Kyntronics
Innovation in Motion

34700 LAKELAND BLVD
EASTLAKE, OHIO 44023

TECHNICAL REPORT
TR-1211

MODBUS INTERFACE MANUAL

Prepared By Title:	Malcolm Meluch Sr. Electrical Engineer	Date:	8-APR-2019
Checked By Title:		Date:	
Approved By Title:		Date:	

REVISION HISTORY

Revision	Description of Change	Date (MM-DD-YYYY)	Revised by
1	Created	8-APR-2019	MM
2			
3			
4			

TABLE OF CONTENTS

1 PURPOSE AND SCOPE

Documentation for Kyntronics SHA devices presenting a Modbus RTU interface over RS-485. This does not apply to devices incorporating an “SCM” motion controller. It only details the electrical and software aspects of the devices serial interface.

1.1 Overview

The drive is controlling position around velocity around current. The drive’s Motion Engine is used to generate a motion profile based on position, velocity, and accel/decel.

The ram position is measured by a linear transducer feeding a 16 bit bipolar ADC.
This is the feedback signal used by the drive to close the loop around position.

Applied pressure is measured by a transducer on the extend side of the ram piston.
This is not used by the drive, but is readable over Modbus.

There are three modes by which position may be commanded: Jogging, Preloaded Indexes, and Dynamic Index. Refer to the sections below for details.

2 CONTROLLING THE SHA

2.1 Quick Start

1. Establish Modbus connection using address 63 at 115200 baud.
2. Set coil 00001 Off, cancelling Software Disable.
3. Set coil 00005 On, enabling Motion Engine
4. Set coils 04022 and 04023 On, allowing index selection via Modbus.
5. Set coil 04001 On, to trigger preloaded index #0.
6. Set coil 04002 On, to trigger preloaded index #1.

Note that coils 04001 and above are “Meta” coils, which take action when set to one (on).
Do not set up the Modbus host to continually refresh these addresses. The drive will attempt to restart the commanded action on each scan.

The preloaded indexes vary from device to device.
Typically indexes 0 and 1 are set near the limits of travel for testing.
To load your own index values, see the Dynamic Index section.

2.2 Scaling for your device

The following parameters are needed to customize this generic manual to your device:

1. Inches/Volt of the feedback device.
2. Offset in Volts of the feedback device.
3. Current capacity of the drive in Amperes.
4. Cylinder diameter in inches.

The first two are available from the Feedback section of the DriveWare file used to set up the drive. The other two are part of the design documents.

The drive accepts values in Machine Counts. You need a coefficient to scale between this and engineering units. This manual calls it MCEU – "MachineCounts(per)EngineeringUnit.

Using Inches, MCEU = (65536/20) / (Inches/Volt)

Using milliMeters MCEU = (65536/20) / (Inches/Volt * 25.4)

The ram position is measured by a linear transducer feeding a 16 bit bipolar ADC.

The ADC is readable at full resolution at Modbus 30258 (MSW) & 30259 (LSW). *[MSW – Most Significant Word, LSW – Least Significant Word]*

You may have noticed that all the information is in the LSW. Don't try to save time by reading just the LSW. The Modbus interface in the drive gets upset.

Divide this value by MCEU to calculate position in engineering units.

The ADC is also readable at Modbus word 30016, at reduced (14 bit) resolution.

It must also be adjusted for the Offset in Volts (Parameter #2 above).

Why would you want to use this instead of the full resolution value? Speed over Modbus.

It can be read as a single word. In addition, the pressure sensor is 3 words away, so you can combine reading both into a single transaction.

Precalculate the ADCOffset as follows:

ADCOffset = OffsetInVolts * 16384 / 20

Then position can be read as:

Engineering units = ([30016] – ADCOffset) / (MCEU * 4)

A fluid pressure sensor on the extend side of the hydraulic circuit measures applied force.

This 300PSI/Volt sensor is read by a 12 bit bipolar ADC, readable at Modbus word 30019.

PSI = [30019] * 300 * 20 / 16384

Convert pressure to force using the CylinderDiameter, parameter #4 above.

Force = PSI * Area = PSI * CylinderDiameter^2 / (4 * Pi)

This can all be boiled down to a single precalculated coefficient:

Q = 300 * 20 * CylinderDiameter ^ 2 / (16384 * 4 * Pi)

Force = [30019] * Q

2.3 Commanding Motion

2.3.1 Jogging

Two Modbus “coils” command jogging extend and retract.
Jogging is defined as motion at a constant speed for as long as the command signal is true.
The drive has four selectable adjustable jog speeds plus accel and decel

In order to use Jogging:

1. SoftwareDisable must be off.
2. MotionEngineEnable must off.
3. The active configuration must be 0. (Default)

Since these are all default conditions, Jogging is an ideal “hello world” mode for the EHA.
Once Modbus communications is established, raising JogForward should bring the device to life. When doing so, care should be taken to prevent damage over the entire range of motion.

There are four jog speeds, which are selectable by the JogSel coils.
Each is adjustable via Modbus. Changes you make will be forgotten upon powerdown.
“EU” signifies Engineering Units; the ones you used to calculate MCEU in section 2.3.

40310 = Jog accel rate LSW = EU/Sec^2 * MCEU * 4 * 65536 / (20000 * 20000)
40311 = Jog accel rate MSW
40312 = Jog decel rate LSW “
40313 = Jog decel rate MSW
40314 = Jog speed 0 LSW = EU/Sec * MCEU * 2 * 65536 / 20000
40315 = Jog speed 0 MSW
40316 = Jog speed 1 LSW
40317 = Jog speed 1 MSW
40318 = Jog speed 2 LSW
40319 = Jog speed 2 MSW
40320 = Jog speed 3 LSW
40321 = Jog speed 3 MSW

2.3.2 Preloaded Indexes

This mode makes use of sixteen preloaded (programmed) indexes, which are not adjustable by Modbus. Adjusting them requires a software package called DriveWare from the drive manufacturer.

Each index moves to a preset position with its own accel, decel, and max speed value.

In order to use Programmed Indexes:

1. SoftwareDisable must be off.
2. MotionEngineEnable must on.
3. The active configuration must be 0. (Default)
4. Meta Coils 04022 and 04023 must be set. Once on powerup will do.

An index is started by setting its corresponding Meta Coil 04001-04016.

Setpoint ramping will run to completion unless stopped by AbortMotion or an error.

Typically the factory only sets two preset indexes for testing. They are near the limits of travel.

2.3.3 Stopping Indexes

There are two agents in play during an index: The Motion Engine and the feedback loops.

The Motion Engine drives the *target* position, and the loops do whatever it takes to make it happen. Aborting an index freezes the target position but does not disable the loops. They are always active unless the drive is disabled via software or hardware.

The host controller can monitor the actual ram position as well as the target position. Refer to Modbus 30258-30261. It can command the drive to quit if desired. This can be done by setting Software Disable (Modbus 00001).This turns off the drive bridge. It consumes minimum power and generates minimum motor heat.

Note that triggering Abort Index (Modbus 04017) or turning off the Motion Engine (Modbus 00005) will probably not achieve the desired result. They will stop the target position from moving, but not necessarily drive the force to zero.

2.3.4 Dynamic Index

This mode is similar to one of the Programmed Indexes, except that all of its associated parameters are adjustable by Modbus. These settings are lost when power is removed.

In order to use Dynamic Index:

1. SoftwareDisable must be off.
2. MotionEngineEnable must on.
3. The active configuration must be 0. (Default)

This register bank should be set up before triggering the Dynamic Index with coil 04020.

NOTE: Prefixing a value with “0x” signifies hexadecimal. For example 0x0020 = 32 decimal.

“EU” signifies Engineering Units; the ones you used to calculate MCEU in section 2.3.

40019 = 0x0020
40020 = 0x0008
40021 = 0x0000
40022 = 0x0000
40023 = Target pos LSW = EU * MCEU
40024 = Target pos MSW
40025 = 0x0000
40026 = Speed LSW = EU/Sec * MCEU * 2 * 65536 / 20000
40027 = Speed MSW
40028 = 0x0000
40029 = Accel LSW = EU/Sec^2 / (EU/Sec * 32)
40030 = Accel MSW
40031 = Decel LSW = EU/Sec^2 / (EU/Sec * 32)
40032 = Decel MSW

Notice that the Accel and Decel calculation incorporates the velocity value.

3 INTERFACE

3.1 Modbus RTU Interface

Each EHA' s controls are based on a servo drive with an RS-485 port, speaking Modbus RTU. The default Modbus address is 63. Communications is 115200 baud, 8 data bits, no parity. Serial connection uses J2 on the drive enclosure. Pinout is on sheet 2 of the wiring diagram.

3.1.1 Modbus Coils – boolean signals from host to EHA

In Modbus, addresses 00000-09999 are defined as "Coils". In the PLC-centric original case, coils were controlled by the device itself, and only forcible by the host. In the case of this EHA drive, coils are written by the host, providing single-bit commands to the drive. They can also be read by the host.

NOTE that in this device, coils 04000 and above are "Meta Coils". A Meta Coil triggers some action when a 1 is written to it. The Modbus host need not, and should not, write a 0 to a Meta Coil. Nor should Meta Coils be refreshed as part of a continuous scan. If the value being conveyed was a 1, the commanded action would be continually restarted, with ugly results.

00001 = SoftwareDisable

Setting this bit disables the drive bridge.

00005 = MotionEngineEnable

The Motion Engine must be enabled when using Programmed or Dynamic Indexes.
It must be disabled when jogging.

00013 = Reset Events

Setting this coil high clears non lethal error conditions. Host must also clear it.
The drive will acknowledge with input 10017, and bit 0 of register 30007.
All other fault conditions are handled by meta coil 04034.

00020 = JogForward
00021 = JogReverse

Setting one of these coils commands a jog motion. Clearing it stops the jog.
Refer to the jog section above.

00022 = Jog_Speed_Select LSBit
00023 = Jog_Speed_Select MSBit

There are four jog speeds, all adjustable via Modbus. See register 40310 et al.
These two coils select which of the four jog speeds is used when a jog is commanded.

04001..04016 = Initiate index 0..15

Writing a 1 to one of these Meta Coils triggers the corresponding preset index.

The Modbus host need not and should not write a zero to a Meta Coil.

04017 = Abort Index

Writing a 1 to this Meta Coil stops an index in process.

04020 = Initiate Dynamic Index

Writing a 1 to this Meta Coil triggers the Dynamic Index which you have set up using Modbus registers. See the Dynamic Index section below.

04022 = Select Communications Channel (allow drive to accept index commands over Modbus)

04023 = Select Indexer mode (as opposed to sequencer mode.)

Sequences involve multiple moves and/or delays. No sequences are set up as shipped.

04022 and 04023 Meta Coils should both be set prior to using Indexes.

04034 = Reset Drive Faults

Writing a 1 to this Meta Coil clears faults not related to the Motion Engine.

See also Coil 00013.

3.1.2 Modbus Holding Registers – integer signals from host to EHA

Modbus addresses 40000 thru 49999 are defined as Holding Registers.
Each 16 bit register is readable and writable by the Modbus host.

In addition to the indexes stored in flash memory, the drive allows you to define a “Dynamic Index” which can be modified and triggered via Modbus. See the Dynamic Index section above.
40019 = Move Index
40020 = Move Type
40021 = Repeat Count
40022 = Dwell Time
40023 = Target position LSW
40024 = Target position MSW
40025 = Speed LSW (Typically left as zero, as these counts represents very fine steps)
40026 = Speed 3SW
40027 = Speed 2SW
40028 = Speed MSW (Always zero, as speed is always positive, and “1” would be overspeed.)
40029 = Accel rate LSW
40030 = Accel rate MSW
40031 = Decel rate LSW
40032 = Decel rate MSW

The jog parameters are also adjustable via Modbus. See the Jogging section above.
40310 = Jog accel rate MSW
40311 = Jog accel rate LSW
40312 = Jog decel rate MSW
40313 = Jog decel rate LSW
40314 = Jog speed 0 MSW
40315 = Jog speed 0 LSW
40316 = Jog speed 1 MSW
40317 = Jog speed 1 LSW
40318 = Jog speed 2 MSW
40319 = Jog speed 2 LSW
40320 = Jog speed 3 MSW
40321 = Jog speed 3 LSW

3.1.3 Modbus Inputs – boolean signals from EHA to host

In Modbus, addresses 10000-19999 are referred to as Inputs.
They are single-bit status signals which can be read but not written by the host.
The drive supports almost a hundred inputs, but almost all of them are also readable in groups of 16 as Input Registers. Reading all pertinent inputs as registers is dramatically more efficient.
The only Input not available as a register is:

10193 = Active configuration

This input reads back which of the two configurations is being used.
Refer to the Configuration select coil 00034.

3.1.4 Modbus Input Registers – integer signals from EHA to host

Modbus addresses 30000-39999 are defined as “Input Registers”.
Each 16-bit value can be read but not written by the host.

30001 = Active Mode State

This integer value defines the active drive state:
0 Standby Mode
1 Homing Mode
2 Jog Mode
3 Motion Engine Enabled

30005 = Motion Engine State

This integer value defines the state of the Motion Engine
0 Inactive
1 Waiting for Motion Start (Motion Engine is enabled and ready for an index)
2 Executing Motion (Index is currently running)
3 Program Load in Progress (Motion Engine is not ready for commanded index)
4 Program Load Failure (Problem loading Index. Must reset Motion Engine to continue)
5 Invalid Data Parameter (Problem loading Index. Must reset Motion Engine to continue)
6 Invalid Op-Code (Problem loading Index. Must reset Motion Engine to continue)
7 Halt Asserted (Motion has been interrupted)
8 Invalid Reference Frame (Problem with index parameters)
9 Invalid Bridge State (Bridge must be enabled to begin indexed motion)
10 Invalid Op-code for Dynamic Motion (Problem with index parameters)

30006 = Drive Bridge Status

In this bitmap, each bit signifies an independent condition.
0001h = Bridge is enabled
0004h = Stop is enabled

30007 = Drive Protection Status
In this bitmap, each bit signifies an independent condition.
0001h = Drive is reset
0002h = Drive internal error
0004h = Drive short circuit
0008h = Drive current overshoot
0010h = Drive undervoltage
0020h = Drive overvoltage
0040h = Drive overtemperature

30008 = System Protection Status
In this bitmap, each bit signifies an independent condition.
0001h = Parameter Restore Error
0002h = Parameter Store Error
0004h = Invalid Hall error
0008h = Phase sync error
0010h = Motor overtemperature
0020h = Phase detection fault
0040h = Feedback sensor error
0080h = Motor overspeed
0100h = Max measured position
0200h = Min measured position
0400h = Comm error
1000h = Motion Engine Error
2000h = Motion Engine Abort

30009 = Drive System Status word #1
In this bitmap, each bit signifies an independent condition.
0002h = Software Disable active
0020h = Current limiting active
0040h = Continuous current foldback active
0080h = Current loop saturated
0400h = Non-sinusoidal commutation
1000h = Motion Engine active

30010 = Drive System Status word #2
In this bitmap, each bit signifies an independent condition.

0001h = Zero speed
0002h = At commanded force
0004h = Following error
0008h = Positive force velocity limit
0010h = Negative force velocity limit
0020h = Command limiter active
0040h = Force following error
0080h = Max target position limit
0100h = Min target position limit

30014 = Current Demand (torque producing)
30015 = Current Measured (torque producing)
To convert to Amps, multiply by the drive rated current and divide by 8192.

30016 = ADC Input 1
This measures the raw voltage from the position sensor to 14 bit resolution.
See section 2.3 for scaling.

30017 = ADC input 2
Not connected.

30018 = ADC input 3
Not connected.

30019 = ADC input 4
This measures the voltage from the 300PSI/volt pressure sensor.
Positive corresponds to thrust. The sensor does not function (very well) in suction.
See section 2.3 for scaling.

30258 = Measured position MSW
30259 = Measured position LSW
30260 = Target position MSW
30261 = Target position LSW
These values are in MachineCounts. Divide by MCEU to calculate engineering units.