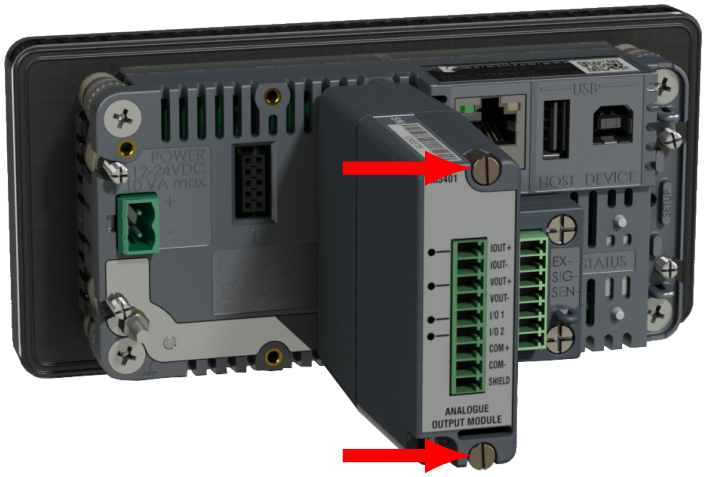
The indicator can be expanded by the installation of optional accessory modules. A range of modules are available:

- M5401 Analogue output module
- M5301 8 I/O module
- M5101 AC power module
- M5201 RS232 full duplex / RS232 transmit only serial communications module
- M5203 RS485 full duplex / RS485 transmit only serial communications module

Two modules can be installed on the indicator.

15.2 Installation

Isolate the indicator from the power before installing any module. Each module is installed onto the back panel of the indicator as shown below.



15.3 Mapping

Serial modules are mapped according to their physical location on the rear of the indicator. The module in slot 1 maps to S1A / S1B. The module in slot 2 maps to S2A / S2B.

15.4 Configuration

Modules are configured using the indicator setup menus. See Sections 5.5, 5.6, 5.9 and 5.10

15.5 Details

See the datasheet for your module.

16 Setpoints

16.1 Introduction

The C500 series has 8 setpoints.

The status of the setpoints is shown on the display (see Section 3.2). An optional output driver module can be fitted to allow the setpoints to drive external devices. The lamps then show the status of the output drivers.

Each of the setpoints provides a comparator function that can be modified in the digital setup for switching direction, hysteresis, logic, etc. These settings are all that are required to configure normal level or limit operation. Free-flight (free-fall) functions can be enabled to configure the C500 series for weight-batching applications.

Weight target and inflight settings can be pre-set in the digital setup. This method is used where the settings are changed infrequently, and are to be as tamper-proof as possible. Optionally, weight target and inflight settings can be set from the front panel keys. This allows settings to be changed much more readily by the operator.

16.2 Connection

Refer to the module datasheet for physical connection details.

16.3 General Settings

Settings in the general (GEN) submenu apply to all setpoints.

- JOG.ON: If the timing has been set to PULSE this sets the on duration of each pulse.
- JOG.OFF: If the timing has been set to PULSE this sets the off time between each pulse.
- JOG.SET: If the timing has been set to PULSE this sets the number of pulses to be output each time the setpoint is triggered before waiting for no motion.
- MAX.SET: If the timing has been set to PULSE this sets the maximum number of jog sets to be output each time the setpoint is triggered. A setting of 0 means no limit.
- FEEDER: Controls the behaviour of multiple weigh-in and weigh-out setpoints.
  - SINGLE allows only one weigh-in or weigh-out setpoint to be active at a time. Preference is given to lower numbered setpoints in single mode.
  - MULT. allows multiple weigh-in and weigh-out setpoints to be active at the same time.
- DLY.CHK: If enabled this starts a delay after a weigh-in or weight-out setpoint has been triggered. During the delay the weight value will not be checked for weigh-in and weigh-out setpoints.

16.4 Common Settings

There are a number of settings that are common to all setpoint types. These are as follows:

- LOGIC: This setting determines whether the output is normally on or normally off.
  - Logic HIGH means the output follows the activity of the setpoint and is on when the setpoint conditions are met.
  - Logic LOW reverses the operation of the output.

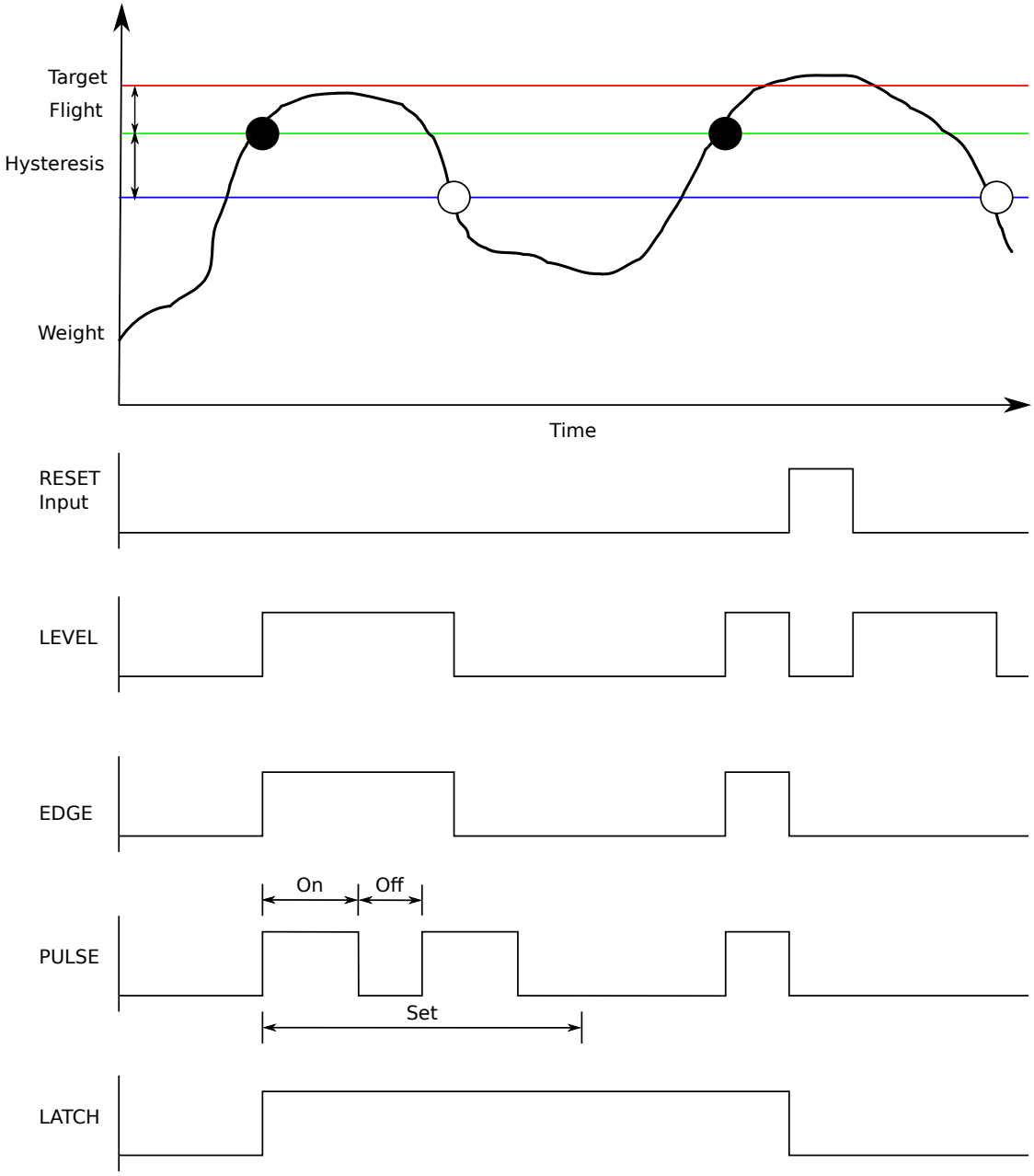


Figure 1: Comparison of output timing on Over setpoint

For example: Consider a Center-of-Zero status setpoint. This type of setpoint is active when the Centre-of-Zero annunciator is lit. With logic HIGH an output would turn on whenever the Centre-of-Zero annunciator was lit. With logic LOW the output would turn off when the Centre-of-Zero annunciator is lit and remain on otherwise.

Note that the outputs revert to the off state when the instrument SETUP menus are active.

- ALARM: Select what alarm response is triggered when the setpoint is active.
  - SINGLE sounds a single beep every two seconds.
  - DOUBLE sounds a double beep every two seconds.
  - CONT beeps continually.
  - FLASH flashes the display.

  Note that the Alarm conditions are not influenced by the LOGIC setting, i.e. they follow the activity of the setpoint regardless of the physical state of the output.

- TIMING: Select the output timing of the setpoint. The following examples are explained in the context of an OVER setpoint however the timing options are available for all setpoint types. See figure 1 on page 65 for further clarification. Options are:
  - LEVEL: the setpoint is active whenever the weight has gone over the target, has not dropped below the hysteresis value and the reset input is not currently active.
  - EDGE: The setpoint becomes active when the weight goes over the target. The setpoint becomes inactive when the weight goes below the hysteresis value or the reset input becomes active.
  - LATCH: The setpoint becomes active when the weight goes over the target. The setpoint becomes inactive when the reset input becomes active.
  - PULSE: Once the weight goes over the target the output will become active and the setpoint will begin the jog on time. Once that time has elapsed the output will become inactive for the jog off time. If the jog set number (JOG.SET) has been set for more than one then the cycle will repeat for the set number of times. The reset input becoming active is the only reason the set number of cycles will not be completed, the weight is completely ignored once the cycle has started until the end of the set when it will wait for no motion to see if another set is required. This process will stop when the maximum number of sets (MAX.SET) is reached, the target is reached or the reset input is triggered.
- RESET: Select which reset input is used as to disable the setpoint. Options are:
  - NONE
  - SP.RST.A
  - SP.RST.B

16.5 Over, Under, Weigh in and Weigh Out Setpoints

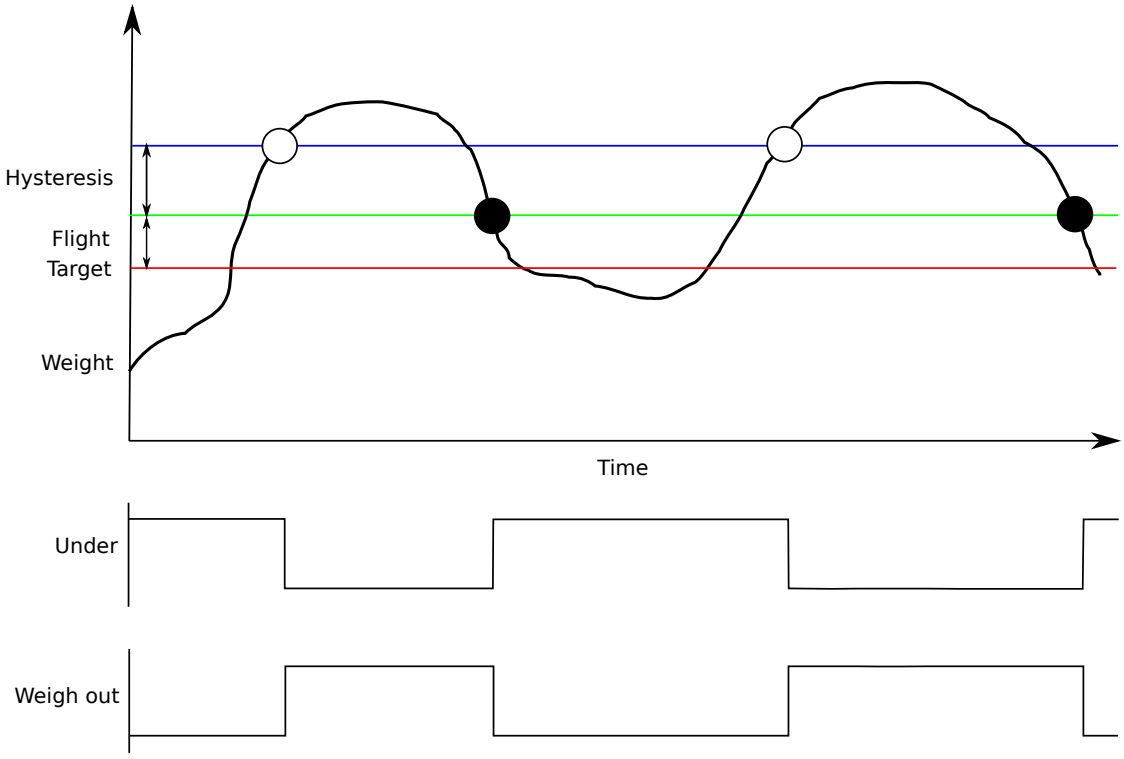


Figure 2: Comparison of Under and Weigh out setpoints

In addition to the common settings the following settings control the operation of the OVER, UNDER, W.IN and W.OUT setpoints

- TARGET: Set the target weight for the setpoint.
- Hysteresis (HYS): This setting determines the change in weight required for an active setpoint to become inactive again. A value of zero still leaves 0.5 graduations of hysteresis.
- FLIGHT: Set the amount of material that is in-flight. This is used to turn the output off before the target it reached so it does not overshoot.
- LOCK: Set if the target and flight values will be locked from editing in the OPER menu.
- SOURCE: Select the weight source for the setpoint to use. Options are:
  - GROSS uses gross weight only
  - NET uses net weight only

16.6 Status Based Setpoint Types

The following setpoint types are all based on the status of the instrument.

- NONE: Setpoint is always inactive.
- ON: Setpoint is always active. This type of setpoint is useful to show that the instrument is running.
- Centre of Zero (COZ): Setpoint is active when the gross weight is within COZ bounds.
- ZERO: Setpoint is active when the gross weight is within the Zero Band setting.
- NET: Setpoint is active when the NET annunciator is lit.
- MOTION: Setpoint is active when the MOTION annunciator is lit.
- ERROR: Setpoint is active when the instrument detects any error condition signified by the display of Exxxxx on the display.
- BUZZER: Setpoint is active when the buzzer beeps.

16.7 Example 1 (filling a bag)

Type = Over, Logic = High, Source = Gross, Target = 2000kg, Inflight = 50 kg, Hysteresis = 5 kg. Initial weight = 0 kg.

Trip point = target - inflight = 2000 - 50 = 1950 kg.

The output will switch ON at a weight over 1950 kg and switch OFF again at a weight under 1945 kg.

If the type is changed to weigh in the output will be ON at 0 kg, switch OFF at a weight over 1950 kg and switch ON again at a weight under 1945 kg.

16.8 Example 2 (loss in weight system)

Type = Under, Logic = High, Source = Net, Target = -100kg, Inflight = 5 kg, Hysteresis = 1 kg. Initial weight = 0 kg.

Trip point = target + inflight = -100 + 5 = -95 kg.

The output will switch ON at a weight under -95 kg and switch OFF again at a weight over -94 kg.

17 Remote Input Functions

17.1 Introduction

The F1, F2 and F3 keys on the front of the C520 can be set to a variety of functions depending on the application. There are also 8 external key inputs (connected to the optional IO modules).

17.2 Functions

Function	F key	Remote	Description
-			No function
0		✓	Zero key (see Section 17.3)
t		✓	Tare key (see Section 17.3)
G		✓	Gross/net key (see Section 17.3)
P	✓	✓	Print key (see Section 17.3)
b		✓	Blank the display Section 17.4. When active this input causes the front display to be blanked to "——" and blocks the operation of the front keys. This function is intended for use with tilt sensors on mobile weighing platforms to block operation of the weight indicator if the scale is not level. This function may also be used to block operation of the instrument pending authorization or payment etc.
L		✓	Lock the indicator (see Section 17.5)
S	✓	✓	Show total (see Section 17.6)
C	✓	✓	Clear total (see Section 17.6)
u	✓	✓	Undo last print (see Section 17.6)
1, 2, 3, 4, 5, 6	✓	✓	Single transmission from serial port (see Section 17.7). 1 = Module 1 Port A, 2 = Module 1 Port B, 3 = Module 2 Port A, 4 = Module 2 Port B, 5 USB slave, 6 USB host.
H	✓	✓	Hold/unhold the current weight. The units will flash while the weight is held.
E	✓	✓	Show/hide the peak weight reading. The units will flash while the peak weight is shown. A long press will clear the current peak weight. This is available in industrial mode only.
F	✓	✓	Acquire and show livestock weight. Display shows "——"until the livestock weight is held. The units will flash while the weight is held. A long press will cancel livestock operation and show the current weight on the scale.
8		✓	Use this IO as part of thumbwheel input to select text line to print when custom print format contains token 152 (see Section 17.8).
n	✓	✓	Switch units. Cycle through available units.
A		✓	Use this input to reset any setpoints using setpoint reset A.
B		✓	Use this input to reset any setpoints using setpoint reset B.

17.3 Remote Access

The function of 4 fixed function keys may be implemented with the remote keys. The Front Panel keys are designated "0TGP" to match Zero, Tare, Gross/Net and Print.

17.4 Blanking

When this input is active, the front display is blanked to "——", and the operation of the front keys is blocked. This function is intended for use with tilt sensors on mobile weighing platforms to block operation of the weight indicator if the scale is not level. This function may also be used to block operation of the instrument pending authorization or payment etc.

17.5 Locking

When this input is active, all keys including the remote keys, are blocked. This may be used with a keylock switch to lock the instrument when not in use.

17.6 Totalising

Totalising keys are only available if `SERIAL:PRN.OPT:PRNT.TP=TOTAL` or `SERIAL:PRN.OPT:PRNT.TP=A.TOTAL`.

Add-to-total is performed by the print function before printing. There are 3 additional other functions are available remotely:

- Show Total: The total weight is displayed.
- Clear Total: Prints the total weight and clears it.
- Undo last print: Subtract the last item added to totals, and prints "Last Entry Cancelled"

17.7 Single Serial Transmission

Single transmit functions (1 .. 6) will transmit the weight once to the following serial ports:

- 1: Module 1 Port A
- 2: Module 1 Port B
- 3: Module 2 Port A
- 4: Module 2 Port B
- 5: USB slave
- 6: USB host

`SERIAL:SERx:TYPE` must be set to SINGLE. See Section 5.5 for details.

The format of the message is set up in the `SERIAL:AUT.OPT` menu.

The single serial transmission functions are a convenient way to implement simple PLC communications and logging without the complexity of setting up two way communications on the serial ports.

17.8 Thumbwheel IO selection of printer text line

This allows a thumbwheel connected to several IO to select one of nine text strings to be printed via the print token 152 (see Section 9.4). The text to be printed can be set using the TXT command (see Section 21.3.52).

All IOs configured for thumbwheel are combined together to select the string from lowest IO giving the least significant bit to highest IO giving the most significant bit.

17.9 Unit switch

This function will switch between the available units.

17.10 Setpoint reset inputs

This allows a setpoint to be disabled by an input.

18 Network Communications

18.1 Introduction

The RS-232, RS-485, Ethernet and the optical communications can be used for networking.

Warning: The calibration counter is incremented when the calibration related settings are changed. This means that calibration via a serial port cannot be carried out without affecting the certification of a trade installation.

Serial communications parameters like BAUD, PARITY, etc for the RS232 or RS485 serial ports are setup in the HDWARE menu. The optical communications port is fixed to operate at 9600 baud, no parity, 8 data bits and 1 stop bit. The optical communications cable must be used.

18.2 Network rinCMD

The rinCMD network protocol, formally known as Protocol B, uses ASCII characters with a single master POLL / RESPONSE message structure. All information and services are provided by registers each of which has its own register address.

18.2.1 Basic Message Format

The basic message format is as follows:

ADDR	CMD	REG	:DATA	↩

ADDR is a two character hexadecimal field corresponding with the following:

ADDR	Field Name	Description
80_H	Response	0 for messages sent from the master (POLL). 1 for messages received from an instrument (RESPONSE).
40_H	Error	Set to indicate that the data in this message is an error code and not a normal response.
20_H	Reply Required	Set by the master to indicate that a reply to this message is required by any slave that it is addressed to. If not set, the slave should silently perform the command.
$00_H..1F_H$	Indicator Address	Valid instrument addresses are 01_H to $1F_H(1..31)$. 00_H is the broadcast address. All slaves must process broadcast commands. When replying to broadcasts, slaves reply with their own address in this field.

CMD is a two character hexadecimal field:

<table>
<tr><th>CMD</th><th>Command</th><th>Description</th></tr>
<tr><td>01_H</td><td>Read Type</td><td>Read the register type.

<table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>INT8: Signed 8 bit value</td></tr><tr><td>1</td><td>UINT8: Unsigned 8 bit value</td></tr><tr><td>2</td><td>INT16: Signed 16 bit value</td></tr><tr><td>3</td><td>UINT16: Unsigned 16 bit value</td></tr><tr><td>4</td><td>INT32: Signed 32 bit value</td></tr><tr><td>5</td><td>UINT32: Unsigned 32 bit value</td></tr><tr><td>6</td><td>STRING: Null terminated string</td></tr><tr><td>9</td><td>WEIGHT: Weight value (weight, status and units)</td></tr><tr><td>11</td><td>EXECUTE: Perform an action (eg tare)</td></tr><tr><td>13</td><td>STREAM: Register for streaming other registers</td></tr><tr><td>18</td><td>IP: IPv4 Address register</td></tr><tr><td>19</td><td>REGISTER: For storing register id for streaming</td></tr><tr><td>20</td><td>BINBUFFER: Binary buffer (like string but can contain nulls)</td></tr></table></td></tr>
<tr><td>02_H</td><td>Read Minimum</td><td>Read the minimum value permitted for this register.</td></tr>
<tr><td>03_H</td><td>Read Maximum</td><td>Read the maximum value permitted for this register.</td></tr>
<tr><td>05_H</td><td>Read Literal</td><td>Read register contents in a human readable format.</td></tr>
<tr><td>$0F_H$</td><td>Read Permission</td><td>Read the register permissions mask.

<table><tr><th>Mask</th><th>Description</th></tr><tr><td>00_H</td><td>Register is not readable</td></tr><tr><td>01_H</td><td>Register is readable in safe setup and higher</td></tr><tr><td>02_H</td><td>Register is readable in full setup only</td></tr><tr><td>03_H</td><td>Register is always readable</td></tr><tr><td>00_H</td><td>Register is not writable</td></tr><tr><td>04_H</td><td>Register is writable in safe setup and higher</td></tr><tr><td>08_H</td><td>Register is writable in full setup only</td></tr><tr><td>$0C_H$</td><td>Register is always writable</td></tr></table></td></tr>
<tr><td>10_H</td><td>Execute</td><td>Execute function defined by the register using parameters supplied in the DATA field.</td></tr>
<tr><td>11_H</td><td>Read Final</td><td>Read register contents in a hexadecimal data format.</td></tr>
<tr><td>12_H</td><td>Write Final</td><td>Write the DATA field to the register.</td></tr>
<tr><td>16_H</td><td>Read Final (Decimal)</td><td>Same as Read Final except numbers are decimal.</td></tr>
<tr><td>17_H</td><td>Write Final (Decimal)</td><td>Same as Write Final except numbers are decimal.</td></tr>
</table>

REG	is a four character hexadecimal field that defines the address of the Register specified in the message. See Section 18.4: rinCMD Registers for a list of registers used by the instrument. The viewer software will show the register address for each setting in the menu structure when they are accessed.
:DATA	carries the information for the message. Some messages require no DATA (eg Read Commands) so the field is optional. When a DATA field is used a : (COLON) character is used to separate the header (ADDR CMD REG) and DATA information.
↩	is the message termination (`CR LF` or `;`).

Note

The hexadecimal codes are combined in the fields described above when multiple options are active at the same time. For example an error response message from instrument address 5 would have an `ADDR` code of $C5_H(80_H + 40_H + 05_H)$.

18.2.2 Termination

Message termination is possible in two ways. For normal communications that do not involve checksums use either a `CR LF` (ASCII 13, ASCII 10) as a terminator or a semicolon (; ASCII). There is no start-of-message delimiter.

To use a checksum the message is framed as:

```
SOH <Message> CRC EOT
```

where:

- SOH (1 byte): 01_H
- CRC is a 4 character hexadecimal field comprising the 16 bit CRC checksum. The CRC uses the 16 bit CCITT polynomial calculation and includes only the contents of the "Message" section of the transmission.
- EOT (1 byte): 04_H

18.2.3 Error Handling

If a command cannot be processed and a reply was requested, the ERROR bit in the ADDR field is set and the DATA field is set to an error message. This 2-byte error message is formed by adding a major and minor error code. For example, the error 0204_H corresponds to a major error of 0200_H, and a minor error of 04_H).

Error Code	Error	Description
0100_H	Parsing Error	The error occurred when interpretting the incoming message.
0101_H	Address error	An error occurred when parsing `ADDR` into a number.
0102_H	Command error	An error occurred when parsing `CMD` into a number.
0103_H	Reg error	An error occurred when parsing `REG` into a number.
0104_H	Chan error	The internal channel string is longer than the maximum (9).
0105_H	Delimiter error	The colon delimator is not present.
0106_H	Data error	The data string is longer than the maximum (200)
0108_H	Framing error	The message is too long for the buffer.
0200_H	CRC Error	There was a problem with the checksum (CRC).
0201_H	CRC Parse Error	The CRC code could not be parsed from the message.
0202_H	CRC Error	The CRC code does not match that of the message. The message was corrupted.
0300_H	Register Error	`REG` is not valid, or does not exist on this device.
0400_H	Read Error	There was an error reading from the register.
0401_H	Permission error	You do not have permission to read this register.
0402_H	Unknown type error	Internal error: Register type is not known to RinCmd.
0403_H	No type data error	Internal error: Unable to read the registers data structure.
0404_H	Command error	The `CMD` is not valid for this register type.
0405_H	Bad data error	The stream data is not valid.
0406_H	Null register error	The register cannot be found.

Error Code	Error	Description
0407_H	Buffer size error	The string reply will not fit in the transmission buffer.
0408_H	Snprintf error	A function within the library has failed to run correctly.
0500_H	Write Error	
0501_H	Permission error	You do not have permission to write to this register.
0502_H	Unknown type error	Internal error: Register type is not known to RinCmd.
0503_H	No type data error	Internal error: Unable to write to the registers data structure.
0504_H	Command error	The `CMD` is not valid for this register type.
0505_H	Parse error	Unable to parse the data for the command type. If this is a decimal write, ensure all characters numbers. If this is a hexadecimal write, ensure all characters are hex (0-9, A-F). If this is a string register, the characters are outside the range of the
0506_H	Range low error	The value that is attempting to be written is below the minimum for the register.
0507_H	Range high error	The value that is attempting to be written is above the maximum for the register.
0508_H	Length low error	The string that is attempting to be written to the register is below the minimum length.
0509_H	Length low error	The string that is attempting to be written to the register is above the maximum length.
$050A_H$	Null register error	The register cannot be found.
$050B_H$	Buffer size error	Internal error: The entire string is not able to be written to the buffer.
$050C_H$	Snprintf error	A function within the library has failed to run correctly.
$050D_H$	Snprintf error	A function within the library has failed to run correctly.
$050E_H$	Type length error	A number above the maximum for the register cannot be written (e.g. UINT8 register cannot store a number greater than 255.)
0600_H	Exec Error	
0601_H	Permission error	You do not have permission to execute this register.
0602_H	Unknown type error	Internal error: Register type is not known to RinCmd.
0603_H	No type data error	Internal error: Unable to execute using the registers data structure.
0604_H	Command error	The `CMD` is not valid for this register type.
0605_H	Null register error	The register cannot be found.
0606_H	Writeback error	The register execute function returned an improper value.
0700_H	Not Implemented Error	A non-implemented `CMD` was given.

18.3 rinCMD Examples

Read Gross Weight	
`COMMAND A:` `20110026:;`	`COMMAND A:` Read Gross Weight (Register 0026_H): `ADDR` = 20_H: Reply required from any instrument `CMD` = 11_H: Read Final `REG` = 0026_H: Gross Weight
`RESPONSE A:` 81110026:00000064;	`RESPONSE A:` Response is from instrument #1 which currently has a Gross weight of 64_H(100) kg.

Read Gross Weight	
COMMAND A: 20050026:;	COMMAND A: Read Gross Weight (Register 0026_H): ADDR = 20_H: Reply required from any instrument CMD = 05_H: Read Literal REG = 0026_H: Gross Weight
RESPONSE A: 81110026: 100 kg G;	RESPONSE A: Same response from instrument #1 but in literal format.

Zero calibrate scale	
COMMAND A: 21100102:;	COMMAND A: Attempt to zero calibrate scale (Register 0102_H): ADDR = 21_H: Reply required from instrument #1 CMD = 10_H: Execute REG = 0102_H: Zero calibrate scale
RESPONSE A: C1100102:0601;	RESPONSE A: Instrument #1 reports "ERROR: Exec Error, Permission error". (Writing to this register requires a passcode).
COMMAND B: 21170019:1234;	COMMAND B: Enter full passcode (Register 0019_H): ADDR = 21_H: Reply required from instrument #1 CMD = 17_H: Write Literal Decimal REG = 0019_H: Gross Weight DATA = 1234: Example passcode
RESPONSE B: 81170019:0000;	RESPONSE B: Instrument #1 reports "Passcode Accepted".
COMMAND C: 21100102:;	COMMAND C: Zero calibrate scale (Register 0102_H): ADDR = 21_H: Reply required from instrument #1 CMD = 10_H: Execute REG = 0102_H: Zero calibrate scale
RESPONSE C: 81100102:0000;	RESPONSE C: Instrument #1 reports successful calibration.
COMMAND D: 21100010:;	COMMAND D: Save the settings (Register 0010_H): ADDR = 21_H: Reply required from instrument #1 CMD = 10_H: Execute REG = 0010_H: Save settings
RESPONSE D: 81100010:0000;	RESPONSE D: The settings have been successfully saved.

Zero Scale	
COMMAND A: 21100300:;	COMMAND A: Execute the zero register (Register 0300_H): ADDR = 21_H: Reply required from instrument #1 CMD = 10_H: Execute REG = 0300_H: Zero function
RESPONSE A: 81100300:00000006;	RESPONSE A: Instrument #1 reports error 6 (scale is in motion).
COMMAND B: 21100300:;	COMMAND B: Execute the zero register (Register 0300_H): ADDR = 21_H: Reply required from instrument #1 CMD = 10_H: Execute REG = 0300_H: Zero function
RESPONSE B: 81100300:00000000;	RESPONSE B: Instrument #1 is zeroed.

Stream the displayed weight on change using 0040_H register set.	
COMMAND A: 21120042:0025;	COMMAND A: Add the display weight register to the streaming registers (Register 0042_H): ADDR = 21_H: Reply required from instrument #1 CMD = 12_H: Write Literal REG = 0042_H: Streaming register 1 DATA = 0025: Displayed weight register (hex)
RESPONSE A: 81120042:0000;	RESPONSE A: Instrument #1 has set streaming register 1 to register 0025_H.
COMMAND B: 21120041:5;	COMMAND B: Set the frequency to stream on change (Register 0041_H): ADDR = 21_H: Reply required from instrument #1 CMD = 12_H: Write Literal REG = 0041_H: Streaming frequency register DATA = 5: Stream on change
RESPONSE B: 81120041:0000;	RESPONSE B: Instrument #1 has been set to stream on change.
COMMAND C: 21100040:1;	COMMAND C: Start streaming (Register 0040_H): ADDR = 21_H: Reply required from instrument #1 CMD = 10_H: Execute REG = 0040_H: Streaming start register DATA = 1: Start streaming
RESPONSE C: 81100040:0000;	RESPONSE C: Instrument #1 will begin streaming on change.

COMMAND D: 21100040:0;	COMMAND D: Stop streaming (Register 0040_H): ADDR = 21_H: Reply required from instrument #1 CMD = 10_H: Execute REG = 0040_H: Streaming start register DATA = 0: Stop streaming
RESPONSE D: 81100040:0000;	RESPONSE D: Instrument #1 will stop streaming on change.

18.4 rinCMD Registers

Name	Address	Type	R	W	Description
Software Model	0003_H	string	A	N	Returns software loaded (e.g. C520)
Software Version	0004_H	string	A	N	Returns software version (e.g. v1.0.0)
Serial Number	0005_H	uint32	A	N	Returns instrument serial number
User Defaults	0007_H	execute	N	F	Set the indicator to default settings. Values are: 0: Default all settings (user, runtime, calibration, network) 1: Default user database except calibration 2: Default runtime database 3: Default calibration 4: Default network settings 5: Default licensing database
Keyboard Buffer	0008_H	uint8	N	A	Adds a key to the key buffer. The short press key codes are shown below. For long presses, set the most significant bit to 1. Key codes are: 11: Zero key 12: Tare key 13: Gross/Net key 14: F1 key 15: F2 key 16: F3 key 32-40: I/O 1 - I/O 8
LCD Bitmap	0009_H	string	A	N	Returns the LCD bitmap
LCD Display Mode	$000D_H$	execute	N	A	Change the display mode. 0 for system control, 2 for register control.
Save settings	0010_H	execute	N	A	Save settings and re-initialise
OIML Calibration Counter	0012_H	uint32	A	N	OIML Calibration Counter
NTEP Calibration Counter	0013_H	uint32	A	N	NTEP Calibration Counter
NTEP Configuration Counter	0014_H	uint32	A	N	NTEP Configuration Counter
Trade Enable	0015_H	uint8	A	N	Is the indicator suitable for trade use. 0 = No, 1 = Yes. Always returns 1, as all C520s are suitable for trade use.
Reset	0016_H	execute	N	A	Reboot the indicator
Set full permissions	0019_H	uint32	F	A	Write full passcode to elevate permissions
Set safe permissions	$001A_H$	uint32	S	A	Write safe passcode to elevate permissions
ADC Sample Number	0020_H	uint32	A	N	Read current sample number since last power on

Name	Address	Type	R	W	Description
System status	0021_H	uint32	A	N	Read current system status. Bits are as follows: 0-7: Last calibration result 8: Reserved 9: Scale is in Net mode 10: Scale is within the zero band 11: Scale is within center of zero 12: Scale is in motion 13: Calibration is in progress 14: Menus are active 15: Scale is in error 16: Scale is underloaded 17: Scale is overloaded 18: ADC operation has succeeded 19: ADC operation in progress 20: Tilt Hi XY (not supported in C500) 21: Tilt Hi Y (not supported in C500) 22: Tilt Hi X (not supported in C500)
System error	0022_H	uint32	A	N	Read current system error. Refer to Section 24.4 for values.
Absolute mV/V	0023_H	int32	A	N	Absolute mV/V reading where 10000 = 1.0mV/V
Gross/Net Weight	0025_H	weight	A	N	Displayed weight in primary units
Gross Weight	0026_H	weight	A	N	Gross weight in primary units
Net Weight	0027_H	weight	A	N	Net weight in primary units
Tare Weight	0028_H	weight	A	N	Tare weight in primary units
Peak Hold	0029_H	weight	A	N	Peak hold weight in primary units
Secondary Gross Weight	$002C_H$	weight	A	N	Gross weight in secondary units
Raw ADC Counts	$002D_H$	int32	A	N	2,560,000 = 1.0mV/V
Secondary Net Weight	$002E_H$	weight	A	N	Net weight in secondary units
System Fullscale	$002F_H$	uint32	A	N	Full scale capacity
Traceable weight available flag	0030_H	uint8	A	N	0: No traceable weights since start up. 1: Traceable weight data is valid
Traceable ID	0031_H	uint32	A	N	The unique ID for the traceable weight
Traceable weight	0032_H	weight	A	N	Traceable weight in primary units
Traceable weight (secondary)	0033_H	weight	A	N	Traceable weight in secondary units
Traceable weight (p)	0034_H	int32	A	N	Traceble weight in pieces
Traceable tare weight	0035_H	int32	A	N	Tare weight valid during traceable weight
Traceable PT Flag	0036_H	uint8	A	N	0: no preset tare 1: preset tare
Traceable date: year	0037_H	uint8	A	N	Year that the traceable weight was acquired
Traceable date: month	0038_H	uint8	A	N	Month that the traceable weight was acquired
Traceable date: date	0039_H	uint8	A	N	Day that the traceable weight was acquired
Traceable date: hour	$003A_H$	uint8	A	N	Hour that the traceable weight was acquired
Traceable date: minute	$003B_H$	uint8	A	N	Minute that the traceable weight was acquired

Name	Address	Type	R	W	Description
Traceable date: second	$003C_H$	uint8	A	N	Second that the traceable weight was acquired
Stream Set 1 Data	0040_H	stream	A	A	Returns a block of data which is selected in Stream Register 1-5. Use a read command to read a single set of data. Use an execute command (with a parameter of 1) to switch on automatic transmition at the mode speed.
Stream Set 1 Mode	0041_H	uint8	A	A	0: Manual - read 'Stream Data' register 1: Auto Sync - Data is sent at sync freqency 2: Auto 10Hz - Data is sent at 10Hz 3: Auto 5Hz - Data is sent at 5Hz 4: Auto 1Hz - Data is sent at 1Hz
Stream Set 1: Register 0	0042_H	register	A	A	Register to stream
Stream Set 1: Register 1	0043_H	register	A	A	Register to stream
Stream Set 1: Register 2	0044_H	register	A	A	Register to stream
Stream Set 1: Register 3	0045_H	register	A	A	Register to stream
Stream Set 1: Register 4	0046_H	register	A	A	Register to stream
IO Status	0051_H	uint32	A	A	32 bit mask indicating the current state of all IO.
IO Mask	0054_H	uint32	A	A	32 bit masking for masking IO to be controlled manually. Bit = 0 for indicator control, bit = 1 for manual control.
Pulse count 1	0055_H	int32	A	N	Pulse count on IO 1
Pulse count 2	0056_H	int32	A	N	Pulse count on IO 2
Clear pulse counts	$005D_H$	execute	N	A	Clear the pulse counts, an argument of 0 clears pulse count 1, 1 clears pulse count 2 and any other value clears all pulse counts
Display Top Left	$00B0_H$	string	N	A	Write to the top left of the screen

Name	Address	Type	R	W	Description
Display Annunciators	$00B2_H$	uint32	N	A	Write the annunciators to display. Bit 0: Zero 1: Motion 2: Gross 3: Net 4: Zero band 5: Held 6: Spare 1 7: Range 1 8: Range 2 9: Range 3 10: Check under 2 11: Check under 1 12: Check pass 13: Check over 1 14: Check over 2 15: IO 1 16: IO 2 17: IO 3 18: IO 4 19: IO 5 20: IO 6 21: IO 7 22: IO 8 23: Spare 2 24: Spare 3
Display Units	$00B3_H$	uint8	N	A	Write the units to display. 0 for none 1 for g 2 for kg 3 for t 4 for lb 5 for oz 8 for spare 16 for all.
Auto Annunciators	$00B6_H$	uint8	N	A	Update the annunciators automatically. 0 for off, 1 for on.
Secondary Gross/Net Weight	$00C0_H$	weight	A	N	Displayed weight in secondary units
Secondary Tare Weight	$00C1_H$	weight	A	N	Tare weight in secondary units
Tertiary Gross Weight	$00C2_H$	weight	A	N	Gross weight in tertiary units
Tertiary Net Weight	$00C3_H$	weight	A	N	Net weight in tertiary units
Tertiary Gross/Net Weight	$00C4_H$	weight	A	N	Displayed weight in tertiary units
Tertiary Tare Weight	$00C5_H$	weight	A	N	Tare weight in tertiary units
Full passcode	$00D0_H$	uint32	F	F	Full passcode
Safe passcode	$00D1_H$	uint32	S	S	Safe passcode
Calibration weight	0100_H	int32	A	A	Weight used for span calibration or linearisation
Zero calibration	0102_H	execute	N	F	Calibrate the scale zero
Span calibration	0103_H	execute	N	F	Calibrate the scale span. Ensure that the calibration weight is set first.
Edit linearisation	0104_H	execute	N	F	Edit a linearisation point, takes the linearisation point as an argument. Ensure that the calibration weight is set first.

Name	Address	Type	R	W	Description
Clear linearisation	0105_H	execute	N	F	Clear a linearisation point, takes the linearisation point to be cleared as an argument
Direct zero calibration	0106_H	execute	N	F	Direct mV/V zero calibration, takes the mV/V value * 10000 at zero as an argument (1.0000 mV/V should be entered as 10000).
Direct span calibration	0107_H	execute	N	F	Direct mV/V span calibration, takes the mV/V value * 10000 at full scale relative to zero as an argument (1.0000 mV/V should be entered as 10000).
Zero mV/V value	0111_H	int32	A	N	mV/V value of the zero point * 10000.
Span weight	0112_H	int32	A	N	Weight used for the span calibration.
Span mV/V value	0113_H	int32	A	N	mV/V value of the span point relative to the zero point * 10000.
Linearisation correction weight	0114_H	execute	A	A	Execute with the linearisation point number (0 - 9) to get the correction weight at that linearisation point.
Linearisation percentage	0115_H	execute	A	A	Execute with the linearisation point number (0 - 9) to get the percentage value between zero and span for that linearisation point.
Number of decimal places	0128_H	uint8	A	F	Set the number of decimal places of the scale.
Scale units	0129_H	uint8	A	F	Values: 0 is none 1 is grams 2 is kilograms 3 is pounds 4 is tonnes 5 is ounces 6 is user.
Current Time/Date	0150_H	string	A	N	Read this register to get current instrument date/time (e.g. 10/12/2016 18:30:10).
Date Format	0151_H	uint8	A	A	Write 0 for DD.MM.YY 1 for DD.MM.YYYY 2 for MM.DD.YY 3 for MM.DD.YYYY 4 for YY.MM.DD 5 for YYYY.MM.DD
RTC Day	0152_H	uint8	A	A	Read/Write current day (1..31)
RTC Month	0153_H	uint8	A	A	Read/Write current month (1..12)
RTC Year	0154_H	uint16	A	A	Read/Write current year (2000..2099))
RTC Hour	0155_H	uint8	A	A	Read/Write current hour (0..23)
RTC Minute	0156_H	uint8	A	A	Read/Write current minute (0..59)
RTC Second	0157_H	uint8	A	A	Read/Write current second (0..59)

Name	Address	Type	R	W	Description
Zero scale	0300_H	execute	N	A	Zero the scale. Returns: 0 for success 1 for operation canceled 2 for operation in progress 3 for scale is in error 4 for scale is over or under loaded 5 for ADC already busy 6 for scale is in motion 7 for outside of allowed (zero, tare etc.) band 8 resolution too low 9 command not implemented A duplicate point B scale is in high resolution mode C print ID has reached its maximum value D current date and time could not be determined E parameter passed was not valid F operation not allowed in trade mode.
Tare scale	0301_H	execute	N	A	Tare the scale. Return value as described for the zero scale register.
Preset tare	0302_H	execute	N	A	Set preset tare value. Return value as described for the zero scale register.
Gross/Net	0303_H	execute	N	A	Set scale to gross or net. Argument values: 1 sets to Gross 2 sets to Net all other values will toggle. Return value as described for the zero scale register.
High Resolution	0304_H	execute	N	A	Set high resolution mode. Argument values: 1 sets to high resolution mode 2 sets normal resolution mode 3 sets resolution to DB setting all other values will toggle between high and normal resolution. Returns 1 for high resolution or 0 for normal resolution.
Lua Status 2	0305_H	uint32	A	N	32 bit number containing lua status bits as follows: 0: Hi-Res 1-2: Dispmode values: 0: Display primary weight 1: Display pieces 2: Display secondary weight 3: Display tertiary weight 3-4: Range: 0 (range 1), 1 (range 2), 2 (range 3) 5: Menu active 8: USB Partition Notification 25: RTC changed 26: Weight changed 27: IO changed

Name	Address	Type	R	W	Description
Primary units, dp, resolutions	0306_H	uint32	A	N	32 bit number containing units, decimal point location and resolution index for all three ranges. Bits are as follows: 0-3: Decimal point 4-7: Units 8-15: Resolution index 3 16-23: Resolution index 2 24-31: Resolution index 1 See E1 setting under Section 5.2 for resolution values.
Secondary units, dp, resolutions	0307_H	uint32	A	N	32 bit number containing units, decimal point location and resolution index for all three ranges. Bits are as follows: 0-3: Decimal point 4-7: Units 8-15: Resolution index 3 16-23: Resolution index 2 24-31: Resolution index 1 See E1 setting under Section 5.2 for resolution values.
Tertiary units, dp, resolutions	0308_H	uint32	A	N	32 bit number containing units, decimal point location and resolution index for all three ranges. Bits are as follows: 0-3: Decimal point 4-7: Units 8-15: Resolution index 3 16-23: Resolution index 2 24-31: Resolution index 1 See E1 setting under Section 5.2 for resolution values.
Comms Start	0309_H	execute	N	A	Dummy register. Executing does nothing.
Init function	$030B_H$	execute	N	A	Re-initialise after changing settings. Use an argument of 1 to re-init setpoints and IOs
Get Key	0321_H	uint8	A	N	Get any keys that have been pressed
Flush Keys	0322_H	execute	N	A	Flush any key presses that haven't been handled yet
Analogue value	0323_H	int32	A	A	Set the analogue out value directly, values are between 0 and 50000.
Set Key	0324_H	uint8	N	A	Send key presses back to the main application. See Keyboard Buffer (0x0008) register for keycodes.
Key handler	0325_H	uint8	A	A	Set whether the application (0) or Lua (1) handles key presses
Buzz Length	0327_H	uint8	A	A	Set then length of the beep: 0 for short 1 for medium 2 for long
Buzz Number	0328_H	uint8	A	A	Set how many beeps to do, buzzer will start beeping after write, value will be cleared

Name	Address	Type	R	W	Description
Lua Status	0329_H	uint32	A	N	32 bit number containing lua status bits as follows: 0: Net 1: Gross 2: Zero Band 3: Not Zero Band 4: Centre-of-zero 5: Not Centre-of-zero 6: Motion 7: Stable 8: Lo-range 9: Hi-range 10: Preset Tare 11: No Preset Tare 12: Error 13: Underload 14: Overload 15: No Error 16: Held 17: Not Held
Lua status 2 RTC enable	$032A_H$	uint8	A	A	Set whether RTC in lua status 2 will be used, 0 for no RTC, 1 for RTC
Lua status 2 weight enable	$032B_H$	uint8	A	A	Set whether weight in lua status 2 will be used, 0 for off, any non-zero sets the number of readings between flag being set
Lua status 2 IO enable	$032C_H$	uint32	A	A	Set whether IO in lua status 2 will be used, 0 for off, 32 bit mask for IOs of interest
Reg instrument begin	$032D_H$	execute	N	A	Stop all streaming, clear the lua display, flush the keys, and take control of event, printer, and serial devices.
Setpoint Status	$032E_H$	uint32	A	N	32 bit number. Lower 16 bits are setpoint status, upper 16 bits are inverted from the lower 16.
Heartbeat	$032F_H$	uint8	A	A	Timeout (in seconds, accurate to $\pm$ 1 seconds) until the display and keyboard is handed back to the C500. Executing this register cleans up immediately.
Lua Stream Set 1 Data	0340_H	stream	A	A	Returns a block of data which is selected in Stream Register 1-5. Use a read command to read a single set of data. Use an execute command (with a parameter of 1) to switch on automatic transmition at the mode speed.
Lua Stream Set 1 Mode	0341_H	uint8	A	A	0: Manual - read 'Stream Data' register 1: Auto Sync - Data is sent at sync freqency 2: Auto 10Hz - Data is sent at 10Hz 3: Auto 5Hz - Data is sent at 5Hz 4: Auto 1Hz - Data is sent at 1Hz
Lua Stream Set 1: Register 0	0342_H	register	A	A	Register to stream
Lua Stream Set 1: Register 1	0343_H	register	A	A	Register to stream
Lua Stream Set 1: Register 2	0344_H	register	A	A	Register to stream

Name	Address	Type	R	W	Description
Lua Stream Set 1: Register 3	0345_H	register	A	A	Register to stream
Lua Stream Set 1: Register 4	0346_H	register	A	A	Register to stream
Lua Stream Set 2 Data	0350_H	stream	A	A	Returns a block of data which is selected in Stream Register 1-5. Use a read command to read a single set of data. Use an execute command (with a parameter of 1) to switch on automatic transmition at the mode speed.
Lua Stream Set 2 Mode	0351_H	uint8	A	A	0: Manual - read 'Stream Data' register 1: Auto Sync - Data is sent at sync freqency 2: Auto 10Hz - Data is sent at 10Hz 3: Auto 5Hz - Data is sent at 5Hz 4: Auto 1Hz - Data is sent at 1Hz
Lua Stream Set 2: Register 0	0352_H	register	A	A	Register to stream
Lua Stream Set 2: Register 1	0353_H	register	A	A	Register to stream
Lua Stream Set 2: Register 2	0354_H	register	A	A	Register to stream
Lua Stream Set 2: Register 3	0355_H	register	A	A	Register to stream
Lua Stream Set 2: Register 4	0356_H	register	A	A	Register to stream
Eject USB	0370_H	execute	N	A	Execute register to safely eject USB
USB Partition Notification	0371_H	string	A	N	When the Luastatus2 USB Notification bit is set high, this should be read to determine what USB partition event has occurred.
Scale type	1120_H	uint8	A	F	Values: 0 is single range 1 is dual interval 2 is dual range 3 is triple interval 4 is triple range.
DHCP Enable	8400_H	uint8	A	A	DHCP Enable / Disable
IP Address	8401_H	ipaddress	A	A	IP Address
Subnet Mask	8402_H	ipaddress	A	A	Subnet Mask
Default Gateway	8403_H	ipaddress	A	A	Default Gateway
DNS Server 1	8404_H	ipaddress	A	A	DNS Server 1
DNS Server 2	8405_H	ipaddress	A	A	DNS Server 2
Network Default	8406_H	execute	N	A	Force network settings to defaults
MAC Address	8408_H	string	A	N	MAC Address
Hostname	8409_H	string	A	A	Hostname
Search Domain	$840A_H$	string	A	A	Search Domain
DNS Server 3	$840B_H$	ipaddress	A	A	DNS Server 3
Autoout Format	$A203_H$	uint8	A	S	Automatic output format
Autoout Source	$A204_H$	uint8	A	S	Automatic output data source
Autoout Custom Format	$A205_H$	string	A	S	Automatic output custom format

Name	Address	Type	R	W	Description
Setpoint 1 type	$\texttt{A401}_{\text{H}}$	uint8	A	A	Set setpoint 1 type: 0 is off 1 is on 2 is over 3 is under 4 is centre of zero 5 is zero band 6 is net 7 is motion 8 is error 9 is buzzer 10 is weigh out 11 is weigh in.
Setpoint 1 lock	$\texttt{A402}_{\text{H}}$	uint8	A	A	Set setpoint 1 lock: 0 is off 1 is on.
Setpoint 1 logic	$\texttt{A403}_{\text{H}}$	uint8	A	A	Set setpoint 1 logic: 0 is high 1 is low.
Setpoint 1 alarm	$\texttt{A404}_{\text{H}}$	uint8	A	A	Set setpoint 1 alarm: 0 is none 1 is single beep 2 is double beep 3 is continuous beeping 4 is flash the display.
Setpoint 1 source	$\texttt{A406}_{\text{H}}$	uint8	A	A	Set setpoint 1 source: 0 is gross weight 1 is net weight.
Setpoint 1 Target	$\texttt{A408}_{\text{H}}$	int32	A	A	Set setpoint 1 target.
Setpoint 1 Hysteresis	$\texttt{A409}_{\text{H}}$	int32	A	A	Set setpoint 1 hysteresis.
Setpoint 1 in flight	$\texttt{A40A}_{\text{H}}$	int32	A	A	Set setpoint 1 in flight.
Setpoint Jog Off	$\texttt{A40C}_{\text{H}}$	uint32	A	A	Duration of setpoint jog off time in ms.
Setpoint Jog On	$\texttt{A40D}_{\text{H}}$	uint32	A	A	Duration of setpoint jog on time in ms.
Setpoint 1 Timing	$\texttt{A410}_{\text{H}}$	uint8	A	A	Setpoint 1 timing: 0 is level 1 is edge 2 is latch 3 is pulse.
Setpoint 1 Reset	$\texttt{A411}_{\text{H}}$	uint8	A	A	Setpoint 1 rest: 0 is none 1 is reset A 2 is reset B.
Setpoint Jog Set	$\texttt{A412}_{\text{H}}$	uint8	A	A	Number of jogs in a set.
Setpoint Max Jog Set	$\texttt{A413}_{\text{H}}$	uint8	A	A	Maximum number of jog sets. Set to 0 for unlimited sets.

Name	Address	Type	R	W	Description
Setpoint 2 type	$\texttt{A421}_{\text{H}}$	uint8	A	A	Set setpoint 2 type: 0 is off 1 is on 2 is over 3 is under 4 is centre of zero 5 is zero band 6 is net 7 is motion 8 is error 9 is buzzer 10 is weigh out 11 is weigh in.
Setpoint 2 lock	$\texttt{A422}_{\text{H}}$	uint8	A	A	Set setpoint 2 lock: 0 is off 1 is on.
Setpoint 2 logic	$\texttt{A423}_{\text{H}}$	uint8	A	A	Set setpoint 2 logic: 0 is high 1 is low.
Setpoint 2 alarm	$\texttt{A424}_{\text{H}}$	uint8	A	A	Set setpoint 2 alarm: 0 is none 1 is single beep 2 is double beep 3 is continuous beeping 4 is flash the display.
Setpoint 2 source	$\texttt{A426}_{\text{H}}$	uint8	A	A	Set setpoint 2 source: 0 is gross weight 1 is net weight.
Setpoint 2 Target	$\texttt{A428}_{\text{H}}$	int32	A	A	Set setpoint 2 target.
Setpoint 2 Hysteresis	$\texttt{A429}_{\text{H}}$	int32	A	A	Set setpoint 2 hysteresis.
Setpoint 2 in flight	$\texttt{A42A}_{\text{H}}$	int32	A	A	Set setpoint 2 in flight.
Setpoint 2 Timing	$\texttt{A430}_{\text{H}}$	uint8	A	A	Setpoint 2 timing: 0 is level 1 is edge 2 is latch 3 is pulse.
Setpoint 2 Reset	$\texttt{A431}_{\text{H}}$	uint8	A	A	Setpoint 2 rest: 0 is none 1 is reset A 2 is reset B.
Setpoint 3 type	$\texttt{A441}_{\text{H}}$	uint8	A	A	Set setpoint 3 type: 0 is off 1 is on 2 is over 3 is under 4 is centre of zero 5 is zero band 6 is net 7 is motion 8 is error 9 is buzzer 10 is weigh out 11 is weigh in.

Name	Address	Type	R	W	Description
Setpoint 3 lock	$\texttt{A442}_{\text{H}}$	uint8	A	A	Set setpoint 3 lock: 0 is off 1 is on.
Setpoint 3 logic	$\texttt{A443}_{\text{H}}$	uint8	A	A	Set setpoint 3 logic: 0 is high 1 is low.
Setpoint 3 alarm	$\texttt{A444}_{\text{H}}$	uint8	A	A	Set setpoint 3 alarm: 0 is none 1 is single beep 2 is double beep 3 is continuous beeping 4 is flash the display.
Setpoint 3 source	$\texttt{A446}_{\text{H}}$	uint8	A	A	Set setpoint 3 source: 0 is gross weight 1 is net weight.
Setpoint 3 Target	$\texttt{A448}_{\text{H}}$	int32	A	A	Set setpoint 3 target.
Setpoint 3 Hysteresis	$\texttt{A449}_{\text{H}}$	int32	A	A	Set setpoint 3 hysteresis.
Setpoint 3 in flight	$\texttt{A44A}_{\text{H}}$	int32	A	A	Set setpoint 3 in flight.
Setpoint 3 Timing	$\texttt{A450}_{\text{H}}$	uint8	A	A	Setpoint 3 timing: 0 is level 1 is edge 2 is latch 3 is pulse.
Setpoint 3 Reset	$\texttt{A451}_{\text{H}}$	uint8	A	A	Setpoint 3 rest: 0 is none 1 is reset A 2 is reset B.
Setpoint 4 type	$\texttt{A461}_{\text{H}}$	uint8	A	A	Set setpoint 4 type: 0 is off 1 is on 2 is over 3 is under 4 is centre of zero 5 is zero band 6 is net 7 is motion 8 is error 9 is buzzer 10 is weigh out 11 is weigh in.
Setpoint 4 lock	$\texttt{A462}_{\text{H}}$	uint8	A	A	Set setpoint 4 lock: 0 is off 1 is on.
Setpoint 4 logic	$\texttt{A463}_{\text{H}}$	uint8	A	A	Set setpoint 4 logic: 0 is high 1 is low.
Setpoint 4 alarm	$\texttt{A464}_{\text{H}}$	uint8	A	A	Set setpoint 4 alarm: 0 is none 1 is single beep 2 is double beep 3 is continuous beeping 4 is flash the display.

Name	Address	Type	R	W	Description
Setpoint 4 source	$A466_H$	uint8	A	A	Set setpoint 4 source: 0 is gross weight 1 is net weight.
Setpoint 4 Target	$A468_H$	int32	A	A	Set setpoint 4 target.
Setpoint 4 Hysteresis	$A469_H$	int32	A	A	Set setpoint 4 hysteresis.
Setpoint 4 in flight	$A46A_H$	int32	A	A	Set setpoint 4 in flight.
Setpoint 4 Timing	$A470_H$	uint8	A	A	Setpoint 4 timing: 0 is level 1 is edge 2 is latch 3 is pulse.
Setpoint 4 Reset	$A471_H$	uint8	A	A	Setpoint 4 rest: 0 is none 1 is reset A 2 is reset B.
Setpoint 5 type	$A481_H$	uint8	A	A	Set setpoint 5 type: 0 is off 1 is on 2 is over 3 is under 4 is centre of zero 5 is zero band 6 is net 7 is motion 8 is error 9 is buzzer 10 is weigh out 11 is weigh in.
Setpoint 5 lock	$A482_H$	uint8	A	A	Set setpoint 5 lock: 0 is off 1 is on.
Setpoint 5 logic	$A483_H$	uint8	A	A	Set setpoint 5 logic: 0 is high 1 is low.
Setpoint 5 alarm	$A484_H$	uint8	A	A	Set setpoint 5 alarm: 0 is none 1 is single beep 2 is double beep 3 is continuous beeping 4 is flash the display.
Setpoint 5 source	$A486_H$	uint8	A	A	Set setpoint 5 source: 0 is gross weight 1 is net weight.
Setpoint 5 Target	$A488_H$	int32	A	A	Set setpoint 5 target.
Setpoint 5 Hysteresis	$A489_H$	int32	A	A	Set setpoint 5 hysteresis.
Setpoint 5 in flight	$A48A_H$	int32	A	A	Set setpoint 5 in flight.
Setpoint 5 Timing	$A490_H$	uint8	A	A	Setpoint 5 timing: 0 is level 1 is edge 2 is latch 3 is pulse.

Name	Address	Type	R	W	Description
Setpoint 5 Reset	$\texttt{A491}_{\text{H}}$	uint8	A	A	Setpoint 5 rest: 0 is none 1 is reset A 2 is reset B.
Setpoint 6 type	$\texttt{A4A1}_{\text{H}}$	uint8	A	A	Set setpoint 6 type: 0 is off 1 is on 2 is over 3 is under 4 is centre of zero 5 is zero band 6 is net 7 is motion 8 is error 9 is buzzer 10 is weigh out 11 is weigh in.
Setpoint 6 lock	$\texttt{A4A2}_{\text{H}}$	uint8	A	A	Set setpoint 6 lock: 0 is off 1 is on.
Setpoint 6 logic	$\texttt{A4A3}_{\text{H}}$	uint8	A	A	Set setpoint 6 logic: 0 is high 1 is low.
Setpoint 6 alarm	$\texttt{A4A4}_{\text{H}}$	uint8	A	A	Set setpoint 6 alarm: 0 is none 1 is single beep 2 is double beep 3 is continuous beeping 4 is flash the display.
Setpoint 6 source	$\texttt{A4A6}_{\text{H}}$	uint8	A	A	Set setpoint 6 source: 0 is gross weight 1 is net weight.
Setpoint 6 Target	$\texttt{A4A8}_{\text{H}}$	int32	A	A	Set setpoint 6 target.
Setpoint 6 Hysteresis	$\texttt{A4A9}_{\text{H}}$	int32	A	A	Set setpoint 6 hysteresis.
Setpoint 6 in flight	$\texttt{A4AA}_{\text{H}}$	int32	A	A	Set setpoint 6 in flight.
Setpoint 6 Timing	$\texttt{A4B0}_{\text{H}}$	uint8	A	A	Setpoint 6 timing: 0 is level 1 is edge 2 is latch 3 is pulse.
Setpoint 6 Reset	$\texttt{A4B1}_{\text{H}}$	uint8	A	A	Setpoint 6 rest: 0 is none 1 is reset A 2 is reset B.

Name	Address	Type	R	W	Description
Setpoint 7 type	$\texttt{A4C1}_{\text{H}}$	uint8	A	A	Set setpoint 7 type: 0 is off 1 is on 2 is over 3 is under 4 is centre of zero 5 is zero band 6 is net 7 is motion 8 is error 9 is buzzer 10 is weigh out 11 is weigh in.
Setpoint 7 lock	$\texttt{A4C2}_{\text{H}}$	uint8	A	A	Set setpoint 7 lock: 0 is off 1 is on.
Setpoint 7 logic	$\texttt{A4C3}_{\text{H}}$	uint8	A	A	Set setpoint 7 logic: 0 is high 1 is low.
Setpoint 7 alarm	$\texttt{A4C4}_{\text{H}}$	uint8	A	A	Set setpoint 7 alarm: 0 is none 1 is single beep 2 is double beep 3 is continuous beeping 4 is flash the display.
Setpoint 7 source	$\texttt{A4C6}_{\text{H}}$	uint8	A	A	Set setpoint 7 source: 0 is gross weight 1 is net weight.
Setpoint 7 Target	$\texttt{A4C8}_{\text{H}}$	int32	A	A	Set setpoint 7 target.
Setpoint 7 Hysteresis	$\texttt{A4C9}_{\text{H}}$	int32	A	A	Set setpoint 7 hysteresis.
Setpoint 7 in flight	$\texttt{A4CA}_{\text{H}}$	int32	A	A	Set setpoint 7 in flight.
Setpoint 7 Timing	$\texttt{A4D0}_{\text{H}}$	uint8	A	A	Setpoint 7 timing: 0 is level 1 is edge 2 is latch 3 is pulse.
Setpoint 7 Reset	$\texttt{A4D1}_{\text{H}}$	uint8	A	A	Setpoint 7 rest: 0 is none 1 is reset A 2 is reset B.
Setpoint 8 type	$\texttt{A4E1}_{\text{H}}$	uint8	A	A	Set setpoint 8 type: 0 is off 1 is on 2 is over 3 is under 4 is centre of zero 5 is zero band 6 is net 7 is motion 8 is error 9 is buzzer 10 is weigh out 11 is weigh in.

Name	Address	Type	R	W	Description
Setpoint 8 lock	$A4E2_H$	uint8	A	A	Set setpoint 8 lock: 0 is off 1 is on.
Setpoint 8 logic	$A4E3_H$	uint8	A	A	Set setpoint 8 logic: 0 is high 1 is low.
Setpoint 8 alarm	$A4E4_H$	uint8	A	A	Set setpoint 8 alarm: 0 is none 1 is single beep 2 is double beep 3 is continuous beeping 4 is flash the display.
Setpoint 8 source	$A4E6_H$	uint8	A	A	Set setpoint 8 source: 0 is gross weight 1 is net weight.
Setpoint 8 Target	$A4E8_H$	int32	A	A	Set setpoint 8 target.
Setpoint 8 Hysteresis	$A4E9_H$	int32	A	A	Set setpoint 8 hysteresis.
Setpoint 8 in flight	$A4EA_H$	int32	A	A	Set setpoint 8 in flight.
Setpoint 8 Timing	$A4F0_H$	uint8	A	A	Setpoint 8 timing: 0 is level 1 is edge 2 is latch 3 is pulse.
Setpoint 8 Reset	$A4F1_H$	uint8	A	A	Setpoint 8 rest: 0 is none 1 is reset A 2 is reset B.
Analogue output type	$A801_H$	uint8	A	A	Set the analogue output type: 0 is current 1 is voltage.
Analogue output absolute	$A803_H$	uint8	A	A	Set if analogue output follows the actual weight value or the absolute value: 0 is actual value 1 is absolute value.
Analogue output source	$A805_H$	uint8	A	A	Set the weight value that analogue output follows: 0 is gross weight 1 is net weight 2 is displayed weight 3 is comms (see register 0323).
Analogue output clip	$A806_H$	uint8	A	A	Clip the analogue output to 0-100%: 0 is off 1 is on.

19 Modbus

The C500 supports a fully featured Modbus implementation which exposes most legacy serial communication commands (see Section 21.2) as Modbus registers.

The following Modbus protocols are supported:

- Modbus ASCII: Available on the USB host serial port. This requires a user supplied USB serial cable (see Section 10.2). MB.ASCI must be selected in `SERIAL:SER.HST:TYPE` (see Section 5.5). In addition baud rate and bit settings may need to be configured.
- Modbus RTU: Available on the USB host serial port. This requires a user supplied USB serial cable (see Section 10.2). MB.RTU must be selected in `SERIAL:SER.HST:TYPE` (see Section 5.5). In addition baud rate and bit settings may need to be configured.
- Modbus TCP: Available on the Ethernet port. The TCP port listening for connections can be configured via `SERIAL:MOD.OPT:PORT` (see Section 5.5).

For legacy 5000 replacement installations using Modbus, please refer to the legacy modbus documentation (see Section 22).

The details of underlying communications protocol are generally looked after by the PLC directly. Please refer to the offical Modbus documentation available from http://modbus.org/specs.php for more information on the protocol.

Registers listed below refer to the Modbus Data Model (MDM), which offsets each register by one in relation to the Protocol Data Unit (PDU). Most PLCs will use MDM to specify registers, whilst PDU is used for the underlying data transfer. For an explanation of these terms, please refer to the Modbus protocol documentation available via the URL above.

Warning!

Be sure to correctly configure the indicator address in `SERIAL:NET.OPT:ADDRES`. The indicator will not reply to messages that do not match its address.

19.1 Register Sizes

When accessing parameters that span more than a single Modbus register (16bits), the read or write MUST consist of sufficient Modbus registers to accomodate the data. If the read or write does not contain enough registers, the indicator will return a Modbus error.

- When reading an SINT32 or UINT32, the read or write must consist of two 16bit Modbus registers. For example, when reading the displayed weight (register 6201), two Modbus registers must be read (6201 and 6202).
- When reading a STR(size), the read or write must consist of ((size + 1) / 2) Modbus registers. For example, when reading the unit ID (register 6101), ((15 + 1) / 2) = 8 Modbus registers must be read (6101 to 6109). The first 15 bytes will be the register data, and the last byte will be zero. When writing a string register, unused bytes should be set to zero. In the register table, strings are given a length in bytes in the following way: STR(bytes).

It is possible to read more than one sequential registers in a single read or write.

19.2 Modbus Holding Registers

The following Modbus functions are implemented for holding registers:

- Read holding registers (function code 0x03)
- Write single register (function code 0x06)

- Write multiple registers (function code 0x10)

The following table lists the translation for each Modbus holding register into the corresponding serial communication command:

Register	Name	Type	Write	Read	Read Response	Description
1001	AVERAGE	UINT16	ASF<value>;	ASF?;	<value>	Averaging
1002	JITTER	UINT16	ASF,<value>;	ASF?;	,<value>	Anti jitter
1003	UNITS	UINT16	ENU<value>;	ENU?;	<value>	Units of measurement
1004	DECIMAL_POINT	UINT16	IAD,,<value>;	IAD?;	,,<value>	Decimal point position
1005	HI_RES	UINT16	IAD,,,,<value>;	IAD?;	,,,,<value>	High resolution (x10 mode)
1006	ADD_TARE	UINT32	IAD,,,,,<value>;	IAD?;	,,,,,<value>	Additive tare
1008	SENSE_CHECK	UINT16	IAD,,,,,,<value>;	IAD?;	,,,,,,<value>	Sense check
1009	SYNC_FREQ	UINT16	ICR<value>;	ICR?;	<value>	Sync frequency
1010	MOTION	UINT16	MTD<value>;	MTD?;	<value>	Motion setting
1011	BUILD_TYPE	UINT16	WMD<value>;	WMD?;	<value>	Build type (single/dual range/etc)
1012	APPROVAL_MODE	UINT16	WMD,<value>;	WMD?;	,<value>	Approval mode (trade/industrial)
1013	ZERO_ON_START	UINT16	ZST<value>;	ZST?;	<value>	Zero on start
1014	ZERO_TRACKING	UINT16	ZST,<value>;	ZST?;	,<value>	Zero tracking
1015	ZERO_RANGE	UINT16	ZST,,<value>;	ZST?;	,,<value>	Range of zero
1016	ZERO_BAND	SINT32	ZST,,,<value>;	ZST?;	,,,<value>	Zero band
1021	CAL_WEIGHT	UINT32	CWT<value>;	CWT?;	<value>	Calibration weight
1023	CAL_ZERO_NORMAL	UINT16	LDN;			Perform zero calibration (normal)
1024	CAL_ZERO_DIRECT	SINT32	LDN<value>;	LDN?;	<value>	Perform zero calibration (direct mv/v)
1026	CAL_ZERO_STATUS	UINT16		LDN?;	,<value>	Zero calibration status
1027	CAL_SPAN_NORMAL	UINT16	LWN;			Perform span calibration (normal)
1028	CAL_SPAN_DIRECT	SINT32	LWN<value>;	LWN?;	<value>	Perform span calibration (direct mv/v)
1030	CAL_SPAN_STATUS	UINT16		LWN?;	,<value>	Span calibration status
1031	FULLSCALE1	UINT32	IAD1,<value>;	IAD?1;	1,<value>	Range 1 full scale
1033	RESOLUTION1	UINT16	IAD1,,,<value>;	IAD?1;	1,,,<value>	Range 1 resolution
1034	FULLSCALE2	UINT32	IAD2,<value>;	IAD?2;	2,<value>	Range 2 full scale
1036	RESOLUTION2	UINT16	IAD2,,,<value>;	IAD?2;	2,,,<value>	Range 2 resolution
1041	SET_LIN1	UINT32	LIC1,<value>;			Set linearisation point 1
1043	SET_LIN2	UINT32	LIC2,<value>;			Set linearisation point 2
1045	SET_LIN3	UINT32	LIC3,<value>;			Set linearisation point 3
1047	SET_LIN4	UINT32	LIC4,<value>;			Set linearisation point 4
1049	SET_LIN5	UINT32	LIC5,<value>;			Set linearisation point 5
1051	CLR_LIN1	UINT16	LIC1;			Clear linearisation point 1 (write any value will clear)
1052	CLR_LIN2	UINT16	LIC2;			Clear linearisation point 2 (write any value will clear)
1053	CLR_LIN3	UINT16	LIC3;			Clear linearisation point 3 (write any value will clear)
1054	CLR_LIN4	UINT16	LIC4;			Clear linearisation point 4 (write any value will clear)
1055	CLR_LIN5	UINT16	LIC5;			Clear linearisation point 5 (write any value will clear)
1061	LIN1_PERCENT	UINT16		LIC?1;	<value>	Linearisation point 1 percent of fullscale value
1062	LIN1_CORRECTION	SINT32		LIC?1;	,<value>	Linearisation point 1 correction value
1064	LIN2_PERCENT	UINT16		LIC?2;	<value>	Linearisation point 2 percent of fullscale value
1065	LIN2_CORRECTION	SINT32		LIC?2;	,<value>	Linearisation point 2 correction value
1067	LIN3_PERCENT	UINT16		LIC?3;	<value>	Linearisation point 3 percent of fullscale value
1068	LIN3_CORRECTION	SINT32		LIC?3;	,<value>	Linearisation point 3 correction value
1070	LIN4_PERCENT	UINT16		LIC?4;	<value>	Linearisation point 4 percent of fullscale value
1071	LIN4_CORRECTION	SINT32		LIC?4;	,<value>	Linearisation point 4 correction value
1073	LIN5_PERCENT	UINT16		LIC?5;	<value>	Linearisation point 5 percent of fullscale value
1074	LIN5_CORRECTION	SINT32		LIC?5;	,<value>	Linearisation point 5 correction value
1101	ANALOG_OUT_TYPE	UINT16	AOC<value>;	AOC?;	<value>	Analog output type
1102	ANALOG_OUT_SOURCE	UINT16	AOC,<value>;	AOC?;	,<value>	Analog output source
1103	ANALOG_OUT_ZERO_ADJ	SINT32	AOC,,<value>;	AOC?;	,,<value>	Analog output zero adjustment
1105	ANALOG_OUT_SPAN_ADJ	SINT32	AOC,,,<value>;	AOC?;	,,,<value>	Analog output span adjustment
1107	ANALOG_OUT_FORCE	UINT16	AOC,,,,<value>;	AOC?;	,,,,<value>	Analog output force output

Register	Name	Type	Write	Read	Read Response	Description
1108	ANALOG_OUT_ABSOLUTE	UINT16	AOC,,,,,<value>;	AOC?;	,,,,,<value>	Analog output absolute
1109	ANALOG_OUT_CLIP	UINT16	AOC,,,,,,<value>;	AOC?;	,,,,,,<value>	Analog output clip
2001	ADDRESS	UINT16	ADR<value>;	ADR?;	<value>	Device network address
2002	PRINT_MODE	UINT16	PRS,<value>;	PRS?;	,<value>	Printer mode
2003	AUTO_OUT_FMT	UINT16	PRS,,<value>;	PRS?;	,,<value>	Automatic output format
2004	PRINT_COLUMNS	UINT16	PRS,,,<value>;	PRS?;	,,,<value>	Printer ticket columns
2005	PRINT_ROWS	UINT16	PRS,,,,<value>;	PRS?;	,,,,<value>	Printer ticket rows
2006	AUTO_OUT_SOURCE	UINT16	PRS,,,,,<value>;	PRS?;	,,,,,<value>	Automatic output source
2101	SER0_M1A_TYPE	UINT16	BDX0,<value>;	BDX?0;	0,<value>	Serial module 1A port type
2102	SER0_M1A_BAUD	UINT16	BDX0,,<value>;	BDX?0;	0,,<value>	Serial module 1A port baud
2103	SER0_M1A_PARITYBITS	UINT16	BDX0,,,<value>;	BDX?0;	0,,,<value>	Serial module 1A port parity bits
2104	SER0_M1A_DATABITS	UINT16	BDX0,,,,<value>;	BDX?0;	0,,,,<value>	Serial module 1A port data bits
2105	SER0_M1A_STOPBITS	UINT16	BDX0,,,,,<value>;	BDX?0;	0,,,,,<value>	Serial module 1A port stop bits
2106	SER0_M1A_TERMINATIONS	UINT16	BDX0,,,,,,<value>;	BDX?0;	0,,,,,,<value>	Serial module 1A port use terminations
2107	SER0_M1A_CTS	UINT16	BDX0,,,,,,,<value>;	BDX?0;	0,,,,,,,<value>	Serial module 1A port use CTS
2111	SER1_M1B_TYPE	UINT16	BDX1,<value>;	BDX?1;	1,<value>	Serial module 1B port type
2116	SER1_M1B_TERMINATIONS	UINT16	BDX1,,,,,,<value>;	BDX?1;	1,,,,,,<value>	Serial module 1B port use terminations
2117	SER1_M1B_CTS	UINT16	BDX1,,,,,,,<value>;	BDX?1;	1,,,,,,,<value>	Serial module 1B port use CTS
2121	SER2_M2A_TYPE	UINT16	BDX2,<value>;	BDX?2;	2,<value>	Serial module 2A port type
2122	SER2_M2A_BAUD	UINT16	BDX2,,<value>;	BDX?2;	2,,<value>	Serial module 2A port baud
2123	SER2_M2A_PARITYBITS	UINT16	BDX2,,,<value>;	BDX?2;	2,,,<value>	Serial module 2A port parity bits
2124	SER2_M2A_DATABITS	UINT16	BDX2,,,,<value>;	BDX?2;	2,,,,<value>	Serial module 2A port data bits
2125	SER2_M2A_STOPBITS	UINT16	BDX2,,,,,<value>;	BDX?2;	2,,,,,<value>	Serial module 2A port stop bits
2126	SER2_M2A_TERMINATIONS	UINT16	BDX2,,,,,,<value>;	BDX?2;	2,,,,,,<value>	Serial module 2A port use terminations
2127	SER2_M2A_CTS	UINT16	BDX2,,,,,,,<value>;	BDX?2;	2,,,,,,,<value>	Serial module 2A port use CTS
2131	SER3_M2B_TYPE	UINT16	BDX3,<value>;	BDX?3;	3,<value>	Serial module 2B port type
2136	SER3_M2B_TERMINATIONS	UINT16	BDX3,,,,,,<value>;	BDX?3;	3,,,,,,<value>	Serial module 2B port use terminations
2137	SER3_M2B_CTS	UINT16	BDX3,,,,,,,<value>;	BDX?3;	3,,,,,,,<value>	Serial module 2B port use CTS
2141	SER4_TCPPORT	UINT16	BDX4,,,,,,,,,<value>;	BDX?4;	4,,,,,,,,,<value>	Serial network TCP listen port
2151	SER5_TCPPORT	UINT16	BDX5,,,,,,,,,<value>;	BDX?5;	5,,,,,,,,,<value>	Serial auto output TCP listen port
2161	SER6_USBSLAVE_TYPE	UINT16	BDX6,<value>;	BDX?6;	6,<value>	Serial USB slave port type
2171	SER7_USBHOST_TYPE	UINT16	BDX7,<value>;	BDX?7;	7,<value>	Serial USB host port type
2172	SER7_USBHOST_BAUD	UINT16	BDX7,,<value>;	BDX?7;	7,,<value>	Serial USB host port baud
2173	SER7_USBHOST_PARITYBITS	UINT16	BDX7,,,<value>;	BDX?7;	7,,,<value>	Serial USB host port parity bits
2174	SER7_USBHOST_DATABITS	UINT16	BDX7,,,,<value>;	BDX?7;	7,,,,<value>	Serial USB host port data bits
2175	SER7_USBHOST_STOPBITS	UINT16	BDX7,,,,,<value>;	BDX?7;	7,,,,,<value>	Serial USB host port stop bits
2177	SER7_USBHOST_CTS	UINT16	BDX7,,,,,,,<value>;	BDX?7;	7,,,,,,,<value>	Serial USB host port use CTS
2191	MODBUS_ENDIAN	UINT16	MBS<value>;	MBS?;	<value>	Modbus endian
2192	MODBUS_TCP_PORT	UINT16	MBS,<value>;	MBS?;	,<value>	Modbus TCP port
2201	AUTO_OUT_FMT_CUSTOM	STR(246)	AFT<value>;	AFT?;	<value>	Auto output format
2401	DHCP_ENABLE	UINT16	ETH<value>;	ETH?;	<value>	Ethernet DHCP enable
2402	IP_ADDR	STR(15)	ETH,<value>;	ETH?;	,<value>	Ethernet IP address
2410	NETMASK	STR(15)	ETH,,<value>;	ETH?;	,,<value>	Ethernet network mask
2418	DEFGW	STR(15)	ETH,,,<value>;	ETH?;	,,,<value>	Ethernet default gateway
2426	DNS	STR(15)	ETH,,,,<value>;	ETH?;	,,,,<value>	Ethernet DNS server
2434	MAC_ADDR	STR(17)		ETH?;	,,,,,,<value>	Ethernet MAC address
2443	HOSTNAME	STR(64)	ETH,,,,,,,<value>;	ETH?;	,,,,,,,<value>	Ethernet hostname
2501	PRINT_FORMAT	STR(246)	PFT<value>;	PFT?;	<value>	Printer format
2701	PRINT_HDR1	STR(30)	PST1,<value>;	PST?1;	<value>	Printer header line 1
2716	PRINT_HDR2	STR(30)	PST2,<value>;	PST?2;	<value>	Printer header line 2
2731	PRINT_HDR3	STR(30)	PST3,<value>;	PST?3;	<value>	Printer header line 3
2746	PRINT_HDR4	STR(30)	PST4,<value>;	PST?4;	<value>	Printer header line 4
2761	PRINT_HDR5	STR(30)	PST5,<value>;	PST?5;	<value>	Printer header line 5
2776	PRINT_HDR6	STR(30)	PST6,<value>;	PST?6;	<value>	Printer header line 6
2801	TEXT_STR1	STR(40)	TXT1,<value>;	TXT?1;	1,<value>	Text string 1 (thumbwheel)
2821	TEXT_STR2	STR(40)	TXT2,<value>;	TXT?2;	2,<value>	Text string 2 (thumbwheel)
2841	TEXT_STR3	STR(40)	TXT3,<value>;	TXT?3;	3,<value>	Text string 3 (thumbwheel)
2861	TEXT_STR4	STR(40)	TXT4,<value>;	TXT?4;	4,<value>	Text string 4 (thumbwheel)
2881	TEXT_STR5	STR(40)	TXT5,<value>;	TXT?5;	5,<value>	Text string 5 (thumbwheel)
2901	TEXT_STR6	STR(40)	TXT6,<value>;	TXT?6;	6,<value>	Text string 6 (thumbwheel)
2921	TEXT_STR7	STR(40)	TXT7,<value>;	TXT?7;	7,<value>	Text string 7 (thumbwheel)
2941	TEXT_STR8	STR(40)	TXT8,<value>;	TXT?8;	8,<value>	Text string 8 (thumbwheel)

Register	Name	Type	Write	Read	Read Response	Description
2961	TEXT_STR9	STR(40)	TXT9,<value>;	TXT?9;	9,<value>	Text string 9 (thumbwheel)
3001	ENABLE	UINT16	BUZ<value>;	BUZ?;	<value>	Buzzer enable
3002	MSV_FORMAT	UINT16	COF<value>;	COF?;	<value>	MSV output format
3011	HOUR	UINT16	CLK<value>;	CLK?;	<value>	Clock hour
3012	MINUTE	UINT16	CLK,<value>;	CLK?;	,<value>	Clock minute
3013	SECOND	UINT16	CLK,,<value>;	CLK?;	,,<value>	Clock second
3014	DAY	UINT16	CLK,,,<value>;	CLK?;	,,,<value>	Clock day of month
3015	MONTH	UINT16	CLK,,,,<value>;	CLK?;	,,,,<value>	Clock month
3016	YEAR	UINT16	CLK,,,,,<value>;	CLK?;	,,,,,<value>	Clock year
3021	FULL_PASS	UINT32	DPF<value>;	DPF?;	<value>	Full passcode
3023	SAFE_PASS	UINT32	DPS<value>;	DPS?;	<value>	Safe passcode
3031	PERMIS_KEY_ZERO	UINT16	LBT0,<value>;	LBT?0;	<value>	Zero key permissions
3032	PERMIS_KEY_TARE	UINT16	LBT1,<value>;	LBT?1;	<value>	Tare key permissions
3033	PERMIS_KEY_GROSSNET	UINT16	LBT2,<value>;	LBT?2;	<value>	Gross/net key permissions
3034	PERMIS_KEY_FUNC1	UINT16	LBT3,<value>;	LBT?3;	<value>	Function key 1 permissions
3035	PERMIS_KEY_FUNC2	UINT16	LBT4,<value>;	LBT?4;	<value>	Function key 2 permissions
3036	PERMIS_KEY_FUNC3	UINT16	LBT5,<value>;	LBT?5;	<value>	Function key 3 permissions
3042	KEY_REM1_FN	UINT16	RBT1,<value>;	RBT?1;	<value>	Remote button 1 function
3043	KEY_REM2_FN	UINT16	RBT2,<value>;	RBT?2;	<value>	Remote button 2 function
3044	KEY_REM3_FN	UINT16	RBT3,<value>;	RBT?3;	<value>	Remote button 3 function
3045	KEY_REM4_FN	UINT16	RBT4,<value>;	RBT?4;	<value>	Remote button 4 function
3046	KEY_REM5_FN	UINT16	RBT5,<value>;	RBT?5;	<value>	Remote button 5 function
3047	KEY_REM6_FN	UINT16	RBT6,<value>;	RBT?6;	<value>	Remote button 6 function
3048	KEY_REM7_FN	UINT16	RBT7,<value>;	RBT?7;	<value>	Remote button 7 function
3049	KEY_REM8_FN	UINT16	RBT8,<value>;	RBT?8;	<value>	Remote button 8 function
3051	QA_ENABLE	UINT16	QAF<value>;	QAF?;	<value>	Quality assurance enable
3052	QA_DATE	UINT16	QAF,<value>;	QAF?;	,<value>	Quality assurance day of month
3053	QA_MONTH	UINT16	QAF,,<value>;	QAF?;	,,<value>	Quality assurance month
3054	QA_YEAR	UINT16	QAF,,,<value>;	QAF?;	,,,<value>	Quality assurance year
3061	KEY_FUNC1_FN	UINT16	FBT1,<value>;	FBT?1;	<value>	Function key 1 function
3062	KEY_FUNC2_FN	UINT16	FBT2,<value>;	FBT?2;	<value>	Function key 2 function
3063	KEY_FUNC3_FN	UINT16	FBT3,<value>;	FBT?3;	<value>	Function key 3 function
4001	USER_ZERO	UINT16	CDL;			Perform User zero
4002	TARE	SINT32	TAR;	TAR?;	<value>	Perform semi-automatic tare
4004	SWITCH_GROSS_NET	UINT16	TAS<value>;	TAS?;	<value>	Switch gross and net
4005	PRESET_TARE	SINT32	TAV<value>;	TAV?;	<value>	Perform preset tare
4101	DEFAULT_USERDB	UINT16	DFT0;			Default user database
4102	DEFAULT_RTDB	UINT16	DFT1;			Default runtime database
4103	DEFAULT_CAL	UINT16	DFT2;			Default calibration
4104	DEFAULT_ETH	UINT16	DFT3;			Default ethernet settings
4151	SAVE_LOAD_DEFAULT_SETTINGS	UINT32	TDD<value>;	TDD?;	<value>	Save settings/load settings/set defaults/read cal count
4201	DISPLAY	STR(32)		FPL?;	<value>	Read display buffer
4222	KEY_REM1_PRESS	UINT16	RBT1,,<value>;			Remote button 1 press (0 short, 1 long)
4223	KEY_REM2_PRESS	UINT16	RBT2,,<value>;			Remote button 2 press (0 short, 1 long)
4224	KEY_REM3_PRESS	UINT16	RBT3,,<value>;			Remote button 3 press (0 short, 1 long)
4225	KEY_REM4_PRESS	UINT16	RBT4,,<value>;			Remote button 4 press (0 short, 1 long)
4226	KEY_REM5_PRESS	UINT16	RBT5,,<value>;			Remote button 5 press (0 short, 1 long)
4227	KEY_REM6_PRESS	UINT16	RBT6,,<value>;			Remote button 6 press (0 short, 1 long)
4228	KEY_REM7_PRESS	UINT16	RBT7,,<value>;			Remote button 7 press (0 short, 1 long)
4229	KEY_REM8_PRESS	UINT16	RBT8,,<value>;			Remote button 8 press (0 short, 1 long)
4231	KEY_ZERO	UINT16	FPL1,<value>;			Zero key press (0 short, 1 long)
4232	KEY_TARE	UINT16	FPL2,<value>;			Tare key press (0 short, 1 long)
4233	KEY_GROSSNET	UINT16	FPL3,<value>;			Gross/net key press (0 short, 1 long)
4234	KEY_FUNC1	UINT16	FPL4,<value>;			Function 1 key press (0 short, 1 long)
4235	KEY_FUNC2	UINT16	FPL5,<value>;			Function 2 key press (0 short, 1 long)
4236	KEY_FUNC3	UINT16	FPL6,<value>;			Function 3 key press (0 short, 1 long)

Register	Name	Type	Write	Read	Read Response	Description
4241	KEY_FUNC1_PRESS	UINT16	FBT1,,<value>;			Function key 1 press (0 short, 1 long)
4242	KEY_FUNC2_PRESS	UINT16	FBT2,,<value>;			Function key 2 press (0 short, 1 long)
4243	KEY_FUNC3_PRESS	UINT16	FBT3,,<value>;			Function key 3 press (0 short, 1 long)
4251	DISPLAY_BRIGHTNESS	UINT16	BRT<value>;	BRT?;	<value>	Display brightness
4301	RESET_UNIT	UINT16	RES;			Reset unit
5001	DOUT1	UINT16	FOP<value>;	FOP?;	<value>	Digital output 1 state
5002	DOUT2	UINT16	FOP,<value>;	FOP?;	,<value>	Digital output 2 state
5003	DOUT3	UINT16	FOP,,<value>;	FOP?;	,,<value>	Digital output 3 state
5004	DOUT4	UINT16	FOP,,,<value>;	FOP?;	,,,<value>	Digital output 4 state
5005	DOUT5	UINT16	FOP,,,,<value>;	FOP?;	,,,,<value>	Digital output 5 state
5006	DOUT6	UINT16	FOP,,,,,<value>;	FOP?;	,,,,,<value>	Digital output 6 state
5007	DOUT7	UINT16	FOP,,,,,,<value>;	FOP?;	,,,,,,<value>	Digital output 7 state
5008	DOUT8	UINT16	FOP,,,,,,,<value>;	FOP?;	,,,,,,,<value>	Digital output 8 state
5011	DOUT1	UINT16	POR<value>;	POR?;	<value>	Digital output 1 state
5012	DOUT2	UINT16	POR,<value>;	POR?;	,<value>	Digital output 2 state
5013	DOUT3	UINT16	POR,,<value>;	POR?;	,,<value>	Digital output 3 state
5014	DOUT4	UINT16	POR,,,<value>;	POR?;	,,,<value>	Digital output 4 state
5015	DOUT5	UINT16	POR,,,,<value>;	POR?;	,,,,<value>	Digital output 5 state
5016	DOUT6	UINT16	POR,,,,,<value>;	POR?;	,,,,,<value>	Digital output 6 state
5017	DOUT7	UINT16	POR,,,,,,<value>;	POR?;	,,,,,,<value>	Digital output 7 state
5018	DOUT8	UINT16	POR,,,,,,,<value>;	POR?;	,,,,,,,<value>	Digital output 8 state
5101	SP1_TYPE	UINT16	LIV1,<value>;	LIV?1;	1,<value>	Setpoint 1 type
5102	SP1_SOURCE	UINT16	LIV1,,<value>;	LIV?1;	1,,<value>	Setpoint 1 source
5104	SP1_TARGET	SINT32	LIV1,,,<value>;	LIV?1;	1,,,<value>	Setpoint 1 target
5106	SP1_INFLIGHT	SINT32	LIV1,,,,,<value>;	LIV?1;	1,,,,,<value>	Setpoint 1 inflight
5108	SP1_HYSTERESIS	SINT32	LIV1,,,,,,<value>;	LIV?1;	1,,,,,,<value>	Setpoint 1 hysteresis
5110	SP1_LOGIC	UINT16	LIV1,,,,,,,<value>;	LIV?1;	1,,,,,,,<value>	Setpoint 1 logic inversion
5111	SP1_LOCK	UINT16	LIV1,,,,,,,,<value>;	LIV?1;	1,,,,,,,,<value>	Setpoint 1 lock
5112	SP1_ALARM	UINT16	LIV1,,,,,,,,,<value>;	LIV?1;	1,,,,,,,,,<value>	Setpoint 1 alarm
5113	SP1_TIMING	UINT16	LIV1,,,,,,,,,,<value>;	LIV?1;	1,,,,,,,,,,<value>	Setpoint 1 timing
5114	SP1_RESET	UINT16	LIV1,,,,,,,,,,,<value>;	LIV?1;	1,,,,,,,,,,,<value>	Setpoint 1 reset
5121	SP2_TYPE	UINT16	LIV2,<value>;	LIV?2;	2,<value>	Setpoint 2 type
5122	SP2_SOURCE	UINT16	LIV2,,<value>;	LIV?2;	2,,<value>	Setpoint 2 source
5124	SP2_TARGET	SINT32	LIV2,,,<value>;	LIV?2;	2,,,<value>	Setpoint 2 target
5126	SP2_INFLIGHT	SINT32	LIV2,,,,,<value>;	LIV?2;	2,,,,,<value>	Setpoint 2 inflight
5128	SP2_HYSTERESIS	SINT32	LIV2,,,,,,<value>;	LIV?2;	2,,,,,,<value>	Setpoint 2 hysteresis
5130	SP2_LOGIC	UINT16	LIV2,,,,,,,<value>;	LIV?2;	2,,,,,,,<value>	Setpoint 2 logic inversion
5131	SP2_LOCK	UINT16	LIV2,,,,,,,,<value>;	LIV?2;	2,,,,,,,,<value>	Setpoint 2 lock
5132	SP2_ALARM	UINT16	LIV2,,,,,,,,,<value>;	LIV?2;	2,,,,,,,,,<value>	Setpoint 2 alarm
5133	SP2_TIMING	UINT16	LIV2,,,,,,,,,,<value>;	LIV?2;	2,,,,,,,,,,<value>	Setpoint 2 timing
5134	SP2_RESET	UINT16	LIV2,,,,,,,,,,,<value>;	LIV?2;	2,,,,,,,,,,,<value>	Setpoint 2 reset
5141	SP3_TYPE	UINT16	LIV3,<value>;	LIV?3;	3,<value>	Setpoint 3 type
5142	SP3_SOURCE	UINT16	LIV3,,<value>;	LIV?3;	3,,<value>	Setpoint 3 source
5144	SP3_TARGET	SINT32	LIV3,,,<value>;	LIV?3;	3,,,<value>	Setpoint 3 target
5146	SP3_INFLIGHT	SINT32	LIV3,,,,,<value>;	LIV?3;	3,,,,,<value>	Setpoint 3 inflight
5148	SP3_HYSTERESIS	SINT32	LIV3,,,,,,<value>;	LIV?3;	3,,,,,,<value>	Setpoint 3 hysteresis
5150	SP3_LOGIC	UINT16	LIV3,,,,,,,<value>;	LIV?3;	3,,,,,,,<value>	Setpoint 3 logic inversion
5151	SP3_LOCK	UINT16	LIV3,,,,,,,,<value>;	LIV?3;	3,,,,,,,,<value>	Setpoint 3 lock
5152	SP3_ALARM	UINT16	LIV3,,,,,,,,,<value>;	LIV?3;	3,,,,,,,,,<value>	Setpoint 3 alarm
5153	SP3_TIMING	UINT16	LIV3,,,,,,,,,,<value>;	LIV?3;	3,,,,,,,,,,<value>	Setpoint 3 timing
5154	SP3_RESET	UINT16	LIV3,,,,,,,,,,,<value>;	LIV?3;	3,,,,,,,,,,,<value>	Setpoint 3 reset
5161	SP4_TYPE	UINT16	LIV4,<value>;	LIV?4;	4,<value>	Setpoint 4 type
5162	SP4_SOURCE	UINT16	LIV4,,<value>;	LIV?4;	4,,<value>	Setpoint 4 source
5164	SP4_TARGET	SINT32	LIV4,,,<value>;	LIV?4;	4,,,<value>	Setpoint 4 target
5166	SP4_INFLIGHT	SINT32	LIV4,,,,,<value>;	LIV?4;	4,,,,,<value>	Setpoint 4 inflight
5168	SP4_HYSTERESIS	SINT32	LIV4,,,,,,<value>;	LIV?4;	4,,,,,,<value>	Setpoint 4 hysteresis
5170	SP4_LOGIC	UINT16	LIV4,,,,,,,<value>;	LIV?4;	4,,,,,,,<value>	Setpoint 4 logic inversion
5171	SP4_LOCK	UINT16	LIV4,,,,,,,,<value>;	LIV?4;	4,,,,,,,,<value>	Setpoint 4 lock
5172	SP4_ALARM	UINT16	LIV4,,,,,,,,,<value>;	LIV?4;	4,,,,,,,,,<value>	Setpoint 4 alarm
5173	SP4_TIMING	UINT16	LIV4,,,,,,,,,,<value>;	LIV?4;	4,,,,,,,,,,<value>	Setpoint 4 timing
5174	SP4_RESET	UINT16	LIV4,,,,,,,,,,,<value>;	LIV?4;	4,,,,,,,,,,,<value>	Setpoint 4 reset
5181	SP5_TYPE	UINT16	LIV5,<value>;	LIV?5;	5,<value>	Setpoint 5 type
5182	SP5_SOURCE	UINT16	LIV5,,<value>;	LIV?5;	5,,<value>	Setpoint 5 source
5184	SP5_TARGET	SINT32	LIV5,,,<value>;	LIV?5;	5,,,<value>	Setpoint 5 target

Register	Name	Type	Write	Read	Read Response	Description
5186	SP5_INFLIGHT	SINT32	LIV5,,,,,<value>;	LIV?5;	5,,,,,<value>	Setpoint 5 inflight
5188	SP5_HYSTERESIS	SINT32	LIV5,,,,,,<value>;	LIV?5;	5,,,,,,<value>	Setpoint 5 hysteresis
5190	SP5_LOGIC	UINT16	LIV5,,,,,,,<value>;	LIV?5;	5,,,,,,,<value>	Setpoint 5 logic inversion
5191	SP5_LOCK	UINT16	LIV5,,,,,,,,<value>;	LIV?5;	5,,,,,,,,<value>	Setpoint 5 lock
5192	SP5_ALARM	UINT16	LIV5,,,,,,,,,<value>;	LIV?5;	5,,,,,,,,,<value>	Setpoint 5 alarm
5193	SP5_TIMING	UINT16	LIV5,,,,,,,,,,<value>;	LIV?5;	5,,,,,,,,,,<value>	Setpoint 5 timing
5194	SP5_RESET	UINT16	LIV5,,,,,,,,,,,<value>;	LIV?5;	5,,,,,,,,,,,<value>	Setpoint 5 reset
5201	SP6_TYPE	UINT16	LIV6,<value>;	LIV?6;	6,<value>	Setpoint 6 type
5202	SP6_SOURCE	UINT16	LIV6,,<value>;	LIV?6;	6,,<value>	Setpoint 6 source
5204	SP6_TARGET	SINT32	LIV6,,,,<value>;	LIV?6;	6,,,,<value>	Setpoint 6 target
5206	SP6_INFLIGHT	SINT32	LIV6,,,,,<value>;	LIV?6;	6,,,,,<value>	Setpoint 6 inflight
5208	SP6_HYSTERESIS	SINT32	LIV6,,,,,,<value>;	LIV?6;	6,,,,,,<value>	Setpoint 6 hysteresis
5210	SP6_LOGIC	UINT16	LIV6,,,,,,,<value>;	LIV?6;	6,,,,,,,<value>	Setpoint 6 logic inversion
5211	SP6_LOCK	UINT16	LIV6,,,,,,,,<value>;	LIV?6;	6,,,,,,,,<value>	Setpoint 6 lock
5212	SP6_ALARM	UINT16	LIV6,,,,,,,,,<value>;	LIV?6;	6,,,,,,,,,<value>	Setpoint 6 alarm
5213	SP6_TIMING	UINT16	LIV6,,,,,,,,,,<value>;	LIV?6;	6,,,,,,,,,,<value>	Setpoint 6 timing
5214	SP6_RESET	UINT16	LIV6,,,,,,,,,,,<value>;	LIV?6;	6,,,,,,,,,,,<value>	Setpoint 6 reset
5221	SP7_TYPE	UINT16	LIV7,<value>;	LIV?7;	7,<value>	Setpoint 7 type
5222	SP7_SOURCE	UINT16	LIV7,,<value>;	LIV?7;	7,,<value>	Setpoint 7 source
5224	SP7_TARGET	SINT32	LIV7,,,,<value>;	LIV?7;	7,,,,<value>	Setpoint 7 target
5226	SP7_INFLIGHT	SINT32	LIV7,,,,,<value>;	LIV?7;	7,,,,,<value>	Setpoint 7 inflight
5228	SP7_HYSTERESIS	SINT32	LIV7,,,,,,<value>;	LIV?7;	7,,,,,,<value>	Setpoint 7 hysteresis
5230	SP7_LOGIC	UINT16	LIV7,,,,,,,<value>;	LIV?7;	7,,,,,,,<value>	Setpoint 7 logic inversion
5231	SP7_LOCK	UINT16	LIV7,,,,,,,,<value>;	LIV?7;	7,,,,,,,,<value>	Setpoint 7 lock
5232	SP7_ALARM	UINT16	LIV7,,,,,,,,,<value>;	LIV?7;	7,,,,,,,,,<value>	Setpoint 7 alarm
5233	SP7_TIMING	UINT16	LIV7,,,,,,,,,,<value>;	LIV?7;	7,,,,,,,,,,<value>	Setpoint 7 timing
5234	SP7_RESET	UINT16	LIV7,,,,,,,,,,,<value>;	LIV?7;	7,,,,,,,,,,,<value>	Setpoint 7 reset
5241	SP8_TYPE	UINT16	LIV8,<value>;	LIV?8;	8,<value>	Setpoint 8 type
5242	SP8_SOURCE	UINT16	LIV8,,<value>;	LIV?8;	8,,<value>	Setpoint 8 source
5244	SP8_TARGET	SINT32	LIV8,,,,<value>;	LIV?8;	8,,,,<value>	Setpoint 8 target
5246	SP8_INFLIGHT	SINT32	LIV8,,,,,<value>;	LIV?8;	8,,,,,<value>	Setpoint 8 inflight
5248	SP8_HYSTERESIS	SINT32	LIV8,,,,,,<value>;	LIV?8;	8,,,,,,<value>	Setpoint 8 hysteresis
5250	SP8_LOGIC	UINT16	LIV8,,,,,,,<value>;	LIV?8;	8,,,,,,,<value>	Setpoint 8 logic inversion
5251	SP8_LOCK	UINT16	LIV8,,,,,,,,<value>;	LIV?8;	8,,,,,,,,<value>	Setpoint 8 lock
5252	SP8_ALARM	UINT16	LIV8,,,,,,,,,<value>;	LIV?8;	8,,,,,,,,,<value>	Setpoint 8 alarm
5253	SP8_TIMING	UINT16	LIV8,,,,,,,,,,<value>;	LIV?8;	8,,,,,,,,,,<value>	Setpoint 8 timing
5254	SP8_RESET	UINT16	LIV8,,,,,,,,,,,<value>;	LIV?8;	8,,,,,,,,,,,<value>	Setpoint 8 reset
6001	ERROR_SYS	UINT32		ESM?0;	<value>	Error status system (equivalent of ESR)
6003	ERROR_LATCH	UINT32		ESM?1;	<value>	Error status latched (equivalent of ESR)
6101	UNIT_ID	STR(15)	IDM<value>;	IDM?;	<value>	Unit identification (equivalent of IDN)
6109	SERIAL_NO	UINT32		IDM?;	,<value>	Unit serial number (equivalent of IDN)
6111	SW_VERSION	STR(10)		IDM?;	,,<value>	Unit software version (equivalent of IDN)
6116	UNIT_MODEL	STR(7)		IDM?;	,,,<value>	Unit model (equivalent of IDN)
6201	WEIGHT_DISPLAYED	SINT32		MSM?1;	<value>	Measured displayed weight (equivalent of MSV)
6203	WEIGHT_STATUS_DISPLAYED	UINT32		MSM?1;	,<value>	Measured displayed weight status
6205	WEIGHT_GROSS	SINT32		MSM?2;	<value>	Measured gross weight (equivalent of MSV)
6207	WEIGHT_STATUS_GROSS	UINT32		MSM?2;	,<value>	Measured gross weight status
6209	WEIGHT_NET	SINT32		MSM?3;	<value>	Measured net weight (equivalent of MSV)
6211	WEIGHT_STATUS_NET	UINT32		MSM?3;	,<value>	Measured net weight status
6213	WEIGHT_TOTAL_ITEMS	SINT32		MSM?4;	<value>	Measured total number of items (equivalent of MSV)
6215	WEIGHT_STATUS_TOTAL_ITEMS	UINT32		MSM?4;	,<value>	Measured total number of items weight status
6217	WEIGHT_TOTAL	SINT32		MSM?5;	<value>	Measured total weight (equivalent of MSV)
6219	WEIGHT_STATUS_TOTAL	UINT32		MSM?5;	,<value>	Measured total weight status
6221	WEIGHT_PEAK_MAX	SINT32		MSM?6;	<value>	Measured peak maximum weight (equivalent of MSV)
6223	WEIGHT_STATUS_PEAK_MAX	UINT32		MSM?6;	,<value>	Measured peak maximum weight status
6225	WEIGHT_LIVEWEIGH	SINT32		MSM?7;	<value>	Measured live weigh weight (equivalent of MSV)
6227	WEIGHT_STATUS_LIVEWEIGH	UINT32		MSM?7;	,<value>	Measured live weigh status

Register	Name	Type	Write	Read	Read Response	Description
6251	MILLI_VOLT_PER_VOLT	SINT32		VAL?;	<value>	Millivolt per volt reading
6301	PRINT_ID	UINT32		PRT?0;	<value>	Print ID
6303	PRINT	UINT16	PRT;			Print standard printout
6351	PRINT_BUFFER	STR(100)		PRT?1;	<value>	Print buffer
6401	PRINT_SUPPLIED_FORMAT	STR(246)	PRT,<value>;			Print using supplied format string

19.3 Modbus Input Registers

The following Modbus functions are implemented for input registers:

- Read input registers (function code 0x04)

The following table lists the translation for each Modbus input register into the corresponding serial communication command:

Register	Name	Type	Write	Read	Read Response	Description
1	INPUT_WEIGHT_GROSS	SINT32		MSM?2;	<value>	Measured weight gross (equivalent of MSV)
3	INPUT_WEIGHT_NET	SINT32		MSM?3;	<value>	Measured weight net (equivalent of MSV)
5	INPUT_WEIGHT_DISPLAYED	SINT32		MSM?1;	<value>	Measured weight displayed (equivalent of MSV)
7	INPUT_WEIGHT_STATUS	UINT32		MSM?1;	,<value>	Measured weight status (equivalent of MSV)
9	INPUT_ERROR	UINT32		ESM?0;	<value>	Error status (equivalent of ESR)

20 Ethernet/IP Optional Software

20.1 Overview

The C500 provides the Ethernet/IP industrial network protocol via the L900-501 licencensed package. For details on licensing this package, please see Section 4.

The Ethernet/IP package provides a number of simple weight and status EIP objects, and more complex objects which allow access to all Rincmd registers.

20.2 Terminology

Throughout this section the following Ethernet/IP terminology is used:

Term	Description
EIP	Ethernet/IP
PLC	Programmable Logic Controller.
T->O	Target to Originator. This is an input in EIP terms (input to PLC). Data is transferred from device to PLC.
O->T	Originator to Target. This is an output in EIP terms (output from PLC). Data is transferred from PLC to device.
GAS	Get attribute single service code used to retrieve an EIP object using explicit messaging.
GAA	Get attributes all service code used to retrieve an all instances for EIP object class using explicit messaging.
SAS	Set attribute single service code used to set an EIP object using explicit messaging.
EDS file	Electronic Data Sheet file that describes the implicit messages supported by this device.

In this document the C500 is the Target, and the PLC is the Originator.

Ethernet/IP uses the following data types:

Data Type	Description
USINT	Unsigned Short Integer (8-bit)
UINT	Unsigned Integer (16-bit)
UDINT	Unsigned Double Integer (32-bit)
SINT	Signed Short Integer (8-bit)
INT	Signed Integer (16-bit)
DINT	Signed Double Integer (32-bit)
STRING	Character String (1 byte per character)
SHORT STRINGnn	Character String (1st byte is length; up to nn characters)
BYTE	Bit String (8-bits)
WORD	Bit String (16-bits)
REAL	IEEE 32-bit Single Precision Floating Point

20.2.1 Electronic Data Sheet (EDS) File

The C500 supplies a Electronic Data Sheet file to describe the supported implicit (assembly) objects. This file can be loaded into your PLC software to simplify programming.

Once the L900-501 package has been installed on the C500, the EDS file is available from the following locations:

1. Standard EIP File Object Class (Class 37_H/ Object $C8_H$)
2. C500 web interface (http://<indicator>/eip/c500.eds)

In addition an icon file is available from the C500 web interface (http://<indicator>/eip/rinstrum_c500.ico).

20.3 Object Model

Ethernet/IP represents data as objects. The objects supported by the C500 are described here.

20.3.1 Identity Object (Class 01_H)

The following table describes the attribute, status, and common services information for the Identity Object.

Instance	Attribute ID	Name	Object	Value	Access
0 (Class)	1	Revision	UINT	1	Get
1	1	Vendor ID	UINT	1501	Get
1	2	Device type	UINT	43	Get
1	3	Product code	UINT	1	Get
1	4	Product revision	<table><tr><th>Bytes</th><th>Type</th><th>Description</th><th>Value</th></tr><tr><td>0</td><td>USINT</td><td>Major revision</td><td>2</td></tr><tr><td>1</td><td>USINT</td><td>Minor revision</td><td>64</td></tr></table>		Get
1	5	Status	UINT	See Section 20.3.2	Get
1	6	Serial number	UDINT	C500 serial number	Get
1	7	Product name	STRING	`Rinstrum C500`	Get
1	8	State	USINT	See Section 20.3.2	Get

The services provided by the identity object are as follows:

Service Code	Service Name	Class Level	Instance Level
$0E_H$	GAS	Yes	Yes
01_H	GAA	No	Yes
10_H	SAS	No	No
05_H	Reset	No	Yes

20.3.2 Identity Object State and Status

The following table maps the indicator errors into EIP faults. The columns are as follows:

- Indicator Error are the error codes listed in Section 24.4
- EIP EDS is the Extended Device Status bits (4-7) within the identity object status UDINT (attribute 5)
- EIP Fault bit are the fault bits (8,9,10,11) within the identity object status UDINT (attribute 5)
- EIP State is the identity object state USINT (attribute 8)
- See ODVA Vol 1: Identity object for more information

Indicator Error	EIP EDS	EIP Fault Bit	EIP State
`E00000`	0	None	Operational (3)
`E00001`	5 (major fault)	Major Recoverable (bit 10)	Major Recoverable Fault (4)
`E00002`	5 (major fault)	Major Recoverable (bit 10)	Major Recoverable Fault (4)
`E00004`	5 (major fault)	Major Recoverable (bit 10)	Major Recoverable Fault (4)
`E00008`	5 (major fault)	Major Recoverable (bit 10)	Major Recoverable Fault (4)
`E00010`	5 (major fault)	Major Recoverable (bit 10)	Major Recoverable Fault (4)
`E00020`	5 (major fault)	Major Unrecoverable (bit 11)	Major Unrecoverable Fault (5)

E00080	5 (major fault)	Major Recoverable (bit 10)	Major Recoverable Fault (4)
E00100	5 (major fault)	Major Unrecoverable (bit 11)	Major Unrecoverable Fault (5)
E00200	5 (major fault)	Major Recoverable (bit 10)	Major Recoverable Fault (4)
E00400	5 (major fault)	Major Unrecoverable (bit 11)	Major Unrecoverable Fault (5)
E00800	5 (major fault)	Major Recoverable (bit 10)	Major Recoverable Fault (4)
E01000	5 (major fault)	Major Recoverable (bit 10)	Major Recoverable Fault (4)
E02000	5 (major fault)	Major Recoverable (bit 10)	Major Recoverable Fault (4)
E04000	5 (major fault)	Major Recoverable (bit 10)	Major Recoverable Fault (4)
E08000	5 (major fault)	Major Unrecoverable (bit 11)	Major Unrecoverable Fault (5)
E10000	5 (major fault)	Major Unrecoverable (bit 11)	Major Unrecoverable Fault (5)
E20000	5 (major fault)	Major Unrecoverable (bit 11)	Major Unrecoverable Fault (5)

20.3.3 Assembly Object (Class 04_H)

<table>
<tr><th>Instance</th><th>Attribute ID</th><th>Name</th><th>Object</th><th>Access</th></tr>
<tr><td>0 (Class)</td><td>1</td><td>Revision</td><td>UINT (value 2)</td><td>Get</td></tr>
<tr><td>0 (Class)</td><td>2</td><td>Max instance</td><td>UINT (value 129)</td><td>Get</td></tr>
<tr><td>100</td><td>3</td><td>T->O Primary Units Simple Weight</td><td><table><tr><th>Bytes</th><th>Type</th><th>Description</th></tr><tr><td>0-3</td><td>DINT</td><td>Displayed Weight</td></tr><tr><td>4</td><td>SINT</td><td>Units</td></tr><tr><td>5</td><td>SINT</td><td>Decimal Point</td></tr><tr><td>6-7</td><td>INT</td><td>Weight Status (16bit)</td></tr></table>See Section 21.3.17 for units values
See Section 21.3.22 for decimal point values
See Section 20.3.3 for weight status bits</td><td>Get</td></tr>
<tr><td>101</td><td>3</td><td>T->O Primary Units Full Weight</td><td>See<table><tr><th>Bytes</th><th>Type</th><th>Description</th></tr><tr><td>0-3</td><td>DINT</td><td>Displayed Weight</td></tr><tr><td>4-7</td><td>DINT</td><td>Gross Weight</td></tr><tr><td>8-11</td><td>DINT</td><td>Net Weight</td></tr><tr><td>12-15</td><td>DINT</td><td>Tare Weight</td></tr><tr><td>16</td><td>SINT</td><td>Units</td></tr><tr><td>17</td><td>SINT</td><td>Decimal Point</td></tr><tr><td>18-19</td><td>INT</td><td>Weight Status (16bit)</td></tr><tr><td>20-23</td><td>DINT</td><td>Extended Status (32bit)</td></tr><tr><td>24-27</td><td>DINT</td><td>I/O Status (32bit)</td></tr></table>Section 21.3.17 for units values
See Section 21.3.22 for decimal point values
See Section 20.3.3 for weight status bits
See Section 20.3.3 for extended status bits
See Section 20.3.3 for I/O status bits</td><td>Get</td></tr>
</table>

<table>
<tr>
<td>102</td>
<td>3</td>
<td>T->O Secondary Units Simple Weight</td>
<td>
<table>
<tr><th>Bytes</th><th>Type</th><th>Description</th></tr>
<tr><td>0-3</td><td>DINT</td><td>Displayed Weight</td></tr>
<tr><td>4</td><td>SINT</td><td>Units</td></tr>
<tr><td>5</td><td>SINT</td><td>Decimal Point</td></tr>
<tr><td>6-7</td><td>INT</td><td>Weight Status (16bit)</td></tr>
</table>
Section 21.3.17 for units values
See Section 21.3.22 for decimal point values
See Section 20.3.3 for weight status bits
</td>
<td>See</td>
<td>Get</td>
</tr>
<tr>
<td>103</td>
<td>3</td>
<td>T->O Secondary Units Full Weight</td>
<td>
<table>
<tr><th>Bytes</th><th>Type</th><th>Description</th></tr>
<tr><td>0-3</td><td>DINT</td><td>Displayed Weight</td></tr>
<tr><td>4-7</td><td>DINT</td><td>Gross Weight</td></tr>
<tr><td>8-11</td><td>DINT</td><td>Net Weight</td></tr>
<tr><td>12-15</td><td>DINT</td><td>Tare Weight</td></tr>
<tr><td>16</td><td>SINT</td><td>Units</td></tr>
<tr><td>17</td><td>SINT</td><td>Decimal Point</td></tr>
<tr><td>18-19</td><td>INT</td><td>Weight Status (16bit)</td></tr>
<tr><td>20-23</td><td>DINT</td><td>Extended Status (32bit)</td></tr>
<tr><td>24-27</td><td>DINT</td><td>I/O Status (32bit)</td></tr>
</table>
Section 21.3.17 for units values
See Section 21.3.22 for decimal point values
See Section 20.3.3 for weight status bits
See Section 20.3.3 for extended status bits
See Section 20.3.3 for I/O status bits
</td>
<td>See</td>
<td>Get</td>
</tr>
<tr>
<td>104</td>
<td>3</td>
<td>T->O Tertiary Units Simple Weight</td>
<td>
<table>
<tr><th>Bytes</th><th>Type</th><th>Description</th></tr>
<tr><td>0-3</td><td>DINT</td><td>Displayed Weight</td></tr>
<tr><td>4</td><td>SINT</td><td>Units</td></tr>
<tr><td>5</td><td>SINT</td><td>Decimal Point</td></tr>
<tr><td>6-7</td><td>INT</td><td>Weight Status (16bit)</td></tr>
</table>
Section 21.3.17 for units values
See Section 21.3.22 for decimal point values
See Section 20.3.3 for weight status bits
</td>
<td>See</td>
<td>Get</td>
</tr>
<tr>
<td>105</td>
<td>3</td>
<td>T->O Tertiary Units Full Weight</td>
<td>
<table>
<tr><th>Bytes</th><th>Type</th><th>Description</th></tr>
<tr><td>0-3</td><td>DINT</td><td>Displayed Weight</td></tr>
<tr><td>4-7</td><td>DINT</td><td>Gross Weight</td></tr>
<tr><td>8-11</td><td>DINT</td><td>Net Weight</td></tr>
<tr><td>12-15</td><td>DINT</td><td>Tare Weight</td></tr>
<tr><td>16</td><td>SINT</td><td>Units</td></tr>
<tr><td>17</td><td>SINT</td><td>Decimal Point</td></tr>
<tr><td>18-19</td><td>INT</td><td>Weight Status (16bit)</td></tr>
<tr><td>20-23</td><td>DINT</td><td>Extended Status (32bit)</td></tr>
<tr><td>24-27</td><td>DINT</td><td>I/O Status (32bit)</td></tr>
</table>
Section 21.3.17 for units values
See Section 21.3.22 for decimal point values
See Section 20.3.3 for weight status bits
See Section 20.3.3 for extended status bits
See Section 20.3.3 for I/O status bits
</td>
<td>See</td>
<td>Get</td>
</tr>
</table>

<table>
<tr>
<td>106</td>
<td>3</td>
<td>T->O System Status</td>
<td>
<table>
<tr><th>Bytes</th><th>Type</th><th>Description</th></tr>
<tr><td>0-3</td><td>DINT</td><td>Extended Status (32bit)</td></tr>
<tr><td>4-7</td><td>DINT</td><td>System Error (32bit)</td></tr>
<tr><td>8-11</td><td>DINT</td><td>I/O Status (32bit)</td></tr>
<tr><td>12-15</td><td>DINT</td><td>I/O Mask (32bit)</td></tr>
<tr><td>16</td><td>SINT</td><td>Primary Units</td></tr>
<tr><td>17</td><td>SINT</td><td>Primary Decimal Point</td></tr>
<tr><td>18</td><td>SINT</td><td>Primary Range 1 Resolution</td></tr>
<tr><td>19</td><td>SINT</td><td>Primary Range 2 Resolution</td></tr>
<tr><td>20</td><td>SINT</td><td>Primary Range 3 Resolution</td></tr>
<tr><td>21</td><td>SINT</td><td>Unused (always 0)</td></tr>
<tr><td>22</td><td>SINT</td><td>Unused (always 0)</td></tr>
<tr><td>23</td><td>SINT</td><td>Unused (always 0)</td></tr>
<tr><td>24</td><td>SINT</td><td>Secondary Units</td></tr>
<tr><td>25</td><td>SINT</td><td>Secondary Decimal Point</td></tr>
<tr><td>26</td><td>SINT</td><td>Secondary Range 1 Resolution</td></tr>
<tr><td>27</td><td>SINT</td><td>Secondary Range 2 Resolution</td></tr>
<tr><td>28</td><td>SINT</td><td>Secondary Range 3 Resolution</td></tr>
<tr><td>29</td><td>SINT</td><td>Unused (always 0)</td></tr>
<tr><td>30</td><td>SINT</td><td>Unused (always 0)</td></tr>
<tr><td>31</td><td>SINT</td><td>Unused (always 0)</td></tr>
<tr><td>32</td><td>SINT</td><td>Tertiary Units</td></tr>
<tr><td>33</td><td>SINT</td><td>Tertiary Decimal Point</td></tr>
<tr><td>34</td><td>SINT</td><td>Tertiary Range 1 Resolution</td></tr>
<tr><td>35</td><td>SINT</td><td>Tertiary Range 2 Resolution</td></tr>
<tr><td>36</td><td>SINT</td><td>Tertiary Range 3 Resolution</td></tr>
<tr><td>37</td><td>SINT</td><td>Unused (always 0)</td></tr>
<tr><td>38</td><td>SINT</td><td>Unused (always 0)</td></tr>
<tr><td>39</td><td>SINT</td><td>Unused (always 0)</td></tr>
</table>
See Section 21.3.17 for units values
See Section 21.3.22 for decimal point and range resolution values
See Section 20.3.3 for extended status bits
See Section 20.3.3 for I/O status bits
See Section 24.4 for system errors
</td>
<td>Get</td>
</tr>
</table>

<table>
<tr><td>120</td><td>3</td><td>T->O Rincmd Result</td><td><table><tr><th>Bytes</th><th>Type</th><th>Description</th></tr><tr><td>0-3</td><td>DINT</td><td>Rincmd Status Flags
Bit 0: In progress
Bit 1: Invalid operation
Bit 2: Timeout
Bit 3: Error
Bit 4: Success
Bit 5: Connection lost
Bit 6-31: Reserved</td></tr><tr><td>4-7</td><td>DINT</td><td>Rincmd Command</td></tr><tr><td>8-11</td><td>DINT</td><td>Rincmd Register ID</td></tr><tr><td>12-15</td><td>DINT</td><td>Rincmd Register Data</td></tr></table></td><td>Get</td></tr>
<tr><td>121</td><td>3</td><td>T->O Simple Operation Result</td><td><table><tr><th>Bytes</th><th>Type</th><th>Description</th></tr><tr><td>0-3</td><td>DINT</td><td>Simple Operation Status Flags
Bit 0: In progress
Bit 1: Invalid operation
Bit 2: Timeout
Bit 3: Error
Bit 4: Success
Bit 5: Connection lost
Bit 6-31: Reserved</td></tr><tr><td>4-7</td><td>DINT</td><td>Operation</td></tr><tr><td>8-11</td><td>DINT</td><td>Data</td></tr></table></td><td>Get</td></tr>
<tr><td>150</td><td>3</td><td>O->T Rincmd Initiate</td><td><table><tr><th>Bytes</th><th>Type</th><th>Description</th></tr><tr><td>0-3</td><td>DINT</td><td>Rincmd Command</td></tr><tr><td>4-7</td><td>DINT</td><td>Rincmd Register ID</td></tr><tr><td>8-11</td><td>DINT</td><td>Rincmd Register Data</td></tr></table></td><td>Get/Set</td></tr>
<tr><td>151</td><td>3</td><td>O->T Simple Operation Initiate</td><td><table><tr><th>Bytes</th><th>Type</th><th>Description</th></tr><tr><td>0-3</td><td>DINT</td><td>Operation</td></tr><tr><td>4-7</td><td>DINT</td><td>Data</td></tr></table>See Section 20.3.3 for the list of supported simple operations, and their required data.</td><td>Get/Set</td></tr>
<tr><td>197</td><td></td><td>Output only heartbeat[1]</td><td>Heartbeat</td><td>N/A</td></tr>
<tr><td>198</td><td></td><td>Input only heartbeat[2]</td><td>Heartbeat</td><td>N/A</td></tr>
<tr><td>199</td><td></td><td>Listen only heartbeat[3]</td><td>Heartbeat</td><td>N/A</td></tr>
</table>

[1]This instance allows Originators (PLCs) to provide output data without having to monitor input data.
[2]This instance allows Originators (PLCs) to monitor input data without providing output data.
[3]This instance allows Originators (PLCs) to monitor input data without providing output data. To use this connection type, an owning connection must exist from a second client and the configuration of the connection must match exactly.

The services provided by the assembly object are as follows:

Service Code	Service Name	Class Level	Instance Level
$0E_H$	GAS	Yes	Yes
01_H	GAA	No	No
10_H	SAS	Yes	Yes

Description of the simple operations supported by instance 151

Operation	Description	Data
0	Idle	-
1	Zero	-
2	Tare	-
3	Preset Tare	Preset tare value
4	Toggle Gross/Net	-
5	Switch To Gross	-
6	Switch To Net	-
7	Keypress (short)	Keycode. See register 0x0008 in Section 18.4 for keycodes.
8	Keypress (long)	Keycode. See register 0x0008 in Section 18.4 for keycodes.
9	Set Outputs	Output bit mask. 0 for inactive, 1 for active.
10	Set I/O Mask	I/O mask for manual drive I/O. 0 for indicator control, 1 for manual control.
1000	Set Target 1	Target value
1001	Set Target 2	Target value
1002	Set Target 3	Target value
1003	Set Target 4	Target value
1004	Set Target 5	Target value
1005	Set Target 6	Target value
1006	Set Target 7	Target value
1007	Set Target 8	Target value
1100	Set Hysteresis 1	Hysteresis value
1101	Set Hysteresis 2	Hysteresis value
1102	Set Hysteresis 3	Hysteresis value
1103	Set Hysteresis 4	Hysteresis value
1104	Set Hysteresis 5	Hysteresis value
1105	Set Hysteresis 6	Hysteresis value
1106	Set Hysteresis 7	Hysteresis value
1107	Set Hysteresis 8	Hysteresis value
1200	Set In-flight 1	In-flight value
1201	Set In-flight 2	In-flight value
1202	Set In-flight 3	In-flight value
1203	Set In-flight 4	In-flight value
1204	Set In-flight 5	In-flight value
1205	Set In-flight 6	In-flight value
1206	Set In-flight 7	In-flight value
1207	Set In-flight 8	In-flight value
1300	Get Target 1	Target value
1301	Get Target 2	Target value
1302	Get Target 3	Target value
1303	Get Target 4	Target value
1304	Get Target 5	Target value

1305	Get Target 6	Target value
1306	Get Target 7	Target value
1307	Get Target 8	Target value
1400	Get Hysteresis 1	Hysteresis value
1401	Get Hysteresis 2	Hysteresis value
1402	Get Hysteresis 3	Hysteresis value
1403	Get Hysteresis 4	Hysteresis value
1404	Get Hysteresis 5	Hysteresis value
1405	Get Hysteresis 6	Hysteresis value
1406	Get Hysteresis 7	Hysteresis value
1407	Get Hysteresis 8	Hysteresis value
1500	Get In-flight 1	In-flight value
1501	Get In-flight 2	In-flight value
1502	Get In-flight 3	In-flight value
1503	Get In-flight 4	In-flight value
1504	Get In-flight 5	In-flight value
1505	Get In-flight 6	In-flight value
1506	Get In-flight 7	In-flight value
1507	Get In-flight 8	In-flight value

Description of the 16bit weight status field

Bit	Description
0	Net
1	Zero Band
2	COZ
3	Motion
4-5	Range
6	Error
7	Underload
8	Overload
9	Hi resolution
10	Held
11-15	Reserved

Description of the 32bit extended status field

Bit	Description
0	Menus active
1	Calibration in progress
2	ADC operation in progress
3	ADC operation success
4-5	Display mode
6-31	Reserved

Description of the 32bit I/O status field

Bit	Description
0-7	I/O 1-8

8-31	Reserved

Operating the Rincmd initiate and result assembly object instances

The PLC should populate the Rincmd initiate structure to initiate a Rincmd message to the C500. The command should be selected from valid Rincmd commands Section 18.2.1, or 0 to clear the previous command. The register ID should be selected from the register list Section 18.4. The data should be set as required for any Rincmd write command, otherwise set to 0 for reads (although not strictly necessary). Once the command has completed, the command, register and data will appear in the Rincmd result instance, along with status flags to indicate the success or otherwise of the rincmd.

The order of writing to initiate a rincmd must be:

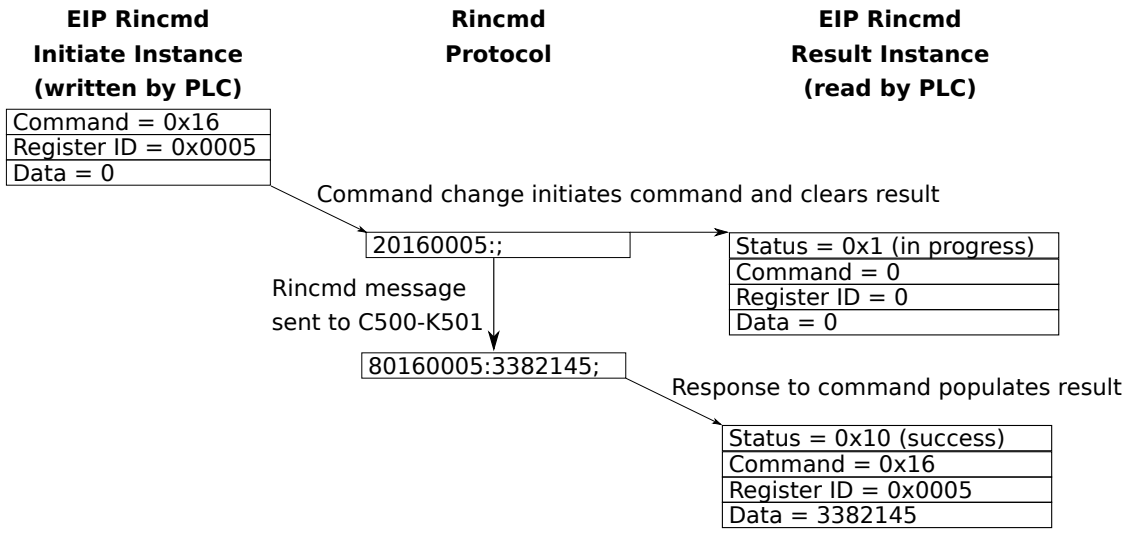
1. Set the initiate instance command to 0
2. Set the initiate instance register ID
3. Set the initiate instance data
4. Set the initiate instance command to the desired command to initiate the rincmd

Polling the rincmd result assembly instance to get the result:

1. While the rincmd is in progress, the in progress flag will be set
2. Once the rincmd has completed, one of the success, or error flags will be set
3. Verify the command and register ID values match those set in the rincmd initiate instance
4. Read the data from the data value

Example: Reading the C500 serial number (rincmd register 0005_{H})

1. Set the initiate instance command to 0
2. Set the initiate instance register ID to 0005_{H}
3. Set the initiate instance data to 0 (not strictly necessary)
4. Set the initiate instance command to 16_{H}Read Final (Decimal)
5. Poll the result instance status flags until the success flag is set
6. Confirm the result instance command matches the initiate instance command (16_{H})
7. Confirm the result instance register ID matches the initiate instance register ID (0005_{H})
8. Read the serial number from the result instance data field



Operating the simple operation initiate and result assembly object instances

The PLC should populate the simple operation initiate structure to initiate a simple operation. The simple operation is translated internally into a rincmd message sent to the C500. The command should be selected from the simple operations Section 20.3.3, or 0 to clear the previous command. The data should be set as required for the simple operation, otherwise set to 0 for reads (although not strictly necessary). Once the command has completed, the command and data will appear in the simple operation result instance, along with status flags to indicate the success or otherwise of the simple operation.

The order of writing to initiate a simple operation must be:

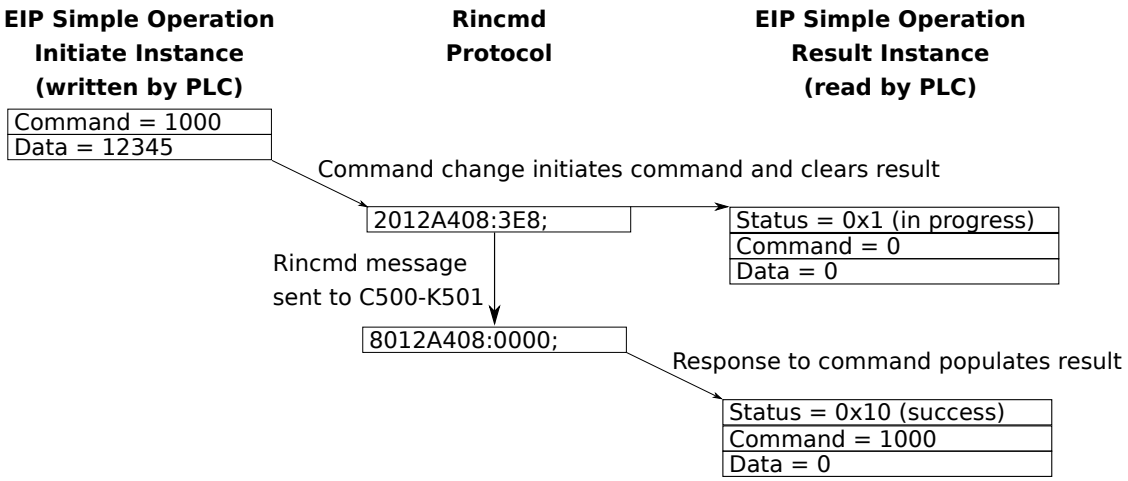
1. Set the initiate instance command to 0
2. Set the initiate instance data
3. Set the initiate instance command to the desired command to initiate the simple operation

Polling the simple operation result assembly instance to get the result:

1. While the simple operation is in progress, the in progress flag will be set
2. Once the simple operation has completed, one of the success, or error flags will be set
3. Verify the command value matches the command set in the simple operation initiate instance
4. Read the data from the data value

Example: Setting the target 1 (simple operation 1000) to value 12345

1. Set the initiate instance command to 0
2. Set the initiate instance data to 12345
3. Set the initiate instance command to 1000 (set target 1)
4. Poll the result instance status flags until the success flag is set
5. Confirm the result instance command matches the initiate instance command (1000)
6. The result data field is not required in this example



20.3.4 Rincmd Protocol Object (Class 65_H)

The following object maps the Rinstrum Rincmd Protocol into the EIP object model. The EIP Instance maps directly to the Rincmd register ID. The EIP Attribute maps to the Rincmd command as follows:

Instance	Attribute ID	Name	Object	EIP Service	Translates to Rincmd command
0 (Class)	1	Revision	UINT (value 1)	GAS	N/A
Rincmd Register ID	1	Read Rincmd Integer Register	DINT	GAS	16_H Read Decimal
Rincmd Register ID	1	Write Rincmd Integer Register	DINT	SAS	17_H Write Decimal
Rincmd Register ID	2	Read Rincmd String Register	STRING	GAS	16_H Read Final (String)
Rincmd Register ID	2	Write Rincmd String Register	STRING	SAS	17_H Write Final (String)
Rincmd Register ID	3	Read Rincmd Register Min Value	DINT	GAS	02_H Read Minimum
Rincmd Register ID	4	Read Rincmd Register Max Value	DINT	GAS	03_H Read Maximum
Rincmd Register ID	5	Read Rincmd Register Type	DINT	GAS	01_H Read Type
Rincmd Register ID	6	Read Rincmd Register Permission	DINT	GAS	$0F_H$ Read Permission

The services provided by the Rincmd protocol object are as follows:

Service Code	Service Name	Class Level	Instance Level
$0E_H$	GAS	Yes	Yes
01_H	GAA	No	No
10_H	SAS	No	Yes

Example: Reading the C500 serial number (rincmd register 0005_H)

Perform a "Get attribute single" read on class 65_H, instance 0005_H, attribute 1 (rincmd register value as integer). Ethernet/IP `GAS C:65H/I:0005H/A:01H` will be translated into rincmd: `20160005:;`.

Example: Reading the C500 model number (rincmd register 0003_H)

Perform a "Get attribute single" read on class 65_H, instance 0003_H, attribute 2 (rincmd register value as string). Ethernet/IP `GAS C:65H/I:0003H/A:02H` will be translated into rincmd: `20160003:;`.

Example: Setting setpoint 1 target in the C500 (rincmd register $A408_H$)

Perform a "Set attribute single" write on class 65_H, instance $A408_H$, attribute 1 (rincmd register value as integer), with data 1234. Ethernet/IP `SAS C:65H/I:A408H/A:01H= 1234` will be translated into rincmd: `2017A408:1234;`.

Note

This method of accessing the Rincmd protocol relies on the PLC supporting 16bit instances. These are widely supported. If your device does not support them, it is recommended you use the Rincmd initiate and result instances within the Assembly object (Section 20.3.3).

20.3.5 TCP Object (Class $F5_H$)

The following table describes the attribute, status, and common services information for the TCP Object.

<table>
<tr><th>Instance</th><th>Attribute ID</th><th>Name</th><th>Object</th><th>Value</th><th>Access</th></tr>
<tr><td>0 (Class)</td><td>1</td><td>Revision</td><td>UINT</td><td>4</td><td>Get</td></tr>
<tr><td>1</td><td>1</td><td>Status</td><td>UDINT</td><td>See ODVA Vol 2: TCP/IP Object Status</td><td>Get</td></tr>
<tr><td>1</td><td>2</td><td>Configuration capability</td><td>UDINT</td><td>See ODVA Vol 2: TCP/IP Object Configuration Capability</td><td>Get</td></tr>
<tr><td>1</td><td>3</td><td>Configuration control</td><td>UDINT</td><td>See ODVA Vol 2: TCP/IP Object Configuration Control</td><td>Get/Set</td></tr>
<tr><td>1</td><td>4</td><td>Physical link object</td><td colspan="2"><table><tr><th>Bytes</th><th>Type</th><th>Description</th><th>Value</th></tr><tr><td>0-1</td><td>UINT</td><td>Path Size</td><td>2 words</td></tr><tr><td>2-n</td><td>UINT Array</td><td>Path</td><td>$20_H F6_H 24_H 01_H$
Class: Ethernet Link $F6_H$
Instance: 1</td></tr></table></td><td>Get</td></tr>
<tr><td>1</td><td>5</td><td>Interface configuration</td><td colspan="2"><table><tr><th>Bytes</th><th>Type</th><th>Description</th><th>Value</th></tr><tr><td>0-3</td><td>UDINT</td><td>IP address</td><td>From C500</td></tr><tr><td>4-7</td><td>UDINT</td><td>Network mask</td><td>From C500</td></tr><tr><td>8-11</td><td>UDINT</td><td>Gateway address</td><td>From C500</td></tr><tr><td>12-15</td><td>UDINT</td><td>Name server 1</td><td>From C500</td></tr><tr><td>16-19</td><td>UDINT</td><td>Name server 2</td><td>From C500</td></tr><tr><td>20-21</td><td>UINT</td><td>Domain name size</td><td>From C500</td></tr><tr><td>22-n</td><td>STRING</td><td>Domain name</td><td>From C500</td></tr></table></td><td>Get/Set</td></tr>
<tr><td>1</td><td>6</td><td>Hostname</td><td colspan="2"><table><tr><th>Bytes</th><th>Type</th><th>Description</th><th>Value</th></tr><tr><td>0-1</td><td>UINT</td><td>Host name size</td><td>From C500</td></tr><tr><td>2-n</td><td>STRING</td><td>Host name</td><td>From C500</td></tr></table></td><td>Get/Set</td></tr>
</table>

The services provided by the TCP object are as follows:

Service Code	Service Name	Class Level	Instance Level
$0E_H$	GAS	Yes	Yes
01_H	GAA	No	Yes
10_H	SAS	No	Yes

20.3.6 Ethernet Link Object (Class $F6_H$)

The following table contains the attribute and common services information for the Ethernet Link Object.

Instance	Attribute ID	Name	Object	Value	Access
0 (Class)	1	Revision	UINT	1	Get
1	1	Interface speed	UDINT	100 Mbps	Get
1	2	Interface flags	UDINT	See ODVA Vol 2: Ethernet Link Object Interface Flags	Get
1	3	Physical address	USINT Array (6)	C500 MAC address	Get

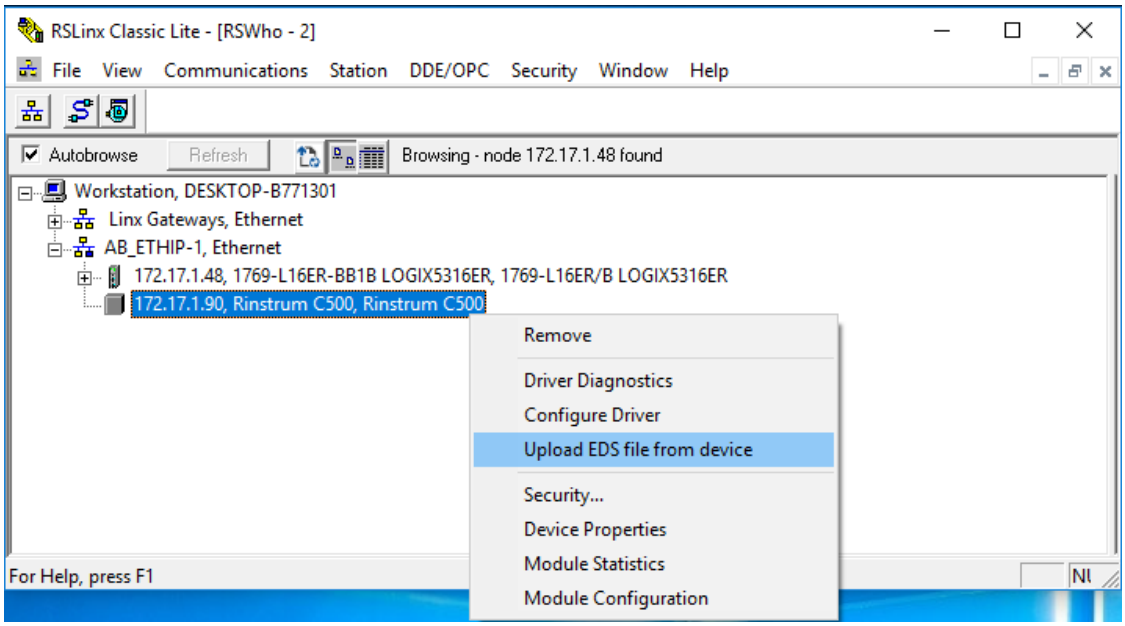
The services provided by the Ethernet Link object are as follows:

Service Code	Service Name	Class Level	Instance Level
$0E_H$	GAS	Yes	Yes
01_H	GAA	No	Yes
10_H	SAS	No	No

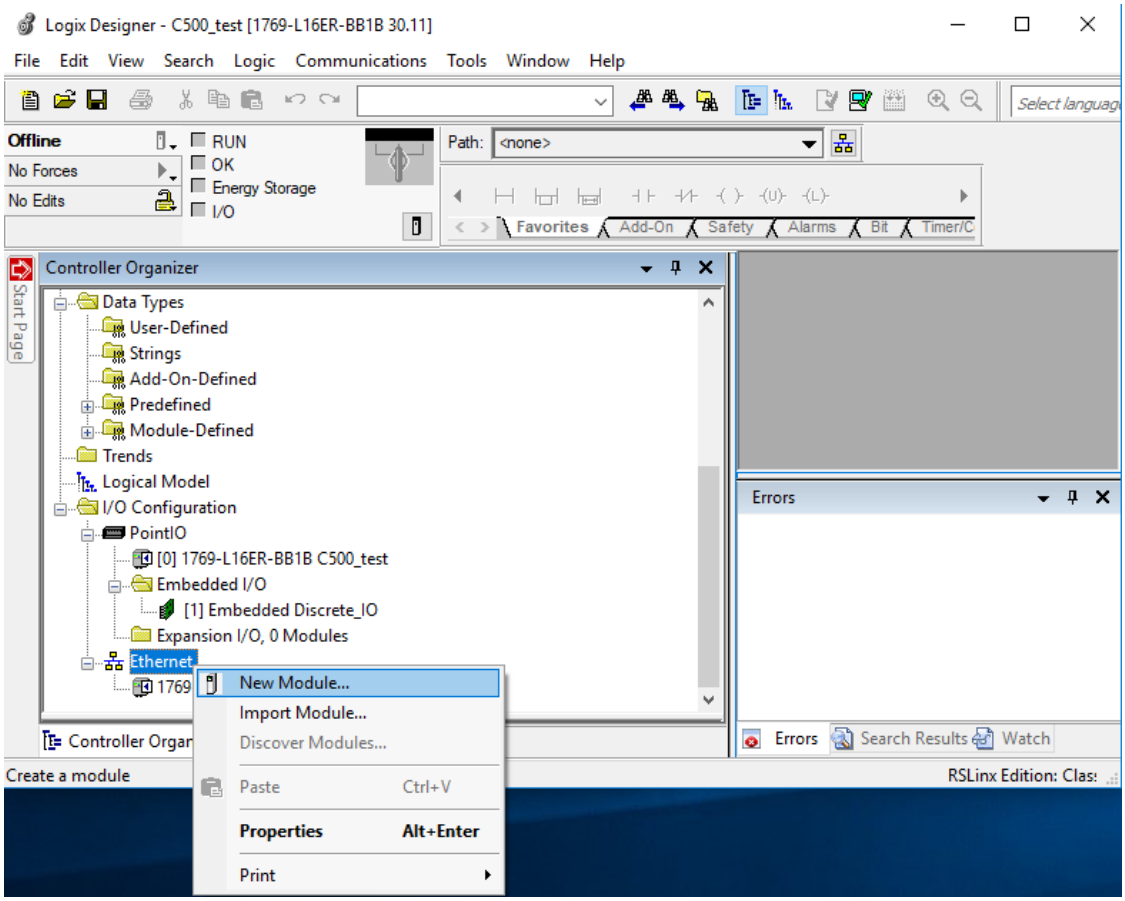
20.3.7 Studio 5000 Example

The following example shows how to set up Rockwell Studio 5000 to retrieve the weight from the C500, perform a tare and then toggle between gross and net modes. This uses both the “Primary Units Simple Weight and Status” and “Simple Operation” connections.

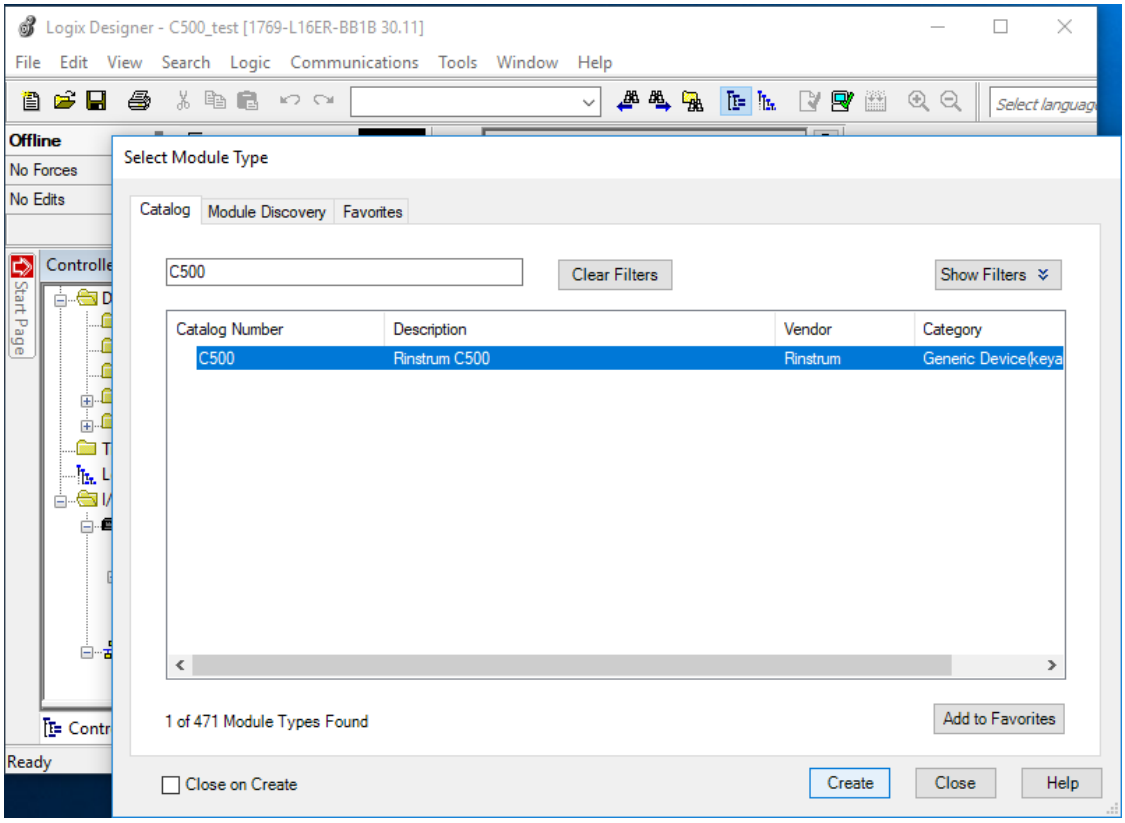
Upload the EDS file from the indicator using RSlinx.



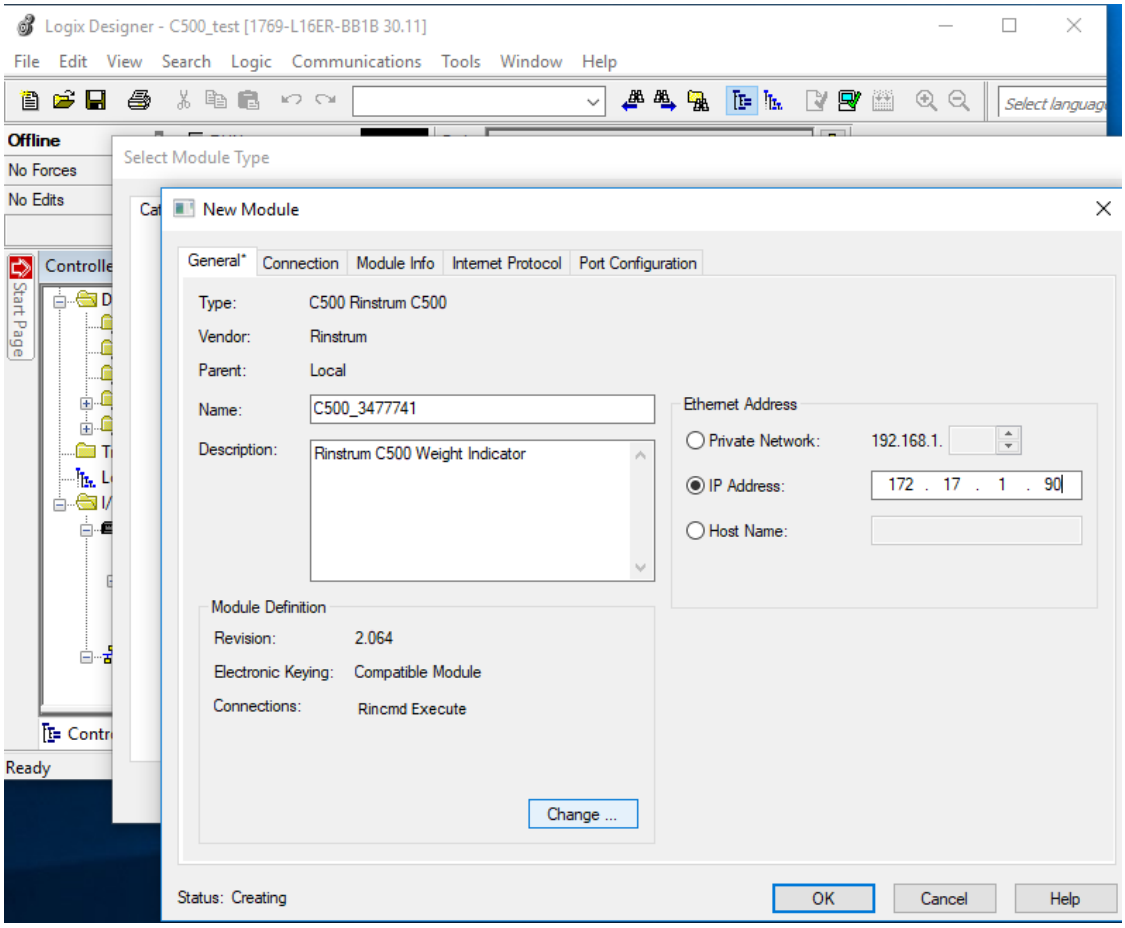
Create a new Logix Designer project and add a new module as shown.



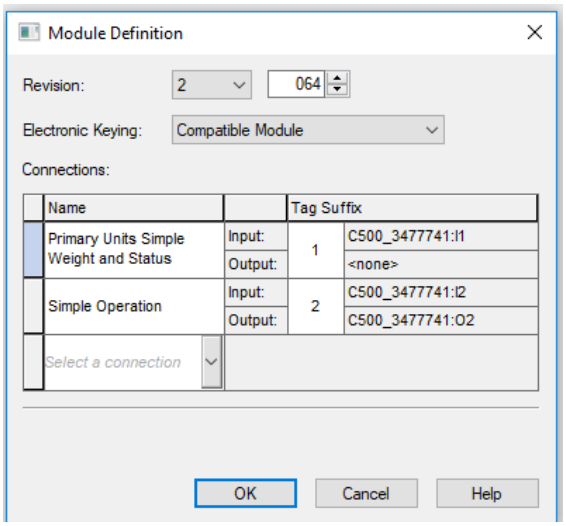
Select “C500” and click create.



Enter the module name, description and IP address then click change.



Select the "Primary Units Simple Weight and Status" and "Simple Operation" connections from the Module Definitions dialog.



The weight (1500) is then available in the `Displayed_Weight` field. It is a gross weight as net status bit is not set (`Status_Bit_NET`).

C500_3477741:I1	{...}	{...}		_05DD:C500_408B115B:I:0
C500_3477741:I1.ConnectionFaulted	0		Decimal	BOOL
C500_3477741:I1.Displayed_Weight	1500		Decimal	DINT
C500_3477741:I1.Primary_Units_Index	2		Decimal	SINT
C500_3477741:I1.Primary_Decimal_Point	3		Decimal	SINT
C500_3477741:I1.Status_Bit_NET	0		Decimal	BOOL
C500_3477741:I1.Status_Bit_ZERO_BAND	0		Decimal	BOOL
C500_3477741:I1.Status_Bit_COZ	0		Decimal	BOOL
C500_3477741:I1.Status_Bit_MOTION	0		Decimal	BOOL
C500_3477741:I1.Status_Bit_RANGE_INDEX_0	0		Decimal	BOOL
C500_3477741:I1.Status_Bit_RANGE_INDEX_1	0		Decimal	BOOL
C500_3477741:I1.Status_Bit_ERROR	0		Decimal	BOOL
C500_3477741:I1.Status_Bit_UNDER	0		Decimal	BOOL
C500_3477741:I1.Status_Bit_OVER	0		Decimal	BOOL
C500_3477741:I1.Status_Bit_HIRES	0		Decimal	BOOL
C500_3477741:I1.Status_Bit_HELD	0		Decimal	BOOL
C500_3477741:I1.Unused_BOOL_bit	0		Decimal	BOOL
C500_3477741:I2	{...}	{...}		_05DD:C500_76F5A266:I:0
C500_3477741:I2.ConnectionFaulted	0		Decimal	BOOL
C500_3477741:I2.Op_IN_PROGRESS	0		Decimal	BOOL
C500_3477741:I2.Op_INVALID_OP	0		Decimal	BOOL
C500_3477741:I2.Op_TIMEOUT	0		Decimal	BOOL
C500_3477741:I2.Op_ERROR	0		Decimal	BOOL
C500_3477741:I2.Op_SUCCESS	0		Decimal	BOOL
C500_3477741:I2.Op_CONN_LOST	0		Decimal	BOOL
C500_3477741:I2.Unused_BOOL_bit	0		Decimal	BOOL
C500_3477741:I2.SimpleOp_Command_In	0		Decimal	DINT
C500_3477741:I2.SimpleOp_Data_In	0		Decimal	DINT
C500_3477741:O2	{...}	{...}		_05DD:C500_C10F89D5:O:0
C500_3477741:O2.SimpleOp_Command_Out	0		Decimal	DINT
C500_3477741:O2.SimpleOp_Data_Out	0		Decimal	DINT

The "Simple Operation" connection can then be used to tare the indicator. Set `SimpleOp_Command_Out` to 2 (from the simple operations table in Section 20.3.3) to perform the tare. `Displayed_Weight` is now 0 and the `Status_Bit_NET` is set.

C500_3477741:I1	{...}	{...}		_05DD:C500_408B115B:I:0
C500_3477741:I1.ConnectionFaulted	0		Decimal	BOOL
C500_3477741:I1.Displayed_Weight	0		Decimal	DINT
C500_3477741:I1.Primary_Units_Index	2		Decimal	SINT
C500_3477741:I1.Primary_Decimal_Point	3		Decimal	SINT
C500_3477741:I1.Status_Bit_NET	1		Decimal	BOOL
C500_3477741:I1.Status_Bit_ZERO_BAND	0		Decimal	BOOL
C500_3477741:I1.Status_Bit_COZ	0		Decimal	BOOL
C500_3477741:I1.Status_Bit_MOTION	0		Decimal	BOOL
C500_3477741:I1.Status_Bit_RANGE_INDEX_0	0		Decimal	BOOL
C500_3477741:I1.Status_Bit_RANGE_INDEX_1	0		Decimal	BOOL
C500_3477741:I1.Status_Bit_ERROR	0		Decimal	BOOL
C500_3477741:I1.Status_Bit_UNDER	0		Decimal	BOOL
C500_3477741:I1.Status_Bit_OVER	0		Decimal	BOOL
C500_3477741:I1.Status_Bit_HIRES	0		Decimal	BOOL
C500_3477741:I1.Status_Bit_HELD	0		Decimal	BOOL
C500_3477741:I1.Unused_BOOL_bit	0		Decimal	BOOL
C500_3477741:I2	{...}	{...}		_05DD:C500_76F5A266:I:0
C500_3477741:I2.ConnectionFaulted	0		Decimal	BOOL
C500_3477741:I2.Op_IN_PROGRESS	0		Decimal	BOOL
C500_3477741:I2.Op_INVALID_OP	0		Decimal	BOOL
C500_3477741:I2.Op_TIMEOUT	0		Decimal	BOOL
C500_3477741:I2.Op_ERROR	0		Decimal	BOOL
C500_3477741:I2.Op_SUCCESS	1		Decimal	BOOL
C500_3477741:I2.Op_CONN_LOST	0		Decimal	BOOL
C500_3477741:I2.Unused_BOOL_bit	0		Decimal	BOOL
C500_3477741:I2.SimpleOp_Command_In	2		Decimal	DINT
C500_3477741:I2.SimpleOp_Data_In	0		Decimal	DINT
C500_3477741:O2	{...}	{...}		_05DD:C500_C10F89D5:O:0
C500_3477741:O2.SimpleOp_Command_Out	2		Decimal	DINT
C500_3477741:O2.SimpleOp_Data_Out	0		Decimal	DINT

To switch back to gross weight, first set the `SimpleOp_Command_Out` to 0. Then set `SimpleOp_Command_Out` to 4 to toggle gross/net. `Displayed_Weight` is now 1500 and the `Status_Bit_NET` is not set.

C500_3477741:I1	{...}	{...}		_05DD:C500_408B115B:I:0
C500_3477741:I1.ConnectionFaulted	0		Decimal	BOOL
C500_3477741:I1.Displayed_Weight	1500		Decimal	DINT
C500_3477741:I1.Primary_Units_Index	2		Decimal	SINT
C500_3477741:I1.Primary_Decimal_Point	3		Decimal	SINT
C500_3477741:I1.Status_Bit_NET	0		Decimal	BOOL
C500_3477741:I1.Status_Bit_ZERO_BAND	0		Decimal	BOOL
C500_3477741:I1.Status_Bit_COZ	0		Decimal	BOOL
C500_3477741:I1.Status_Bit_MOTION	0		Decimal	BOOL
C500_3477741:I1.Status_Bit_RANGE_INDEX_0	0		Decimal	BOOL
C500_3477741:I1.Status_Bit_RANGE_INDEX_1	0		Decimal	BOOL
C500_3477741:I1.Status_Bit_ERROR	0		Decimal	BOOL
C500_3477741:I1.Status_Bit_UNDER	0		Decimal	BOOL
C500_3477741:I1.Status_Bit_OVER	0		Decimal	BOOL
C500_3477741:I1.Status_Bit_HIRES	0		Decimal	BOOL
C500_3477741:I1.Status_Bit_HELD	0		Decimal	BOOL
C500_3477741:I1.Unused_BOOL_bit	0		Decimal	BOOL
C500_3477741:I2	{...}	{...}		_05DD:C500_76F5A266:I:0
C500_3477741:I2.ConnectionFaulted	0		Decimal	BOOL
C500_3477741:I2.Op_IN_PROGRESS	0		Decimal	BOOL
C500_3477741:I2.Op_INVALID_OP	0		Decimal	BOOL
C500_3477741:I2.Op_TIMEOUT	0		Decimal	BOOL
C500_3477741:I2.Op_ERROR	0		Decimal	BOOL
C500_3477741:I2.Op_SUCCESS	1		Decimal	BOOL
C500_3477741:I2.Op_CONN_LOST	0		Decimal	BOOL
C500_3477741:I2.Unused_BOOL_bit	0		Decimal	BOOL
C500_3477741:I2.SimpleOp_Command_In	4		Decimal	DINT
C500_3477741:I2.SimpleOp_Data_In	0		Decimal	DINT
C500_3477741:O2	{...}	{...}		_05DD:C500_C10F89D5:O:0
C500_3477741:O2.SimpleOp_Command_Out	4		Decimal	DINT
C500_3477741:O2.SimpleOp_Data_Out	0		Decimal	DINT

21 Network Commands

21.1 Basic Command Set

The C500 series supports two levels of networking, Basic and Extended. The Basic level allows for simple weight acquisition by PLC or computer from a number of C500s on a simple RS232 or RS422 network. The extended network language allows for full control over all functions of the instrument. The section here describes only the basic command structure. For the Extended structure refer to Section 21.2.

The indicator may be set to respond to commands sent via the serial port from a PLC or PC. The command structure is:

`STX K Command Poll ETX`

where:

- STX (1 byte): 02_H
- K (1 byte): $4B_H$
- Command (1 byte): 1 character command (defined below)
- Poll (2 bytes): the network address (set in `SERIAL:NET.OPT:ADDRESS`). 00 is the broadcast address
- ETX (1 byte): 03_H

Command	Description
z	Zero key press
Z	Zero key long press
T, t	Tare key press
G, g	Gross/net key press
P, q	Print key press
Q	Print key long press
p	Print. This triggers the transmission of the current weight using the format as set in the (Type.A) item of the (SERIAL) group. Use this command to request weight readings on serial port 1. Use the 'P' command to trigger printing from serial port 2.
S, s	Show total
C, c	Clear total
U, u	Undo last print
1, 2, 3, 4	Single transmit on serial 1, 2, 3 or 4
H, h	Hold key press
e	Peak hold key press
E	Peak hold long key press
f	Livestock hold key press
F	Livestock hold long key press

21.1.1 Examples

- To zero the scale of unit 1, send: `STX Kz01 ETX`
- To print, send: `STX KP01 ETX`

21.2 Extended Command Set: Overview

21.2.1 Command

Commands consist of three ASCII-characters (e.g. ADR).

The indicator responds with `0CRLF` to indicate that a command has been accepted, or `?CRLF` to indicate that the command was either not understood or could not be performed. Only the Sxx command and RES command do not have a response.

Some commands will respond with additional failure codes to help diagnose the problem (e.g: CDL, TAR, TAV, TAS and PRN): Response of the commands:

`?CRLF`	Command not understood
`0CRLF`	Command correctly operated
`1CRLF`	Command failed: Scale in motion
`2CRLF`	Command failed: Parameter or scale out of range
`3CRLF`	Command failed: System error
`4CRLF`	Command failed: Device not ready

Note

`CRLF` represents 2 bytes: carriage return ($0D_H$) followed by line feed ($0A_H$).

21.2.2 Query

A query consists of 3 ASCII characters plus a question mark (e.g. `IDN?`).

Normally, a query is a request for information. In this case, the indicator will respond with the information requested. If the query was not understood, the indicator will respond with `?CRLF`.

21.2.3 Parameters

A command or query can be followed by one or more parameters.

Parameters are either numeric (e.g. 3000) or strings (e.g. “Otto”).

String parameters are delimited by quote characters (‘“’ ASCII 34). They are taken literally so that “AbCd” is not the same as “abcd”.

Numeric parameters are variable and leading and trailing spaces are ignored. As a result 003 03 and 3 are identical.

Parameters are separated by a comma character (‘,’ ASCII 44).

Parameters may be left out completely so that it is possible to change one parameter without altering the others. For example `IAD1,,2`; will change the position of the decimal point only.

21.2.4 Termination

Termination characters define the end of a command, query or response.

Permissible termination characters are ‘;’ (ASCII 59), `LF` (ASCII 10), `CRLF` (ASCII 13 10), `LFCR` (ASCII 10 13). E.g. `ADR?;` is the same as `ADR? CRLF`

The indicator always uses CRLF as the termination of its responses.

21.2.5 Initial sequence to start communication

The first command to the indicator should be the select command (see Section 21.3.46). This should be sent even if only one unit is connected.

> **Note**
> S99; will select all connected indicators.

21.2.6 Trade Counter

The indicator does not check to see if new data is different from the old data before incrementing the trade counter, so sending `IAD1,6000` will increment the counter even if the indicator is set up with a fullscale of 6000 kg already. Query the trade counter with command `TDD?` (see Section 21.3.51).

21.3 Extended Command Set: Details

21.3.1 ACL: Auto Calibration

This command no longer performs any function, but is still available for backward compatibility. The C500 series hardware does not need to perform an auto-calibration routine.

Important:

- ACL only valid in INDUSTRIAL mode
- Power loss resets the indicator i.e. ACL1,1 is valid
- Without auto-calibration the accuracy of the instrument will drop over longer periods

General

No. of parameters	2
Save changes	Not possible
Increment Trade Counter	No

Parameter Details

Parameter	Description	Range	Default
1	Auto-calibration at standstill	0 = Off 1 = On	1
2	Auto-calibration at motion	0 = Off 1 = On	1

21.3.2 ADR: Network Address

Set the address of a unit.

General

No. of parameters	2
Save changes	With `TDD1`
Increment Trade Counter	No

Parameter Details

Parameter	Description	Range	Default
1	Address	0 .. 31	31
2	Serial Number	“0000001” .. “9999999”	“xxxxxxx” factory set

Each indicator must be assigned a unique address to enable the implementation of a multi-drop network. This address can be set using the digital setup menus as described in the reference manual. It is also possible to use the network itself to set the unit addresses. The address command is used to assign the unit address via the communications network.

Example 1: Change address of unit from 1 to 2

Command	Answer	Description
S01;		Select unit 1
ADR2;	0 CRLF	Set address to 2
TDD1;	0 CRLF	Save change
S02;		Select new unit 2
IDN?;	"C520","0123456","1.0.4","C520"CRLF	Ask for ID

Example 2: Two units with unknown addresses are configured using their serial numbers.

Command	Answer	Description
S99;		Select all units
ADR01,"123456";	0 CRLF	Unit with serial number "123456" gets address 01
ADR02,"123457";	0 CRLF	Unit with serial number "123457" gets address 02
TDD1;	0 CRLF	Save address against power loss
S01;		Select the new unit 1
ADR?;	1 CRLF	Address is 1
IDN?;	"C520","0123456","1.0.4","C520"CRLF	Ask for ID

21.3.3 AFT: Custom Auto Format

Alters the programmable auto format string. This is the output format used when automatic printing and custom auto format have been selected via the PRS command.

General

No. of parameters	1
Save changes	With TDD1
Increment Trade Counter	No

Parameter Details

Parameter	Description	Range	Default
1	Format String (up to 50 printed characters) Backslash followed by a three digit number for special ASCII characters. See Section 8.2.6 for details.	"string up to 250 characters"	""

This command accepts up to 250 characters, but will truncate the printed string to 50 characters. This allows for long escaped characters to be given. For example, '\002' requires 4 character input of the 250, but will only print a single STX character in the 50 available to print. See Sections 9.4 and 9.5 for printable characters.

Example

Command	Answer	Description
S01;		Select unit 1
AFT?;	""CRLF	Empty format string
AFT"Weight = \200\210 \213";	0 CRLF	New auto format would look like: Weight = 628 kg G
TDD1;	0 CRLF	Save settings

Note

The auto output single start and two end characters are printed at each end of the auto output format. Refer to Section 8.3 for the full list of formatting tokens.

21.3.4 AOC: Analog Output Configuration

Configures the analog output.

General

No. of parameters	1
Save changes	With `TDD1`
Increment Trade Counter	No

Parameter Details

Parameter	Description	Range	Default
1	Analog output type	0 = Current output 1 = Voltage output	0 (Current output)
2	Analog output source	0 = Gross weight 1 = Net weight 2 = Displayed weight 3 = Comms control	2 (Displayed weight)
3	Zero adjustment	-2500 .. 2500	0
4	Span adjustment	-2500 .. 2500	0
5	Force analog output	0 = Normal operation 1 = Force output low 2 = Force output high	0 (Normal operation)
6	Absolute output	0 = Off 1 = On	0 (Off)
7	Clip output	0 = Off 1 = On	0 (Off)

Example

Command	Answer	Description
`S01;`		Select unit 1
`AOC?;`	`0,2,0,0,0,0,0 CRLF`	Get analog output settings
`AOC,,,,2;`	`0 CRLF`	Force analog output high
`AOC,,,40;`	`0 CRLF`	Change span adjustment
`AOC,,,,0;`	`0 CRLF`	Turn off force analog output high
`TDD1;`	`0 CRLF`	Save settings

21.3.5 ASF: Filtering

Set the filtering characteristics of a unit.

General

No. of parameters	2
Save changes	With `TDD1`
Increment Trade Counter	No

Parameter Details

Parameter	Description	Range	Default
1	Number of consecutive readings to average	0 = 1 1 = 2 2 = 3 3 = 4 4 = 5 5 = 6 6 = 7 7 = 8 8 = 9 9 = 10 10 = 25 11 = 50 12 = 75 13 = 100 14 = 200	9
2	Anti-Jitter Setting	0 = off 1 = fine 2 = coarse	0

Example

Command	Answer	Description
`S01;`		Select unit 1
`ASF?;`	`9,0 CRLF`	Query filtering setting
`ASF4,1;`	`0 CRLF`	Changed to a 5 reading average with fine anti-jitter setting
`TDD1;`	`0 CRLF`	Save new settings

21.3.6 BDR: Legacy Serial Port Settings

Replaced by the BDX command. Command doesn't do anything, but is provided for backward compatibility.

General

No. of parameters	4
Save changes	With `TDD1`
Increment Trade Counter	No

Parameter Details

Parameter	Description	Range	Default
1	Baud Rate	1 = 300 2 = 600 3 = 1200 4 = 2400 5 = 4800 6 = 9600 7 = 19200	6
2	Parity	0 = None 1 = Odd 2 = Even	0
3	Data Bits	7, 8	8
4	Stop Bits	1, 2	1
5	Termination Resistors	0 = Off 1 = On	0

Example: Change baud rate settings of unit 1

Command	Answer	Description
`S01;`		Select unit 1
`BDR?;`	`6,0,8,1,0 CRLF`	Query baud rate setting
`BDR4,1,7,1,1;`	`0 CRLF`	Settings changed to 2400 baud, odd parity, 7 data bits, 1 stop bit, termination on.
`TDD1;`	`0 CRLF`	Save new settings

Note

If `BDR` is used to change the settings, the reply is sent using the new settings.

21.3.7 BDX: Serial Port Settings

Set the communication parameters, baud rate, parity, etc.

General

No. of parameters	12
Save changes	With `TDD1`
Increment Trade Counter	No

Parameter Details

Parameter	Description	Range	Default
1	Port	0 = Module 1A 1 = Module 1B 2 = Module 2A 3 = Module 2B 4 = TCP Net 5 = TCP Auto 6 = USB Slave Serial Port 7 = USB Host Serial Port 11 = UDP Outgoing 1 12 = UDP Outgoing 2 13 = UDP Incoming 1 14 = UDP Incoming 2	

Parameter	Description	Range	Default
2	Type	0 = Off 1 = Auto Lo 2 = Auto Hi 3 = Print 4 = Single 5 = Network 6 = 5000 Modbus ASCII Legacy 7 = Modbus ASCII 8 = Modbus RTU Availability: • SER.M.xA: Off - MB.LGCY • SER.M.xB: Off - Single • SER.SLV: Off - Net • SER.HST: Off - MB.RTU • TCP.IN.1: Net (fixed) • TCP.IN.2: Auto Lo - Auto Hi • UDP.x: Off - Net	Default depends on port.
3	Baud Rate	1 = 300 2 = 600 3 = 1200 4 = 2400 5 = 4800 6 = 9600 7 = 19200 8 = 38400 9 = 57600 10 = 115200 Setting applicable to module and USB host serial ports.	6 (9600)
4	Parity	0 = None 1 = Odd 2 = Even Setting applicable to module and USB host serial ports.	0 (None)
5	Data Bits	7, 8 Setting applicable to module and USB host serial ports.	8
6	Stop Bits	1, 2 Setting applicable to module and USB host serial ports.	1
7	Termination Resistors	0 = Off 1 = On Setting applicable to RS485 modules.	0 (Off)

Parameter	Description	Range	Default
8	DTR	0 = Off 1 = On Setting applicable to RS232 modules.	0 (Off)
9	Switch RS232/RS485	0 = RS485 1 = RS232 Legacy setting, not required as module type determines this.	1
10	TCP/UDP Port	TCP Ports: 1024 .. 65535 UDP Ports: 0 (disabled) .. 65535 Setting applicable to TCP and UDP ports.	TCP Net: 2222 TCP Auto: 2223 UDP Ports: 0
11	TCP Timeout	0 (disabled) .. 86400 seconds. Setting applicable to TCP ports.	0
12	UDP Destination IP	0.0.0.0 (disabled) - 255.255.255.255 Setting applicable to outgoing UDP ports.	0.0.0.0

Example: Change baud rate settings of Unit 1, Module 1A to 19200

Command	Answer	Description
`S01;`		Select unit 1
`BDX?0;`	`0,5,6,0,8,1,0,0,1,0,0,"0.0.0.0"CRLF`	Query module 1A settings
`BDX0,,7;`	`0 CRLF`	Baud rate changed
`TDD1;`	`0 CRLF`	Save new settings

21.3.8 BRT: Display Brightness

Adjust the display brightness.

General

No. of parameters	1
Save changes	With `TDD1`
Increment Trade Counter	No

Parameter Details

Parameter	Description	Range	Default
1	Display brightness	0 .. 100	100

Example

Command	Answer	Description
`S01;`		Select unit 1
`BRT?;`	`100 CRLF`	Query setting, brightness at 100%
`BRT50;`	`0 CRLF`	Set brightness to 50%
`TDD1;`	`0 CRLF`	Save new settings

21.3.9 BUZ: Buzzer

Enable/disable the buzzer.

General

No. of parameters	1
Save changes	With `TDD1`
Increment Trade Counter	No

Parameter Details

Parameter	Description	Range	Default
1	Enable	0 = Off 1 = On	1

Example

Command	Answer	Description
S01;		Select unit 1
BUZ?;	1 CRLF	Query setting, buzzer currently enabled
BUZ0;	0 CRLF	Buzzer disabled
TDD1;	0 CRLF	Save new settings

21.3.10 CDL: Zero

Set the zero dead load cancellation. This is analogous with pressing the Zero key on the front of the instrument.

General

No. of parameters	0
Save changes	At input
Increment Trade Counter	No

Example: Set zero dead load of unit 1

Command	Answer	Description
S01;		Select unit 1
CDL;	0 CRLF	Zero dead load set successfully
	<load disturbed >	
CDL;	1 CRLF	Scale in motion

Table of possible replies:

? CRLF	Command not understood
0 CRLF	Command correctly operated
1 CRLF	Scale in motion
2 CRLF	Range out. For example, zero setting out of range
3 CRLF	System error
4 CRLF	Device not ready. For example, printer not ready

21.3.11 CFM: Clock formats

Clock time and date formats.

General

No. of parameters	2
Save changes	With TDD1
Increment Trade Counter	No

Parameter Details

Parameter	Description	Range	Default
1	Time format	0 = 24 HR 1 = 12 HR	0
2	Date format	0 = DD.MM.Y4 1 = MM.DD.Y4 2 = Y4.MM.DD 3 = DD.MM.Y2 4 = MM.DD.Y2 5 = Y2.MM.DD	0

Example

Command	Answer	Description
`S01;`		Select unit 1
`CFM?;`	`1,0 CRLF`	Query setting, 12 hour time and DD/MM/YYYY date
`CFM0,2;`	`0 CRLF`	Set 24 hour time and YYYY/MM/DD date
`TDD1;`	`0 CRLF`	Save new settings

21.3.12 CLK: Date and Time

Set the time and date.

General

No. of parameters	6
Save changes	At input
Increment Trade Counter	No

Parameter Details

Parameter	Description	Range
1	Hour	0 .. 23
2	Minute	0 .. 59
3	Second	0 .. 59
4	Day	1 .. 31
5	Month	1 .. 12
6	Year	0 .. 99 or 1998 .. 2097

Note

When reading `CLK?;`, the year is always returned as short year (0 .. 99). When writing the year using `CLK`, either short year (0 .. 99) or long year (1998 .. 2097) formats can be used.

Example

Command	Answer	Description
`S01;`		Select unit 1
`CLK?;`	`9,20,10,22,6,97 CRLF`	Query current time and date
`CLK10,0,0,29,11,2016;`	`0 CRLF`	Change to 10 am 29/11/16
`CLK10,0,0,29,11,16;`	`0 CRLF`	Change to 10 am 29/11/16

21.3.13 COF: Output Format

Set the output format of the MSV? query.

General

No. of parameters	1
Save changes	With TDD1
Increment Trade Counter	No

Parameter Details

Parameter	Description	Range	Default
1	Format setting	0 .. 12	3

Binary Formats

Format	Data	Order
0	4 Byte (binary) CRLF	MSB before LSB(=00h)
2	2 Byte (binary) CRLF	MSB, LSB
4	4 Byte (binary) CRLF	LSB(=00h) before MSB
6	2 Byte (binary) CRLF	LSB, MSB
8	4 Byte (binary) CRLF	MSB before LSB (=Status)

Detect end of answer only by length. CR and LF can be part of the weighing data.

ASCII Formats

Format	Parameter 1		Parameter 2		Parameter 3		Parameter 4	
1 & 3	Weight (8)							CRLF
5 & 7	Weight (8)	,	Address (2)					CRLF
9 & 10	Weight (8)	,	Address (2)	,	Status (3)			CRLF
11	Weight (8)	,	Address (2)	,	Extended Status (3)			CRLF
12	Weight (8)	,	Address (2)	,	Extended Status (3)	,	I/O Status (3)	CRLF

Values in brackets signify the number of characters in the fixed length response. The weight format is the sign (space or minus), followed by 7 digits 0 .. 9 including the decimal point if used.

STATUS Details

Status	Description	Bit	Comment
001	Overload	0	Weight reading out of range overload or underload
002	Standstill	1	
004	Gross	2	
008	Range 2 active	3	Only with multi-range or multi-interval
016	I/O 1 state	4	Always 0 in format 12. See Output Status Details below for Output status.
032	I/O 2 state	5	
064	I/O 3 state	6	
128	I/O 4 state	7	
256	Centre of Zero	8	This status bit is only available in the extended status contained in formats 11 & 12 only.

Note

The status bits are added together to form the final status. For example a status of 6 (4 + 2) means the weight reading is a gross value with no motion, range 1, and all I/O states are inactive.

Output Status Details

Status	Description	Bit
001	I/O 1 state	0
002	I/O 2 state	1
004	I/O 3 state	2
008	I/O 4 state	3
016	I/O 5 state	4
032	I/O 6 state	5
064	I/O 7 state	6
128	I/O 8 state	7

Example

Command	Answer	Description
`S01;`		Select unit 1
`COF?;`	`3 CRLF`	Query format
`MSV?;`	`-00001.0 CRLF`	Query weight reading
`COF9;`	`0 CRLF`	Change to format 9
`TDD1;`	`0 CRLF`	Save new setting
`MSV?;`	`-00001.0,01,006 CRLF`	Query weight reading using the new format

21.3.14 CWT: Calibration Weight

Set the calibration weight to be used for span calibration. This must be set before using the LWT; or LWN; span calibration commands.

General

No. of parameters	1
Save changes	With `TDD1`
Increment Trade Counter	No

Parameter Details

Parameter	Description	Range	Default
1	Calibration Weight	2% - 100% of full scale weight. Send IAD? to read full scale setting.	3000

Example

Command	Answer	Description
`S01;`		Select unit 1
`CWT?;`	`3000 CRLF`	Query calibration weight setting
`CWT4000;`	`0 CRLF`	Change calibration weight to 4000. Note that weight is sent without any decimal point, so 400.0 kg is sent as 4000 not 400.0.
`TDD1;`	`0 CRLF`	Save new setting

21.3.15 DFT: Default Values

Set the user database, runtime database, calibration, Ethernet settings or licensing database to defaults.

General

No. of parameters	1
Save changes	At input
Increment Trade Counter	Yes (only Calibration)

Parameter Details

Parameter	Description	Range
0	Settings to default	0 = User database 1 = Runtime database 2 = Calibration 3 = Ethernet 4 = Licensing database

Example

Command	Answer	Description
`S01;`		Select unit 1
`DFT1;`	`0 CRLF`	Reset the user database

21.3.16 DPF, DPS: Passcodes

The Safe Passcode protects against misuse by the operator, and prevents access to Safe Setup without entering the correct passcode. It does not block changes to settings from the serial ports. The Full Setup can be accessed normally.

The Full Passcode protects against the access to the Full setup through the keys and blocks the change of all trade relevant items through the serial ports. To prevent attempts to crack the full passcode via the serial ports, the unit will only permit 5 unsuccessful attempts. After this any further attempts to enter the full passcode will be ignored until the unit has been restarted.

Sending the passcode in the free stage defines the passcode and locks the unit, sending the passcode in locked stage opens the unit again until it is deselected.

General

No. of parameters	1
Save changes	With `TDD1`
Increment Trade Counter	No

Parameter Details

Parameter	Description	Range	Default
1	Passcode	000000 - 999999	000000

Note

0 is not a valid passcode, and should instead be written to the parameter to clear the existing passcode.

Example

Command	Answer	Description
S01;		Select unit 1
DPF?;	0 CRLF	No passcode set
DPF123456;	0 CRLF	Full passcode set to 123456. The device is now locked.
DPF?;	1 CRLF	Unit is locked
DPF666666;	? CRLF	Wrong passcode sent
DPF123456;	0 CRLF	Open unit with passcode 123456.
ENU2;	0 CRLF	Weighing unit set to kg
TDD1;	0 CRLF	Save new setting.
S02;		Select unit 2, deselect unit 1
S01;		Select unit 1
ENU1;	? CRLF	Unit 1 is locked.
DPF123456;	0 CRLF	Full passcode set to 123456. The device is now locked.
ENU1;	0 CRLF	Weighing unit set to g
TDD1;	0 CRLF	Save new setting.

21.3.17 ENU: Weight Units

Set the units of weight to be displayed and printed.

General

No. of parameters	1
Save changes	With TDD1
Increment Trade Counter	Yes

Parameter Details

Parameter	Description	Range	Default
1	Weight units	0 = NONE 1 = G 2 = KG (default) 3 = LB 4 = T 5 = OZ 6 = USER	2

Example

Command	Answer	Description
S01;		Select unit 1
ENU?;	2 CRLF	Query units setting
ENU1;	0 CRLF	Change units to grams
TDD1;	0 CRLF	Save new setting

21.3.18 ESR?: Status

Query the error status of the instrument.

General

No. of parameters	1
Save changes	-
Increment Trade Counter	-

Parameter Details

Parameter	Description	Range	Default
1	Select type of status information	0 .. 1	0

The indicator contains both current and latched error status flags. The latched errors are only cleared by resetting the unit (`RES` command or power off). The response string is 5 hexadecimal characters representing the 20 error bits. See Section 24.4.

Example

Command	Answer	Description
`S01;`		Select unit 1
`ESR?;`	`00000 CRLF`	No current errors
`ESR?1;`	`000C0 CRLF`	Positive and negative sense lines were not connected at some time in the past.

21.3.19 ETH: Ethernet

Set the Ethernet port parameters - Internet Protocol (IP) address, network mask, etc.

General

No. of parameters	10
Save changes	ETH; command will save changes
Increment Trade Counter	No

Parameter Details

Parameter	Description	Range	Default
1	DHCP Enable	0 .. 1	1
2	IP address	0.0.0.0 - 255.255.255.255	192.168.1.254
3	Subnet mask	0.0.0.0 - 255.255.255.255	255.255.255.0
4	Gateway address	0.0.0.0 - 255.255.255.255	192.168.1.1
5	DNS1 address	0.0.0.0 - 255.255.255.255	192.168.1.1
6	DNS2 address	0.0.0.0 - 255.255.255.255	0.0.0.0
7	MAC address	(read only)	
8	Hostname	1 .. 64 characters from a-z, 0-9, -	c520-<serial number> or c530-<serial number>
9	Search domain	0 .. 255 characters from a-z, 0-9, -, .	“”
10	DNS3 address	0.0.0.0 - 255.255.255.255	0.0.0.0

Note

Setting the ETH command over the network may result in a lost connection if the IP address changes as a result. If this occurs, use the device finder to re-locate the indicator and re-connect. Alternatively, use the rinlink port to change these parameters.

Example

Command	Answer	Description
`S01;`		Select unit 1
`ETH?;`	`0,"192.168.1.100","255.255.255.0","192.168.1.1", "192.168.1.1","0.0.0.0","70:4A:E4:00:00:01", "c520-3382121","search.domain","0.0.0.0"CRLF`	Read Ethernet settings (static)
`ETH1;`	`0 CRLF`	Enable DHCP
`ETH?;`	`1,"172.17.1.110","255.255.255.0","172.17.1.1", "172.17.1.2","172.17.1.3","70:4A:E4:00:00:01", "c520-3382121","new.search.domain","0.0.0.0"CRLF`	Read Ethernet settings (set by DHCP)

21.3.20 FBT: Front Function Keys

Setup the functions of each of the 3 front function keys or artificially force the execution of the function.

General

No. of parameters	3
Save changes	With `TDD1`
Increment Trade Counter	No

Parameter Details

Parameter	Description	Range	Default
1	Input number	1 .. 3	1
2	Operation	0 = None (NONE) 1 = Print (PRINT) 2 = Show Total (SHW.TOT) 3 = Clear Total (CLR.TOT) 4 = Undo Print (UNDO) 5 = Single Tx Serial 1 (SIN.M1A) 6 = Single Tx Serial 2 (SIN.M1B) 7 = Manual Hold (M.HOLD) 8 = Show Peak (P.HOLD) 9 = Livestock Hold (L.HOLD) 10 = Single Tx Serial 3 (SIN.M2A) 11 = Single Tx Serial 4 (SIN.M2B) 12 = Switch Units (UNITS) 13 = Single Tx Serial 5 (SIN.SLV) 14 = Single Tx Serial 6 (SIN.HST)	-
3	Duration of key press	0 = Short press 1 = Long press	0

Example

Command	Answer	Description
`S01;`		Select unit 1
`FBT?1;`	`0 CRLF`	Get current operation of function key 1 - none at the moment
`FBT1,1;`	`0 CRLF`	Set function key 1 to “Print”
`TDD1;`	`0 CRLF`	Save setting
`FBT1;`	`0 CRLF`	Simulate “Print”
`FBT1,,1;`	`0 CRLF`	Simulate “De-Print” - Long press of print key

21.3.21 FPL: Read display buffer and front key press simulation

Read the contents of the indicator display buffer and simulate key presses of the front keys.

General

No. of parameters	2
Save changes	-
Increment Trade Counter	No

Query Parameter Details

Parameter	Description	Range	Example
1	Display Buffer. Hexadecimal string representing every segment on the indicator display. Used by the viewer to mimic the indicator display.	“string of 32 hexadecimal characters [0-9A-F]”	“0476003F088100000000000000000000”

Command Parameter Details

Parameter	Description	Range	Default
1	Key	1 = Zero key 2 = Tare key 3 = Gross/net key 4 = Function 1 key 5 = Function 2 key 6 = Function 3 key	-
2	Duration of key press	0 = Short press 1 = Long press	0 (short press)

Example

Command	Answer	Description
`S01;`		Select unit 1
`FPL?;`	`"0476003F088100000000000000000000"CRLF`	Get current display buffer
`FPL1;`	`0 CRLF`	Simulate short press of the zero key
`FPL2,0;`	`0 CRLF`	Simulate short press of the tare key
`FPL3,1;`	`0 CRLF`	Simulate long press of the gross/net key (enter setup)

21.3.22 IAD: Scale Build

Set the scale build parameters including max1,e1,max2,e2,decimal point etc.

General

No. of parameters	7
Save changes	With `TDD1`
Increment Trade Counter	Yes

Parameter Details

Parameter	Description	Range	Default
1	Range	1 .. 2	1
2	Maximum Load (max1 or max2)	100 .. 999999	Range 1: 3000 Range 2: 6000
3	Number of right side digits (decimal point position)	0 .. 5	0
4	Resolution (e1 or e2)	1 = 1 2 = 2 3 = 5 4 = 10 5 = 20 6 = 50 7 = 100	Range 1: 1 Range 2: 2
5	x10 mode	0 = Off 1 = On	0
6	Additive Tare	0 .. full scale	0
7	Sense line check	0 = Off 1 = On	1

Note

The full scale weight of the instrument is set to nominal load 1 for single range installations, and nominal load 2 for dual-range and dual-interval installations. In single range installations nominal load 2 is not used.

Example

Command	Answer	Description
`S01;`		Select unit 1
`IAD?1;`	`1,3000,0,1,0,500,1 CRLF`	Current settings are: max1 = 3000, no decimal point, e1 = 1, x10 mode is off, additive tare is 500, sense line check is on
`IAD1,4000,1,2,0,200,0;`	`0 CRLF`	New settings are: max1 = 4000, 1 digit after decimal point, e1 = 2, x10 mode is off, additive tare is 200, sense line check is off
`TDD1;`	`0 CRLF`	Save setting

If `IAD?` is issued without the range parameter then the returned data is range 1 for single range setup or range 2 for dual interval or dual range setup. In this way it is possible to query the maximum load without the need to issue a `WMD?` Command to determine the weighing mode.

21.3.23 ICR: Measurement Rate

Set the fundamental measurement frequency of the instrument.

General

No. of parameters	1
Save changes	With TDD1
Increment Trade Counter	Yes

Parameter Details

Parameter	Description	Range	Default
1	Measurement Rate in Hertz	10 = 10Hz 12 = 12.5Hz 15 = 15Hz 20 = 20Hz 25 = 25Hz 30 = 30Hz 50 = 50Hz 60 = 60Hz 100 = 100Hz	50

Note

Frequencies other than those listed above will be accepted but the indicator will operate at the nearest frequency in the table.

Example

Command	Answer	Description
S01;		Select unit 1
ICR?;	50 CRLF	Query current measurement rate
ICR60;	0 CRLF	Change to 60 Hz
TDD1;	0 CRLF	Save setting

21.3.24 IDN: Identification

Set the unit identification string.

General

No. of parameters	1
Save changes	With TDD1
Increment Trade Counter	No

Parameter Details

Parameter	Description	Range	Default
1	Identification string (15 bytes max)	“<string>”	“C520” or “C530”
2	Serial number string	“0000001” .. “9999999”	Factory set, unique to each unit
3	Software version string	“0.0.0” .. “9.9.9”	
4	Model number string	“C520”	

Note

Only the identification string may be changed. The serial number, version and model number are fixed at the factory and are available for information only by using the IDN? query.

Example

Command	Answer	Description
S01;		Select unit 1
IDN?;	"C520","0123456","000","C520" CRLF	Query current identification
IDN"Site A";	0 CRLF	Change identification string to “Site A”
TDD1;	0 CRLF	Save setting

21.3.25 LBT: Key Locking

Set the operation status of each of the 6 front panel keys.

General

No. of parameters	2
Save changes	With TDD1
Increment Trade Counter	No

Parameter Details

Parameter	Description	Range	Default
1	Key	0 = Zero 1 = Tare 2 = Gross/net 3 = F1 4 = F2 5 = F3	-
2	Operation	0 = Lock 1 = Normal 2 = Immediate	1

Operation of each of the 6 front panel keys may be set independently to the following settings:

- Normal operation is the normal function of the key
- Lock operation means that the key is locked and its normal operation is blocked
- Immediate operation allows for the key function to operate without waiting for stable readings

The gross/net key (2) does not support immediate operation.

Example

Command	Answer	Description
S01;		Select unit 1
LBT?0;	1 CRLF	Zero key is currently set to normal operation
LBT0,0;	0 CRLF	Block operation of the zero key
TDD1;	0 CRLF	Save setting

21.3.26 LDN: Zero Calibration

Calibrate the zero dead weight of the scale. This is the new form of the LDW; command (see Section 21.3.27). Because the C500 series has no concept of normal and direct calibration mode, it was necessary to introduce a new command.

Note

There are two different methods of calibration. Either calibrate with standard weight or direct input of mV/V.

General

No. of parameters	2
Save changes	With `TDD1`
Increment Trade Counter	Yes

Parameter Details

Parameter	Description	Range	Default
1	Zero signal in deci-micro volts per volt. 10000duV/V = 1000uV/V = 1 mV/V	-30000 .. 30000	-
2	Status (see table below)	0 .. 106	-

Calibration Status

Status value	Description
0	Calibration finished successfully
1	Calibration in process (busy)
101	Error zero too high (> 3.0 mV/V) or too low (< -3.0 mV/V). Calibration aborted.
106	Other calibration error. Calibration aborted.

Example: Normal Calibration with Weight

The calibration process takes some time to complete. As a result it is necessary to monitor the calibration process to determine when it is finished. To do this, issue a LDN? query.

Command	Answer	Description
`S01;`		Select unit 1
`LDN;`	`0 CRLF`	Start normal zero calibration
`LDN?;`	`496,1 CRLF`	Query status of the zero calibration process. “1” indicates busy.
`LDN?;`	`496,1 CRLF`	Still busy
`LDN?;`	`396,0 CRLF`	Zero calibration finished. New mV/V zero point updated.
`TDD1;`	`0 CRLF`	Save settings

Example: Calibration in mV/V

The calibration process takes some time to complete. As a result it is necessary to monitor the calibration process to determine when it is finished. To do this, issue a LDN? query.

Command	Answer	Description
`S01;`		Select unit 1
`VAL?;`	`5673 CRLF`	Current reading is 0.5673 mV/V
`LDN5673;`	`0 CRLF`	Zero set to 0.5673 mV/V
`LDN?;`	`496,1 CRLF`	Query status of the zero calibration process. “1” indicates busy.
`LDN?;`	`496,1 CRLF`	Still busy
`LDN?;`	`5673,0 CRLF`	Zero calibration finished. New mV/V zero point updated.
`TDD1;`	`0 CRLF`	Save settings

21.3.27 LDW: Legacy Zero Calibration

Calibrate the zero dead weight of the scale. Due to the removal of the normal/direct calibration mode, this command only supports normal zero calibration. For mV/V calibration see the new LDN; command (see Section 21.3.26).

General

No. of parameters	0
Save changes	With TDD1
Increment Trade Counter	Yes

The calibration process takes some time to complete. As a result it is necessary to monitor the calibration process to determine when it is finished. To do this, issue a LDW? query. Refer to the calibration status table for the LDN command in Section 21.3.26.

Example

Command	Answer	Description
S01;		Select unit 1
LDW;	0 CRLF	Start zero calibration
LDW?;	1 CRLF	Query status of the zero calibration process
LDW?;	1 CRLF	Still busy
LDW?;	0 CRLF	Zero calibration finished
TDD1;	0 CRLF	Save settings

21.3.28 LIC: Linearisation

This command gives access to the dual-point linearisation functions of the indicator. A special query LIC? is available to verify the linearisation correction.

Important: The linearisation changes the entire scale sensitivity without zero and end point. Therefore please double check the scale in critical load ranges.

General

No. of parameters	2
Save changes	With TDD1
Increment Trade Counter	Yes

Parameter Details

Parameter	Description	Range	Default
1	Linearisation point	1 .. 5	1
2	Test weight value without decimal point (none = cancel lin. of this point)	0 .. 999999	-

Query Parameter Details

Parameter	Description	Range
1	Percentage of full scale reading (this value is given as integer - i.e. 24,999 is given as 24)	-100 .. 100
2	Correction (weight value without decimal point x10)	-100000 .. 100000

To clear one of the linearisation points leave the test weight value off.

Example: Scale Build is max1 = 500.0 kg, e1 = 0.1 kg

Command	Answer	Description
S01;		Select unit 1
LIC1;	0 CRLF	Clear linearisation point 1
LIC?1;	0,0 CRLF	No correction for point 1
MSV?;	120.5 CRLF	
LIC1,1200;	0 CRLF	Set linearisation point 1 to correct for the current test weight of 120.0 kg. (Note that weight is sent without any decimal point. So 400.0 kg is send as 4000 not 400.0)
LIC?1;	24,-50 CRLF	Current linearisation is approx. -0.5 kg at approx. 24% of fullscale reading
TDD1;	0 CRLF	Save setting

21.3.29 LIV: Setpoints

Set the parameters for the eight setpoints.

General

No. of parameters	12
Save changes	With TDD1
Increment Trade Counter	No

Parameter Details

Parameter	Description	Range	Default
1	Setpoint number	1 .. 8	-
2	Type	0 = OFF 1 = ON 2 = OVER 3 = UNDER 4 = COZ 5 = ZERO 6 = NET 7 = MOTION 8 = ERROR 9 = BUZZER 10 = W. OUT 11 = W. IN	0
3	Data source	1 = Gross 2 = Net	1
4	Switching direction. Legacy do not use, always returns 1. Use type over or under instead.	1 = Over 2 = Under	1
5	Target Weight	-999999 .. 999999	0
6	Inflight Weight	0 .. 999999	0
7	Hysteresis Weight	0 .. 999999	0
8	Logic	1 = Active High 2 = Active Low	1
9	Lock	0 = Off 1 = On	0
10	Alarm	0 = Off 1 = Single 2 = Dual 3 = Continuous 4 = Flashing	0
11	Timing	0 = LEVEL 1 = EDGE 2 = LATCH 3 = PULSE	0
12	Reset	0 = NONE 1 = SP.RST.A 2 = SP.RST.B	0

Example

Command	Answer	Description
`S01;`		Select unit 1
`LIV?1;`	`1,0,1,1,0,0,0,1,0,0,0,0 CRLF`	Query setpoint 1 parameters
`LIV1,2,1,1,1000,100,10,1,0,0,0,0;`	`0 CRLF`	Change setpoint 1 to: over, gross, over switching, target = 1000, inflight = 100, hysteresis = 10, active high logic, no lock, alarm off, level timing, no reset.
`TDD1;`	`0 CRLF`	Save setting

21.3.30 LIW: General Setpoints

Set the general parameters for the setpoints.

General

No. of parameters	6
Save changes	With `TDD1`
Increment Trade Counter	No

Parameter Details

Parameter	Description	Range	Default
1	Jog on time (milliseconds)	100 .. 60000	100
2	Jog off time (milliseconds)	100 .. 60000	100
3	Jogs in set	1 .. 20	1
4	Max jog sets	0 .. 20	0
5	Feeder type	0 = MULT. 1 = SINGLE	0
6	Delay check (milliseconds)	0 .. 60000	0

Example

Command	Answer	Description
`S01;`		Select unit 1
`LIW?;`	`100,100,1,0,0,0 CRLF`	Query general setpoint parameters
`LIW200,100,5,0,0,0;`	`0 CRLF`	Change to: 200ms jog on, 100ms jog off, 5 jogs in a set, unlimited sets, multiple feeders, no check delay.
`TDD1;`	`0 CRLF`	Save setting

21.3.31 LWN: Span Calibration

Calibrate the span of the scale. This is the new form of the `LWT;` command (see Section 21.3.33). Because the C500 series has no concept of normal and direct calibration mode, it was necessary to introduce a new command.

Note

There are two different methods of calibration. Either calibrate with standard weight or direct input of mV/V.

General

No. of parameters	2
Save changes	With `TDD1`
Increment Trade Counter	Yes

Parameter Details

Parameter	Description	Range	Default
1	Span signal in deci-micro volts per volt. 10000duV/V = 1000uV/V = 1 mV/V	-50000 .. 50000	-
2	Status (see table below)	0 .. 106	-

Calibration Status

Status value	Description
0	Calibration finished successfully
1	Calibration in process (busy)
104	Error span too high (> 5.0 mV/V) or too low (< -5.0 mV/V). Calibration aborted.
106	Other calibration error. Calibration aborted.

Example: Normal Calibration with Weight

The calibration process takes some time to complete. As a result it is necessary to monitor the calibration process to determine when it is finished. To do this, issue a LWN? query.

Command	Answer	Description
S01;		Select unit 1
LWN;	0 CRLF	Start span calibration
LWN?;	496,1 CRLF	Query status of the span calibration process. "1" indicates busy.
LWN?;	496,1 CRLF	Still busy
LWT?;	21471,0 CRLF	Span calibration finished
TDD1;	0 CRLF	Save settings

Example: Calibration in mV/V

The calibration process takes some time to complete. As a result it is necessary to monitor the calibration process to determine when it is finished. To do this, issue a LWN? query.

Command	Answer	Description
S01;		Select unit 1
LWN15000;	0 CRLF	Span set to 1.5000 mV/V
LWN?;	21076,1 CRLF	Query status of the span calibration process. "1" indicates busy.
LWN?;	21076,1 CRLF	Still busy
LWN?;	15000,0 CRLF	Span calibration is finished. Span is 1.5000 mV/V.
TDD1;	0 CRLF	Save settings

21.3.32 LRP: License Packages

License and unlicense packages and query package licence status.

To prevent attempts to crack the licensing via the serial ports, the unit will only permit 3 unsuccessful attempts. After this any further attempts to license a package will be ignored until the unit has been restarted.

General

No. of parameters	5
Save changes	At input
Increment Trade Counter	No

Parameter	Description	Range	Default
1	Package Index	0 .. 49	0
2	Number of Installed Packages	0 .. 50	-
3	Name of Package		-
4	Package Licensed	0 = Package is not licensed 1 = Package is not licensed	-
5	Licence Code	AAAAAA .. ZZZZZZ	-

Command	Answer	Description
S01;		Select unit 1
LRP?0;	0,2,"C500-K501",1,"SWTTSG"CRLF	Package 0 is C500-K501, it is licensed with code SWTTSG. There are 2 packages installed.
LRP?1;	1,2,"L900-500",1,"IAHUZA"CRLF	Package 1 is L900-500, it is licensed with code IAHUZA.
LRP,"L900-500",0;	0 CRLF	Unlicense package L900-500.
LRP?1;	1,2,"L900-500",0,""CRLF	Package 1 is L900-500, it is not licensed.
LRP,"L900-500",1,"IAHUZA";	0 CRLF	License package L900-500 with code IAHUZA.
LRP?1;	1,2,"L900-500",1,"IAHUZA"CRLF	Package 1 is L900-500, it is licensed with code IAHUZA.

21.3.33 LWT: Legacy Span Calibration

Calibrate the span of the scale. Due to the removal of the normal/direct calibration mode, this command only supports normal span calibration. For mV/V calibration see the new LWN; command (see Section 21.3.31).

General

No. of parameters	0
Save changes	With TDD1
Increment Trade Counter	Yes

The calibration process takes some time to complete. As a result it is necessary to monitor the calibration process to determine when it is finished. To do this, issue a LWT? query. Shown below is a list of the possible calibration status responses. Refer to the calibration status table for the LWN command in Section 21.3.31.

Example

Command	Answer	Description
S01;		Select unit 1
LWT;	0 CRLF	Start span calibration
LWT?;	1 CRLF	Query status of the span calibration process
LWT?;	1 CRLF	Still busy
LWT?;	0 CRLF	Span calibration finished
TDD1;	0 CRLF	Save setting

21.3.34 MBS: Modbus settings

Set parameters for Modbus.

General

No. of parameters	2
Save changes	With TDD1
Increment Trade Counter	-

Parameter Details

Parameter	Description	Range	Default
1	Modbus 32bit value endian	0 = Big endian 1 = Little endian	0 (Big endian)
2	Modbus TCP listening port	502 .. 65535	502

Example

Command	Answer	Description
S01;		Select unit 1
MBS?;	0,502 CRLF	Query Modbus settings
MBS1,3000;	0 CRLF	Change to little endian and TCP port 3000
TDD1;	0 CRLF	Save new settings

21.3.35 MSM?: Current Weight for Modbus

Query weight readings for Modbus. This command is not influenced by the COF setting.

General

No. of parameters	1
Save changes	-
Increment Trade Counter	-

Parameter Details

Parameter	Description	Range	Default
1	Type of reading	1 = Displayed weight 2 = Gross weight 3 = Net weight 4 = Number of items in total weight 5 = Total 1 6 = Total 2 7 = Peak weight 8 = Live weight	1 (Displayed weight)

Query Format

Parameter 1		Parameter 2	
Weight (8)	,	Status	CRLF

Values in brackets signify the number of characters in the fixed length response. The weight format is the sign (space or minus), followed by 7 digits 0 .. 9. The decimal point is not included as Modbus the registers are integer only.

Status Details

Status	Description	Bit	Comment
1	Underload	0	Weight reading out of range underload
2	Overload	1	Weight reading out of range overload
4	Motion	2	Scale is in motion
8	Gross	3	Weight is gross
16	Range bit 1	4	Current scale range. Values: 00 = Range 1, 01 = Range 2
32	Range bit 2	5	
64	Centre of Zero	6	Scale is within center of zero
65536	I/O 1 state	16	
131072	I/O 2 state	17	
262144	I/O 3 state	18	
524288	I/O 4 state	19	
1048576	I/O 5 state	20	
2097152	I/O 6 state	21	
4194304	I/O 7 state	22	
8388608	I/O 8 state	23	

Example

Command	Answer	Description
S01;		Select unit 1
MSM?;	002000,12 CRLF	Query displayed weight. Status is gross and motion.
MSM?3;	000100,0 CRLF	Query net weight

21.3.36 MSV?: Current Weight

Query weight readings.

General

No. of parameters	2
Save changes	-
Increment Trade Counter	-

Parameter Details

Parameter	Description	Range	Default
1	Type of reading	1 = Displayed weight 2 = Gross weight 3 = Net weight 4 = Number of items in total weight 5 = Total 1 6 = Total 2 7 = Peak weight 8 = Live weight	-
2	Number of consecutive readings	0 .. 60000 (0 means continuous output)	1

Example

Command	Answer	Description
S01;		Select unit 1
COF3;	0 CRLF	Set output format 3
MSV?;	00200.0 CRLF	Query displayed weight
MSV?2;	00400.0 CRLF	Query gross weight
MSV?2,5;	00400.0 CRLF 00400.1 CRLF 00400.2 CRLF 00400.3 CRLF 00400.4 CRLF CRLF	Query the next 5 consecutive gross weight readings
MSV?,0;	00400.0 CRLF 00400.1 CRLF 00400.2 CRLF	Enable continuous output
STP;		Stop continuous output

Note

CRLF is sent after each reading for the ASCII formats but not for the binary formats. With the binary formats a single CRLF is sent at the end of the response regardless of the number of readings requested. To stop continuous output send a STP; command. During continuous output the indicator will not respond to other commands.

21.3.37 MTD: Motion

Alter the motion settings.

General

No. of parameters	1
Save changes	With TDD1
Increment Trade Counter	Yes

Parameter Details

Parameter	Description	Range	Default
1	Motion setting	0 = OFF 1 = 0.5d in 1sec 2 = 1.0d in 1 sec 3 = 2.0d in 1 sec 4 = 5.0d in 1 sec 5 = 0.5d in 0.5 sec 6 = 1.0d in 0.5 sec 7 = 2.0d in 0.5 sec 8 = 5.0d in 0.5 sec 9 = 0.5d in 0.2 sec 10 = 1.0d in 0.2 sec 11 = 2.0d in 0.2 sec 12 = 5.0d in 0.2 sec	1

Example

Command	Answer	Description
S01;		Select unit 1
MTD?;	1 CRLF	Current motion detection is 0.5 divisions in 1 second
MTD2;	0 CRLF	Set motion detection to 1.0 divisions in 1 second
TDD1;	0 CRLF	Save setting

21.3.38 PFT: Custom Print Format

Set the customer ticket format.

General

No. of parameters	1
Save changes	With TDD1
Increment Trade Counter	No

Parameter Details

Parameter	Description	Range	Default
1	Format String (up to 50 printed characters)	“string up to 250 characters”	“”

This command accepts up to 250 characters, but will truncate the printed string to 50 characters. This allows for long escaped characters to be given. For example, ‘\002’ requires 4 character input of the 250, but will only print a single STX character in the 50 available to print. See Sections 9.4 and 9.5 for printable characters.

Example

Command	Answer	Description
S01;		Select unit 1
PFT?;	""CRLF	Query string
PFT"Weight = \W \E";	0 CRLF	New print ticket would look like: Weight = 127.8 kg G CRLF
TDD1;	0 CRLF	Save settings

21.3.39 POR: External IO

Force digital outputs on or off. This is only enabled for IO lines not configured as setpoints. `POR?` replies with the status of the 8 digital IO lines.

General

No. of parameters	8
Save changes	With `TDD1`
Increment Trade Counter	No

Parameter Details

Parameter	Description	Range	Default
1 - 8	Output 1 - 8	0 or 1	-

Example

Command	Answer	Description
`S01;`		Select unit 1
`POR1,1,1,1,1,1,1,1;`	`0 CRLF`	IO 1 - 8 set as outputs and set to on
`TDD1;`	`0 CRLF`	Save setting
`LIV1,1;`	`0 CRLF`	Setpoint 1 is active
`POR1,1,1,1;`	`?  CRLF`	Not possible as POR not enabled for setpoint 1
`POR,1,1,1;`	`0 CRLF`	Ok, only deactivated setpoint changed
`POR?;`	`1,1,1,1,0,0,0,0 CRLF`	IO 1 - 4 are on, IO 5 - 8 are off

21.3.40 PRS: Print and Automatic Output Settings

General

No. of parameters	10
Save changes	With `TDD1`
Increment Trade Counter	No

Parameter Details

Parameter	Description	Range	Default
1	Mode of Operation. Note: This is a legacy setting. The BDX command should be used.	0 = Off 1 = Auto Low 2 = Print 3 = Single 4 = Auto Hi 5 = Network	2
2	Printing Function	1 = Single 2 = Double 3 = Ticket 4 = Automatic Single 5 = Automatic Double 6 = Automatic Ticket 7 = Total 8 = Automatic Total	1
3	Auto Transmit Format	1 = AUTO.A 2 = AUTO.B 3 = AUTO.C 4 = AUTO.D 5 = CUSTOM 6 = AUTO.F	1
4	Number of columns to pad	0 .. 20	0
5	Number of rows to pad	0 .. 10	0
6	Auto Transmit Source	1 = Display 2 = Gross 3 = Net 4 = Total	1
7	Print interlock	0 = No interlock (NONE) 1 = Motion between printouts (MOTION) 2 = Return to zero between printouts (RET.Z)	0
8	Automatic output start character	0 .. 255	2
9	Automatic output end character 1	0 .. 255	3
10	Automatic output end character 2	0 .. 255	0

Example

Command	Answer	Description
`S01;`		Select unit 1
`PRS?;`	`0,1,1,0,0,1,0,2,3,0 CRLF`	Currently serial 2 is off
`PRS2,3,,2,3;`	`0 CRLF`	Set for ticket printing with 2 columns and 3 rows padding
`TDD1;`	`0 CRLF`	Save setting

21.3.41 PRT: Print

Force the instrument to print using configured printer. This is analogous with pressing the print key on the front of the instrument. Sending a string to the indicator allows free layout of the printout. If the printout could not be performed the reply from the indicator is `?CRLF`.

General

No. of parameters	2
Save changes	-
Increment Trade Counter	-

Command Parameter Details

Parameter	Description	Range	Default
1	Mode of operation	0 = Normal print 1 = Perform displayed weight printout, and reply with print ID, date, time and displayed weight 2 = Same as 1, but using gross weight 3 = Same as 1, but net weight 4 = Same as 1, but tare weight	0
2	String to be printed	• Up to 250 characters • No escaping required for printable ASCII characters (see Section 9.5) • Escape non-printable ASCII characters using a backslash followed by a three digit number (see Section 9.5) • Escape printing tokens using a backslash followed by a letter (see Section 9.4)	-

Table of possible replies:

`?  CRLF`	Command not understood
`0 CRLF`	Command correctly operated
`1 CRLF`	Scale in motion
`2 CRLF`	Range out. For example, zero setting out of range
`3 CRLF`	System error
`4 CRLF`	Device not ready. For example, printer not ready

Query Parameter Details

Parameter	Description	Range
1	Data selection	0 = ID of last printout 1 = Next line of printout as a string in quotation marks. For example, " WEIGHT ". A 2048 byte buffer is used to store printouts. This command returns a single line of a printout. Where lines longer than 100 characters, they are cut into pieces. Each line is discarded after it is sent. An empty string is sent when the buffer is empty.

Note

Poll PRT?0 to determine when new data is available, then use PRT?1 for each line of the printout until an empty string is returned. Non-printable characters are sent using \xxx format. For example, `CRLF` is `"\013\010"`.

Example

Command	Answer	Description
S01;		Select unit 1
PRT;	0 CRLF	Force unit to print using the printer port. This is exactly the same as pushing the print key.
PRT?;	38 CRLF	ID number of the last printout is 38
PRT0,"Weight = \G\010\013";	0 CRLF	Prints: Weight = 100.0 kg G
PRT1,"Weight = \G\010\013";	40,9,20,10,22,6,97,00100.0 CRLF	Prints: Weight = 100.0 kg G Replies with: Printout ID: 40, Hour: 9, Minute: 20, Second: 10, Day: 22, Month: 6, Year: 97, Displayed Weight: 100.0
PRT;	? CRLF	Printout was not successful
PRT;	0 CRLF	Standard printout
PRT?1; PRT?1; PRT?1; PRT?1; PRT?1; PRT?1; PRT?1; PRT?1; PRT?1;	"WEIGHT\013\010"CRLF "TICKET\013\010"CRLF "05/10/94 16:50:12\013\010"CRLF "ID: 000008\013\010"CRLF "T: 654 kg\013\010"CRLF "G: 3654 kg\013\010"CRLF "N: 3000 kg\013\010"CRLF "----------\013\010"CRLF ""CRLF	This enables the entire contents of the printout to be collected line by line.

21.3.42 PST: Print Header

Set the 6 line header for printed tickets.

General

No. of parameters	2
Save changes	With TDD1
Increment Trade Counter	No

Parameter Details

Parameter	Description	Range	Default
1	Line number	1 .. 6	1
2	Line contents (Max 30 characters)	“string up to 30 chars”	“WEIGHT” “TICKET”

Note that while the input may be up to 30 characters, the string returned from this command will be padded with trailing spaces to exactly 30 characters.

Example

Command	Answer	Description
S01;		Select unit 1
PST?1;	" Weight "CRLF	Query line 1 data
PST?2;	" Ticket "CRLF	Query line 2 data
PST1,"Joe Bloggs Pty Ltd";	0 CRLF	Change line 1
PST2,"ph 3312 1234";	0 CRLF	Change line 2
TDD1;	0 CRLF	Save setting

21.3.43 QAF: Maintenance Date

General

No. of parameters	4
Save changes	With TDD1
Increment Trade Counter	No

Parameter Details

Parameter	Description	Range	Default
1	Enable	0 = Off 1 = On	0
2	Day	1 .. 31	1
3	Month	1 .. 12	1
4	Year	0 .. 99 or 1998 .. 2097	2010

Note

When reading QAF?;, the year is always returned as short year (0 .. 99). When writing the year using QAF, either short year (0 .. 99) or long year (1998 .. 2097) formats can be used.

Example

Command	Answer	Description
S01;		Select unit 1
QAF?;	0,15,6,00 CRLF	QA function off, date currently set to 15/6/2000
QAF1,29,11,2016;	0 CRLF	Turn QA function on and set date to 29/11/16
QAF1,29,11,16;	0 CRLF	Turn QA function on and set date to 29/11/16
TDD1;	0 CRLF	Save setting

21.3.44 RBT: Remote Keys

Setup the functions of each of the 8 remote inputs or artificially force the execution of the function. This may be used to implement the extended features without external keys and no need of the option cards.

General

No. of parameters	3
Save changes	With TDD1
Increment Trade Counter	No

Parameter Details

Parameter	Description	Range	Default
1	Input number	1 .. 8	1
2	Operation	0 = None (NONE) 1 = Zero (ZERO) 2 = Tare (TARE) 3 = Gross/Net (GROSS) 4 = Print (PRINT) 5 = Blank (BLANK) 6 = Lock (LOCK) 7 = Show Total (SHW.TOT) 8 = Clear Total (CLR.TOT) 9 = Undo Print (UNDO) 10 = Single Tx Serial 1 (SIN.M1A) 11 = Single Tx Serial 2 (SIN.M1B) 12 = Manual Hold (M.HOLD) 13 = Show Peak (P.HOLD) 14 = Livestock Hold (L.HOLD) 15 = Thumbwheel (TXT.PRN) 16 = Single Tx Serial 3 (SIN.M2A) 17 = Single Tx Serial 4 (SIN.M2B) 18 = Switch Units (UNITS) 19 = Single Tx Serial 5 (SIN.SLV) 20 = Single Tx Serial 6 (SIN.HST) 21 = Setpoint Reset A (SP.RST.A) 22 = Setpoint Reset B (SP.RST.B)	-
3	Duration of key press	0 = Short press 1 = Long press	0

Example

Command	Answer	Description
`S01;`		Select unit 1
`RBT?1;`	`0 CRLF`	Get current operation of input 1, currently none
`RBT1,1;`	`0 CRLF`	Set input 1 to “Zero”
`TDD1;`	`0 CRLF`	Save setting
`RBT1;`	`0 CRLF`	Simulate “Zero”
`RBT1,,1;`	`0 CRLF`	Simulate “De-Zero” - Long press of zero key

21.3.45 RES: Reset

Use this command to simulate a power-on reset.

General

No. of parameters	0
Save changes	-
Increment Trade Counter	-

Example

Command	Answer	Description
`S01;`		Select unit 1
`RES;`	`0 CRLF`	Reset unit

21.3.46 Sxx: Select Address

The Sxx command is used to select one or more units with which to communicate. It should be used under all circumstances as the first command sent to the indicator even if only one instrument is connected.

- S00 to S31: Select a single unit with the matching address 00 to 31.
- S96: De-select all units.
- S97 & S98: All units are selected but none reply to commands. This mode is very useful for blanket commands for an entire network of units.
- S99: Select all units and all respond. S99 is useful when a single unit is connected to the network as it is possible to select this unit regardless of its address setting.

Example

Command	Answer	Description
`S01;`		Select unit 1
`MSV?;`	`00400.0 CRLF`	Query current weight
`S02;`		Select unit 2
`MSV?;`	`00623.5 CRLF`	Query current weight
`S96;`		De-select all units

21.3.47 STP: Stop Continuous Data

Stop continuous weight transmission started by `MSV?,0;` command.

General

No. of parameters	0
Save changes	-
Increment Trade Counter	-

Example

Command	Answer	Description
`S01;`		Select unit 1

`MSV?,0;`	`00400.0 CRLF`
	`00400.1 CRLF`
	`00400.2 CRLF`
	...
		Start continuous data transmission

`STP;`		Stop continuous data transmission

21.3.48 TAR: Tare

Force a tare operation, and read the current tare value.

General

No. of parameters	0 (command) / 1 (query)
Save changes	At input
Increment Trade Counter	No

This command is exactly the same as pressing the tare key on the front of the instrument except that the indicator does not wait for no motion. If the current weight reading is not stable the indicator will return `1 CRLF` and ignore the `TAR` command.

A `TAR?` query will return the currently applied tare value.

Parameter	Description	Range	Default
1	Current tare value (query only)	000000 - 999999	-

Table of possible replies:

`? CRLF`	Command not understood
`0 CRLF`	Command correctly operated
`1 CRLF`	Scale in motion
`2 CRLF`	Range out. For example, zero setting out of range
`3 CRLF`	System error
`4 CRLF`	Device not ready. For example, printer not ready

Example

Command	Answer	Description
`S01;`		Select unit 1
`MSV?;`	`00400.0 CRLF`	Query current weight
`TAR;`	`0 CRLF`	Tare
`MSV?;`	`00000.0 CRLF`	Query current weight reading
`MSV?1;`	`00400.0 CRLF`	Query gross weight
`TAR?;`	`4000 CRLF`	Query current tare weight

21.3.49 TAS: Gross / Net

Select Gross or Net weight display.

General

No. of parameters	1
Save changes	At input
Increment Trade Counter	No

Parameter Details

Parameter	Description	Range	Default
1	Gross or Net	0 = Net 1 = Gross	-

Table of possible answers:

Table of possible replies:

`? CRLF`	Command not understood
`0 CRLF`	Command correctly operated
`1 CRLF`	Scale in motion
`2 CRLF`	Range out. For example, zero setting out of range
`3 CRLF`	System error
`4 CRLF`	Device not ready. For example, printer not ready

Example

Command	Answer	Description
`S01;`		Select unit 1
`MSV?;`	`00200.0 CRLF`	Query current weight
`TAS?;`	`0 CRLF`	Unit is in net mode
`TAS1;`	`0 CRLF`	Switch to Gross weight
`MSV?;`	`00400.0 CRLF`	Query current weight
`TAS?;`	`1 CRLF`	Unit is in gross mode

21.3.50 TAV: Tare Value

Set a numeric tare value directly.

General

No. of parameters	1
Save changes	At input
Increment Trade Counter	No

Parameter Details

Parameter	Description	Range	Default
1	Tare value	0 .. full scale	-

Table of possible replies:

`? CRLF`	Command not understood
`0 CRLF`	Command correctly operated
`1 CRLF`	Scale in motion
`2 CRLF`	Range out. For example, zero setting out of range
`3 CRLF`	System error
`4 CRLF`	Device not ready. For example, printer not ready

Example

Command	Answer	Description
`S01;`		Select unit 1
`MSV?2;`	`00300.0 CRLF`	Query net weight
`TAV?;`	`1000 CRLF`	Tare value is 100.0
`TAV2000;`	`0 CRLF`	Set tare value to 200.0
`MSV?2;`	`00200.0 CRLF`	Query net weight
`TAV?;`	`2000 CRLF`	Tare value is 200.0

21.3.51 TDD: Save/load Setup

Save or restore instrument settings. Query trade counter.

General

No. of parameters	1
Save changes	-
Increment Trade Counter	Yes (TDD0 only)

Parameter Details

Parameter	Description	Range	Default
1	Command	0 = Load ROM default values 1 = Save current settings 2 = Reload previous settings	-

Example

Command	Answer	Description
`S01;`		Select unit 1
`IDN"Site A";`	`0 CRLF`	Set ID string
`TDD1;`	`0 CRLF`	Save settings
`TDD?;`	`123 CRLF`	Trade counter is 123

21.3.52 TXT: Thumb Wheel Text Strings

Store or read text strings for printing with token 152 when selected by IOs configured for thumbwheel operation.

General

No. of parameters	2
Save changes	At input
Increment Trade Counter	No

Parameter Details

Parameter	Description	Range	Default
1	Line	1 .. 9	
2	String	0 .. 40 printable ASCII characters	“TXT LINE n”

Example

Command	Answer	Description
`S01;`		Select unit 1
`TXT?1;`	`1,"TXT LINE 1"CRLF`	Read text line 1
`TXT2,"new text";`	`0 CRLF`	Set text line 2

21.3.53 UNT: Unit switching settings

Unit switching settings.

General

No. of parameters	4
Save changes	With `TDD1`
Increment Trade Counter	No

Parameter Details

Parameter	Description	Range	Default
1	Secondary Units	0 = OFF 1 = G 2 = KG 3 = LB 4 = OZ 5 = CUSTOM	0
2	Tertiary units	0 = OFF 1 = G 2 = KG 3 = LB 4 = OZ	0
3	Custom conversion factor Scaled by factor of 1000 (ie 1000 = 1.000)	1 .. 999999	1000
4	Show ounce weights as pounds : ounces	0 = OFF (default) 1 = ON	0

Example

Command	Answer	Description
`S01;`		Select unit 1
`UNT?;`	`0,0,1000,0 CRLF`	Read unit switching settings
`UNT3,4,1000,1;`	`0 CRLF`	Set secondary units to pounds, teritary units to ounces, 1.000 conversion factor, pounds : ounces on

21.3.54 VAL?: mV/V Value

Reads the current signal in mV/V.

General

No. of parameters	1
Save changes	-
Increment Trade Counter	No

Parameter Details

Parameter	Description	Range	Default
1	Current signal in deci-micro volts per volt. 10000duV/V = 1000uV/V = 1 mV/V	-50000 .. 50000	-

Example

Command	Answer	Description
`S01;`		Select unit 1
`VAL?;`	`5076 CRLF`	Current reading is 0.5076 mV/V

21.3.55 WFP: Which serial port is this

Determine which serial port this connection is on.

General

No. of parameters	1
Save changes	-
Increment Trade Counter	No

Parameter Details

Parameter	Description	Range	Default
1	Port	0 = Module 1 Port A 1 = Module 1 Port B 2 = Module 2 Port A 3 = Module 2 Port B 4 = TCP Network Port 5 = TCP Auto Port 6 = USB Slave Port 7 = USB Host Serial Port 8 = USB Line Printer 9 = Front optical rinlink interface 10 = Rear optical rinlink interface 11 = UDP Outgoing Port 1 12 = UDP Outgoing Port 2 13 = UDP Incoming Port 1 14 = UDP Incoming Port 2	N/A

Example

Command	Answer	Description
`S01;`		Select unit 1
`WFP?;`	`4 CRLF`	Connected to the TCP network port (usually TCP port 2222)
`WFP?;`	`0 CRLF`	Connected to serial module 1, port A

21.3.56 WMD: Weighing Mode

Set the weighting mode of the instrument. This selects between single range, dual range and dual interval weighing modes.

General

No. of parameters	2
Save changes	With `TDD1`
Increment Trade Counter	Yes

Parameter Details

Parameter	Description	Range	Default
1	Weighing mode	1 = Single range 2 = Dual range 3 = Dual interval	1
2	Trade mode	0 = OIML 1 = INDUST 2 = NTEP	1

Use the WMD command to setup the weighing mode of the instrument. This setting is a fundamental scale build parameter and should be used along with the IAD and ICR commands before the unit is calibrated.

Example

Command	Answer	Description
`S01;`		Select unit 1
`WMD?;`	`1,0 CRLF`	Query current weighing mode
`WMD2,1;`	`0 CRLF`	Change to dual range, industrial mode
`WMD?;`	`2,1 CRLF`	Weighing mode is dual range, industrial
`TDD1;`	`0 CRLF`	Save settings

21.3.57 ZST: Zero Settings

Set the various options associated with zero balance.

General

No. of parameters	4
Save changes	With `TDD1`
Increment Trade Counter	Depends on parameter

Parameter Details

Parameter	Description	Range	Default	Trade Counter
1	Zero on startup	0 = Off 1 = On	0	No
2	Zero tracking	0 = Off 1 = 0.5d in 1sec 2 = 1.0d in 1 sec 3 = 2.0d in 1 sec 4 = 5.0d in 1 sec 5 = 0.5d in 0.5 sec 6 = 1.0d in 0.5 sec 7 = 2.0d in 0.5 sec 8 = 5.0d in 0.5 sec 9 = 0.5d in 0.2 sec 10 = 1.0d in 0.2 sec 11 = 2.0d in 0.2 sec 12 = 5.0d in 0.2 sec	0	Yes
3	Zero setting range	1 = -20% .. 20% 2 = -100% .. 100% 3 = -2% .. 2% 4 = -1% .. 3%	3	Yes
4	Zero dead band	0 .. 100000	0	Yes

Example

Command	Answer	Description
`S01;`		Select unit 1
`ZST?;`	`0,0,3,0 CRLF`	Query current zero settings
`ZST1;`	`0 CRLF`	Change to zero on startup
`ZST,,,10;`	`0 CRLF`	Change Zero Dead Band to 10
`ZST?;`	`1,0,3,10 CRLF`	Query new settings
`TDD1;`	`0 CRLF`	Save settings

22 5000 Legacy Modbus ASCII Networking

This section details the 5000 legacy Modbus communications provided by the C500. This has been provided to enable existing 5000 applications using Modbus to be replaced by the C500.

It is recommended that new installations use the better featured Modbus implementation (see Section 19).

The Modbus protocol was originally developed for use by Modicon programmable controllers but has been adopted by many manufacturers as a common protocol standard.

The original Modicon standard specifies two transmissions modes: ASCII or RTU.

The C500 legacy modbus supports the ASCII transmission mode only.

The details of underlying communications protocol are generally looked after by the PLC directly. Please refer to the offical Modbus documentation available from http://modbus.org/specs.php for more information on the protocol.

All that is normally required for setup is to specify the addresses of the registers of interest to your application. Much of the data from the C500 is implemented as 2 consecutive registers to make up a complete 32 bits of data. Unless the C500 has been set up with a fullscale of more than 30000 units it is possible to only access the lower order 16 bits contained in a single register.

Warning!

Be sure to correctly configure the indicator address in `SERIAL:NET.OPT:ADDRES`.
The indicator will not reply to messages that do not match its address.

22.1 Connection of the C500 Network

22.1.1 RS232 and RS422 Connection

Please refer to the documentation supplied with your accessory module (see Section 15).

22.2 Modbus Register Definitions for the 5000

The C500 legacy Modbus implements a number of holding registers, input registers and coils. Generally all that is required to configure your Modbus master to work with the C500 is to specify the address of the C500 unit, and the addresses of the registers that you are interested in.

Following is a list of the C500 Modbus registers.

22.2.1 Holding Registers

Holding registers in the C500 are as follows. Please note that the C500 supports additional setpoints to the 5000. These are accessable via registers 40017 to 40032.

Register	Contents
40001	Target 1 32 bits
40003	Inflight 1 32 bits
40005	Target 2 32 bits
40007	Inflight 2 32 bits
40009	Target 3 32 bits
40011	Inflight 3 32 bits
40013	Target 4 32 bits
40015	Inflight 4 32 bits

40017	Target 5 32 bits
40019	Inflight 5 32 bits
40021	Target 6 32 bits
40023	Inflight 6 32 bits
40025	Target 7 32 bits
40027	Inflight 7 32 bits
40029	Target 8 32 bits
40031	Inflight 8 32 bits

22.2.2 Input Registers

Input Registers in the C500 are as follows:

Register	Contents
30001	Unused (always 0)
30002	Reading Status
30003	Reading Error 32 bits
30005	Displayed Reading 32 bits
30007	Gross Reading 32 bits
30009	Net Reading 32 bits
30011	Number of items in the total 32 bits
30013	Total 1 32 bits
30015	Total 2 32 bits
30017	Peak Displayed Reading 32 bits

22.2.3 Commands (implemented as coils)

Modbus coils are used in the C500 to implement specific operator commands. A single coil is set to trigger the function. These coils are self clearing once the function has been executed. It is possible to configure the operation of the 4 remote keys (See the Reference Manual for details of the available functions). Each of the remote keys has a normal function triggered by a single press and a second function which is triggered by a long 2 second key press. Coils 5..12 simulate the operation of these physical buttons. In this way it is possible to operate these extended functions of the C500 without the need for the setpoint accessory card or the physical buttons.

The C500 commands available as coils are as follows:

Coil	Function
00001	Zero
00002	Tare
00003	Gross/Net
00004	Print
00005	Remote Key 1 function single press
00006	Remote Key 2 function single press
00007	Remote Key 3 function single press
00008	Remote Key 4 function single press
00009	Remote Key 1 function long press
00010	Remote Key 2 function long press
00011	Remote Key 3 function long press
00012	Remote Key 4 function long press

23 Securing the Device

The C500 has been designed with security in mind, however steps must be taken by the user on first use to secure the instrument against malicious actors.

23.1 Setting a Safe and Full Passcode

The safe and full passcode should be configured by the user to prevent access to the device setup menus. Passcodes can be set with the instrument's physical interface using the methods covered in Section 5.1.2, or remotely using Network Commands (see Section 21.3.16).

23.2 Changing the Web Interface Passcode

The web interface controls access to the syslog and the installed packages. This interface must be secured to prevent malicious actors from tampering with the scale operation by removing critical packages.

To change the default password:

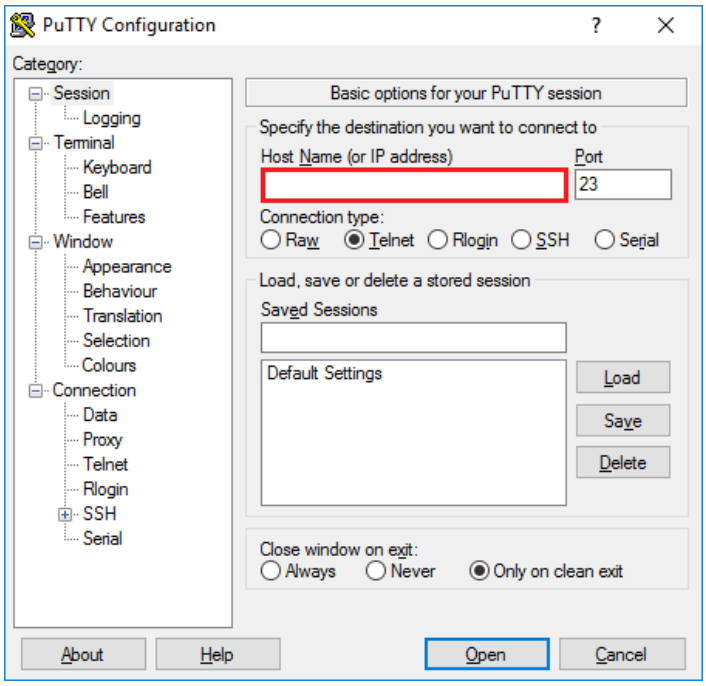
1. Follow the steps outlined in Section 26.2.1 to access the web interface.
2. From the web interface, select 'User Access' to open the password change interface.
3. Enter in the old password. By default, this is 'password'.
4. Enter and re-enter the new password in the appropriate text boxes.
5. Press 'Change password' to confirm the change to the password.

23.3 Securing Against Threats via Network

23.3.1 Changing the User Login Password

If the developer package (L900-500) is installed and telnet access is enabled, the user must change the default password for 'user' to secure this interface.

1. Open your telnet client and connect to the device. For this you will need the device IP address (available via the menus at `ETH.NET:IP`). The recommended telnet client for Windows is Putty.



2. When prompted for `c520-SERIALNUMBER login:`, enter 'user'.
3. When prompted for `password:`, enter the indicator's serial number.
4. The rinstrum logo and some welcome messages will be printed to the screen, before a command prompt `/home/user $`
5. Type into the console 'passwd' and press enter
6. The console will prompt `Old Password:`. Enter the indicator's serial number and press enter.
7. The console will prompt `New Password:`. Enter the new password and press enter.
8. Finally, the console will prompt `Retype password:`. Re-enter the new password and press enter.
9. The password will then be changed.

Warning!

The C500 prevents the user from setting weak passwords. If you encounter any error while changing the password, simply restart the process at Step 2 and continue as before.

23.4 Enabling External Access

It is recommended that the C500 only be connected to secure local networks. However it is recognised that in some cases access from external hostile networks (e.g. the Internet) is required for remote sites. To secure the C500 against threats from unauthorised users the device should be placed behind a firewall on a local network, and a secure VPN should be used to access the local network remotely.

24 Error Messages

A number of error messages may be displayed to warn of operation outside of acceptable limits. These messages are given below. Short messages will appear as a single message on the display. Longer massages will appear on the display in two parts, shown alternately.

24.1 Weighing Errors

These messages show status messages or errors that have occurred during the normal weighing operation.

Error message	Description
`U - - - - -`	Trade mode: The weight is below the zero range (see setting `SCALE:OPTION:Z.RANGE`). Industrial mode: The weight is less than -105% of scale capacity.
`O - - - - -`	Trade mode: The weight is more than 9 divisions over scale capacity. Industrial mode: The weight is more than 105% of scale capacity.
`ZERO ERROR`	The weight reading is beyond the limit set for Zero setting. The operation of the Zero key is limited in the setup during installation. Zero cannot be done at this weight. Use Tare instead.
`STABLE ERROR`	Scale motion has prevented a Zero, Tare or Print operation from occurring on command. Try the operation again once the scale is stable.
`PRINT ERROR`	A printer problem has prevented the printout from being completed. Look for loss of printer power, no paper or cable fault.
`CAL DUE`	Maintenance required. See Section 7.2.4.
`SAVING`	Saving data to the EEPROM.
`PRINT`	Currently operating a print command.

24.2 Setup Errors

These messages warn of setup entries that are not acceptable to the program.

Error message	Description
`RES LO`	The scale build is configured for less than 100 intervals. Check the interval and capacity settings
`RES HIGH`	The scale build is configured for more than 100,000 intervals. Check the interval and capacity settings.
`CHECK TRADE.#`	At least one parameter is not acceptable to OIML requirements. Check the trade relevant items. This is only checked in the TRADE mode when leaving setup. Error numbers: 1. Verification interval e1 or e2 >50 2. Graduations >10000d 3. No weighing unit selected 4. Motion detection is set to NONE 5. Zero tracking not OFF or 0.5d/s 6. Zero setting range not ± 2% or -1% +3% 7. Zero dead band not set to 000000 8. Keys set for instant operation 9. Direct mV/V calibration selected

24.3 Calibration Errors

These messages warn of incorrect calibration technique, or of attempts to calibrate the indicator beyond its specification.

Error message	Description
ZERO HI	The loadcell output is beyond allowable zero calibration range. Check for incorrect scale connection. Reduce the dead load, or shunt the loadcells.
ZERO LO	The loadcell output is below allowable zero calibration range. Check for incorrect scale connection. Increase the dead load, or shunt the loadcells.
SPAN LO	The loadcell signal range (span) is too small for these settings. Incorrect span weight entered. Scale wiring incorrect. Wrong loadcell capacity (too large). Wrong or no calibration weight added to scale.
SPAN HI	The loadcell signal range (span) is too large for these settings. Incorrect span weight entered. Scale wiring incorrect. Loadcell capacity too small for application.
NO ZERO	Zero must be calibrated before span calibration can be attempted.

24.4 System Errors

The condition of the internal circuits are continuously monitored. Faults or out-of-tolerance conditions are shown on the display as an E type error message.

Errors are added together in hexadecimal:

- Example 1: If the power supply voltage is low which causes a low excitation voltage, the error message will be E00005 ($00001_{\mathrm{H}} + 00004_{\mathrm{H}}$).
- Example 2: The loadcell is not connected (i.e. both sense lines are disconnected). This will cause error E000C0 ($00080_{\mathrm{H}} + 00040_{\mathrm{H}}$).

The numbers add in hexadecimal as follows:

1 - 2 - 3 - 4 - 5 - 6 - 7 - 8 - 9 - A - B - C - D - E - F

For example:

$$2_H + 4_H = 6_H$$
$$4_H + 8_H = C_H$$
$$A_H + 2_H = C_H$$

Error	Description	Action
E00001	The power supply voltage is too low	Check power supply and cables
E00002	The power supply voltage is too high	Check power supply and cables
E00004	The positive sense line is not connected	Check the loadcell connection. This error can be deactivated for EX-I applications with OPTION:SENS.CH.
E00008	The negative sense line is not connected	Check the loadcell connection. This error can be deactivated for EX-I applications with OPTION:SENS.CH.
E00010	The temperature is too hot or cold	Check the location
E00020	Accessory module failure	Restart. Replace faulty module
E00080	Resolution error	Recalibrate indicator with valid resolution
E00100	The real time clock has failed	Return for service
E00200	The setup and calibration information has been lost	Recalibrate and reconfigure the unit

E00400	The factory information has been lost	Return for service
E00800	The loadcell excitation voltage is too low	Check the scale
E01000	The loadcell excitation voltage is too high	Check the scale
E02000	The ADC input is out of range	Check the scale
E04000	The runtime information has been lost	Check the scale
E08000	Incorrect software has been installed on the unit	Return for service
E10000	The ADC hardware has failed	Return for service
E20000	Board has not passed factory testing	Return for service

25 Troubleshooting

Symptom	Solution
The weight is not stable	• Check the connection to the loadcells • Ensure that the indicator is connected properly • Check that the resolution is OK • Check the `OPTION:FILTER` setting
There is no communication with a PC using RS232	• Check the port settings in the `SERIAL:SERx` menu • Check that a null modem cable is not used. Pins 6..9 should be disconnected • Ensure that a select (Sxx) command is sent before a command/query
There is no bus communication using RS485	• Ensure that a select (Sxx) command is sent before a command/query • Check that connection is 4 wire • Check the port settings in the `SERIAL:SERx` menu • Check the cable and converter (if used)
Keys do not work	A successful operation of a key is signalled by a short beep. An unsuccessful operation is signalled by a long beep. If a key beeps to acknowledge the press, but does not appear to trigger the desired action, check: 1. The key has been disabled in setup (in the `SPEC:KEY` setting). 2. The weight is outside of limits permitted for that key operation. An error message will be shown. 3. The operation is being blocked by scale instability. (a) Zero, Tare and Print functions require a stable weight. If the scale is not stable when these keys are pressed, the indicator will wait for stability for a maximum of 15s before cancelling the operation and displaying an error message. (b) In industrial mode, it is possible to force these functions to be performed immediately upon key press using the `SPEC:KEY` setting.

26 Upgrading Firmware

The C500 series firmware is field upgradable. There are two ways to upgrade the firmware. The first is via the web interface, and the second is via USB disk and the full setup menus.

Note

If the trade firmware is upgraded the calibration counter will be incremented and an entry will be made in the changelog

Note

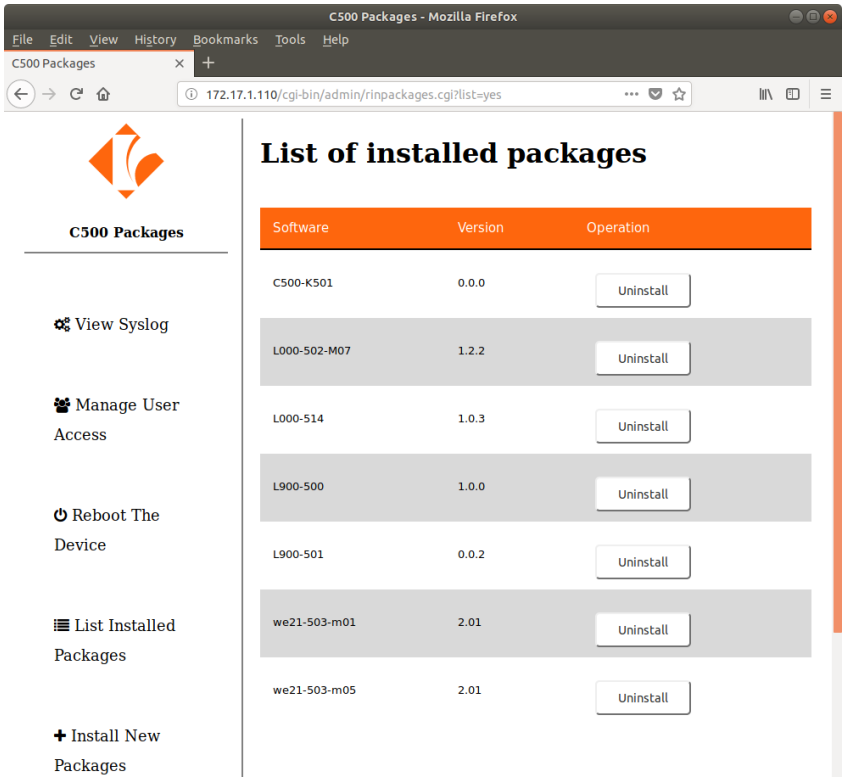
The trade firmware is only compatible with supported web browsers. Current versions of Mozilla Firefox, Google Chrome, Microsoft Edge and Internet Explorer are supported platforms. In additon, Javacript and HTML 5 must be enabled.

26.1 Packages for the C500

The C500 allows for multiple packages to be installed to provide different functionality. Below is a list of packages available for the C500.

Package	Name	Description
Essential packages		
we21-503-m01	File system	Base filesystem for the indicator. Cannot be removed.
we21-503-m05	Operating system kernel	Operating system kernel for the indicator. Cannot be removed.
L000-502-M07	Rinstrum firmware upgrade	Firmware upgrading package to permit packages to be installed, removed and upgraded. Cannot be removed.
C500-K501	C520/C530 main application	Main weighing application. Do not remove.
L000-514	Device finder	Allows indicators to be located on the network in the viewer. Do not remove.
Optional packages		
L900-500	Rinstrum C500 Licensed Developer Package	Allows telnet and FTP connections to the indicator to permit Lua development.
L900-501	Ethernet/IP	Provides access to the indicator via the Ethernet/IP industrial protocol.
L000-521	Rinstrum Lua libraries	Provides Lua interpreter and libraries to access and control the main application via the rincmd protocol.

To see what packages are installed on an indicator access the indicator web interface as explained in Section 26.2.1, then click on the “List Installed Packages” link. The list of installed packages will look like:

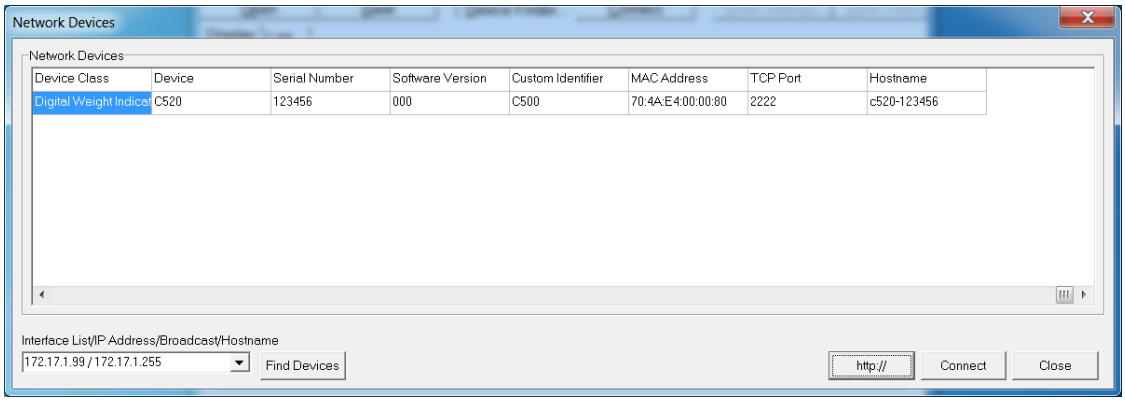


26.2 Upgrading Firmware via the Web Interface

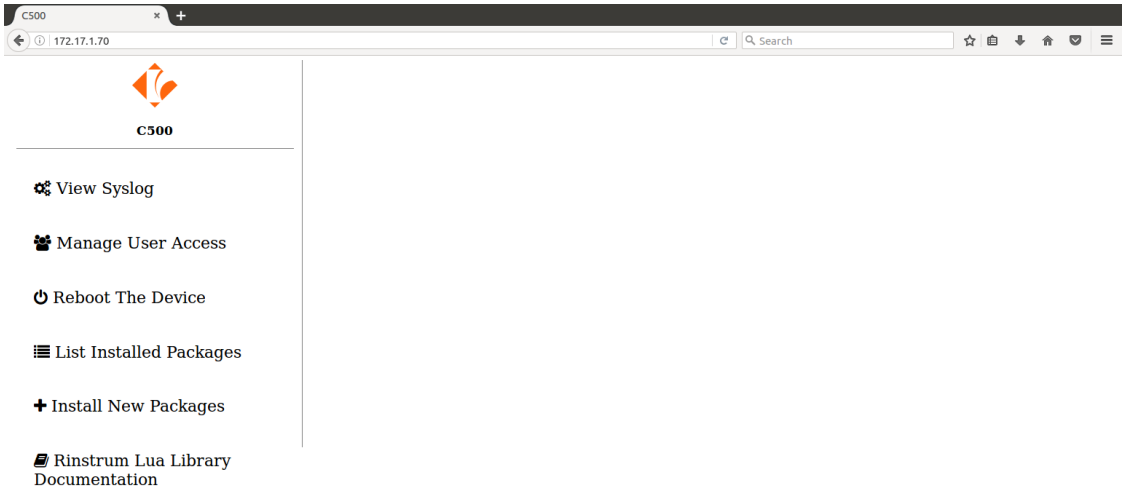
26.2.1 Accessing the Web Interface

Follow the instructions below to access the web interface.

1. Connect the indicator to the network and power up
2. Use the ETH.NET setup menu (see Section 5.6), or Viewer device finder to determine the IP address of the indicator:



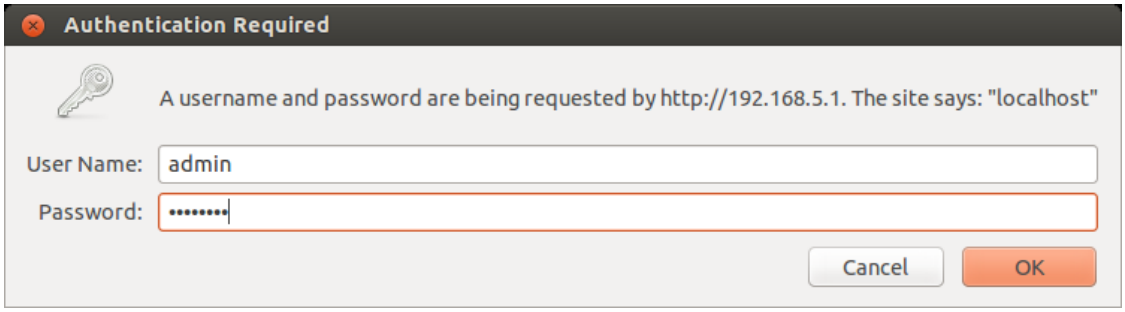
3. Open the IP address in your web brower (clicking the http:// button in the Viewer will do this, otherwise enter it manually in the browser:



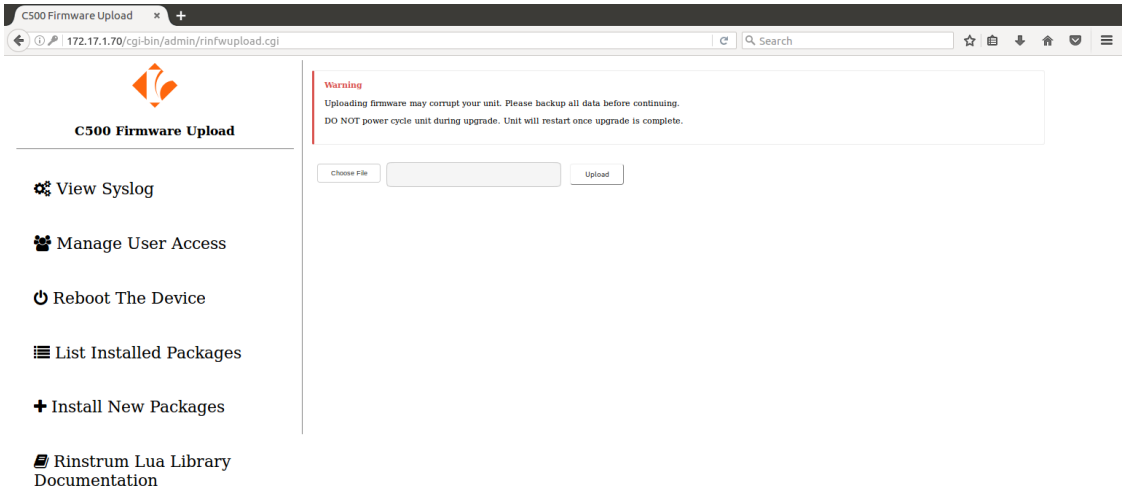
26.2.2 Upgrading the Firmware

Follow the instructions below to upgrade the firmware via the indicator web interface:

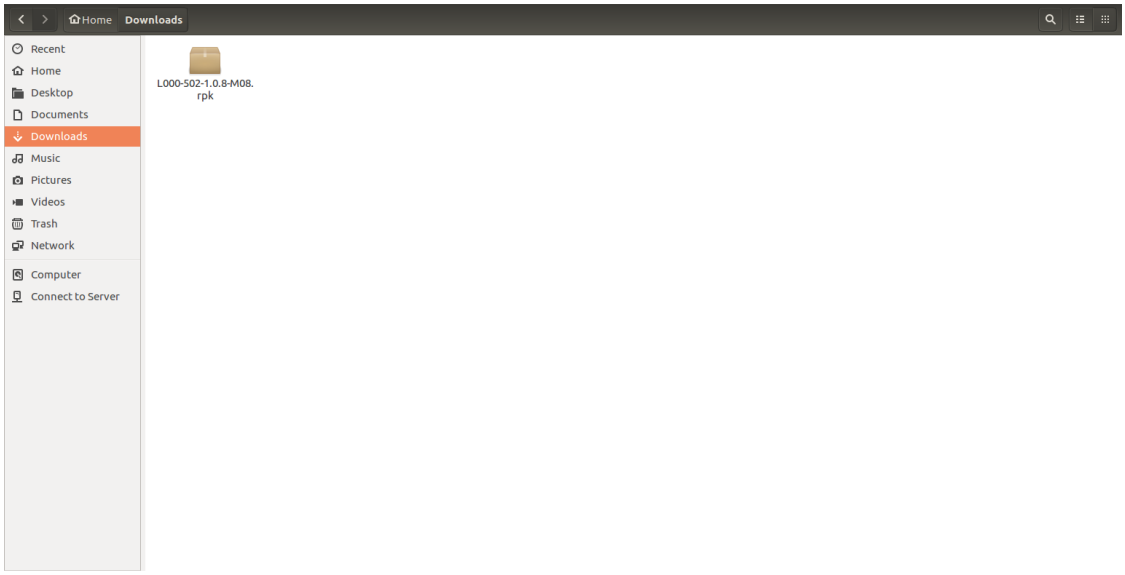
1. Click on the “Install New Packages” link and you will be prompted to enter a username and password. The default username is “admin”, and the default password is “password”. These can be changed from the “User Access” page, and it is recommended that this be done as part of any installation.



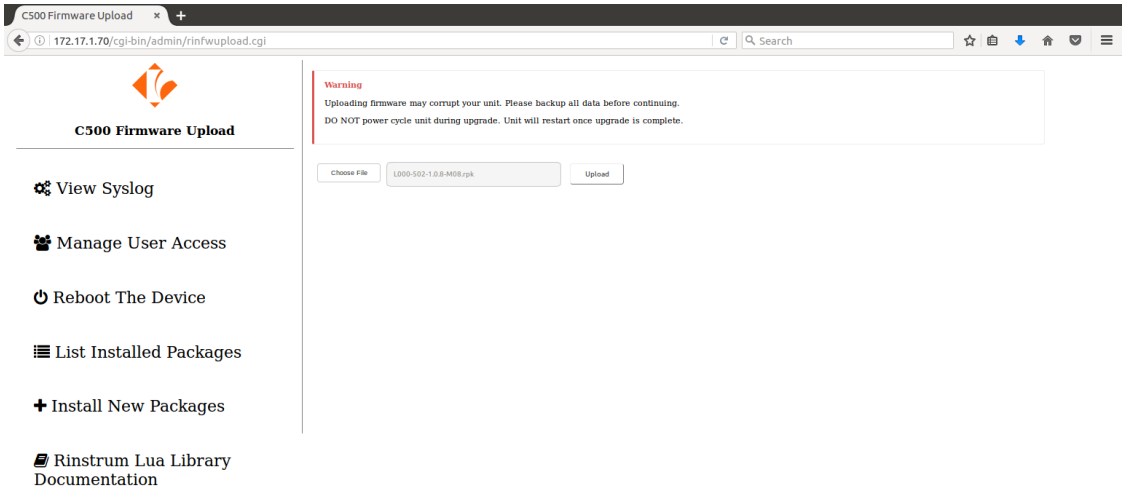
2. The following page will display:



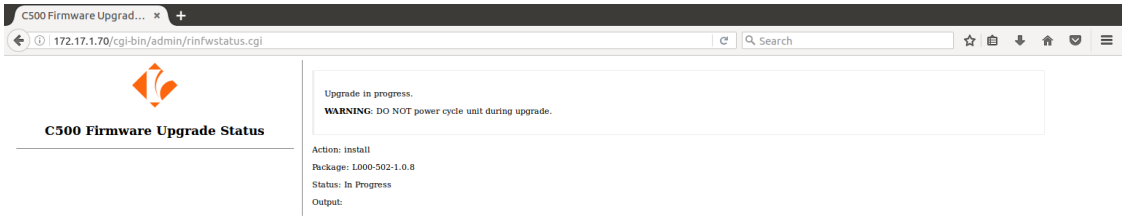
3. Click “Choose File” and select the new firmware file to upload. The file is named as follows: C500-500-<version>-M02.rpk, for example C500-500-1.0.0-M02.rpk.



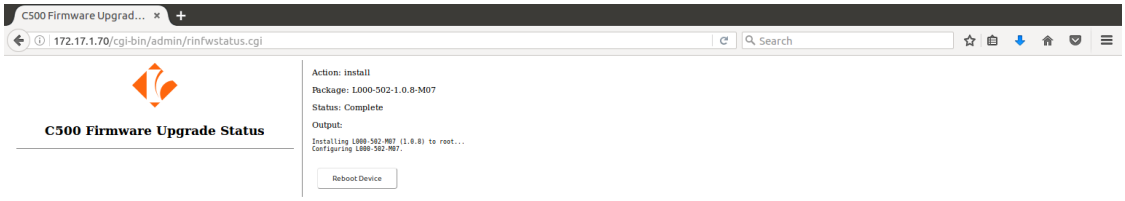
4. Click Open, then click the “Upload” button:



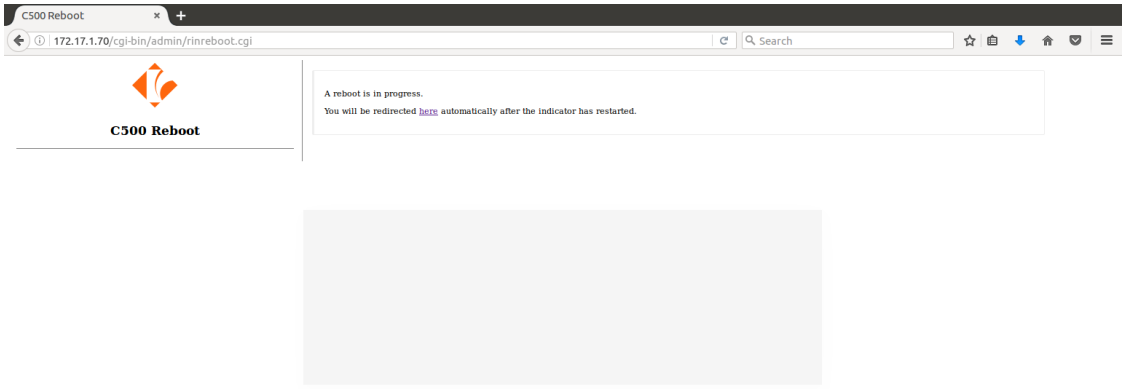
5. The following page will display. Do not disconnect power while the upgrade is in progress. The indicator will display “UPLOAD”, “PROG” during this process.



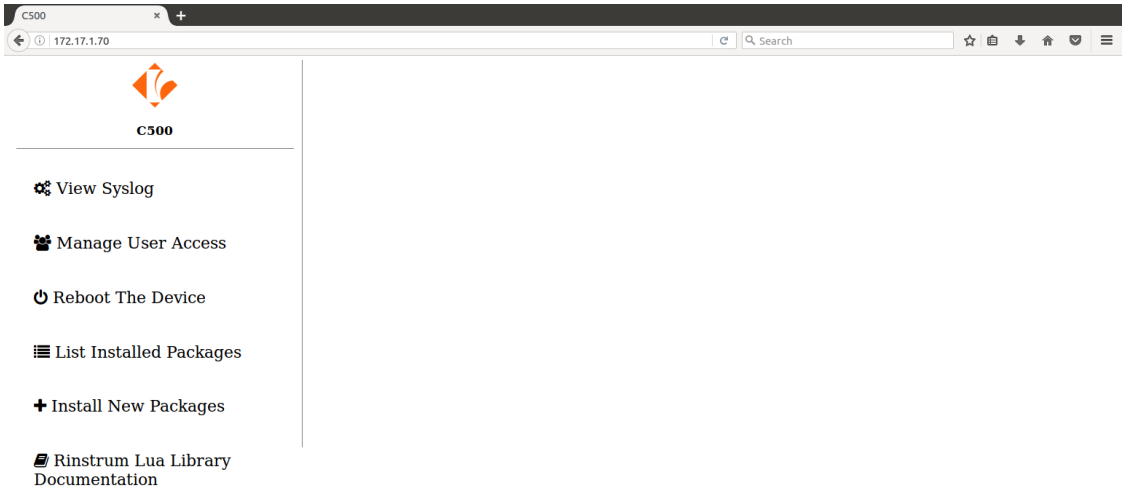
6. Once the upgrade is complete, the following page will display, and the indicator will display “P.DONE”:



7. You should now click “Reboot Device” to reboot the indicator, and start the upgraded firmware.



8. Once the indicator has restarted, the following page will display, and the upgrade is complete:



26.3 Upgrading firmware via the USB host interface

Follow the instructions below to upgrade the firmware via the C500 series full setup menus:

1. Copy the firmware file to a USB flash disk formatted with FAT32 file system. The file must be placed in the root (top level) folder of the disk. There must only be one file with a ".rpk" extension on the disk, as there is no way to select between multiple files in the indicator setup menus. The file is named as follows: C500-500-<version>-M02.rpk, for example C500-500-1.0.0-M02.rpk.

2. Connect the USB disk to the indicator

3. Enter full setup on the indicator

4. Go into the "FILE" menu

5. Select "FW.UPD"

6. The following errors could occur:

   (a) If there is no USB disk connected, or it is not formatted as FAT32 the indicator will display "NO", "DISK"

   (b) If there is no ".rpk" file on the disk, the indicator will display "NO", "RPK"

   (c) If there is more than one ".rpk" file on the disk, the indicator will display "MANY", "RPK"

7. Change to "CONT Y", and press Ok

8. The indicator will display "SAVING" and then "HALTED"

9. Whilst the firmware update is taking place the indicator will display "PROG"

10. Once the firmware update is complete, the indicator will display "P.DONE"

11. You should now restart the indicator, and start the upgraded firmware.