

Helm Instrument Company, Inc. 361 West Dussel Drive Maumee, Ohio 43537 USA 419/ 893-4356 Fax: 419/ 893-1371 www.helminstrument.com

HM-SCM4-WM High Resolution Strain Gage Input Module User Manual 3/10/2021

March, 2021

Force Measurement and Control Solutions

IMPORTANT USER INFORMATION	
PREFACE2	
WHO SHOULD USE	
PURPOSE OF THIS MANUAL	
TECHNIQUES USED IN THIS MANUAL	
PRODUCT SUPPORT	
HARDWARE OVERVIEW4	
HM-SCM4-WM SPECIFICATIONS4	
GETTING STARTED	
REQUIRED TOOLS AND EQUIPMENT	
SETTING IP ADDRESS	
FRONT PANEL	
MODULE I/O CONFIGURATION	
ADDING LADDER PROGRAM9	
MODULE INITIAL SETUP PROCEDURE	
OUTPUT BITS AND CONTROLLER TAGS14	
NORMAL & AUTO CALIBRATION PROCEDURE15	
APPENDIX A: ILLUSTRATION	
APPENDIX B: SYSTEM CONNECTIONS	

IMPORTANT USER INFORMATION

Solid state equipment has operational characteristics differing from those of electromechanical equipment. "Safety Guidelines for the Application, Installation and Maintenance of Solid-State Controls" (Allen-Bradley Publication SGI-1.1) describes some important differences between solid state equipment and hard-wired electromechanical devices. Because of this difference, and also because of the wide variety of uses for solid state equipment, all persons responsible for applying this equipment must satisfy themselves that each intended application of this equipment is acceptable.

In no event will the Helm Instrument Company be responsible or liable for indirect or consequential damages resulting from the use or application of this equipment.

The examples and diagrams in this manual are included solely for illustrative purposes. Because of the many variables and requirements associated with any particular installation, the Helm Instrument Company cannot assume responsibility or liability for actual use based on the examples and diagrams.

No patent liability is assumed by Helm Instrument Company with respect to use of information, circuits, equipment, or software described in this manual.

Throughout this manual we use notes to make you aware of safety considerations.

PREFACE

Read this preface to become familiar with the rest of this manual. This preface covers the following topics:

- Who should use this manual
- The purpose of this manual
- Terms and abbreviations
- Conventions used in this manual
- Helm Instrument support

WHO SHOULD USE?

Use this manual if you are responsible for the design, installation, programming, or maintenance of an automation control system that uses Allen-Bradley small logic controllers.

You should have a basic understanding of ControlLogix products. You should understand electronic process control and be able to interpret the ladder logic instructions required to generate the electronic signals that control your application. If you do not, contact your local Helm representative for the proper training before using this product.

PURPOSE OF THIS MANUAL

This manual is a learning and reference guide for the Helm ControlLogix Strain Gage Input Module. It contains the information you need to install, wire, and use the module.

TECHNIQUES USED IN THIS MANUAL

The following conventions are used throughout this manual:

- Bulleted lists such as this one provides information, not procedural steps.
- Numbered lists provide sequential steps or hierarchical information.

PRODUCT SUPPORT

Contact your Helm representative or call Helm direct at 419-893-4356:

- sales and order support
- product technical training
- warranty support
- support service agreements

Download up to date manuals and ladder logic files at http://www.helminstrument.com/manuals-and-downloads/

Your Questions or Comments on this Manual

If you have any suggestions for how this manual could be made more useful to you, please send us your ideas.

HARDWARE OVERVIEW

The HM-SCM4-WM module fits into any single-slot. It is a Class 1 module.

The module can accept 4 channels of strain gage input. Maximum of 4 load cells at 350 ohms Module configuration requires manual and user programmable setup. The module receives and stores digitally converted analog data into its image table for retrieval.

HM-SCM4-WM SPECIFICATIONS

Type of input	Strain Gage (350-ohm, 700-ohm, 1100-ohm)
Input Impedance	10k
Display Resolution	Up to .0025% of full scale
Overall Module Accuracy	.01% of full scale
Number of Channels	4 (isolated)
Module Update Time	2 millisecond
A/D Conversion Method	Successive Approximation - 18 bit
Normal Mode Rejection: (between +/- input)	116DB CMRR
Amplifier Bandwidth	200 kHz

GETTING STARTED

This chapter can help you to get started using the Helm Strain Gage module. The procedures included here assume that you have a basic understanding of ControlLogix products. You should understand electronic process control and be able to interpret the ladder logic instructions required to generate the electronic signals that control your application.

Because it is a start-up guide, this chapter does not contain detailed explanations about the procedures listed. It does, however, reference other chapters in this book where you can get more information about applying the procedures described in each step. It also references other documentation that may be helpful if you are unfamiliar with programming techniques or system installation requirements. If you have any questions or are unfamiliar with the terms used or concepts presented in the procedural steps, always read the referenced chapters and other recommended documentation before trying to apply the information.

This chapter will:

- tell you what equipment you need
- explain how to install and wire the module
- show you how to calibrate the module

REQUIRED TOOLS AND EQUIPMENT

Have the following tools and equipment ready:

- small blade screwdriver
- programming equipment (All programming examples shown in this manual demonstrate the use of Rockwell RSLogix 5000 Software).

GETTING STARTED CONT.

Use power supply supplied with module. Refer to wiring drawing for connections.

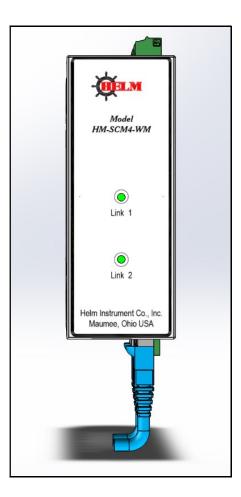
SETTING IP ADDRESS

Unzip hms-ip config tool Install utility Run utility to change IP address

Open Ladder Logic file

- copy all controller tags into project
- copy ladder logic routine "Main Program"

FRONT PANEL



Link Status Indicator Lights

OK light is on (green) when PLC communication is OK.

MODULE I/O CONFIGURATION

This shows the preliminary setup and operation required before the module can function in a 1756 I/O system using RSLogix5000

Adding Module to I/O Configuration

Select Generic Ethernet Module from *Select Module Type* window.

generic	Clear Filters		Show Filters 📚
Catalog Number	Description	Vendor	Category
ETHERNET-BRIDGE	Generic EtherNet/IP CIP Bridge	Allen-Bradley	Communication
ETHERNET-MODULE	Generic Ethernet Module	Allen-Bradley	Communication

Configuration Module's Properties

From the Controller Organizer, right click on the added module and open up Module Properties windows

Module Properties Report: AB1756ENBT (ETHER	RNET-MODULE 1.1)
General Connection Module Info Type: ETHERNET-MODULE Generic Ethernel Vendor: Allen-Bradley Parent: AB1756ENBT Name: nicktest Description: Image: Connection in the section in the s	Connection Parameters Assembly Instance: Size: Input: 100 4 = (32-bit)
Comm Format: Data - DINT Comm Format: Data - DINT Address / Host Name IP Address: 10 . 31 . 16 . 102 C Host Name:	Output: 150 1 (22-bit) Configuration: 3 0 (8-bit) Status Input:
Status: Offline OK	Cancel Apply Help

Type in a name for the module, select a slot number.

General: Connection Parameters

	Assembly	Size
	Instance	
Input	100	4
Output	150	1
Configuration	3	0

Connection: Requested Packet Interval (RPI): 2.0ms

Module Properties Report: AB1756ENBT (ETHERNET-MODULE 1.1)	×
General Connection Module Info	
Requested Packet Interval (RPI): 2.0 ms (1.0 - 3200.0 ms) Inhibit Module Major Fault On Controller If Connection Fails While in Run Mode Use Unicast Connection over EtherNet/IP	
Module Fault	
Status: Offline OK Cancel Apply He	lp

ADDING LADDER PROGRAM

OUTPUT BIT TAGS

Bit Tags Output	Data Type	Description
output_bits	INT	
output_bits.0	BOOL	cal mode
output_bits.1	BOOL	run mode
output_bits.2	BOOL	reset average
output_bits.3	BOOL	ch1 clear tare
output_bits.4	BOOL	ch1 tare
output_bits.5	BOOL	ch2 clear tare
output_bits.6	BOOL	ch2 tare
output_bits.7	BOOL	ch3 clear tare
output_bits.8	BOOL	ch3 tare
output_bits.9	BOOL	ch4 clear tare
output_bits.10	BOOL	ch4 tare
output_bits.11	BOOL	autocal enable ch1
output_bits.12	BOOL	autocal enable ch2
output_bits.13	BOOL	autocal enable ch3
output_bits.14	BOOL	autocal enable ch4
output_bits.15	BOOL	
set_average_count	DINT	limit 1 to 500

Run Mode Bit: When reading or downloading the module's configuration data using. Data**[0].0** and **.Data[0].2**, this bit needs to be at 0. For any other operation, such as reading weigh value, leave the bit at 1.

Cal Mode: Use for diagnostics, no scale, raw A/D values.

Weigh Mode: Scaled value.

Clear Tare Offset Bit: Resets or removes tare value from module for the channel selected.

Set Tare: Set tare values.

CH1 Auto-tune: Set to known weight.

CH2 Auto-tune: Set to known weight.

CONTROLLER TAGS

Controller Tags	
auto_tare	INT
ch1_autocal_mv_v	REAL
ch1_known_weight	DINT
ch1_scale_set	DINT
ch1adtrimset	DINT
ch1average	DINT
ch1weigh	DINT
ch2_autocal_mv_v	REAL
ch2_known_weight	DINT
ch2_scale_set	DINT
ch2adtrimset	DINT
ch2average	DINT
ch2weigh	DINT
ch3_autocal_mv_v	REAL
ch3_known_weight	DINT
ch3_scale_set	DINT
ch3adtrimset	DINT
ch3average	DINT
ch3weigh	DINT
ch4_autocal_mv_v	REAL
ch4_known_weight	DINT
ch4_scale_set	DINT
ch4adtrimset	DINT
ch4average	DINT
ch4weigh	DINT
MSF1	INT
MSF2	INT
MSF3	INT
MSF4	INT
set_mv_v_ch1	REAL
set_mv_v_ch2	REAL
set_mv_v_ch3	REAL
set_mv_v_ch4	REAL

[NOP]-
SCM4_WM_DINT
SCM4_WM_DINT SCM4WM1 SetCalMode output_bits.0 0
SetRunMode output_bits.1
SetScaleChx ch1_scale_set 100000
Set_mV_V_Chx set_mv_v_ch1 2.0
SetRefChx ch1_known_weight 100000
AutoCalEnChx output_bits.11 0
Sample_Set set_average_count
Chx_mV_V_Out ch1_autocal_mv_v 2.0172822
set_ch_adtrim ch1adtrimset 108000
resetaverage output_bits.2
chx_tare output_bits.4
chx_cleartare output_bits.3
sample_rpi sample_rate
auto_tare_band auto_tare 0 getvaluein nicktest:I.Data[0]
msf msf1
0 getvalueout ch1weigh
-46
SCM4_WM_DINT
SCM4_WM_DINT SCM4WM2 SetCalMode output_bits.0
0 SetRunMode output_bits.1
SetScaleChx ch2_scale_set 100000
Set_mV_V_Chx set_mv_v_ch2 2.0
SetRefChx ch2_known_weight 100000
AutoCalEnChx output_bits.12 0
Sample_Set set_average_count
Chx_mV_V_Out ch2_autocal_mv_v 2.0172822
set_ch_adtrim ch2adtrimset 108000
resetaverage output_bits.2
chx_tare output_bits.6
chx_tare output_bits.6 0 chx_cleartare output_bits.5 0
0 chx_cleartare output_bits.5 0 sample_rpi sample_rate
0 chx_cleartare output_bits.5 0 sample_rpi sample_rate 1 auto_tare_band auto_tare 0
0 chx_cleartare output_bits.5 sample_rpi sample_rate 1 auto_tare_band auto_tare 0 getvaluein nicktest:I.Data[1] 0
0 chx_cleartare output_bits.5 0 sample_rpi sample_rate 1 auto_tare_band auto_tare 0 getvaluein nicktest:I.Data[1]

4 SCM4_WM_DINT SCM4WHS SetCalifytics SCM4WHS SetCalifytics <th></th> <th>SCM4_WM_D</th> <th>INT</th>		SCM4_WM_D	INT
4 SetCaMAdede output_bits SetCaMAdede output_bits SetScaleChx chd_scale set SetMadede output_bits SetScaleChx output	3	SCM4_WM_DINT	SCM4WM3
4 4 4 4 4 4 4 5 5 5 5 5 5 5 5 5 5 5 5 5		SetCalMode	output_bits.0
4		SetRunMode of	output_bits.1
4		SetScaleChx ch	3_scale_set 100000
4 SefRefChx ch3_known_weight AutoCatEnChx ch4_known_weight Chx_mV_V_Cut ch3_autocal_mv_v 2010000 restaverage output_bits_0 chx_tare output_bits_0 chx_tare output_bits_0 auto_tare_band auto_tare getvaluein nicktest1Data[2] msf msg set_ch_astrin ch3-weight 4 SCM4_VM_DINT SCM4_WM_DINT		Set_mV_V_Chx se	t_mv_v_ch3
4 AutoElEnChk output_bits.13 Sample_Stat set_average_count Chx_mV_V_Out ch3_autocal mv, v 200000 resetaverage output_bits.3 chx_clarate output_bits.3 chx_clarate output_bits.3 auto_tare_band auto_tare getvalueout ch3weigh 4 SCM4_WM_DINT		SetRefChx ch3_km	nown_weight
4		AutoCalEnChx ou	utput_bits.13
4		Sample_Set set_ave	erage_count
4		Chx_mV_V_Out ch3_au	utocal_mv_v
4		set_ch_adtrim c	ch3adtrimset
4 the set of the set		resetaverage	output_bits.2
4		chx_tare c	output_bits.8
4		chx_cleartare c	output_bits.7
4 4 4 4 4 4 4 4 4 4		sample_rpi	sample_rate
4 4 SCM4_WM_DINT		auto_tare_band	auto_tare
4 SCM4_WM_DINT_SCM4WM4 SCM4_WM_DINT_SCM4W44 SCM4_SCM4W44 SC		getvaluein nickt	est:I.Data[2]
4 SCM4_WM_DINT_SCM4WM4 4 SCM4_WM_DINT_SCM4W4 ScM4_WM_DINT_SCM4W4 ScM4_WM_DINT_SCM4W4 4 ScM4_WM_DINT_SCM4W4 4 ScM4_ScM4_SCM4W4 4 ScM4_ScM4_ScM4_SCM4W4 4 ScM4_ScM4_ScM4_SCM4W4 4 ScM4_ScM4_SCM4W4 4 ScM4_Sc		msf	msf3
4 SCM4_WM_DINT SCM4WM4 SCM4_WM_DINT SCM4WM4 SCM4_WM_DINT SCM4WM4 SetCalMode output_bits.1 SetScaleChx ch4_scale_set 100000 Set_mV_V_Chx set_mv_v.ch4 2.0186808 set_ch4_known_weight 100000 AutoCalEnChx output_bits.14 Chx_mV_V_Out ch4_autocal mv_v 2.0186808 set_ch_adrim ch4adtimset 106000 resetaverage output_bits.2 chx_tare output_bits.10 chx_tare output_bits.9 sample_rpi sample_rate auto_tare_band auto_tare getvalueout ch4weigh 		getvalueout	ch3weigh
4 SCM4_WM_DINT_SCM4WM4_int SetGalMode_output_bits.0 SetRunMode_output_bits.1 SetScaleChx_ch4_scale_set_n00000 Set_mV_V_Chx_set_mv_v_ch4 2.0 SetRefChx_ch4_known_weight 100000 AutoCalEnChx_output_bits.14 0 Sample_Set_set_average_count 1 Chx_mV_V_Out_ch4_autocal_mv_v 2.0168608 set_ch_adtrim_ch4adtrimset 108000 resetaverage_output_bits.2 0 chx_tare_output_bits.10 chx_cleartare_output_bits.10 chx_tare_band_auto_tare 0 getvaluein_nicktest.1.Data[3] msfs msfs			
SetCalMode output_bits.0 O SetRunMode output_bits.1 SetScaleChx ch4_scale_set 100000 Set_mV_V_Chx set_mv_v_ch4 2.0 SetRefChx ch4_known_weight 100000 AutoCalEnChx output_bits.14 0 Sample_Set set_average_count 1 Chx_mV_V_Out ch4_autocal_mv_v 2.0186808 set_ch_adtrim ch4adtrimset 108000 resetaverage output_bits.2 0 chx_tare output_bits.10 0 chx_tare output_bits.10 0 chx_tare output_bits.2 0 chx_tare output_bits.10 0 getvaluein nicktest.10ats.3] msf msf msf 0 getvalueout ch4weigh	4		
SetRunMode output_bits.1 SetScaleChx ch4_scale_set 100000 Set_mV_V_Chx SetRefChx ch4_scole_weight 100000 AutoCalEnChx output_bits.14 0 Sample_Set set_average_count 1 Chx_mV_V_Out ch4_autocal_mv_v 2.0186808 set_ch_adtrim ch4adtrimset 108000 resetaverage output_bits.2 0 chx_tare output_bits.9 0 chx_cleartare output_bits.9 0 sample_rpi sample_rate auto_tare auto_tare_band auto_tare 0 getvaluein nicktest.1.Data[3] 0 msf msf -8		SCM4 WW DINT	CCMANA/MA
SetScaleChx ch4_scale_set 100000 Set_mV_V_Chx 20 20 SetRefChx ch4_known_weight 100000 AutoCalEnChx output_bits.14 0 Sample_Set set_average_count 1 Chx_mV_V_Out ch4_autocal_mv_v 2.0186808 set_ch_adtrim ch4adtrimset 108000 resetaverage output_bits.10 0 chx_tare output_bits.10 0 chx_cleartare output_bits.9 0 sample_rpi sample_rate 1 auto_tare_band auto_tare 0 getvaluein nicktestl.Data[3] msf msf4 .8		SetCalMode o	output_bits.0
Set_mV_V_Chx set_mv_v_ch4 2.0 SetRefChx ch4_known_weight 100000 AutoCalEnChx output_bits.14 0 sample_Set set_average_count 1 Chx_mV_V_Out ch4_adtrimset 2.0168088 set_ch_adtrim ch4adtrimset 108000 resetaverage output_bits.2 108000 resetaverage output_bits.10 chx_tare output_bits.10 output_bits.10 chx_cleartare output_bits.10 output_bits.10 getvaluein nicktest:I.Data[3] output_bits.4 getvaluein nicktest:I.Data[3] output_bits.3 sample_rpi sample_rate 0 auto_tare_band auto_tare 0 getvaluein nicktest:I.Data[3] output_bits.3		SetCalMode	output_bits.0 0 output_bits.1
SetRefChx ch4_known_weight 100000 AutoCalEnChx output_bits.14 O Sample_Set set_average_count Chx_mV_V_Out ch4_autocal_mv_v 2.0186808 set_ch_adtrim set_ch_adtrim ch4adtrimset 0 008000 resetaverage output_bits.2 0 chx_tare output_bits.9 0 chx_cleartare output_bits.9 0 sample_rpi sample_rate 1 auto_tare_band auto_tare 0 msf msf4 0 msf -8		SetCalMode of SetRunMode of	output_bits.0 0 output_bits.1 4_scale_set
AutoCalEnChx output_bits.14 O Sample_Set set_average_count Chx_mV_V_Out ch4_autocal_mv_v 2.0186808 set_ch_adtrim ch4adtrimset 100000 resetaverage output_bits.2 0 chx_tare output_bits.10 0 chx_cleartare output_bits.9 0 sample_rpi sample_rate 1 auto_tare_band auto_tare 0 getvaluein nicktest:I.Data[3] 0 msf msf4 0 getvalueout ch4weigh -8		SetCalMode of SetRunMode of SetScaleChx ch	output_bits.0 0 putput_bits.1 4_scale_set 100000 t_mv_v_ch4
Sample_Set set_average_count 1 Chx_mV_V_Out ch4_autocal_mv_v 2.0186808 set_ch_adtrim ch4adtrimset 108000 resetaverage output_bits.0 0 chx_tare output_bits.10 0 chx_cleartare output_bits.9 0 sample_rpi sample_rate 1 auto_tare_band auto_tare 0 getvaluein nicktest:I.Data[3] 0 msf msf4 0 getvalueout ch4weigh -8		SetCalMode of SetRunMode of SetScaleChx ch Set_mV_V_Chx set	output_bits.0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
Chx_mV_V_Out_ch4_autocal_mv_v 2.0186808 set_ch_adtrim_ch4adtrimset 108000 resetaverage0 chx_tare0 chx_cleartare0 chx_cleartare0 sample_rpi1 auto_tare_band0 getvalueinnicktest:I.Data[3] 0 msf0 getvalueout6 0 getvalueout6		SetCalMode of SetRunMode of SetScaleChx ch Set_mV_V_Chx set SetRefChx ch4_km	output_bits.0 0 0 1 4_scale_set 100000 t_mv_v_ch4 2.0 100000 utput_bits.14
set_ch_adtrim ch4adtrimset 108000 resetaverage output_bits.2 0 chx_tare output_bits.10 0 chx_cleartare output_bits.9 0 sample_rpi sample_rate 1 auto_tare_band auto_tare 0 getvaluein nicktest:1.Data[3] 0 msf msf4 0 getvalueout ch4weigh -8		SetCalMode of SetRunMode of SetScaleChx ch Set_mV_V_Chx set SetRefChx ch4_km AutoCalEnChx ou	output_bits.0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
resetaverage output_bits.2 0 chx_tare output_bits.10 0 chx_cleartare output_bits.9 0 sample_rpi sample_rate 1 auto_tare_band auto_tare 0 getvaluein nicktest:1.Data[3] 0 msf msf4 0 getvalueout ch4weigh -8		SetCalMode of SetRunMode of SetScaleChx ch Set_mV_V_Chx set SetRefChx ch4_km AutoCalEnChx ou Sample_Set set_ave	output_bits.0 0 0 4_scale_set 100000 t_mv_v_ch4 2.0 0 00wn_weight 100000 ttput_bits.14 0 erage_count 1 utocal_mv_v
chx_tare output_bits.10 0 chx_cleartare output_bits.9 0 sample_rpi sample_rate 1 auto_tare_band auto_tare 0 getvaluein nicktest:I.Data[3] 0 msf msf4 0 getvalueout ch4weigh -8		SetCalMode of SetRunMode of SetScaleChx ch Set_mV_V_Chx set SetRefChx ch4_km AutoCalEnChx ou Sample_Set set_ave Chx_mV_V_Out ch4_au	output_bits.0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
chx_cleartare output_bits.9 0 sample_rpi sample_rate 1 auto_tare_band auto_tare 0 getvaluein nicktest:I.Data[3] 0 msf msf4 0 getvalueout ch4weigh -8		SetCalMode od SetRunMode od SetScaleChx ch Set_mV_V_Chx set SetRefChx ch4_kn AutoCalEnChx ou Sample_Set set_ave Chx_mV_V_Out ch4_ave set_ch_adtrim od	output_bits.0 0 0 0 0 0 0 1 4_scale_set 100000 t_mv_v_ch4 2.0 0 0 0 0 0 0 0 0 0 0 0 0 0
sample_rpi sample_rate 1 auto_tare_band auto_tare 0 getvaluein nicktest:I.Data[3] 0 msf msf4 0 getvalueout ch4weigh -8		SetCalMode of SetRunMode of SetScaleChx ch Set_mV_V_Chx set SetRefChx ch4_km AutoCalEnChx ou Sample_Set set_ave Chx_mV_V_Out ch4_au set_ch_adtrim of resetaverage of	output_bits.0 0 output_bits.1 1 4_scale_set 100000 t_mv_v_ch4 2.0 own_weight 100000 utput_bits.14 0 erage_count 1 vtocal_mv_v 2.0186808 ch4adtrimset 108000 output_bits.2 0 utput_bits.10
auto_tare_band auto_tare 0 getvaluein nicktest:I.Data[3] 0 msf msf4 0 getvalueout ch4weigh -8		SetCalMode of SetRunMode of SetScaleChx ch Set_mV_V_Chx set SetRefChx ch4_kn AutoCalEnChx ou Sample_Set set_ave Chx_mV_V_Out ch4_au set_ch_adtrim of resetaverage of chx_tare ou	output_bits.0 0 0 0 0 0 0 0 0 0 0 0 0 0
getvaluein nicktest:I.Data[3] 0 msf msf4 0 getvalueout ch4weigh -8		SetCalMode of SetRunMode of SetScaleChx ch Set_mV_V_Chx set SetRefChx ch4_km AutoCalEnChx ou Sample_Set set_ave Chx_mV_V_Out ch4_au set_ch_adtrim of resetaverage of chx_tare ou chx_tare ou	output_bits.0 0 output_bits.1 1 4_scale_set 100000 t_mv_v_ch4 2.0 own_weight 100000 utput_bits.14 0 erage_count 1 utocal_mv_v 2.0186808 ch4adtrimset 108000 output_bits.2 0 utput_bits.10 0 output_bits.9 0 sample_rate
msf msf4 0 getvalueout ch4weigh ~8		SetCalMode of SetRunMode of SetScaleChx ch Set_mV_V_Chx set SetRefChx ch4_km AutoCalEnChx ou Sample_Set set_ave Chx_mV_V_Out ch4_au set_ch_adtrim of resetaverage of chx_tare ou chx_tare ou chx_cleartare of sample_rpi set	output_bits.0 0 0 0 0 0 0 0 0 0 0 0 0 0
getvalueout ch4weigh -8		SetCalModeSetCalModeSetRunModeCSetScaleChxChSet_mV_V_ChxsetSetRefChxCh4_knAutoCalEnChxouSample_Setset_aveChx_mV_V_Outch4_auset_ch_adtrimcresetaveragecchx_tareouchx_cleartarecsample_rpisauto_tare_bands	output_bits.0 0 output_bits.1 1 4_scale_set 100000 t_mv_v_ch4 2.0 own_weight 100000 utput_bits.14 0 erage_count 1 utocal_mv_v 2.0186808 ch4adtrimset 108000 output_bits.2 0 utput_bits.10 0 output_bits.9 0 sample_rate 1 auto_tare 0 est:1.Data[3]
		SetCalMode d SetRunMode d SetScaleChx ch Set_mV_V_Chx set SetRefChx ch4_kn AutoCalEnChx ou Sample_Set set_ave Chx_mV_V_Out ch4_au set_ch_adtrim d chx_tare ou chx_cleartare d sample_rpi set auto_tare_band getvaluein getvaluein nickt	output_bits.0 0 0 0 0 0 0 0 0 0 0 0 0 0
(End)		SetCalMode of SetRunMode of SetScaleChx ch Set_mV_V_Chx set SetRefChx ch4_km AutoCalEnChx ou Sample_Set set_ave Chx_mV_V_Out ch4_au set_ch_adtrim of resetaverage of chx_tare ou chx_tare ou chx_cleartare of sample_rpi st auto_tare_band getvaluein nickt msf	output_bits.0 0 0 0 0 0 0 0 0 0 0 0 0 0
(End)		SetCalMode of SetRunMode of SetScaleChx ch Set_mV_V_Chx set SetRefChx ch4_km AutoCalEnChx ou Sample_Set set_ave Chx_mV_V_Out ch4_au set_ch_adtrim of resetaverage of chx_tare ou chx_tare ou chx_cleartare of sample_rpi st auto_tare_band getvaluein nickt msf	output_bits.0 0 0 0 0 0 0 0 0 0 0 0 0 0
		SetCalMode of SetRunMode of SetScaleChx ch Set_mV_V_Chx set SetRefChx ch4_km AutoCalEnChx ou Sample_Set set_ave Chx_mV_V_Out ch4_au set_ch_adtrim of resetaverage of chx_tare ou chx_tare ou chx_cleartare of sample_rpi st auto_tare_band getvaluein nickt msf	output_bits.0 0 0 0 0 0 0 0 0 0 0 0 0 0

MODULE INITIAL SETUP PROCEDURE

A complete listing of a sample ladder logic program is included at the back of this manual. Examples shown here are for reference.

All values are 0 (default) on initial start-up. This means that all alarms are disabled. You must make the following adjustments for proper operation:

- Balance sensor input(s)
- Set Calibration numbers

Follow Steps 1 and 2 for each channel.

Step 1. Balance Sensor Input

- 1. Set to CAL mode.
- 2. Set Clear Tare bit momentarily.
- 3. Check Raw A/D value. (131,000)
- 4. Set Zero Tare bit momentarily.

Step 2. Set Calibration Numbers

- 1. Set Scale to capacity of load cell.
- 2. Set mV/V to load cell specification.

Example:

100 ton load cell, 2.025 $m\mathrm{V/V}$

For scale set, enter 100

For mV/V set, enter 2.025

3. Set to RUN mode.

OUTPUT BITS

CAL-MODE

All scale parameters are bypassed, raw a/d values, use full diagnostics

RUN MODE

Weigh values are scaled based upon scale set parameters.

RESET AVERAGE

Toggle bit, needed for first run, initializes values in memory for average.

CLEAR TARE

Toggle bit to remove offset calc. used for diagnostics.

TARE

Toggle bit to set weigh value to zero

AUTOCAL ENABLE

Set bit to run auto-cal

CONTROLLER TAGS

AUTO TARE

Used to set weigh values to zero when weigh station is at rest or idle condition (no-weight) Set to value in counts, if weigh is with in band set by counts, weigh will be set to zero Set to zero to bypass this function Normal setting <2> counts

AUTOCAL MV_V

Value that is updated when auto cal is used

KNOWN WEIGHT

Enter value for known weight when using auto cal

SCALE SET Set to capacity of load cell

ADTRIMSET Set to value on cal sticker on the side of the module

CH1-CH4 WEIGH

Current weigh value

MSF1-MSF2

Meter stabilization filter Set value in counts, keeps weigh value from changing if weight is not moving Normal setting 1 or 2 set to zero to bypass

SAMPLE RATE

Set to module RPI setting Normal setting <2>

SET AVERAGE COUNT

Number of samples for average Normal setting <100> Set to 0 or 1 for no average

SET MV-V CH1-CH4

Set to values on load cell or on cal. cell certificate

NORMAL CALIBRATION PROCEDURE

- 1. Enter mV/V from load (set mV/V)
- 2. Enter capacity from load (set scale)
- 3. Resolution determined by scale set

EXAMPLE:

- 1. Enter 1,000 = 1 gram / count
- 2. Enter 10,000 = .1 gram / count

AUTO CALIBRATION PROCEDURE

Before running, weight must be stable and weight returns to zero when weight is removed

- 1. Enter known weight at chx_known_wei
- 2. Enter load cell capacity at scale
- 3. Enter 2.000 at set mv_v_chx

EXAMPLE:

1kg load cell, 500 gram known weight, for 1-gram resolution

- 1. Enter 1,000 at scale set
- 2. Enter 500 at known_weight
- 3. Enter 2.000 at set mv_v_chx

Toggle tare bit (no weight at start) Apply known weight Set autocal_enable_chx Auto cal mv_v_v will change to correct value and weigh value will update to known weight Clear auto cal enable Enter <u>auto cal mV/V</u> at set mv_v_chx

