7929 SW Burns Way
Suite B
Wilsonville, OR

Phone: 503 344-5085
Fax: 503 682-9014
sales@rinehartmotion.com

RMS CAN Protocol

Revision 3.9

7929 SW Burns Way
Suite B
Wilsonville, OR

Phone: 503 344-5085
Fax: 503 682-9014
sales@rinehartmotion.com

Table of Contents

7929 SW Burns Way
Suite B
Wilsonville, OR

Phone: 503 344-5085
Fax: 503 682-9014
sales@rinehartmotion.com

1. Introduction

This document defines the CAN protocol used by the RMS PM controllers.

The PM controllers have two CAN interfaces (CAN A and CAN B). The controller is configured to communicate only over CAN A, currently CAN B is reserved for future use.

The CAN interface has multiple purposes:

- Direct control of the motor
- Adjust EEPROM parameters
- Diagnostics and monitoring

1.1 CAN EEPROM Parameters Overview

RMS CAN communication is set up with a default rate of 250 K baud. The available baud rates include 125K, 250K, 500K, and 1M. The terminator resistor is available and active by default. RMS CAN is configurable through parameters that are available through the RMS GUI[1] application as well as CAN. GUI parameters have the same name as mentioned in this document with the exception that they end with the keyword “EEPROM”. Following parameters are used to configure RMS CAN:

- **Inverter Command Mode:**
  This parameter gives the option to operate PM controller unit in either VSM mode or CAN mode. In VSM mode, RMS GUI is the main interface to monitor and modify parameters. The inverter takes messages from the Vehicle State Machine which is operated from the various inputs and outputs of the inverter such brake, accelerator pedal, etc. In VSM mode, broadcast messages are still sent out over the CAN lines. In CAN mode, both GUI and CAN interfaces are active and can be used to monitor and modify parameters. However, any inputs from the vehicle state machine will be ignored.

  0 = CAN Mode
  1 = VSM Mode (Default)

- **CAN ID Offset:**

[1] Please refer to the document “Programming EEPROM Parameters using GUI”.

This parameter allows the user to choose their own set of contiguous CAN message identifiers starting with the value in CAN ID Offset. This offset covers a range of 0 – 0x7C0. The default offset is 0x0A0. The default range is 0x0A0 – 0x0CF. This feature is especially useful when there are more than one PM controllers on the same CAN network. While setting base address for a controller, it must be made sure that the address range for controllers does not contain overlapping addresses.

- **CAN Extended Message Identifier**
  This parameter allows switching between CAN standard and extended message identifiers.

  0 = Standard CAN Messages (11-bit identifiers)
  1 = Extended CAN Messages (29-bit identifiers)

- **CAN J1939 Option Active**
  This parameter allows switching to J1939 format in extended mode. This parameter works in conjunction with 'CAN Extended Message Identifier' parameter above.

  0 = Extended CAN Messages (if CAN Extended Message Identifier is set to 1, otherwise it will be standard messaging).
  1 = Extended CAN Messages in SAE J1939 Format

  RMS provides a limited functionality for J1939 protocol with the following fixed parameters:

Priority	= 3 (011b)
Data Page	= 0
PDU Format	= 0xFF
PDU Specific	= **CAN ID Offset**
Source Address	= 0x01

  For example, if CAN ID Offset is set to 0xA0, the J1939 CAN message will be broadcast as 0xCFFA001 and so on. The heartbeat command message should be sent out as 0xCFFA000, where the source address of the sending node is assumed to be 0x00.

  Please refer to the section, *1.3 CAN Format* to learn about how to send out a torque or a speed command using the heartbeat command message.

- **CAN Term. Resistor Present:**
  In order to use CAN communication, the CAN bus needs to be terminated with a 120 Ohm resistor. RMS PM controller units are equipped with this resistor which is activated through this parameter.

  0 = Term. Resistor not active
  1 = Term. Resistor active (Default)
  If CAN Terminator Resistor is deactivated, it may be necessary to use the GUI interface only[2] since CAN communication may fail without a terminator resistor.

- **CAN Command Message Active:**
  RMS CAN requires a "heartbeat" command message to be sent at least once every half a second. This command message controls the inverter, motor direction, and torque or speed. In the absence of a regular broadcast of this message, PM controller will assume there is a problem and will flag a fault unless the fault has been deactivated by setting this parameter to 0.

  0 = The command message 0xC0 is **not** sent at least once in half a second.
  1 = The command message 0xC0 is sent once every half a second (Default)

- **CAN Bit Rate:**
  250Kbps is the default bit rate. Bus speed can be changed using CAN parameter command message. However, changing this parameter requires a power reset on PM controller unit since bus speed is setup only at the initialization of CAN modules in the microcontroller. Also, this input is restricted to valid baud rates. The 4 options for valid baud rate are:

  125 = 125Kbps
  250 = 250 Kbps (Default)
  500 = 500 Kbps
  1000 = 1Mbps

[2] Please refer to the document "Programming EEPROM Parameters using GUI".

7929 SW Burns Way
Suite B
Wilsonville, OR

Phone: 503 344-5085
Fax: 503 682-9014
sales@rinehartmotion.com

- **CAN Active Messages Word:**
  This parameter is used to enable/disable CAN Broadcast Messages[3]. This parameter is represented as two parameters, CAN Active Messages (Low Word) and CAN Active Messages (High Word) in RMS GUI. Each bit represents a CAN Message broadcast status as follows:

  0 = CAN Messages broadcast disabled
  1 = CAN Message broadcast enabled (Default)

  Please refer to the table of CAN Broadcast Messages in section 2.1 for details on how to enable/disable each message.

- **CAN Diagnostic Data Transmit Active:**
  This parameter is used to enable/disable the broadcast of the diagnostic data.
  0 = CAN Diagnostic Data broadcast disabled
  1 = CAN Diagnostic Data broadcast enabled (Default)
  Please refer to the document, CAN Diagnostic Data, for more details on this feature.

- **CAN Inverter Enable Switch Active:**
  This parameter is used in CAN mode only.
  1 = DIN1 digital input is taken into consideration and the inverter will only be enabled if both DIN1 and inverter command are active. If either one is inactive, the inverter will be disabled.
  0 = DIN1 will have no effect on enabling or disabling the inverter (Default)

- **CAN Timeout:**
  This parameter is used in CAN mode only. This parameter is set as a multiple of 3 msec. For example, the default value is set to 333 which is equivalent to the actual timeout value of 1 second (333 x 3msec). This parameter delays setting the CAN Timeout fault for the amount of time it represents.

[3] Please refer to the section 2.1 - Broadcast Messages.

1.2 CAN Diagnostic Parameters Overview

RMS CAN communication also provides a couple of diagnostic parameters that can be monitored through the GUI. These parameters are shadow parameters of the actual CAN Error and Status register.

- **CAN Status Register:**
  The Status register is displayed in real time in the GUI. This is a shadow byte that reflects the actual CAN register. This shadow register is updated every 3 msec.

CAN Status Word	Description
0x0001	**Transmit Mode (TM):** The CAN module is in transmit mode. This bit reflects what the CAN module is actually doing regardless of mailbox configuration. 1: The CAN module is transmitting a message 0: The CAN module is not transmitting a message
0x0002	**Receive Mode (RM):** The CAN module is in receive mode. This bit reflects what the CAN module is actually doing regardless of mailbox configuration. 1: The CAN module is receiving a message 0: The CAN module is not receiving a message
0x0004	**Power Down Acknowledge (PDA):** 1: The CAN module has entered the power-down mode 0: Normal operation
0x0008	**Change Configuration Enable (CCE):** This bit displays the configuration access right. This bit is set after a latency of one clock cycle. 1: The CPU has write access to the configuration registers 0: The CPU is denied write access to the configuration registers
0x0010	**Suspend Mode Acknowledge (SMA):** This bit is set after a latency of one clock cycle—up to the length of one frame—after the suspend mode was activated. The suspend mode is activated with the debugger tool when the circuit is not in run mode. 1: The module has entered suspend mode 0: The module is not in suspend mode
0x0020	**Warning Status (EW):** 1: One of the two error counters (CANREC or CANTEC) has reached the warning level of 96. 0: Values of both error counters (CANREC and CANTEC) are less than 96.
0x0040	**Error Passive (EP):** 1: CAN module is in error-passive mode. CANTEC has reached 128 0: The CAN module is in error-active mode

0x0080	**Buss Off (BO):** The CAN module is in bus-off state. 1: There is an abnormal rate of errors on the CAN bus. This condition occurs when the transmit error counter (CANTEC) has reached the limit of 256. During Bus Off, no messages can be received or transmitted. The bus-off state can be exit by clearing the CCR bit in CANMC register or if the Auto Bus On (ABO) (CANMC.7) bit is set after 128 * 11 receive bits have been received. After leaving Bus Off, the error counters are cleared. 0: Normal operation

- **CAN Fault Register:**
  The Faults register is also a shadow register. However, once a CAN fault occurs, shadow register will continue to display it through the GUI. The shadow faults can be cleared by setting Clear Fault Command to 0[4].

CAN Faults Word	Description
0x01	**Acknowledge Error (ACKE):** 1: The CAN module received no acknowledge 0: All messages have been correctly acknowledged
0x02	**Stuff Error (SE):** 1: A stuff bit error occurred. 0: No stuff bit error occurred.
0x04	**Cyclic Redundancy Check Error (CRCE):** 1: The CAN module received a wrong CRC. 0: The CAN module never received a wrong CRC.
0x08	**Stuck at Dominant 1 Error (SA1):** The SA1 bit is always at 1 after a hardware reset, a software reset, or a Bus-Off condition. This bit is cleared when a recessive bit is detected on the bus. 1: The CAN module never detected a recessive bit. 0: The CAN module detected a recessive bit.
0x10	**Bit Error (BE):** 1: The received bit does not match the transmitted bit outside of the arbitration field or during transmission of the arbitration field, a dominant bit was sent but a recessive bit was received. 0: No bit error detected.
0x20	**Form Error (FE):** 1: A form error occurred on the bus. This means that one or more of the fixed-form bit fields had the wrong level on the bus. 0: No form error detected; the CAN module was able to send and receive correctly.

[4] Please refer to "RMS PM Software User Manual" section 11.6.2 – Clear Faults Command.

7929 SW Burns Way
Suite B
Wilsonville, OR

Phone: 503 344-5085
Fax: 503 682-9014
sales@rinehartmotion.com

1.3 CAN Format

The CAN protocol conforms to CAN 2.0A (11 bit identifiers) as well as CAN 2.0B (29 bit identifiers)[5]. CAN Messages are transmitted at a default rate of 250kbps[6]. All messages have a data length code (DLC) of 8 bytes and follow little-endian format which implies that the least significant byte is stored at the lowest address. For example, if the command message is setup to turn the inverter on in CAN Speed mode with a speed command of 500 RPM the data bytes should look like this:

Data Byte 0	Data Byte 1	Data Byte 2	Data Byte 3	Data Byte 4	Data Byte 5	Data Byte 6	Data Byte 7
44	1	244	1	0	1	0	0

- Torque Command: Sent as a value in N.m. times 10.
  For example, 30 Nm should be entered as 300 = (1 x 256) + 44
  - Data Byte 0 = 44 (Low byte)
  - Data Byte 1 = 1 (High byte)

- Speed Command: Sent as a value in RPM.
  For example, 500 RPM is entered as 500 = (1 x 256) + 244
  - Data Byte 2 = 244 (Low byte)
  - Data Byte 3 = 1 (High byte)

- Direction Command
  - Data Byte 4 = 0 (Clockwise = Reverse)
  - Data Byte 4 = 1 (Anticlockwise = Forward)

- Inverter Run Command
  - Data Byte 5 = 0 (Disable Inverter)
  - Data Byte 5 = 1 (Enable Inverter)

Each data frame is 89 bits long thus at 250kbps the bus can handle a maximum of 2808 messages per second.

[5] Please refer to “CAN Extended Message Identifier” in section 1 - Introduction.
[6] Please refer to “CAN Bit Rate” in section 1 - Introduction.

1.4 Data Formats

Each message contains one or more items. Each item is formatted and scaled per the definitions below:

Format	Description	Range
Temperature	Signed integer, actual temperature (in °C) times 10	-3276.8 to +3276.7 °C
Low Voltage	Signed integer, actual voltage (in Volts) times 100	-327.68 to +327.67 volts
Torque	Signed integer, actual torque (in N.m) times 10	-3276.8 to +3276.7 N-m
High Voltage	Signed integer, actual voltage (in Volts) times 10	-3276.8 to +3276.7 volts
Current	Signed Integer, actual current (in Amps) times 10	-3276.8 to +3276.7 amps
Angle	Signed integer, actual angle (in degrees) times 10	0.0 to ±359.9 degrees
Angular velocity (Speed)	Signed integer, actual velocity (in RPM)	-32768 to +32767 rpm
Boolean	Unsigned byte, 1 = true/on, 0 = false/off	0 or 1
Frequency	Signed integer, actual frequency (in Hz) times 10	-3276.8 to +3276.7 Hz
Power	Signed integer, actual power (in kW) times 10	-3276.8 to +3276.7 kW
Time	Unsigned long integer or Unsigned integer. These are scaled values in counts that can be calculated by using their respective Scale Factors. For each Scale Factor, see the description column for that parameter.	NA
Flux	Signed integer, actual flux (in Webers) times 1000	-32.768 to 32.767 Webers
Proportional Gain	Unsigned integer, actual gain (unit-less) times 100 OR actual gain (unit-less) times 10000	0 – 655.35 OR 0 – 6.5535
Integral Gain	Unsigned integer, Actual gain (unit-less) times 10000	0 – 6.5535
Derivative Gain	Unsigned integer, actual gain (unit-less) times 100	0 – 655.35
Low-pass Filter Gain	Unsigned integer, Actual gain (unit-less) times 10000	0 – 6.5535
Per-unit Value	These are scaled values that can be calculated by using their respective Scale Factors. For each Scale Factor, see the description column for that parameter.	NA
ADC Count	The value for ADC counts as read directly by the registers of a microcontroller.	0 - 4095

1.5 CAN Database File

Each firmware is provided with a specific CAN database file. The database file can be matched with the firmware file through the date code in *yyyymmdd* format. The database file is located under “Firmware” folder. It also located under “Tools\CAN Tools” folder of the RMS online repository with box.net.

A CAN database file stores information for a given CAN network. For example, it includes information about CAN nodes, messages, and data bytes for each message. A CAN database file has a .dbc extension and can be used with several CAN data loggers such CANTrace, CANalyzer, CANoe, etc to log CAN data.

This file can also be edited by the user with his/her choice of a CAN database editor. RMS has used Kvaser’s Database Editor (DbEdit) to create and edit the CAN database file. Kvaser’s DbEdit tool can be downloaded using the following link:
http://www.kvaser.com/index.php?option=com_php&Itemid=288&lang=&swprod=1cf9896f527dc55dc8a219a048c6c149&ean=7330130003514

The user guide for Kvaser’s database editor can be directly downloaded using the following link:
www.kvaser.cn/support/pdf/kvaser_database_editor_users_manual.pdf

7929 SW Burns Way
Suite B
Wilsonville, OR

Phone: 503 344-5085
Fax: 503 682-9014
sales@rinehartmotion.com

2. CAN Messages

2.1 Broadcast Messages

Broadcast messages are sent by the PM controller continuously irrespective of VSM or CAN command mode. The table below shows the messages that are broadcast and the frequency at which they are sent. The addresses below are default addresses and can be changed using the Read/Write Parameter Command message (see section Parameter Messages). This would be specifically useful in case of more than one PM controller units on a CAN network. While setting base address for a controller, it must be made sure that the address range for controllers does not contain overlapping addresses.

A parameter 'CAN Active Messages Word' with parameter address 148 is defined to enable/disable CAN Broadcast Messages. This is a 32-bit parameter that is further divided into two 16-bit parameters, CAN Active Messages Low Word and CAN Active Messages High Word. Each bit in this parameter represents a CAN Message broadcast status as follows:

0 = CAN Messages broadcast disabled
1 = CAN Message broadcast enabled

The controller broadcasts the following messages:

Address	Frequency	Content	CAN Active Messages (High Word)	CAN Active Messages (Low Word)
0x0A0	10 Hz	Temperatures #1	0x0000	0x0001
0x0A1	10 Hz	Temperatures #2	0x0000	0x0002
0x0A2	10 Hz	Temperatures #3	0x0000	0x0004
0x0A3	100 Hz	Analog Inputs Voltages	0x0000	0x0008
0x0A4	100 Hz	Digital Input Status	0x0000	0x0010
0x0A5	100 Hz	Motor Position Information	0x0000	0x0020
0x0A6	100 Hz	Current Information	0x0000	0x0040
0x0A7	100 Hz	Voltage Information	0x0000	0x0080
0x0A8	100 Hz	Flux Information	0x0000	0x0100
0x0A9	10 Hz	Internal Voltages	0x0000	0x0200
0x0AA	10 Hz	Internal States	0x0000	0x0400
0x0AB	10 Hz	Fault Codes	0x0000	0x0800
0x0AC	100 Hz	Torque & Timer Information	0x0000	0x1000
0x0AD	100 Hz	Modulation Index & Flux Weakening Output Information	0x0000	0x2000
0x0AE	10 Hz	Firmware Information	0x0000	0x4000
0x0AF	100 Hz	Diagnostic Data	0x0000	0x8000

All of the above messages are in standard 11-bit format. The addresses in extended 29-bit and J1939 format are listed below:

Standard 11-bit Format	Extended 29-bit Format	J1939 Format
0x0A0	0x0A0 **X**	0x0CFFA001
0x0A1	0x0A1 **X**	0x0CFFA101
0x0A2	0x0A2 **X**	0x0CFFA201
0x0A3	0x0A3 **X**	0x0CFFA301
0x0A4	0x0A4 **X**	0x0CFFA401
0x0A5	0x0A5 **X**	0x0CFFA501
0x0A6	0x0A6 **X**	0x0CFFA601
0x0A7	0x0A7 **X**	0x0CFFA701
0x0A8	0x0A8 **X**	0x0CFFA801
0x0A9	0x0A9 **X**	0x0CFFA901
0x0AA	0x0AA **X**	0x0CFFAA01
0x0AB	0x0AB **X**	0x0CFFAB01
0x0AC	0x0AC **X**	0x0CFFAC01
0x0AD	0x0AD **X**	0x0CFFAD01
0x0AE	0x0AE **X**	0x0CFFAE01
0x0AF	0x0AF **X**	0x0CFFAF01

Extended format is often denoted by the letter ‘X’. However, based on the CAN logger, this format may be represented differently. Please refer to the manual for the CAN logger that is used.

J1939 messages have the following fixed configuration:

- Priority = 3
- PDU Format = 0xFF
- PDU Specific = CAN ID Offset EEPROM
- Source Address = 1

Enabling/Disabling Broadcast of CAN Messages

As an example, in order to disable Temperature #1, #2 and #3 messages in the above table, the parameter command message should be configured as follows:

Data Byte 7 (Low Byte)	Data Byte 6 (Low Byte)	Data Byte 5 (High Byte)	Data Byte 4 (Low Byte)	Data Byte 3	Data Byte 2	Data Byte 1	Data Byte 0
CAN Active Messages High Word		CAN Active Messages Low Word		Reserved	R/W Command	Parameter Address	
255 (0xFF)	255 (0xFF)	255 (0xFF)	248 (0xF8)	0	1	0	148

Data Byte 4 controls the following messages:

Bit 0: Temperature #1
Bit 1: Temperature #2
Bit 2: Temperature #3
Bit 3: Analog Input Voltages
Bit 4: Digital Input Status
Bit 5: Motor Position Information
Bit 6: Current Information
Bit 7: Voltage Information

In little-endian format, Byte 4 can be looked at as: Bit 7 - Bit 6 - Bit 5 - ... - Bit 1 - Bit 0

To enable all messages above, Byte 4 should be set to 0xFF (all bits set to 1).
To disable temperature messages, Byte 4 should be set to 0xF8 (Bit 0, 1, and 2 are set to 0)
To disable Motor position information, Byte 4 should be set to 0xDF (Bit 5 set to 0)

Data Byte 5 controls the following messages:
Bit 0: Flux Information
Bit 1: Internal Voltages
Bit 2: Internal States
Bit 3: Fault Codes
Bit 4: Torque & Timer Information
Bit 5: Not used
Bit 6: Not used
Bit 7: Not used

Data Byte 6 has no messages assigned yet:
Bit 0: Not used
Bit 1: Not used
Bit 2: Not used
Bit 3: Not used
Bit 4: Not used
Bit 5: Not used
Bit 6: Not used
Bit 7: Not used

Data Byte 7 controls the following messages but user cannot disable any of those messages:

Bit 0: Not used
Bit 1: Not used
Bit 2: Not used
Bit 3: Not used
Bit 4: Not used
Bit 5: Parameter Response Message
Bit 6: Parameter Command Message
Bit 7: CAN Command Message (Heartbeat command continuously broadcast)

Note:
Higher nibble of Data Byte 7 contains CAN Command Messages that should never be disabled. Hence, it will always return a hex value of Fx, where x can be any hex number between 0 and F (15).

7929 SW Burns Way
Suite B
Wilsonville, OR

Phone: 503 344-5085
Fax: 503 682-9014
sales@rinehartmotion.com

Broadcast Message Definitions

0x0A0 – Temperatures #1

Byte #	Name	Format	Description
0,1	Module A Temperature	Temperature	Temperature of IGBT Module, Phase A
2,3	Module B Temperature	Temperature	Temperature of IGBT Module, Phase B
4,5	Module C Temperature	Temperature	Temperature of IGBT Module, Phase C
6,7	Gate Driver Board Temperature	Temperature	Temperature of Gate Driver Board

0x0A1 – Temperatures #2

Byte #	Name	Format	Description
0,1	Control Board Temperature	Temperature	Temperature of Control Board.
2,3	RTD #1 Temperature	Temperature	Temperature read from RTD input #1
4,5	RTD #2 Temperature	Temperature	Temperature read from RTD Input #2
6,7	RTD #3 Temperature	Temperature	Temperature read from RTD Input #3, Gen 2 only.

0x0A2 – Temperatures #3 & Torque Shudder

Byte #	Name	Format	Description
0,1	RTD #4 Temperature	Temperature	Temperature read from RTD Input #4, Gen 2 only
2,3	RTD #5 Temperature	Temperature	Temperature read from RTD Input #5, Gen 2 only
4,5	Motor Temperature	Temperature	Filtered temperature value from the motor temperature sensor.
6,7	Torque Shudder	Torque	A value of torque used in shudder compensation.

0x0A3 – Analog Input Voltages

Byte #	Name	Format	Description
0,1	Analog Input #1	Low Voltage	Voltage on Analog Input #1
2,3	Analog Input #2	Low Voltage	Voltage on Analog Input #2
4,5	Analog Input #3	Low Voltage	Voltage on Analog Input #3
6,7	Analog Input #4	Low Voltage	Voltage on Analog Input #4

0x0A4 – Digital Input Status

Byte #	Name	Format	Description
0	Digital Input #1	Boolean	Status of Digital Input #1, Forward switch
1	Digital Input #2	Boolean	Status of Digital Input #2, Reverse switch
2	Digital Input #3	Boolean	Status of Digital Input #3, Brake switch
3	Digital Input #4	Boolean	Status of Digital Input #4, REGEN Disable Switch
4	Digital Input #5	Boolean	Status of Digital Input #5, Ignition switch
5	Digital Input #6	Boolean	Status of Digital Input #6, Start switch

0x0A5 – Motor Position Information

Byte #	Name	Format	Description
0,1	Motor Angle (Electrical)	Angle	The electrical angle of the motor as read by the encoder or resolver.
2,3	Motor Speed	Angular velocity	The measured speed of the motor
4,5	Electrical Output Frequency	Frequency	The actual electrical frequency of the inverter.
6,7	Delta Resolver Filtered	Angle	This is used in calibration of resolver angle adjustment. The range of this parameter is ±180°. Values between 180° and 360° are shown as negative angle. For example, 270° is equal to -90°, and 190° is equal to -170°.

0x0A6 – Current Information

Byte #	Name	Format	Description
0,1	Phase A Current	Current	The measured value of Phase A current.
2,3	Phase B Current	Current	The measured value of Phase B current
4,5	Phase C Current	Current	The measured value of Phase C current
6,7	DC Bus Current	Current	The calculated DC Bus current.

0x0A7 – Voltage Information

Byte #	Name	Format	Description
0,1	DC Bus Voltage	High Voltage	The actual measured value of the DC bus voltage.
2,3	Output Voltage	High Voltage	The calculated value of the output voltage, in peak line-neutral volts.
4,5	Phase AB Voltage	High Voltage	Measured value of the voltage between Phase A and Phase B
6,7	Phase BC Voltage	High Voltage	Measured value of the voltage between Phase B and Phase C

0xA8 – Flux Information

Byte #	Name	Format	Description
0,1	Flux command	Flux	The commanded flux
2,3	Flux feedback	Flux	The estimated flux
4,5	Id feedback	Current	D-axis current feedback
6,7	Iq feedback	Current	Q-axis current feedback

0x0A9 – Internal Voltages

Byte #	Name	Format	Description
0,1	1.5V Reference voltage	Low Voltage	One of the low voltage references
2,3	2.5V Reference voltage	Low Voltage	One of the low voltage references
4,5	5.0V Reference voltage	Low Voltage	One of the low voltage references
6,7	12V System voltage	Low Voltage	One of the low voltage references

0x0AA – Internal States

Byte #	Name	Format	Description
0,1	VSM State	Internal	0 = VSM Start State 1 = Pre-charge Init State 2 = Pre-charge Active State 3 = Pre-charge Complete State 4 = VSM Wait State 5 = VSM Ready State 6 = Motor Running State 7 = Blink Fault Code State 14 = Shutdown in Process – in key switch mode 1, user has turned the key switch to off position. 15 = Recycle Power State – user must recycle power when the unit is in this state.
2	Inverter State	Internal	0 = Power on State 1 = Stop State 2 = Open Loop State 3 = Closed Loop State 4 = Wait State 5, 6, 7 = *Internal states* 8 = Idle Run State 9 = Idle Stop State 10,11,12 = *Internal states*

3	Relay State	Internal	Bit 0: Relay 1 Status Bit 1: Relay 2 Status Bit 2: Relay 3 Status Bit 3: Relay 4 Status
4 – Bit0	Inverter Run Mode	Internal	0 = Torque Mode 1 = Speed Mode
4 – Bits5-7	Inverter Active Discharge State	Internal	Current Inverter Active Discharge State: 000 (0) = Discharge Disabled 001 (1) = Discharge Enabled, waiting 010 (2) = Performing Speed Check 011 (3) = Discharge Actively occurring 100 (4) = Discharge Completed All other states are reserved for future use.
5	Inverter Command Mode	Internal	0 = CAN Mode 1 = VSM Mode When in CAN Mode the inverter takes commands from the CAN messages. When in VSM Mode the inverter takes messages from the Vehicle State Machine which is operated from the various input and outputs of the inverter.
6 - Bit0	Inverter Enable State	Internal	0 = Inverter is disabled 1 = Inverter is enabled
6 – Bit7	Inverter Enable Lockout	Internal	0 = Inverter can be enabled 1 = Inverter cannot be enabled This feature is added so that the inverter cannot be accidentally enabled. This feature requires that before sending out an Inverter Enable command, the user must send out a Inverter Disable command. Once the inverter sees a Disable command, the lockout is removed and PM can receive the Inverter Enable command.
7	Direction Command	Internal	1 = Forward 0 = Reverse, if inverter is enabled Stopped, if inverter is disabled

0x0AB – Fault Codes

Byte #	Name	Format	Description

0,1	POST Fault Lo	Internal	Each bit represents a fault
2,3	POST Fault Hi	Internal	Each bit represents a fault
4,5	Run Fault Lo	Internal	Each bit represents a fault
6,7	Run Fault Hi	Internal	Each bit represents a fault

7929 SW Burns Way
Suite B
Wilsonville, OR

Phone: 503 344-5085
Fax: 503 682-9014
sales@rinehartmotion.com

POST Faults

CAN Byte	CAN Bit	POST Fault	Fault Word	Byte Value
Byte 0	0	Hardware Gate/Desaturation Fault	00000001	1
	1	HW Over-current Fault	00000002	2
	2	Accelerator Shorted	00000004	4
	3	Accelerator Open	00000008	8
	4	Current Sensor Low	00000010	16
	5	Current Sensor High	00000020	32
	6	Module Temperature Low	00000040	64
	7	Module Temperature High	00000080	128
Byte 1	8	Control PCB Temperature Low	00000100	1
	9	Control PCB Temperature High	00000200	2
	10	Gate Drive PCB Temperature Low	00000400	4
	11	Gate Drive PCB Temperature High	00000800	8
	12	5V Sense Voltage Low	00001000	16
	13	5V Sense Voltage High	00002000	32
	14	12V Sense Voltage Low	00004000	64
	15	12V Sense Voltage High	00008000	128
Byte 2	16	2.5V Sense Voltage Low	00010000	1
	17	2.5V Sense Voltage High	00020000	2
	18	1.5V Sense Voltage Low	00040000	4
	19	1.5V Sense Voltage High	00080000	8
	20	DC Bus Voltage High	00100000	16
	21	DC Bus Voltage Low	00200000	32
	22	Pre-charge Timeout	00400000	64
	23	Pre-charge Voltage Failure	00800000	128
Byte 3	24	EEPROM Checksum Invalid	01000000	1
	25	EEPROM Data Out of Range	02000000	2
	26	EEPROM Update Required	04000000	4
	27	Reserved	08000000	8
	28	Reserved	10000000	16
	29	Reserved	20000000	32
	30	Brake Shorted	40000000	64
	31	Brake Open	80000000	128

7929 SW Burns Way
Suite B
Wilsonville, OR

Phone: 503 344-5085
Fax: 503 682-9014
sales@rinehartmotion.com

RUN Faults

CAN Byte	CAN Bit	RUN Fault	Fault Word	Byte Value
Byte 4	32	Motor Over-speed Fault	00000001	1
	33	Over-current Fault	00000002	2
	34	Over-voltage Fault	00000004	4
	35	Inverter Over-temperature Fault	00000008	8
	36	Accelerator Input Shorted Fault	00000010	16
	37	Accelerator Input Open Fault	00000020	32
	38	Direction Command Fault	00000040	64
	39	Inverter Response Time-out Fault	00000080	128
Byte 5	40	Hardware Gate/Desaturation Fault	00000100	1
	41	Hardware Over-current Fault	00000200	2
	42	Under-voltage Fault	00000400	4
	43	CAN Command Message Lost Fault	00000800	8
	44	Motor Over-temperature Fault	00001000	16
	45	Reserved	00002000	32
	46	Reserved	00004000	64
	47	Reserved	00008000	128
Byte 6	48	Brake Input Shorted Fault	00010000	1
	49	Brake Input Open Fault	00020000	2
	50	Module A Over-temperature Fault[7]	00040000	4
	51	Module B Over-temperature Fault[7]	00080000	8
	52	Module C Over-temperature Fault[7]	00100000	16
	53	PCB Over-temperature Fault[7]	00200000	32
	54	Gate Drive Board 1 Over-temperature Fault	00400000	64
	55	Gate Drive Board 2 Over-temperature Fault[7]	00800000	128
Byte 7	56	Gate Drive Board 3 Over-temperature Fault[7]	01000000	1
	57	Current Sensor Fault	02000000	2
	58	Reserved	04000000	4
	59	Reserved	08000000	8
	60	Reserved	10000000	16
	61	Reserved	20000000	32
	62	Resolver Not Connected	40000000	64
	63	Inverter Discharge Active	80000000	128

[7] This is a new fault used only for Gen-3 board which is used in all PM150 units.

7929 SW Burns Way
Suite B
Wilsonville, OR

Phone: 503 344-5085
Fax: 503 682-9014
sales@rinehartmotion.com

0x0AC – Torque & Timer Information

Byte #	Name	Format	Description
0,1	Commanded Torque	Torque	The commanded torque.
2,3	Torque Feedback	Torque	The estimated motor torque based on motor parameters and feedbacks.
4,5,6,7	Power on Timer	(Counts x .003) sec	This timer is updated every 3 msec. This timer will roll-over in approximately 5 months.

0x0AD – Modulation Index & Flux Weakening Output Information

Byte #	Name	Format	Description
0,1	Modulation Index	Per-unit Value	This is the modulation index. The scale factor is x100. To get the actual modulation index divide the value by 100.
2,3	Flux Weakening Output	Current	This is the current output of the flux regulator.
4,5	Id command	Current	The commanded D-axis current
6,7	Iq command	Current	The commanded Q-axis current

0x0AE – Firmware Information

Byte #	Name	Format	Description
0,1	EEPROM Version / Project Code	NA	This is an EEPROM version that is assigned to each RMS project. For factory use only!
2,3	Software Version	NA	This is the software version with major and minor release values.
4,5	Date Code (mmdd)	NA	This is the portion of date code that displays month and date information in *mmdd* format.
6,7	Date Code (yyyy)	NA	This is the portion of date code that displays year information in *yyyy* format.

0x0AF – Diagnostic Data

Byte #	Name	Format	Description
Please refer to the manual, “Download Diagnostic Data” for details.			

2.2 Command Message

The Command Message is used to transmit data to the controller. This message is sent from a user-supplied external controller to the PM controller. The Control Message (0x0C0) is used to operate the PM controller via the CAN interface.

0x0C0 – Command Message (Data Length = 8 bytes)

Byte.Bit	Name	Format	Description
0,1	Torque Command	Torque	Torque command used when in torque mode.
2,3	Speed Command	Angular Velocity	Speed command used when in speed mode.
4	Direction	Boolean	0 = CW, 1 = CCW as viewed from the shaft end of the motor.
5.0	Enable	Boolean	0 = Inverter Off, 1 = Inverter On
5.1	Discharge[8]	Boolean	0 = Disable Discharge, 1 = Enable Discharge
6,7	Commanded Torque Limit	Torque	If set to 0, the default torque limits sets in the EEPROM parameters are used. If set to a positive number then the Motor and Regen Torque limits are set to the torque value sent.

This message should be continuously broadcast at 500 milliseconds rate or faster.

If the Command Message is not received faster than the CAN TimeOut time and the Command Message Active Parameter is set to 1 then a CAN Command Message Lost fault will be generated.

When in CAN mode the Command messages should be sent to the controller before the inverter is powered on. If they are not then the Command message Lost fault will have to be cleared up on power up.

Note: Commanded Torque Limit feature was added in software version 1953. For previous versions of software these two bytes should be set to 0 and do not have any function.

[8] Please refer to the document, “RMS Inverter Discharge Process”.

7929 SW Burns Way
Suite B
Wilsonville, OR

Phone: 503 344-5085
Fax: 503 682-9014
sales@rinehartmotion.com

2.2.1 Inverter Enable Safety Options

2.2.1.1 Inverter Enable Lockout:

This feature is added so that the inverter cannot be accidentally enabled. This feature requires that before sending out an Inverter Enable command, the user must send out a Inverter Disable command. Once the inverter sees a Disable command, the lockout is removed and PM can receive the Inverter Enable command.

2.2.2.2 Inverter Enable Safety Switch:

A new EEPROM parameter, CAN Inverter Enable Switch Active EEPROM, is added as a safety option.

Setting this parameter to 1 will take DIN1 digital input into consideration and the inverter will only be enabled if both DIN1 and inverter command are active. If DIN1 or Inverter Enable Command is inactive, the inverter will be disabled.

If CAN Inverter Enable Switch Active EEPROM is set to 0, DIN1 will have no effect on enabling or disabling the inverter.

2.2.2.3 Sudden Reversal of the Direction Command:

This safety feature keeps the user from changing the direction command while the inverter is enabled. If the direction command is changed suddenly when the inverter is still enabled, inverter is disabled without triggering any faults. Also, the lockout condition is set again which will force the user to send an Inverter Disable command before re-enabling it.

2.2.2 CAN Message Sequence Example

Here is an example of sending out torque commands to the inverter in 'CAN' mode with run mode required to be 'Torque'. These two EEPROM parameters can be set via GUI after powering up the inverter.

For the message sequence example described below, following assumptions hold true:

GUI EEPROM Parameter	Default Value	Description
Inv_Cmd_Mode_EEPROM(CAN=0_VSM=1)	0	CAN mode
Run_Mode_EEPROM(Trq=0_Spd=1)	0	Torque mode
CAN_ID_Offset_EEPROM	0xA0	Default CAN ID offset
CAN_TimeOut_(/3ms)_EEPROM	333	1 second timeout period

7929 SW Burns Way
Suite B
Wilsonville, OR

Phone: 503 344-5085
Fax: 503 682-9014
sales@rinehartmotion.com

Message Type	CAN ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Description
Rxd	0xAA	4	0	9	0	**0**	0	**128**	0	Torque mode Is active. Lockout is enabled.
Txd	0xC0	0	0	0	0	0	**0**	0	0	Send out inverter disable command to release lockout. This command should have been set up to be transmitted at a rate of 500 ms or faster before the inverter is powered up.
Txd	0xC0	**100**	**0**	x [9]	x	**1**	**1**	x	x	Enable the inverter with a torque command of +10 Nm in forward direction.
Txd	0xC0	**200**	**0**	x	x	1	1	x	x	Set the torque to +20 Nm (motoring) in forward direction.
Txd	0xC0	**156**	**255**	x	x	1	1	x	x	Set the torque to -10 Nm (regenerative) in forward direction.
Txd	0xC0	X	x	x	x	1	**0**	x	x	Disable the inverter before changing the direction. If the direction is changed without disabling the inverter first. The inverter will be automatically disabled as a safety precaution.
Txd	0xC0	**100**	**0**	x	x	**0**	**1**	x	x	Set the command to +10 Nm (motoring) in reverse direction.

[9] "x" indicates a DON'T CARE value

2.2.3 Sign Convention for Torque and Speed

When using the PM inverter in CAN mode, it is important to make sure that commands are entered properly to move the motor in the intended direction. Following description provides details on speed command, speed feedback, torque command, torque feedback and direction command and the possible outcome in each scenario.

2.2.3.1 CAN Speed command:

The Speed Command is a signed number. If the speed command is positive then the direction will be the direction of the direction command bit. If the Speed Command is negative then the direction will be opposite of the direction of the direction command bit.

2.2.3.2 CAN Torque command:

For a forward direction command:

- Positive torque command will give a positive torque feedback and is motoring for positive speed.
- Negative torque command will give a negative torque feedback and is regen for positive speed.
- Positive torque command will give a positive torque feedback and is regen for negative speed.
- *A negative torque command should not be allowed if already going negative speed.*

For reverse direction command:

- Positive torque command will give a negative torque feedback & is motoring for negative speed.
- Negative torque command will give a positive torque feedback and is regen for negative speed.
- Positive torque command will give a negative torque feedback and is regen for positive speed.
- *A negative torque command should not be allowed if already going positive speed.*

7929 SW Burns Way
Suite B
Wilsonville, OR

Phone: 503 344-5085
Fax: 503 682-9014
sales@rinehartmotion.com

2.3 Parameter Messages

The Parameter Messages (0x0C1 and 0x0C2) are used to read and write parameters in the PM controller. These parameters have many different functions. Some parameters are used to set non-volatile information (EEPROM data). Some are used to change functionality. Some are used to monitor various operating parameters that are not part of the broadcast messages.

To write a parameter use message 0x0C1 with byte #2 set to 1 (write). The PM controller will then respond with message 0x0C2 and if successful byte #2 will be set to 1.

To read a parameter use message 0x0C1 with byte #2 to set 0 (read). The PM controller will then respond with message 0x0C2 containing the requested data.

Both parameter messages contain 4 bytes for the data that is read or written. Some parameters will only occupy a single byte. If the data occupies less than 4 bytes it will be loaded into byte #4 first, followed by #5, and so on.

If the parameter address is not recognized then the controller parameter response message (0x0C2) will contain 0 in both bytes 0 and 1 of the return data.

2.3.1. Parameter Message Format

0x0C1 – Read / Write Parameter Command – sent to PM

Byte #	Name	Format	Description
0,1	Parameter Address	Unsigned int	Each command is identified by a unique address. Refer to sections, 'Command Parameters' and 'EEPROM Parameters' for each parameters address.
2	R/W Command	Boolean	0 = read, 1 = write
3	Reserved	NA	NA
4,5	Data	See "Data Formats" section	Data should be entered as dictated in "Data Formats" section.
6,7	Reserved	NA	NA

0x0C2 – Read / Write Parameter Response – response from PM

Byte #	Name	Format	Description
0,1	Parameter Address	Unsigned int	Each command is identified by a unique address. Refer to sections, 'Command Parameters' and 'EEPROM Parameters' for each parameters address. Will return 0,0 if parameter address is not recognized.
2	Write Success	Boolean	0 = not written, 1 = success
3	Reserved	NA	NA
4,5	Data	See "Data Formats" section	Response data is in the format dictated in "Data Formats" section.
6,7	Reserved	NA	NA

2.3.2. Parameter Address Ranges

The parameters are categorized in several general categories. Some parameters are read-only, and some can be written and read.

Address Range	Category	Description
0 – 99	General	This address range contains general parameters for control and monitoring
100 – 499	User EEPROM	This address range is for EEPROM variables. These can only be written when the PM controller is not operating the motor.

2.3.3. Command Parameters

Address	Name	Format	Description
1	Relay Command	Unsigned integer (0 – 65535)	0xAA00: Normal Run mode 0x55***nn***: External Relay Control mode Please see description below.
10	Flux command	Flux	Modify the flux command.
11	Resolver PWM Delay Command	Unsigned integer (0 – 6250)	This command is used in calibration of the timing of the A/D reading of the resolver. It is used in determining the peak of the sine wave coming from the resolver. Its default value is 1100.
12	Gamma Adjust GUI Command	Degrees	This is a calibration parameter used in the alignment of the magnetic field of the motor with the resolver. This command parameter behaves the same way as its equivalent GUI Command parameter.
20	Fault Clear	Boolean	Writing a 0 to this parameter clears any active faults. This command can be sent through CAN in CAN as well as VSM mode.

Relay Command

This command is used to control relay outputs. The Gen 2 units (older PM100 units) have 4 relay outputs. Gen 3 units, newer PM100, all PM150/250, have 6 relay outputs.

In order to control a relay, the inverter needs to be put into “External Relay Control” mode. This is achieved by setting byte 5 to 0x55. For byte 4, each bit corresponds to a relay. Bit 0 corresponds to relay 1 and bit 1 corresponds to relay 2. Similarly, bit 7 corresponds to relay 8 as shown in the table below. Note, the hardware does not currently support 8 relays.

Byte 5	Byte 4							
0x55	R8	R7	R6	R5	R4	R3	R2	R1

For example, if the user wants to turn on relay 3, he/she needs to set the data field of the parameter command messages to 0x5504. Similarly, if relay 1 and 2 need to be turned on, a command of 0x5503 must be sent.

Setting the data byte 5 to a value other than 0x55 will kick the inverter out of the “External Relay Control” mode and in to the “Normal Run” mode. The default value for normal run mode is 0xAA.

2.3.4. EEPROM Parameters

These Parameters are sent using the command message format described in 0x0C1 to modify the EEPROM parameters used by the PM controller. These parameters can only be written when the motor is not enabled. Each parameter will be stored in non-volatile memory at the time of programming. When the power is recycled the parameter will become effective and used by the inverter. Parameters highlighted in yellow will take immediate effect.

All EEPROM parameters can also be modified through the GUI interface.

2.3.4.1 Motor Configuration[10]

Address	Name	Format	Description
150	Motor Parameter Set	Unsigned char (0 – 255)	This represents a set of parameter that is used for each type of motor.
151	Resolver PWM Delay	Unsigned integer (0 – 6250)	This is used in calibration of the timing of the A/D reading of the resolver. It is used in determining the peak of the sine wave coming from the resolver.
152	Gamma Adjust	Degrees	This is a calibration parameter used in the alignment of the magnetic field of the motor with the resolver. It will automatically default to the correct value when the motor type is changed.
154	Sin Offset	Low Voltage	Please refer to the manual, "RMS Encoder Calibration for SIN_COS Encoder".
155	Cos Offset	Low Voltage	
156	Sin ADC Offset	ADC Count	This feature is dependent on the hardware version of the PM unit. In some cases, the resolver sine and cosine outputs may require adjustments for improved signals. These offsets are added as ADC counts to calibrate the sin and cosine signals directly.
157	Cos ADC Offset	ADC Count	

[10] For a detailed process on how to configure a specific motor, please refer to the document "Setting up the PM for *XYZ* Motor", where XYZ refers to the specific motor for individual customer.

2.3.4.2 System Configuration

Address	Name	Format	Description
140	Pre-charge Bypassed	Boolean	0 = Pre-charge is in effect 1 = Pre-charge is bypassed
142	Inverter Run Mode	Boolean	0 = Torque Mode 1 = Speed Mode
143	Inverter Command Mode	Boolean	This parameter gives the option to operate PM unit in either VSM mode or CAN mode. In VSM mode, GUI is the main interface to monitor and modify parameters. The inverter takes messages from the Vehicle State Machine which is operated from the various inputs and outputs of the inverter such brake, accelerator pedal, etc. In VSM mode, broadcast messages are still sent out over the CAN lines. In CAN mode, both GUI and CAN interfaces are active and can be used to monitor and modify parameters. However, any inputs from the vehicle state machine will be ignored. 0 = CAN Mode 1 = VSM Mode (Default)
149	Key Switch Mode	Unsigned integer	Added alternate key switch modes. This allows different types of ignition for vehicles. 0 = Allows a simple on/off switch for powering up the inverter. 1 = Provides the functionality of a more traditional ignition switch with momentary START signal that powers up the inverter and keeps it powered until the ignition switch is turned off. Key Switch Mode is only effective in VSM Mode. CAN mode remains unaffected. However, the parameter can be updated through both GUI and CAN.
170	Relay Output Command	Unsigned integer	Sets whether the Relay outputs have their normal function or are under CAN control. See the Software manual for more information.

2.3.4.3 CAN Configuration

Address	Name	Format	Description
141	CAN ID Offset	Unsigned integer	This parameter allows the user to choose their own set of contiguous CAN message identifiers starting with the value in CAN ID Offset. This offset covers a range of 0 – 0x7C0. The default offset is 0x0A0. The default range is 0x0A0 – 0x0CF. This feature is especially useful when there are more than one PM controllers on the same CAN network. While setting base address for a controller, it must be made sure that the address range for controllers does not contain overlapping addresses.
144	CAN Extended Message Identifier	Boolean	This parameter allows switching between CAN standard and extended message identifiers. 0 = Standard CAN Messages 1 = Extended CAN Messages
171	CAN J1939 Option Active	Boolean	This parameter allows switching between extended message identifiers with or without SAE J1939 format. 0 = J1939 formatting is not active 1 = J1939 formatting is active
145	CAN Term Resistor Present	Boolean	In order to use CAN communication, the CAN bus needs to be terminated with a 120 Ohm resistor. RMS PM units are equipped with this resistor which is activated through this parameter. 0 = Term. Resistor not active 1 = Term. Resistor active (Default) If CAN Terminator Resistor is deactivated, it may be necessary to use the GUI interface only[11] since CAN communication may fail without a terminator resistor.
146	CAN Command Message Active	Boolean	RMS CAN requires a “heartbeat” command message. This command message controls the inverter, motor direction, and torque or speed. In the absence (time set by CAN TimeOut parameter) of a regular broadcast of this message, PM unit will assume there is a problem and will flag a fault unless the fault has been deactivated by setting this parameter to 0. 0 = The command message 0xC0 is **not** sent every half a second. 1 = The command message 0xC0 is sent every half a second. (Default)

[11] Please refer to the document “Programming EEPROM Parameters using GUI”.

147	CAN Bit Rate	Unsigned integer	250Kbps is the default bit rate. Bus speed can be changed using CAN parameter command message. However, changing this parameter requires a power reset on PM unit since bus speed is setup only at the initialization of CAN modules in the microcontroller. Also, This input is restricted to valid baud rates. The 4 options for valid baud rate are: 125 = 125Kbps 250 = 250 Kbps (Default) 500 = 500 Kbps 1000 = 1Mbps
148	CAN Active Messages Word	Unsigned long integer (32-bits)	This parameter is used to enable/disable CAN Broadcast Messages. Each bit represents a CAN Message broadcast status as follows: 0 = CAN Messages broadcast disabled 1 = CAN Message broadcast enabled (Default) Please refer to the table of CAN Broadcast Messages in section 2.1 for details on how to enable/disable each message.
158	CAN Diagnostic Data Transmit Active	Boolean	This parameter is used to enable/disable the broadcast of the diagnostic data. 0 = CAN Diagnostic Data broadcast disabled 1 = CAN Diagnostic Data broadcast enabled (Default) Please refer to the document, CAN Diagnostic Data, for more details on this feature.
159	CAN Inverter Enable Switch Active	Boolean	1 = DIN1 digital input is taken into consideration and the inverter will only be enabled if both DIN1 and inverter command are active. If either one is inactive, the inverter will be disabled. 0 = DIN1 will have no effect on enabling or disabling the inverter (Default)
172	CAN TimeOut	Unsigned integer	This parameter sets how long before the CAN timeout error is set. The timeout is only active if the CAN Command Message Active is set to 1. The time is set in counts of 3ms. So for example setting a value of 333 will give a timeout time of 1 second.

2.3.4.4 Current

Address	Name	Format	Description
100	Iq Limit	Current	This parameter sets the Q-axis current limit.
101	Id Limit	Current	This parameter sets the D-axis current limit.
107	Ia Offset EEPROM	ADC Count	The offset values are in ADC counts. Each offset should be set to 2048 which is the 0-Amps mid-point.
108	Ib Offset EEPROM		
109	Ic Offset EEPROM		

2.3.4.5 Voltage & Flux

Address	Name	Format	Description
102	DC Voltage Limit	High Voltage	This parameter sets the over-voltage limit to protect the unit. This limit should be set based on total voltage provided by the power supply/battery pack. This parameter does not generate any faults.
103	DC Voltage Hysteresis	High Voltage	This is the hysteresis value used to bring the inverter out of the over-voltage condition. Most of the time, the default value is sufficient and this value seldom needs to be changed.
104	DC Under-voltage Limit	High Voltage	This parameter sets the under-voltage limit. This limit should be set based on total voltage provided by the power supply/battery pack. A fault is generated when the voltage drops below this limit. To disable the under-voltage fault, set this limit to 0.
106	Vehicle Flux Command	Flux	This parameter sets the back EMF (flux) constant for the motor. It will automatically default to the correct value when the motor type is changed. Most of the time, the default value is sufficient and this value seldom needs to be changed.

2.3.4.6 Temperature

Address	Name	Format	Description
112	Inverter Over-Temperature	Temperature	This parameter sets the Inverter temperature limit. The temperature is set in degrees Celsius times 10 (85°C is set as 850). If the temperature exceeds this value then the inverter will turn off and declare a fault.
113	Motor Over-Temperature	Temperature	This parameter sets the Motor temperature limit (if the motor has a temperature sensor). The temperature is set in degrees Celsius times 10 (150°C is set as 1500). If the temperature exceeds this value then the inverter will turn off and declare a fault.
114	Zero Torque Temperature	Temperature	Temperature threshold where the torque is zeo. This value should be less than Mtr_OverTemp_Limit_EEPROM_(C)_x_10.
115	Full Torque Temperature	Temperature	This is the temperature threshold where the full torque is operational. This parameter affects the calculation of torque capability. Based on the calculation of the slope and offset of the line from Full_Torque_Temp_EEPROM_(C)_x_10 to Zero_Torque_Temp_EEPROM_(C)_x_10, the new torque capability is reduced by a factor of (slope * Motor Temperature + offset). This parameters should be less than Zero_Torque_Temp_EEPROM_(C)_x_10 which should be less than Mtr_OverTemp_Limit_EEPROM_(C)_x_10.
203	RTD Selection	Boolean	This parameter allows the user to select either 100 Ohms or 1000 Ohms for RTD1 and RTD2 (Gen 3 units only). Bit 0 = 0 -> RTD1 is 1000 Ohms = 1 -> RTD1 is 100 Ohms Bit 1 = 0 -> RTD2 is 1000 Ohms = 1 -> RTD2 is 100 Ohms

2.3.4.7 Accelerator Pedal[12]

Address	Name	Format	Description
120	ACCEL Pedal Low	Low Voltage	This parameter sets a limit below which the torque command is 0. This should be set to a value that is lower than the lowest possible acceleration position but higher than 0. If the accelerator input goes below this value, torque command is set to 0, the inverter will turn off and declare the ACCEL SHORTED fault.
121	ACCEL Pedal Min	Low Voltage	This parameter sets a limit such that between that limit and ACCEL Pedal Low, torque command is set to a constant value set through REGEN Torque Limit.
122	ACCEL Coast Low	Low Voltage	This parameter sets a limit such that between that limit and ACCEL Pedal Min, torque command is linear from REGEN Torque Limit to 0.
123	ACCEL Coast High		This parameter sets a limit such that between that limit and ACCEL Coast Low, torque command is 0.
124	ACCEL Pedal Max	Low Voltage	This parameter sets a limit such that between that limit and ACCEL Coast High, torque command is linear from 0 to Motor Torque Limit. Normally, this will be the driving range.
125	ACCEL Pedal High	Low Voltage	This parameter sets a limit such that between that limit and ACCEL Pedal Max, torque command is set to a constant value of Motor Torque Limit. This should be set to a value that is higher than the highest possible acceleration position but less than 500. If the accelerator input goes above this value, torque command is set to 0, the inverter will turn off and declare the ACCEL OPEN fault.
132	Accel Pedal Flipped	Boolean	This parameter determines if the pedal increases in value as it is pressed or behaves the other way around. If the pedal increases in voltage as it is pressed use a value of 0 (not flipped). If the pedal decreases in voltage as it is pressed use a value of 1 (flipped).

[12] Please refer to the document "RMS PM Software User Manual" for more details.

2.3.4.8 Torque[13]

Address	Name	Format	Description
129	Motor Torque Limit	Torque	This parameter sets the upper limit of the torque that can be commanded by the controller in motoring mode. However, if the current limit of the drive is reached before the torque command has been achieved the controller will limit on the current first. If this happens the operator will feel an additional amount of unused pedal range at the top end.
130	REGEN Torque Limit	Torque	This parameter sets the upper limit of torque that is commanded when the accelerator pedal is released. Normally this value would be set to a small percentage of the available motor braking torque, to simulate engine braking. This value only sets the torque limit when the pedal is released with no brake applied. The torque applied when the brake is active is set by a separate parameter (Braking Torque Limit).
131	Braking Torque Limit	Torque	This parameter sets the amount of the torque applied when the brake is active.
164	Kp Torque	Proportional Gain	This parameter sets the proportional gain for the torque regulator. This is a times 10000 value. Multiply the value within the valid range by 10000 before programming. Input is restricted to a valid range as indicated in the format table.
165	Ki Torque	Integral Gain	Integral gain for the torque regulator. This is a times 10000 value. Multiply the value within the valid range by 10000 before programming. Input is restricted to valid range as indicated in the format table.
166	Kd Torque	Derivative Gain	Derivative gain for the torque regulator. This is a times 100 value. Multiply the value within the valid range by 100 before programming. Input is restricted to valid range as indicated in the format table.
167	Klp Torque	Low-Pass Filter Gain	Low-pass filter gain for the torque regulator. This is a times 10000 value. Multiply the value within the valid range by 10000 before programming. Input is restricted to valid range as indicated in the format table.

[13] Please refer to the document “RMS PM Software User Manual” for more details.

168	Torque Rate Limit	Torque	This parameter sets the filtering for the torque command. During filtering, the change in torque command is limited to this value. The smaller this value is the slower the ramp for the torque command will be and vice versa. This parameter is limited between 0.1 and 25 Nm.

2.3.4.9 Speed[14]

Address	Name	Format	Description
111	Motor Over-speed	Angular Velocity	This parameter sets the over-speed value for the motor. If motor speed exceeds this value, inverter will turn off and declare the MOTOR OVERSPEED fault.
128	Max Speed	Angular Velocity	The parameter sets the maximum allowable speed. If the speed is above this value the torque command will be reduced to zero.
126	REGEN Fade Speed	Angular Velocity	The parameters sets at which the amount of REGEN torque available is reduced.
127	Break Speed	Angular Velocity	The parameter sets the speed at which the maximum torque command is reduced to compensate for a reduction of available torque due to field weakening.
160	Kp Speed	Proportional Gain	Proportional gain for the speed regulator. Input is restricted to valid range as indicated in the format table.
161	Ki Speed	Integral Gain	Integral gain for the speed regulator. Input is restricted to valid range as indicated in the format table.
162	Kd Speed	Derivative Gain	Derivative gain for the speed regulator. Input is restricted to valid range as indicated in the format table.
163	Klp Speed	Low-Pass Filter Gain	Low-pass filter gain for the speed regulator. Input is restricted to valid range as indicated in the format table.
169	Speed Rate Limit	Speed	This parameter sets the filtering for the speed command. During filtering, the change in speed command is limited to this value. The smaller this value is the slower the ramp for the speed command will be and vice versa. This parameter is limited between 100 and 5100 RPM.

[14] Please refer to the document "RMS PM Software User Manual" for more details.

7929 SW Burns Way
Suite B
Wilsonville, OR

Phone: 503 344-5085
Fax: 503 682-9014
sales@rinehartmotion.com

2.3.4.10 Shudder Compensation[15]

Address	Name	Format	Description
187	Shudder Compensation Enable	Boolean	0: Shudder compensation feature is disabled. 1: Shudder compensation feature is enabled.
188	Kp Shudder	Counts x 100	This parameter provides shudder compensation gain. It is entered as a x 100 value.
189	TCLAMP Shudder	Torque	This parameter provides maximum compensation torque.
190	Shudder Filter Frequency	Frequency	This parameter provides the frequency value for the shudder filter.
191	Shudder Speed Fade	Angular velocity	The parameter provides a value such that from 0 speed to this value, the shudder compensation begins to fade from 0 to shudder torque, and vice versa.
192	Shudder Speed Low	Angular velocity	The parameter provides the value at which the shudder compensation begins to fade to 0.
193	Shudder Speed High	Angular velocity	The parameter provides the value at which the shudder compensation is 0.

[15] Please refer to the document "Shudder Compensation Manual" for more details.

2.3.4.11 Brake Pedal

Address	Name	Format	Description
180	Brake Mode	Boolean	This parameter is used to switch mode as follows: 0: Brake switch mode allows using brake as a binary switch input. Also, this mode activates the 'Automatic REGEN Torque Ramp Down' feature using the EEPROM Parameter REGEN Ramp Period. 1: Brake Pot mode allows using brake as a variable input between 0 – 5 volts. This mode activates the 'Automatic REGEN Torque Ramp Down' feature using EEPROM parameters from address 181 – 184 below.
181	Brake Low	Low Voltage	This parameter is used in Brake mode 1. This parameter sets the lower limit of the brake pot below which brake short fault is set.
182	Brake Min	Low Voltage	This parameter is used in Brake mode 1. This parameter sets the minimum limit for calculating the braking torque.
183	Brake Max	Low Voltage	This parameter is used in Brake mode 1. This parameter sets the maximum limit for calculating the braking torque.
184	Brake High	Low Voltage	This parameter is used in Brake mode 1. This parameter sets the high limit above which brake open fault is set.
185	REGEN Ramp Period	(Counts x 0.001) sec	This parameter is used in Brake mode 0. This value of time is entered in milliseconds. This is the time in which REGEN torque value ramps down to the braking torque limit.
186	Brake Pedal Flipped	Boolean	0: Brake pedal reads 0 V when completely released and 5 V when completely pressed. 1: Brake pedal reads opposite to the above.
199	Brake Input Bypassed	Boolean	This parameter decides if the brake input should be ignored or not in VSM mode: 0: Do not ignore brake input (process as usual) 1: Ignore brake input (brake is considered OFF)

7929 SW Burns Way
Suite B
Wilsonville, OR

Phone: 503 344-5085
Fax: 503 682-9014
sales@rinehartmotion.com

2.3.4.12 Reserved *(for future use)*

Address	Name	Format	Suggested Category
153, 194-198	**Reserved**	**Reserved**	**For factory use only**
105	TBD	TBD	Voltage, or Flux
110	TBD	TBD	Speed
116-119	TBD	TBD	Temperature
133-139	TBD	TBD	Accelerator Pedal or Speed
173-179	TBD	TBD	System Configuration
200-499	TBD	TBD	TBD

7929 SW Burns Way
Suite B
Wilsonville, OR

Phone: 503 344-5085
Fax: 503 682-9014
sales@rinehartmotion.com

Revision History

Version	Description of Versions / Changes	Responsible Party	Date
0.1	Initial version	Chris Brune	9/30/2009
0.2	• Moved Inverter Command Mode (CAN Mode Type) and Inverter Run Mode from 0x0C0 – Command Message to 0x0AA – Internal States • Added more EEPROM parameters.	Azam Khan	11/17/2009
0.3	• For Parameter Messages, changed the range of General and User EEPROM parameters from 1 – 99 and 100 – 500 to 0 – 99 and 100 – 499 respectively. • Added a broadcast rate of 5 msec for the command message (0x0C0). • Added messages 105, 106, and 145 • Changed messages 110 and 141 to be reserved for future use.	Azam Khan	11/30/2009
0.4	• Changed message 104 to “Reserved” for future use. • Changed Message 141 to “CAN ID Offset” and added its description in Broadcast Messages section and also in System Configuration table. • Added message 146 as “Restricted” for factory use only.	Azam Khan	12/02/2009
0.5	• Added a new address 147 for CAN Bit Rate in table “System Configuration” for Parameter Messages. • Updated the description column for Motor Parameter Set to add more motor types.	Azam Khan	01/25/2010
0.6	• Added an example on the use of little-endian format for CAN data bytes	Azam Khan	01/27/2010
0.7	• Added a new address 113 for Motor Over-temperature in table Fault Limits for Parameter Messages. • Added Motor Temperature to the broadcast message 0x0A2. • Added speed regulator gains	Azam Khan	02/18/2010
0.8	• Added Speed Regulator Gains (section 2.3.4.6) • Updated section 1.2 with the formats of Speed Regulator Gains. • Corrected the description for Fault Clear in section 2.3.3.	Azam Khan	02/22/2010
0.9	• Added a new address 148 for CAN Active Messages Word in System Configuration table in Parameter Messages section. • Updated section 2.1 Broadcast Messages with details on how to enable/disable broadcast messages. • Added Power-on Timer to the broadcast msg 0x0AC.	Azam Khan	03/24/2010

1.0	• Updated description for Digital Input Status in Broadcast Messages (Message ID 0x0A4) • Added a new address 149 for Key Switch Mode in System Configuration table in Parameter Messages section.	Azam Khan	04/14/2010
1.1	• Added a new address 170 for Pre-charge Output in System Configuration table in Parameter Messages section.	Azam Khan	04/26/2010
1.2	• Added more description on how to disable CAN broadcast messages in section “2.1 Broadcast Messages”.	Azam Khan	04/27/2010
1.3	• Added Delta Resolver Filtered to the broadcast message 0x0A5. • Added a new broadcast message 0x0AD to send Modulation Index and Flux Weakening Output information. • Added Inverter Enable State and Direction Command to byte 6 and 7 respectively, to the broadcast message 0x0AA.	Azam Khan	06/29/2010
1.4	• Added Id- and Iq-commands to the broadcast message 0x0AD. • Updated the format for modulation index in message 0x0AD.	Azam Khan	08/10/2010
1.5	• Changed the description and range for parameters with the format, Angle.	Azam Khan	11/24/2010
1.6	• Rearranged EEPROM parameter sections into categories based on the type of parameters. • Described CAN configuration in more detail and also added it to the introduction section.	Azam Khan	12/1/2010
1.7	• Added two more parameters in section CAN Configuration: - CAN Extended Message Identfiers - CAN J1939 Option Active • Added two new EEPROM parameter sections: - Shudder compensation - Brake Pedal • Updated the format table for Kp, Ki, Kd, and Klp gains.	Azam Khan	1/19/2011
1.8	• In section 2.3.1, changed the number of data bytes from 4 bytes (4, 5, 6, 7) to 2 bytes (4, 5). • Added a new section on “CAN Diagnostics Parameter Overview” in section 1.	Azam Khan	3/15/2011
1.9	• Corrected the example on page 10 for the direction command which is supposed to be on Byte 4 and Inverter command on Byte 5.	Azam Khan	4/4/2011

2.0	• Updated section 1.3, CAN Format, to elaborate CAN command using example values. • Updated section 2.1, Broadcast Messages, Message 0xAC to show the correct resolution for the Power on Timer. • Added description for several EEPROM parameters. • Clarified that the CAN Command Message must be broadcast every 500 msec or faster.	Azam Khan	6/1/2011
2.1	• Added a new section 1.5 for the CAN Database file.	Azam Khan	6/20/2011
2.2	• Added Resolver PWM Delay Command and Gamma Adjust GUI Command to provide the Resolver Calibration Process through CAN. • Updated section 2.3.4 EEPROM Parameters on how EEPROM parameters are programmed. • Added Inverter Enable Lockout bit to message ID 0x0AA in section 2.1, Broadcast Messages.	Azam Khan	9/11/2011
2.3	• Added a new torque parameter, 'Torque Rate Limit' that controls the ramp rate for torque command.	Azam Khan	12/07/2011
2.4	• Added two new messages 0xAE (Firmware information) and 0xAF (diagnostic data). • Added a new parameter CAN Diag Data Tx Active that controls the broadcast of diagnostic data. • Updated section 2.1 to show all faults with different assignment views.	Azam Khan	9/6/2012
2.5	• Added the "discharge" command to the heartbeat command message.	Azam Khan	11/5/2012
2.6	• Updated section 1.1: Added a new EEPROM parameter CAN Inverter Enable Switch EEPROM. • Added a new section 2.2.1, Inverter Enable Safety Options • Updated section 2.3.4.3, CAN Configuration for the new EEPROM parameters, CAN Inverter Enable Switch EEPROM.	Azam Khan	12/5/2012
2.7	• SWRP 1805 • Updated broadcast message 0x0AB for the two new faults, "Resolver Not Connected" and "Inverter Discharge Active".	Azam Khan	12/13/2012
2.8	• Added description regarding the limited J1939 CAN Messaging option in section 1.1.	Azam Khan	02/05/2013
2.9	• Updated section 2.2.1 Inverter Enable Safety Options with the safety feature "Sudden Reversal of the Direction Command".	Azam Khan	6/21/2013
3.0	• Added CAN message IDs in extended and J1939 format.	Azam Khan	7/8/2013
3.1	• Updated section, Command Parameter, for issuing relay commands (Address 1). This change in the method of controlling relays is effective for firmware release 1909 and later.	Azam Khan	10/22/2013

3.2	• Added a new section 2.2.2 to provide example to the user regarding an example of CAN message sequence. • Added section 2.2.3 which explains the sign convention for torque and speed commands. • Highlighted EEPROM parameters that will take effect immediately as opposed to the parameters that take effect after power cycle.	Azam Khan	11/07/2013
3.3	Updated the section, 1.4 Data Formats to cover each data type to its full range.	Azam Khan	04/02/2014
3.4	Corrected the CAN byte numbers for Run Faults to be 4, 5, 6, and 7 and adjusted bit numbers accordingly.	Azam Khan	04/15/2014
3.5	Corrected the value of -10 Nm to be (255x256) + 156 instead of (255x256) + 246 (which is equal to -1 Nm).	Azam Khan	07/28/2014
3.6	Removed the references to the GUI addresses. GUI addresses are correct in the defsyms files and may change based on the availability of memory. Mentioning these addresses in the documentation is redundant.	Azam Khan	8/12/2014
3.7	Added CAN Timeout count EEPROM parameter (implemented in firmware 1935). Clarification added to the command message formatting. Added some clarification to the parameter message.	Chris Brune	12/12/2014
3.8	Updated the Internal States Message (0xAA) to show the Inverter Active Discharge state parameter (byte 4, bits 5 thru 7).	Chris Brune	3/9/2015
3.9	Corrected issue with the number of relay outputs. Added information about Commanded Torque Limit. Corrected information about sign of Speed command. Moved Motor Temperature foldback parameters to the Temperature section.	Chris Brune	8/26/2015