A PRODUCTION READY SOLUTION IN A FRACTION OF THE TIME

Mako Hydraulics
by davisnicols

# MakoECU Software Users Manual

Version 3.2

Aug 1, 2025

# Software Architecture Overview:

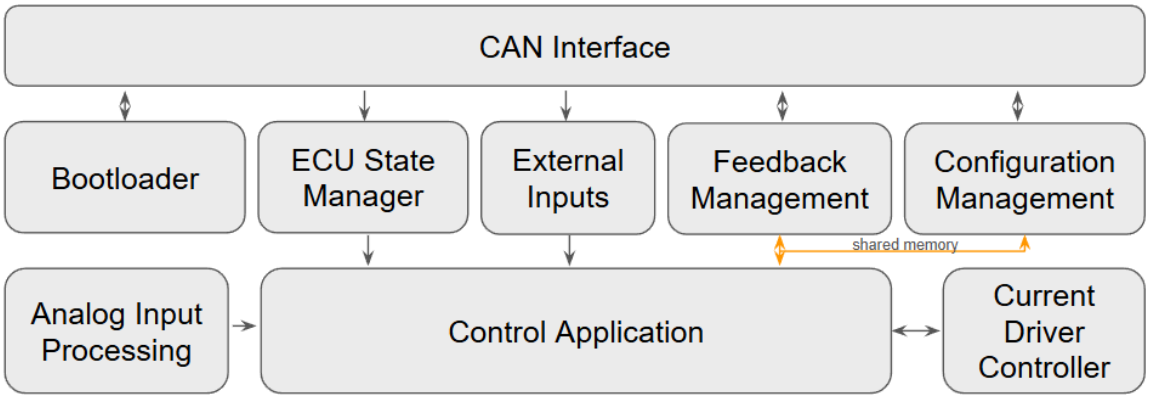


Figure. MakoECU Software Architecture

The MakeECU software architecture consists of 9 functional areas:

1. CAN Interface - The CAN interface is the communication link to the host controller and other CAN network devices. The MakoECU has 1 CAN bus interface and supports 100K, 250k, 500k, and 1M bitrates.

2. Bootloader - The bootloader handles hardware initialization, program loading, and firmware updates received via CAN.

3. ECU State Manager - Manages the ECUs enabled and disabled states

4. External Inputs - Manages the reception of external input signals over CAN and routes them to the Control Application

5. Feedback Management - Manages the transmission of feedback signals onto the CAN bus.

6. Configuration Management - Stores, receives, and sends configuration parameters used by the Control Application. Also performs the function of storing configuration to persistent memory. Configuration can also be restored to the default values.

7. Analog Input Processing - Reads and processes the 4 analog inputs on the MakoECU. The processed voltage value is then passed to the Control Application where further signal processing and diagnostics are applied.

8. Control Application - The Control Application contains the logic for diagnostics, signal processing, functional safety, and the hydraulic control algorithms.

9. Current Driver Controller - Provides closed loop current control to the six valve coil drivers on the MakoECU. Current commands are received by the Control Application and sensed currents are returned for diagnostics.

# Control Application Overview:

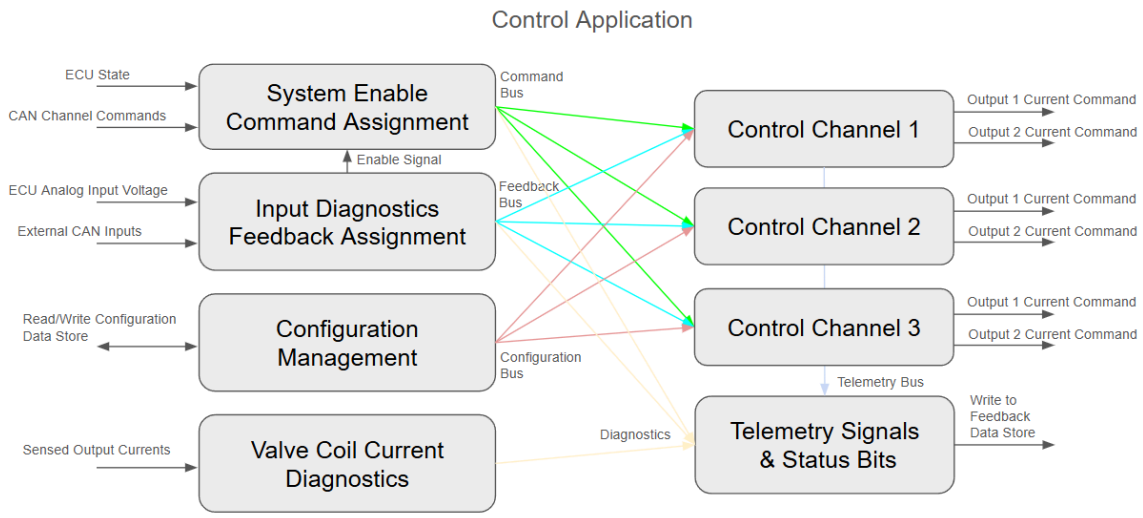


Figure. Control Application Architecture

This section is intended to give a brief overview of the Control Application to help the user understand the overall function of the MakoECU before digging deeper into the functionality of each subsystem. The Control application houses the majority of the control functions and logic that the user will configure, command, calibrate, and monitor. It is broken down into the following functional areas:

1. System Enable and Channel Command Assignment - This subsystem acts as the safety supervisor of the control application. Current commands will not be sent to the valve drivers unless the criteria for enabling a control channel met. The enable criteria is based on ECU State, valid channel command reception, and the enable input.

2. Input Diagnostics and Channel Feedback Assignment - This subsystem receives inputs from the MakoECU analog inputs or external inputs received via CAN, performs diagnostics on them, and assigns as the feedback signal to a control channel depending on the ECU configuration.

3. Configuration Management - This system manages and distributes tunable configuration parameters to the subsystems within the Control Application. It can also write configuration parameters back to memory if a routine which generates a configuration parameter (ex. auto-calibration) is executed.

4. Valve Coil Diagnostics - The subsystem monitors that the current sensed by the current driver is tracking the commanded current and generates a diagnostic flag if not.

5. Control Channels - The control channel subsystem is where the motion control logic is located as well as logic relating to the other channel control modes. More on the control channel and its function later.

6. Telemetry Signals and Status Bits - This subsystem collates all the signals available over telemetry and status bits generated within the control application and writes them to the telemetry data store.

# CAN Network Setup and Configuration:

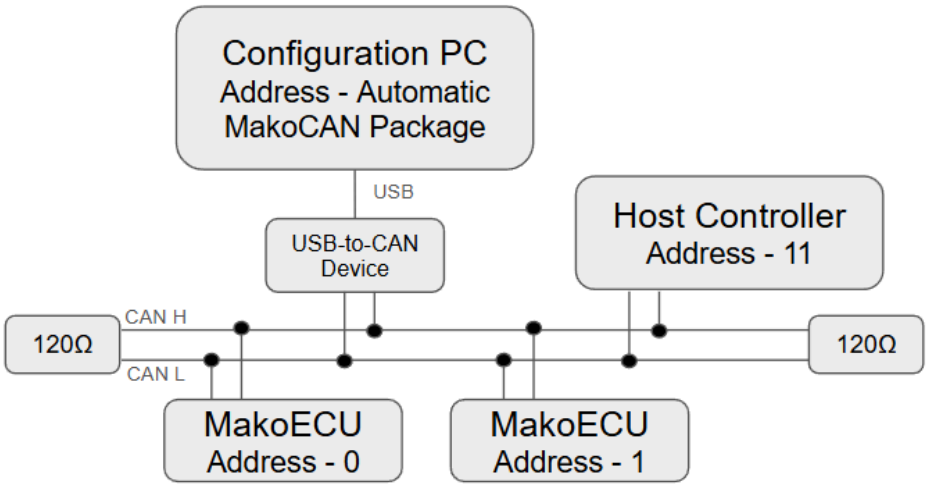


Figure. A Typical MakoECU CAN Network

The MakoECU is designed to operate on a standard CAN 2.0b data bus network. Additional devices can operate on the bus and provide command and configuration information to the MakoECU. Each MakoECU has a unique address which allows the host controller to directly communicate with each ECU as well as determine from which device data originated. Typical devices on the network worthy of a definition are:

1. Host Controller - The host controller is the system level device which commands the MakoECUs during normal operation. This device may be a system PLC, body control module (BCM), or another MakoECU configured as a supervisor.

2. Configuration PC - The configuration PC is a computer the user can add to the CAN network to flash, configure, and monitor the MakoECUs. This device is usually connected with a USB-to-CAN adapter. It is normally removed during normal operations.

## MakoCAN Package

The MakoCAN package is a software package used on the configuration PC to communicate with a MakoECU. It contains useful tools and functions for flashing, configuring, controlling, and monitoring. Functions include:

- Setting device CAN bitrate
- Setting device Address
- Enabling and Disabling the device
- Sending commands control channel commands
- Reading, writing, and saving configuration parameters
- Monitoring and logging telemetry signals and status bits

Please refer to the MakoCAN Package documentation for more detailed instructions.

# CAN Setup

**CAN Bitrate:** The MakoECU is shipped with a default CAN bus bitrate of 500kbps. The device can be configured to use 100k, 250k, 500k, and 1M bps. Use MakoCAN to configure the CAN bitrate.

**Device Address:** A MakoECU (and Host Controller) can be assigned any address between 0 to 126. Address 127 is reversed for all broadcast messages (messages intended for all Mako devices on the network). The MakoECU is shipped with a default address of 0. Use MakoCAN to change a device address.

**Device Interoperability:** The MakoECU is designed to operate with other CAN 2.0b compliant devices. Please contact us for support on network settings and address allocations.

**CAN Termination:** The MakoECU is not terminated and is not user configurable. There is an option to terminate the unit at the factory, please reach out if this is needed.

# CAN Message Scheme

The MakoECU was designed to be used within a network of multiple MakoECUs and host controllers. In order to facilitate efficient and reliable communication between the nodes, the Mako uses a proprietary CAN communication scheme. The section is intended to help the user understand how this scheme works so that host controller code can be developed and ensure interoperability with non Mako devices on the bus.

**CAN Message Address:**

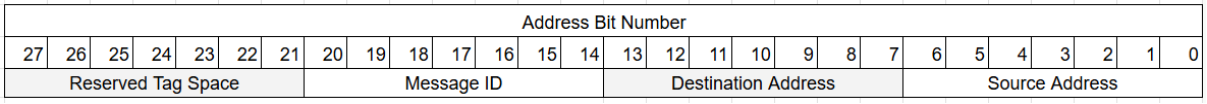


Figure. CAN Message Address Format

The address for a CAN message sent from a Mako device contains 4 pieces of information:

1. Source Address - Address of device sending the message
2. Destination Address - Address of the target device. Destination 127 is all devices (broadcast)
3. Message ID - ID of type of message being sent
4. Tags - Reserved space for specialized functions like firmware load

**CAN Message IDs:**

| Msg ID | Sender | Src ID | Dest ID | Msg Address | Address Hex | Name | Send Rate |
|---|---|---|---|---|---|---|---|
| 0 | ECU | 0 | 10 | 1280 | 500 | INTERNAL USE - ECU BEACON | 100ms |
| 1 | ECU | 0 | 10 | 17664 | 4500 | INTERNAL USE | |
| 2 | Host | 10 | 0 | 32778 | 800A | ECU COMMAND | On-Demand |
| 3 | Host | 10 | 0 | 49162 | C00A | SET ADDRESS | On-Demand |
| 4 | Host | 10 | 0 | 65546 | 1000A | INTERNAL USE | |
| 5 | Host | 10 | 0 | 81930 | 1400A | RESERVED FOR FW LOAD | |
| 6 | Host | 10 | 0 | 98314 | 1800A | RESERVED FOR FW LOAD | |
| 7 | Host | 10 | 0 | 114698 | 1C00A | RESERVED FOR FW LOAD | |
| 8 | Host | 10 | 0 | 131082 | 2000A | RESERVED FOR FW LOAD | |
| 9 | Host | 10 | 0 | 147466 | 2400A | INTERNAL USE | |
| 10 | Host | 10 | 0 | 163850 | 2800A | INTERNAL USE | |
| 20 | ECU | 0 | 10 | 328960 | 50500 | ECU STATUS | Configurable |
| 21 | Host | 10 | 0 | 344074 | 5400A | CHANNEL COMMAND | Periodic |
| 22 | Host | 10 | 0 | 360458 | 5800A | SET CONFIGURATION VALUE | On-Demand |
| 23 | Host | 10 | 0 | 376842 | 5C00A | GET CONFIGURATION VALUE | On-Demand |
| 24 | ECU | 0 | 10 | 394496 | 60500 | REQUESTED CONFIGURATION VALUE | Response |
| 25 | ECU | 0 | 10 | 410880 | 64500 | TELEMETRY MESSAGE | Configurable |
| 26 | ECU | 0 | 10 | 427264 | 68500 | CONTROL STATUS BITS | Configurable |
| 27 | ECU | 0 | 10 | 443648 | 6C500 | VALVE DRIVER OUTPUTS CURRENT | Configurable |
| 28 | ECU | 0 | 10 | 460032 | 70500 | ANALOG INPUTS VOLTAGE | Configurable |
| 29 | Host | 10 | 0 | 475146 | 7400A | GET TELEMETRY VALUE | On-Demand |
| 30 | ECU | 0 | 10 | 492800 | 78500 | REQUESTED TELEMETRY VALUE | Response |
| 31 | Host | 10 | 0 | 507914 | 7C00A | SET EXTERNAL INPUT VALUE | Periodic |

Table. Example CAN Message Table for a Two Device Network (Host =10, ECU=0)

The table above shows all the CAN message IDs possible for a MakoECU network. The example addresses are for a two device network where the ECU has an ID = 0 and the host controller = 10.

Refer to the appendix for details on each message IDs dataframe structure.

# Configuration Parameters

## Configuration Pages

The configuration management system allows the user to configure parameters within the MakoECU. The configuration parameters are separated into sections, called pages, pages group parameters together relevant to functional areas of the ECU. The pages are as follows:

- Page 0 - ECU Configuration - Contains parameters relating to CAN configuration and assignment of signals to telemetry.
- Page 1 - System Configuration - Contains parameters to configure system monitor, input signals, and diagnostics
- Page 2 - Control Channel 1 Configuration - Contains channel 1 configuration parameters
- Page 3 - Control Channel 2 Configuration - Contains channel 2 configuration parameters
- Page 4 - Control Channel 3 Configuration - Contains channel 3 configuration parameters

Each page contains rows of parameters. The number of parameters per page is not limited. Each row is referred to as an index and the index system is zero based.

Refer to the appendix for a detailed list of each page's parameters.

## Configuration Parameters

The configuration parameter is a float value. Each parameter has four properties:

1. Page ID - 0-4 - ID of page where parameter is located
2. Page Index - Index of value on the page
3. Name - name of parameter
4. Value - value of parameter

## Configuration Management

To change the configuration parameter values the MakoCAN package can be used. This package contains many useful functions which facilitate the management of configuration within the MakoECU:

- Get Configuration Value - retrieves the current value
- Set Configuration Value - sets a configuration value
- Get Configuration Page - retrieves all the values from a page
- Set Configuration Page - sets all the values of a page based on a page configuration .csv file
- Set ECU Configuration - sets pages based on a .csv file containing page configuration .csv files for each page

There are a few other functions for configuration management:

- Save Configuration - the save configuration function saves all the current pages to persistent memory
- Default Configuration - sets the configuration to the default values associated with the firmware build

# Control Application Function

As stated in the Control Application Overview section, the Control Application is the core of the control logic within the MakoECU. The section will focus more in-depth on the Control Application functionality and how to configure it for applications.

## System Monitor

The System Monitor function is to monitor system parameters which relate to safety and system level diagnostics. Control Channels will only be enabled and allowed to send current to valve coils (motion enabled) if the criteria within the system monitor is satisfied.

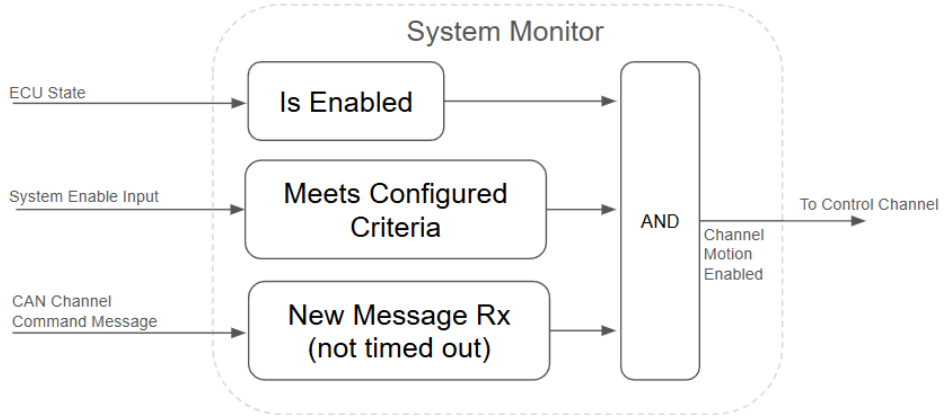


Figure. System Monitor Motion Enablement Logic

The system monitor motion enable logic looks at three criteria based on ECU state, a system enable input, and validity of the channel command message.

**ECU State:** The ECU State must be Enabled for motion to be enabled. This is the only criteria that can't be modified or disabled. The ECU State also directly enables the valve drivers so this acts as a double safety mechanism.

The ECU State is set using the makocan ecu command function.

**System Enable Input:** This input allows the user to configure an analog input or external input to act as a kill switch. If the system enable input value does not meet the configured criteria, then motion will be disabled. This function can be disabled.

Configuration parameters related to the system enable input are in the table below:

| Page 1 - System Configurations | | |
|---|---|---|
| **Page Index** | **Parameter Name** | **Expected Value** |
| see App. | sys_monitor_enables_low | 0 or 1 |
| see App. | sys_monitor_enables_high | 0 or 1 |
| see App. | sys_monitor_enables_within_range | 0 or 1 |
| see App. | sys_monitor_low_threshold_mv | millivolt value |
| see App. | sys_monitor_high_threshold_mv | millivolt value |

system_monitor_enables_low - enables the low threshold criteria meaning motion will be disabled if the system enable input falls below the system_monitor_low_threshold_mv value. An example of this is a kill switch that goes to ground (0v) when disconnected (stop motion state).

system_monitor_enables_high - enables the high threshold criteria meaning motion will be disabled if the system enable input raises above the system_monitor_high_threshold_mv value. An example of this is a kill switch that goes to supply (5v) when disconnected (stop motion state).

system_monitor_enables_within_range - enables the within range criteria meaning motion will be disabled if the system enable input goes outside of the range set by system_monitor_low_threshold_mv and system_monitor_high_threshold_mv value. An example of this is a kill switch that sends 2.5v when connected (enable motion state).

**Channel Command Message Timeout:** Channel command messages must be received at a periodic rate for the channel motion to stay enabled. This protects against unintended motion if the host controller loses communication with the MakoECU. The timeout criteria can be disabled.

Configuration parameters related to the channel command timeout are in the table below:

| Page 1 - System Configurations | | |
|---|---|---|
| **Page Index** | **Parameter Name** | **Expected Value** |
| see App. | sys_monitor_cmd_timeout_ms | time in ms |
| see App. | sys_monitor_cmd_timeout_enable | 0 or 1 |

system_monitor_cmd_timeout_ms - sets the timeout value
system_monitor_cmd_timeout_enable - enables the channel command timeout criteria

## Input Signals

The MakoECU uses input signals as control feedback for the control channels and as the system enable input. There are two types of input signals:

1. ECU Analog Inputs (Internal Inputs): The MakoECU has 4x 0-5v analog inputs.
2. External Inputs: Input signals received through the set external input value CAN message from a remote device.

The input numbers have permanent assignments:

- Input Signal 1 - Channel 1 Control Feedback - default is analog input 1
- Input Signal 2 - System Enable Input - default is analog input 2
- Input Signal 3 - Channel 2 Control Feedback - default is analog input 3
- Input Signal 4 - Channel 3 Control Feedback - default is analog input 4

**Internal/External Assignment:** An input signal can be assigned the internal analog input or external input using these configuration parameters.

| Page 1 - System Configurations | | |
|---|---|---|
| **Page Index** | **Parameter Name** | **Expected Value** |
| see App. | analog_in1_use_external | 0 or 1 |
| see App. | analog_in2_use_external | 0 or 1 |
| see App. | analog_in3_use_external | 0 or 1 |
| see App. | analog_in4_use_external | 0 or 1 |

**Input Signal Diagnostics:** Each input signal can have diagnostics configured for it which will set diagnostic status bits in the control status CAN message. There are three types of input signal diagnostics:

1. Input Fault - This indicates a fault with the analog input circuitry or electrical condition causing an input fault. This fault can not be disabled. If this fault does occur the control application will hold the last good known input value.
2. Input Level Low - Indicates the analog input is below a configured threshold. For example a 0.5-4.5v range sensor reading 0v (open circuit).
3. Input Level High - Indicates the analog input is above a configured threshold. For example a 0.5-4.5v range sensor reading 5v (short circuit).

Configuration parameters for configuring input diagnostics are: (see appendix for specific index for each input)

| Page 1 - System Configurations | | |
|---|---|---|
| **Page Index** | **Parameter Name** | **Expected Value** |
| see App. | analog_inX_diag_enable_low | 0 or 1 |
| see App. | analog_inX_diag_enable_high | 0 or 1 |
| see App. | analog_inX_diag_low_threshold_mv | millivolt value |
| see App. | analog_inX_diag_high_threshold_mv | millivolt value |

# Valve Coil Current Diagnostics

Coil current sensing in the MakoECU allows for closed loop current control and identifying when the current driver is not working properly. The Control Application monitors the sensed value current and compares it to the commanded current. If these values do not agree, then the ECU will set a valve driver fault diagnostic. These diagnostic flags are part of the Control Status message. The current diagnostic can be configured to give reliable results and reduce false indications. It can also be disabled with the configuration.

| Page 1 - System Configurations | | |
|---|---|---|
| **Page Index** | **Parameter Name** | **Expected Value** |
| see App. | curr_driver_diag_enable | 0 or 1 |
| see App. | curr_driver_error_filt_s | filter constant in s |
| see App. | curr_driver_error_threshold_ma | threshold in ma |

curr_driver_diag_enable - enables the current driver diagnostics
curr_driver_error_filt_s - sets the time constant on the error filter. The filter reduces the chance of a false indication due to a fast transient.
curr_driver_error_threshold_ma - amount of error (current sensed - commanded) needed to trigger diagnostic

# Control Channel

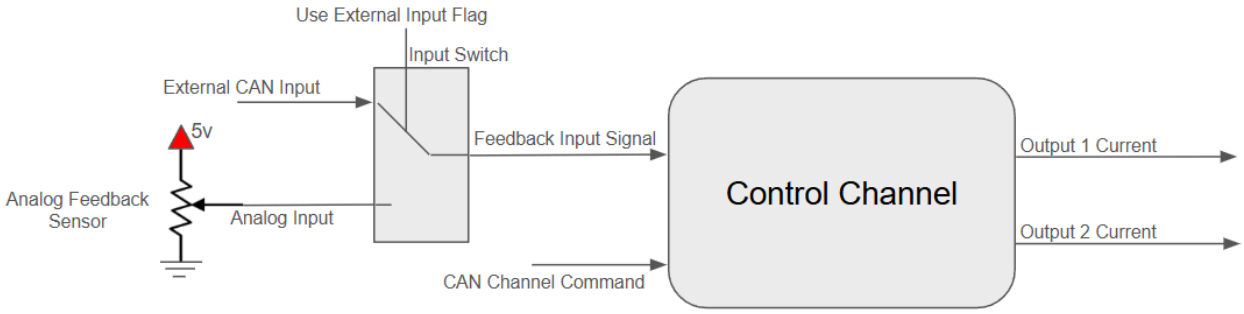


Figure. Control Channel Input/Output Diagram

The MakoECU has 3 control channels. Each control channel controls has 1 input signal, 1 command input, and 2 output current commands (see figure). This gives each channel the ability to do closed loop feedback control on a hydraulic circuit with a 2 coil control valve. The control channel has many control modes that offer both closed loop and open loop control options. The control modes are:

1. Closed Loop Position Control
2. Closed Loop Velocity Control
3. Two Output Closed Loop Valve Current Control
4. Two Output Digital Control (On/Off)
5. Hot Shot Sequence

In addition to these control modes there are two control modes to support system calibration:

1. Valve Calibration Process
2. Feedback Sensor Calibration Point Capture

This section will cover the details of a control channel and how to configure it for applications.

# Control Channel Function

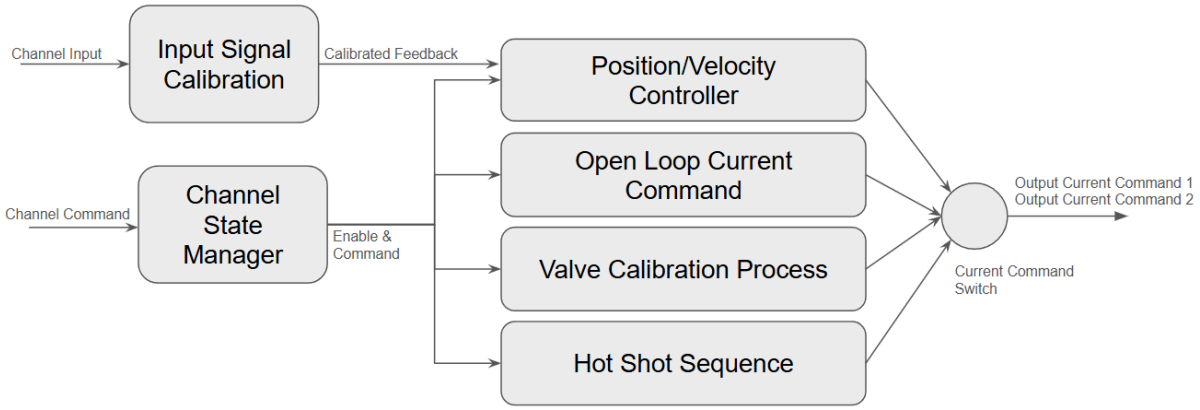


Figure. Control Channel Internal Functions

The control contains several internal functions.  Here is a description of each function:

# Input Signal Calibration

The input single calibration function applies a linear calibration to the input signal and outputs the calibrated feedback signal used by the position/velocity controller.  The linear calibration is defined by two points.  The controller uses normalized feedback so the extents of the calibrated feedback are 0 and 1.  Typically for a hydraulic actuator the 0 position is fully retracted and 1 position is extended (see figure).

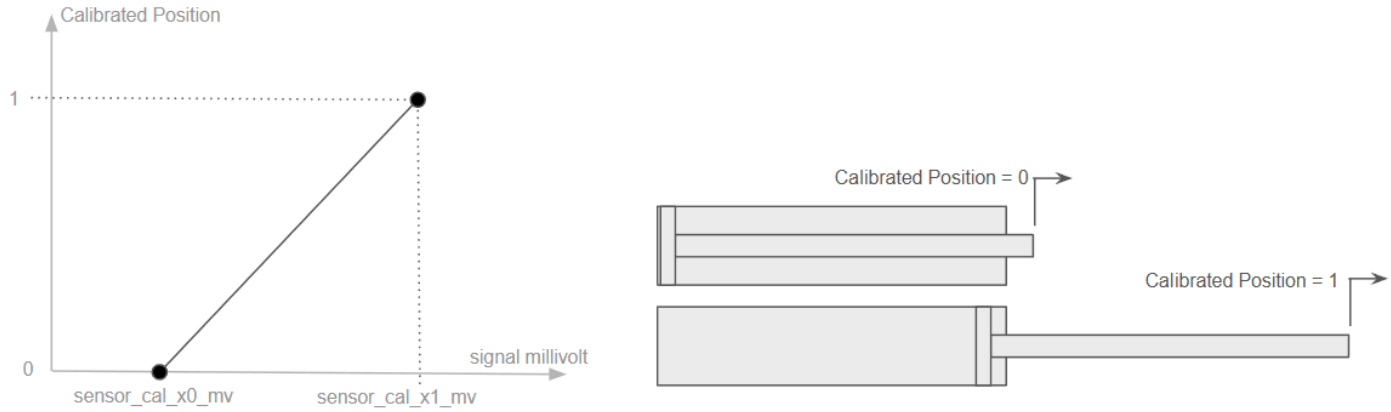


Figure. Diagram of Typical Actuator Calibration and Linear Calibration Form

The configurations related to the input signal calibration function are listed below:

| Page 2,3,4 - Channel Configurations | | |
|---|---|---|
| **Page Index** | **Parameter Name** | **Expected Value** |
| see App. | sensor_cal_x0_mv | cal point in mv |
| see App. | sensor_cal_x1_mv | cal point in mv |
| see App. | sensor_filter_cut_hz | cutoff frequency |
| see App. | sensor_filter_mode | 0,1, or 2 |
| see App. | sensor_min_valid_x0cal_mv | 0-5000 mv |
| see App. | sensor_max_valid_x0cal_mv | 0-5000 mv |
| see App. | sensor_min_valid_x1cal_mv | 0-5000 mv |
| see App. | sensor_max_valid_x1cal_mv | 0-5000 mv |

**Sensor Calibration:** The sensor calibration table endpoints are set with the sensor_cal_x0_mv, and sensor_cal_x1_mv configuration parameters.

Sensor calibration can be automated by using the sensor calibration channel control mode (control mode = 6). Sending this control model, writes the current sensor reading as a calibration parameter to the configuration.

| Channel Control Modes | Command # | Command Definition | Units | Resolution | Min | Max | Comments |
|---|---|---|---|---|---|---|---|
| 6 = Sensor Calibration | Command 1 | calibration sample point | | 1 | 1 | 2 | Command = 1, writes the calibrated 0 postion |
| | Command 2 | not used | | | | | Command = 2, writes the calibrated 1 position |

Example: “makocan_channel_command 1 6 1 0” and “makocan_channel_command 1 6 2 0”

**Sensor Calibration Diagnostics:** Sensor calibration diagnostics are available to detect if a calibration point is outside of an expected range. The configuration parameters to set the expected range are sensor_min_valid_x0cal_mv, sensor_max_valid_x0cal_mv, sensor_min_valid_x1cal_mv, and sensor_max_valid_x1cal_mv

**Input Signal Filtering:** The input signal calibration function also applies filtering to the input signal, using the sensor_filter_mode and sensor_filter_cut_hz parameters, the filtering can be tuned to achieve best performance for the feedback sensor being used. The filter modes are:

0 - No Filtering
1 - First Order Filter
2 - Second Order Butterworth Filter

The amount of filtering is adjusted with the sensor_filter_cut_hz parameter.

# Channel State Manager

The channel state manager function handles the channel state based on the commanded state. All but one of the control modes are stateless so that the state commanded is the state the channel will enter.

The valve calibration process is a control mode which has a latched state while the valve is being calibrated. The control mode mode will remain in the valve calibration control mode until the process is finished. The channel state manager returns the channel to the disabled mode after.

| Channel Control Modes | Command # | Command Definition | Units | Resolution | Min | Max |
|---|---|---|---|---|---|---|
| 0 = Disabled | Command 1 | not used | | | | |
| | Command 2 | not used | | | | |
| 1 = Position Control | Command 1 | position target | normalized value | 0.00005 | 0 | 1 |
| | Command 2 | not used | | | | |
| 2 = Velocity Control | Command 1 | not used | | | | |
| | Command 2 | velocity target | normalized value | 0.01 | -100 | 100 |
| 3 = Proportional Current Control | Command 1 | output 1 command | milliamp | 1 | 0 | 1000 |
| | Command 2 | output 2 command | milliamp | 1 | 0 | 1000 |
| 4 = Digital Current Control | Command 1 | output 1 command | 0 or 1 | 1 | 0 | 1 |
| | Command 2 | output 2 command | 0 or 1 | 1 | 0 | 1 |
| 5 = Valve Calibration Process | Command 1 | not used | | | | |
| | Command 2 | not used | | | | |
| 6 = Sensor Calibration | Command 1 | calibration sample point | | 1 | 1 | 2 |
| | Command 2 | not used | | | | |
| 7 = Hot Shot Sequence | Command 1 | not used | | | | |
| | Command 2 | not used | | | | |

Table. Control Modes and Associated Commands

## Hot Shot Mode

The hot shot mode (control mode = 7) is a mode used to clear silt from valves to improve a valve that is sticking or reacting slowly.

**WARNING - ONLY USE THE HOT SHOT MODE WHEN THERE IS NO HYDRAULIC PRESSURE, DAMAGE COULD OCCUR OTHERWISE!**

To enable the hot shot mode send a control mode = 7 command to the channel.  No additional channel commands are needed with this mode

The on and off current and frequency of the hot shot signal can be set using the following configuration parameters:

| Page 2,3,4 - Channel Configurations | | |
|---|---|---|
| Page Index | Parameter Name | Expected Value |
| see App. | hot_on_current_cmd_ma | on current in ma |
| see App. | hot_off_current_cmd_ma | off current in ma |
| see App. | hot_frequency_hz | on/off frequency |

## Valve Calibration Mode

**The valve calibration mode (control mode = 5):** This mode is used to automatically determine the deadband characteristics of the valve.

**WARNING - MOTION DOES OCCUR AND HYDRAULIC PRESSURE MUST BE PRESENT FOR THE PROCESS TO WORK CORRECTLY. ENSURE THE ACTUATOR IS CLEAR OF AN OBSTRUCTIONS.**

Refer to the supplemental instructions for detailed instructions on configuring and using this mode.

# Current Command Modes

There are 2 current current command modes:

1. Proportional Current (control mode = 3)
2. Digital On/Off (control mode = 4)

Both modes work similarly as they pass current command from the user onto the valve driver. Feedback is not used in these modes.

**Proportional Current Mode (control mode = 3):** The user commands the current desired to each valve output of the channel.

| Channel Control Modes | Command # | Command Definition | Units | Resolution | Min | Max |
|---|---|---|---|---|---|---|
| 3 = Proportional Current Control | Command 1 | output 1 command | milliamp | 1 | 0 | 1000 |
| | Command 2 | output 2 command | milliamp | 1 | 0 | 1000 |

**Digital Current Mode (control mode = 4):** The user commands to turn the channel's outputs on and off. The on and off current commands are set with a configuration parameter

| Channel Control Modes | Command # | Command Definition | Units | Resolution | Min | Max |
|---|---|---|---|---|---|---|
| 4 = Digital Current Control | Command 1 | output 1 command | 0 or 1 | 1 | 0 | 1 |
| | Command 2 | output 2 command | 0 or 1 | 1 | 0 | 1 |

Configuration parameters for digital current mode are:

| Page 2,3,4 - Channel Configurations | | |
|---|---|---|
| Page Index | Parameter Name | Expected Value |
| see App. | icmd_on_current_cmd_ma | on current in ma |
| see App. | icmd_off_current_cmd_ma | off current in ma |

# Closed Loop Position & Velocity Control Modes

The position and velocity control modes are the two closed loop feedback modes. Each control channel is assigned a feedback analog signal. This signal is used as the position and velocity (derivative of position) feedback.

Channel feedback signal assignments

| Channel Name | Pin # | ECU Internal Channel |
|---|---|---|
| analog_input_1 | 17 | channel1_feedback |
| analog_input_2 | 18 | channel 2 feedback |
| analog_input_3 | 19 | channel 3 feedback |
| analog_input_4 | 20 | system feedback |

CAN message feedback can be assigned to a control channel using the ‘analog_inX_use_external’ configuration in the system tunable config file.  If this configuration is set to 1, then the CAN message value is used instead of the analog input.

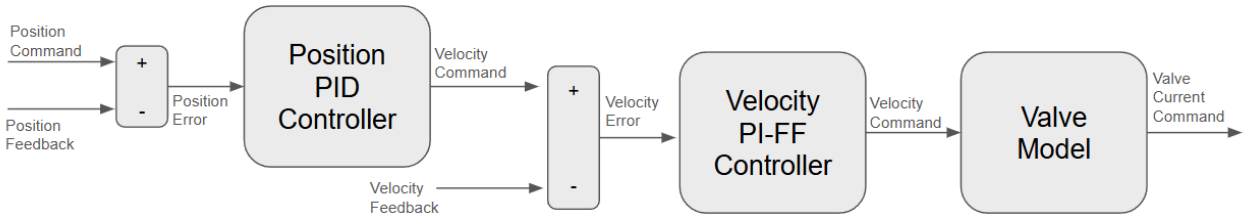


Figure. Diagram of closed loop controller architecture

**Position Control Mode (control mode = 1):** In the position control mode the user commands the position of the actuators and the controller positions the actuator to meet the command.

| Channel Control Modes | Command # | Command Definition | Units | Resolution | Min | Max | Comments |
|---|---|---|---|---|---|---|---|
| 1 = Position Control | Command 1 | position target | normalized value | 0.00005 | 0 | 1 | 0 is fully retracted |
| | Command 2 | not used | | | | | 1 is fully extended |

**Velocity Control Mode (control mode = 2):** In the velocity control mode the user commands the velocity of the actuator and the controller works to keep that actuator speed.

| Channel Control Modes | Command # | Command Definition | Units | Resolution | Min | Max |
|---|---|---|---|---|---|---|
| 2 = Velocity Control | Command 1 | not used | | | | |
| | Command 2 | velocity target | normalized value | 0.01 | -100 | 100 |

### Closed Loop Control Configurations

| Page 2,3,4 - Channel Configurations | | |
|---|---|---|
| **Page Index** | **Parameter Name** | **Config Name** |
| see App. | position control P gain | pos_p_gain |
| see App. | position control I gain | pos_i_gain |
| see App. | position control D gain | pos_d_gain |
| see App. | velocity control P gain | vel_p_gain |
| see App. | velocity control I gain | vel_i_gain |
| see App. | velocity control FF gain | vel_ff_gain |
| see App. | max position command rate of change | pos_dtarget_max |
| see App. | maximum velocity command | pos_vel_cmd_max |
| see App. | minimum velocity command | pos_vel_cmd_min |
| see App. | velocity command deadband | vel_vel_cmd_deadband |
| see App. | extend (P-A) valve gain | vel_ext_vlv_gain |
| see App. | retract (P-B) valve gain | vel_ret_vlv_gain |
| see App. | maximum extend valve current command | vel_ext_vlv_max_ma |
| see App. | maximum retract valve current command | vel_ret_vlv_max_ma |

## Current Driver Configuration

The current drivers are configured through the channel configuration csv file.  They are the coil electrical characteristics of the valve coil to help the feedforward part of the current driver.

| Page 2,3,4 - Channel Configurations | | |
|---|---|---|
| **Page Index** | **Parameter Name** | **Config Name** |
| see App. | Channel Output 1 Max Current | out1_max_current_ma |
| see App. | Channel Output 2 Max Current | out2_max_current_ma |
| see App. | Output 1 Dither Frequency | out1_dither_freq_hz |
| see App. | Output 2 Dither Frequency | out2_dither_freq_hz |
| see App. | Output 1 Dither Amplitude | out1_dither_amp_ma |
| see App. | Output 2 Dither Amplitude | out2_dither_amp_ma |
| see App. | Output 1 Coil Resistance | out1_resistance_ohms |
| see App. | Output 2 Coil Resistance | out2_resistance_ohms |
| see App. | Output 1 Coil Inductance | out1_inductance_mh |
| see App. | Output 2 Coil Inductance | out2_inductance_mh |

# Appendix. ECU Configuration Page 0 (csv file)

| Index | Name | Value |
|---|---|---|
| 0 | ecu_config_id_can_bitrate_khz | 500 |
| 1 | ecu_config_id_status_message_interval_ms | 10 |
| 2 | ecu_config_id_diagnostic_message_interval_ms | 10 |
| 3 | ecu_config_id_analog_message_interval_ms | 10 |
| 4 | ecu_config_id_valve_driver_outputs_current_interval_ms | 10 |
| 5 | ecu_config_id_feedback_message_0_interval_ms | 10 |
| 6 | ecu_config_id_feedback_message_0_value_id_0 | 0 |
| 7 | ecu_config_id_feedback_message_0_value_id_1 | 1 |
| 8 | ecu_config_id_feedback_message_0_value_id_2 | 2 |
| 9 | ecu_config_id_feedback_message_0_value_id_3 | 3 |
| 10 | ecu_config_id_feedback_message_0_value_id_4 | 4 |
| 11 | ecu_config_id_feedback_message_0_value_id_5 | 5 |
| 12 | ecu_config_id_feedback_message_0_value_id_6 | 6 |
| 13 | ecu_config_id_feedback_message_0_value_id_7 | 7 |
| 14 | ecu_config_id_feedback_message_0_value_id_8 | 8 |
| 15 | ecu_config_id_feedback_message_0_value_id_9 | 9 |
| 16 | ecu_config_id_feedback_message_1_interval_ms | 10 |
| 17 | ecu_config_id_feedback_message_1_value_id_0 | 12 |
| 18 | ecu_config_id_feedback_message_1_value_id_1 | 13 |
| 19 | ecu_config_id_feedback_message_1_value_id_2 | 14 |
| 20 | ecu_config_id_feedback_message_1_value_id_3 | 15 |
| 21 | ecu_config_id_feedback_message_1_value_id_4 | 16 |
| 22 | ecu_config_id_feedback_message_1_value_id_5 | 17 |
| 23 | ecu_config_id_feedback_message_1_value_id_6 | 18 |
| 24 | ecu_config_id_feedback_message_1_value_id_7 | 19 |
| 25 | ecu_config_id_feedback_message_1_value_id_8 | 20 |
| 26 | ecu_config_id_feedback_message_1_value_id_9 | 21 |
| 27 | ecu_config_id_feedback_message_2_interval_ms | 10 |
| 28 | ecu_config_id_feedback_message_2_value_id_0 | 24 |
| 29 | ecu_config_id_feedback_message_2_value_id_1 | 25 |
| 30 | ecu_config_id_feedback_message_2_value_id_2 | 26 |
| 31 | ecu_config_id_feedback_message_2_value_id_3 | 27 |
| 32 | ecu_config_id_feedback_message_2_value_id_4 | 28 |
| 33 | ecu_config_id_feedback_message_2_value_id_5 | 29 |
| 34 | ecu_config_id_feedback_message_2_value_id_6 | 30 |
| 35 | ecu_config_id_feedback_message_2_value_id_7 | 31 |
| 36 | ecu_config_id_feedback_message_2_value_id_8 | 32 |
| 37 | ecu_config_id_feedback_message_2_value_id_9 | 33 |

# Appendix. System Configuration Page 1 (csv file)

| Index | Name | Value |
|---|---|---|
| 0 | system_test_enable | 0 |
| 1 | sys_monitor_cmd_timeout_ms | 100 |
| 2 | sys_monitor_cmd_timeout_enable | 0 |
| 3 | sys_monitor_enables_low | 0 |
| 4 | sys_monitor_enables_high | 0 |
| 5 | sys_monitor_enables_within_range | 0 |
| 6 | sys_monitor_low_threshold_mv | 500 |
| 7 | sys_monitor_high_threshold_mv | 4500 |
| 8 | sys_voltage_enables_low | 1 |
| 9 | sys_voltage_enables_high | 0 |
| 10 | sys_voltage_low_threshold_v | 20 |
| 11 | sys_voltage_high_threshold_v | 30 |
| 12 | curr_driver_diag_enable | 1 |
| 13 | curr_driver_error_filt_s | 0.5 |
| 14 | curr_driver_error_threshold_ma | 500 |
| 15 | analog_in1_diag_enable_low | 1 |
| 16 | analog_in1_diag_enable_high | 1 |
| 17 | analog_in1_diag_low_threshold_mv | 0 |
| 18 | analog_in1_diag_high_threshold_mv | 5000 |
| 19 | analog_in2_diag_enable_low | 1 |
| 20 | analog_in2_diag_enable_high | 1 |
| 21 | analog_in2_diag_low_threshold_mv | 0 |
| 22 | analog_in2_diag_high_threshold_mv | 5000 |
| 23 | analog_in3_diag_enable_low | 1 |
| 24 | analog_in3_diag_enable_high | 1 |
| 25 | analog_in3_diag_low_threshold_mv | 0 |
| 26 | analog_in3_diag_high_threshold_mv | 5000 |
| 27 | analog_in4_diag_enable_low | 1 |
| 28 | analog_in4_diag_enable_high | 1 |
| 29 | analog_in4_diag_low_threshold_mv | 0 |
| 30 | analog_in4_diag_high_threshold_mv | 5000 |
| 31 | analog_in1_use_external | 0 |
| 32 | analog_in2_use_external | 0 |
| 33 | analog_in3_use_external | 0 |
| 34 | analog_in4_use_external | 0 |

# Appendix. Channel Configuration Page 2,3,4 (csv file)

| Index | Name | Value |
|---|---|---|
| 0 | sensor1_cal_x0_mv | 500 |
| 1 | sensor1_cal_x1_mv | 4500 |
| 2 | sensor1_filter_cut_hz | 100 |
| 3 | sensor1_filter_mode | 1 |
| 4 | sensor1_min_valid_x0cal_mv | 0 |
| 5 | sensor1_max_valid_x0cal_mv | 5000 |
| 6 | sensor1_min_valid_x1cal_mv | 0 |
| 7 | sensor1_max_valid_x1cal_mv | 5000 |
| 8 | vel_p_gain | 0 |
| 9 | vel_i_gain | 0 |
| 10 | vel_ff_gain | 0 |
| 11 | vel_ext_vlv_gain | 0 |
| 12 | vel_ret_vlv_gain | 0 |
| 13 | vel_vel_cmd_deadband | 0 |
| 14 | vel_ext_vlv_max_ma | 0 |
| 15 | vel_ret_vlv_max_ma | 0 |
| 16 | pos_p_gain | 0 |
| 17 | pos_i_gain | 0 |
| 18 | pos_d_gain | 0 |
| 19 | pos_dtarget_max | 0 |
| 20 | pos_vel_cmd_max | 0 |
| 21 | pos_vel_cmd_min | 0 |
| 22 | icmd_on_current_cmd_ma | 500 |
| 23 | icmd_off_current_cmd_ma | 0 |
| 24 | icmd_peak_current_ma | 0 |
| 25 | icmd_peak_duration_ms | 0 |
| 26 | vcal_ext_bias_ma | 0 |
| 27 | vcal_ret_bias_ma | 0 |
| 28 | hot_on_current_cmd_ma | 500 |
| 29 | hot_off_current_cmd_ma | 0 |
| 30 | hot_frequency_hz | 2 |
| 31 | out1_max_current_ma | 1200 |
| 32 | out1_resistance_ohms | 34 |
| 33 | out1_inductance_mh | 1 |
| 34 | out1_dither_freq_hz | 140 |
| 35 | out1_dither_amp_ma | 50 |
| 36 | out2_max_current_ma | 1200 |
| 37 | out2_resistance_ohms | 34 |
| 38 | out2_inductance_mh | 1 |
| 39 | out2_dither_freq_hz | 140 |
| 40 | out2_dither_amp_ma | 0 |