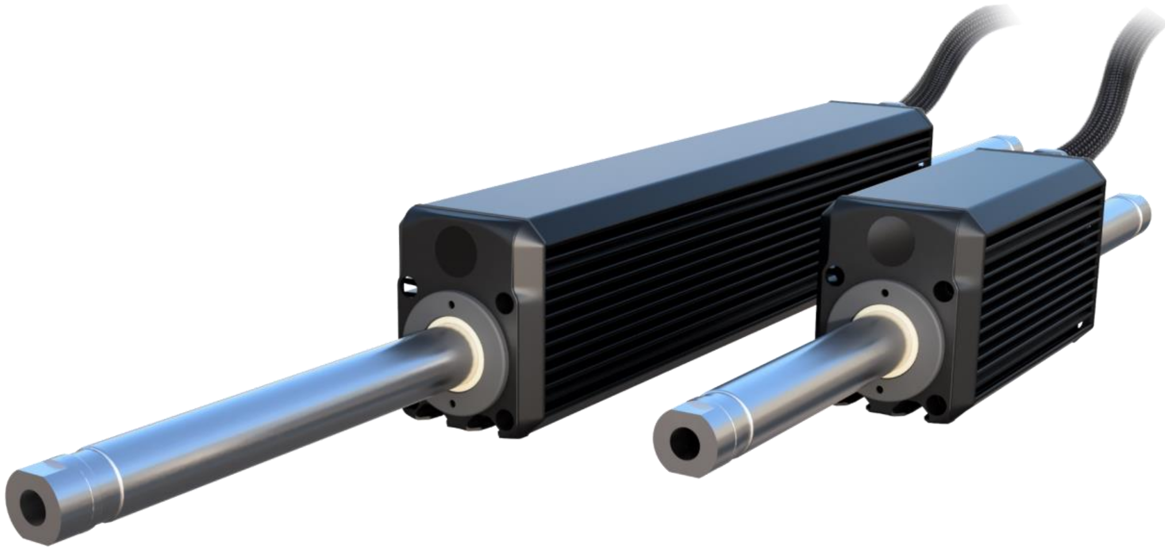
ORCA™ SERIES MOTOR

Reference Manual 220115



This document applies to the following Orca™ Series linear motor firmware:

- 6.2.8

For more recent firmware versions, please download the latest version of this reference manual at https://irisdynamics.com/downloads

CONTENTS

Revision History

Version	Date	Author	Reason
0.0	January, 2022	kh	Initial Draft
1.0	April, 2022	sj	Include GUI and Kinematic sections
1.2	August, 2022	kh	Review and update
1.3	September, 2022	sj	Update for 6.x.5 changes.
1.4	February, 2023	Sj	Update for 6.1.6 changes.
1.5	April, 2023	rm	Formatting, memory map update
1.6	May, 2023	rm, ab, sj	Update GUI, add Haptics section, change section order
1.7	August, 2023	kh, sj	Expand on control register
1.8	December, 2023	sj, rm	Update for 6.1.8 changes, autozeroing, negative positions, safety warnings

Introduction

This document describes the functions and operation of Orca™ Series linear motors having integrated drivers and an integrated sensor suite.

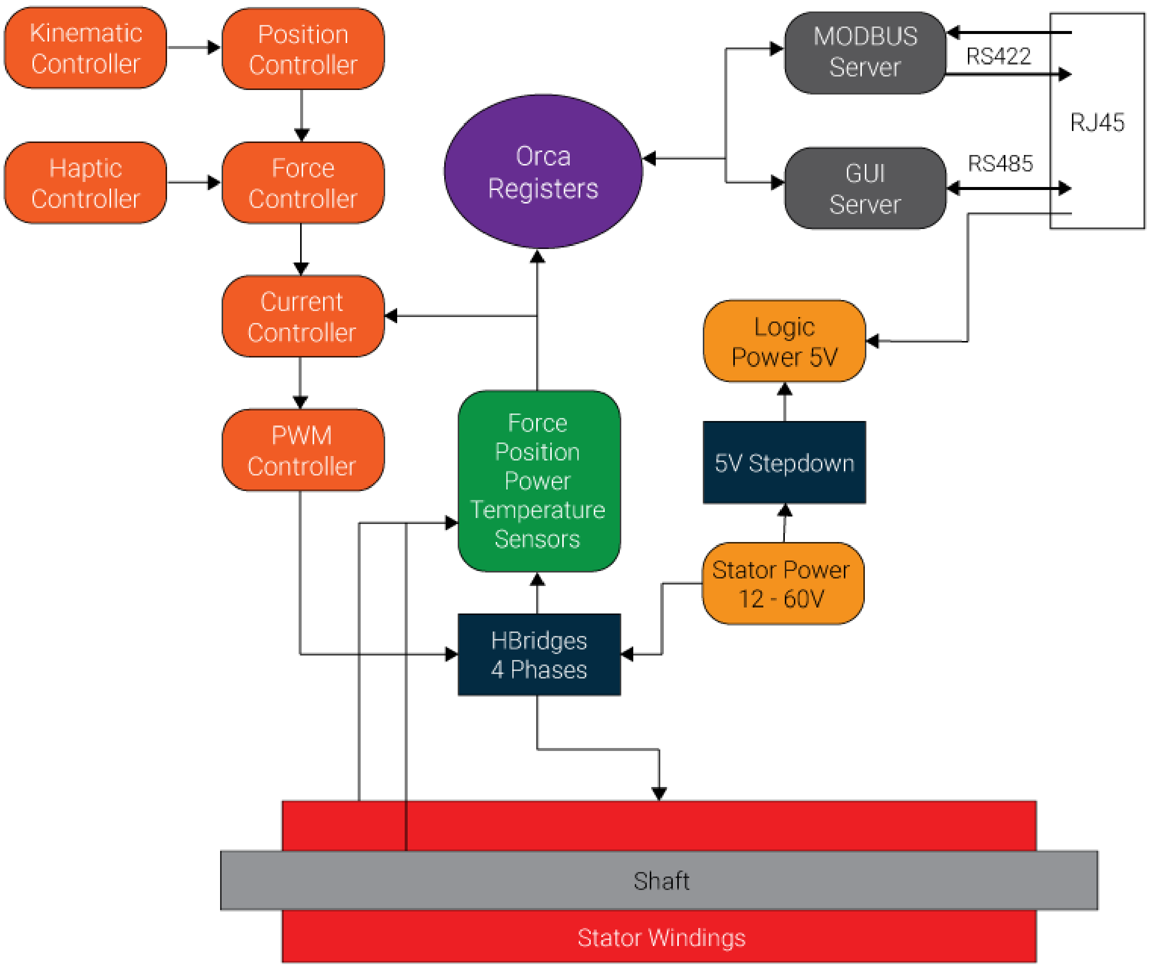


Figure 1: Block Diagram

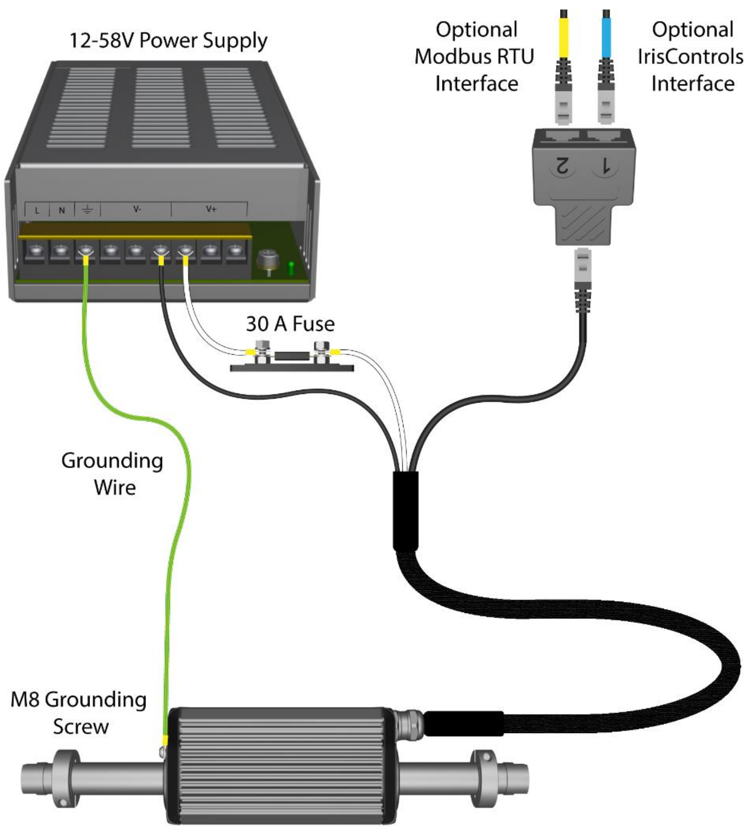


Figure 2: Orca Series Motor Setup

Safety Warnings

⚠WARNING⚠ The motor's shaft contains STRONG magnets. Keep away from ferrous metals and sensitive electronics.

⚠WARNING⚠ Ensure the motor is mounted in a safe location to avoid danger or damage to people or property.

⚠WARNING⚠ Pitching hazard between shaft collars and stator body.

⚠WARNING⚠ On start up the current motor position is zero. Absolute positions are not kept through a power cycle. (BETA feature available for autozeroing on start up).

⚠WARNING⚠ In the event of loss of power the shaft moves freely within the stator.

Powering the Motor

Orca Series motors are powered by DC voltage. The Orca Series variant name specifies the voltage at which the motor is most efficient, however all Orca Series motors can be powered by 12 – 60 VDC.

Flying lead power connections are provided for each motor. Orca Series motors are configured to be able to draw significant currents and proper attachment of the power lines to a suitable power source is important.

Suitable power sources for Orca Series motors are AC-to-DC converters, batteries of various chemistry, capacitor banks, or other sources of DC voltages. Engineering support from Iris Dynamics on the selection of a suitable power source is available.

Fuse

A 30-amp fast-blow fuse must be installed between the power supply and the motor. This fuse will be destroyed if negative voltage is applied to the motor.

Ground

In all cases, the black power lead is "Ground" and should be connected to the negative terminal of the power supply or battery.

It is important that the power supply Ground, and the Ground of any electronics attached to the device's communication cable is kept at the same level. Usually this is done by making sure the power supplies of the motor and external controller (if different) are connected to Earth at both their chassis and negative output terminal.

Chassis

As an important safety measure, the chassis of the motor must be securely connected to Earth. Proper grounding of the chassis is also important for proper operation of the motor.

The motor enclosure and cable shields are connected but these are not connected to the Ground circuit of the on-board electronics. Therefore, a separate connection from the Chassis to Earth must be made which can be accommodated by threading a conductive bolt into any of the provided thread patterns on the enclosure.

Communication Interfaces

Communication cables are terminated with an RJ45 connector. The 5V and GND lines on the connector can be used to power the motor's logic so that communications and diagnosis can continue when the motor's power source is switched off or disconnected.

Pin	Signal	Notes
1	RX_2+	Modbus
2	RX_2-	
3	TX_2+	
4	RX/TX_1+	IrisControls™
5	RX/TX_1-	
6	TX_2-	Modbus
7	+5V	Can be used to power logic, or indicate logic power status
8	GND	Must be connected to the communication hub ground.

There are two serial communication interfaces available which enable a Windows-based GUI (via IrisControls), firmware updates, and a Modbus RTU communication stream. An RJ45 splitter can be used to allow both communication streams simultaneously.

Splitter Port	Signaling	Protocols
1	RS485 Half Duplex	IrisControls (GUI) IrisBootloader
2	RS422 Full Duplex*	Modbus RTU

* See UG230323 – Orca Series Modbus over Half-Duplex RS485 for information on setting up a half-duplex connection on this port.

Methods of Motor Control

Modbus RTU Serial Interface

Orca Series motors feature a 'field-bus' serial communication interface which allows configuration, control, and monitoring. Features of the motors are offered by exposing data fields (registers) which can be written to and read from by sending and receiving characters over the serial interface. Serial communications are implemented using a subset of the Modbus RTU specification, with additional functionality to support a high-speed stream of commands and feedback.

See the Orca Memory Map for a list of register addresses. A PLC, PC, or other Modbus client can be used to command the motor. Third party programs with built in Modbus libraries or serial libraries such as Labview, or Matlab can also be used. There is an IrisSDK for Windows available with C++ libraries that abstract the Modbus communications for use in custom software solutions.

The Modbus RTU User Manual, along with manuals for specific client interfaces, links to git repository with source code are available for download at irisdynamics.com/downloads.

IrisControls Software

Orca Series motors feature an optional graphical user interface called IrisControls which can be used to monitor details and configure settings. This interface provides an easy way to visually tune the internal PID position controller, set up motion profiles, add performance restrictions, and capture information while connected. See the Getting Started With IrisControls section of this document for full details.

Orca IO SmartHub *(optional and sold separately)*

The Orca IO SmartHub provides control of Orca Series motors in Force, Position, and Kinematic Modes through simple digital and analog inputs. Real-time force and position data are fed back from the motor and provided as analog outputs. The IO SmartHub attaches to the motor's data cable (RJ45) and allows easier integration with existing industrial control methods such as PLCs with 4-20 mA current loop outputs. Find more information in the Orca IO SmartHub User Guide at irisdynamics.com/downloads.

Modes of Operation

The motor operates in five main modes of operation: Sleep Mode (1), Force Mode (2), Position Mode (3), Haptic Mode (4), and Kinematic Mode (5). Force Mode (2) and Position Mode (3) are meant to be used with a Modbus communication stream and cannot be entered directly through the IrisControls GUI.

Sleep Mode (1)

In this mode, power drivers are not active and windings in the stator are all shorted (producing a braking effect). Force and position commands are ignored.
When Sleep Mode (1) is commanded (even when Sleep Mode is currently active), any active errors will be re-evaluated and cleared where appropriate.

Force Mode (2)

The force controller will control the power delivered to the motor to achieve the forces written to the FORCE_CMD register. Note: this register is double-wide (32 bits) and read as a signed 32 bit integer.
Once in Force Mode (2), communications must be made regularly over the Modbus interface to avoid a timeout error. In the event of communications failure or other motor errors, the motor will cease all power draw and produce zero force output.

Position Mode (3)

The position controller will calculate forces for and command the force controller according to the configured PID tuning, the setpoint, and the current shaft position.
Before entering Position Mode (3), the user should make sure the shaft has been zeroed to a known position, either manually or through the BETA autozeroing feature.
Once in Position Mode (3), communications must be made regularly over the Modbus interface to avoid a timeout error. In the event of communications failure or other motor errors, the motor will cease all power draw and put the windings into a passive braking effect.

Haptic Mode (4)

In Haptic Mode, the force controller works similarly to Force Mode, however, force commands will be dictated by the haptic controller rather than a Modbus stream. The haptic controller comprises several haptic force effects that can be configured and enabled. The sum of forces resulting from all enabled haptic effects is used as the force target. A Modbus stream can be used to update the constant force effect, or updating other effect configurations.

Kinematic Mode (5)

In Kinematic Mode, the position controller works similarly to when in Position Mode (3), however, position commands will be dictated by the kinematic controller rather than a Modbus stream.

The kinematic controller outputs motion profiles to move to a shaft position over a specified time, while respecting the chosen kinematic constraints. The Modbus interface can be used in this mode to configure motion profiles and trigger motions.

Auto Zeroing Mode (55) (BETA)

This mode is available to put the motor into a routine that will automatically retract the shaft and set the zero position to that location. Once the zeroing routine is complete the motor will move into the specified Exit Mode.

Getting Started with IrisControls

Required Software

Orca Series motors can be configured through an integrated graphical user interface (GUI). To connect to an Orca Series motor's GUI, first download the latest version of the IrisControls software on a Windows PC.

The latest release is always available at www.irisdynamics.com/downloads under the 'Software' section.

Once downloaded, unzip the folder, run the Setup_IrisControls_vX.X.X-... application and allow the installer to complete setup.

Figure 3: IrisControls Software without an Orca Series Motor Connected

Required Hardware

Besides the motor itself, connecting to the IrisControls application requires two additional pieces of hardware detailed below.

Figure 4: Orca Series Motor

Figure 5: RS485 to USB cable

Figure 6: RJ45 Splitter

Connect the RJ45 communication cable from the Orca Series motor to the single port side of the RJ45 splitter. Connect the RJ45 connector end of the blue RS485 cable to the splitter input labeled 1. Finally, connect the USB end of the blue RS485 cable to the Windows PC running IrisControls.

Connecting to IrisControls

Ensure that the motor is connected to a PC with an RS485 to USB cable. Open the IrisControls application. Using the COM selection dropdown menu at the bottom of the window, select either Orca (if it is present) or the COM port of the USB to RS485 connection. If neither option is present, ensure that the hardware is connected correctly as specified.

Figure 7: IrisControls Connection Information

Once the correct device is selected, press the connect button to begin attempting to connect to the motor. If successful, the Orca GUI should launch its homepage.

Troubleshooting Connection

Correct Splitter Port

If the COM port does not show up in the drop down menu, ensure that the blue RS485 to USB cable is plugged into port 1 of the RJ Splitter and that the motor's data cable is plugged into the single side of the splitter.

FTDI Drivers

If the COM port does not show up in the drop down menu, it may be due to not having the FTDI Virtual COM Port drivers installed. Most newer Windows operating systems have these drivers included, but they can also be found at https://ftdichip.com/drivers/vcp-drivers/.

Too Many Rows / Columns Requested Error

This error can arise from a combination of a screen's scaling and the size of the display. The first option to resolve this is to turn down the screen scaling (windows key -> 'scaling' -> "Change the size of text, apps, and other items").

GUI Baudrate

If the motor does not connect, toggle the connect button off and press the gear icon in the top right of the IrisControls window and select a baudrate of 460800 from the dropdown menu, and press apply.

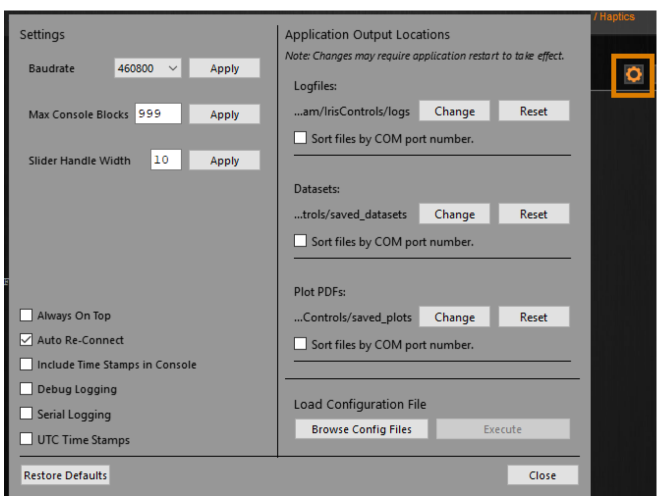


Figure 8: IrisControls Settings Dialog

Navigating the GUI

Navigation of the GUI is done using the page buttons above the console. Pressing each button will open a different page of the GUI within the page content area. The default page upon connection is the home page, shown below. Several other indicators, such as power draw and mode of operation will remain visible on the GUI regardless of the page selection.

Home Page

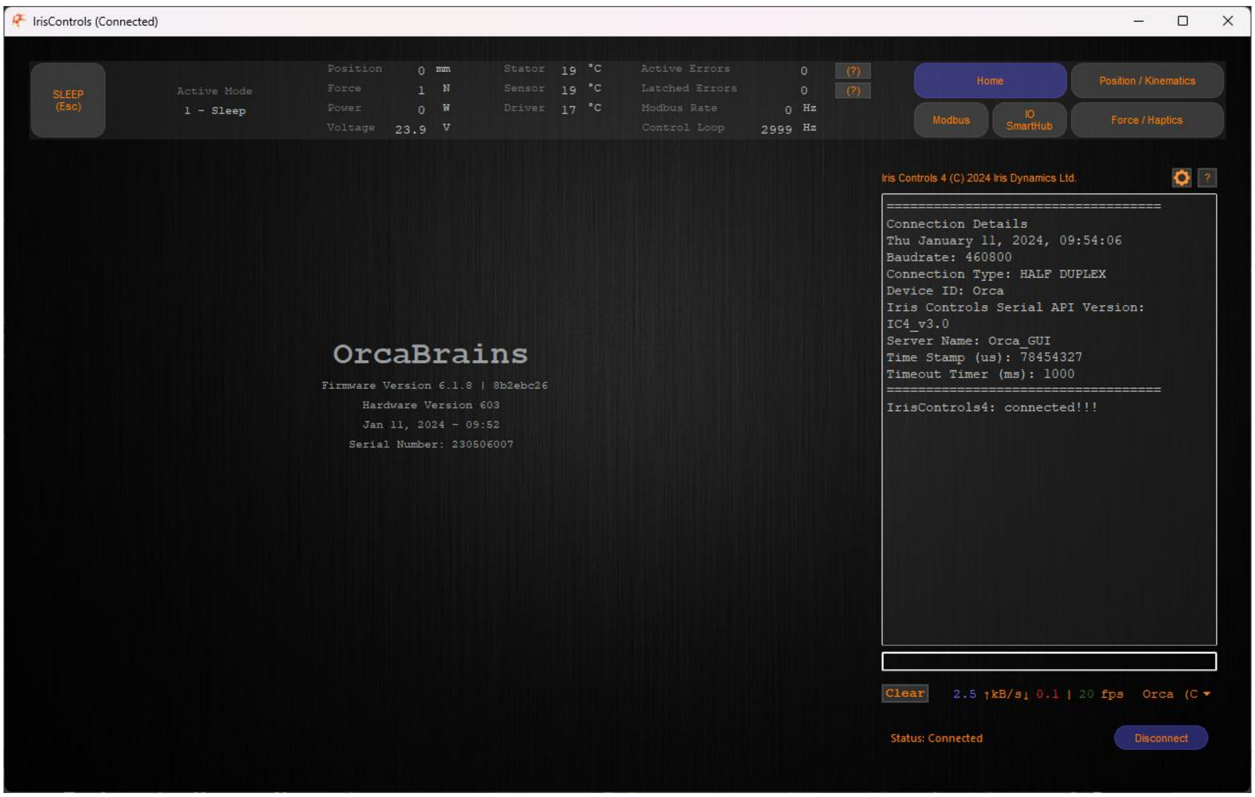


Figure 9: Orca GUI with Home Page Open

The home page is the default page opened upon connecting to an Orca Series motor. A screenshot of the home page is shown in Figure 9. This page displays the firmware version, hardware version, firmware build date, and the motor serial number. There are no interactive elements specific to this page.

Position / Kinematics Page

The position page provides a real time graph displaying the target position, measured position, and target force of the motor. Below the graph, the currently set parameters of the position controller are displayed, including proportional gain, integral gain, derivative gain, and maximum force output. These values can be tuned and are described in further detail in the [PID Position Controller]() section.

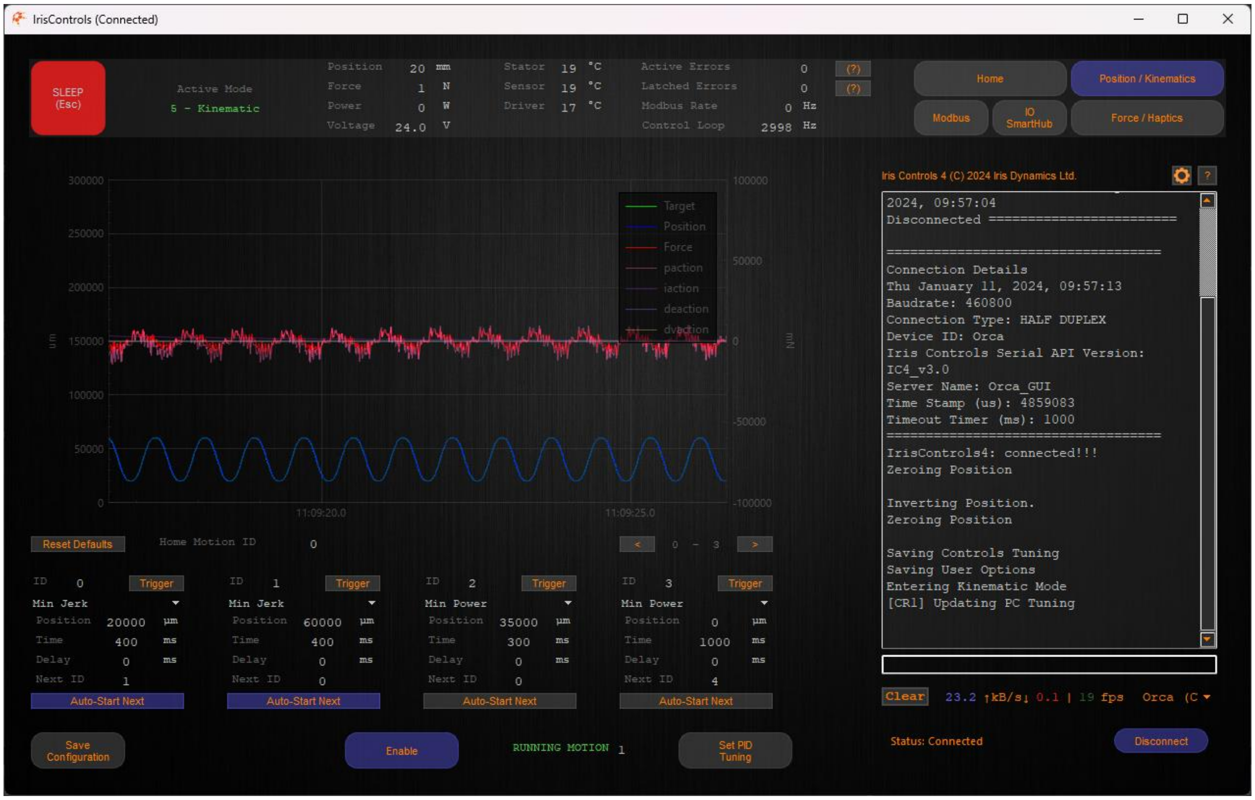


Figure 10: IrisControls Position Page

Position and Force Graph

This graph will plot the position controller's target position, the commanded force used to reach that position, the measured shaft position, and the individual actions of each component of the PID controller. The left y-axis is position, and the right y-axis is force.

The green "Target" line will show the active commanded position. (Modbus or Kinematic Controller).

The blue "Position" line indicates the motor's sensed position.

The other lines indicate the overall force being commanded to the motor and the contributions from each PID parameter to that force.

Tuning Panel

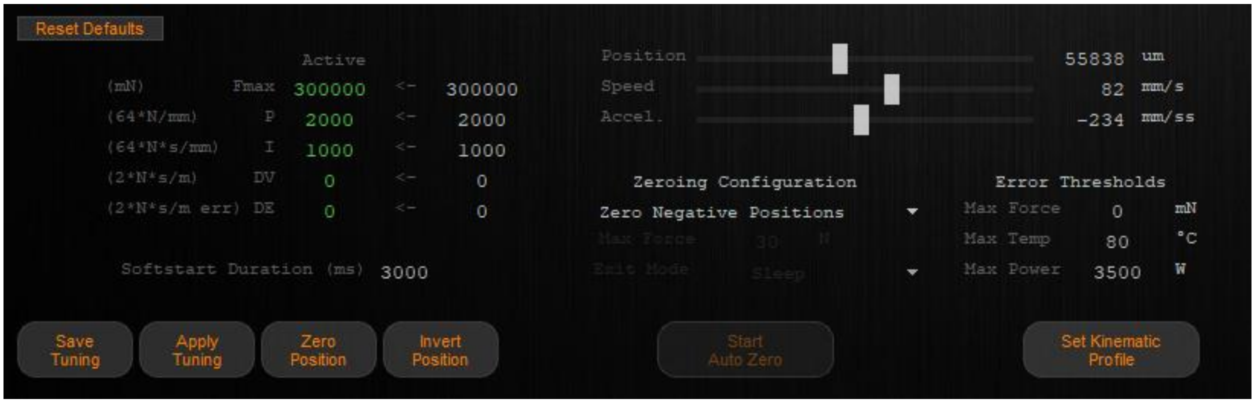


Figure 11: IrisControls Tuning Panel

PID Tuning - Max Force / Gain Adjustment

The maximum force output and position controller gains can be edited here.
The maximum force (Fmax) determines the upper limit of force that the position controller can command.

The number on the left of the arrow is the actual value that is in effect and is not editable. This value will only be populated when the position controller is active either in Position Mode (3) or Kinematic Mode (5)

The number on the right side of the arrow is the target gain value. This value can be manually edited and applied with the Apply Tuning button and saved with the Save Tuning button.

Softstart Duration configures the duration that forces generated by the position controller will linearly ramp up over when entering the mode.

Sensor Sliders

Three sliders show the current values for position, speed and acceleration.

Error Thresholds

Allows configuration of the force, temperature and power thresholds that trigger their respective errors when exceeded.

Save Tuning Button

This button will save the gain target values and the maximum force value to permanent memory to allow them to persist through a power cycle.

Apply Tuning Button

To update the gains to their target values while operating in Position Mode (3), press the update gains button.

Invert Position Button

Change direction of positive movement. This should only be done while the motor is in Sleep Mode (1) to avoid unexpected movement.

Zero Position Button

Set the motor's zero position to the current location. Zero position does not persist through motor power cycles.

Zeroing Configuration

There are four options for zeroing the motor's position.
Zero Negative Positions: moving the motor past its zero position in the negative direction will reset the zero position to the furthest negative position.
Manual Zeroing: Zero position will only be set at start up or when zero position is commanded (through button or Modbus).
Auto Zero Enabled (BETA): In this mode the motor can additionally be zeroed through an automated zeroing routine, this can be configured to use a specified maximum force, and exit into a specified mode. The routine is triggered through the "Start Auto Zero" button or writing to Control Register 3 to place the motor in Auto Zeroing mode.
Auto Zero on Boot (BETA): On startup/reboot the motor will automatically perform the auto zeroing routine and then move into the specified exit mode.

Start Auto Zero Button

This button is available if one of the BETA auto zero configurations are selected. These configurations can be selected from the drop down above the button. This will trigger the zeroing routine and then put the motor into its specified exit mode.

Set Kinematic Profile Button

This button will switch the interface to the kinematic panel to configure kinematic motion profiles.

Kinematic Panel

The kinematic page provides an interface to the kinematic controller feature of the Orca Series motor. For more information on the capabilities of the kinematic controller, refer to the Kinematic Controller section of this manual.

Figure 12: IrisControls Kinematic Panel

Reset Defaults Button

Resets current kinematic settings to the factory default demo motion profile.

Home Motion ID

This is the ID of the motion that will start when the kinematic controller is enabled, either via the GUI, Modbus, or the Orca IO SmartHub.

Configuration Page Buttons

The 32 possible motions can be paged through using the left and right arrow buttons at the top right of the page. The current range of motion IDs being configured is shown between the page buttons.

Motion Configuration Boxes

The motion configuration boxes below the main graph represent the configuration of four motions at a time. The motion ID can be seen at the top of each box.

Save Configuration Button

Saves the global settings and motion ID configurations to permanent memory.

Enable Button

This button toggles the motor between Kinematic Mode (5) and Sleep Mode (1).

State Indicator

The state indicator will read CONTROLLER IDLE if the kinematic controller is not currently executing a motion. It will read RUNNING if a motion is in progress, with a number to the right indicating which motion ID is running.

Configuring a Motion

1. Use the page buttons to navigate to the page with the desired motion ID.
2. In the configuration box with the desired motion ID, enter the motion parameters (position, time, type, delay, next ID, and auto-start). Press the enter key after inputting a value in one of the text boxes.
3. Enable the kinematic controller to test the configured motion settings.

Press the Save Configuration button to save changes to permanent memory.

Trigger Buttons

A trigger button is available for each motion ID which will start that motion and any sequential motions (Must be in Kinematic Mode (5) for trigger to have effect).

Application Note:

Continuous Looping Sequence

It is possible to set up a single motion profile that will start on motor boot up and loop indefinitely without any external control, i.e., motor with power supply only, data cable not used.

This can be done by setting a sequence of motions that all 'Auto-start Next' and having the last motion's 'Next ID' be the first in the sequence. The 'Home Motion ID' should be set to the first motion in the sequence. When Kinematic Mode (5) is entered, each motion will be performed sequentially as defined by the 'Next ID' parameter. Delays between motions can be added as required.

Figure 13: Kinematic Looping Sequence Configuration

Force / Haptics Page

The haptics page provides an interface for the haptics features of the Orca Series motor.

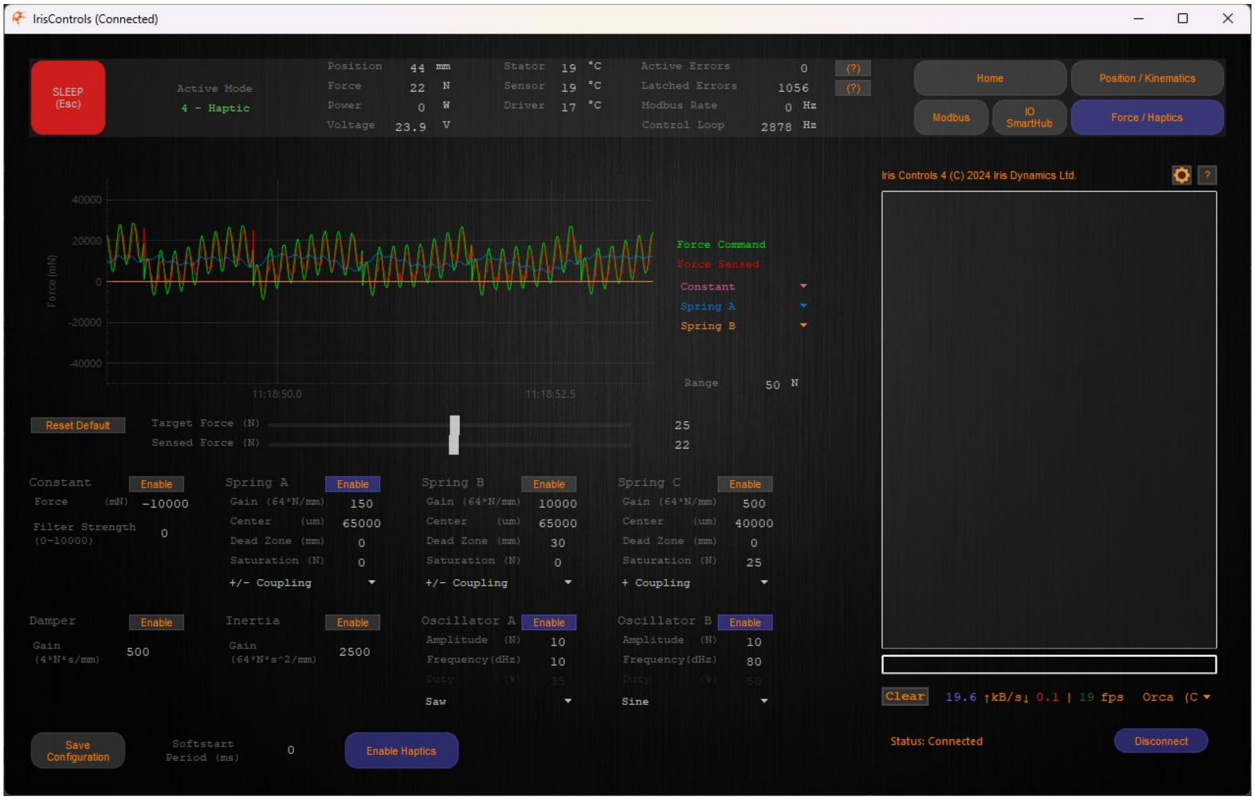


Figure 14: IrisControls Force / Haptics Page

Force Graph

This plot displays the force sensed by the motor and the force commanded by all haptic effects. The plot can also show the individual forces commanded for up to three haptic effects. The force range on the plot is configurable using the Range input.

Enable Haptics Button

Toggling this button will switch the motor into Haptics Mode (4) and will begin applying all enabled haptics effects. Toggling it again will return the motor to Sleep Mode (1).

Save Configuration Button

Clicking this button will save the haptic effects currently displayed on the GUI to the long-term memory on the motor.

Target and Sensed Force Sliders

These sliders cannot be interacted with, they display the current target (or commanded) force along with the sensed force.

Individual Haptic Effect Panels

Each of the other panels on this page provide an interface for adjusting the various haptics effects available to the motor. Each of the data fields in these panels can be modified to adjust the effect. Clicking the enable button contained in a panel will enable that individual haptic effect. The save configuration button will save any changes made here to permanent memory. As many effects as desired can be layered on each other.
The following is a list of the effects with a brief description. For more detailed information on each effect and their configurations, see the Haptic Controller section.

Constant

Constant force in either the positive or negative direction, regardless of shaft position. This input can be low pass filtered with an adjustable strength to accommodate updates from slower communication streams.

Spring

Force that linearly increases with distance from the configured center position. Adjustable in strength, saturation force, dead-zone, and coupling direction.

Damper

Force that linearly increases with speed to oppose the current direction of travel.

Inertia

Force that linearly increases with acceleration to maintain motion of the motor in the direction it is moving.

Oscillator

Force waveform generator with adjustable frequency, amplitude, and wave type.

Application Notes:

50 N force for 1 second followed by 9 seconds of rest.

To accomplish this type of setup the ‘Oscillator’ and ‘Constant’ effects are used in combination. Using a 25 N amplitude for the oscillator will give a 50 N spread between the maximum and minimum values. Using a ‘Constant’ effect with 25 N will add an offset to the oscillator. Instead of alternating between +25 N and -25 N, the force will alternate between 0 N and 50 N. Setting the Frequency to 1 dHz will have the waveform repeat every 10 seconds (which is the total time ‘high’ and ‘low’). Setting the Duty to 10 will give 1 second high and 9 seconds low.

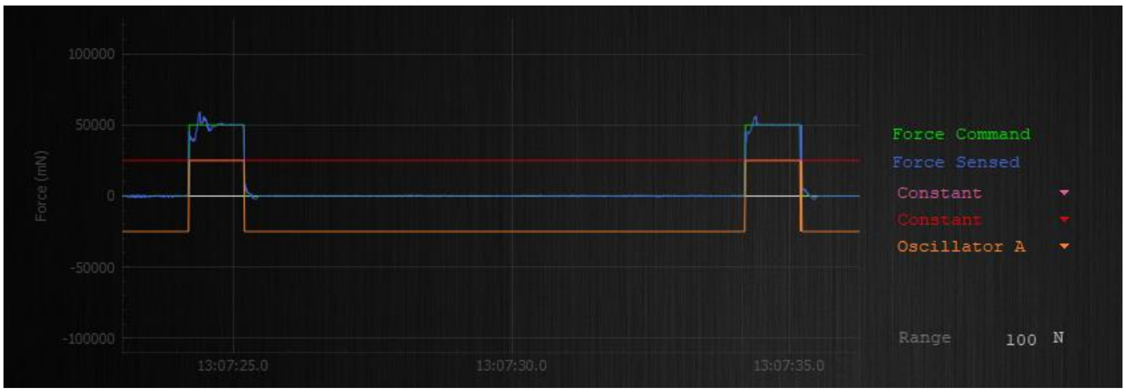


Figure 15: Force Graph when 25 N Constant Force and 25 N Amplitude Pulse Oscillator with 10% Duty Cycle

Testing of High Force

Set up a ‘Spring’ effect with a large dead zone and high Gain to create virtual hard stops. Then use a ‘Constant’ effect to the desired force. This will allow for the demonstration of high forces within a range of the shaft without pushing against hard stops.

Modbus Page

Orca Series motors support the Modbus RTU serial communication protocol as a control method. This GUI page allows the user to view the status of the Modbus communication.

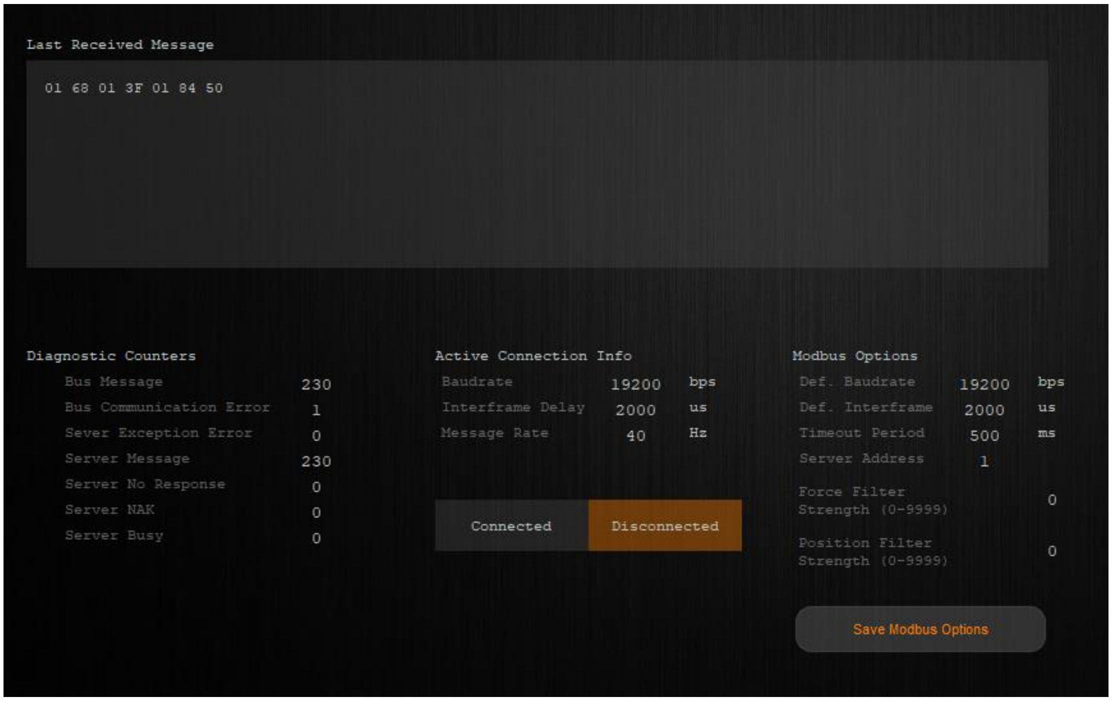


Figure 16: IrisControls Modbus Page

Last Received Modbus Message

All bytes of the last received Modbus message are displayed on this page. If the message was successful, bytes will be displayed in white, if there is an error with the message; incorrect length, CRC byte missing/wrong, unsupported function code, wrong address etc, the bytes text will be red.

Diagnostic Counters

Modbus diagnostic counters are shown below the last received message, on the left-hand side of the page. See Modbus RTU specification for more information on each counter.

Active Connection Info

Connection information is shown below the last received message, in the center of the page. Information shown includes current baudrate, rate of successful messages and connection status. If a high speed negotiation has taken place successfully (i.e., baudrate and interface delay have been negotiated other than the default using function code 65 - manage high speed stream) the Connected label will be highlighted orange. If communication is taking place without connection negotiation, Disconnected will be highlighted.

Modbus Options

Default Modbus behaviour can be configured here to match any Modbus client setup or to increase messages framerate without requiring negotiation.
Default baudrate up to 1250000 bps can be configured. Interframe delay can be decreased as low as 0. Timeout period which will dictate when a message timeout error is triggered can also be configured here.
Force and position input filters are available to smooth out force or position target commands if Modbus communication rate is slow. A value of 0 will result in no filtering whereas a value of 9999 will provide maximum filtering. Note that due to the nature of the filter values between 9000 and 9999 are most likely to be useful.

Save Modbus Options Button

Pressing the will put new configurations into effect immediately and will be saved to permanent memory.

Orca IO SmartHub Page

This panel contains the status and configuration options for the Orca IO SmartHub. To find more information on how to configure and use the Orca IO SmartHub,, please see the Orca IO SmartHub User Manual available at irisdynamics.com/downloads.

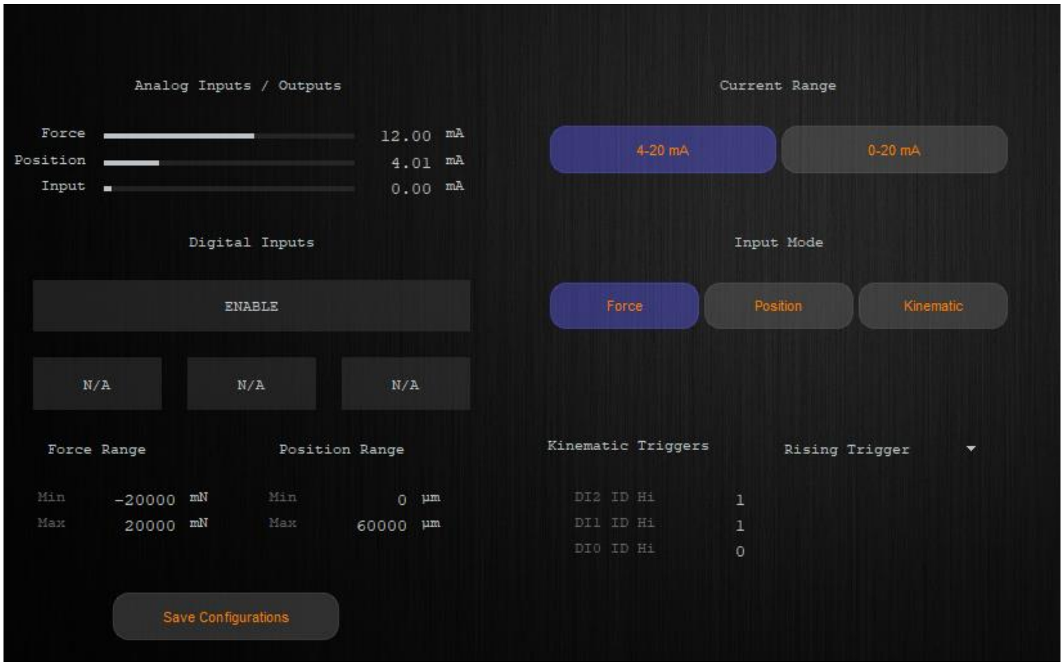


Figure 17: IrisControls Orca IO SmartHub Page

IrisControls Logging

While connected to IrisControls, Orca Series motors provide periodic logging of various sensor data. Logging will start automatically upon connection with IrisControls. Log data will be saved to a file named "Orca_[serial number]_data_log.txt" in the the logs folder located by default in the IrisControls folder in the Users directory. The save location can be changed in the IrisControls Settings menu which is accessed via the gear icon.

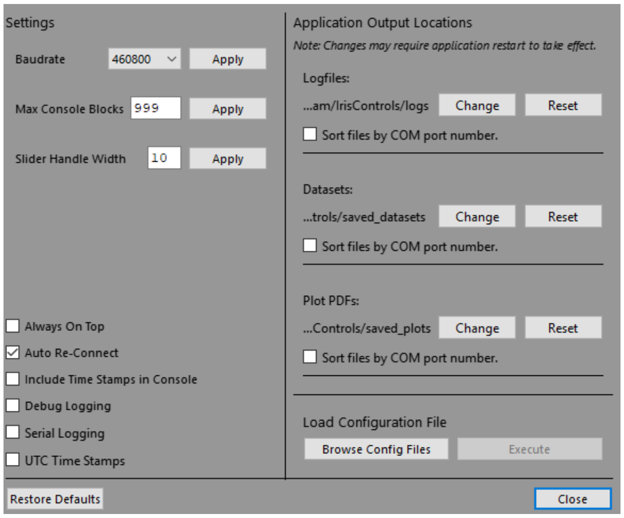


Figure 18: Changing Log File Location Through Settings

The frequency of data logging can be changed by typing "log [time ms]" into the IrisControls console.

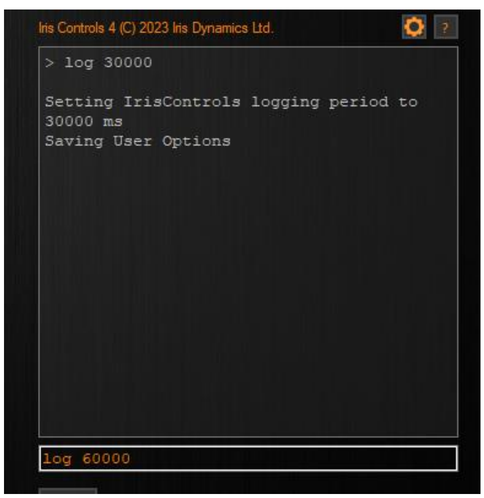


Figure 19: Changing Logging Period Through Console

Orca Registers

System data like sensor readings, user settings, and controller setpoints are stored in and can be accessed from a collection of registers. Registers can be read and, in some cases, modified by the Modbus or GUI interface.

For the list and description of available registers on an Orca Series motor, see Orca Memory Map.

For information on accessing registers from the Modbus interface, see the Orca Series Modbus User Guide.

Control Registers

Orca registers labeled CTRL_REG_n are registers that are used to initiate various processes in the Orca. Some control registers use arrays of bits to have multiple functions available in each register. Below are tables describing the bit positions and functions for each control register.

After a control register has parsed an input and performed the function, the read value of the register will return to 0.

CTRL_REG_0

Control register 0 is used for basic system functions.

Bit Position	Write Value	Name	Description
0	1	reset	Full reset of the Orca.
1	2	clear errors	Clear all active and latched errors.
2	4	zero position	Rest zero position to current shaft position.
3	8	invert position	Change the direction of positive movement.

CTRL_REG_1

Control register 1 is reserved.

CTRL_REG_2

Control register 2 is used for functions that save register data to nonvolatile memory.

Bit Position	Write Value	Name	Description
0	1	save parameters	Save the parameter section of registers to flash memory. (400 - 419)
1 - 4	-	RESERVED	
5	32	tuning save	Save the tuning section of registers to flash memory. (128 - 153)
6	64	user opt save	Save the user options section of registers to flash memory. (160 - 178)
7	128	motion config save	Save the kinematic section of registers to flash memory. (778 - 973)
8	256	current loop save	Save the Orca IO SmartHub section of registers to flash memory. (760 - 776)
9	512	haptic config save	Save the haptic section of registers to flash memory. (640 - 674)

CTRL_REG_3

Control register 3 is used to change the mode of operation of the Orca.

Bit Position	Name	Description
0-15	mode	The desired mode number can be written to this register. If the number is a valid mode, the Orca will change modes. Each mode is detailed in the [Modes of Operation] section.

CTRL_REG_4

Control register 4 is used to set defaults for sections of registers that are saved to permanent memory. Note that the sections must still be saved through control register 2 to keep these values through power cycles.

Bit Position	Write Value	Name	Description
0	1	parameter defaults	Set the parameter section of registers to their default values. (400 - 419)
1	2	tuning defaults	Set the tuning section of registers to their default values. (128 - 153)
2	4	motor user options defaults	Set the motor settings in the user options section of registers to their default values. (163, 166, 167, 171,172, 173) [1]
3	8	Modbus user options defaults	Set the Modbus settings in the user options section of registers to their default values. (164, 165, 168, 169) [1]
4	16	kinematic defaults	Set the kinematic section of registers to their default values. (780 - 972)
5	32	haptic defaults	Set the haptic section of registers to their default values. (641 - 673)
6	64	IOSH defaults	Set the Orca IO SmartHub section of registers to their default values. (755 - 776)

Notes:

1. The user options section of registers share motor settings and Modbus settings in the same block of flash memory. Therefore they are saved together with 1 bit in CTRL_REG_2, but can have their defaults restored separately.

Sensors

Several solid-state, contactless sensors are embedded in each Orca Series motor.

Sensor	Register	Units	Required Voltage
Force	FORCE (double wide)	Millinewtons	>10 V
Position	SHAFT_POS_UM (double wide)	Micrometres	Any
Power	POWER	Watts	>10 V
Driver Temperature	DRIVER_TEMP	Degrees Celsius	>10 V
Stator Temperature	STATOR_TEMP	Degrees Celsius	Any
Coil Temperature	COIL_TEMP	Degrees Celsius	>10 V

Force Sensor

Forces resulting from current in the motor windings, whether caused by the motor drivers, or by currents induced due to shaft movement, is calculated, and reported with low latency and high bandwidth.

Position Sensor

The position of the shaft is measured and reported with low latency and high bandwidth. The reported position can be set to zero by writing to the “Zero Position” bit of “Control Register 0.”
The Orca Series motor will take its current shaft position as zero at startup every time the motor is power cycled as the absolute position will be lost if the motor loses 5V.
Depending on the configuration, the motor will either continuously zero the position value when moving in the negative direction or negative values can be valid. The default direction of positive travel is the shaft moving away from the cable side. The positive position direction can also be inverted by writing to the invert position bit flag in Control Register 0.
The shaft position is obtained by measuring and integrating the ‘shaft alignment,’ or the position of the shaft’s magnets with respect to the stator.

The shaft alignment is absolute in nature (persists throughout power cycles); however, it repeats every shaft period interval. This interval is listed on the motor's datasheet as "Shaft Magnetic Period."

The position sensor is active in all modes of operation.

Power Sensor

Power being consumed or generated is measured and reported with low latency and high bandwidth. This power is a measure of the heat being generated by the stators and is not necessarily an accurate reflection of the power drawn from the supply, especially when the shaft is moving quickly. The power sensor reflects the rate at which the stator will be increasing in temperature.

Temperature Sensors

The temperature of the power driver and the stator windings are measured and reported to the Orca registers.

Power driver temperature can be obtained by reading the DRIVER_TEMP register. The stator winding temperature can be obtained by reading the STATOR_TEMP register.

Stator temperature readings are active in all modes of operation, but driver temperature readings are inaccurate when less than 10 V is supplied to the V_{dd}.

The coil temperature can be obtained by reading the COIL_TEMP register. This value represents the temperature of the windings and is a calculated estimate based on thermocouple reading and power draw.

Controllers

The integrated logic included in Orca Series motors carries out several feedforward and feedback controllers with very low latency loops that provide high performance motion and force control without a lot of tuning and setup required.

Force Controller

Internal control loops control the phase currents in the motor to achieve a commanded force, compensating for shaft position, speed, winding temperature, and supply voltage. There are no user tuning parameters for the force controller, and calibration is typically done during manufacturing.

The force controller acts on commands from the position controller when the motor is in Position Mode (3) or Kinematic Mode (5), acts on commands from Modbus when in Force Mode (2), and commands from the haptic controller when in Haptic Mode (4).

Position Controller

A common use of linear motors, especially in kinematic robotic applications, is to move to and hold position in the presence of dynamic disturbances. The integrated PID position controller accomplishes this by using the position sensor as feedback and generating a force command which will realize the position setpoint effectively. Position setpoint means the position target commanded via Modbus or from the kinematic controller.

When tuning the PID position controller in a new system, it is always recommended to start with as low a force saturation level as possible.

General Tuning Guide

The easiest way to tune the position controller is using the Orca GUI. The "Position" tab allows easy editing of the gains, enabling and disabling of the controller, and a real-time view of the target position, the actual position, and the controller force output.

There is no shortage of PID tuning guides available on the internet and tuning of the integrated PID position controller follows most traditional PID controllers. The reader is encouraged to research the term "PID position controller" if the concepts are unfamiliar prior to using this feature.

Specific optimal tuning will depend on the nature of the load and any disturbances experienced. Default tuning values are based on standard shaft lengths and default kinematic motion profile. Different speed profiles and load may affect tuning requirements.

Force Saturation Level

Units: millinewtons (mN)

This setting limits the maximum force the position controller will generate in either direction.

Proportional Action

Units: millinewtons-per-micrometer (mN / μm)

The 'P' in PID is "proportional gain" and will act like a spring; the further the shaft is from its setpoint, the more force the controller will apply. Higher P-gain results in a stronger spring effect.
Orca Series motors have fast feedback loops which enable relatively high proportional gains. In systems where some steady-state error may be acceptable, it is possible to achieve good motion control using only the proportional action.
In precise systems where small errors are unacceptable, Integral gain should be added after Proportional gain to remove steady state errors.
The proportional gain register value is scaled up by a factor of 64 for increased resolution.

Integral Action

Units: millinewton-seconds-per-micrometer (mN·s / μm)

Integral action takes any small errors in position and accumulates action to correct them over time. For example, if an error of 1 millimeter is present for 1 second, an I-gain of 1000 will generate 1 Newton of corrective force. A second later this force will be 2 N. This will continue until enough force is accumulated to remove the 1 mm of error.
Integral gain is inherently unstable and must be used in combination with proportional gain to prevent oscillation.
Higher I-gain will remove steady state errors faster but could cause instability, especially when the position setpoint is moved rapidly, or sharp disturbances (*e.g.*, kicking the shaft) are encountered.
The integral gain register value is scaled up by a factor of 64 for increased resolution.

Derivative Action (Velocity)

Units: millinewton-millimeters-per-second (mN·mm / s)

In this context, derivative action is a force acting against the speed of the shaft. The action feels like a damper on the shaft and can prevent it from reaching high speeds.

This action is often not required, especially when fast transient response is desired. However, D-gain will prevent the shaft from moving too fast and can improve stability and reduce overshoot.
It is safe to use derivative action alone if a brake force is required. The derivative gain register value is scaled up by a factor of 2 for increased resolution.

Derivative Action (Error)

Units: millinewton-millimeters-per-second (mN·mm / s)

As an alternative to derivative action based on velocity, a derivative action based on error is also available. The end effect of both derivative actions is quite similar and can be used in similar situations.
The derivative action acts proportional to the change in error between the setpoint and measured position.
The derivative gain register value is scaled up by a factor of 2 for increased resolution.

Position Auto Zeroing (BETA Feature)

The auto zeroing algorithm is intended to bring the shaft to the fully retracted zero position and zero out the position reading at that point. This is accomplished with a state machine that runs when the Orca is commanded to enter Auto Zero Mode (55). This can be configured to happen when the Orca boots up, or initiated manually through CTRL_REG_3.

Configuration

Auto zeroing mode must be enabled through the ZERO_MODE register to use the auto zeroing algorithm. Valid values for this register are shown in the Position Zeroing table of the User Configurations section of this manual.

The auto zeroing process has configuration options that may need to be tweaked for different applications.

Maximum Force

- The absolute maximum force allowed to move the shaft during auto zeroing.
- Units: Newtons
- Configurable through the AUTO_ZERO_FORCE_N register or the position page on the IrisControls GUI.

Exit Mode

- The mode of operation which should be entered when auto zeroing is successfully completed.

- Must be one of the following Orca modes of operation or will default to mode 1 (Sleep).
    - Sleep Mode (1)
    - Force Mode (2)
    - Position Mode (3)
    - Haptic Mode (4)
    - Kinematic Mode (5)
- Configurable through the AUTO_ZERO_EXIT_MODE register or the dropdown on the position page on the IrisControls GUI.

Enable on Startup

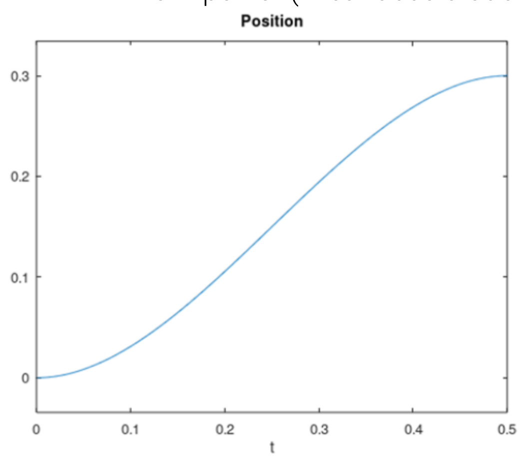
- The auto zeroing algorithm can be configured to run on motor startup through the ZERO_MODE register, where Auto Zero on Boot (3) must be the selected option.

All configuration options can be saved by saving the user options section of flash memory through CTRL_REG_2.

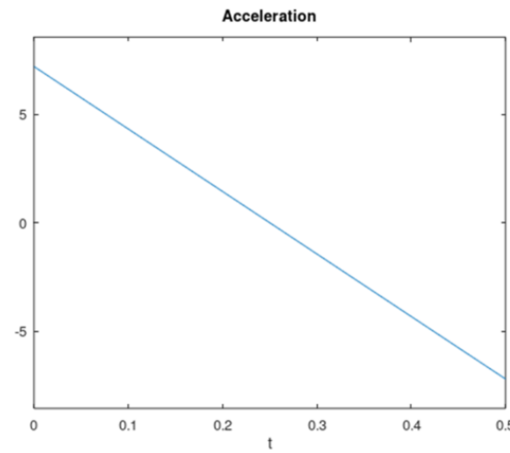
Kinematic Controller

Orca Series motors are equipped with a kinematic controller that provides configurable motion profiles which allow movement to a shaft position over a specified time, while respecting the chosen kinematic constraints. The motion profiles are fed to the position controller as position targets.
Types of kinematic motions available on the motor include:

- Minimum power (linear acceleration).



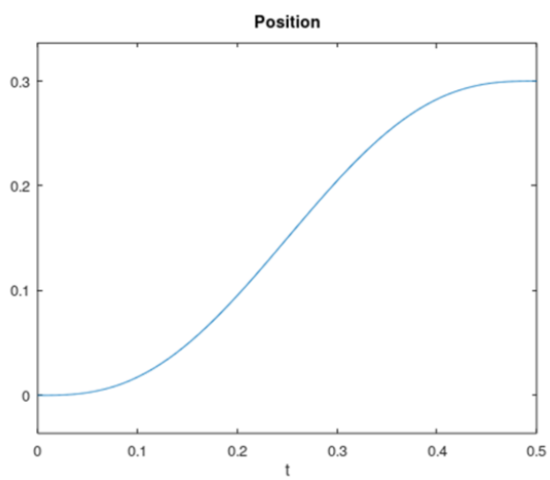
Position vs Time with Linear Acceleration



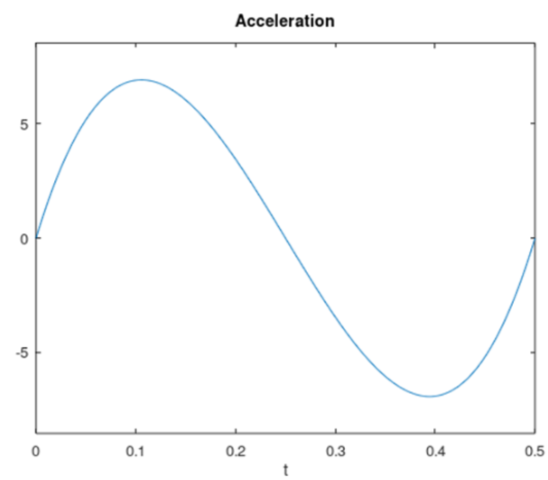
Acceleration vs Time with Linear Acceleration

Figure 20: Position and Acceleration of Type 0 Kinematic Motion

- Maximum smoothness (minimum jerk).



Position vs Time with Minimum Jerk



Acceleration vs Time with Minimum Jerk

Figure 21: Position and Acceleration of Type 1 kinematic motion

Up to 32 motions can be saved to a single Orca Series motor. Motions are configured either from the Kinematic GUI page, or through direct writes to the Orca memory map. Motions can be initiated either by Modbus messages, or from digital triggers through the Orca IO SmartHub.

Software Triggering

Motions can be initiated by writing the desired motion ID number to the KIN_SW_TRIG[MOTION_ID] field. This register will be set to 0x1000 initially and will return to 0x1000 after processing a software trigger request. Software triggers of motions with a number higher than 32 will be ignored.

Register	Bits 15-6	Bits 4-0
KIN_SW_TRIG	Reserved	MOTION_ID

Configuring a Motion

Individual kinematic motions consist of 5 variables:

- Motion target position (μm).
- Motion time period (ms).
- Motion type (Minimum power or maximum smoothness).
- Auto-start delay (Delay in ms before triggering next motion if auto-start is enabled).
- Next ID (motion ID that will be queued after this motion finishes).
- Auto-start next (Start next ID motion after this one).

The kinematic GUI page provides an interface for programming each of the motions, or the motions can be written directly to the memory map. A single motion configuration takes up 6 consecutive registers in the memory map, totaling 192 registers allocated for the motions (KIN_MOTION_0 to KIN_MOTION_31). The layout of each set of these registers is shown in table 4.

Offset from KIN_MOTION_#	Description
0	Position Target (Low 16 bits)
1	Position Target (High 16 bits)
2	Settling Time (Low 16 bits)
3	Settling Time (High 16 bits)
4	Auto-start Delay
5	Next ID, Type, and Auto-Start Next

The motion type and auto-start next options occupy the same register with the following structure.

Register	Bits 15-8	Bits 7-3	Bit 2-1	Bit 0
KIN_MOTION_# + 5	Reserved	Next ID	Type 00 = min. power	Auto-Start Next

			01 = min. jerk	

The Type field is interpreted as a 2-bit number indicating motion type.
The Auto-Start Next bit is 1 when the chain feature is enabled and 0 when it is disabled.

Kinematic Status

The KINEMATIC_STATUS registers is used to indicate the state of the kinematic controller to indicate the active motion ID and whether a motion is in progress or finished,

Register	Bit 15	Bits 14-0
KINEMATIC_STATUS	Running Flag	Active ID

Haptic Controller

Orca Series motors are equipped with a haptic controller that provides a series of force effects which can be combined to create force targets based on shaft position, speed, and acceleration, in addition to periodic force options. The sum of the haptic effects is sent as the force target to the force controller.

Effects

The status of each effect is contained in the HAPTIC_STATUS register. Each bit position represents whether a single effect is enabled or disabled. A 1 represents an active effect that will produce force, and a 0 represents a disabled effect. Effects may be configured as well as toggled on and off while the motor is in Haptic Mode for dynamic effects. Note that the motor must be in Haptic Mode for any of the effects to produce force.

Register	Bits 15-8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
HAPTIC_STATUS	-	Osc B	Osc A	Inertia	Damper	Spring C	Spring B	Spring A	Constant

Below is a description of each type of effect and their configurations.

- Constant: This effect applies a constant force to the haptic controller. This is also a useful effect to stream updates through Modbus commands to implement unique, dynamic effects.
  - Force (mN): Constant force.
  - Filter Strength (0-10000): Strength of low pass filter applied to the constant force input.
- Spring: The spring effects apply a force that linearly increases as the shaft is moved away from the center location.
  - Gain (N/mm): Rate at which the force will increase proportional to the change in position.
  - Center (μm): Location of spring center.
  - Dead Zone (mm): Indicates a distance around either side of the center in which the spring effect doesn't apply.
  - Saturation (N): The maximum force that will be output by this spring effect, if set to 0 no force limit will exist.
  - Coupling (+/-):Type of spring behaviour.
    - +/-: Spring will act in both directions.
    - +: Spring forces only applied when moving the shaft in the positive direction beyond the center.
    - -: Spring forces only applied when moving the shaft in the negative direction beyond the center.

- Damper: The damper effect applies a force to reduce the speed of the motor's movement.
    - Gain (N·s/mm): Rate at which the force increases proportional to the speed of the motor.
- Inertia: The inertia effect applies a force that attempts to reduce the acceleration of the motor. This makes it both more difficult to get the shaft in motion, but also keep the shaft in motion once moved.
    - Gain (N·s^2/mm): Rate at which the force increases proportional to the acceleration of the motor.

- Oscillator: The oscillator effects apply periodic forces in the shape of various waveforms.
    - Amplitude (N): The peak force of the waveform, oscillators are centered on 0 N of force so the force will oscillate between negative amplitude and positive amplitude values. Note: adding a constant force effect with an oscillator will change the force around which the waveform oscillates.
    - Frequency (dHz): Frequency of the oscillation in decihertz. (1 dHz = 0.1 Hz)
    - Duty (%): This field only applies to 'Pulse' waveforms, the duty indicates what percentage of the waveform should be spent in a 'high' state (applying positive force) with the remaining percent being in the 'low' state.
    - Type: The waveform pattern can be selected from the dropdown.
        - Pulse: Square wave that alternates between the positive amplitude value and the negative amplitude value. The duty cycle is used to determine the portion of the period each direction of force is active.
        - Sine: Sinusoidal waveform.
        - Triangle: Linearly increases force, then reverses and linearly decreases force.
        - Sawtooth: Linearly increases from negative amplitude value to positive amplitude value over entire period, then sharply go back to negative amplitude value

Configuring Effects

Effects are individually configured through sets of registers shown in a table below. Note: the units for each effect configuration, as well as the scaling factors applied to each gain.

Effect	Parameter	Register
Constant Force	Force (mN)	CONTANT_FORCE_MN
		CONSTANT_FORCE_MN_H
	Filter Strength (0-10000)	CONST_FORCE_FILTER
Spring n	Gain (64·N/mm)	Sn_GAIN_N_MM
	Center (mm)	Sn_CENTER_UM
		Sn_CENTER_UM_H
	Coupling	Sn_COUPLING
	Dead Zone (mm)	Sn_DEAD_ZONE_MM
	Saturation (N)	Sn_FORCE_SAT_N
Damper	Gain (4·N/mm)	D0_GAIN_NS_MM
Inertia	Gain (64·N·s²/mm)	I0_GAIN_NS2_MM
Oscillator n	Amplitude (N)	On_GAIN_N
	Frequency (dHz)	On_FREQ_DHZ
	Duty (x / 65535)	On_DUTY
	Type	On_TYPE

User Configurations

Some properties of the motor can be configured to suit the application. In all cases, this configuration is done by writing to various registers and, when satisfied, saving those settings to permanent memory by writing the CONTROL_REG_2 [usr_opt_save_flag].

Motor Performance Limits

Parameter	User Register	Default Register (read only)	Units
Maximum Temperature	USER_MAX_TEMP	MAX_TEMP	Degrees Celsius
Maximum Force	USER_MAX_FORCE	-	Millinewtons
Maximum Power	USER_MAX_POWER	MAX_POWER	Watts
Communication Timeout	USER_COMMS_TIMEOUT	COMMS_TIMEOUT	Milliseconds

Maximum Temperature

The motor will have a maximum temperature at which it will shut off. A lower temperature can be set to cause the motor to shut down at a lower temperature. The temperature shutoff threshold cannot be disabled or raised beyond the default maximum temperature.

Maximum Force

The motor has no default maximum force threshold, but one can be set which will limit forces beyond this level. Setting the register to zero disables this threshold.

Maximum Power

If the power burned in the motor exceeds this or the device's default threshold, the drivers will be disabled, preventing power draw. Setting this register to zero or higher than the default setting will result in the motor only powering down when unsafe levels are reached. See the Power Exceeded section for more information.

Communication Timeout

By default, Force Mode (2), Position Mode (3), and Haptic Mode (4) will timeout when serial communications fail to send a successful message within a timeout period. This timeout period can be adjusted with this user setting.

Input and Output Filters

Input and output filters are available for both force and position values. These are infinite impulse response (IIR) filters. In this case 0 represents no filter and 9999 is maximum filtering.

User Register	Description	Valid Range
FORCE_FILT	Filter on motor's sensed force.	0-9999
POS_FILT	Filter on motor's sensed position.	0-9999
MB_FORCE_FILT	Filter on target force values received over Modbus.	0-9999
MB_POS_FILT	Filter on target position values received over Modbus.	0-9999

Modbus Communications

The configurations that govern Modbus communication can be adjusted. This can be used to match a certain Modbus client configuration or to speed up message frame rate.

Parameter	User Register	Default Value	Valid Range
Baudrate	USR_MB_BAUD_LO USR_MB_BAUD_HI	19200	9600 - 1250000
Interframe Delay	USR_MB_DELAY	2000 µs	0 - 10000
Server Address	USR_MB_ADDR	1	1 - 65535

Position Zeroing

User Register	Description	Valid Range
ZERO_MODE	Type of zeroing.	Negative Zeroing (0) Manual Zeroing (1) Auto Zero Enabled (2) Auto Zero on Boot (3)
AUTO_ZERO_FORCE_N	The maximum force the auto-zeroing algorithm will use to move the motor. In Newtons.	0-800
AUTO_ZERO_EXIT_MODE	Mode to enter after completing auto-zeroing.	Sleep (1) Force (2) Position (3) Haptic (4)

		Kinematic (5)

Errors

The motor will generate error codes when a user setting, or a device limit is reached or exceeded. Depending on the error, certain features will not be available until the error is cleared.

Some errors, like temperature and power, which trigger when a parameter exceeds a device maximum can be configured using motor registers to trigger earlier as the application requires.

Other errors, like force, will never trigger unless the registers are configured.

Active and Latched Error Registers

Errors which are currently in effect are reflected in the ERROR_0 register. Any time an error is asserted, its flag will also appear in the ERROR_1 register and will remain asserted until cleared by a write to the CONTROL_REG_0 [clear_errors_flag]. For example, if a temperature error becomes active, it will show up in both registers. Once the temperature falls below the error level and the motor is brought to Sleep Mode (1), the error will be cleared from ERROR_0 but not from ERROR_1.

Error	Mask	Trigger Level Registers	Modules disabled	Cleared By
Configuration Errors	1 (0x001F)	-	Position, Force	Calibration Routines
Force Clipping	32 (0x0020)	-	-	Automatically
Temperature Exceeded	64 (0x0040)	USER_MAX_TEMP MAX_TEMP	Position, Force, Calibration	Sleep Mode
Force Exceeded	128 (0x0080)	USER_MAX_FORCE	-	Automatically
Power Exceeded	256 (0x0100)	USER_MAX_POWER MAX_POWER	Position, Force	Sleep Mode
Shaft Image Failed	512 (0x0200)	-	Position, Force	Sleep Mode + Insert or Calibrate Shaft
Voltage Invalid	1024 (0x0400)	MIN_VOLTAGE MAX_VOLTAGE	Position, Force, Calibration	Providing a valid voltage source
Comms Timeout	2048 (0x0800)	USER_COMMS_TIMEOUT COMMS_TIMEOUT	Position, Force	Sleep Mode

Configuration Errors

These errors indicate calibrations or settings have not been done or have been made invalid. These errors can be cleared by running the appropriate calibration routine, followed by committing the valid results to permanent memory.

Force Clipping

When the force controller is asked to play forces that would require a phase to exceed its rated current, the Force Clipping error is asserted. This error has no effect on operation except to inform the user that linear force output has been compromised. The error self clears as soon as the condition is removed.

Temperature Exceeded

When the temperature of the stator windings or of the motor driver exceeds the device or user-set maximum, a Temperature Exceeded error is asserted. This error is reset when both temperatures fall below both the maximum and user-set temperatures, and the motor is commanded into Sleep Mode (1).

Force Exceeded

When the measured force output of the motor exceeds the user-set force limit, a Force Exceeded error is asserted. This error is cleared when the force falls below the measured threshold.

The force controller will attempt to control the force output to minimize force output beyond the user setting.

Power Exceeded

When the power burned in the stator exceeds the device or user-set maximum value, a Power Exceeded error is asserted. This error is cleared by commanding the motor into Sleep Mode (1).

If this error is experienced, either the maximum power user setting can be increased, or the maximum force user setting should be decreased.

If the position controller (*i.e., Position* Mode (3)) is causing this error, the saturation level can also be decreased to prevent this error.

Shaft Image Failed

The shaft image is a calculation of the shaft's alignment with the windings based on magnetic field measurements. If this image is detected to be invalid, the shaft might not be inserted, it might be an invalid shaft for the device, or the device may require calibration. This error is cleared by resolving the underlying issue and commanding the motor into Sleep Mode (1).

Voltage Invalid

When the supply voltage of the motor is measured to be less than 10V, a Voltage Invalid error is asserted. This error will disable the current sensors, resulting in force and power calculations being disabled as well. The driver temperature sensor is also invalid in this configuration, so it is disabled. This error is cleared by providing a supply voltage of greater than 10V.

Communications Timeout

When in Force Mode (2), Position Mode (3), or Haptic Mode (4), a steady stream of communications must be successfully received to avoid this error. If the timeout specified in either the COMMS_TIMEOUT or USER_COMMS_TIMEOUT register elapsed between successful messages, theComms Timeout error is asserted.

Users can adjust the communications timeout by writing a non-zero value to the USER_COMMS_TIMEOUT register. This register has units of milliseconds (ms). This can also be configured on the Modbus Page of the GUI.

APPENDIX : Orca Memory Map

All registers can be assumed to be unsigned 16 bit integers unless otherwise specified.

Address	Name	Width	Description
0	CTRL_REG_0	1	Basic system functions.
1	CTRL_REG_1	1	Reserved.
2	CTRL_REG_2	1	Saving registers to permanent memory.
3	CTRL_REG_3	1	Set mode of operation.
4	CTRL_REG_4	1	Return to default.
5 - 7	Reserved		
8	GUI_PERIOD_CMD	1	Commanded period between IrisControls GUI frames in milliseconds.
9	KIN_SW_TRIGGER	1	Software trigger for initiating kinematic movements.
10 - 27	Reserved		
28	FORCE_CMD	1	Commanded actuator output force in millinewtons. Lower 2 bytes. Upper and lower bytes combine create signed 32 bit integer.
29	FORCE_CMD_H	1	Commanded actuator output force in millinewtons. Upper 2 bytes.

30	POS_CMD	1	Commanded actuator position in micrometers. Lower 2 bytes. Upper and lower bytes combine create signed 32 bit integer.
31	POS_CMD_H	1	Commanded actuator position in micrometers. Upper 2 bytes.
32 - 132	Reserved		
133	PC_PGAIN	1	Position controller proportional gain.
134	PC_IGAIN	1	Position controller integral gain.
135	PC_DVGAIN	1	Position controller velocity gain.
136	PC_DEGAIN	1	Position controller error derivative gain.
137	PC_FSATU	1	Position controller maximum force output. Lower 2 bytes.
138	PC_FSATU_H	1	Position controller maximum force output. Upper 2 bytes.
139	USER_MAX_TEMP	1	User configurable maximum motor temperature before over temperature error in degrees Celsius.
140	USER_MAX_FORCE	1	User configurable maximum force output in millinewtons. Lower 2 bytes. Upper and lower bytes combine create signed 32 bit integer.

141	USER_MAX_FORCE_H	1	User configurable maximum force output in millinewtons. Upper 2 bytes.
142	USER_MAX_POWER	1	User configurable maximum power burn in watts.
143	SAFETY_DGAIN	1	Speed damping gain value used when communications are interrupted.
144 - 149	Reserved		
150	PC_SOFTSTART_PERIOD	1	Time in ms over which the position controller max force output will ramp from zero any time a mode of operation in which the position controller used is entered.
151	FORCE_UNITS	1	Determines whether forces will be interpreted in legacy unitless form or in millinewtons.
152	POS_SIGN	1	Determines which direction of shaft movement is the zero direction.
153 - 161	Reserved		
162	LOG_PERIOD	1	Period between data log entries.
163	USER_COMMS_TIMEOUT	1	Time between successful force or position commands before a communications error occurs. In milliseconds.

164	USR_MB_BAUD_LO	1	Default Modbus baudrate low 16 bits. Leaving this register at 0 will use the system default of 19200 bps. Upper and lower bytes combine create unsigned 32 bit integer.
165	USR_MB_BAUD_HI	1	Default Modbus baudrate high 16 bits. Leaving this register at 0 will use the system default of 19200 bps.
166	FORCE_FILT	1	Force output IIR filter alpha value. Maps 0-9999 to alpha values of 0 to 1. Where 0 is no filter.
167	POS_FILT	1	Position output IIR filter alpha value. Maps 0-9999 to alpha values of 0 to 1. Where 0 is no filter.
168	USR_MB_DELAY	1	Default Modbus inter frame delay in microseconds. Default value is 2000 μs.
169	USR_MB_ADDR	1	Start up Modbus server address. Default value is 1.
170	Reserved		
171	ZERO_MODE	1	Selects the type of zeroing the motor should use.
172	AUTO_ZERO_FORCE_N	1	The maximum force the auto-zeroing algorithm will use to move the motor. In Newtons.

173	AUTO_ZERO_EXIT_MODE	1	Mode to enter after completing auto-zeroing.
174	Reserved		
175	MB_FORCE_FILTER	1	Input filter value for Modbus Force inputs. 0-9999 where 0 is no filtering.
176	MB_POS_FILTER	1	Input filter value for Modbus Position inputs. 0-9999 where 0 is no filtering.
177 - 262	Reserved		
263	UART0_UP_RATE	1	Number of bytes transmitted in the last second by UART0.
264	UART1_UP_RATE	1	Number of bytes transmitted in the last second by UART1.
265	UART0_DOWN_RATE	1	Number of bytes received in the last second by UART0.
266	UART1_DOWN_RATE	1	Number of bytes received in the last second by UART1.
267	GUI_DROPPED_FRAMES	1	Total number of skipped IrisControls GUI transactions.
268	GUI_DROPPED_FPS	1	Number of skipped IrisControls GUI transactions in the last second.
269	LOOP_FREQ	1	Main loop frequency in kilohertz.

270 - 271	Reserved		
272	MOTOR_FRAME_COUNT	1	Number of complete motor frames in the last second.
273	MB_FREQ	1	Number of successful Modbus messages in the last second.
274 - 312	Reserved		
313	GUI_PERIOD	1	Period between IrisControls GUI communications in milliseconds.
314 - 316	Reserved		
317	MODE_OF_OPERATION	1	Active mode the actuator is currently running in.
318	Reserved	1	
319	KINEMATIC_STATUS	1	Indicates the state of the kinematic controller, and which motion is currently being performed.
320 - 335	Reserved		
336	STATOR_TEMP	1	Temperature of the motor stator in degrees Celsius.
337	DRIVER_TEMP	1	Temperature of the motor driver in degrees Celsius.
338	VDD_FINAL	1	Motor supply voltage in volts.

339 - 341	Reserved		
342	SHAFT_POS_UM	1	Shaft absolute position in micrometers. Lower 2 bytes. Upper and lower bytes combine create signed 32 bit integer.
343	SHAFT_POSITION_H	1	Shaft absolute position in micrometers. Upper 2 bytes.
344	SHAFT_SPEED_MMPS	1	Shaft speed in millimeters per second. Lower 2 bytes. Upper and lower bytes combine create signed 32 bit integer.
345	SHAFT_SHEED_H	1	Shaft speed in millimeters per second. Upper 2 bytes.
346	SHAFT_ACCEL_MMPSS	1	Shaft acceleration in millimeters per second per second. Lower 2 bytes. Upper and lower bytes combine create signed 32 bit integer.
347	SHAFT_ACCEL_H	1	Shaft acceleration in millimeters per second per second. Upper 2 bytes.
348	FORCE	1	Sensed actuator output force in millinewtons. Lower 2 bytes. Upper and lower bytes combine create signed 32 bit integer.
349	FORCE_H	1	Sensed actuator output force in millinewtons. Upper 2 bytes.
350	POWER	1	Sensed actuator output power in watts.

351 - 354	Reserved		
355	AVG_POWER	1	Average sensed actuator output power in watts.
356	COIL_TEMP	1	Estimated coil temperature in degrees Celsius.
357 - 400	Reserved		
401	MAX_TEMP	1	Absolute maximum motor temperature before over temperature error.
402	MIN_VOLTAGE	1	Minimum motor voltage in volts before invalid voltage error.
403	MAX_VOLTAGE	1	Maximum motor voltage in volts before invalid voltage error.
404	MAX_CURRENT	1	Maximum motor current output in milliamps.
405	MAX_POWER	1	Maximum motor power burn in watts.
406	SERIAL_NUMBER_LOW	1	Actuator serial number. Lower 2 bytes. Upper and lower bytes combine create unsigned 32 bit integer.
407	SERIAL_NUMBER_HIGH	1	Actuator serial number. Upper 2 bytes.
408	MAJOR_VERSION	1	Firmware major version.

409	RELEASE_STATE	1	Firmware minor version.
410	REVISION_NUMBER	1	Firmware revision number.
411	COMMIT_ID_LO	1	Firmware commit ID lower 2 bytes.
412	COMMIT_ID_HI	1	Firmware commit ID upper 2 bytes.
413	Reserved		
414	HW_VERSION	1	Target hardware version for this firmware.
415 - 416	Reserved		
417	COMMS_TIMEOUT	1	Time between successful force or position commands before a communications error occurs. In milliseconds.
418	STATOR_CONFIG	1	Physical stator configuration type.
419 - 431	Reserved		
432	ERROR_0	1	Currently active error flags. Only reflects error conditions that have not been cleared.
433	ERROR_1	1	Latched error flags. Reflects all errors that have occurred since reset.
434 - 463	Reserved		

464	MB_CNT0	1	Return bus message count. Refer to Modbus specification.
465	MB_CNT1	1	Return bus communication error. Refer to Modbus specification.
466	MB_CNT2	1	Return server exception error count. Refer to Modbus specification.
467	MB_CNT3	1	Return server message count. Refer to Modbus specification.
468	MB_CNT4	1	Return server no response count. Refer to Modbus specification.
469	MB_CNT5	1	Return server NAK count. Refer to Modbus specification.
470	MB_CNT6	1	Return server busy count. Refer to Modbus specification.
471	MB_CNT7	1	Return bus character overrun count. Refer to Modbus specification.
472	MB_CNT8	1	Rx line error.
473	MB_CNT9	1	Ignoring state error.
474	MB_CNT10	1	Unexpected interchar.
475	MB_CNT11	1	Unexpected interframe.
476	MB_CNT12	1	Timeout sequence error.

477	MB_CNT13	1	Unexpected emission.
478	MB_CNT14	1	Unexpected reception.
479 - 481	Reserved		
482	MB_BAUD	1	Current Modbus serial baudrate. Lower 2 bytes.
483	MB_BAUD_H	1	Current Modbus serial baudrate. Upper 2 bytes.
484	MB_IF_DELAY	1	Current Modbus inter frame delay in microseconds.
485	MB_ADDRESS	1	Current Modbus server address.
486 - 495	Reserved		
496	MESSAGE_0_SIZE	1	Size of last received Modbus message in bytes.
497	MESSAGE_0	128	-
626 - 640	Reserved		
641	HAPTIC_STATUS	1	Enabled state of effects.
642	CONSTANT_FORCE_MN	1	Value of constant force effect in millinewtons, low 2 bytes Upper and lower bytes combine create signed 32 bit integer.
643	CONSTANT_FORCE_MN_H	1	Value of constant force effect in millinewtons, high 2 bytes.

644	S0_GAIN_N_MM	1	Strength of spring force.
645	S0_CENTER_UM	1	Location of spring center, low 2 bytes Upper and lower bytes combine create signed 32 bit integer.
646	S0_CENTER_UM_H	1	Location of spring center, high 2 bytes.
647	S0_COUPLING	1	Coupling type, 0 (Both), 1 (Positive), 2 (Negative).
648	S0_DEAD_ZONE_MM	1	Zone from center where no spring effect exists.
649	S0_FORCE_SAT_N	1	Maximum force that the spring can output.
650	S1_GAIN_N_MM	1	Strength of spring force.
651	S1_CENTER_UM	1	Location of spring center, low 2 bytes Upper and lower bytes combine create signed 32 bit integer.
652	S1_CENTER_UM_H	1	Location of spring center, high 2 bytes.
653	S1_COUPLING	1	Coupling type, 0 (Both), 1 (Positive), 2 (Negative).
654	S1_DEAD_ZONE_MM	1	Zone from center where no spring effect exists.
655	S1_FORCE_SAT_N	1	Maximum force that the spring can output.

656	S2_GAIN_N_MM	1	Strength of spring force.
657	S2_CENTER_UM	1	Location of spring center, low 2 bytes.
658	S2_CENTER_UM_H	1	Location of spring center, high 2 bytes.
659	S2_COUPLING	1	Coupling type, 0 (Both), 1 (Positive), 2 (Negative).
660	S2_DEAD_ZONE_MM	1	Zone from center where no spring effect exists.
661	S2_FORCE_SAT_N	1	Maximum force that the spring can output.
662	D0_GAIN_NS_MM	1	Strength of damping force.
663	I0_GAIN_NS2_MM	1	Strength of inertia force.
664	O0_GAIN_N	1	Amplitude of periodic effect.
665	O0_TYPE	1	Type of periodic effect 0 (square), 1 (sine), 2 (triangle), 3 (sawtooth).
666	O0_FREQ_DHZ	1	Period of oscillation.
667	O0_DUTY	1	Pulse width modulation of signal as a % of duty cycle max value.
668	O1_GAIN_N	1	Amplitude of periodic effect.
669	O1_TYPE	1	Type of periodic effect 0 (square), 1 (sine), 2 (triangle), 3 (sawtooth).

670	O1_FREQ_DHZ	1	Frequency of periodic effect.
671	O1_DUTY	1	Pulse width modulation of signal as a % of duty cycle max value.
672	CONST_FORCE_FILTER	1	Amount of filtering on constant force inputs. 0 - 9999 where 0 is no filter.
673	HAPTIC_SOFTSTART	1	Amount of time in milliseconds for force to ramp up upon enabling Haptic Mode.
674 - 755	Reserved		
756	ILOOP_DIN	1	Status of IO SmartHub digital inputs. Each input is represented by a bit.
757	ILOOP_OUT_CH1	1	4-20 mA output channel 1. Force.
758	ILOOP_OUT_CH2	1	4-20 mA output channel 2. Position.
759	ILOOP_IN	1	4-20 mA input.
760	Reserved		
761	ILOOP_CONFIG	1	Configuration for 4-20mA control.
762	ILOOP_FORCE_MIN	1	Force that maps to 4 mA. Low 2 bytes. In millinewtons. Upper and lower bytes combine create signed 32 bit integer.
763	ILOOP_FORCE_MIN_HI	1	Force that maps to 4 mA. High 2 bytes. In millinewtons.

764	ILOOP_FORCE_MAX	1	Force that maps to 20 mA. Low 2 bytes. In millinewtons. Upper and lower bytes combine create signed 32 bit integer.
765	ILOOP_FORCE_MAX_HI	1	Force that maps to 20 mA. High 2 bytes. In millinewtons.
766	ILOOP_POS_MIN	1	Position that maps to 4 mA. Low 2 bytes. In micrometers. Upper and lower bytes combine create signed 32 bit integer.
767	ILOOP_POS_MIN_HI	1	Position that maps to 4 mA. High 2 bytes. In micrometers.
768	ILOOP_POS_MAX	1	Position that maps to 20 mA. Low 2 bytes. In micrometers. Upper and lower bytes combine create signed 32 bit integer.
769	ILOOP_POS_MAX_HI	1	Position that maps to 20 mA. High 2 bytes. In micrometers.
770	ILOOP_KIN_TYPE	1	Type of trigger behaviour.
771	ILOOP_D0_HIGH_ID	1	Kinematic motion id value for rising edge digital 0.
772	ILOOP_D0_LOW_ID	1	Kinematic motion id value for falling edge digital 0.
773	ILOOP_D1_HIGH_ID	1	Kinematic motion id value for rising edge digital 1.
774	ILOOP_D1_LOW_ID	1	Kinematic motion id value for falling edge digital 1.

775	ILOOP_D2_HIGH_ID	1	Kinematic motion id value for rising edge digital 2.
776	ILOOP_D2_LOW_ID	1	Kinematic motion id value for falling edge digital 2.
777 - 779	Reserved		
780 - 785	KIN_MOTION_0	6	
786 - 791	KIN_MOTION_1	6	
792 - 797	KIN_MOTION_2	6	
798 - 803	KIN_MOTION_3	6	
804 - 809	KIN_MOTION_4	6	
810 - 815	KIN_MOTION_5	6	
816 - 821	KIN_MOTION_6	6	
822 - 827	KIN_MOTION_7	6	
828 - 833	KIN_MOTION_8	6	
834 - 839	KIN_MOTION_9	6	
840 - 845	KIN_MOTION_10	6	
846 - 851	KIN_MOTION_11	6	
852 - 857	KIN_MOTION_12	6	

858 - 863	KIN_MOTION_13	6	
864 - 869	KIN_MOTION_14	6	
870 - 875	KIN_MOTION_15	6	
876 - 881	KIN_MOTION_16	6	
882 - 887	KIN_MOTION_17	6	
888 - 893	KIN_MOTION_18	6	
894 - 899	KIN_MOTION_19	6	
900 - 905	KIN_MOTION_20	6	
906 - 911	KIN_MOTION_21	6	
912 - 917	KIN_MOTION_22	6	
918 - 923	KIN_MOTION_23	6	
924 - 929	KIN_MOTION_24	6	
930 - 935	KIN_MOTION_25	6	
936 - 941	KIN_MOTION_26	6	
942 - 947	KIN_MOTION_27	6	
948 - 953	KIN_MOTION_28	6	
954 - 959	KIN_MOTION_29	6	

960 - 965	KIN_MOTION_30	6	
966 - 971	KIN_MOTION_31	6	
972	KIN_HOME_ID	1	ID of kinematic motion triggered when Kinematic Mode enabled or when Home signal asserted from Analog interface.