# Evaluation Board User Guide

# MAX22216

Evaluates the MAX22216, MAX22217, and MAX22216V

## General Description

The MAX22216-EVAL-KIT (MAX22216EVKIT) includes the MAX22216-EVAL board alongside the Eselsbruecke and Landungsbruecke boards, to be used together with the TMCL-IDE as part of the Analog Devices Inc. Trinamic TMCL ecosystem.

The MAX22216-EVAL evaluates the MAX22216, MAX22217, and MAX22216V in combination with the TRINAMIC evaluation board system or as a standalone board. It uses the standard schematic and offers several options to test different modes of operation. To evaluate the MAX22217, set SNSF[1:0] = "10" for all channels while in use.

Advanced diagnostic functions are available to improve system reliability and enable predictive maintenance. These include the detection of plunger movement (DPM), travel time measurement, inductance measurement, open-load detection (OL), and real-time current monitoring through the serial peripheral interface (SPI).

The MAX22216, MAX22217, and MAX22216V feature a full set of protection circuits, including overcurrent protection (OCP), overtemperature protection (OVT), and undervoltage lockout (UVM). A fault indicator pin is asserted whenever faults are detected.

## Features and Benefits

- Quad smart serial-controlled 4.5V to 36V half-bridges up to 1.7A/3.2A full scale
- SPI and OTP registers
- Highly flexible control methods (example, bridge-tied load, and parallel channels)
- Two-level current/voltage sequencer
- Detection of plunger movement
- Power-saving features
- Advanced diagnostics
- Full set of protections

## MAX22216EVKIT Content

| ITEM | DESCRIPTION |
|---|---|
| MAX22216-EVAL | MAX22216 evaluation board |
| Landungsbruecke | PC interface board |
| Eselsbruecke | Bridge connection board |

## Ordering Information

| PART | TYPE |
|---|---|
| MAX22216EVKIT# | Evaluation kit |

*#Denotes a RoHS-compliance.*

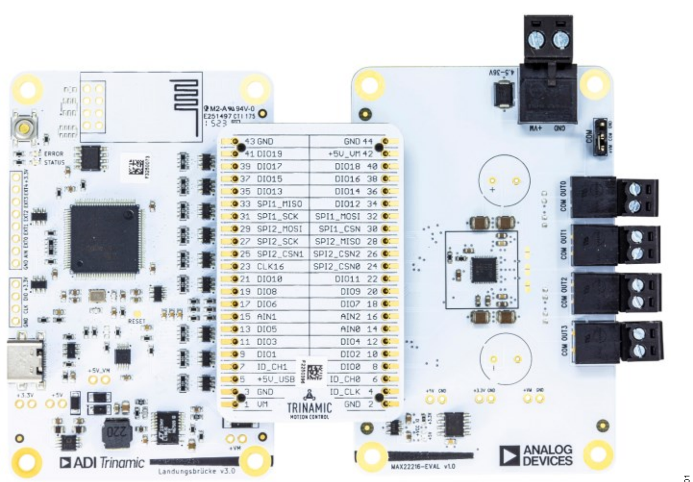



*Figure 1. Assembled MAX22216EVKIT*



One Analog Way, Wilmington, MA 01887-2356, U.S.A. | Tel: 781.329.4700 

# Quick Start

## Required Equipment and Software

- MAX22216EVKIT
- PC with latest TMCL-IDE (TMCL-IDE | Analog Devices)
- USB-C data cable
- Power supply (4.5V to 36V; power connector included in the kit)
- Inductive load (example, solenoid, contactor, and motor; load connectors included in the kit)

# Procedure

The MAX22216EVKIT contain the parts specified in the MAX22216EVKIT content table, and it requires assembly. The use of the MAX22216EVKIT also requires the installation and use of the TMCL-IDE (the integrated development environment made for developing applications using ADI Trinamic™ modules and chips).

While observing safe ESD practices, carefully remove the Landungsbruecke, Eselsbruecke, and MAX22216-EVAL boards out of the packaging. Quickly inspect the boards to ensure no damage occurred during shipment. Jumpers/shunts and needed output and power connectors are preinstalled before testing and packaging.

Use this evaluation kit with the following documents:

- MAX22216/MAX22217/MAX22216V data sheet
- MAX22216EVKIT user guide (this document)
- MAX22216EVKIT design files

For the latest versions of the documents listed above, use the following links:

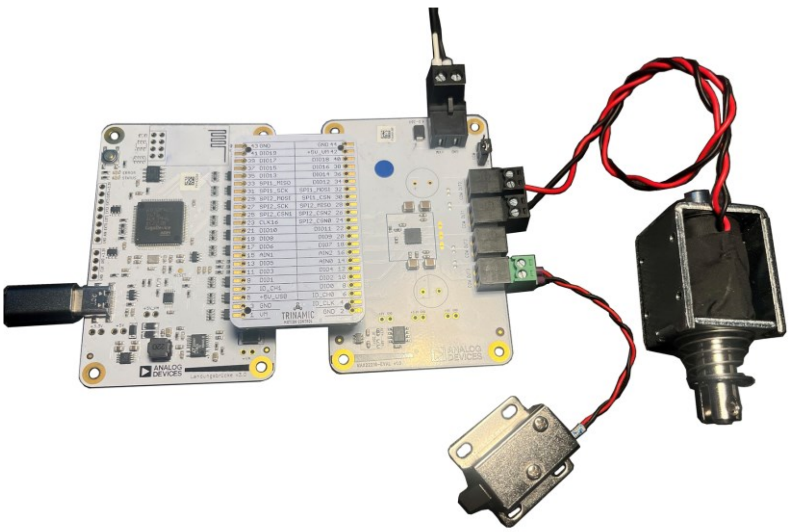
*MAX22216 data sheet and product info*

*MAX22216EVKIT evaluation board*

## TABLE OF CONTENTS

## Setup and Operation

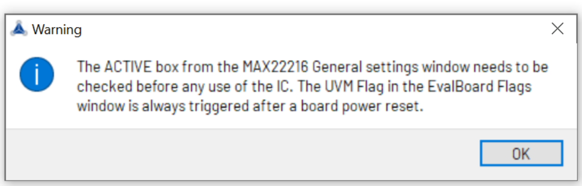
Connect the load/solenoid to the MAX22216-EVAL board. The actual configuration can be changed inside the MAX22216, but the default output is the single-ended, low-side MOSFET control. As the COM jumper is set to +VM-COM by default, the COM part of the output connector is connected to the VM of the board. As such, the board is set to drive a load connected to one of the board connectors (one wire to the COM, one wire to OUTx), as shown in the connection example. This can be done using the pluggable screw connectors provided with the kit. **Warning: Do not connect/disconnect the load while power is connected!**



002

*Figure 2. Example Setup*

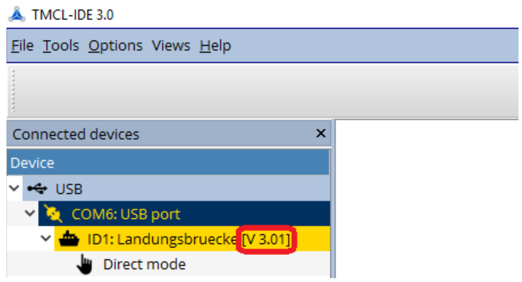
1. Make sure the latest version of the TMCL-IDE is installed. Download the TMCL-IDE from TMCL-IDE.
2. Open the TMCL-IDE and connect the Landungsbruecke with the attached MAX22216-EVAL by USB to the computer. For Microsoft Windows® 8 and higher, no driver is needed. For Microsoft Windows® 7, the TMCL-IDE installs the driver automatically.
3. After starting the TMCL-IDE, the part is automatically detected and configured. The following chapters present more details. When the TMCL-IDE detects the MAX22216EVKIT, a message pops up reminding to set the ACTIVE bit in the MAX22216 “General Settings” to HIGH. This is part of the safety systems of the MAX22216.




003

*Figure 3. MAX22216 Connection Warning Message*

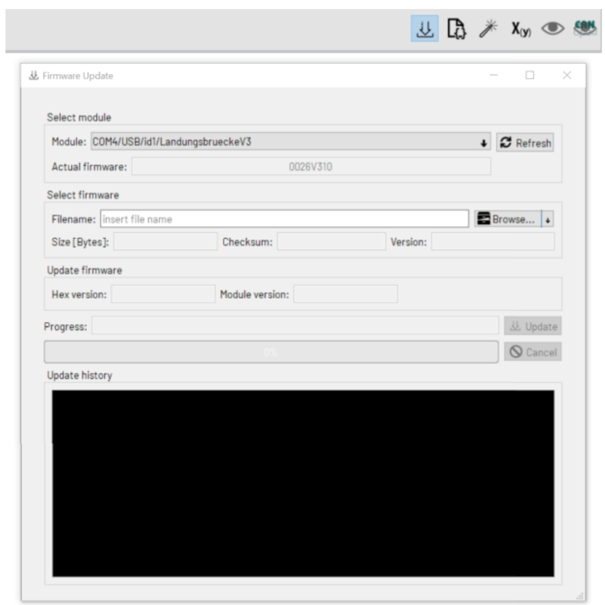
4. Verify the Landungsbruecke is using the latest firmware version. The connected device tree shows the firmware version. Download the newest firmware from LANDUNGSBRUECKE Evaluation Board *or* GitHub - analogdevicesinc/TMC-EvalSystem (from the “Releases” section).




004

*Figure 4. Firmware Version*

5. If needed, update the firmware. The firmware update tool can be found on the right side of the top bar.




005

*Figure 5. TMCL-IDE Firmware Update Tool*

6. The TMCL-IDE needs space to display all important information and to provide a good overview. Therefore, arrange the main window as needed. Using the full-screen mode is recommended.

7. In case the MAX22216EVKIT is not detected, the TMCL-IDE includes a dialog box for diagnostic tasks. The dialog box provides an overview of the connected motion controller and driver chips. A window pops up immediately after connecting the Landungsbruecke the first time. If it does not open automatically, access it by clicking the yellow tab named IDx: Landingsbruecker [V x.xx]. The “Board Assignment” tab shows the actual status of the connections. The “Settings” tab allows to choose basic settings or to reset the module to the factory default settings. The MAX22216EVKIT appears as a “Driver” board and should be detected automatically. If not, it can be set from the “Driver” board dropdown menu. The MAX22216-EVAL detection triggers the pop-up message mentioned in step 4.

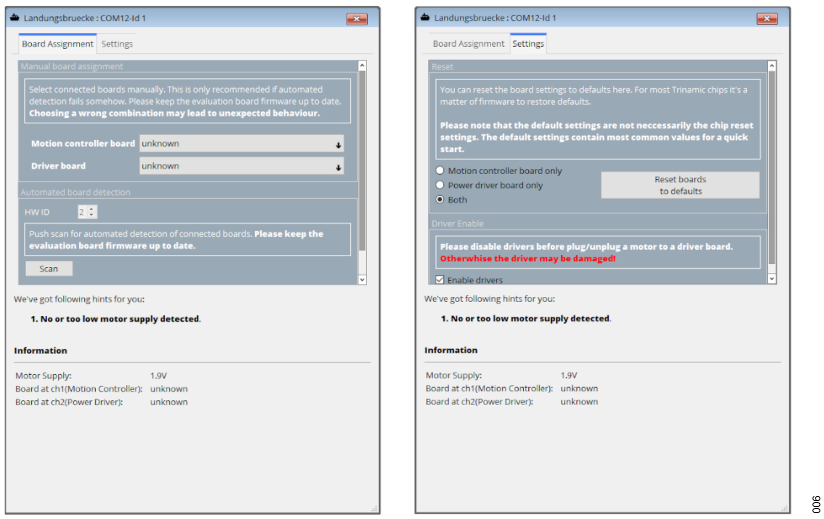



*Figure 6. Landungsbruecke Dialog Box*

8. The last hardware setup step before using the MAX22216 is to plug in power, using the provided power connector. The +VM input range is 4.5V to 36V, but it is recommended to start with a 12V/24 V power supply. Check the power by looking at the top left corner of the TMCL-IDE, where there is a VM monitor (with VM measurements done by the Landungsbruecke). The MAX22216 is designed to function only while power is supplied and cannot store settings when it is powered OFF (except if the register settings are OTPd). **Warning: The MAX22216 cannot be set without having VM power!**

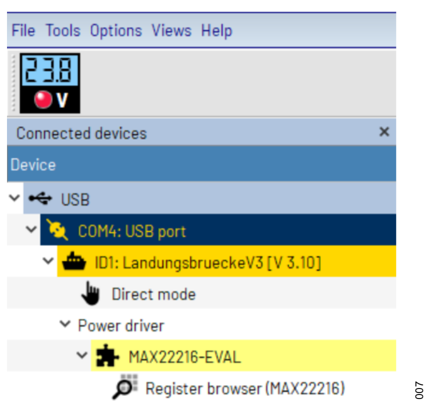



*Figure 7. TMCL-IDE Voltage Monitor*

9. Now the MAX22216EVKIT can be controlled using the TMCL-IDE, by setting up the MAX22216 using the GUI tools or direct access to the registers using the “Register Browser” tool.

10. **MAX22216 General Settings:** For safety, the MAX22216 must have the ENABLE pin connected to $V_{CC}$ (which is done automatically by the MAX22216EVKIT) and the ACTIVE register set to HIGH (easily accessible using the "General Settings" tool). This tool has a couple of more global settings such as the channel configuration (CHS) and the option to use the internal voltage reference for voltage control (VDRnVDRDUTY). In the example setup, one solenoid is connected to Channel 0 and Channel 1 lines, and the second solenoid is connected to Channel 3 and COM (COM connected by default to +VM). In this case, the CHS must be set to 6: 1IFB_2IHB (one independent full-bridge and two independent half-bridges).

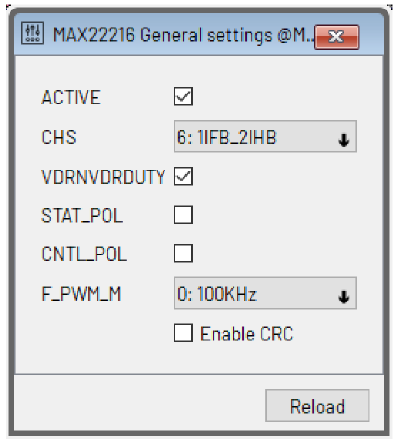



*Figure 8. MAX22216 General Settings: Example Setup*

11. **EvalBoard Flags:** Another step in using the MAX22216 is to check the faults. At startup, the "EvalBoard Flags" should show the UVM flag (and sometimes also the COMER flag), as the MAX22216 went through a power cycle. If any other FLAGS appear in the "EvalBoard Flags" tool, then the Landungsbruecke may not have the latest firmware, or the MAX22216 may have a problem. As the part is already set to ACTIVE, the flag can be cleared by pressing "Clear" flags. It is also recommended to have the "Auto Reload" active (by checking the box), so a real-time update of the MAX22216 faults can be seen.

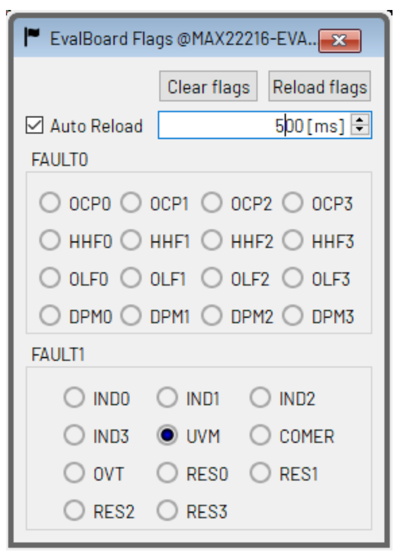
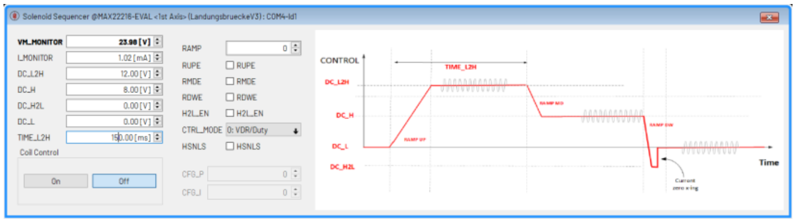



*Figure 9. MAX22216 EvalBoard Flags: Auto Reload Enabled*

12. **Channel 0 > Solenoid Sequencer:** The first solenoid in the example setup is connected to Channel 0 and Channel 1 lines. The part is already configured to have them in full-bridge configuration (from the MAX22216 “General Settings”), so now the solenoid can be controlled by either the Channel 0 or Channel 1 “Solenoid Sequencer”. The default control scheme of the MAX22216 is voltage control (VDR), and the basic two-level control is achieved by setting the HIT voltage (DC_L2H) and period (TIME_L2H), and the HOLD period (DC_H). See this control in the image presented in the tool. The channel is turned ON or OFF using the “Coil Control” buttons. The “Solenoid Sequencer” tool also allows to see the +VM voltage monitored by the MAX22216 in the VM_MONITOR section.




010

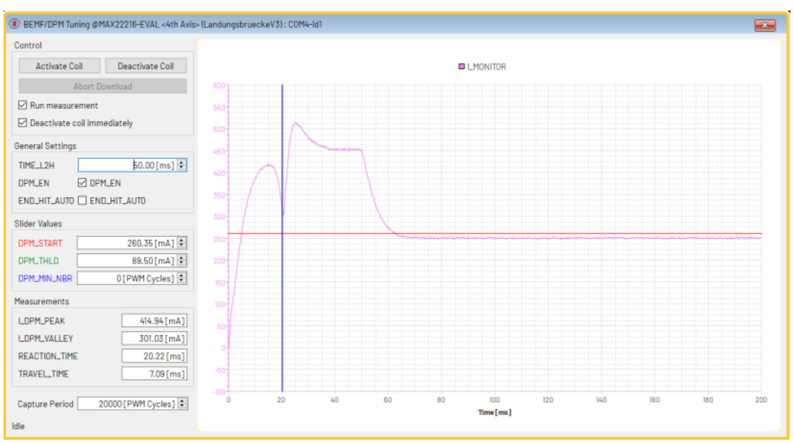
*Figure 10. Channel 0 > Solenoid Sequencer: Example Setup for 12V Solenoid (VDR Full-Bridge Mode)*

13. **Channel 3 > Solenoid Sequencer:** The second solenoid is connected to Channel 3 and COM (COM connected by default to +VM). The MAX22216 is already set to use this channel as a half-bridge, and the default half-bridge configuration uses the low-side MOSFET. As the COM is connected to +VM, this setup is correct. As the channel setup is done, the actual channel control settings can be done from the “Solenoid Sequencer”. In this example, the channel is driven using “Current Control (CDR)” - DC_L2H and DC_H are set as current levels, and P and I values must be set for the PI current regulator (if no values are set, the controller does not start).


011

*Figure 11. Channel 3 > Solenoid Sequencer: Example Setup for 12V/450mA Solenoid (CDR Half-Bridge Mode)*

14. **Channel 3 > BEMF/DPM Tunning:** This tool can be used to set up the “Detection of Plunger Movement” feature, and to calibrate the PI values for current control. The graph in this tool shows channel current measurements over a “Capture Period”. From here, the channel can be monitored and the settings tuned in the “Solenoid Sequencer”. The DPM feature must be set by running the DPM tool a couple of times. First time, just run it using the “Active Coil” button. This shows the output current graph. From there, the DPM_START should be set to a current value a bit under the “valley” of the DIP, and the DPM_THLD should be a value smaller than the “peak” – “valley” current value.




012

*Figure 12. Channel 3 > BEMF/DPM Tuning: Example DPM Setup and Tuned PI with a Bit of Overshoot*

## Detailed Description of Software

The TMCL-IDE is an easy-to-use graphical user interface (GUI) for the ADI Trinamic evaluation products, using the TMCL ecosystem.

Once the MAX22216EVKIT is detected, all the MAX22216 tools should appear under the MAX22216-EVAL yellow tab.

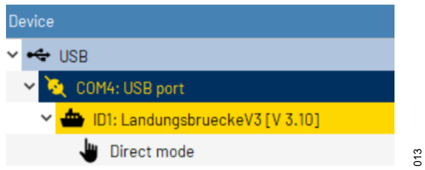



*Figure 13. MAX22216-EVAL Tools Tab*

Most MAX22216 features can be set using the GUI tools, but there are specific settings that must be set directly in the MAX22216 using the TMCL-IDE's "Register Browser".

### MAX22216 Tools Structure Overview

The MAX22216 tools are divided into system-level tools and channel-specific tools. Most system-level tools are not needed for the setup of MAX22216, but they are useful for debugging and verification of advanced setups.

#### System-Level Tools

Most of these tools are common for ADI Trinamic evaluation boards. To start using the MAX22216EVKIT, firstly, set the MAX22216 in the "General Settings".

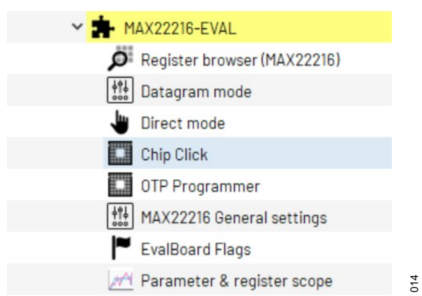



*Figure 14. MAX22216-EVAL System-Level Tools*

#### Channel-Specific Tools

These are MAX22216-specific settings for each channel. Based on the channel configuration (set in the "General Settings" tool), these can control both single-channel and full-bridge outputs, and these control most settings of the MAX22216.

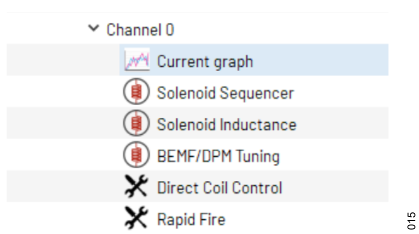
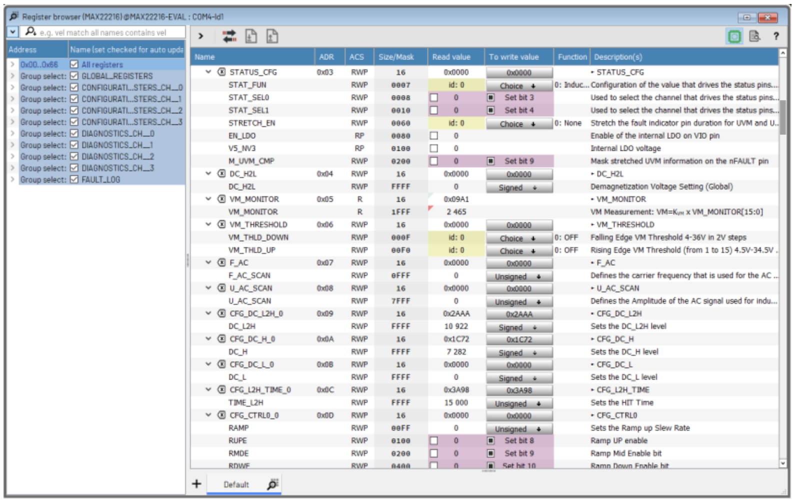



*Figure 15. MAX22216-EVAL Channel 0 Tools*

### Register Browser Tool

The TMCL-IDE has a simple yet powerful tool to gain direct access to the MAX22216 registers. It offers an arranged visual representation of the registers and their categories, and it allows direct register setup. It also allows real-time view of the register setups and read-only registers (like the VM or current of each channel).




016

*Figure 16. MAX22216-EVAL Register Browser Tool*

To control a register, edit the value in the “To write value” column. The actual register value can be seen in the “Read value” column.

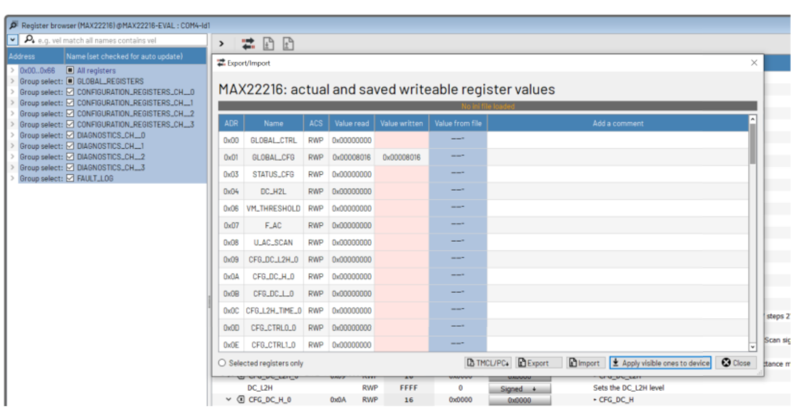
The registers are divided into specific categories:

- **Global Registers:** Registers that affect all the IC and all the channels. This includes IC activation, channels ON/OFF control, masking and stretching flags, status flags setup, global PWM frequency, channel configuration, voltage output reference, fast demagnetization level, VM monitor overview and threshold control, and sine wave generator voltage and frequency settings.
- **Configuration Registers** (for each channel): Channel-specific control and feature settings.
- **Diagnostics:** Channel-specific current, time, and duty cycle measurements used for different features like real-time current measurement, DPM current and time measurement, inductance measurement reactance and resistance measurement, and channel PWM duty cycle output
- **Fault Log:** All the fault flags and settable flags.

## Export/Import Sub-Tool

The settings can be exported as TPC values (format for settings in the TMCL-IDE), or directly as C values.

The tool can be accessed through the two arrows in the top grey bar of the “Register Browser”.




017

*Figure 17. Register Browser (Export/Import Sub-Tool)*

The “Active Register” set allows the export of only the registers opened in the “Register Browser”. If a full register export is needed, unselect this. The settings can also be sent directly to the TMCL/PC for use with the TMCL scripting.

## Chip Click Tool

The TMCL-IDE allows direct access to the pins of the MAX22216. The MAX22216 allows channel ON/OFF control through the CNTL_ pins, and there is an N_FAULT output pin that triggers when a fault/flag is detected by the MAX22216.

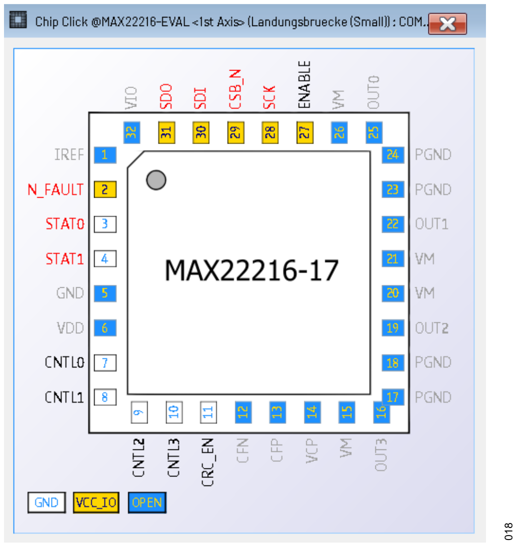



*Figure 18. Chip Click Tool*

There are also output STAT_ pins that can be connected to the STAT functions inside the MAX22216, and the output polarity can be changed from the GLOBAL_CFG (ADD 0x01 MASK 0040).

The MAX22216 has a lot of safety systems, one of them being the ENABLE pin. It must be connected to VCC_IO, or the communication with MAX22216 does not work. This is done automatically by the TMCL evaluation platform. The other safety requirement is the ACTIVE register (ADD 0x01 MASK 8000) to be set to HIGH.

## OTP Programmer Tool (Customer OTP)

The MAX22216 allows customer-programmable OTP using a simple interface. The OTP programming requires the +VM power supply to be 8.7V (± 0.13V). Each MAX22216 register can be OTPd only once!

The OTP programming tool works by selecting which register must be OTPd and the respective value. The OTP already shows the values from the “Register Browser”. However, these can be programmed differently in the “Value” column.

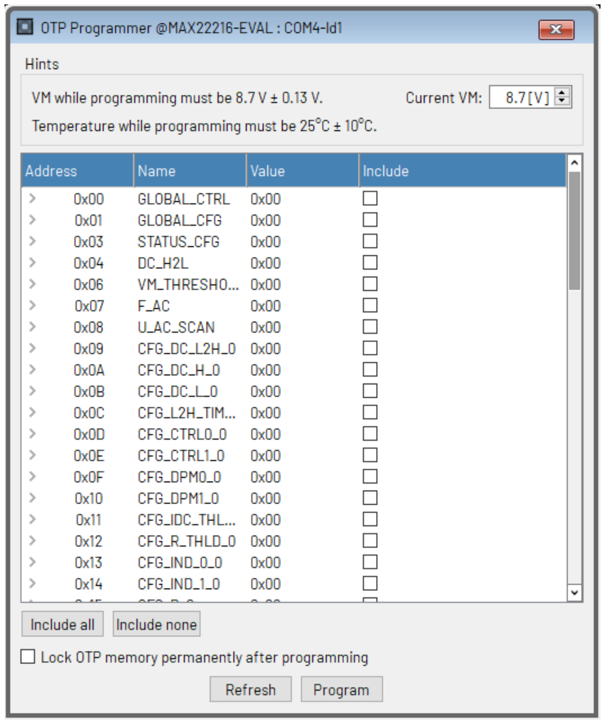



*Figure 19. OTP Programmer Tool*

The “Lock OTP memory” completely blocks the MAX22216 from using the OTP for any other registers.

## Parameter and Register Scope Tool (Mini Register Oscilloscope)

This tool is used for a visual response of the registers inside the MAX22216, for development and debugging.

It works by using the Landungsbruecke to read and store up to four registers for a set period, and downloading the stored data onto the PC and putting it into a graph.

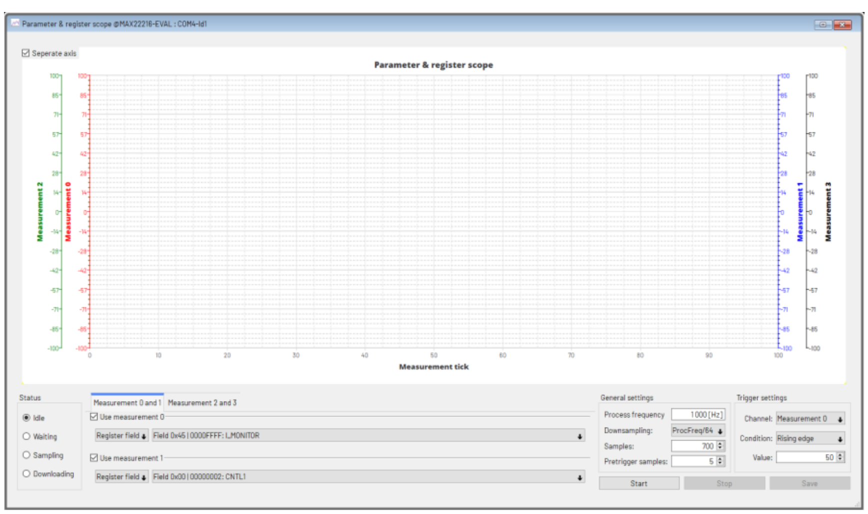



*Figure 20. Parameter and Register Scope Tool*

## MAX22216 System Setup (General Settings and EvalBoard Flags Tools)

The simplest way for initial system setup is to use the MAX22216 “General Settings” and “EvalBoard Flags” tools of the TMCL-IDE.

The following are the recommended start settings.

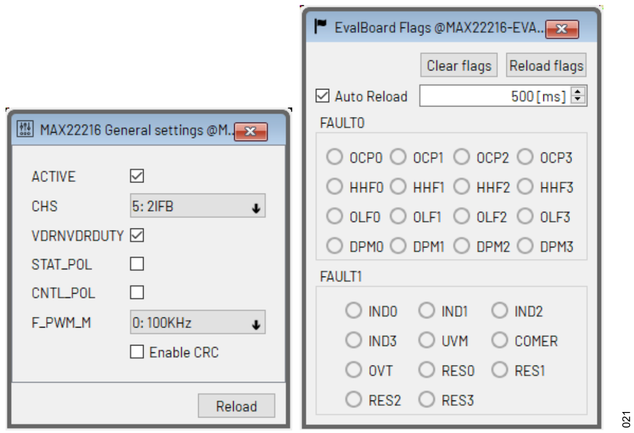



*Figure 21. General Settings and EvalBoard Flags Tools*

### EvalBoard Flags

At each power reset of the MAX22216, the UVM flag (register address: 0x66 mask: 0010) and sometimes the COMER flag (register address: 0x66 mask: 0020) are triggered. It is recommended to clean all flags using "Clear Flags" or from the registers before using the IC. The FLAGs registers can be found in FAULT0 (ADD 0x65) and FAULT1(0x66) registers inside the “Register Browser”, and they are cleared by setting them HIGH.

It is recommended to also set an “Auto Reload” (box check + time) for easier view of diagnostics and debugging in general. Example of good reload times are 500ms or 2000ms.

**FAULT0:**

- **OCPx:** Overcurrent protection flag for each channel. It stops the channel until the flag is reset.
- **HHFx:** This function can be set for each channel to trigger when the target set current is not reached. This works only in current drive regulation (CDR), and it can show a problem with the load.
- **OLFx:** Continuity check for each channel. If it triggers, it means the coil might be broken. For the full-bridge channel setup, it requires extra settings.
- **DPMx:** This is a channel-specific alarm for when the DPM is set but not detected.

**FAULT1:**

- **INDx:** For the inductance measurement function, it checks the channel measured value I_AC against a set value of IAC_THLD. This comparison is different in the DC_L2H and DC_H periods versus DC_L period.
- **RESx:** For the inductance measurement function, it checks the channel calculated value RES if it is bigger than a set value of RES_THLD.
- **UVM:** When VM is under 4.5V, it triggers the undervoltage alarm. This is triggered at every power reset, and it can be affected by the VM_THRESHOLD register (ADD 0x06).
- **COMER:** Communication error flag. It triggers when the SPI communication has a problem (the recommended SPI timings are not respected) or when the cyclic redundancy check (CRC) shows an error.
- **OVT:** Overtemperature alarm. It is triggered when the internals of the MAX22216 are too hot. It stops all channels until the core goes back to an acceptable temperature range.

### MAX22216 General Settings

ACTIVE: At every startup of the MAX22216, the first thing to do is to set the "ACTIVE" bit to HIGH! (register address: 0x01, mask: 8000).

CHS: Set the channel configuration from the CHS dropdown box (register address: 0x01, mask: 000F).

VDRnVDRDUTY: If the chosen channel control mode (CTRL_MODE) has voltage control, and the internally compensated voltage reference scheme is planned to be used, the VDRnVDRDUTY register should be set to HIGH (register address: 0x01, mask: 0010). With this, the output uses the internal 36V reference to calculate the output instead of the VM measurement. If this register is not used, the VM noise might be transferred to the output signal. The default setting is 4IHB (four independent half-bridges).

STAT_POL: Allows for the reverse of the STAT pin output.

CNTL_POL: Reverses the input control of the CNTL pins. This is recommended only in standalone mode, and the settings should have OTP.

F_PWM_M: Set the global PWM frequency (default/maximum 100kHz). It controls a lot of systems inside the MAX22216. At the same time, each channel has a separate PWM divider that can control the final output PWM.

Enable CRC: The MAX22216 has a cyclic redundancy check (CRC) with 5-bit frame check sequence (FCS). If selected, the TMCL evaluation platform calculates the response and checks communications. If an error occurs, the COMER flag is triggered.

### CHS Configuration

The following tables show the possible output configuration of the MAX22216 and register CNTL register groups that set up each channel (single-ended or full-bridge).

## Table 1. CHS Configuration (Control) (from the MAX22216 Data Sheet)

| **OUTPUT SETTINGS** | | | **ACTIVE CONFIGURATION REGISTER/CONTROL CHANNEL** | | | |
|---|---|---|---|---|---|---|
| **CHS[3:0]** | **OUTPUT CONFIGURATION** | | **CH0 OUTPUT** | **CH1 OUTPUT** | **CH2 OUTPUT** | **CH3 OUTPUT** |
| 0x0 | 4x Independent HBs | | CFG_0/CNTL0 | CFG_1/CNTL1 | CFG_2/CNTL2 | CFG_3/CNTL3 |
| 0x1 | 3x Parallel HBs | 1x Independent HB | CFG_0/CNTL0 | | | CFG_3/CNTL3 |
| 0x2 | 2x Parallel HBs | 2x Independent HBs | CFG_0/CNTL0 | | CFG_2/CNTL2 | CFG_3/CNTL3 |
| 0x3 | 2x Parallel HBs | 2x Parallel HBs | CFG_0/CNTL0 | | CFG_2/CNTL2 | |
| 0x4 | 4x Parallel HBs | | CFG_0/CNTL0 | | | |
| 0x5 | 1x Independent FB | 1x Independent FB | See *Table 2* | | See *Table 2* | |
| 0x6 | 1x Independent FB | 2x Independent HBs | See *Table 2* | | CFG_2/CNTL2 | CFG_3/CNTL3 |
| 0x7 | 1x Independent FB | 2x Parallel HBs | See *Table 2* | | CFG_2/CNTL2 | |
| 0x8 | 1x Parallel FB | | See *Table 2* | | | |

## Table 2. CHS Configuration (Full-Bridge Control) (from the MAX22216 Data Sheet)

| CHS[3:0] | BRIDGE CFG | CNTLx | CNTLy | OUTx | OUTy | FB STATUS |
|---|---|---|---|---|---|---|
| 0 x 05 | OUTx vs. OUTy<br>(x, y) = (0, 1) or (2, 3) | 0 | 0 | HiZ | HiZ | HiZ |
| | | 1 | 0 | CFG_x | | DRIVEN BY CHx |
| | | 0 | 1 | CFG_y | | DRIVEN BY CHy |
| | | 1 | 1 | 50% PWM | 50% PWM | BRAKE |
| **CHS[3:0]** | **BRIDGE CFG** | **CNTL0** | **CNTL1** | **OUT0** | **OUT1** | **FB STATUS** |
| 0x 06 or 0x 07 | OUT0 vs. OUT1 | 0 | 0 | HiZ | Hiz | HiZ |
| | | 1 | 0 | CFG_0 | | DRIVEN BY CH0 |
| | | 0 | 1 | CFG_1 | | DRIVEN BY CH1 |
| | | 1 | 1 | 50% PWM | 50% PWM | BRAKE |
| **CHS[3:0]** | **BRIDGE CFG** | **CNTL0** | **CNTL1** | **OUT0 = OUT1** | **OUT2 = OUT3** | **FB STATUS** |
| 0x 08 | OUT0 vs. OUT1 = OUT2 vs. OUT3 | 0 | 0 | HiZ | HiZ | HiZ |
| | | 1 | 0 | CFG_0 | | DRIVEN BY CH0 |
| | | 0 | 1 | CFG_1 | | DRIVEN BY CH1 |
| | | 1 | 1 | 50% PWM | 50% PWM | BRAKE |

Settings explanation:

- Single-ended configurations half-bridge (HB): Each channel is controlled by its specific registers.
- Full-bridge configurations full-bridge ((FB): From *Table 2*, the outputs are controlled by either of the channel's settings, depending on how the channel is turned ON. While using either channel control, the polarity of the output is set to positive on the smaller channel number and negative on the bigger channel number (for single full-bridge modes: CH0+/CH1- and CH2+/CH3-, and parallel full-bridge mode: CH0CH1+/CH2CH3-).

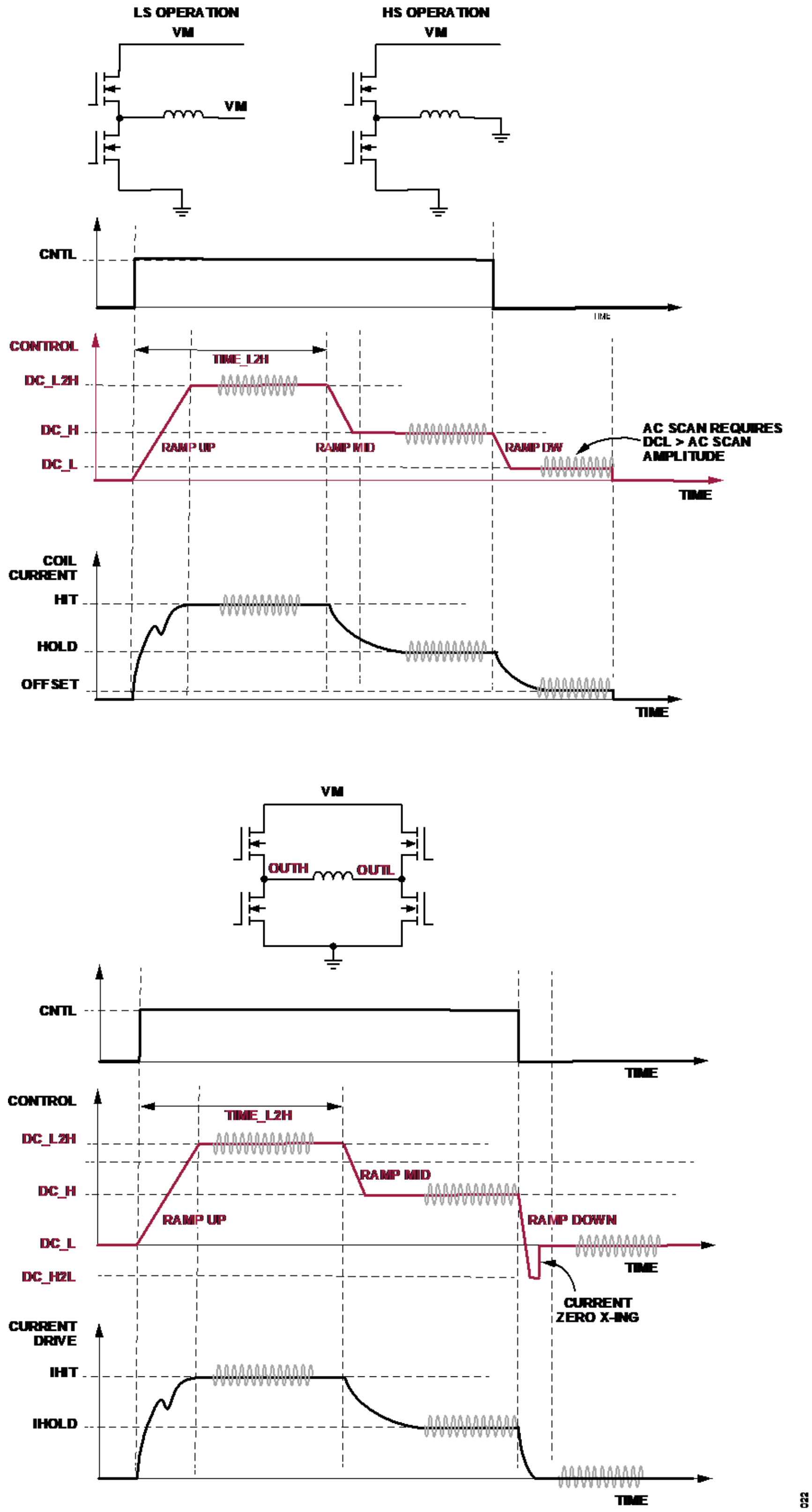



*Figure 22. CHS Single-Ended and Full-Bridge Configurations*

Example: CH0CH1 (full-bridge CHS 0x5)

- Output using CNTL0 (ADD 0x00 MASK 0001): MAX22216 uses all the settings from the registers related to this CNTL channel (addresses 0x9 to 0x16).
- Output using CNTL1 (ADD 0x00 MASK 0002): MAX22216 uses all the settings from the registers related to this CNTL channel (addresses 0x17 to 0x24).

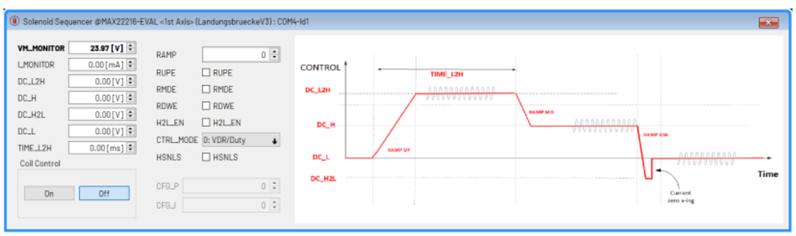
### Recommended System Setup

A recommended startup setup is to set the ACTIVE bit HIGH and the VDRnVDRDUTY bit HIGH.

The CHS and F_PWM_M should be set depending on the need (by default set to four independent channels and 100kHz), and it is mentioned in the following examples if different from the default, or to accentuate a point.

## Solenoid Sequencer Tool (Channel Settings)

Firstly select the “Solenoid Sequencer” of the specific channel (if using a half-bridge configuration) or of one of the channels of the bridge (if using a full-bridge configuration).




023

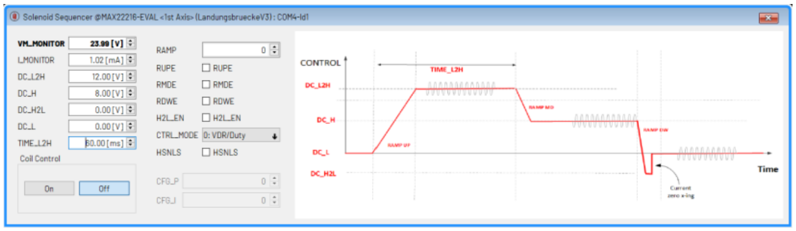
*Figure 23. Solenoid Sequencer Tool*

The graph describes most of the values that can be set, but the current zero x-ing (fast demagnetization) can be used only in full-bridge modes (CHS). DC_L2H and TIME_L2H control the level and duration of the first level (HIT period - opening of the solenoid) and DC_H controls the HOLD level until the channel is closed.

This control methodology can be seen in the graph provided in the tool, and it allows for huge energy savings and extends the life of the coil (except in motor control mode). There is an option to only set DC_H (no DC_L2H or TIME_L2H) for a constant output.

- **VM_MONITOR:** It shows the VM voltage, measured by the MAX22216 (not like the IDE voltage monitor measured by the Landungsbruecke board). In normal DUTY mode, it is used as the reference for the output.
- **I_MONITOR:** Monitors the channel output current.
- **Coil Control:** Turn the channel ON or OFF using the on and off buttons.
- **RAMP, RUPE, RMDE, RDWE:** Set the RAMP limiting slew rate and the section it applies to. These settings slow down the solenoid movement and can make it more silent.
- **DC_H2L, H2L_EN:** (Only full-bridge) fast demagnetization. Set the global value (negative) and enable it for this channel. This is used to force a faster closure of the solenoid when the channel is turned OFF.
- **CTRL_MODE:** This is the specific control setting for the channel. Voltage drive regulation (VDR) (default) controls just the output PWM and current drive regulation (CDR) controls the output based on the output current measurements, Motor control mode sets a fixed DC_H output voltage and limits the current to the DC_L2H.
- **CFG_P, CFG_I:** These are the values used by the PI current control engine. In CDR, these are used for the current output regulation and in motor control mode are used by the current limiter.
- **HSnLS:** High-side not low-side. In single ended, by default, the MAX22216 controls the output using the low-side MOSFET, where the output current is also measured. If set to HIGH, the output happens using the high-side MOSFET, which has limitations. It is also used by the open-load detection in the full-bridge configuration. It is not a recommended mode of control.

### CTRL_MODE 00: VDR (Voltage Drive Regulation)




024

*Figure 24. Solenoid Sequencer VDR Example*

The VDR mode has two control methods discussed in "System Setup" (VDRnVDRDUTY - ADD 0x01 MASK 8000):

- Duty (default: VDRnVDRDUTY LOW): The output voltage is calculated as a percentage/pulse width (or duty) of the input voltage. The formula is: $V_{OUT}$ (V) = KVDR x VM x DC_X [15:0]DEC.
- Internally compensated VDR (VDRnVDRDUTY HIGH): The output voltage is calculated based on an internal 36V reference. The formula is: $V_{OUT}$ (V) = KVDR x 36 x DC_X [15:0]DEC.

There is also the possibility to set only DC_H (no DC_L2H or TIME_L2H) for a constant voltage output.

#### VDR Example: CH0 Single Channel/VDR (VDRnVDRDUTY)/Two-Level Control

In this example, a solenoid is connected between Channel 0 and the COM line (connected to +VM) and controlled using VDR. In single-ended mode, the output is set by default to use the low-side MOSFET. In the VDR two-level control scheme, both the HIT and HOLD levels are set as voltages.

## Table 3. VDR Example Register Table

| NAME | ADDRESS | MASK | VALUE | COMMENT |
|---|---|---|---|---|
| ACTIVE | 0x01 | 8000 | HIGH | Turn ON the MAX22216 |
| CHS | 0x01 | 000F | 0x00 | 4x independent half bridges |
| COMER | 0x66 | 0020 | HIGH | Clear bit |
| CTRL_MODE | 0x0D | C000 | 0x0 | VDR |
| DC_H | 0x0A | FFFF | 8191 | 9V |
| DC_L2H | 0x09 | FFFF | 10922 | 12V |
| TIME_L2H | 0x0C | FFFF | 6000 | 60ms |
| UVM | 0x66 | 0010 | HIGH | Clear bit |
| VDRnVDRDUTY | 0x01 | 0010 | HIGH | |

- DC_L2H and DC_H (channel specific - CONFIGURATION_REGISTERS_CH_X > CFG_DC_L2H/H_X): In VDRnVDRDUTY (internally compensated with 36V reference), the output HIT and HOLD voltage are calculated using the following formula: $V_{OUT}$ (V) = KVDR x 36 x DC_xxx [15:0]DEC, where KVDR = 30.518 x 10^(−6) or 1/32767 → DC_ [15:0]DEC = ($V_{OUT}$/36) x 32767.
- TIME_L2H (channel specific - CONFIGURATION_REGISTERS_CH_X > CFG_L2H_TIME_X): Defines the period (in ms) the DC_L2H HIT level is kept, every time the channel starts. The formula is: TIME_L2H(s) = TIME_L2H[15:0]DEC/(F_PWM_M (Hz) x F_PWM). F_PWM_M (register address: 0x00, mask: 00F0) is the global clock frequency (default: 100kHz) and F_PWM (register address: 0x0E, mask: 0300 for CH0) is a channel-specific clock divider (default is 1) → TIME_L2H[15:0]DEC = TIME_L2H(s) x (F_PWM_M (Hz) x F_PWM).

The output DUTY can be monitored from the PWM_DUTYCYCLE in the diagnostics (register address: 0x49, mask: FFFF for CH0). This can be used to calculate the voltage output (disregarding the loss over the output MOSFETs).

### CTRL_MODE 01: CDR (Current Drive Regulation)

For a constant output, only DC_H needs to be set (as shown in this example), but some of the functions that are dependent on the DC_L2H level (like the “Detection of Plunger Movement”) cannot be used. The channel can be turned ON or OFF using “Coil Control”.

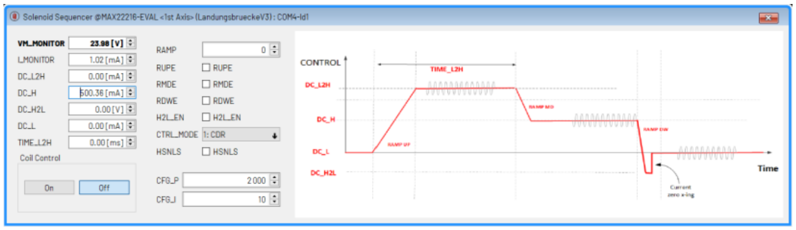



*Figure 25. Solenoid Sequencer CDR Setup Example*

As CDR controls the output current using a closed-loop system, it can also be used as a constant torque control for DC motors!

### CDR Example: CH0CH1 Full-Bridge/CDR/One-Level Control

In this example, a solenoid is connected between Channel 0 and Channel 1, and it is controlled using a constant current.

## Table 4. CDR Example Register Table

| NAME | ADDRESS | MASK | VALUE | COMMENT |
|---|---|---|---|---|
| ACTIVE | 0x01 | 8000 | HIGH | Turn ON the MAX22216 |
| CHS | 0x01 | 000F | 0x05 | 2x independent full-bridges (CH0CH1 and CH2CH3) |
| UVM | 0x66 | 0010 | HIGH | Clear bit |
| COMER | 0x66 | 0020 | HIGH | Clear bit |
| CTRL_MODE | 0x0D | C000 | 0x1 | CDR |
| DC_H | 0x0A | FFFF | 492 | 500mA |
| CFG_P | 0x15 | FFFF | 2000 | |
| CFG_I | 0x16 | FFFF | 10 | |

- DC_H (channel specific - CONFIGURATION_REGISTERS_CH_X > CFG_DC_H_X): The output current formula: $I_{OUT}$ (mA) = KCDR x GAIN x SNSF x DC_[15:0]DEC, where KCDR = 1.017. GAIN and SNSF are current monitor dividers (used to scale the currents for more accurate monitoring). The scaling is explained in the CDR of the data sheet, but for a quick setup, the default values are used (default value: 1), which means the maximum current is 3.2A. The values can also be calculated using the simplified formula → DC_[15:0]DEC = $I_{OUT}$ (mA)/KCDR.
- CFG_P/CFG_I (channel specific - CONFIGURATION_REGISTERS_CH_X > CFG_P/I_X): The P and I parameters of the PI that control the output current. These are load-specific and very dependent on the inductance and capacitance of the system. The example numbers are from random valve.

If the two-level control is needed, set DC_L2H and TIME_L2H (as shown in the VDR example) to the needed values, but this also requires new P and I values to work with both the DC_L2H and DC_H currents.

The current of the CH0CH1 full-bridge can be monitored in the I_MONITOR of the CH0 (register address: 0x45, mask: FFFF).

The channel can be controlled ON and OFF from the CNTL0 register (register address: 0x00, mask: 0001) to use the settings from CH0 registers, and CNTL1 register (register address: 0x00, mask: 0002) to use the settings from CH1 registers. Independent of the settings, the positive output is at CH0 and negative output at CH1.

Depending on which CNTL is used, these settings are applied to the whole full-bridge. If both channels CNTL0 and CNTL1 are turned ON, the CH0CH1 full-bridge enters a high Z state.

In full-bridge mode, both in VDR and CDR, the output can be set with a negative value, and the DC_L2H and DC_H are registers with the signed format of 2's complements.

## PI Tuning

The P and I parameters are in the Q8.8 format, which means that a digital value of 256 is equivalent to a multiplication of 1 for the P and I variables in the MAX22216 PI current control loop.

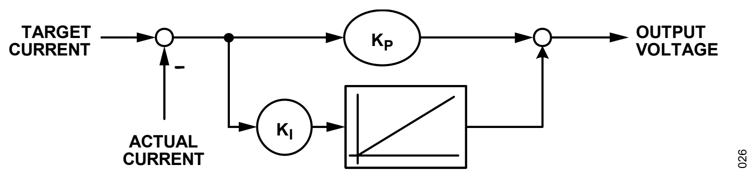



*Figure 26. MAX22216 PI Current Control Loop (from Data Sheet)*

Set DC_L2H, DC_H and TIME_L2H according to manufacturer specifications.

Set P and I to a multiplier of 1 (digital value 256).

Increase P until the current graph curve is critically damped or a bit underdamped (it goes nicely to the set target or it has a bit of overshooting).

Change I to tune the undershooting/overshooting.

To decrease the solenoid opening time and overcome any mechanical force required to open the solenoid, set the final PI to overshoot a little bit.

## HHF ("Hit Current Not Reached" Functionality)

The MAX22216 has a tool and flag for checking if the DC_L2H current level is reached. The HHF_EN can be found in the "Register Browser" under CONFIGURATION_REGISTERS_CH_X > CFG_CTRL0_X > MASK 2000, and the only setting needed is to turn it HIGH. If the DC_L2H current level is not reached, the channel HHF is triggered.

## MIN_T_ON (CDR Single-Ended Limitation)

In single-ended CDR, the current controller has an output blanking time by default, with extra settings that can affect it (like T_BLANKING and SRC slew rate).

This minimal on-time limits the voltage output (controlled by the current PI regulator). So, depending on the configuration (the default low-side or the high-side), there is a minimum or maximum voltage that must be on the output line for the CDR to function properly.

The CDR minimum ON time is calculated using the following formula: MIN_T_ON (s) = (MIN_TON_SINGLE_ENDED + 2SLEW_RATE[1:0] + 1 x T_BLANKING) x (2/CLK_FREQ (Hz)).

- T_BLANKING (channel specific - CONFIGURATION_REGISTERS_CH_X > CFG_CTRL1_X MASK 00C0): Adds a shift/blanking time.
- SLEW_RATE (channel specific - CONFIGURATION_REGISTERS_CH_X > CFG_CTRL1_X MASK 0030): Controls the PWM shape to lower EMI.
- MIN_TON_SINGLE_ENDED = 15 and CLK_FREQ = 25MHz. SLEW_RATE bit value is based on the settings and T_BLANKING is a value that can also be set using the following table.

## Table 5. Blanking Time Values

| T_BLANKING_[1:0] | T_BLANKING VALUE |
|---|---|
| 00 | 0 |
| 01 | 24 |
| 10 | 48 |
| 11 | 96 |

This MIN_T_ON time is used to calculate the actual minimum/maximum voltage using the following formulas:

- Single-ended LS configuration: $V_{MIN}$ (V) = VM (V) x MIN_T_ON (s) x F_PWM (Hz)
- Single-ended HS configuration: $V_{MAX}$ (V) = VM (V) x (1 - MIN_T_ON (s) x F_PWM (Hz))

where, F_PWM (Hz) is F_PWM_M (global PWM frequency - ADD 0x00 MASK 00F0 ) x F_PWM (channel-specific frequency divider - CONFIGURATION_REGISTERS_CH_X > CFG_CTRL1_X MASK 0300).

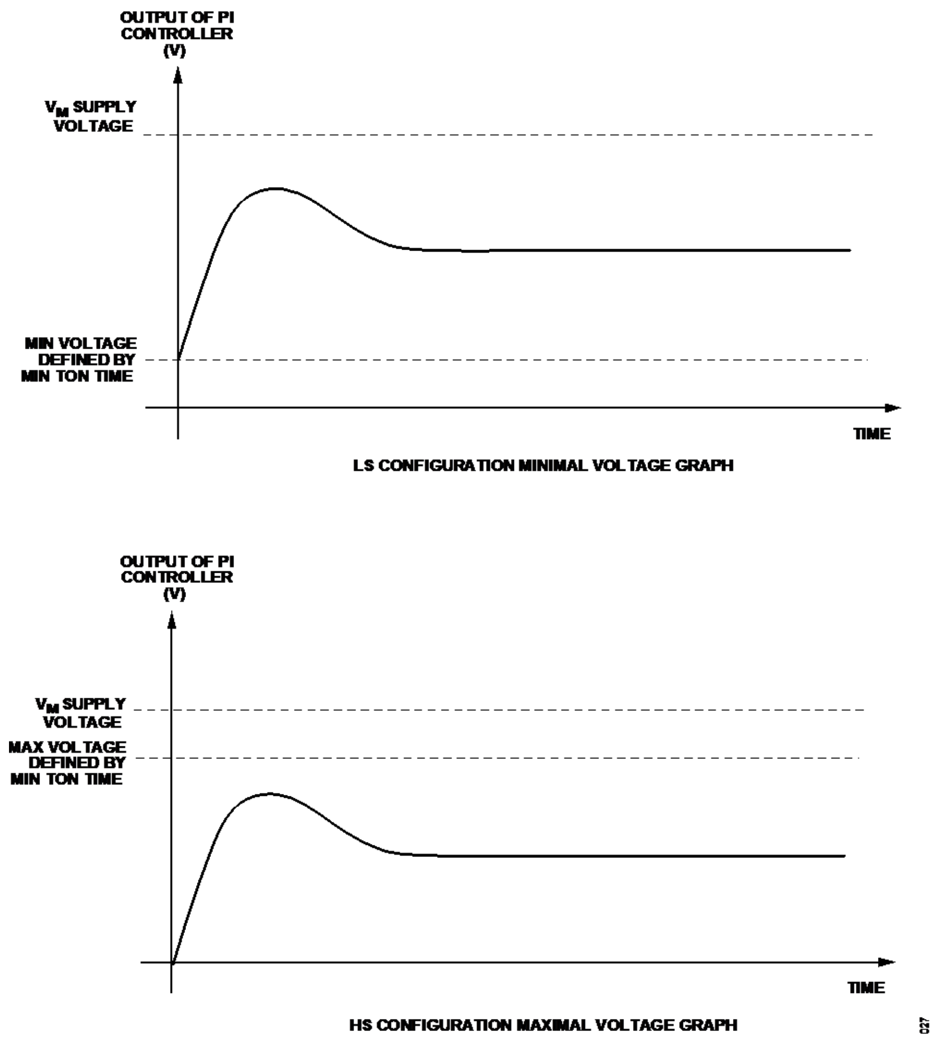



*Figure 27. Minimal/Maximal Voltage Graph for LS/HS Configurations (from Data Sheet)*

The output to control the solenoid must be calculated using this minimal output voltage, power supply voltage, coil resistance (and in some situations impedance), and the minimum/maximum channel current.

### CTRL_MODE 10: CDR/VDR or Current Limiter or Motor Control Mode

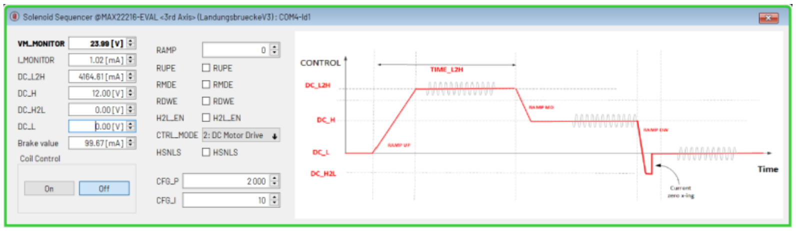



*Figure 28. Solenoid Sequencer DC Motor Drive Setup Example*

It is recommended to use the DC motor mode in full-bridge configuration (otherwise, the BRAKE does not function), but it is technically possible to use the single-ended configuration.

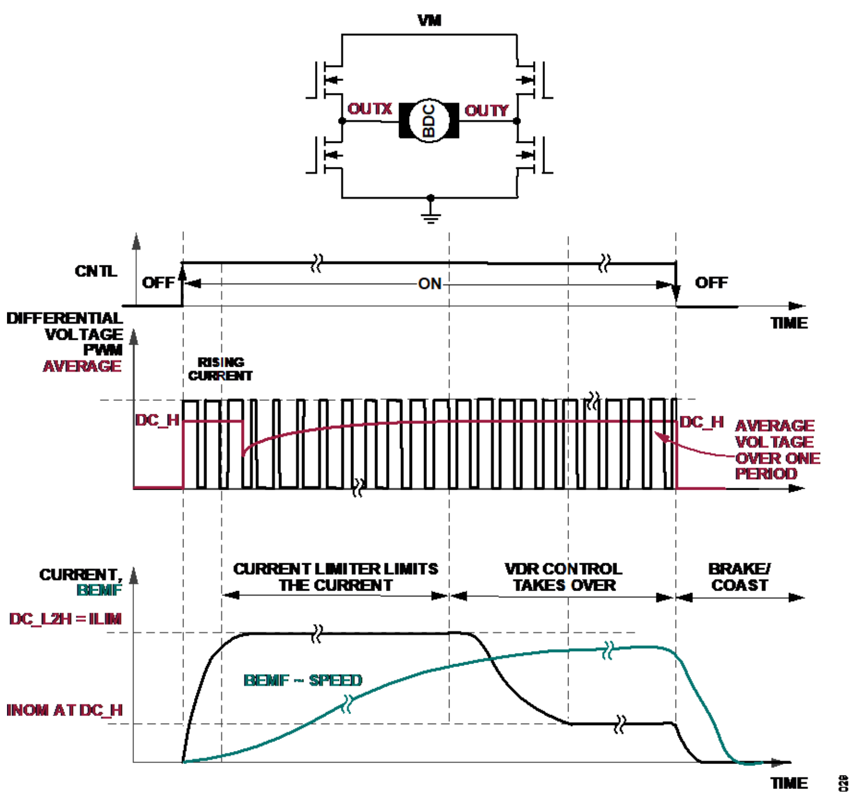



*Figure 29. CHS Full-Bridge Configuration for CNTL_MODE DC Motor Mode (from Data Sheet)*

### DC Motor Mode Example: CH2CH3 (H-Bridge Motor Control)

In this example, the motor is connected between Channel 2 and Channel 3, and the motor mode does not have a HIT and HOLD period. This control mode works by setting an output voltage, and in parallel setting a current regulation (PI regulated) triggered if the output current goes over the set limit. If the current is under the set limit, the voltage control is unaffected. This mode also has some special functions and requires the motor to be set and controlled in the full-bridge configuration.

The BRAKE function of the DC motor control mode works by creating negative current (set in the TIME_L2H register) when both channels of the full-bridge are set to output. After the motor stops, the output goes to the normal HIGH_Z state.

## Table 6. DC Motor Mode Example Register Table

| NAME | ADDRESS | MASK | VALUE | COMMENT |
|---|---|---|---|---|
| ACTIVE | 0x01 | 8000 | HIGH | Turn ON the MAX22216 |
| CHS | 0x01 | 000F | 0x05 | 2x independent full-bridges (CH0CH1 and Ch2CH3) |
| UVM | 0x66 | 0010 | HIGH | Clear bit |
| COMER | 0x66 | 0020 | HIGH | Clear bit |
| CTRL_MODE | 0x0D | C000 | 0x2 | DC Motor Drive |
| DC_L2H | 0x09 | FFFF | 295 | 300mA |
| DC_H | 0x0A | FFFF | 10922 | 12V |
| TIME_L2H | 0x0C | FFFF | 98 | 100mA |
| CFG_P | 0x15 | FFFF | 2000 | |
| CFG_I | 0x16 | FFFF | 10 | |

- DC_L2H (channel specific - CONFIGURATION_REGISTERS_CH_X > CFG_DC_L2H_X): Current limiter (CDR) for the start of the motor, the control scheme stays in this mode until the current is reached. DC_L2H is calculated based on the current formula presented in the CDR section.
- DC_H (channel specific - CONFIGURATION_REGISTERS_CH_X > CFG_DC_H_X): After the current is reached, the IC changes the control mode to VDR. DC_H is set based on the voltage formula presented in the VDR section (and it is influenced by the VDR-VDRDUTY).
- TIME_L2H (channel specific - CONFIGURATION_REGISTERS_CH_X > CFG_L2H_TIME_X): Brake current that can be used to stop the motor motion, triggered when both CNTL channels of the full-bridge are high (in the other modes is high Z). TIME_L2H is calculated based on the DC_L2H and DC_H current formula from CDR.
- CFG_P/CFG_I (channel specific -CONFIGURATION_REGISTERS_CH_X > CFG_P_X / CFG_I_X): The P and I parameters of the PI that control the output current. These are load-specific and very dependent on the inductance and capacitance of the system. The example numbers are from a general solenoid and they also work for a motor.

### Fast Demagnetization

This control mode can only be used in channel full-bridge configurations (CHS 0x5 to 0x8). There is always a negative current when the solenoid control is turned OFF, as a response to the plunger moving into the unengaged position.

DC_H2L is a setting that forces a negative voltage to the direction of control of the full-bridge channel when the channel is turned OFF (and the current passes the zero x-crossing level), so the coil is demagnetized faster than the default discharge time and completes the closing movement of the plunger faster. The DC_H2L value must be negative.

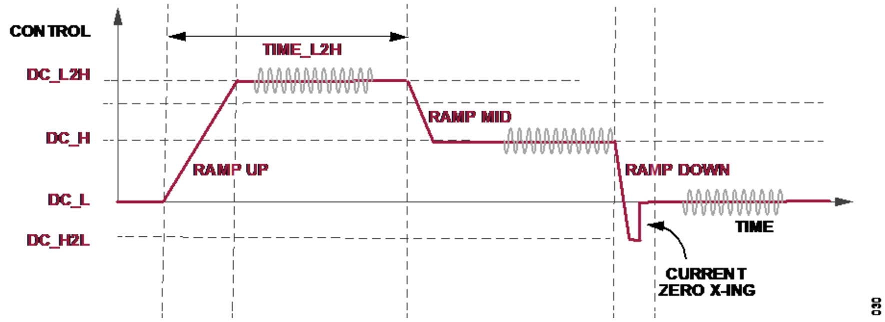



*Figure 30. MAX22216 Full-Bridge Control Graph with Current Zero X-ing (from Data Sheet)*

- DC_H2L (global variable - ADD 0x04 MASK FFFF): Voltage level used for fast demagnetization. The formula is: $V_{OUT}$ (V) = KVDR x 36 x DC_H2L [15:0]DEC/$V_{OUT}$ (V) = KVDR x VM x DC_H2L [15:0]DEC.
- H2L_EN (channel specific - CONFIGURATION_REGISTERS_CH_X > CFG_CTRL_0_X MASK 0800): Set HIGH to use fast demagnetization on the specific channel (the main channel of the full bridge configuration) - using the global voltage level set by DC_H2L.

The DC_H2L must be set as a negative voltage! The set value depends on the coil characteristics and the coil driving current, but the voltage consumed at DC_H can be a starting point.

### RAMP Control

The MAX22216 is capable of limiting the transition between different DC_ levels. This tool is mostly used to lower the acoustic noise of the solenoid, and it is intended in applications where the solenoid is near humans.

For each channel, the maximum RAMP slew rate can be set, and then selected where it should be applied (RAMP UP, RAMP MD, and/or RAMP DW). RAMP control works in both single-ended and full-bridge configurations.

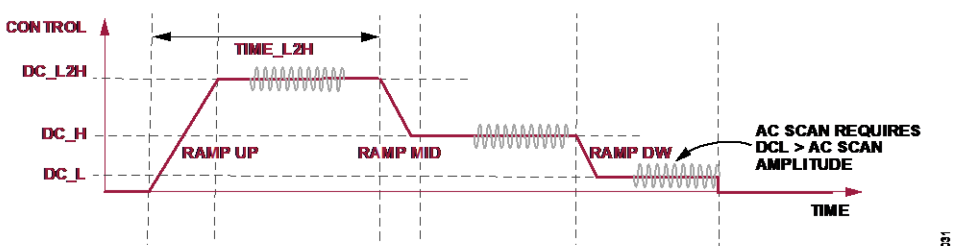



*Figure 31. MAX22216 Control Graph (from Data Sheet)*

- RAMP (channel specific - CONFIGURATION_REGISTERS_CH_X > CFG_CTRL_0_X MASK 00FF): This is the slew rate limit value. The formula is: VDRDUTY: ramp slew rate (V/ms) = KVDR x VM x (RAMP[7:0]DEC + 1) x F_PWM (kHz))/VDR: ramp slew rate (V/ms) = KVDR x 36 x (RAMP[7:0]DEC + 1) x F_PWM (kHz)/CDR: ramp slew rate (mA/ms) = KCDR x GAIN x SNSF x (RAMP[7:0]DEC + 1) x F_PWM (kHz)
- RAMP_UP/RAMP_MD/RAMP_DW (channel specific - CONFIGURATION_REGISTERS_CH_X > CFG_CTRL_0_X MASK 0100 / 0200 / 0400): Set HIGH to select the regions where RAMP control is applied.

To show the DPM tool, the example is set to control the solenoid in full-bridge CH0CH1 VDR configuration, and the example solenoid is set for a big difference between the HIT (DC_L2H) and HOLD (DC_H) periods.

The RAMP control can be checked using the “Parameter & Register Scope” tool. The example is set to show the RAMP control in VDR over CH0CH1 and the graphs shows the PWM duty cycle in BLUE and channel current in RED.

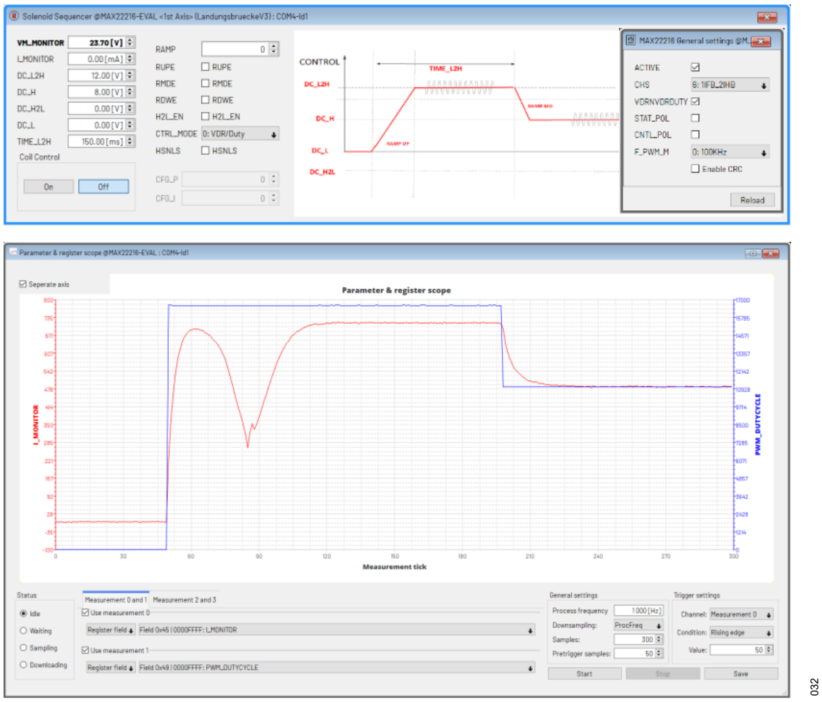


Figure 32. VDR Output with No Ramp Control Example

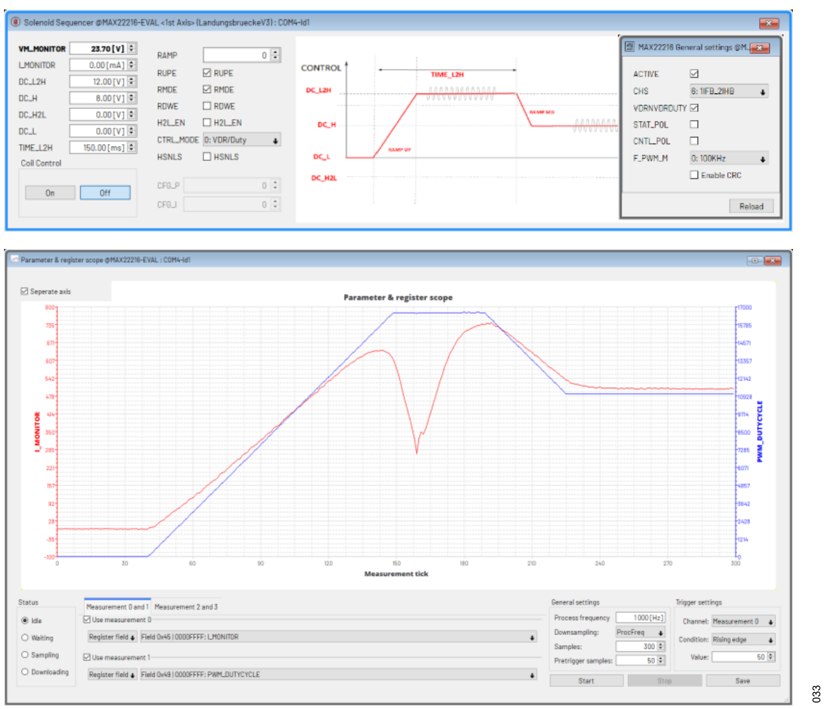


Figure 33. VDR Output with Extreme Ramp Control on Ramp UP and Ramp MID Example

### BEMF/DPM Tuning Tool (Detection of Plunger Movement)

The detection of plunger movement (DPM) can be used to confirm the plunger movement and detect the characteristics of the plunger movement for solenoid/valve diagnostics and predictive maintenance, and has a specific energy-saving mode.

This diagnostics tool functions by controlling the valve in VDR with two-level control (DC_L2H, DC_H, and TIME_L2H), and measuring the current changes. The creation of the "valley" in the current consumption of the valve represents the actual valve movement.

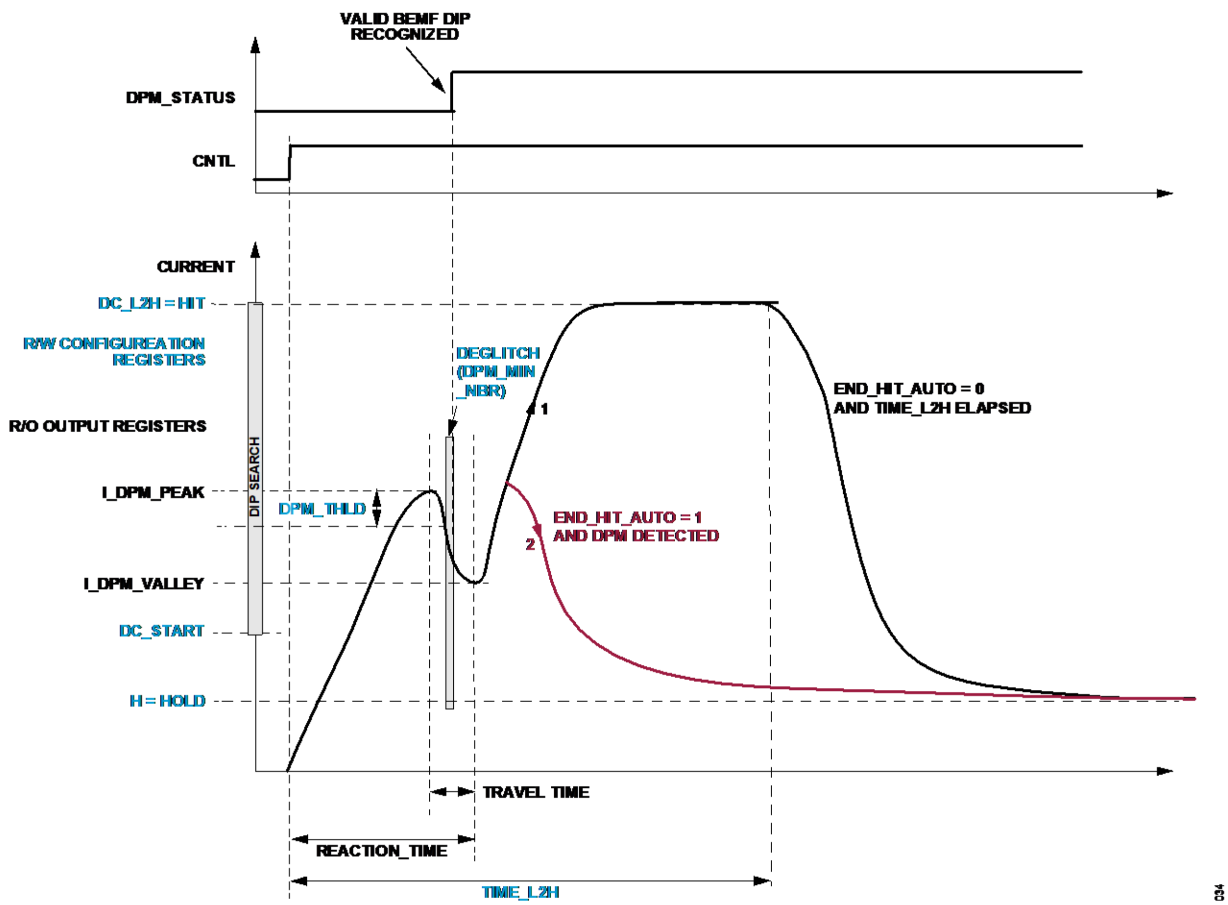


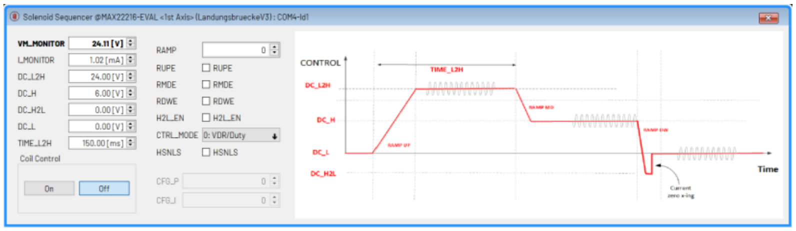

*Figure 34. Detection of Plunger Movement (from Data Sheet)*

DPM_START sets up the DIP SEARCH region (between the DPM_START current and the current consumption achieved when the valve reaches DC_L2H voltage), and the DPM_THLD (minimum valley dip threshold to filter noise) and DPM_MIN_NBR (repeated valley detection/period of detection for deglitching) are used as protections/filters for detection conditions.

### BEMF/DPM Tuning Tool

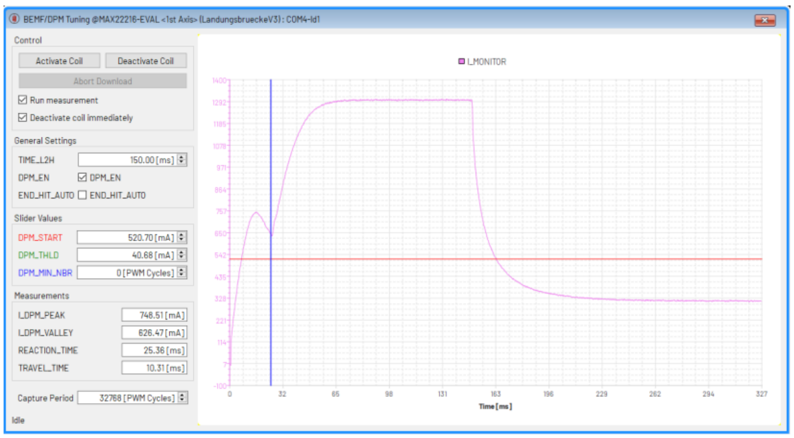
Firstly, select the BEMF/DPM tuning of the used channel (if using a half-bridge configuration), or if using a full bridge configuration, the main current measurement channel (usually the channel with a smaller number).

When opening the BEMF/DPM tuning tool for the first time, there is a pop-up message.




035

*Figure 35. VDR Solenoid Sequencer Settings Example for Use with the DPM Tool*




036

*Figure 36. BEMF/DPM Tuning Tool (DPM Setup Example)*

By default, the tool is set to “Run measurement” and “Deactivate coil immediately”. The “Run measurement” uses the “Parameter & Register Scope” to capture the channel current for the set “Capture Period” after each “Activate Coil” button press.

After each run, the tool must download the MAX22216 measurements from the Landungsbruecke board, where they are stored. “Deactivate coil immediately” automatically sets the solenoid channel OFF after the measurement period ends. The BEMF/DPM tuning measurement cannot run at the same time as the “Parameter & Register Scope”.

After each successful DPM, the BEMF/DPM tuning tool also reads the DPM measurements and displays the DPM I_DPM_VALLEY position in time with a blue line on the graph. The red line represents the DPM_START current and should be set under the current value of the DPM valley.

### DPM Example: CH0CH1 Full-Bridge/VDR - VDRnVDRDUTY/Two-Level Control/DPM

In this example, the solenoid is connected between Channel 0 and Channel 1, and it is driven using “Voltage Drive Regulation” and two-level control. The “Detection of Plunger Movement” can be used only in VDR mode. The following is a recommended way to set it.

**Observation:** TIME_L2H must be big enough to allow the complete DPM_VALLEY to happen during the plunger movement. Otherwise, the DPM flag is triggered.

## Table 7. DPM Example Register Table

| NAME | ADDRESS | MASK | VALUE | COMMENT |
|---|---|---|---|---|
| ACTIVE | 0x01 | 8000 | HIGH | Turn on the MAX22216 |
| CHS | 0x01 | 000F | 0x05 | 2x independent full-bridges (CH0CH1 and Ch2CH3) |
| VDRnVDRDUTY | 0x01 | 0010 | HIGH | |
| UVM | 0x66 | 0010 | HIGH | Clear bit |
| COMER | 0x66 | 0020 | HIGH | Clear bit |
| CTRL_MODE | 0x0D | C000 | 0x0 | VDR |
| DC_L2H | 0x09 | FFFF | 21845 | 24V |
| DC_H | 0x0A | FFFF | 5461 | 6V |
| TIME_L2H | 0x0C | FFFF | 15000 | 150ms |
| DPM_EN | 0x10 | 4000 | 1 | Enable DPM |
| DPM_START | 0x10 | 00FF | 8 | 520.70mA |
| DPM_THLD | 0x0F | 0FFF | 5 | 40.68mA |
| DPM_MIN_NBR | 0x10 | 0F00 | 0 | Not needed in this case |

- DC_L2H and DC_H (channel specific -CONFIGURATION_REGISTERS_CH_X > CFG_DC_L2H_X): In VDRnVDRDUTY (internally compensated with 36V reference), the output voltage can be calculated using the following formula: $V_{OUT}$ (V) = KVDR x 36 x DC_ [15:0]DEC, where KVDR = 30.518 x 10^(-6) OR 1/32767 → DC_ [15:0]DEC = ($V_{OUT}$/36) x 32767.
- TIME_L2H (channel specific -CONFIGURATION_REGISTERS_CH_X > CFG_L2H_TIME_X): defines the period (in ms) when the DC_L2H level is kept, every time the channel starts. The formula is: TIME_L2H(s) = TIME_L2H[15:0]DEC/(F_PWM_M (Hz) x F_PWM). F_PWM_M (register address: 0x00 mask: 00F0) is the global clock frequency (default: 100kHz) and F_PWM (register address: 0x0E mask: 0300 for CH0) is a channel-specific clock divider (default is 1). The formula is: TIME_L2H[15:0]DEC = TIME_L2H(s) x (F_PWM_M (Hz) x F_PWM).
- DPM_START (channel specific - CONFIGURATION_REGISTERS_CH_X > CFG_DPM1_X MASK 00FF): The current value at which the DPM starts (used to cut off the base noise of the lines). The formula is: DPM_START(mA) = 64 x KCDR x GAIN x SNSF x DPM_START[7:0]DEC.
- DPM_THLD (channel specific - CONFIGURATION_REGISTERS_CH_X > CFG_DPM0_X MASK 0FFF): A current level, minimum depth of the valley before the DPM recognizes it as a valley (used to filter signal noise). The formula is: DPM_THLD(mA) = 8 x KCDR x GAIN x SNSF x DPM_THLD[11:0]DEC.
- DPM_MIN_NBR (channel specific -CONFIGURATION_REGISTERS_CH_X > CFG_DPM1_X MASK 0F00): A check of the condition for multiple PWM cycles (deglitch). The formula is: DPM_DEGLITCH = 2 x DPM_MIN_NBR[3:0]DEC x 1/FPWM.

The DPM diagnostics tools store the data in the following registers:

- I_DPM_PEAK (DIAGNOSTICS_CH_X > I_DPM_PEAK_X): The peak of the current before the valley. The formula is: I_DPM_PEAK(mA) = 8 x KCDR x GAIN x SNSF x I_DPM_PEAK[11:0]DEC.
- I_DPM_VALLEY (DIAGNOSTICS_CH_X > I_DPM_VALLEY_X): The lowest current point of the valley. The formula is: I_DPM_VALLEY(mA) = 8 x KCDR x GAIN x SNSF x I_DPM_VALLEY[11:0]DEC.
- REACTION_TIME (DIAGNOSTICS_CH_X > REACTION_TIME_X): The time between the start of the channel and the lowest current point of the valley. The formula is: REACTION_TIME (s) = REACTION_TIME[15:0]/(F_PWM_M x F_PWM).

- TRAVEL_TIME (DIAGNOSTICS_CH_X > TRAVEL_TIME_X): The time between the peak current before the valley and the lowest current point of the valley (the time it took the plunger to move). The formula is: TRAVEL_TIME (s) = TRAVEL_TIME[15:0]/(F_PWM_M x F_PWM).

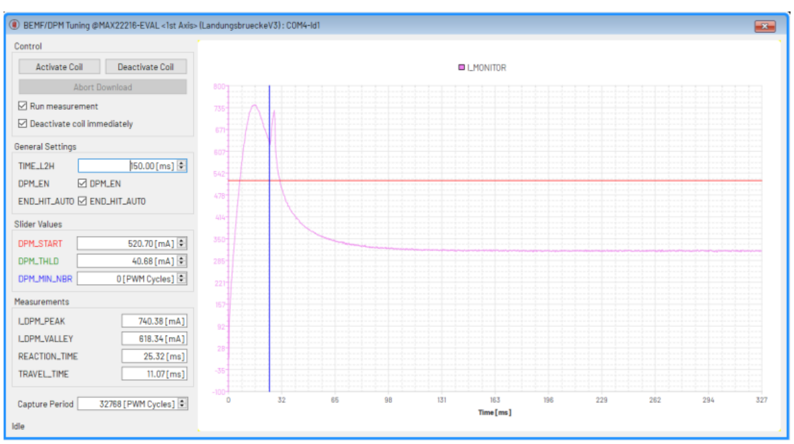
## DPM Tuning

1. Set DC_L2H, DC_H and TIME_L2H according to the valve data sheet.
2. DPM_EN HIGH
3. Set DPM_START to 1 (65.09 mA) and DPM_THLD:1 (8.14 mA) (minimum levels).
4. Once the graph shows the I_DPM_PEAK and I_DPM_VALLEY, the values can be tuned: DPM_START should be smaller than 0.95 x I_DPM_VALLEY and DPM_THLD should be smaller than 0.9 x (I_DPM_PEAK - I_DPM_VALLEY).
5. In extremely noisy environments, it is recommended to also use DPM_MIN_NBR to make the IC check the DPM condition for multiple system cycles.

## DPM Energy Saving Mode

END_HIT_AUTO is a special energy-saving mode that looks at the current rise after the I_DPPM_VALLEY, and when the current reaches the I_DPM_PEAK, it automatically changes the voltage level from DC_L2H to DC_H, saving a lot of energy. Even in this energy-saving mode, all the data about the DPM is stored in the IC.

To use the special energy-saving feature of the DPM, the END_HIT_AUTO must be set to HIGH.




037

*Figure 37. BEMF/DPM Tuning Tool - DPM Setup + END_HIT_AUTO Example*

## DPM Notice

DPM also works in CDR mode, but the valley is much smaller.

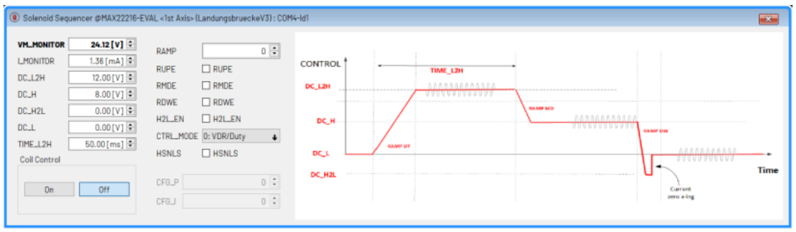
The DPM flag is triggered when the DPM is not detected!

For more current resolution, the SNSF and GAIN can be used to scale the current! In the example case, the maximum current during the DC_L2H is ~750mA. So, the GAIN can be set to 4 so the maximum current measurement is 1/4 x 4.165 A = ~1.041 A.

## Solenoid Inductance Tool (Inductance and Resistance Measurement)

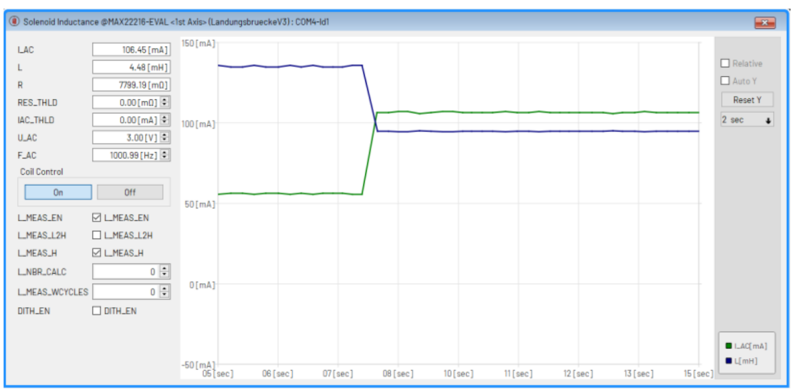
Inductance measurement works only in VDR! It works by creating a sine wave on top of the selected levels and then measuring the current and processing after every sine cycle.

The measurement starts only after the first turn-ON of the channel and it always measures during the subsequent OFF periods (DC_L). If the measurement is not needed when the channel is OFF, it is recommended to deselect the inductance measurement (L_MEAS_EN) before every turn OFF.



038

*Figure 38. VDR Solenoid Sequencer Settings Example for Use with the Inductance Measurement Tool*



039

*Figure 39. Inductance Measurement Tool (Solenoid Inductance Measurement Example)*

The graph shows how the signal looks when the solenoid plunger is forcefully opened, and it even has the resolution to see in-between positions. For a better use of this tool, a good current scaling is needed (SNSF and GAIN registers). The **Current Monitoring and Scaling** section shows this.

The inductance measurement is only for the VDR mode. It works by adding a sine wave on top of the level and then measuring the current at the same time. Afterwards, the current is internally processed and separated:

- AC part (I_AC): Reactance/inductance changes with the movement of the plunger. With this, the approximate position of the plunger can be monitored in real time. Also, monitor the open/closed inductance values to see the mechanical degradation of the system or coil degradation over time.
- DC part (R): Resistance changes with temperature. If the start coil/environment temperatures are known, the resistance change can be used to approximate temperature change and get live coil temperature.

To turn on the inductance measurement, set L_MEAS_EN to HIGH, and then select on which regions the measurements should be set: by default, on DC_L (when the signal is closed), and DC_H and DC_L2H by selection. Mostly, the interest area is the DC_H, as that is where the stable live data about the coil can be connected while it is in use. Afterwards, set the sine wave generator Vpp (U_AC) and frequency (F_AC). This measurement starts automatically next time the channel is used and provides I_AC and RES values in real-time (through SPI)!

As the measurement is set to continue during the channel OFF period, due to safety reason, the TMCL-IDE turns it OFF when the channel is turned OFF. If the inductance measurement must be used while the channel is OFF, the measurement can be set and then the channel should be turned ON and OFF directly from the register browser.

In single-ended mode, to measure during channel OFF, the DC_L voltage must be set to a value higher than the U_AC (which might not be good for the life of the solenoid). In full-bridge mode, this is not a problem, as negative values can be controlled.

**Inductance and Resistance Measurement Example: CH0 Single Channel/VDR (VDRnVDRDUTY)/Two-Level Control/Inductance Measurement on DC_H**

## Table 8. Inductance and Resistance Measurement Example Register Table

| NAME | ADDRESS | MASK | VALUE | COMMENT |
|---|---|---|---|---|
| ACTIVE | 0x01 | 8000 | HIGH | Turn ON the MAX22216 |
| CHS | 0x01 | 000F | 0x01 | 4x independent half bridges |
| VDRnVDRDUTY | 0x01 | 0010 | | |
| UVM | 0x66 | 0010 | HIGH | Clear bit |
| COMER | 0x66 | 0020 | HIGH | Clear bit |
| CTRL_MODE | 0x0D | C000 | 0x0 | VDR |
| DC_L2H | 0x09 | FFFF | 10922 | 12V |
| DC_H | 0x0A | FFFF | 7282 | 8V |
| TIME_L2H | 0x0C | FFFF | 5000 | 50ms |
| L_MEAS_EN | 0x13 | 0400 | HIGH | |
| L_MEAS_H | 0x13 | 0100 | HIGH | |
| L_MEAS_L2H | 0x13 | 0200 | 0 | Too short to be worth measuring (50ms) |
| F_AC | 0x07 | 0FFF | 656 | 1000.99Hz |
| U_AC | 0x08 | 7FFF | 2731 | 3V |
| SNSF | 0x0E | 0003 | 1 | 2/3 current scale |
| GAIN | 0x0E | 000C | 3 | 1/4 current scale |

- L_MEAS_EN (channel specific - CONFIGURATION_REGISTERS_CH_X > CFG_IND_0_X MASK 0400): Turn on the inductance measurement on this channel (by default on DC_L level).
- L_MEAS_H and L_MEAS_L2H (channel specific - CONFIGURATION_REGISTERS_CH_X > CFG_IND_0_X MASK 0100 / 0200): Select if the measurement should be on DC_L2H and/or DC_H.
- U_AC (global variable - ADD 0x08 MASK 7FFF): Sine wave generator level/voltage. The formula is: VDR: VAC(V) = KVDR x 36 x U_AC_SCAN[14:0]DEC/VDRDUTY: VAC (V) = KVDR x VM x U_AC_SCAN[14:0]DEC.
- F_AC (global variable - ADD 0x07 MASK 0FFF): Sine wave generator frequency. The formula is: F_AC = (F_PWM_M (Hz) x F_PWM) x (F_AC_SCAN[11:0]DEC/65535).
- SNSF and GAIN (channel specific - CONFIGURATION_REGISTERS_CH_X > CFG_CTRL1_X MASK 0003/000C): Current scaling (total 1/6) (for better I_AC measurement). The final current after scaling must be bigger than the maximum current output (DC_L2H during the HIT period).

The data from this tool can be used for diagnostics and predictive maintenance, and it is updated after each sine wave cycle:

- I_AC (channel specific - DIAGNOSTICS_CH_X > I_AC_X): AC part of the current that can be used to calculate the reactance and then inductance of the coil. The formula is: I_AC (mA) = KCDR x GAIN x SNSF x I_AC[15:0]DEC.

- R (channel specific - DIAGNOSTICS_CH_X > RES_X): Can be used to calculate resistance of the coil and create a temperature change approximation. The formula is: R(mΩ) = R[15:0] x ( KR/(SNSF x GAIN)).

### Inductance and Resistance Measurement Flags

This tool also has two flags that can be controlled: IND and RES flags (for each channel). These flags can be set in the register browser and by default are set to 0. That is why they trigger if the "Solenoid Inductance" GUI tool is used with default IND and RES register values.

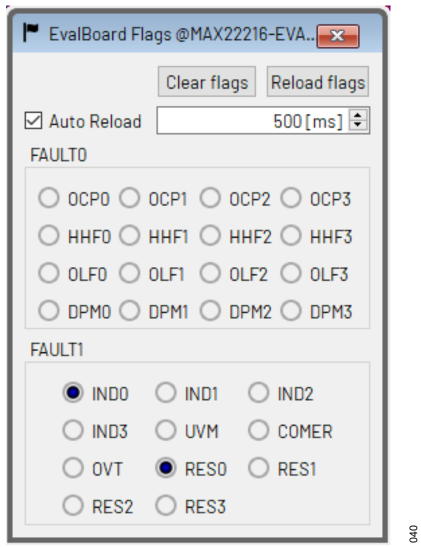



*Figure 40. EvalBoard Flags (IND0 and RES0 Flags Triggered due to Inductance and Resistance Measurement)*

- IND Flag (IAC_THLD - CONFIGURATION_REGISTERS_CH_X > CFG_IND_1_X): The flag is triggered when the I_AC gets over a certain threshold. The induction value can increase or decrease due to a lot of factors. So, this has a limited use case, but the live values can be monitored to develop more advanced position detection and predictive maintenance systems. When the channel is off, the IND flag can also be triggered due to channel noise.
- RES Flag (RES_THLD - CONFIGURATION_REGISTERS_CH_X > CFG_R_THLD_X): As the resistance increases with temperature, the flat can be set to trigger when the coil gets too hot for the specific application.

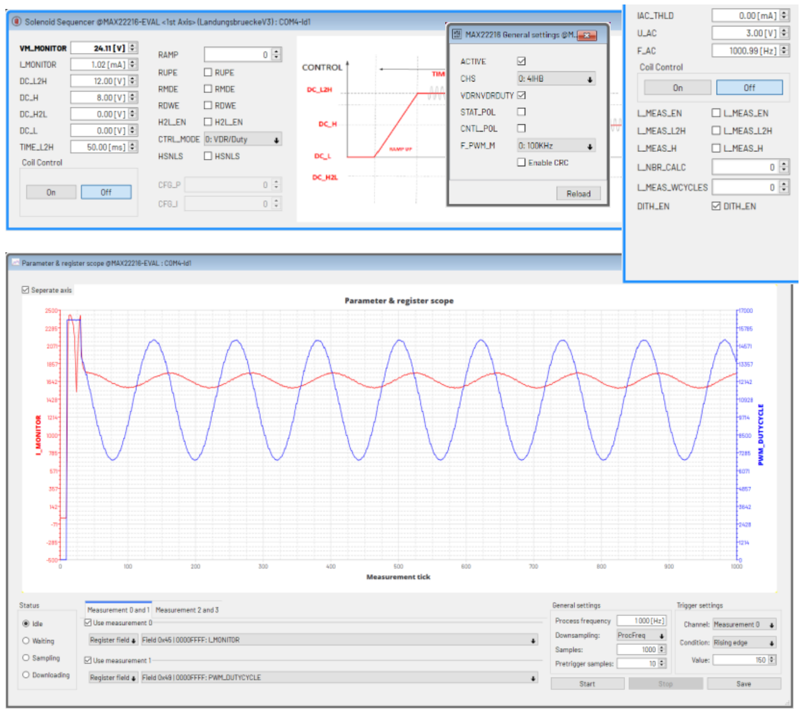
## Solenoid Inductance Tool (Dithering)

Similar to induction measurement, firstly set up the output (like in the "Solenoid Sequencer") and it creates a sine wave on the DC_H and DC_L levels. This mode is specifically developed for differential valves, as they cannot be allowed to have a stable level (as they get stuck), and they need a sine wave even on the DC_L level (that must be set as a voltage or current above the set U_AC). For this, use a full-bridge CHS configuration. Dithering works with both VDR and CDR, and while using CDR, it is important to have proper PI settings. Otherwise, the valve might get stuck. The dithering can be used by checking the DITH_EN box.

The formulas are like inductance measurement and there is an extra one added for U_AC with current control:

- U_AC (global variable - ADD 0x08 MASK 7FFF): Sinewave generator level/voltage. The formula is: VDR: VAC(V) = KVDR x 36 x U_AC_SCAN[14:0]DEC/VDRDUTY: VAC (V) = KVDR x VM x U_AC_SCAN[14:0]DEC/CDR mode: IAC (mA) = KCDR x GAIN x SNSF x U_AC[14:0]DEC.
- F_AC (global variable - ADD 0x07 MASK 0FFF): Sine wave generator frequency. The formula is: F_AC = (F_PWM_M (Hz) x F_PWM) x (F_AC_SCAN[11:0]DEC/65535).

- DITH_EN (channel specific -CONFIGURATION_REGISTERS_CH_X > CFG_IND_0_X MASK 0800): Enables dithering on the specific channel.



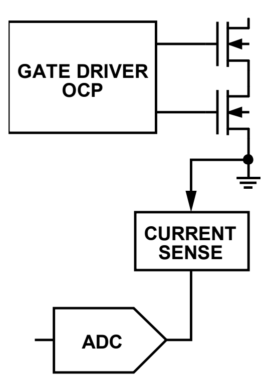

041

*Figure 41. VDR Dithering Output Example*

VDR dithering example output, shown through the PWM duty cycle in blue and channel current in red. As this is VDR, the output voltage looks like the inductance measurement output, but there is no measurement feedback.

## Current Monitoring and Scaling

The MAX22216 has 12-bit (signed 13-bit) current monitoring on the low-side FET of each output channel. It also provides advanced scaling configuration settings that can also control the $R_{ON}$ of the MOSFET.




042

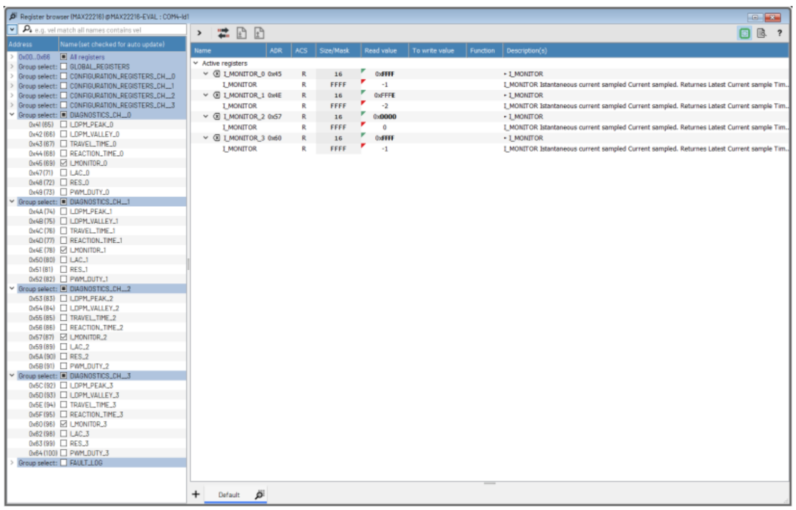
*Figure 42. Current Measurement Block Diagram (from Data Sheet)*

### Current Monitoring (I_MONITOR)

By default, the ADC is set to have the maximum scale of 4.165A. The current monitoring for each channel can be seen in the I_MONITOR register, in the DIAGNOSTICS_CH_X > I_MONITOR_X of each channel (where X is the channel).

The I_MONITOR can be seen in the “Solenoid Sequencer” tool of each channel or in the “Register Browser” tool.

The I_MONITOR register is 16-bit, but the biggest value that is stored is signed 15-bit at I_MONITOR_0 when in the CHS:4 (four parallel half-bridges) configuration. Every time channels are set-up in parallel, the summed value is stored in the I_MONITOR of the channel with the smallest number. Example: for CHS:3 (two parallel half-bridges + two parallel half-bridges), there is 14-bit stored in I_MONITOR_0 for the CH0 and CH1 parallel channels, and I_MONITOR_2 for the CH2 and CH3 parallel channels. When the channels are in full-bridge configurations, the I_MONITORs of the channels show opposite currents (one positive, one negative depending on the current orientation).



043

*Figure 43. I_MONITOR of All Channels Shown in the Register Browser Tool*

### Current Scaling (SNSF and GAIN)

The important thing is to always use a scaling bigger than the current consumption! Otherwise, the current is not measured correctly, and it affects the functionality of some features. This might also affect the internal ADCs.

In cases where the solenoid consumes a lot less current, while also needing a lot of precision, the current scale can be changed to keep the measurement 12-bit resolution (signed 13bit).

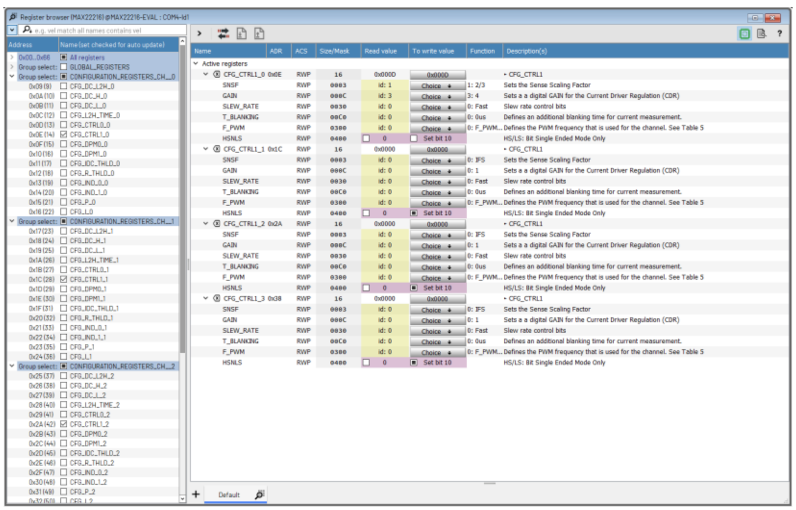
To change this scale, use the SNSF and GAIN registers. SNSF also affects the $R_{ON}$. So, it is recommended to start the current scaling using GAIN and not use SNSF, if not really needed.

## Table 9. Full Scale and Sense Scale Configurations

| GAIN [1:0] | GAIN | SNSF [1:0] | SNSF | TYPICAL LS $R_{ON}$ (Ω) | TOTAL FACTOR | MULTIPLIER | MAXIMUM CURRENT (A) |
|---|---|---|---|---|---|---|---|
| 00 | 1 | 00 | 1 | 0.17 | 1 | 1.00 | 4.165 |
| 01 | 1/2 | | | | 1/2 | 0.5 | 2.082 |
| 10 | 1/3 | | | | 1/3 | 0.333 | 1.388 |
| 11 | 1/4 | | | | 1/4 | 0.25 | 1.041 |
| 00 | 1 | 01 | 2/3 | 0.23 | 2/3 | 0.667 | 2.776 |
| 01 | 1/2 | | | | 2/6 | 0.333 | 1.388 |
| 10 | 1/3 | | | | 2/9 | 0.222 | 0.925 |

| GAIN [1:0] | GAIN | SNSF [1:0] | SNSF | TYPICAL LS $R_{ON}$ (Ω) | TOTAL FACTOR | MULTIPLIER | MAXIMUM CURRENT (A) |
|---|---|---|---|---|---|---|---|
| 11 | 1/4 | | | | 2/12 | 0.167 | 0.694 |
| 00 | 1 | 10 | 1/3 | 0.43 | 1/3 | 0.333 | 1.388 |
| 01 | 1/2 | | | | 1/6 | 0.167 | 0.694 |
| 10 | 1/3 | | | | 1/9 | 0.111 | 0.463 |
| 11 | 1/4 | | | | 1/12 | 0.083 | 0.347 |

All the current formulas for the MAX22216 contains SNSF and GAIN, and these registers can be found in the “Register Browser” under CONFIGURATION_REGISTERS_CH_X > CFG_CTRL1_X (where X is the channel).



044

*Figure 44. All Channels SNSF and GAIN Shown in the Register Browser Tool*

## PWM Engine

The MAX22216 has a complex PWM engine, used not only to control the output PWM frequency, but also to synchronize all the internal blocks and measurements.

### F_PWM_M and Channel F_PWM

## Table 10. F_PWM_M (ADD 0x00 MASK 00F0) Global PWM Engine

| PWM MAIN FREQUENCY SETTING | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| **HEX** | **0x0** | **0x1** | **0x2** | **0x3** | **0x4** | **0x5** | **0x6** | **0x7** |
| **F_PWM_M** | 100 | 80 | 60 | 50 | 40 | 30 | 25 | 20 |
| **HEX** | **0x8** | **0x9** | **0xA** | **0xB** | **0xC** | | | |
| **F_PWM_M** | 15 | 10 | 7.5 | 5 | 2.5 | | | |

**Note:** 0xD, 0xE, and 0xF are same as 0x0 (100kHz).

## Table 11. F_PWM Channel Frequency Divider for Output PWM Frequency

| INDIVIDUAL PWM FREQUENCY | | |
|---|---|---|
| F_PWM[1:0] | | CHOPPING FREQUENCY OF INDIVIDUAL CHANNEL |
| 0 | 0 | F_PWM_M |
| 0 | 1 | F_PWM_M/2 |
| 1 | 0 | F_PWM_M/4 |
| 1 | 1 | F_PWM_M/8 |

The channel F_PWM can be found at CONFIGURATION_REGISTERS_CH_X > CFG_CTRL1_X MASK 0300.

F_PWM_M and channel F_PWM dividers can be changed on-the-fly, and the channel F_PWM dividers can be used to set different PWM frequencies on each channel at the same time.

In a full-bridge channel configuration, the actual output frequency is doubled due to the use of center-aligned PWM.

### PWM Slew-Rate

The MAX22216 can control the shape of the output PWM signal with the objective or reducing EMI (alongside lower channel F_PWM). SLEW_RATE channel specific - CONFIGURATION_REGISTERS_CH_X > CFG_CTRL_1_X MASK 0030) gives four PWM slew rate options (with the default value set to 0 / FAST).

## Table 12. SRC (Slew Rate Control) Settings for Output PWM

| SRC[1:0] | SLEW RATE CONTROL |
|---|---|
| 00 | Fast |
| 01 | 400V/µs |
| 10 | 200V/µs |
| 11 | 100V/µs |

### Open-Load Detection

This function is a continuity detection tool when the channel is OFF.

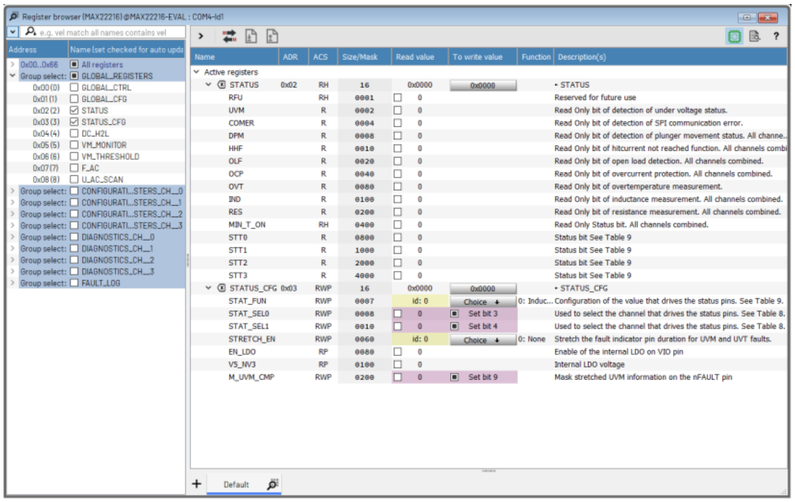
The OL_EN can be found under CONFIGURATION_REGISTERS_CH_X > CFG_CTRL0_X MASK 1000, and it just needs to be set to HIGH in the channel settings.

In a full-bridge channel configuration, set OL_EN on both channels, and set one of the channels to HSnLS (from CONFIGURATION_REGISTERS_CH_X > CFG_CTRL1_X).

If a contact on the configured line is not detected, it automatically triggers the OLF flags for the specific channels.

### STAT Function and STAT Pins

The status monitor STT registers are channel-specific, user-selectable settable alarms. The STAT function output can be seen under STATUS as STT (ADD 0x02 MASK 0800, 1000, 2000, 4000). The STATUS (ADD 0x02) can be used to monitor if any group of FLAGs is triggered. Only one STAT function can be set for all channels, from the STAT_FUN register (ADD 0x03 MASK 0007).



045

*Figure 45. STATUS (ADD 0x02) and STATUS_CFG (ADD 0x03) Registers Shown in the Register Browser*

### STAT_FUN (Status Function Settings)

## Table 13. Status Function Settings

| STAT_FUN | FUNCTION | CONDITION | STAT<br>STAT_POL = “0” | STAT<br>STAT_POL = “0” | STT BITS |
|---|---|---|---|---|---|
| **0x0** | Status detection based on inductance measurement | If IAC > IAC_THLD | Low | High | “0” |
| | | If IAC < IAC_THLD | High | Low | “1” |
| **0x1** | PWM monitor | - | PWM | $\overline{PWM}$ | NA |
| **0x2** | Status detection based on resistance measurement | If RES > RES_THLD | Low | High | “0” |
| | | If RES < RES_THLD | High | Low | “1” |
| **0x3** | Status detection based on successful plunger movement (DPM) | If CTRL = LOW or HIGH but DPM is not detected | Low | High | “0” |
| | | If CTRL = HIGH and DPM is detected | High | Low | “1” |
| **0x4** | Status detection based on VM detection | If VM < UVLO | Low | High | “0” |
| | | If VM > UVLO | High | Low | “1” |
| **0x5** | Status detection based on I_MONITOR measurement | If I_MONITOR < IDC_THLD | Low | High | “0” |
| | | If I_MONITOR > IDC_THLD | High | Low | “1” |

STT options from the STATUS_FUN (ADD 0x03 MASK 0007):

- **IAC (0x00):** While using inductance measurement, the I_AC value is compared with the IAC_THLD and triggered if bigger. This I_AC represents the reactance of the solenoid, not directly the inductance. This can be used as an alarm for the solenoid position.
- **PWM (0x01):** Shows the channel PWM_DUTYCYCLE output in real-time.
- **RES (0x02):** While using inductance measurement, the calculated R value is compared with the RES_THLD and triggered if bigger. This can be used to set a temperature alarm.
- **DPM (0x03):** It is triggered when a successful DPM is detected.
- **VM (0x04):** Checks if the IC is in the functional voltage range. It is triggered when VM => UVLO.
- **IDC (0x05):** Compares the I_MONITOR to IDC_THLD and triggered if bigger.

### STAT Pins

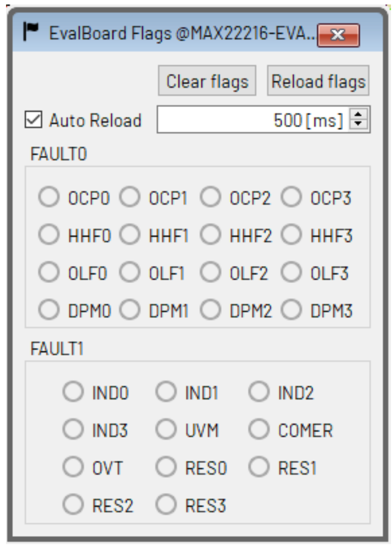
The STT alarms can be set to be sent out through the STAT pins. As there are four STT alarms and only two STAT pins, there is a matrix that must be configured to connect them. By default, STAT pin 0 is connected to STT0 and STAT pin 1 is connected to STT2.

## Table 14. STAT Pins Connection to STAT Function Based on CHS

| CHS | CONFIGURATION SETTINGS | STAT0 | STAT1 |
|---|---|---|---|
| 0x0 | 4x Independent HB | STT0 if STAT_SEL(0) = 0<br>STT1 if STAT_SEL(0) = 1 | STT2 if STAT_SEL(1) = 0<br>STT3 if STAT_SEL(1) = 1 |
| 0x1 | 3x Parallel HB<br>1x Independent HB | STT0 = STT1 = STT2 | STT3 |
| 0x2 | 2x Parallel HB<br>2x Independent HB | STT0 = STT1 | STT2 if STAT_SEL(1) = 0<br>STT3 if STAT_SEL(1) = 1 |
| 0x3 | 2x Parallel HB<br>2x Parallel HB | STT0 = STT1 | STT2 = STT3 |
| 0x4 | 4x Parallel HB | STT0 = STT1 = STT2 = STT3 | - |
| 0x5 | 1x Independent FB<br>1x Independent FB | STT0 in PWM monitoring (STT0 or STT1) all other modes | STT2 in PWM monitoring (STT2 or STT3) all other modes |
| 0x6 | 1x Independent FB<br>2x Independent HB | STT0 in PWM monitoring (STT0 or STT1) all other modes | STT2 if STAT_SEL(1) = 0<br>STT3 if STAT_SEL(1) = 1 |
| 0x7 | 1x Independent FB<br>2x Parallel HB | STT0 in PWM monitoring (STT0 or STT1) all other modes | STT2 = STT3 |
| 0x8 | 1x Parallel HB | STT0 if STAT_SEL(0) = 0<br>STT1 if STAT_SEL(0) = 1 | STT0 if STAT_SEL(0) = 0<br>STT1 if STAT_SEL(0) = 1 |

## Fault Masking

FLAGs can be erased by simply setting their specific registers to HIGH. The fault registers are inside FAULT0 (ADD 0x65) and FAULT1 (ADD 0x66). They can also be seen on the "EvalBoard Flags" tool.

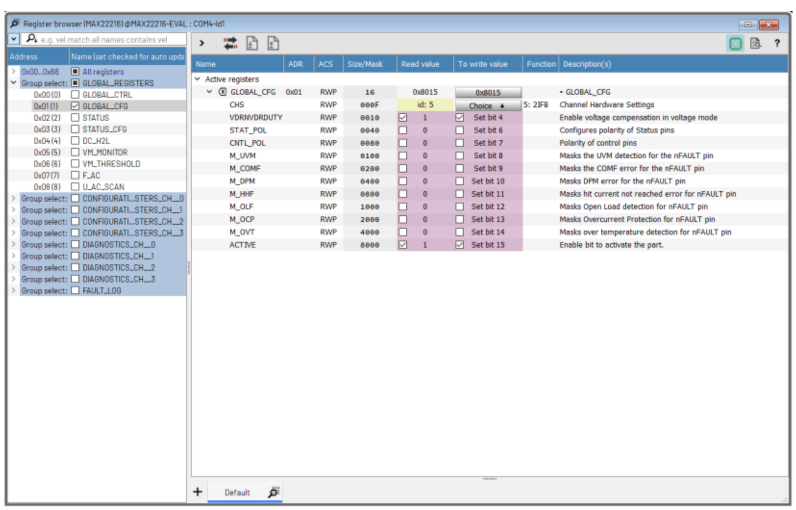



046

*Figure 46. EvalBoard Flags Tool*

In standalone use, if there is a need for some flags to not appear on the nFaut pin, there is the option to mask them. Recommended when the MAX22216 is OTPd with the control settings and used without SPI.

The masking registers are in the global configuration GLOBAL_CFG (ADD 0x01 MASK 0100, 0200, 0400, 0800, 1000, 2000, 4000) and they can be set using OTP.



047

*Figure 47. Masking Registers in the Register Browser Tool*

### Fault Stretch

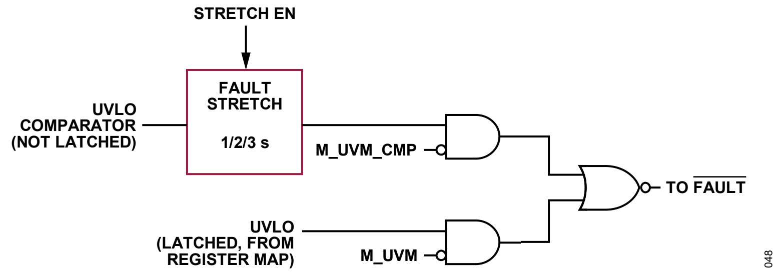



*Figure 48. Stretch Enable Logic (from Data Sheet)*

- STRETCH_EN (ADD 0x03 MASK 0060): This function stretches the UVM and OVT flags by a settable time.
- M_UVM_CMP (ADD 0x03 MASK 0200): Masks the stretched UVM signal from the nFAULT pin.

## Power Supply

This section gives insights into both the VM and the internal regulators that can be set for the standalone mode.

### VM Monitoring

The VM has a special register to check supply voltage called VM_MONITOR (ADD 0x05 MASK 1FFF).

VM (mV) = KVM x VM_MONITOR, where KVM = 9.73.

### VM_THRESHOLD

It is possible to set the functional voltage range of the MAX22216. Thses registers are intended to be OTPd when the MAX22216 is in standalone mode to set the valve to function only in specific VM ranges.

Internally, there is a hysteresis system where VM_THLD_DOWN (ADD 0x06 MASK 000F) is the minimum voltage before the IC turns OFF (by default 4.3V) and VM_THLD_UP (ADD 0x06 MASK 00F0) is the minimum voltage level for the IC to turn on (by default 4.5V).

The OFF values (0) and first set values (1) are the same (4.3V and 4.5V)! The next values are in steps of 2V, and they stop at 32.3V and 32.5V.

## Table 15. VM_THLD_DOWN Settings (from Data Sheet)

| VALUE | ENUMERATION | DECODE |
|---|---|---|
| 0x0 | OFF | Disable |
| 0x1 | 4_3V | |
| 0x2 | 6_3V | |
| 0x3 | 8_3V | |
| 0x4 | 10_3V | |
| 0x5 | 12_3V | |
| 0x6 | 14_3V | |
| 0x7 | 16_3V | |
| 0x8 | 18_3V | |
| 0x9 | 20_3V | |
| 0xA | 22_3V | |
| 0xB | 24_3V | |

| VALUE | ENUMERATION | DECODE |
|---|---|---|
| 0xC | 26_3V | |
| 0xD | 28_3V | |
| 0xE | 30_3V | |
| 0xF | 32_3V | |

## Table 16. VM_THLD_UP Settings (from Data Sheet)

| VALUE | ENUMERATION | DECODE |
|---|---|---|
| 0x0 | OFF | Disable |
| 0x1 | 4_5V | |
| 0x2 | 6_5V | |
| 0x3 | 8_5V | |
| 0x4 | 10_5V | |
| 0x5 | 12_5V | |
| 0x6 | 14_5V | |
| 0x7 | 16_5V | |
| 0x8 | 18_5V | |
| 0x9 | 20_5V | |
| 0xA | 22_5V | |
| 0xB | 24_5V | |
| 0xC | 26_5V | |
| 0xD | 28_5V | |
| 0xE | 30_5V | |
| 0xF | 32_5V | |

### Internal Regulator

There is an internal 5V/3.3V regulator that can be used to supply the logic of the IC directly from the VM. It is intended for use in standalone modes, where there is no microcontroller or external power supply.

It can be turned on ONLY using OTP:

- EN_LDO (ADD 0x03 MASK 0080): Turns on the LDO by default to 3.3V.
- V5_nV3 (ADD 0x03 MASK 0100): Changes the LDO from 3.3V to 5V (EN_LDO also must be programmed for this register to have any effect).

## Cyclic Redundancy Check (CRC)

To ensure correct SPI data communication and mitigate communication errors, the MAX22216 has an integrated cyclic redundancy check (CRC) engine. The CRC can be activated by connecting the CRC_EN pin of the MAX22216 to $V_{CC}$.

For the MAX22216EVKIT, the TMCL evaluation platform sets and calculates the CRC when it is enabled in the GUI. The CRC should not affect the use of the MAX22216EVKIT with the TMCL-IDE in any way.

The CRC adds 8 bits at the end of the SPI message, from which the last 5 bits are the frame check sequence (FCS) bits.

The 5-bit FCS is based on the generator polynomial X5 + X4 + X2 + 1 (CRC-5-ITU), with a CRC calculation message XOR starting value = 11111. The FCS bits (CIx/COx) are calculated on all the data sent in one SPI command, including the three “0” in the MSBs of the check byte. Therefore, the CRC is calculated from 19 bits. CI0 is the LSB of the FCS.

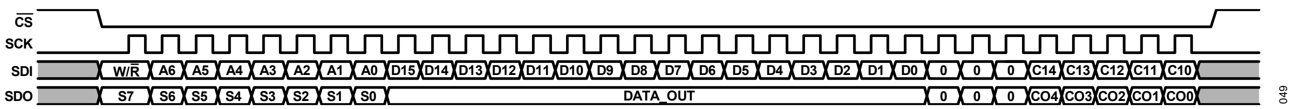



*Figure 49. SPI Datagram with CRC (from Data Sheet)*

**CRC Calculation Example**

```
Polynomial: x^5 + x^4 + x^2 + x^0
Binary:     110101


10000000                         <- Register address (Example)
                0000000000000001 <- Register data (Example)
100000000000000000000001000      <- Data bits with 3 padding zero bits
11111                            <- Starting value that gets XORed into the data bits
011110000000000000000001000      <- Data bits with starting value XORed in
01111000000000000000000100000000 <- Data bits with starting value XORed in and five 0 bits appended
000000
 1111000000000000000000100000000 <- Step 1
 110101
  010010000000000000000100000000 <- Step 2 (XOR applied)
  000000
   10010000000000000000100000000 <- Step 3
   110101
    1000100000000000000100000000 <- Step 4 (XOR applied)
    110101
     101110000000000000100000000 <- Step 5 (XOR applied)
     110101
      11011000000000000100000000 <- Step 6 (XOR applied)
      110101
       0001100000000000100000000 <- Step 7 (XOR applied)
       000000
        001100000000000100000000 <- Step 8
        000000
         01100000000000100000000 <- Step 9
         000000
          1100000000000100000000 <- Step 10
          110101
```

```
00101000000010000000 <- Step 11 (XOR applied)
000000
 0101000000010000000 <- Step 12
 000000
 101000000010000000 <- Step 13
 110101
  11101000010000000 <- Step 14 (XOR applied)
  110101
   0111100010000000 <- Step 15 (XOR applied)
   000000
    111100010000000 <- Step 16
    110101
     01001010000000 <- Step 17 (XOR applied)
     000000
     1001010000000 <- Step 18
     110101
      100000000000 <- Step 19 (XOR applied)
      110101
       10101000000 <- Step 20 (XOR applied)
       110101
        1111100000 <- Step 21 (XOR applied)
        110101
         010110000 <- Step 22 (XOR applied)
         000000
          10110000 <- Step 23
          110101
           1100100 <- Step 24 (XOR applied)
           110101
            001110 <- Step 25 (XOR applied)
            000000
             01110 <- Step 26
             000000
              1110 <- Step 27
```

Final Datagram: 10000000 00000000 00000001 00011100

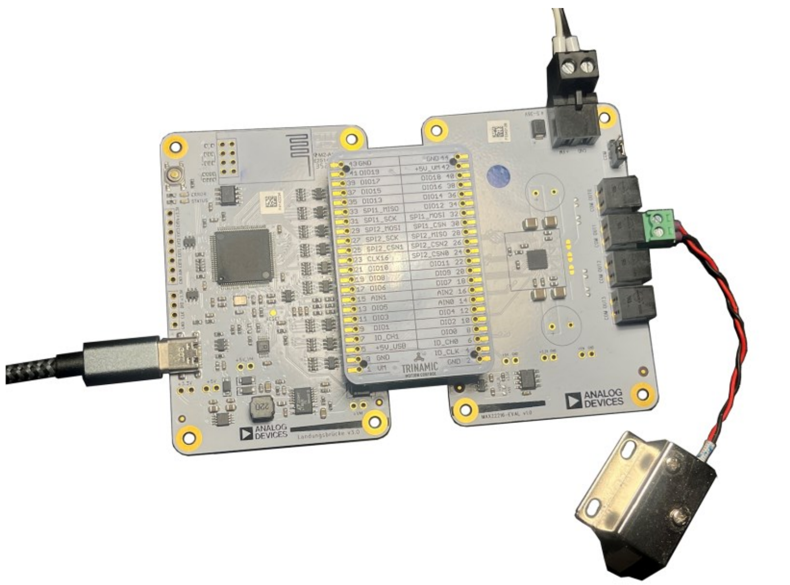
### Setup Example: VDR Solenoid

**Load:** Small 12V Door Lock Pull Solenoid (Non-Polarized)

**Setup:** Load Between OUT1 and COM(VM+)

**PSU:** 24V

In this example setup, the MAX22216 controls a 6V small solenoid. The focus of this example is to showcase the detection of plunger movement (DPM) with automatic transition to HOLD level and impedance measurement while in voltage drive regulation (VDR) control mode. This example also shows other important features of the MAX22216: the open-load detection and the internal selectable STAT alarms that can be set to output a digital signal on the STAT1 pin.

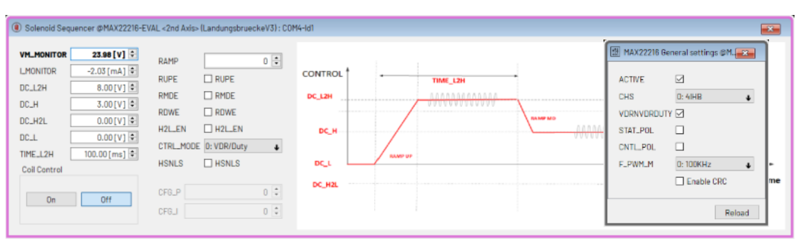


050

*Figure 50. Example Setup Using a Small Solenoid*

**(Example) Required Functionality:**

- **Two-level sequencer in VDR:** Create a HIT period to open the solenoid (DC_L2H, TIME_L2H) and a HOLD period to keep it open (DC_H). This is a big energy saving system.
- **DPM:** Detect plunger movement and do an automatic transition to the hit period (END_HIT_AUTO) (for even more energy savings).
- **STAT:** Good DPM alarm with reversed output on the STAT0 pin (simple digital output when DPM triggers).
- **OL:** Detects if the solenoid is connected well and the coil is not broken. Otherwise, FLAG is triggered.
- **IndMeas + I Scale:** Position approximation using IndMeas and I Scale for better measurement.




051

*Figure 51. MAX22216 General Settings and CH1 Solenoid Sequencer Example Settings for VDR Solenoid*

The data sheet mentions 12V to open (HIT) and 6V to HOLD. It is recommended to start with a long time (like 100ms or 200ms) that is updated afterwards. Now, the settings are needed for a complete initial actuation, not to be perfect. After a bit of testing, it can be seen that the solenoid needs lower voltage levels to work (8V HIT and 3V HOLD).

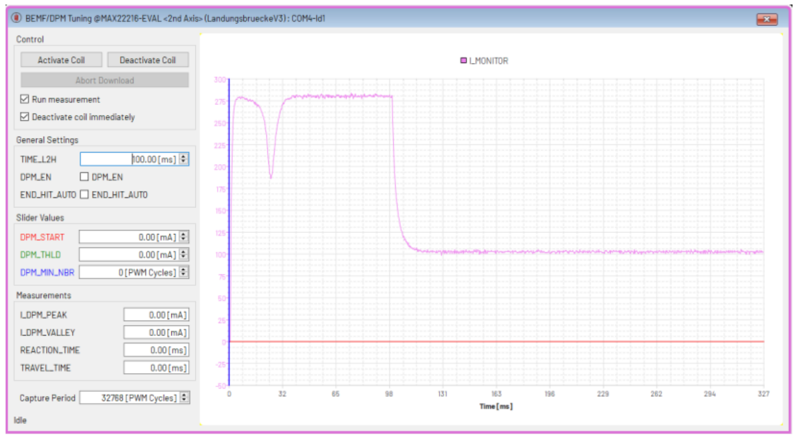



*Figure 52. MAX22216 CH1 DPM Tuning Tool Example Empty Read*

Inside the DPM tuning tool, if the “Activate Coil” button is pressed without any settings, the graph shows a reading of the current for the preset capture period (with no data in the “Measurements” section). The current graph shows data about the position of the actual DPM and can be used to tune the TIME_L2H period. To set the DPM, the DPM_EN box needs to be checked. Set the DPM_START (red Line) under the “Valley” minimum current point and set the DPM_THLD as a value in-between the maximum current point and “Valley” minimum current point (with at least a 5% margin).

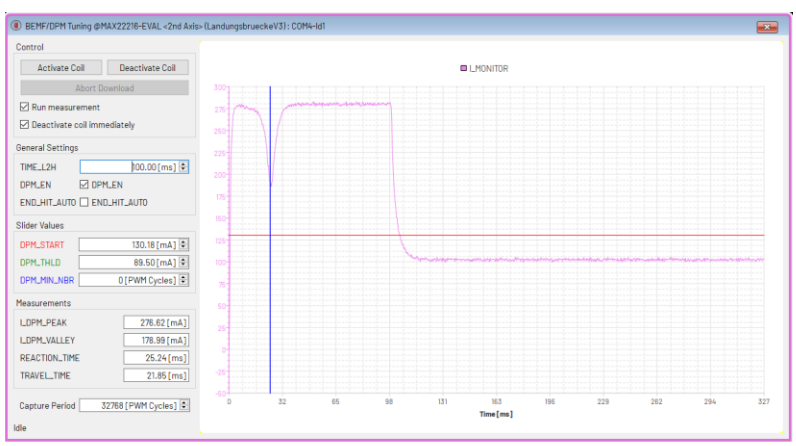
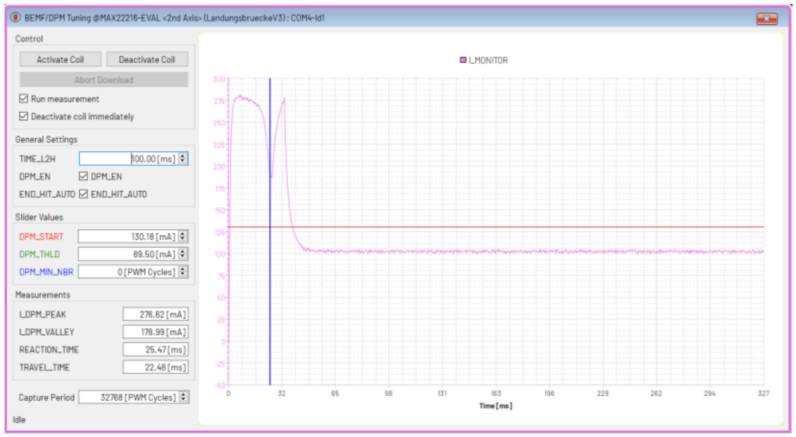



*Figure 53. MAX22216 CH1 DPM Tuning Tool Example Settings*

Once the DPM is set up properly, measurements should appear in the “Measurements” section (and the DPM1 flag should not be triggered in the “EvalBoard Flags”).

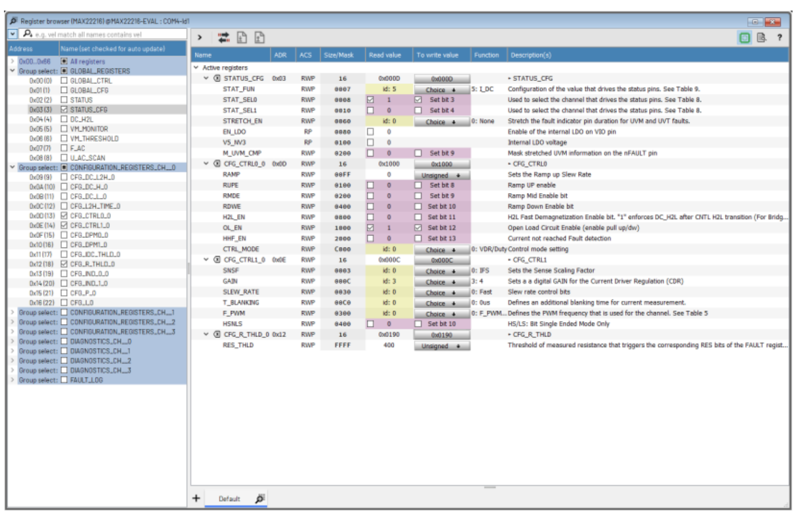



054

*Figure 54. MAX22216 CH1 DPM Tuning Tool and Auto Level Change Example Settings*

On top of the DPM, the MAX22216 has another energy saving tool. The END_HIT_AUTO sets the transition between the HIT and HOLD sections to be triggered after the DPM is detected (after the “Valley”), when the current is increasing again, and reaches the previously stored “Peak” current value. This setting overwrites the TIME_L2H setting.

To set the CH1 OL (OLF_EN), “CH1 Current Scaling” (GAIN, SNSF), and “STAT Alarm”, the registers must be written using the “Register Browser”. As the DPM setting values are based on the current scaling factors, the DPM automatically scales to the measured current so the changes in GAIN and SNSF do not require new DPM settings. In cases where the DPM values are set too close to the actual “Peak” and “Valley” values, the change in resistance created by the SNSF can also create small changes in the current, which can affect the DPM.



055

*Figure 55. OLF, STAT, and Current Scaling Registers in the Register Browser Tool*

**Open-load detection:**

- CH1 OL_EN (ADD 0x1B MASK 1000): It just needs to be set to HIGH.

**Positive DPM detection with reverse output:**

- STAT_FUN (ADD 0x03 MASK 0007): Set the STAT function to 3:DPM (0x03) so the function triggers when a DPM is detected.
- STAT_SEL0 (ADD 0x03 MASK 0008): Connects the STAT0 pin to the internal STAT1 function (by default, STAT0 pin is connected to STAT0 function).

- STAT_POL (ADD 0x01 MASK 0040): Changes the STAT pins output polarity so they output LOW when the function is triggered, and HIGH otherwise. This can also be set from the MAX22216 “General Settings” tool.

When the DPM is detected, the STAT0 pin is LOW. This becomes an extra DPM alarm on the STAT0 pin line.

**Current scaling:**

- CH1 SNSF (ADD 0x1C MASK 0003): Left with default value of 0. This register changes the current scale and the MOSFET resistance. So, it is not recommended to be used for current scaling, unless needed.
- CH1 GAIN (ADD 0x1C MASK 000C): Set to maximum division of the current: ¼. This changes only the current scaling, and it is the recommended scaling register.

As the inductance measurement tool is very dependent on solenoids resistance and impedance range, and it is influenced by mechanical noise, the tool is not recommended for precise position measurements.

The inductance measurement tool is intended for use as an approximation of the relative position, and it is recommended to scale the current as much as possible (otherwise, the solenoid position/movement might not be visible).

The U_AC and F_AC settings must be tested and set with the specific load.

056

*Figure 56. Solenoid Inductance Tool Example Settings*

Firstly, set the F_AC to a general value like 100. The U_AC must be set to a voltage where a noticeable change can be seen in the graph. Afterwards, change the F_AC: if the frequency is too big, the solenoid makes noise, and if it is too small, the solenoid vibrates.

While testing this solenoid, the HOLD level can be changed (DC_H from “Solenoid Sequencer”) if the solenoid does not go back to the actuated position after it is forcefully open. This situation is noticed in this example, and the HOLD voltage is changed to 6V. This change does not affect the DPM settings. As multiple changes are made after the DPM tool is set, it is recommended to do a final check with the DPM tool.

The exported settings:

0x01 = 0x00008050 ;; writing GLOBAL_CFG @ address 1=0x01 with 0x00008050=32848=0.0

0x03 = 0x0000000B ;; writing STATUS_CFG @ address 2=0x03 with 0x0000000B=11=0.0

0x07 = 0x0000028F ;; writing F_AC @ address 5=0x07 with 0x0000028F=655=0.0

0x08 = 0x00000AAB ;; writing U_AC_SCAN @ address 6=0x08 with 0x00000AAB=2731=0.0

0x17 = 0x00001C72 ;; writing CFG_DC_L2H_1 @ address 21=0x17 with 0x00001C72=7282=0.0

0x18 = 0x00001555 ;; writing CFG_DC_H_1 @ address 22=0x18 with 0x00001555=5461=0.0

0x1A = 0x00002710 ;; writing CFG_L2H_TIME_1 @ address 24=0x1A with 0x00002710=10000=0.0

0x1B = 0x00001000 ;; writing CFG_CTRL0_1 @ address 25=0x1B with 0x00001000=4096=0.0

0x1C = 0x0000000C ;; writing CFG_CTRL1_1 @ address 26=0x1C with 0x0000000C=12=0.0

0x1D = 0x0000000A *;; writing CFG_DPM0_1 @ address 0=0x1D with 0x0000000A=10=0.0*

0x1E = 0x00005003 *;; writing CFG_DPM1_1 @ address 1=0x1E with 0x00005003=20483=0.0*

### Setup Example: CDR Contactor Full-Bridge

**Load:** Contactor with 24V DC Coil (Non-Polarized)

**Setup:** Load Between OUT2 and OUT3

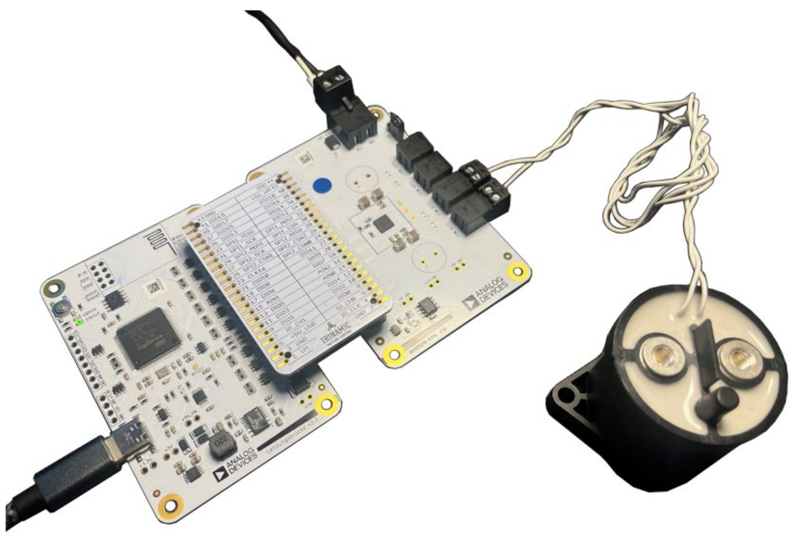
**PSU:** 24V

In this example setup, the MAX22216 drives a 24V contactor. The focus of this example is to showcase the current drive regulation (CDR) and detection of plunger movement (DPM), and an alarm to check if the current target is reached (HHF), alongside other more advanced analog features. The MAX22216 is also set to use the open-load detection, as it requires an extra setting when used in full-bridge mode, and the global and channel PWM frequency are set to different values to showcase the flexibility of the part.

The MAX22216 also has some more advanced analog and mixed features: the SlewRate of the PWM output can be limited to lower the EMC, and the transition periods between (RAMPs) the different levels can be controlled, which can lower the actual noise produced by the solenoid/contactor and limit the mechanical impact force of the plunger actuation.

To help with opening the solenoid faster when the channel is turned OFF in full-bridge mode, the MAX22216 can create a reverse voltage that helps with the coil discharge (fast demagnetization – DC_H2L).

For contactors in particular, as they are used to drive high powers, there are a lot of requirements related to both safety and control. Generally, EMC emissions are not a big concern, and noise is not a concern at all. Also, the fast demagnetization has a limited use, as the plunger physical movement is limited by eddy currents.



057

*Figure 57. Example Setup using a Contactor*

**(Example) Required Functionality:**

**Two-level sequencer in CDR:** Energy saving system and current control for better driving signal with an alarm to check if the current target is reached (HHF). This can be used to check coil health.

**DPM:** Detect plunger movement.

**OL:** Detects if the solenoid is connected well and the coil is not broken. Otherwise, FLAG is triggered.

**F_PWM_M and F_PWM:** Lower system and channel PWM frequency.

**Slew Rate:** Lower PWM slew rate to lower EMI.

**RAMP:** Limit transition section slew rate to lower noise (and mechanical force for RUPE).

**Fast Demagnetization (DC_H2L):** When the channel is turned OFF, a negative voltage is set on the output to help coil discharge and make the opening faster.

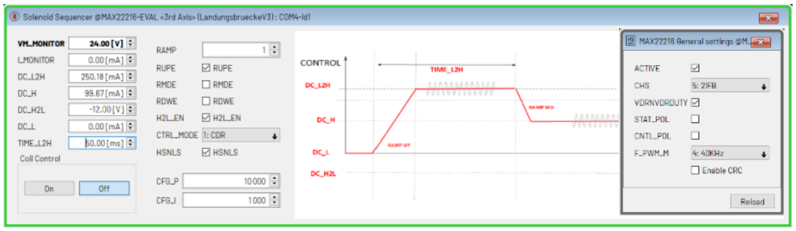


*Figure 58. MAX22216 General Settings and CH2 Solenoid Sequencer Example Settings for CDR Contactor*

In the “General Settings”, the CHS is set to 5: 2IFB. So, the output of the MAX22216 is set to have two full-bridges (CH0CH1 and CH2CH3). Also, the global F_PWM is lowered to 40kHz to lower EMI.

The “Solenoid Sequencer” is set for CDR. From the contactor’s data sheet, the maximum current at 24V is around 260mA, but the contactor can open from 18V/19V. While using CDR, the DC_L2H should be set to 250mA (a bit under the limit), TIME_L2H to 100ms (arbitrary big value), and DC_H to 100mA (from the data sheet, the contactor should be able to hold at around 30mA, but it seems to have problems with values smaller than 70mA. The value of 100mA is set to be a good value, with an error margin over the minimal value that is found to hold). CDR also requires PI settings to function, but these are set based on future measurement. As this is a contactor that needs low currents to be driven, 1000 can be used as starting values for P and I.

The RAMP is set to a very low value to show the effects on the current going in this contactor. The actual RAMP slew rate for CDR is based on internal current scaling factor, channel PWM frequency, and set slew rate. The RAMP is set to affect only the beginning of the contactor actuation (RUPE). This feature is developed to lower the plunger impact during the initial actuation, which lowers the sound the solenoid/contactor makes, and decreases the mechanical impact of the plunger, which can extend the life of the solenoid.

The fast demagnetization (DC_H2L) is set to −12V (0xD556 using 2’s complement) and is enabled on this channel by setting H2L_EN to HIGH. This is a functionality that requires a full-bridge and outputs a negative voltage when the channel is turned off to help with the discharge of the coil.

If using the open-load detection in full-bridge configuration, both channels must enable it, but one of the channels must have the HSnLS bit enabled. So, in this example it is set on the CH2.

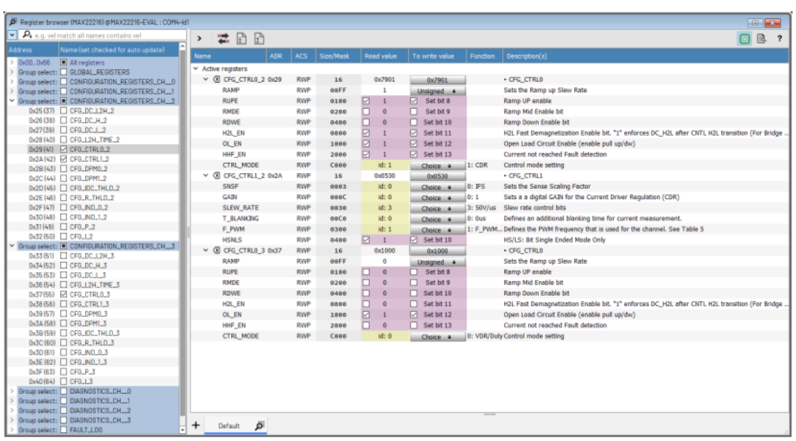


*Figure 59. Register Browser Showing the Configure Registers of CH2 and OL_EN of CH3*

**Open-Load**

- CH2 and CH3 OL_EN (ADD 0x29 MASK 1000 and ADD 0x37 MASK1000): In full-bridge, open-load must be set on both channels, and one of the channels must be set to HSnLS.
- CH2 HSnLS (ADD 0x2A MASK 0400): For OL, one of the channels must be set to use the lower MOSFET to measure OL when the channel is not in use (already set in the "Solenoid Sequencer").

**Current Not Hit Alarm**

- CH2 HHF_EN (ADD 0x20 MASK 2000): Triggers an alarm if the channel current is not reaching the target current.

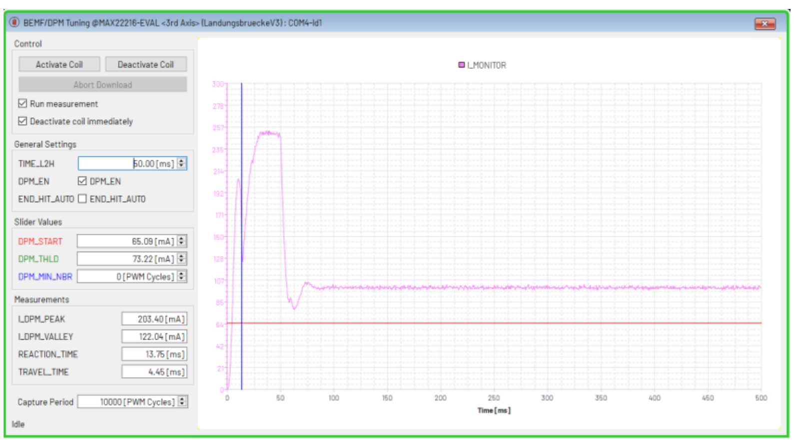
**Lower EMI**

- CH2 SLEW_RATE (ADD 0x2A MASK 0030): Set to 3, lowest slew rate for the lowest EMI. This heavily affects the current control loop and most other features. So, it should be set before PI tuning.

**Channel PWM Frequency**

- F_PWM_M (ADD 0x00 MASK 00F0): Set to 4: 40kHz – global frequency affecting all the blocks of the MAX22216 (not in the "Register Browser" image, as it is already set in the MAX22216 "General Settings").
- CH2 F_PWM (ADD 0x2A MASK 0300): Set to 1: PWM/2 (channel-specific PWM frequency divider).

In full-bridge mode, the channel is driven using center-aligned PWM scheme. So, the output PWM frequency effectively doubles. The global PWM is set to 40kHz and the channel PWM divider is set to ½. So, the effective output is 40kHz. In single-ended configuration, the same settings have the output PWM set to 20 kHz.




060

*Figure 60. MAX22216 CH1 DPM Tuning Tool Example Settings*

Now that all the registers are set, the DPM tool can be used to both set the DPM and tune the PI values. The RAMP cannot be seen very clearly in the current graph as the contactor has a charge like the slow RAMP setting.

The SLEW_RATE and channel PWM frequency affect the PI regulator. For the example, the P is updated to 40000 and the I is set to 500 for a smoother current curve. After the PI values are turned to appropriate values, the actual DPM settings can be set.

The TIME_L2H is shortened to only 50ms as the contactor does not need too much time to arrive at the set HIT current. The END_HIT_AUTO feature is not used in this example as it might interfere with the HHF feature (alarm if the DC_L2H target current is not reached).

The exported settings:

| | | | |
|---|---|---|---|
| 0x00 | = | 0x00000040 | *;; writing GLOBAL_CTRL @ address 0=0x00 with 0x00000040=64=0.0* |
| 0x01 | = | 0x00008015 | *;; writing GLOBAL_CFG @ address 1=0x01 with 0x00008015=32789=0.0* |
| 0x04 | = | 0x0000D556 | *;; writing DC_H2L @ address 2=0x04 with 0x0000D556=54614=0.0* |
| 0x25 | = | 0x000000F6 | *;; writing CFG_DC_L2H_2 @ address 3=0x25 with 0x000000F6=246=0.0* |
| 0x26 | = | 0x00000062 | *;; writing CFG_DC_H_2 @ address 4=0x26 with 0x00000062=98=0.0* |
| 0x28 | = | 0x000003E8 | *;; writing CFG_L2H_TIME_2 @ address 5=0x28 with 0x000003E8=1000=0.0* |
| 0x29 | = | 0x00007901 | *;; writing CFG_CTRL0_2 @ address 6=0x29 with 0x00007901=30977=0.0* |
| 0x2A | = | 0x00000530 | *;; writing CFG_CTRL1_2 @ address 7=0x2A with 0x00000530=1328=0.0* |
| 0x31 | = | 0x00009C40 | *;; writing CFG_P_2 @ address 8=0x31 with 0x00009C40=40000=0.0* |
| 0x32 | = | 0x000001F4 | *;; writing CFG_I_2 @ address 9=0x32 with 0x000001F4=500=0.0* |
| 0x37 | = | 0x00001000 | *;; writing CFG_CTRL0_3 @ address 10=0x37 with 0x00001000=4096=0.0* |

## Setup Example: DC Motor

**Load:** 12V Brushed DC Motor

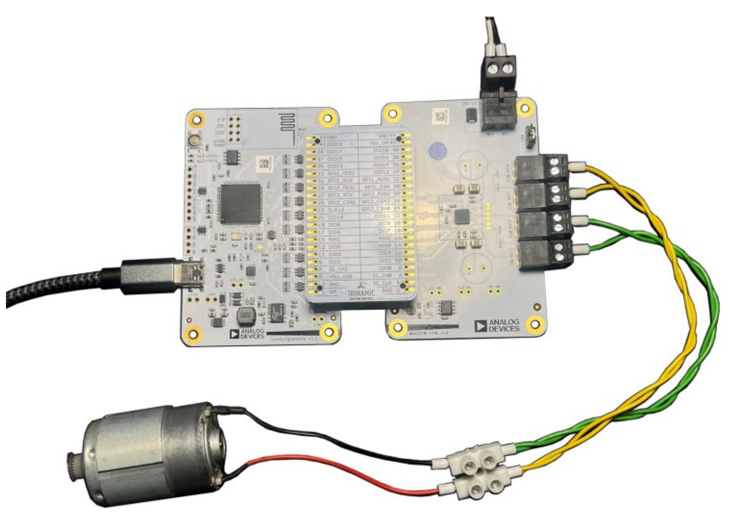
**Setup:** Load Between OUT0OUT1 and OUT2OUT3

**PSU:** 24V

In this example setup, the MAX22216 drives a 12V brushed DC motor. The MAX22216 has a special drive mode, where the output is driven with a set voltage (DC_H), and in parallel the MAX22216 creates a current limit with a set threshold (DC_L2H). Brushed DC motors have a big inrush current at startup and the current consumption increases based on motor load. The current limiter should be set to a current level above the maximum current consumption of the motor under load. The motor mode also has a brake feature (TIME_L2H), where if both channels of the full-bridge configuration are set to output, the MAX22216 forces a set current in the opposite direction of the previous current flow, which forces the motor to stop.

The STAT function can be set to monitor the measured current and flag when it goes above a set threshold.

This example also uses the CRC integrated in the MAX22216 and the TMCL-IDE automatically does all the calculations to ensure there are no errors in the communication with the part.



061

*Figure 61. Example Setup Using a Brushed DC Motor*

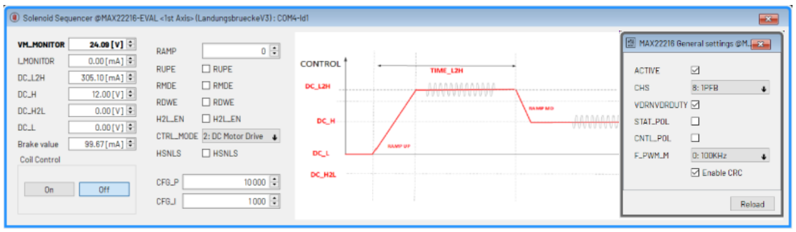
**(Example) Required Functionality:**

**Motor Mode:** Special motor drive functionality with current limiter.

**Brake:** Special feature of motor mode to stop the motor.

**STAT Current Observer:** Triggers a STAT flag when the current goes over a set threshold (it can be set under the current limiter value and can be used to detect if the motor stalls while in use).

**CRC:** Check for SPI communication errors.



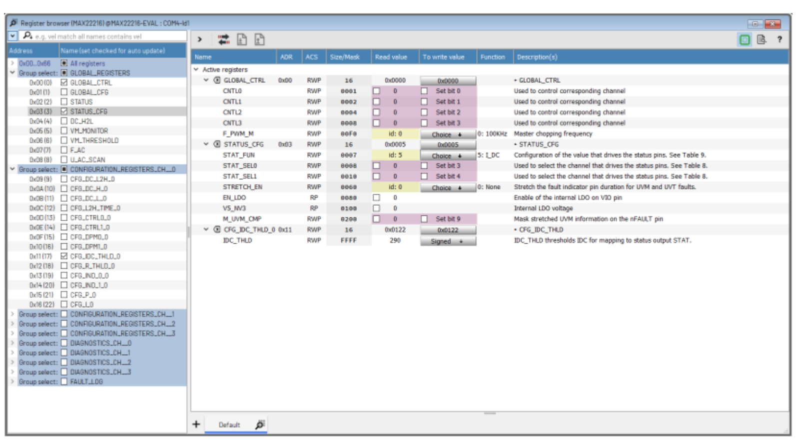
062

*Figure 62. MAX22216 General Settings and CH0 Solenoid Sequencer Example Settings for Brushed DC Motor Example*

In the “General Settings”, the channels are set to a parallel full-bridge configuration, where CH0 and CH1 are controlled as one side of the bridge, and CH2 and CH3 are the other. This setup requires the sole of the channels to be connected outside of the eval (CH0 to CH1 and CH2 to CH3), which is done using cables in this example. The MAX22216 is also set to use the internal voltage regulator (VDRnVDRDUTY) for the voltage output.

The CRC inside the MAX22216 can be used by setting the CRC pin to high. In the TMCL-IDE, the CRC can be used by setting the “Enable CRC” box in the “General Settings”. No other settings are required. The MAX22216EVKIT sets the CRC pin to high and automatically changes the communication and calculates the CRC.

In the “Solenoid Sequencer”, the “Control Mode” is set to “DC Motor Drive”. The 12V motor can consume a maximum of 300mA under load, but without a load it consumes around 100mA. The output voltage (DC_H) is set to 12V, the current limiter (DC_L2H) is set to around 300mA, and the brake current (TIME_L2H) is set to around the consumption of the motor in this application (100mA with no load, but if there is a big inertia, it might need to be bigger than the nominal current consumption). The P and I values are initially set to 1000 for both, but after further testing, the P is increased. The testing can be done using the BEMF/DPM tuning tool or the “Parameter & Register Scope” set to monitor the current register (for CH0, register 0x45: I_MONITOR).



063

*Figure 63. Register Browser Showing the STAT Settings, Current Observer Threshold, and Channels Control*

**STAT Current Observer:**

- STAT_FUN (ADD 0x03 MASK 0007): The STAT is set to 5:I_DC, which monitors the current and triggers an alarm if the current is higher than a set current threshold (IDC_THLD).
- IDC_THLD (ADD 0x11 MASK FFFF): Current threshold for STAT I_DC, compared directly to the current register. In this example, the alarm is set to 290 (~295mA), a bit under the set current limiter value (305mA).

**Channel Control:**

To use the BRAKE of the DC Motor Drive, both output channels must be set high. In the CHS 8 configuration, the two control channels are CH0 and CH1 (in this mode, the control channels are different from the actual output channels). To use the BRAKE, the simplest way is to set both CNTL0 and CNTL1 in the GLOBAL_CONTROL to HIGH, but it can also trigger if GUI tools from both channels turn on the channels (for example, the "Solenoid Sequencer" from both CH0 and CH1 set the channels on).

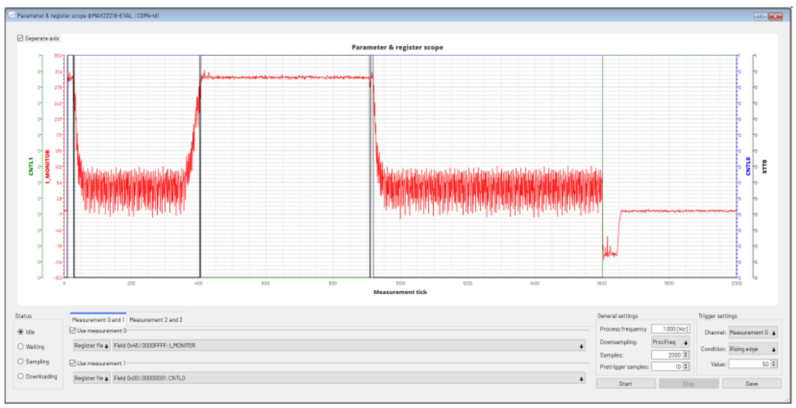



*Figure 64. Parameter & Register Scope Set to Monitor the CH0 Current, CNTL0, CNTL1, and STAT Alarm*

The scope shot is divided into four sections:

- **Startup and normal use:** CNTL0 is turned HIGH. At startup, the inrush current is cut by the current limiter and the STAT is triggered. After the motor starts rotating, the current consumption is lowered, and shows periodic sine-like impulses, created by the brushes of the DC motor when it is changing contacts.
- **Motor stall:** The motor is blocked to simulate a stuck rotor and STAT alarm is triggered. Here, an external microcontroller can check the channel current and STAT alarm and know the motor is blocked.
- **Motor unblocked:** Motor can go back to normal operations and the STAT alarm turns OFF.
- **Motor BRAKE:** Both channels are set to HIGH (CNTL0 and CNTL1) and the MAX22216 creates a negative 100mA output on the line to stop the motor.

The STAT is set as an alarm for motor stall, using the current compare functionality (STAT_FUN 5: I_DC) with a threshold value (IDC_THLD) under the maximum recommended current (which, in thisr case, is also the current limit DC_L2H). This STAT function can be set to output a signal on the STAT pin, and a microcontroller can check for this signal and detect the motor stall. In full-bridge mode, the registers related to the STAT alarm configuration must be set on both channels. To make the STAT pin output when the motor is in stall, the IDC_THLD must be set to the appropriate value (290 or 0x0122) on both CH0 (CH0 IDC_THLD: ADD 0x11) and CH1 (CH0 IDC_THLD: ADD 0x1F).

The exported settings:

| | | |
|---|---|---|
| 0x01 = | 0x00008018 | *;; writing GLOBAL_CFG @ address 0=0x01 with 0x00008018=32792=0.0* |
| 0x03 = | 0x00000005 | *;; writing STATUS_CFG @ address 1=0x03 with 0x00000005=5=0.0* |
| 0x09 = | 0x0000012C | *;; writing CFG_DC_L2H_0 @ address 2=0x09 with 0x0000012C=300=0.0* |
| 0x0A = | 0x00002AAA | *;; writing CFG_DC_H_0 @ address 3=0x0A with 0x00002AAA=10922=0.0* |
| 0x0C = | 0x00000062 | *;; writing CFG_L2H_TIME_0 @ address 4=0x0C with 0x00000062=98=0.0* |
| 0x11 = | 0x00000122 | *;; writing CFG_IDC_THLD_0 @ address 5=0x11 with 0x00000122=290=0.0* |
| 0x15 = | 0x00002710 | *;; writing CFG_P_0 @ address 6=0x15 with 0x00002710=10000=0.0* |
| 0x16 = | 0x000003E8 | *;; writing CFG_I_0 @ address 7=0x16 with 0x000003E8=1000=0.0* |
| 0x1F = | 0x00000122 | *;; writing CFG_IDC_THLD_1 @ address 29=0x1F with 0x00000122=290=0.0* |

The “CRC Enable” setting is not exported as it is controlled using a digital input pin.

## Detailed Description of Hardware

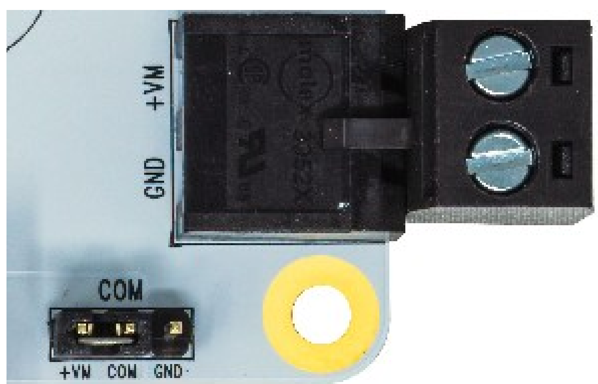
All design files for ADI TRINAMIC evaluation boards are available for free. The original ECAD files, Gerber data, the BOM, and PDF copies are available. Typically, the ECAD files are in KiCAD format. Some (older) evaluation boards may only be available in Eagle, Altium, or PADS format. Check the schematics for jumper settings and input/output connector descriptions.

Download the files from the ADI TRINAMIC evaluation boards page under EVALUATION DESIGN FILES.

*MAX22216EVKIT Evaluation Board*

### Onboard Jumper

The MAX22216-EVAL evaluation board has one jumper to change the COM of all of the four outputs between +VM and GND. This setting is needed if the outputs are used in the half-bridge mode.




065

*Figure 65. Jumper for COM Signal Next to the Supply Plug*

Adapt the HSnLS (high-side not low-side) setting for each used half-bridge accordingly. The COM jumper sets the COM pin of the connector for all the connectors to either +VM or GND. The default COM jumper setting is +VM and matches the MAX22216 HSnLS = false default.

## Table 17. COM Jumper Configuration

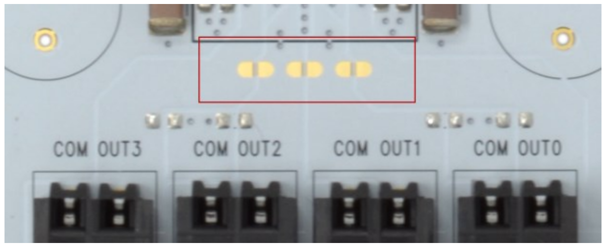
| COM CONFIGURATION | OUTPUT CONNECTOR COM | MAX22216 SETUP FOR SINGLE-ENDED USE |
|---|---|---|
| COM to +VM (Default) | +VM | HSnLS = LOW (Default) |
| COM to GND | GND | HSnLS = HIGH |

To use one output in HSnLS = HIGH mode, connect one terminal of the load to the OUTx pin and the other terminal to the COM output with the COM jumper set to COM-GND, or directly to the GND of the supply.

### Onboard Options

#### Solder Bridges

There are three solder bridges (SB301, SB302, and SB303) near the MAX22216. They can be used to bridge outputs for bridge-tied load (BTL) operation without the need of external wiring. Make sure to use the correct CHS within the general settings tool. Change the CHS first and activate the part afterwards.




066

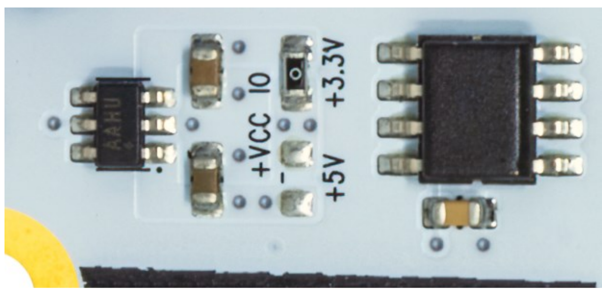
*Figure 66. Three Solder Bridges (Red Rectangle) and Four Optional SMD Capacitors Near the Outputs*

### Capacitors

As shown above, there are four positions next to the outputs for optional 0603 surface-mount device (SMD) capacitors (C203, C204, C205, and C206). Two THT EL100 electrolytic capacitors are also optional next to the MAX22216.

### Voltage Selection

If the MAX22216 VIO (+VCC_IO on this evaluation board) is used with +5V instead of +3.3V, there is a solder selection near the EEPROM. The selection should be changed if an external electronic with 5V levels is connected.




067

*Figure 67. +VCC_IO Selection Near the EEPROM*

Do not bridge both selections at the same time. This can disturb the onboard voltage regulator.

In the rare case of the OTP output voltage being used, neither selection should be present. This happens when the MAX22216-EVAL is started after the VIO output is configured by the OTP setting.

## Onboard Connectors

MAX22216-EVAL has seven onboard connectors. The following table contains information on the connector type and mating connectors. The connector pinning and signal names can be derived from the board design and schematic files.

### Table 18. VCC_IO Selection Near the EEPROM

| NUMBER | CONNECTOR | CONNECTOR TYPE | DESCRIPTION |
|---|---|---|---|
| 1 | Power Supply | *MOLEX 0395221002* | Connects a battery or power supply to the evaluation board. An example of a mating connector is *MOLEX 0395200002*. |
| 2 | 4x Load | *MOLEX 0395021002* | Connects the loads to the MAX22216 outputs. An example of a mating connector is *MOLEX0395000002*. |
| 3 | Landungsbruecke | 46-3492-44-3-00-10-PPTR from *W+P Series 3492* | Main I/O and digital supply connector to connect to the Landungsbruecke controller board through the Eselsbruecke connector or to connect to an own controller board. |

| NUMBER | CONNECTOR | CONNECTOR TYPE | DESCRIPTION |
|---|---|---|---|
| 4 | COM | Standard 2.54mm header | Use to connect COM to +VM or GND through a jumper. |

## Landungsbruecke Connector

All signals are connected to the MAX22216 directly without any additional protection. Refer to the MAX22216 data sheet for electrical ratings.

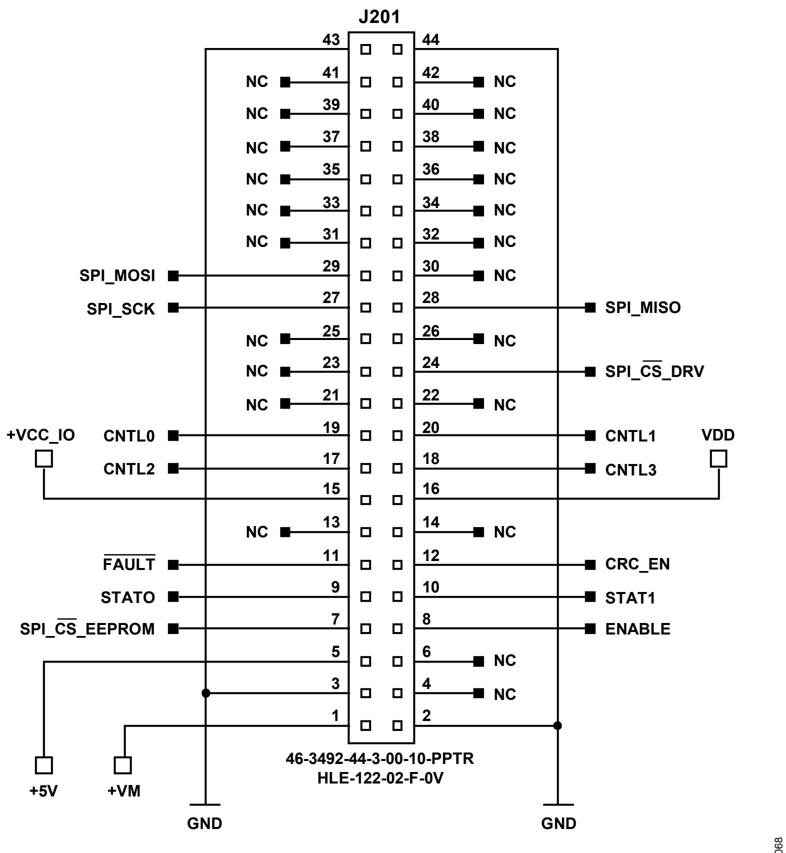



*Figure 68. Pin Assignment on Landungsbruecke Connector*

## Load Connector

The MAX22216-EVAL consists of four plugs to attach loads. Each of the four plugs has one COM and one OUT_ connection. All COM pins are connected to the selection jumper (see the **Onboard Jumper** section) and each OUT_ is connected to the MAX22216 directly. Check the currently selected channel hardware settings (CHS) using “General Settings” or “Register Browser”.

### Half-Bridge Usage

IHB shows individual half-bridges. PHB shows parallel half-bridges. Connect the load to the output terminals COM and OUT_. COM is set globally by the jumper for all outputs. If a parallel half-bridge configuration is set, connect the corresponding OUT_ together externally or use the solder bridges.

### Full-Bridge Usage

IFB shows individual full-bridges. PFB shows parallel full-bridges. Connect the load to the output terminals that form the full-bridge (example, OUT0 and OUT1 at CHS = 0x05). COM has no influence and is therefore not used. If a parallel full-bridge configuration (CHS = 0x08) is set, connect OUT0 to OUT1 and OUT2 to OUT3 externally or use the solder bridges. The load is then connected between OUT0 to OUT1 and OUT2 to OUT3.

### Load Connector Example

In this example, assume CHS = 0x07, which is 1IFB_2PHB and translates to 1x full-bridge at OUT0 and OUT1 with 1x parallel half-bridge at OUT2 and OUT3. Connect the first load to OUT0 and OUT1. Connect OUT2 and OUT3 to each other and the second load between this OUT2 to OUT3 connection and COM. The low-side and high-side behavior of OUT2 to OUT3 is controlled by the onboard jumper and the HSnLS register-field of OUT2.

## EV Kit Package

While observing safe ESD practices, carefully remove the Landungsbruecke, Eselsbruecke, and MAX22216-EVAL boards out of the packaging. Quickly inspect the boards to ensure no damage occurred during shipment. Jumpers/shunts and needed output and power connectors are preinstalled before testing and packaging.

## Table 19. MAX22216EVKIT Content

| ITEM | DESCRIPTION |
|---|---|
| MAX22216-EVAL | MAX22216 Evaluation Board |
| Landungsbruecke | PC Interface Board |
| Eselsbruecke | Bridge Connection Board |

## Power Supply

This MAX22216EVKIT can be powered trough the VM power connector. The mating connector is provided in the kit, already plugged in the VM connector.

The +VM input range is 4.5V to 36V, but it is recommended to start with a 12V/24V power supply. The MAX22216 is designed to function only while power is supplied and cannot store settings when it is powered off (except if the register settings are set using OTP).

For OTP, the VM power supply must be 8.7V (± 0.13V).

## Ordering Information

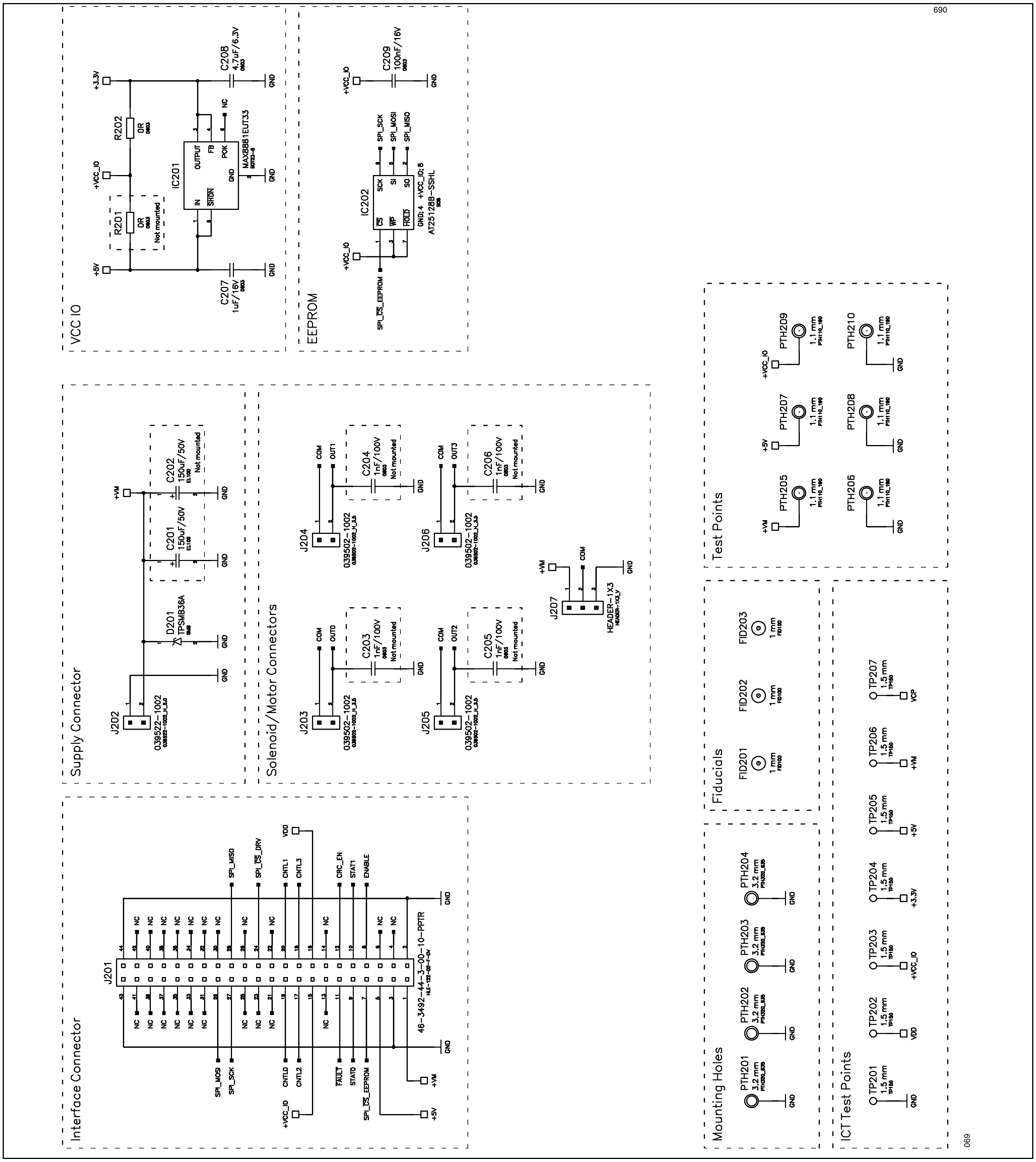
| PART | TYPE |
|---|---|
| MAX22216EVKIT# | Evaluation Kit |

*#Denotes a RoHS-compliance.*

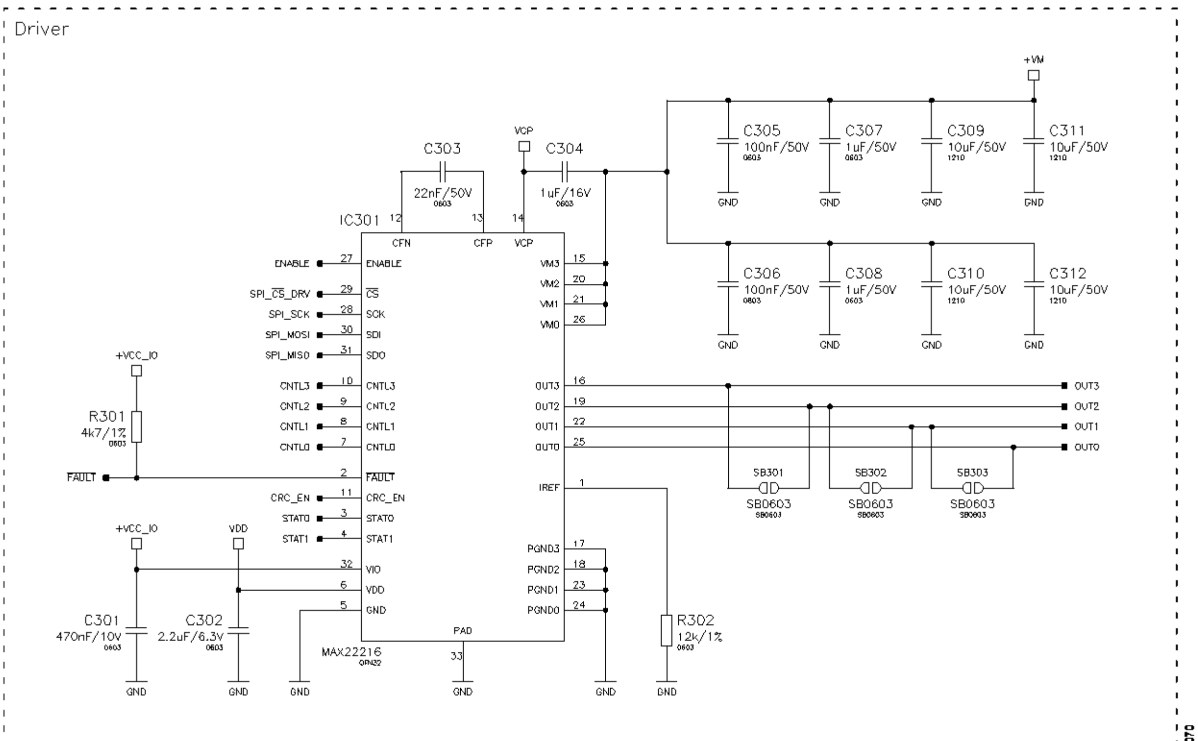
## MAX22216EVKIT Bill of Materials

| PART | QUANTITY | PACKAGE | DESCRIPTION |
|---|---|---|---|
| R202 | 1 | 0603 | 0R |
| C207, C304 | 2 | 0603 | 1µF/16V |
| C307, C308 | 2 | 0603 | 1µF/50V |
| C302 | 1 | 0603 | 2.2µF/6.3V |
| C208 | 1 | 0603 | 4.7µF/6.3V |
| R301 | 1 | 0603 | 4k7/1% |
| C309 to C312 | 4 | 1210 | 10µF/50V |
| R302 | 1 | 0603 | 12k/1% |
| C303 | 1 | 0603 | 22nF/50V |
| J201 | 1 | HLE-122-02-F-DV | 46-3492-44-3-00-10-PPTR |
| C209 | 1 | 0603 | 100nF/16V |
| C305, C306 | 2 | 0603 | 100nF/50V |
| C301 | 1 | 0603 | 470nF/10V |
| IC202 | 1 | SO8 | AT25128B-SSHL |
| J207 | 1 | Header-1X3_V | Header-1X3 |
| IC201 | 1 | SOT23-6 | MAX8881EUT33 |
| IC301 | 1 | QFN32 | MAX22216 |
| J203 to J206 | 4 | 039502-1002_H_3.5 | 039502-1002 |
| J202 | 1 | 039522-1002_H_5.0 | 039522-1002 |
| D201 | 1 | SMB | TPSMB36A |
| R201 | 1 | 0603 | Not Mounted - 0R |
| C201, C202 | 2 | EL100 | Not Mounted - 150µF/50V |
| C203 to C206 | 4 | 0603 | Not Mounted - 1nF/100V |
| Connect to J202 | 1 | – | 039520-0002 |
| Connect to J203 to J206 | 4 | – | 039500-0002 |
| Connect to J207 Pins 1 to 2 | 1 | – | Jumper |

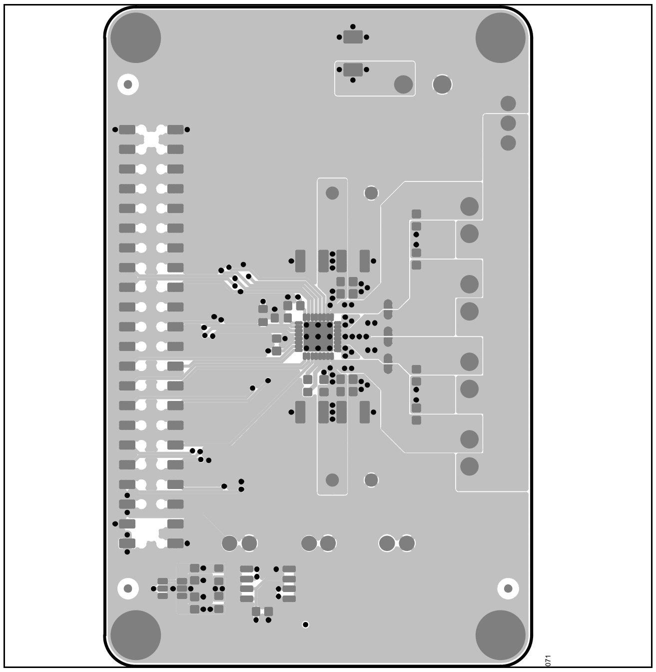
## MAX22216EVKIT Schematic



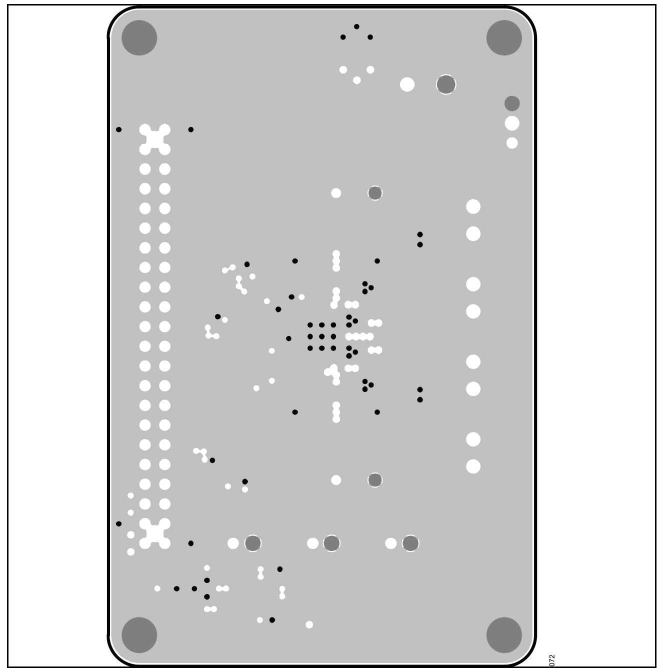
## MAX22216EVKIT Schematic (continued)



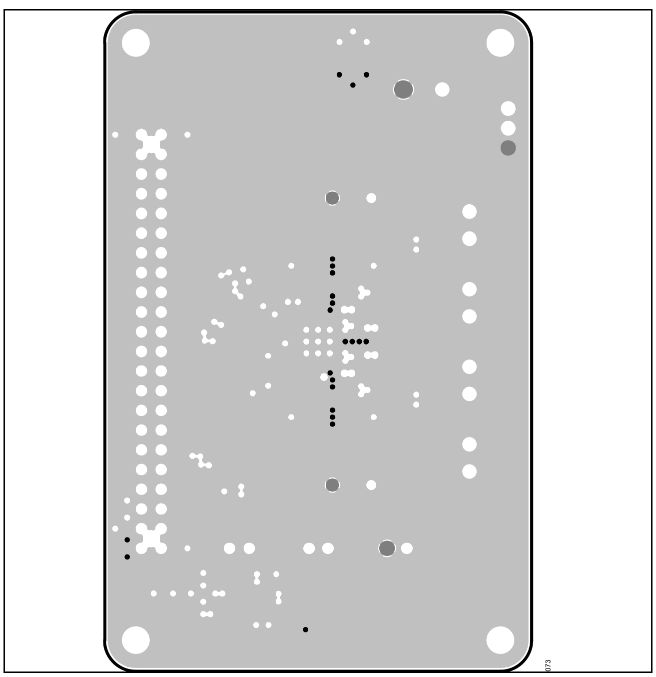
## MAX22216EVKIT PCB Layout



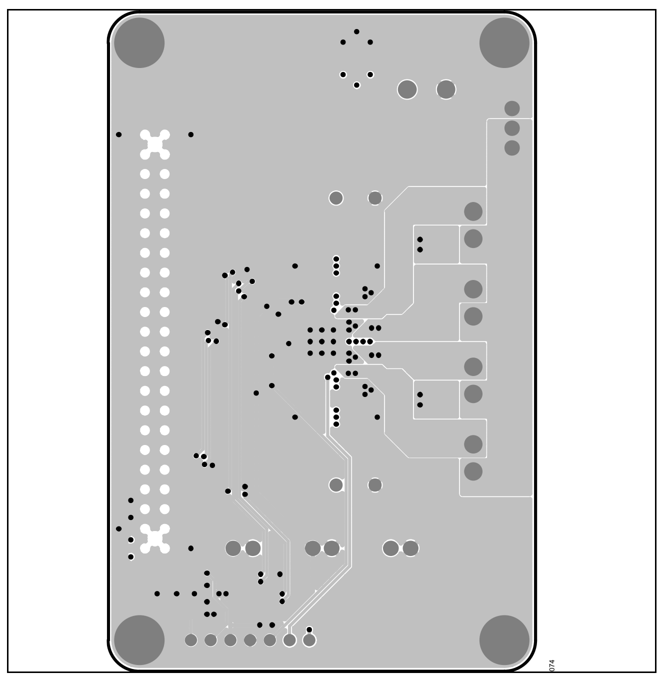
*MAX22216EVKIT PCB Layout—Layer 1 (Top)*



*MAX22216EVKIT PCB Layout—Layer 2*

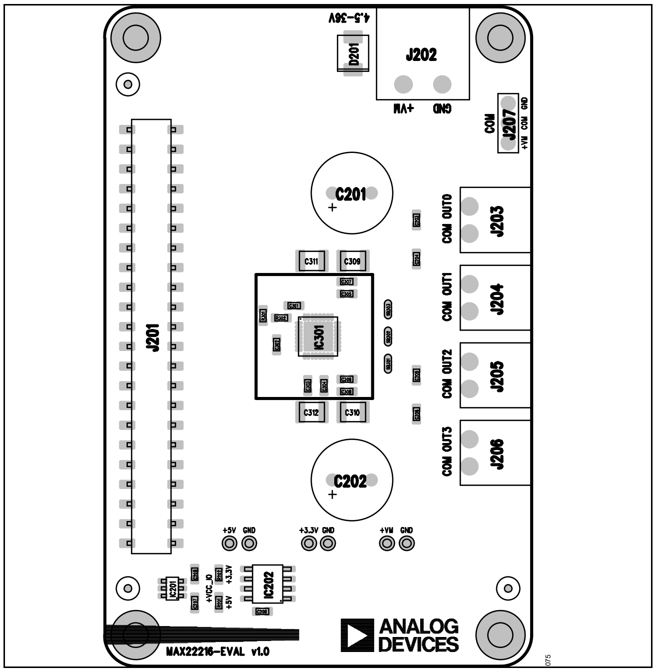


*MAX22216EVKIT PCB Layout—Layer 3*

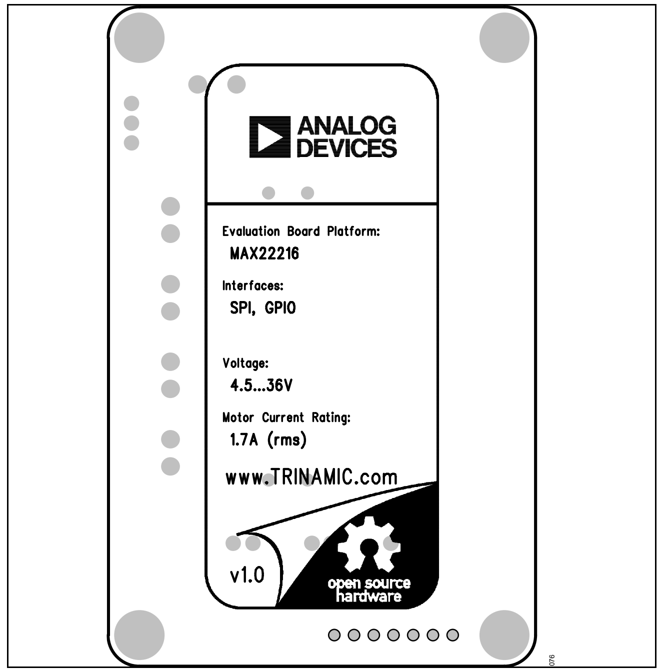


*MAX22216EVKIT PCB Layout—Layer 4 (Bottom)*

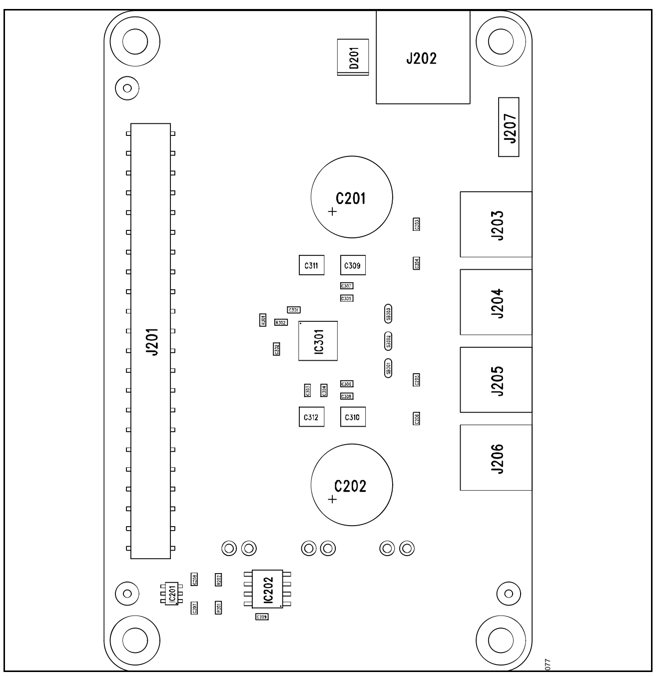
## MAX22216EVKIT PCB Layout (continued)



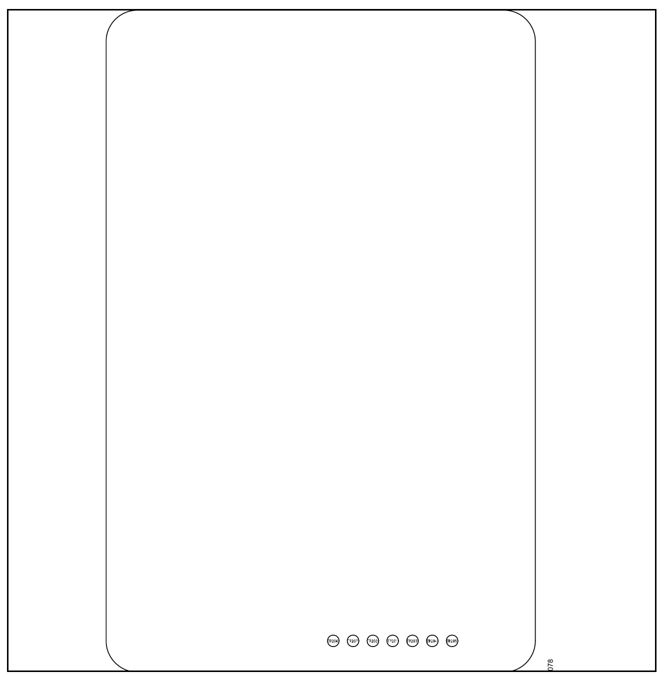

MAX22216EVKIT Silkscreen—Top




MAX22216EVKIT Silkscreen—Bottom




MAX22216EVKIT Component Placement Guide—Top



MAX22216EVKIT Component Placement Guide—Bottom

## Revision History

| REVISION NUMBER | REVISION DATE | DESCRIPTION | PAGES CHANGED |
|---|---|---|---|
| 0 | 10/23 | Initial release | — |
| 1 | 3/24 | Added MAX22217 support explanation. | — |
| 2 | 2/25 | Added and updated sections across the document. | — |
| 3 | 9/25 | Updated the document title. Updated the General Description, Features and Benefits, Ordering Information, Procedure, and Setup and Operation sections. | 1, 2, 6 to 10, 61 |

## Notes