IQS7221E GUI Setup Guide

User Guide to Configure the IQS7221E Using the Debug and Display Tool

Contents

1 Introduction

The IQS7221E is a Hall encoder IC with applications ranging from control and volume knobs to mouse scroll-wheels. It contains four Hall plates to measure the magnetic field of a nearby diametrically-polarised magnet, and calculates the angle of the magnet. It also features a capacitive-sensing channel to detect user interaction, and to support the virtual Freewheel UI.

This document describes how to use the IQS7221E Debug and Display Tool / Graphical User Interface (GUI) to configure the device. The software can stream all on-chip data in real time, and provides convenient access to all the settings of the IC. This document will explore which settings to change to achieve the desired behaviour of the device, including how to set up the Hall-rotation sensing, capacitive touch sensing, and virtual freewheeling function.

This guide is intended to be used alongside the IQS7221E Hall-Effect Evaluation Kit (EV kit) to explain many of the concepts of the IQS7221E and its various features and settings. The EV kit illustrates the use of the Hall encoder on a control knob, and supports a capacitive touch channel on the metal knob to detect user presence and interaction.

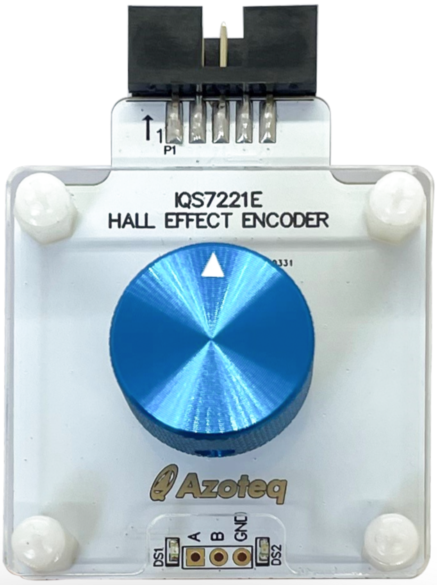


Figure 1.1: IQS7221E Hall-Rotation Evaluation Kit

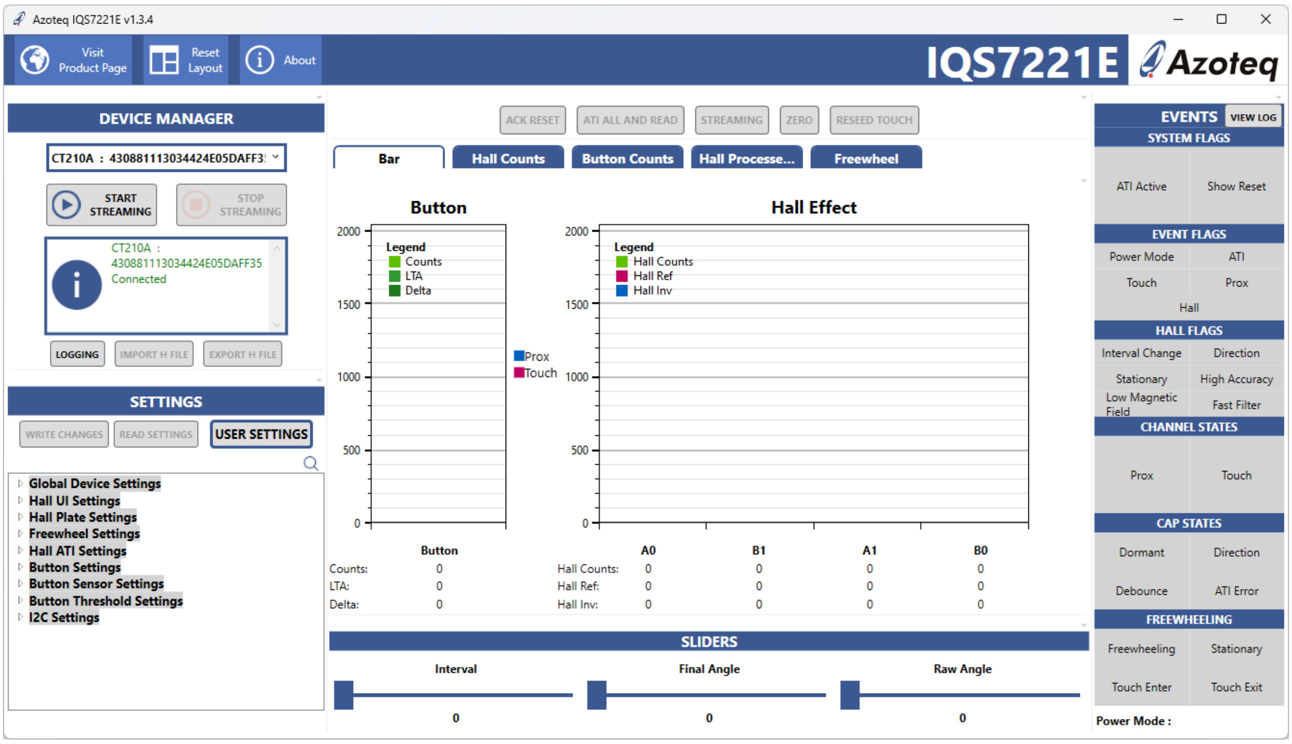
2 IQS7221E GUI Installation and Quick-Start Guide

This section provides a brief explanation of the installation of the IQS7221E GUI, as well as how to get started with the IQS7221E evaluation kit.

2.1 Installation

The installation package for the IQS7221E Debug and Display Tool can be found on the IQS7221E's product page, under the Design Support tab.

Extract the installer from the downloaded zip file and follow the installation wizard. Once installation is complete, launch the IQS7221E software, and the window shown in Figure 2.1 should appear.



CT210A : 430881113034424E05DAFF3

CT210A : 430881113034424E05DAFF35 Connected

- Global Device Settings
- Hall UI Settings
- Hall Plate Settings
- Freewheel Settings
- Hall ATI Settings
- Button Settings
- Button Sensor Settings
- Button Threshold Settings
- I2C Settings

	Button
Counts:	0
LTA:	0
Delta:	0

	A0	B1	A1	B0
Hall Counts:	0	0	0	0
Hall Ref:	0	0	0	0
Hall Inv:	0	0	0	0

Interval	Final Angle	Raw Angle
0	0	0

Power Mode :

Figure 2.1: IQS7221E Graphical User Interface Overview

2.2 Quick-start Guide

The IQS7221E EV kit can be connected to a computer via the CT210A configuration tool. Simply connect the CT210A to a computer with a micro USB data cable, and connect the EV kit to the CT210A via the included ribbon cable.

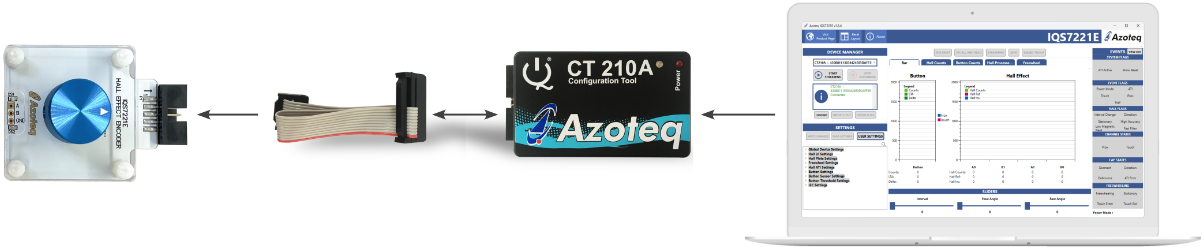


Figure 2.2: Hardware Connection for Streaming and Testing

The CT210A can also be connected to the EV kit directly with the following pins:

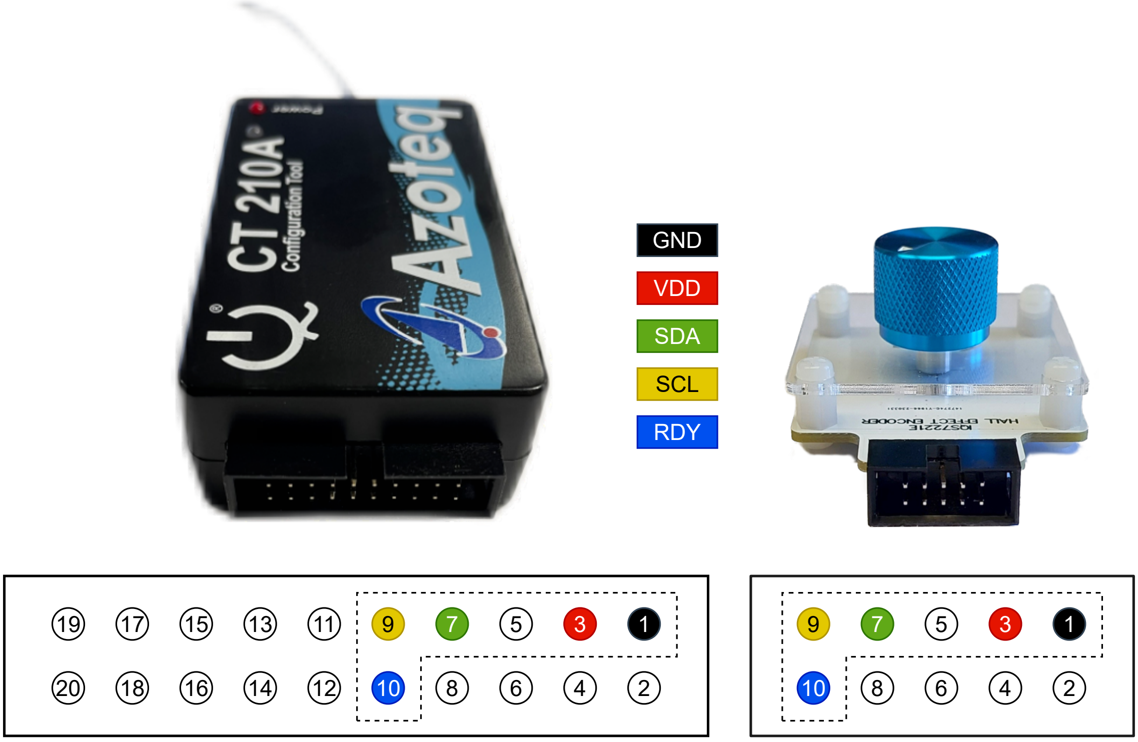
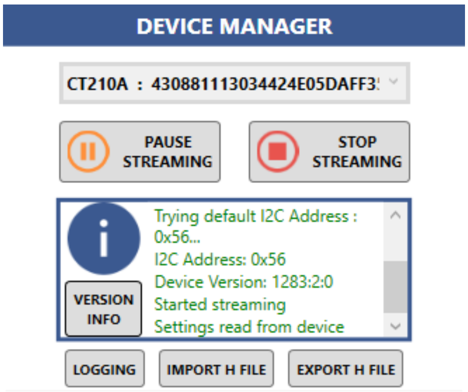


Figure 2.3: CT210A Pinout

Once the IQS7221E is connected to the CT210A, follow these steps to begin streaming data.

Step 1: To connect to the IQS7221E in the software, click the “Start Streaming” button. This will provide power to the IQS7221E and begin communication. If the connection is successful, the information panel below the Start Streaming button will show the I^{2}C address and device version.



Trying default I2C Address : 0x56...
I2C Address: 0x56
Device Version: 1283:2:0
Started streaming
Settings read from device

Figure 2.4: IQS7221E Connected Message

Step 2: Click the “Ack Reset” button to clear the reset flag on the IQS7221E.

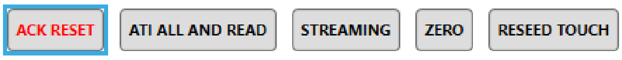


Figure 2.5: Ack Reset Button

Step 3: Click the “Streaming” button to enter I^{2}C streaming mode.

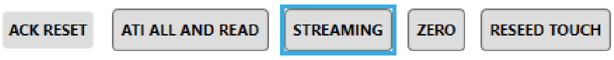
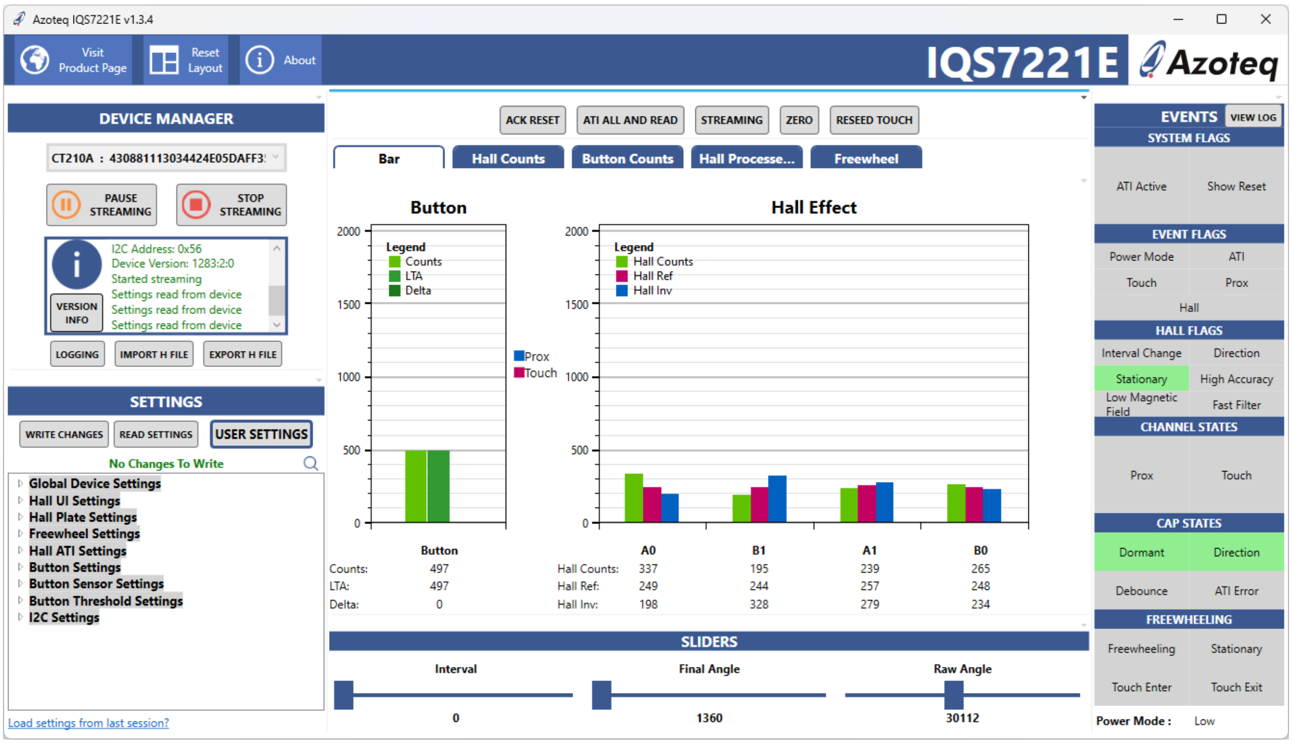


Figure 2.6: Enable Streaming Button

The IQS7221E will now stream all sensor data, as shown in Figure 2.7. The counts for the various channels should change as the control knob is rotated.



I2C Address: 0x56
Device Version: 1283:2:0
Started streaming
Settings read from device
Settings read from device
Settings read from device

	Button
Counts:	497
LTA:	497
Delta:	0

	A0	B1	A1	B0
Hall Counts:	337	195	239	265
Hall Ref:	249	244	257	248
Hall Inv:	198	328	279	234

Interval	Final Angle	Raw Angle
0	1360	30112

Power Mode : Low

Figure 2.7: IQS7221E Streaming

2.3 Device Version

When starting streaming, the IQS7221E version number is printed in the Device Manager information panel, as seen in Figure 2.4.

- Device version `1283:1:1` corresponds to *IQS7221E001* product code.
- Device version `1283:2:0` corresponds to *IQS7221E101* product code.

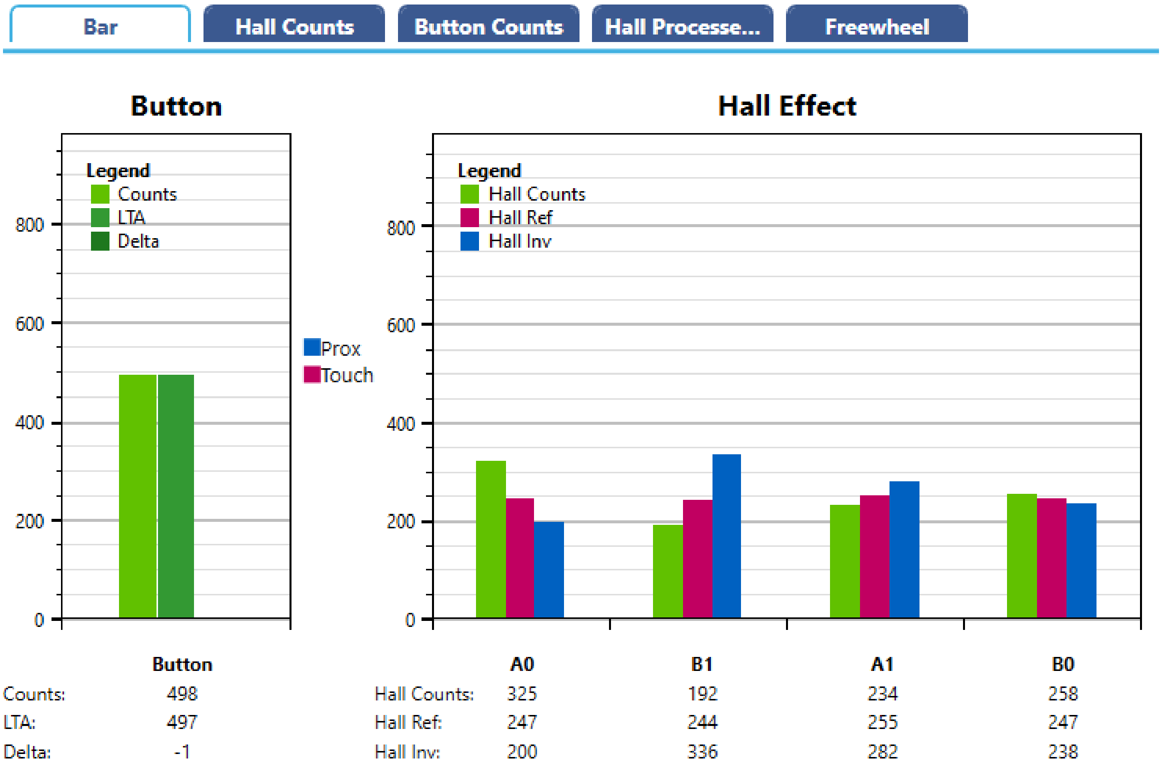
Please refer to the order codes section in the datasheet for more information.

3 IQS7221E Debug and Display Software Overview

This section briefly explains the GUI elements such as the sensor graphs, device events, and commands, as well as some additional core functionality such as data logging and exporting of device settings.

3.1 IQS7221E Streaming Data

The IQS7221E GUI displays all the streaming data in the graph panel in the centre of the GUI. The default graph view is the bar graph, which plots the instantaneous counts of each channel. There are four additional scope views that plot additional information over time. These are explained later in this document, where relevant.



	Button
Counts:	498
LTA:	497
Delta:	-1

	A0	B1	A1	B0
Hall Counts:	325	192	234	258
Hall Ref:	247	244	255	247
Hall Inv:	200	336	282	238

Figure 3.1: Streaming Graphs

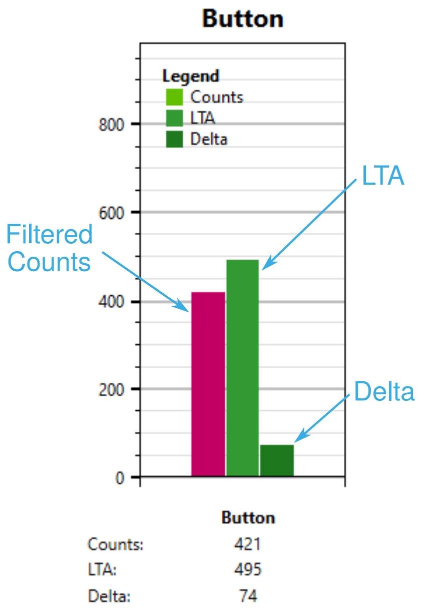
The graph views can be manipulated with the following controls:

- Scroll wheel to zoom in and out.
- Hold and drag middle-mouse button to zoom to a bounding box.
- Hold and drag right-mouse button to pan.
- Double left-click to reset the graph view.

The IQS7221E has five channels in total; one capacitive “Button” or “touch” channel, and four “Hall” channels, one for each of the Hall plates in the IC. The “counts” of a channel is a representation of the signal strength measured by the sensor.

All the signals recorded in the graphs and sliders are read directly from the IC. For more information regarding the register map, please consult the IQS7221E datasheet.

3.1.1 Button Channel



	Button
Counts:	421
LTA:	495
Delta:	74

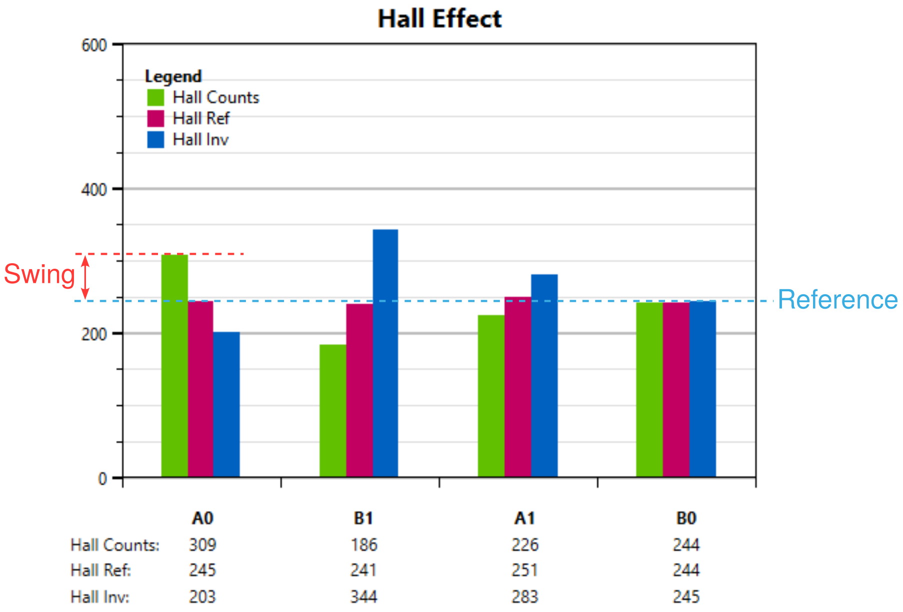
Figure 3.2: Button Channel Counts

The Button channel shows the counts of the capacitive/touch sensor. The **counts** value shows the raw measurement of the sensor, after filtering. The **LTA** is the Long Term Average of the counts signal. It tracks slow variations in the environment, and is used as a reference to detect movement. The **delta** is simply the difference between the LTA and the counts, and is used to detect activity or movement.

In the default configuration of the sensor, a user touching the sensor causes the counts to decrease. On the EV kit, the aluminium knob is used as the capacitive-sensing electrode, so touching it will cause the counts to dip on the channel. In this way, the IQS7221E can detect the proximity of the user's hand before the knob is rotated, allowing it to wake up and transition to a faster sampling rate to measure the rotation more accurately.

The Button channel has two activation levels — **Prox** and **Touch** — which are applied to the delta. These activation thresholds are configurable in the settings, described later in Section 6.3.

3.1.2 Hall Channels



	A0	B1	A1	B0
Hall Counts:	309	186	226	244
Hall Ref:	245	241	251	244
Hall Inv:	203	344	283	245

Figure 3.3: Hall Channel Counts

The four Hall plates are designated as **A0**, **A1**, **B0**, and **B1**. Each plate shows three sets of counts: **Hall counts**, **Hall inverted counts**, and **Hall reference**.

The Hall counts value represents a raw measurement of the vertical (or Z-axis) magnetic field strength at each plate. The Hall inverted counts value (Hall Inv) represents a second measurement on the same Hall plate, but with the polarity of the measurement inverted. Finally, the Hall reference (Hall Ref) represents the midpoint between the two measurements; a DC value around which the normal and inverted counts swing. This Hall reference effectively represents the counts of the Hall plate with *no external magnetic field*.

These four Hall plates are used to calculate the absolute angle of a nearby magnet.

3.1.3 Angle Measurements

Three sliders at the bottom of the GUI show the output of the Hall encoder.

- **Raw Angle** shows the magnet angle calculated directly from the Hall counts.
- **Final Angle** shows the output angle as a 16-bit value, after filtering and auto zero compensation. This value may be significantly different from the raw angle, as an offset is added whenever the Hall sensor is "zeroed".
- **Interval** shows the coarse angle output of the Interval UI, and is derived from the final angle. This is the value that would typically be used if a discrete number of positions per rotation is required, such as in a mouse scroll wheel that uses 24 intervals per rotation.

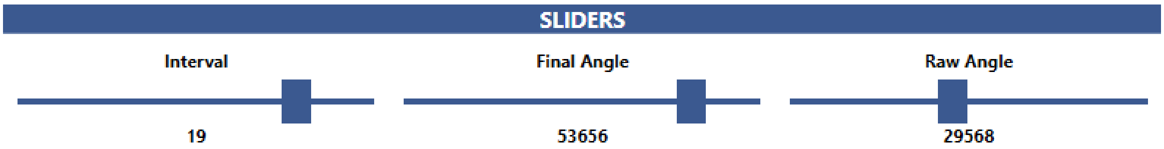
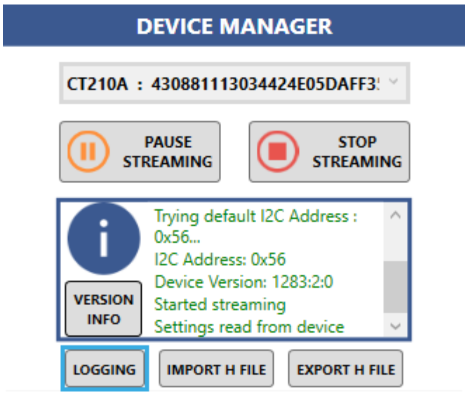


Figure 3.4: Angle Measurement Sliders

3.2 Data Logging

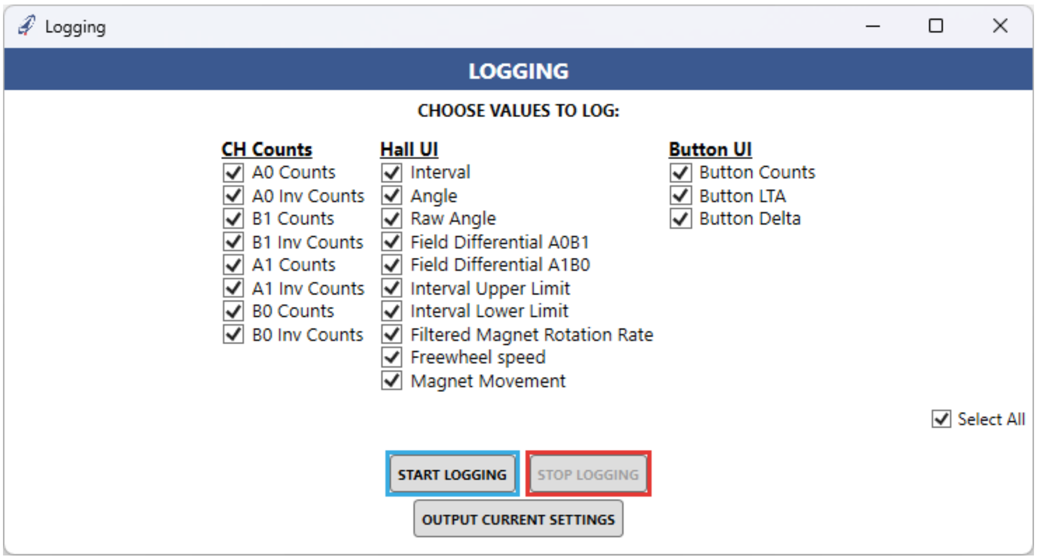
It may be necessary to save all the above streaming data to a file for debugging or testing purposes. The logging function allows the GUI to save all streaming data from the IQS7221E to a CSV file. Click the "Logging" button in the Configuration Tool Manager panel to open the logging window.



```
Trying default I2C Address :
0x56...
I2C Address: 0x56
Device Version: 1283:2:0
Started streaming
Settings read from device
```

Figure 3.5: Logging Function Using the Configuration Tool Manager

From here, the desired variables from the IQS7221E can be enabled or disabled. To start logging, click the "Start Logging" button, and choose the destination of the CSV log file.



Logging

LOGGING

CHOOSE VALUES TO LOG:

CH Counts

- ☑ A0 Counts
- ☑ A0 Inv Counts
- ☑ B1 Counts
- ☑ B1 Inv Counts
- ☑ A1 Counts
- ☑ A1 Inv Counts
- ☑ B0 Counts
- ☑ B0 Inv Counts

Hall UI

- ☑ Interval
- ☑ Angle
- ☑ Raw Angle
- ☑ Field Differential A0B1
- ☑ Field Differential A1B0
- ☑ Interval Upper Limit
- ☑ Interval Lower Limit
- ☑ Filtered Magnet Rotation Rate
- ☑ Freewheel speed
- ☑ Magnet Movement

Button UI

- ☑ Button Counts
- ☑ Button LTA
- ☑ Button Delta

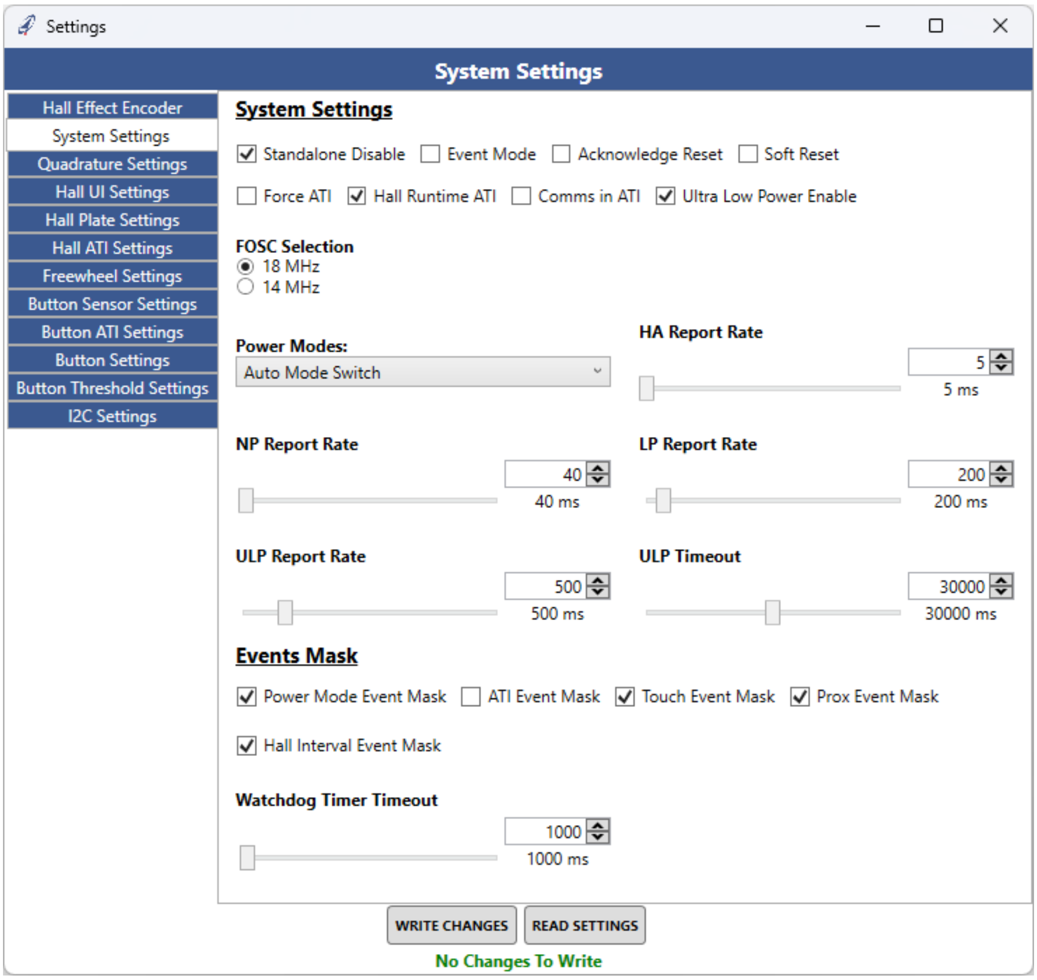
☑ Select All

Figure 3.6: Logging Configuration Window

Once the file destination is confirmed, data logging will begin. To stop logging, click the “Stop Logging” button.

3.3 Modifying IQS7221E Settings

To view all the IQS7221E device settings, click the “User Settings” button. This will open a new window where all the device settings can be modified.



System Settings

☑ Standalone Disable □ Event Mode □ Acknowledge Reset □ Soft Reset

□ Force ATI ☑ Hall Runtime ATI □ Comms in ATI ☑ Ultra Low Power Enable

FOSC Selection
● 18 MHz
○ 14 MHz

Power Modes: Auto Mode Switch

HA Report Rate 5 — 5 ms

NP Report Rate 40 — 40 ms

LP Report Rate 200 — 200 ms

ULP Report Rate 500 — 500 ms

ULP Timeout 30000 — 30000 ms

Events Mask

☑ Power Mode Event Mask □ ATI Event Mask ☑ Touch Event Mask ☑ Prox Event Mask

☑ Hall Interval Event Mask

Watchdog Timer Timeout 1000 — 1000 ms

No Changes To Write

Figure 3.7: User Settings Window

Changes to any setting only take effect after the “Write Changes” button has been clicked. The GUI warns of any unwritten changes with a “Changes Made” indicator.

Some settings, such as ATI parameters, may change automatically during the device’s operation. To read and update all the settings from the IQS7221E, click the “Read Settings” button. This may be necessary in certain cases, such as after an ATI event or zero command, or just before exporting settings to a *.h*-file.

> Many of these settings are explained in later sections of this document. For more information, please refer to the IQS7221E datasheet.

3.4 Exporting Settings

Once the desired settings on the IC have been configured, they can be exported and stored as a C header (*.h*) file by clicking the “Export H File” button on the Configuration Tool Manager panel. Each byte of the IQS7221E’s I^{2}C memory map is stored as a `#define` macro. A short snippet of a settings file is shown below.

iqs7221e_init.h

```
#ifndef IQS7221E_INIT_H
#define IQS7221E_INIT_H

/* Change the Global Device Settings */
/* Memory Map Position 0x60 - 0x6F */
#define UI_SETTINGS                              0x33 // 0x60, least-significant byte
#define UI_SETTINGS_1                            0x69 // 0x60, most-significant byte
#define I2C_SETTINGS                             0x00 // 0x61, least-significant byte
#define OPEN_60_3                                0x00 // 0x61, most-significant byte
#define HA_REPORT_0                              0x05 // 0x62, least-significant byte
#define HA_REPORT_1                              0x00 // 0x62, most-significant byte
...
```

Similarly, the “Import H File” button loads the settings stored in an existing header file, and writes them to the IC.

> See the IQS7221E Arduino example code on how to use the exported header file in an embedded project.

3.5 Command Buttons

At the top centre of the GUI is a row of buttons that execute commonly-used commands.

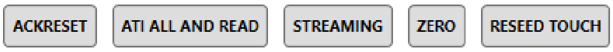


Figure 3.8: Command Buttons

3.5.1 Acknowledge Reset

The “Ack Reset” button clears the IQS7221E’s reset flag by writing the **Acknowledge Reset** bit to the IC. This should be the fist step after powering on any Azoteq IQS-device. On start-up, the IC will set its reset flag to indicate that a reset event has occurred. The GUI will show that a reset has occurred by changing the Ack Reset button colour to red.

> After acknowledging a reset, it may sometimes appear that the device stops streaming. Since the IQS7221E is an I^{2}C-configurable chip, it always resets to I^{2}C streaming mode at start-up, and will remain in streaming mode until the reset flag has been cleared. However, the IQS7221E transitions into event mode as soon as the reset is acknowledged.

3.5.2 Enable Streaming

The IQS7221E has three communication modes:

- **I^{2}C Streaming Mode**: The IQS7221E regularly opens an I^{2}C communications window, directly after it has performed a measurement.
- **I^{2}C Event Mode**: The IQS7221E only opens an I^{2}C window if an event has occurred.
- **Standalone Mode**: The IQS7221E does not open any communications windows, and the master can only use the quadrature outputs to read rotation measurements. (A Force Communication command can still be used to open a communication window.)

The “Streaming” button sets the IC into I^{2}C streaming mode. This streams all the sensor information from the IC to the GUI, which is useful during design and debugging.

> ℹ Event mode can be entered by *setting* the Event Mode bit in the Systems Settings register. Standalone mode can be entered by *clearing* the Streaming Enable bit in the Systems Settings register. The Streaming button in the GUI clears the Event Mode bit and sets the Streaming Enable bit.

3.5.3 ATI All and Read

The "ATI All and Read" button writes the **Force ATI** command to the IQS7221E. The ATI routine is a calibration algorithm on the IC that will recalibrate all the sensors to their target or reference counts.

Once ATI is complete, the GUI reads all the IQS7221E settings to update any parameters that the ATI routine may have changed.

3.5.4 Zero

The "Zero" button resets the current interval to 0. This is done by adjusting the offset between the raw angle and the final angle so that the final angle sits at the centre of interval 0.

The Hall encoder can be zeroed on-chip by setting the **Zero Now** bit in the Hall UI Settings register.

3.5.5 Reseed Touch

The "Reseed" command can be used to update the LTA of the touch channel by setting it equal to the counts. The **Reseed Touch** command is found in the Button Reseed Settings register. Note that this may trigger an ATI routine if the resulting LTA is significantly different from the target.

3.6 Events

The panel on the right-hand side of the GUI shows the current event flags that are set on the IC. On the IQS7221E, these flags can be found at registers 0x10 – 0x14.

3.6.1 System Flags

- **ATI Active**: Set while an ATI routine is busy executing.
- **Show Reset**: Indicates that a recent power-on event has occurred, and should be cleared by sending an Acknowledge Reset command (Ack Reset button). This flag also indicates that all registers and settings of the device have been restored to their default states, and must be reconfigured.

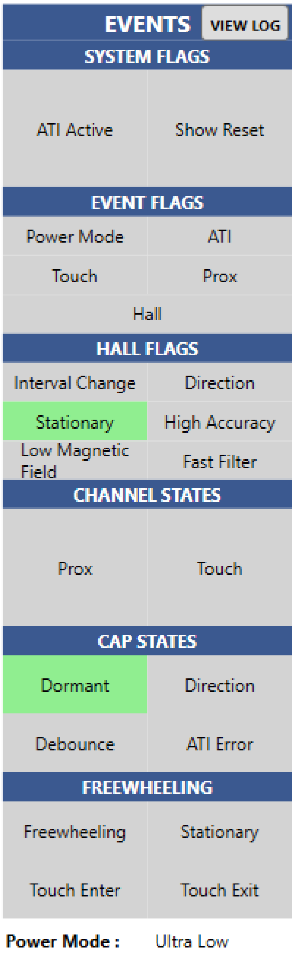
3.6.2 Event Flags

- **Power Mode**: The power mode of the device has changed.
- **ATI**: The touch channel or Hall channels have undergone an ATI routine. This would imply that the counts of the channels have changed, and some on-chip settings may have been altered.
- **Touch**: The touch channel has entered or exited a Touch state.
- **Prox**: The touch channel has entered or exited a Proximity state.
- **Hall**: The Hall encoder has entered a new interval.

3.6.3 Hall Flags

These flags relate specifically to the Hall angle measurements.

- **Interval Change**: The Hall encoder has entered a new interval.
- **Direction**: Set to ‘1’ for a forward rotation (interval value increased), and ‘0’ for a reverse rotation (interval value decreased).
- **Stationary**: Is set if the current interval has not changed for some period of time. The IQS7221E will only transition into low power mode if this flag is set.
- **High Accuracy**: Indicates that the device is in high-accuracy power mode (a temporary high-report-rate mode that samples the Hall channels quickly to avoid potential rotation aliasing).
- **Low Magnetic Field**: An indication that the magnetic fields are too weak to provide an accurate angle measurement, and the current angle output is unreliable. (This flag is only present on the *IQS7221E101* version.)
- **Fast Filter**: Is set when the Hall UI filters the output angle with a higher-bandwidth filter during rapid movements to avoid potential aliasing.



Power Mode : Ultra Low

Figure 3.9: Events Panel

3.6.4 Channel and Cap States

These flags are specific to the touch channel.

- **Prox**: The delta of the touch channel has exceeded the proximity threshold. This flag is usually a simple indication to wake the IQS7221E (and possibly other devices) from low power modes.
- **Touch**: The delta of the touch channel has exceeded the touch threshold. This flag indicates the user has touched or interacted with the touch sensor.
- **Dormant**: The touch sensor has not been activated for some period of time. The IQS7221E will only enter low power mode if this flag (along with the Hall Stationary flag) is set.
- **Direction**: This indicates the sign (positive or negative) of the delta. This is useful if the channel is set to use dual-direction thresholds.
- **Debounce**: When crossing the prox threshold, the touch sensor will be sampled multiple times in rapid succession. This flag is set during this rapid sensing stage.
- **ATI Error**: Set if the touch channel fails to recalibrate correctly. This indicates that the current output of the touch channel may be invalid. (This flag is only present on the *IQS7221E101* version.)

3.6.5 Freewheeling

These flags are only used when the virtual freewheeling function is enabled.

- **Freewheeling**: The IQS7221E is currently performing virtual freewheeling. Changes in the final angle and interval are generated in software, and are not due to physical rotation of the magnet.
- **Stationary**: Distinct from the Hall Stationary flag, this indicates that since virtual freewheeling started, the magnet became stationary, even if for a brief moment.

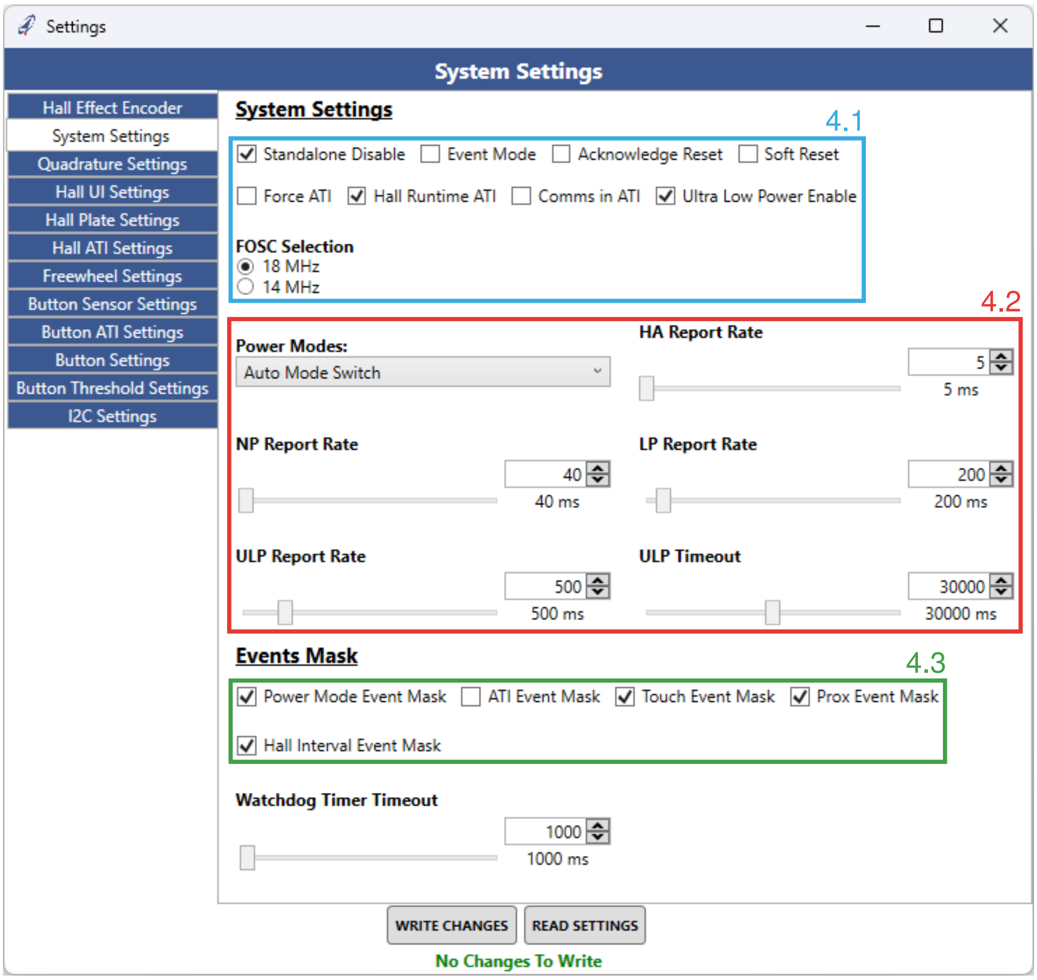
- **Touch Enter**: Touch event flag used when the Release UI is enabled. (Release UI is disabled by default.) The Touch Enter event flag is set briefly when entering the touch state with a positive delta (Direction flag is set). This occurs when a user touches the sensor.
- **Touch Exit**: Release event flag used when the Release UI is enabled. The Touch Exit event flag is set briefly when entering the touch state with a negative delta (Direction flag is cleared). This occurs when a user moves away from the sensor.

3.6.6 Power Mode

Finally, the current **Power Mode** is displayed at the bottom of the events panel.

4 System Settings

This section quickly explains some of the basic system settings and commands that are not specific to any sensor or UI. These settings can be accessed in the “User Settings” window, on the first tab.



System Settings

☑ Standalone Disable □ Event Mode □ Acknowledge Reset □ Soft Reset

□ Force ATI ☑ Hall Runtime ATI □ Comms in ATI ☑ Ultra Low Power Enable

FOSC Selection
● 18 MHz
○ 14 MHz

Power Modes: Auto Mode Switch

HA Report Rate 5 — 5 ms

NP Report Rate 40 — 40 ms

LP Report Rate 200 — 200 ms

ULP Report Rate 500 — 500 ms

ULP Timeout 30000 — 30000 ms

Events Mask

☑ Power Mode Event Mask □ ATI Event Mask ☑ Touch Event Mask ☑ Prox Event Mask

☑ Hall Interval Event Mask

Watchdog Timer Timeout 1000 — 1000 ms

No Changes To Write

Figure 4.1: User Settings Tab

4.1 System Settings Register

The first set of check boxes are settings that are largely controlled by the buttons at the top of the GUI, and have been covered in Section 3.5. However, the following settings are also worth noting:

- **Runtime ATI**: This allows the device to recalibrate the Hall encoder automatically during normal operation, if necessary. It is generally recommended to leave this option enabled.
- **Comms in ATI**: The IQS7221E will still open regular communication windows during ATI. It is recommended to leave this disabled, as it will slow the ATI recalibration stage.
- **POR**: Performs a soft (warm) reset of the IC.
- **Ultra Low Power (ULP) Enable**: This allows the IQS7221E to enter ULP mode during automatic power-mode switching. In ULP mode only the touch channel is sampled. The IC will not perform any sensing on the Hall plates, and thus will not respond to rotation. This setting is aimed at low power applications where the IC does not need to sample the Hall channels until it detects user input on the capacitive-sensing channel. For applications that only use the Hall-rotation sensing, without any touch sensing, this option *must* be disabled.

- **F_{OSC} Selection**: Specifies whether the IQS7221E uses 14 MHz or 18 MHz as its main system clock. This setting has implications on the power supply requirements. Please consult the Electrical Characteristics section of the Datasheet. (This setting is only applicable to *IQS7221E101*.)

4.2 Power Modes

The IQS7221E supports a number of different power modes. The current power mode of the IQS7221E can be set via the Power Modes dropdown menu in Figure 4.1.

- **High Accuracy (HA)**: Fastest sampling rate, used during rotation to prevent aliasing.
- **Normal Power (NP)**: Fast report rate used during touch events and directly after a high accuracy event.
- **Low Power (LP)**: Slower report rate for power saving. This is the lowest power mode that still samples Hall channels.
- **Ultra Low Power (ULP)**: Only the touch channel is sampled.
- **Halt**: Sets the IC into the lowest power state where it does not perform any sensing. Note that only an I^{2}C force communications request (described in the datasheet) may wake the device from halt mode.
- **Automatic Power Modes**: Automatically switches between power modes based on touch events and Hall UI events. In this mode, the device will only enter ULP mode if the ULP Enable bit is set. Automatic power mode will never set the device into halt mode.

The sample rate of each power mode can be configured by changing the associated report rate. The report rate is the time between consecutive cycles, in milliseconds, where each cycle performs measurements on all channels. Higher power modes sample faster at the cost of higher current consumption.

> ⚠ The high accuracy report period must always be shorter than the normal power period. The normal power period must be shorter than for low power mode.

4.3 Event Mask

The event mask is used to enable or disable specific events, and is particularly useful in event mode, as disabled events will not open new communication windows. As an example, it is often useful to ignore ATI events. By clearing the ATI Event Mask bit, the IQS7221E will not open a communication window to report ATI events.

This only affects the behaviour of I^{2}C Event Mode — the associated event flags will still be set.

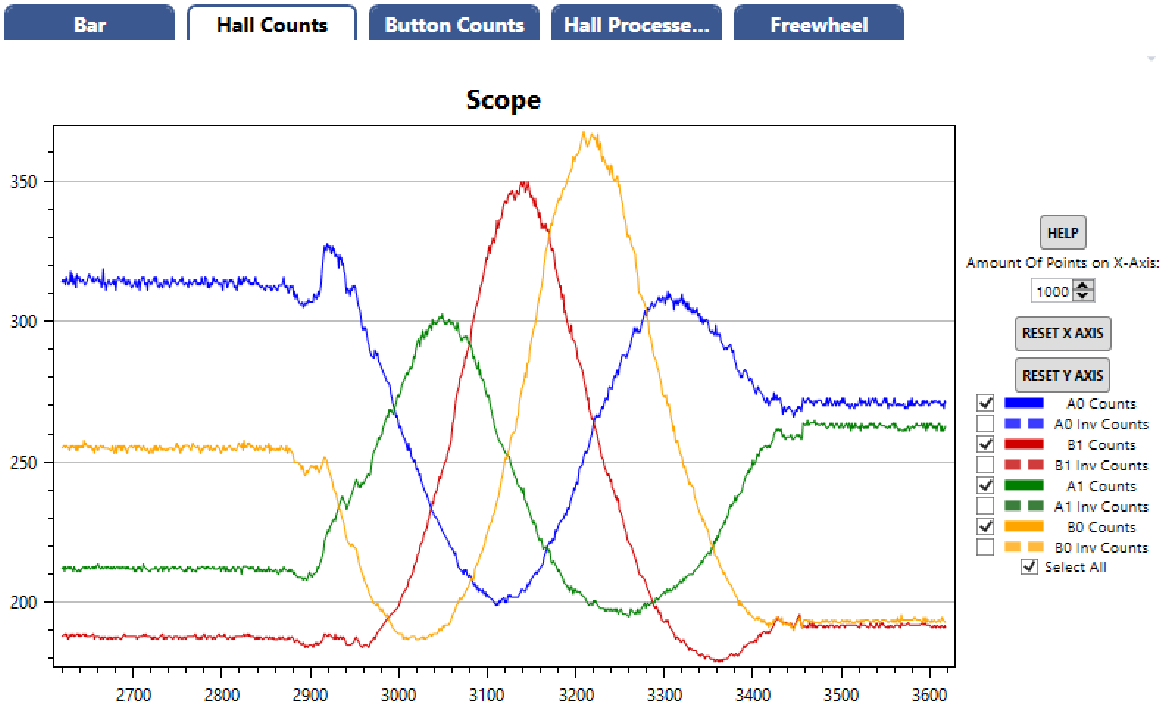
5 Hall Configuration

This section describes the steps to configure the Hall UI settings for the IQS7221E EV kit. It is recommended to force the device into normal power mode when experimenting with sensor settings, as it provides a consistent sample rate for evaluating the sensor performance. For this section, the device is set to Normal Power mode, with a 10 ms report period. See Section 4.2 regarding setting the current power mode.

5.1 IQS7221E Hall-Effect Sensing Overview

The IQS7221E has four integrated Hall plates that measure the strength of the vertical component of an external magnetic field. The magnetic field induces a Hall voltage on the Hall plates that may be positive or negative depending on the polarity of the magnetic field.

The IQS7221E measures the voltage on each plate and reports a counts value. However, the internal measurement circuit only measures positive voltages. The IQS7221E therefore applies a “Hall Plate Offset” to the Hall plate measurement to ensure that the measured voltages are always in the positive range. This offset determines the level of the Hall reference counts, around which the Hall counts and Hall inverse counts swing.



Amount Of Points on X-Axis: 1000

☑ A0 Counts
☐ A0 Inv Counts
☑ B1 Counts
☐ B1 Inv Counts
☑ A1 Counts
☐ A1 Inv Counts
☑ B0 Counts
☐ B0 Inv Counts
☑ Select All

Figure 5.1: Hall Counts Over a Rotation

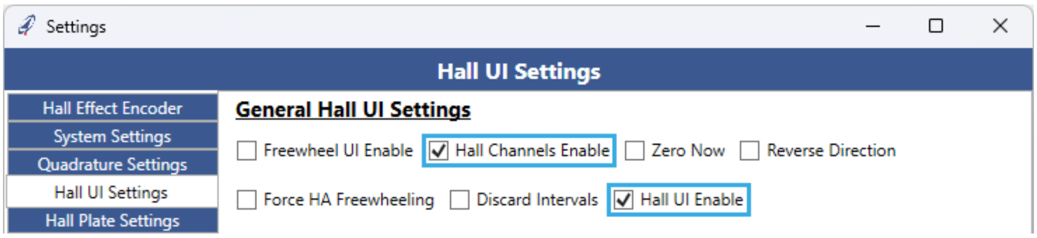
Figure 5.1 shows the counts of the Hall channels in the scope view, over a full rotation. Note the channels do not have equal swing magnitudes, likely due to the mechanical tolerances of the EV kit assembly, or process variations between the Hall plates. However, the use of the four Hall plates for the angle calculations mitigates the effect of these differences in counts swing.

The IQS7221E uses the strength and orientation of the magnetic fields at the four plates to calculate the magnet’s angle relative to the IC, and then applies an angle offset and filtering to generate the final angle.

For more information regarding the Hall-rotation sensing principles, please refer to the AZD127 Application Note.

5.2 Enable the Hall UI

The first step in configuring the Hall-rotation sensing is to confirm that the Hall UI and Hall channels are enabled. These options can be found in the “Hall UI Settings” tab, and are already enabled by default at start-up, as shown in Figure 5.2.



General Hall UI Settings

□ Freewheel UI Enable ☑ Hall Channels Enable □ Zero Now □ Reverse Direction

□ Force HA Freewheeling □ Discard Intervals ☑ Hall UI Enable

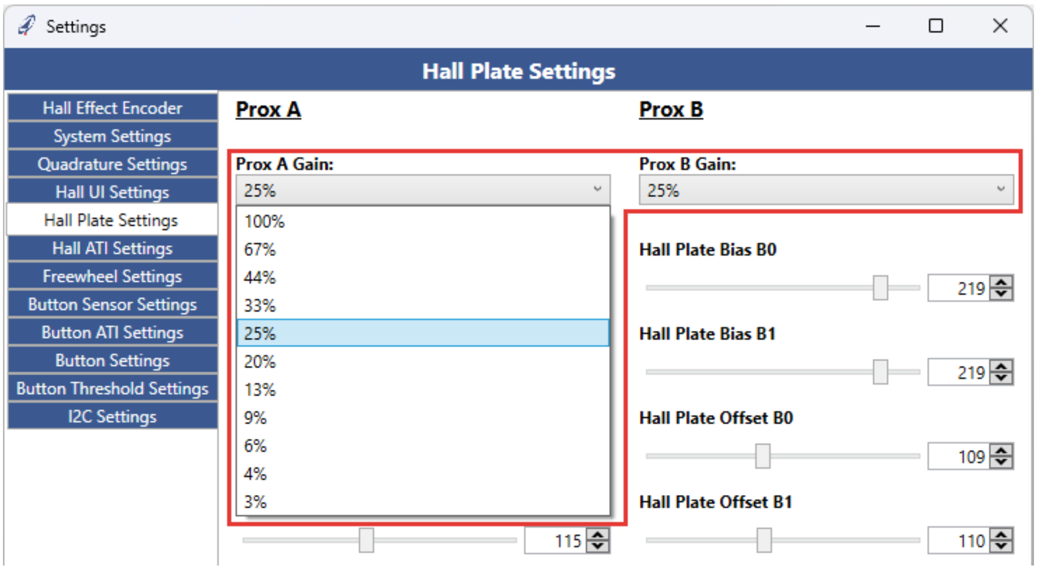
Figure 5.2: Enable Hall Rotation UI

5.3 Adjusting Hall Sensitivity

The next step is to ensure that the magnet can induce enough swing in the counts to provide adequate signal for the angle calculation. Without sufficient signal strength, the angle output may be prone to jitter. A swing of at least 100 counts peak-to-peak is recommended for an application with 24 intervals per rotation, with 200 counts swing being ideal.

The amount of swing is dependent on the magnitude of the magnetic field. Using a stronger magnet, or bringing the magnet closer to the IC, will increase the counts swing. However, the gain of the Hall sensors can also be adjusted to increase the counts swing, specifically by changing the **Prox A Gain** and **Prox B Gain** values. Increasing the gain values will increase the sensitivity of the Hall sensors, and results in larger counts swing.

The gain configuration is found under “Hall Plate Settings”, shown below. These gain options provide optimised selections for the Hall plate Fine and Coarse Multipliers. Prox A and Prox B gain should be set to the same value.



Prox A

Prox A Gain: 25%

115

Prox B

Prox B Gain: 25%

Hall Plate Bias B0: 219

Hall Plate Bias B1: 219

Hall Plate Offset B0: 109

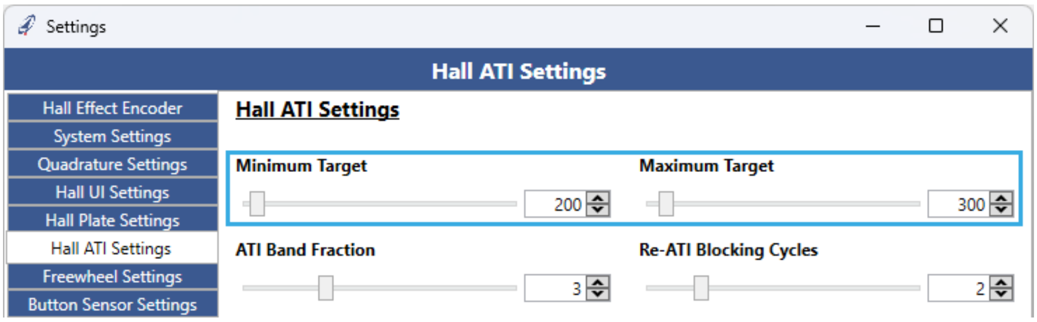
Hall Plate Offset B1: 110

Figure 5.3: Hall Multipliers and Dividers

Note that changing the gain of the Hall sensors will also affect the counts reference. If the gain is increased, the Hall offset must be decreased to maintain the same counts reference. The Hall Automatic Tuning implementation (ATI) is able to re-adjust the offsets to maintain the correct reference. Any time the gain values (Hall multipliers) are changed, click the "ATI All and Read" to update the Hall offsets.

The standard procedure for tuning the Hall channels' sensitivity is as follows:

1. Set the reference counts. This is done by setting the **Minimum Target** and **Maximum Target** values, shown in Figure 5.4. The required reference counts is automatically calculated.



Minimum Target 200
Maximum Target 300
ATI Band Fraction 3
Re-ATI Blocking Cycles 2

Figure 5.4: Hall Targets

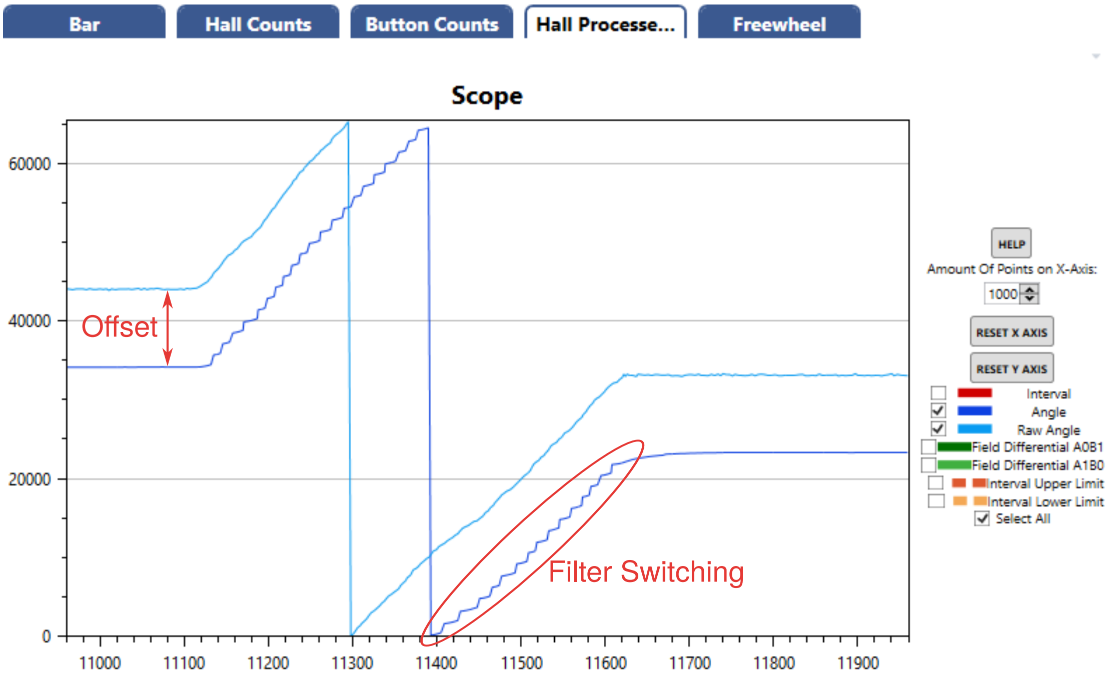
2. Perform an ATI routine. In the GUI, click the "ATI All and Read" button. This will calculate applicable Hall offsets for each Hall channel.
3. Rotate the magnet above the IC, and take note of the minimum and maximum counts values of the Hall channels.
4. Adjust the Gain value. If the observed swing is too low, increase the gain value. If the observed swing is too high, lower the gain value. Repeat the process from step 2.

If the Hall counts cannot achieve the desired swing, even with maximum gain, this is an indication that the Hall plates are not receiving a strong enough magnetic field. Please refer to the AZD127 Application Note for more details on magnet strength and positioning.

In the case of the IQS7221E EV kit, the minimum and maximum targets can be set to 200 and 400, respectively. The resulting Hall counts should swing approximately between 200 and 400 counts, without requiring any changes to the gain.

5.4 Evaluating the Angle Output

To view the angle output over time, switch to the "Hall Processed Data" scope view, and enable only the **Angle** and **Raw Angle** lines. Rotate the knob, and confirm that the angle changes as expected. A clockwise rotation should result in an increase in angle.



Amount Of Points on X-Axis: 1000

□ 🟥 Interval
☑ 🟦 Angle
☑ 🟦 Raw Angle
□ 🟩 Field Differential A0B1
□ 🟩 Field Differential A1B0
□ 🟧 Interval Upper Limit
□ 🟧 Interval Lower Limit
☑ Select All

Figure 5.5: Scope View of Hall Angle

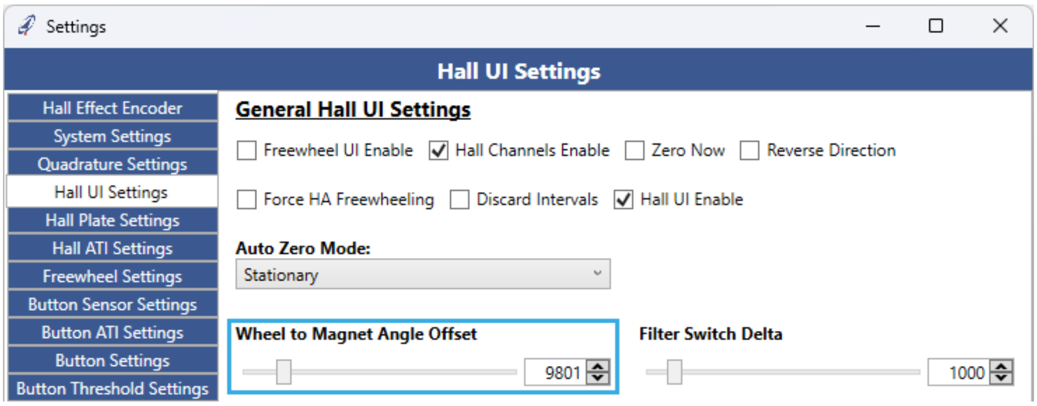
Figure 5.5 shows the scope view of the raw and final angle values over a full rotation. These angles are 16-bit values, where a full rotation from 0° to 360° maps to 0 – 65,536. The raw angle is unfiltered, and represents the absolute angle of the magnet relative to the IC. The final angle value is filtered, and has an offset from the raw angle. The offset in this figure is approximately 10,000.

Note the “stepping” behaviour in the final angle. This is an artefact of the filter switching, explained in Section 5.4.2.

While initially testing the Hall-rotation sensing, it may be useful to set the Auto Zero When Stationary setting to “Off”. This setting is configured in Section 5.5.3.

5.4.1 Wheel-to-Magnet Angle Offset

The offset between the raw angle and final angle is set when the Zero command is given to the IQS7221E, but may also be changed by the Auto Zero function, or by virtual freewheeling. The offset can be read from the **Wheel-to-Magnet Angle Offset** register. In this example, the offset was set to 9801, which agrees with the observation from the scope view in Figure 5.5. This register can be overwritten, but altering this register during runtime is discouraged.



General Hall UI Settings

□ Freewheel UI Enable ☑ Hall Channels Enable □ Zero Now □ Reverse Direction

□ Force HA Freewheeling □ Discard Intervals ☑ Hall UI Enable

Auto Zero Mode: Stationary

Wheel to Magnet Angle Offset: 9801

Filter Switch Delta: 1000

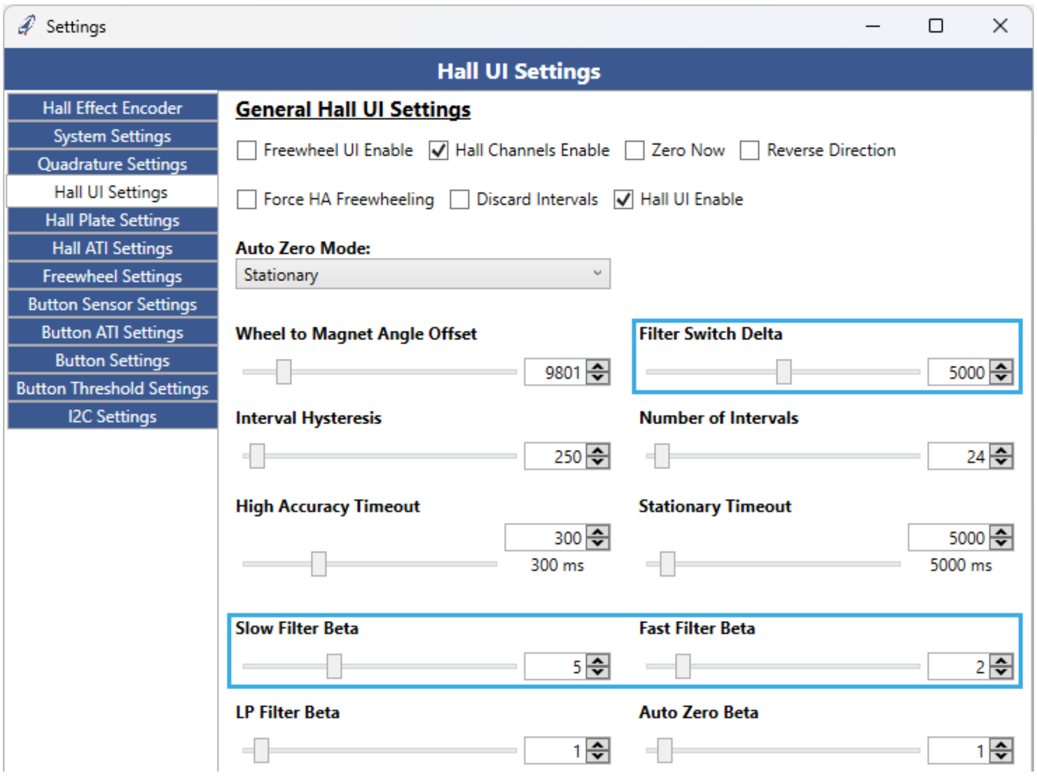
Figure 5.6: Angle Offset Setting

5.4.2 Angle Filtering

As mentioned, the output angle is filtered to reduce jitter. However, two separate filters are used to filter the angle; a “slow” filter and a “fast” filter, which are controlled by the **Slow Filter Beta** and **Fast Filter Beta**, respectively.

The slow filter is used when there is very little or no movement on the magnet. This filter can have a large beta value, providing a slow but smooth response that is ideal for reducing jitter and avoiding unwanted interval changes when the magnet is stationary. However, this slow filter cannot track quick rotations of the magnet.

The fast filter uses a much smaller beta value, allowing it to track rapid rotations. During quick rotations, the IQS7221E switches to the fast filter beta for the angle filtering.



General Hall UI Settings

□ Freewheel UI Enable ☑ Hall Channels Enable □ Zero Now □ Reverse Direction

□ Force HA Freewheeling □ Discard Intervals ☑ Hall UI Enable

Auto Zero Mode: Stationary

Wheel to Magnet Angle Offset: 9801

Filter Switch Delta: 5000

Interval Hysteresis: 250

Number of Intervals: 24

High Accuracy Timeout: 300 (300 ms)

Stationary Timeout: 5000 (5000 ms)

Slow Filter Beta: 5

Fast Filter Beta: 2

LP Filter Beta: 1

Auto Zero Beta: 1

Figure 5.7: Angle Filter Settings

The **Filter Switch Delta** represents the maximum allowed difference between the outputs of the slow and fast filters. If the difference between the filter outputs exceed this threshold, the fast filter beta is used. The switching between filter betas can cause a perceived "stepping" effect, as seen in Figure 5.5. Note that this does not adversely affect the Interval UI. However, if desired, this effect can be reduced by increasing the filter switch delta.

As an example, the value can be increased to 5000, effectively increasing the minimum rotation speed at which the fast filter is activated. This is shown in Figure 5.8. The filter switch value must be adjusted for different applications based on the expected rotation speeds in the application.

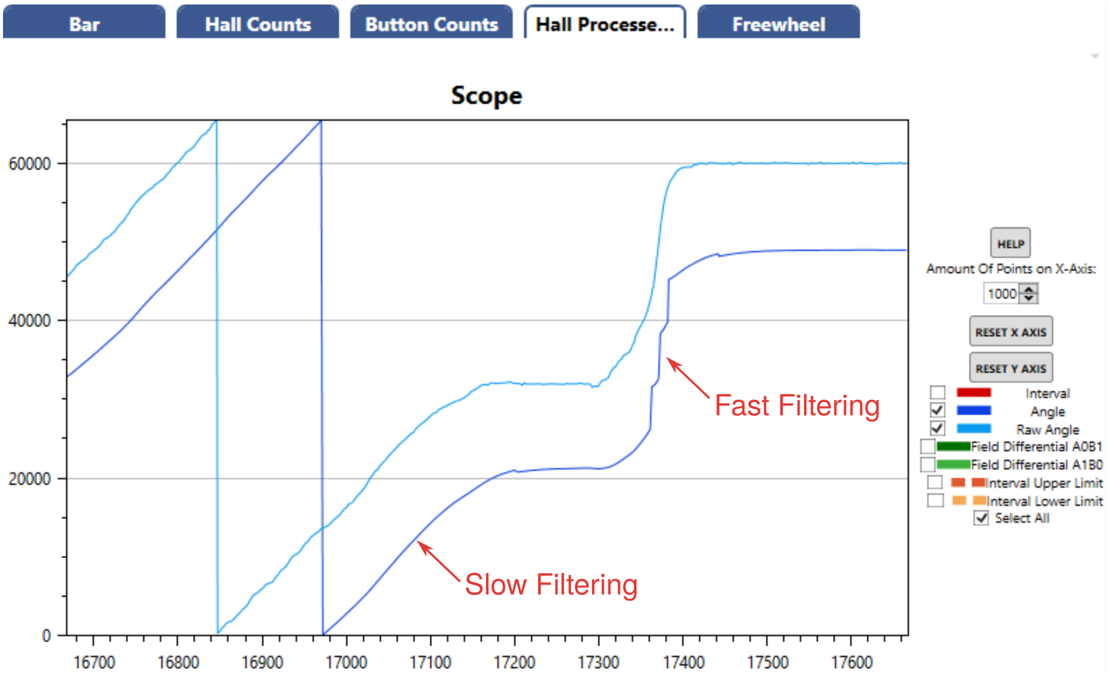


Figure 5.8: Angle Filtering With Larger Filter Switch Threshold

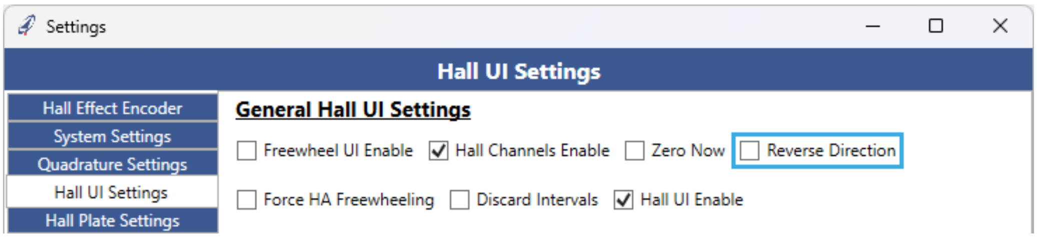
For applications with mechanical ratchets, it is recommended to set the filter switch delta to half the size of an interval, or smaller. This ensures that there is minimal delay between the mechanical click and the IQS7221E reporting the new interval.

Filter switching can also be disabled by setting the slow and fast beta values equal.

⚠ Angle filtering may introduce aliasing during rapid movements. If high rotation rates of the magnet are expected, the fast beta value can be set to 0 to disable filtering completely.

5.4.3 Rotation Direction

Finally, the direction of the output rotation can be changed by setting the **Reverse Direction** bit. An anti-clockwise rotation of the magnet will now result in a positive angle change. Note that this changes the direction of all output angles, from the raw angle to the interval output, as well as the quadrature outputs.



Settings

Hall UI Settings

General Hall UI Settings

□ Freewheel UI Enable ☑ Hall Channels Enable □ Zero Now □ Reverse Direction

□ Force HA Freewheeling □ Discard Intervals ☑ Hall UI Enable

Figure 5.9: Reverse Hall Rotation Direction

By default, the output angle of the IQS7221E EV kit increases for a clockwise rotation, so the reverse direction can be left disabled.

5.5 Configuring the Interval UI

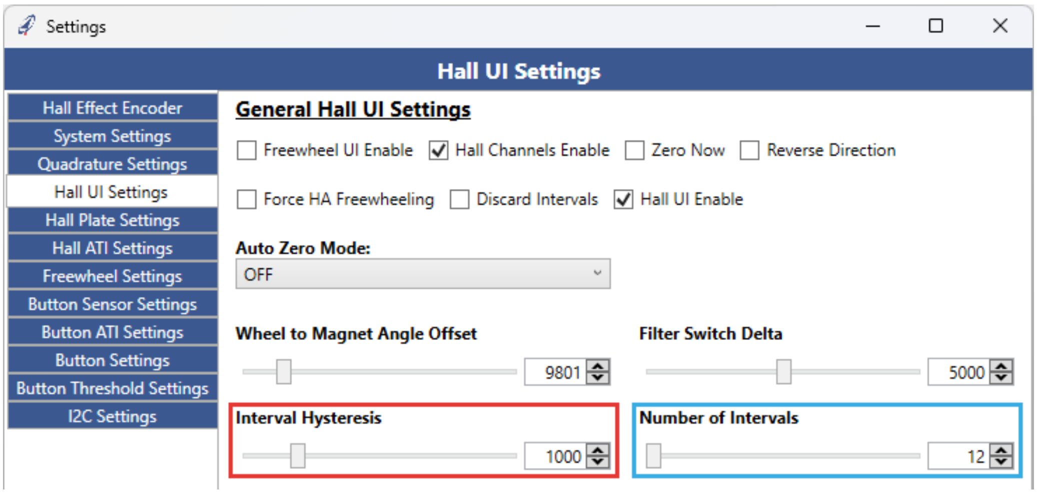
5.5.1 Interval Size

The IQS7221E divides the 16-bit final angle into discrete intervals, based on the **Number of Intervals** setting. For example, a control knob may need 12 discrete positions (or intervals) per rotation. A mouse scroll wheel may use a mechanical ratchet with 24 positions.

> ⚠ Whenever changing the number of intervals, it is necessary to perform a Zero Now command afterwards.

As an example, set the number of intervals for the EV kit to 12, as shown in Figure 5.10, then click the Zero button. Since the Interval UI divides the 16-bit final angle based on the number of intervals, the size of each interval can be calculated as:

$$\frac{2^{16}}{12} \approx 5461. \tag{1}$$



General Hall UI Settings

□ Freewheel UI Enable ☑ Hall Channels Enable □ Zero Now □ Reverse Direction

□ Force HA Freewheeling □ Discard Intervals ☑ Hall UI Enable

Auto Zero Mode: OFF

Wheel to Magnet Angle Offset: 9801

Filter Switch Delta: 5000

Interval Hysteresis: 1000

Number of Intervals: 12

Figure 5.10: Changing Number of Intervals

The interval size can be visualised in the scope view by enabling the **Interval Upper Limit** and **Interval Lower Limit** plots. These limits are the boundaries of the current interval, and if the (final) angle value crosses one of the limits, the current interval will change.

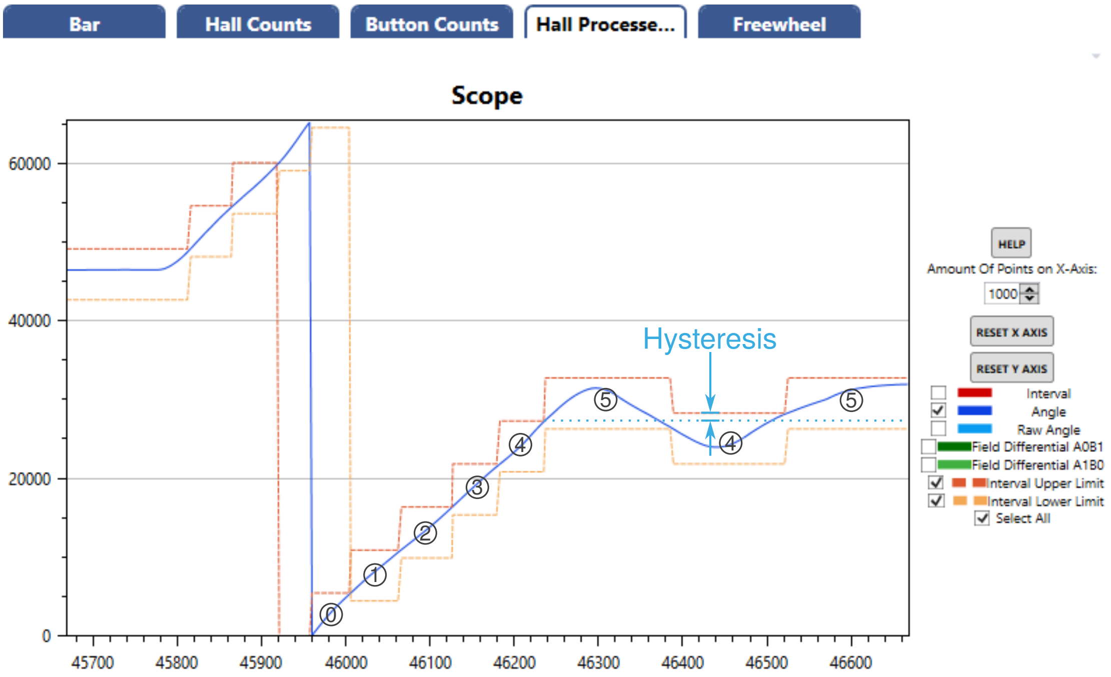


Figure 5.11: Scope View of Hall Angle

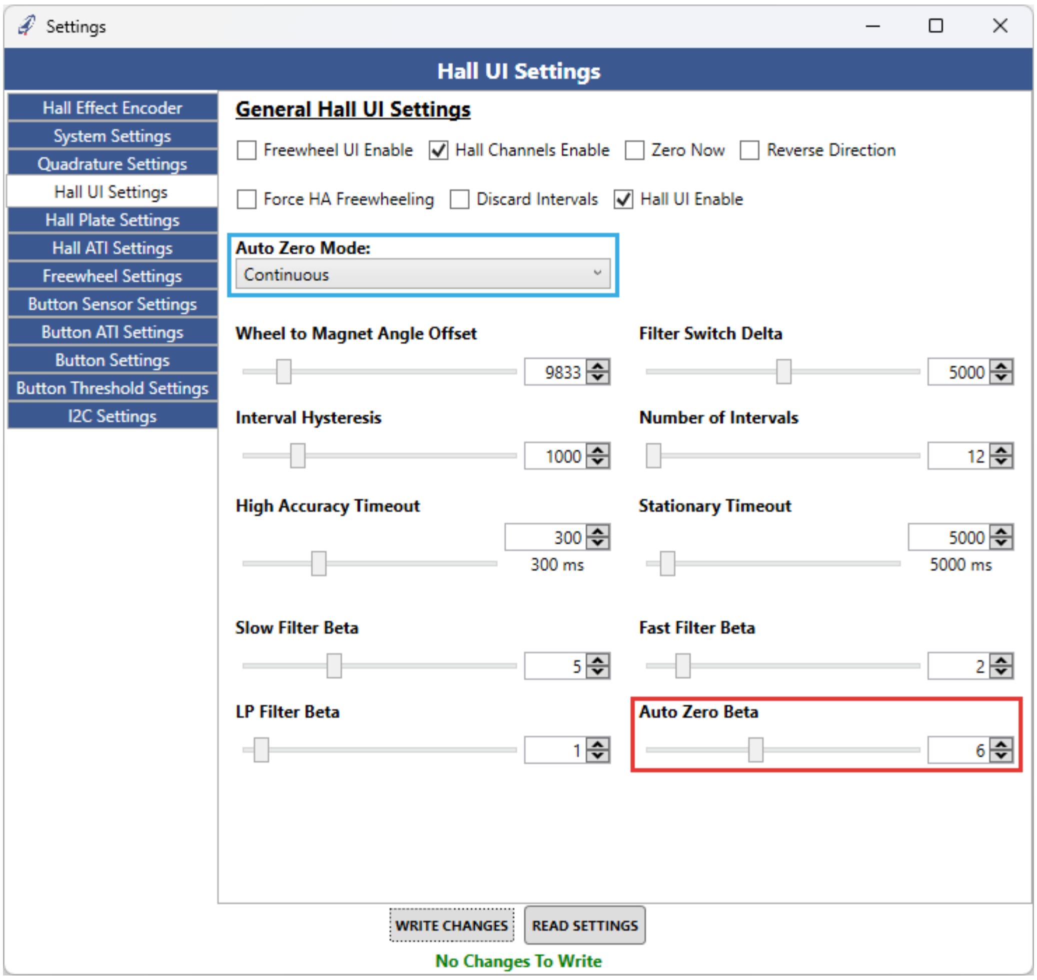
5.5.2 Interval Hysteresis

Interval Hysteresis provides a debouncing effect that prevents the interval output from jittering between two intervals when the final angle is near an interval boundary. The effect of hysteresis can be seen in Figure 5.11, where the angle switches from interval 5 to interval 4. As the angle crosses the lower limit of interval 5, the upper limits and lower limits shift down to indicate the edges of interval 4, but the new upper limit for interval 4 is slightly higher than before. The angle can now drift back upwards past the original threshold without triggering a new interval. This means the user has to rotate the magnet deliberately to move back to interval 5.

The Interval Hysteresis setting can be found under “Hall UI Settings”, as shown in Figure 5.10. In this example the hysteresis was set to 1000. Note that in practice this amount of hysteresis may be excessive, but is acceptable in this example that uses few intervals. The hysteresis value should always be significantly smaller than the size of the interval, which was calculated above as 5461.

5.5.3 Automatic Interval Centering

The final setting related to the Interval UI is the **Auto Zero** function. This feature attempts to align the centre of the IQS7221E’s intervals with the mechanical intervals in a ratchet so that the transition between two intervals happens at the correct/expected place. Applications with a smooth rotation (without a ratchet) may also take advantage of this feature, as it will automatically adjust the current angle towards the centre of the current interval, effectively adding additional hysteresis and reducing the possibility of a false interval change. The Auto Zero settings are available on the “Hall UI Settings” tab, shown in Figure 5.12.



Settings

Hall UI Settings

General Hall UI Settings

□ Freewheel UI Enable ☑ Hall Channels Enable □ Zero Now □ Reverse Direction

□ Force HA Freewheeling □ Discard Intervals ☑ Hall UI Enable

Auto Zero Mode: Continuous

Wheel to Magnet Angle Offset: 9833

Filter Switch Delta: 5000

Interval Hysteresis: 1000

Number of Intervals: 12

High Accuracy Timeout: 300 (300 ms)

Stationary Timeout: 5000 (5000 ms)

Slow Filter Beta: 5

Fast Filter Beta: 2

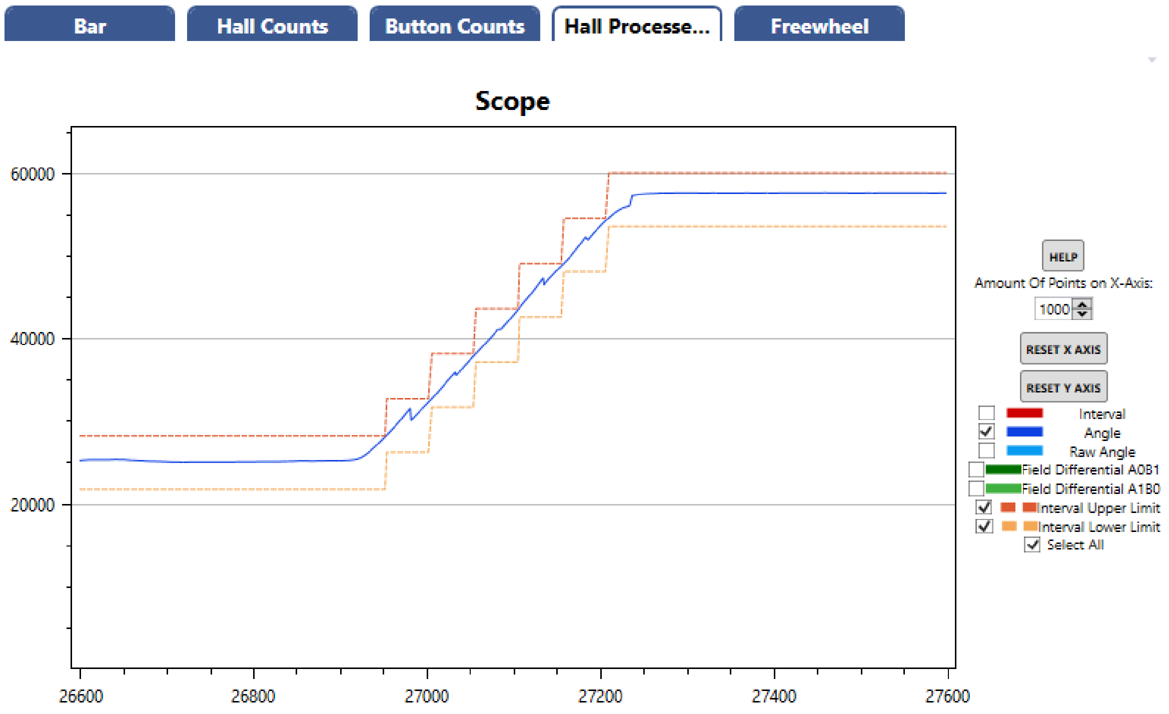
LP Filter Beta: 1

Auto Zero Beta: 6

No Changes To Write

Figure 5.12: Auto Zero Settings

The default mode is the **Stationary Auto Zero** mode. Section 5.5.1 recommended disabling this feature temporarily. The reason for this can be seen in the following scope capture of a forward rotation with Stationary auto zero mode enabled:



Amount Of Points on X-Axis: 1000

☐ 🟥 Interval
☑ 🟦 Angle
☐ 🟦 Raw Angle
☐ 🟩 Field Differential A0B1
☐ 🟩 Field Differential A1B0
☑ 🟥 Interval Upper Limit
☑ 🟧 Interval Lower Limit
☑ Select All

Figure 5.13: Stationary Auto Zero Adjustment

The Stationary auto zero mode applies a discrete step to the angle offset value shortly after a new interval is entered, moving the final angle towards the centre of the new interval. This causes the effect where the angle steps backwards suddenly, as seen in Figure 5.13. This adjustment only happens once, until the interval changes again. The size of the step adjustment is controlled by the **Auto Zero Beta** value.

This mode is intended for ratchet applications that have a discrete number of physical positions, where the auto zero update only happens after the wheel has settled in the centre of the new mechanical interval.

For the EV kit, and other applications with smoothly-rotating wheels, the **Continuous Auto Zero** mode is recommended. This mode continuously filters the offset towards the centre of the current interval, as shown in Figure 5.14. A higher Auto Zero Beta value, at least 6 or greater, is needed for this feature to work correctly.

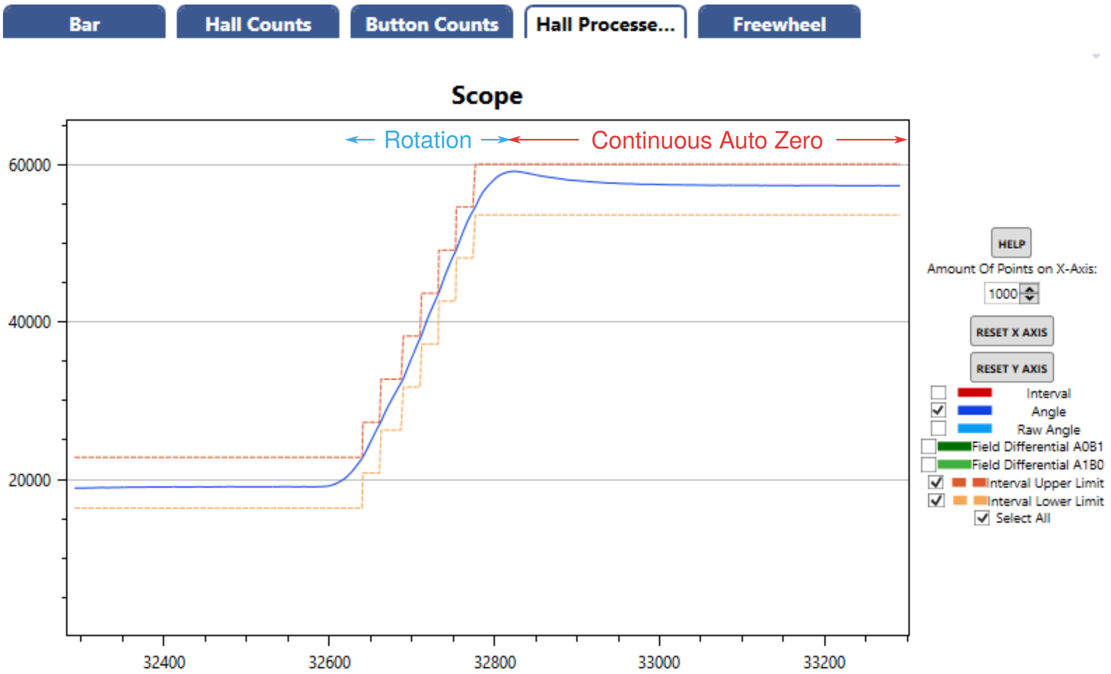
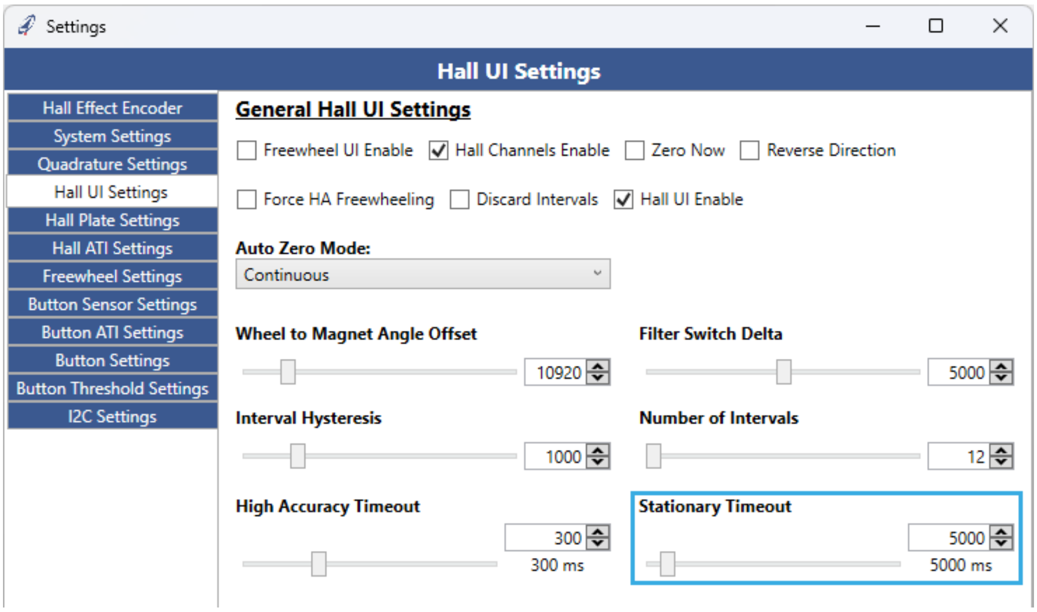


Figure 5.14: Continuous Auto-Zero Adjustment

5.6 Stationary Timeout

The **Stationary Timeout** is the duration that the IQS7221E will wait after an interval change before setting the Stationary flag. The Stationary flag determines when the IQS7221E may transition to low power mode.



General Hall UI Settings

□ Freewheel UI Enable ☑ Hall Channels Enable □ Zero Now □ Reverse Direction

□ Force HA Freewheeling □ Discard Intervals ☑ Hall UI Enable

Auto Zero Mode: Continuous

Wheel to Magnet Angle Offset: 10920

Filter Switch Delta: 5000

Interval Hysteresis: 1000

Number of Intervals: 12

High Accuracy Timeout: 300 (300 ms)

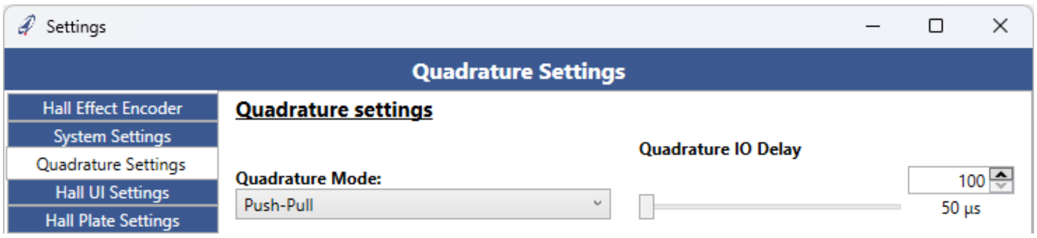
Stationary Timeout: 5000 (5000 ms)

Figure 5.15: Stationary Timeout Setting

5.7 Quadrature Output

The IQS7221E features two quadrature output pins to report interval changes without the need for I^{2}C communication. With the quadrature outputs enabled, the IQS7221E can be placed in Standalone Mode, which aids in reducing current consumption.

On an interval change, the two quadrature outputs will both change state, with a short delay between them. The delay between the two pin changes is configurable via the **Quadrature IO Delay** parameter. It may be necessary to set this value to at least 1 millisecond, depending on how quickly the MCU can react to pin interrupts.



Quadrature settings

Quadrature Mode: Push-Pull

Quadrature IO Delay: 100
50 µs

Figure 5.16: Quadrature Output Settings

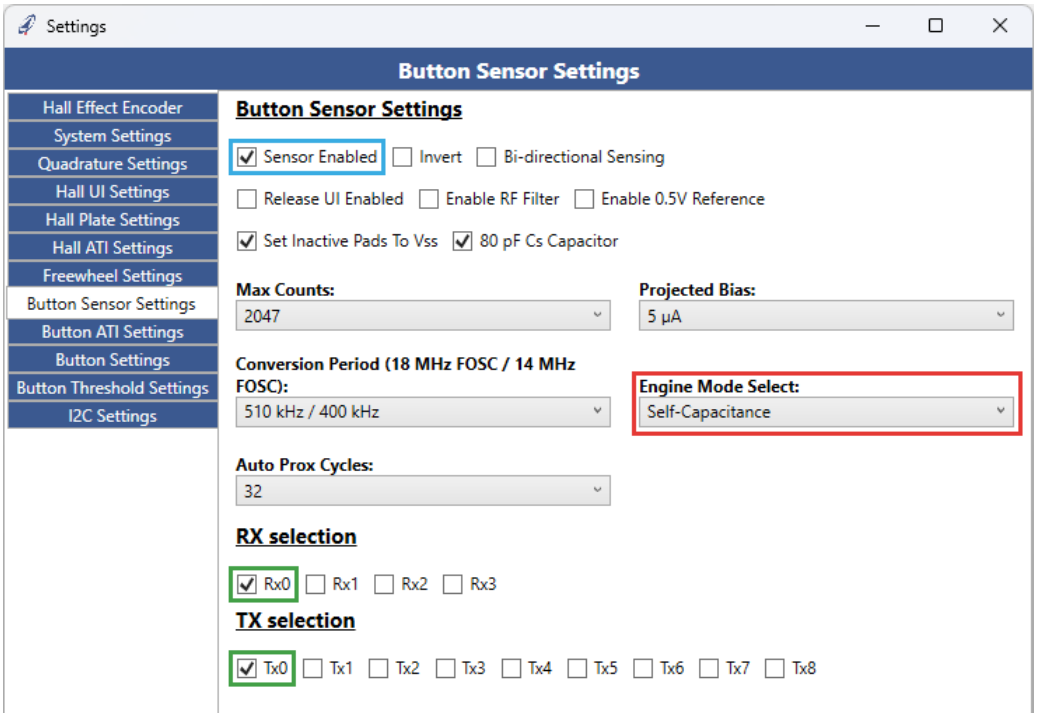
6 Touch Channel Configuration

The IQS7221E supports a single touch sensor that can be used for low power wake-up, and for the Freewheel UI. The sensor can be configured as a self-capacitance sensor (that requires a single RX electrode) or a mutual-capacitance sensor (that requires separate RX and TX electrodes). Self-capacitance is a very simple sensor architecture, and the EV kit thus features an RX electrode connected to the CRx0 pin. The capacitive sensor couples through the magnet and metal shaft to the knob, and reacts to user touch. The counts on the Button channel will decrease when the user touches the sensor. For more information regarding the EV kit's touch sensor, please see Appendix A.

6.1 Initial Touch Setup

Under "Button Sensor Settings", first make sure the touch sensor is enabled by selecting **Sensor Enable**. Then ensure that the correct pins are used for the capacitive sensing. The EV kit has a self-capacitance sensing pad connected to the CRx0 pin, thus Rx0 and Tx0 must be enabled. Finally, ensure that the **Self-capacitance Engine Mode** is selected. See Figure 6.1.

It is recommended to keep all other settings at their defaults. The Release UI will be revisited in Section 7 when configuring the Freewheel UI.



Button Sensor Settings

☑ Sensor Enabled □ Invert □ Bi-directional Sensing

□ Release UI Enabled □ Enable RF Filter □ Enable 0.5V Reference

☑ Set Inactive Pads To Vss ☑ 80 pF Cs Capacitor

Max Counts: 2047

Projected Bias: 5 µA

Conversion Period (18 MHz FOSC / 14 MHz FOSC): 510 kHz / 400 kHz

Engine Mode Select: Self-Capacitance

Auto Prox Cycles: 32

RX selection

☑ Rx0 □ Rx1 □ Rx2 □ Rx3

TX selection

☑ Tx0 □ Tx1 □ Tx2 □ Tx3 □ Tx4 □ Tx5 □ Tx6 □ Tx7 □ Tx8

Figure 6.1: Button Sensor Settings

This document assumes the application (or EV kit) is using a self-capacitance sensor. When using self-capacitance sensing, the Button counts will decrease on a touch. A mutual-capacitance sensor will instead typically see an *increase* in counts. This must be taken into account when using the Release UI, and will require the Invert bit to be set.

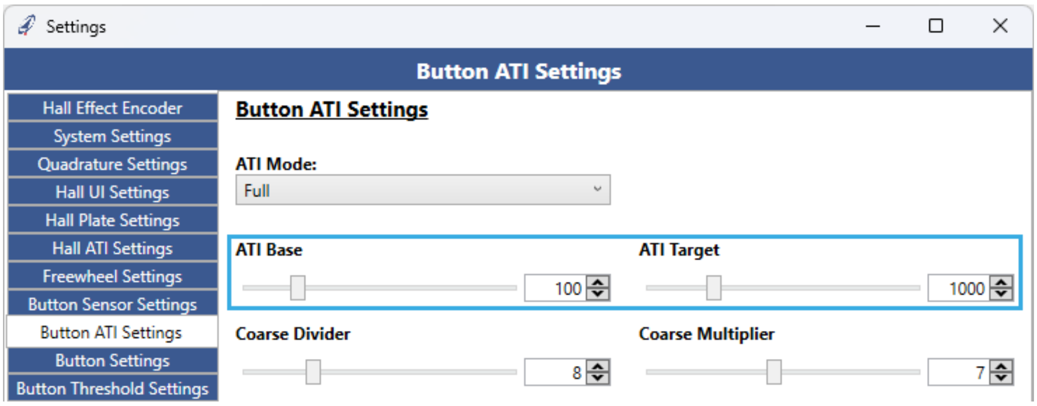
6.2 Adjusting Touch Sensitivity

The sensitivity of the touch channel can be adjusted by altering the **ATI Base** and **ATI Target** values. The ATI parameters' relationship to sensitivity is generally described as

$$\text{Sensitivity} \propto \frac{\text{ATI Target}}{\text{ATI Base}}.$$

Therefore, to increase the sensitivity of the touch sensor, increase the target value and/or decrease the base value, and then manually run an ATI. The ATI routine for the touch sensor will automatically adjust the multipliers, dividers, and compensation to reach the target.

As an example, set the ATI Target to 1000 as shown in Figure 6.2, and then click the "ATI All and Read" button. The counts and LTA of the Button channel on the graph should change to approximately 1000 counts. This increase in target increases the sensitivity of the channel, and a touch will now result in a larger delta than before.



Button ATI Settings

ATI Mode: Full

ATI Base: 100

ATI Target: 1000

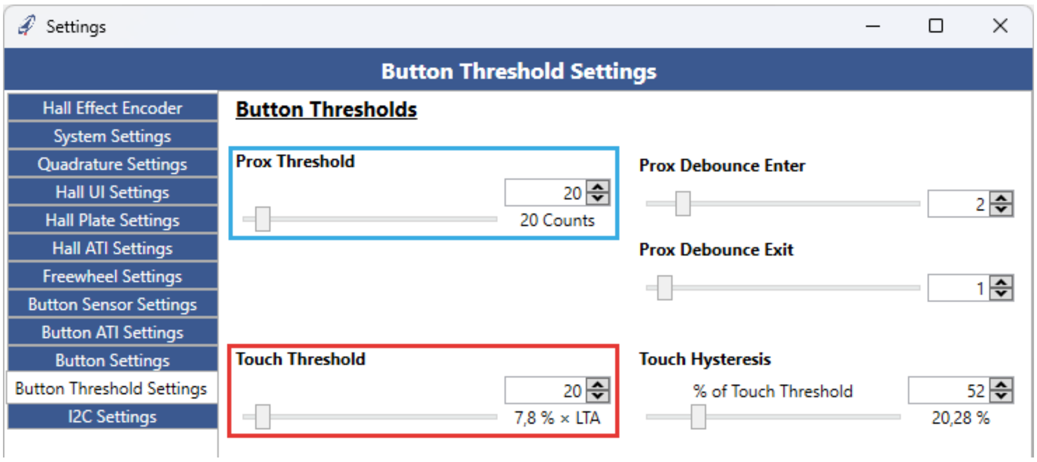
Coarse Divider: 8

Coarse Multiplier: 7

Figure 6.2: Button ATI Settings

6.3 Thresholds

The **Prox** and **Touch Thresholds** can be changed under the "Button Threshold Settings" tab. These determine when the sensor will transition into a Proximity state (meant as a wake-up flag) or a Touch state (to indicate that the user is interacting with the device, or to engage freewheeling).



Button Thresholds

Prox Threshold: 20 — 20 Counts

Prox Debounce Enter: 2

Prox Debounce Exit: 1

Touch Threshold: 20 — 7,8 % × LTA

Touch Hysteresis: % of Touch Threshold 52 — 20,28 %

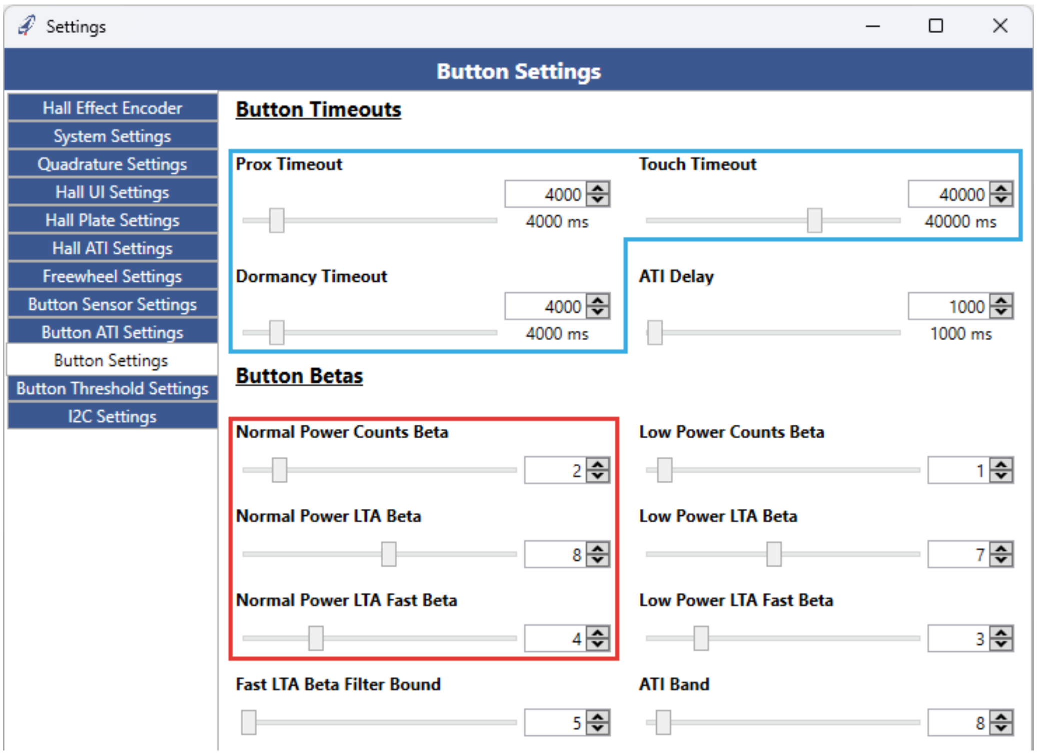
Figure 6.3: Button Thresholds

6.4 Button Timeouts

The **Prox** and **Touch Timeouts** determine for how long the touch channel may stay in either the Proximity or Touch state before resetting and automatically performing an ATI routine. This is meant to clear stuck states automatically.

The **Dormancy Timeout** determines how soon after the most recent prox or touch event the Dormant flag will be set. Recall that the device will enter lower power modes as soon as the magnet is *stationary* and the touch channel is *dormant*.

The timeout settings are available under the “Button Settings” tab, shown in Figure 6.4.



Button Timeouts

Prox Timeout: 4000 — 4000 ms

Touch Timeout: 40000 — 40000 ms

Dormancy Timeout: 4000 — 4000 ms

ATI Delay: 1000 — 1000 ms

Button Betas

Normal Power Counts Beta: 2

Low Power Counts Beta: 1

Normal Power LTA Beta: 8

Low Power LTA Beta: 7

Normal Power LTA Fast Beta: 4

Low Power LTA Fast Beta: 3

Fast LTA Beta Filter Bound: 5

ATI Band: 8

Figure 6.4: Button Timeout and Filter Settings

6.5 Button Filter Betas

The final touch sensor settings to be discussed are the **Button Filter Betas**, highlighted in Figure 6.4. The filter betas determine the strength of the filtering. Larger beta values remove more high-frequency content from the signals, and tend to cause the filter to respond more slowly to changes.

Both the raw counts and the LTA are filtered with beta filters. The counts filter should react quickly to user input, and the beta value should therefore be kept at smaller values such as 2 or less. The LTA is meant to track slowly-changing environmental effects, but not user interactions, so beta values of 7 or larger are recommended.

There are two additional behaviours of the LTA that are worth noting.

1. The LTA filter is halted during prox or touch events. This means that the LTA will not be updated, but will keep its value fixed so long as the channel is in a prox or touch state.
2. The LTA filter will switch to the **LTA Fast Beta** when counts are larger than the LTA. A case where the counts are above the LTA signifies that the capacitance of the overall system has decreased, or the LTA has incorrectly drifted too low. To compensate for this situation, the LTA follows the counts more quickly, with a smaller beta value.

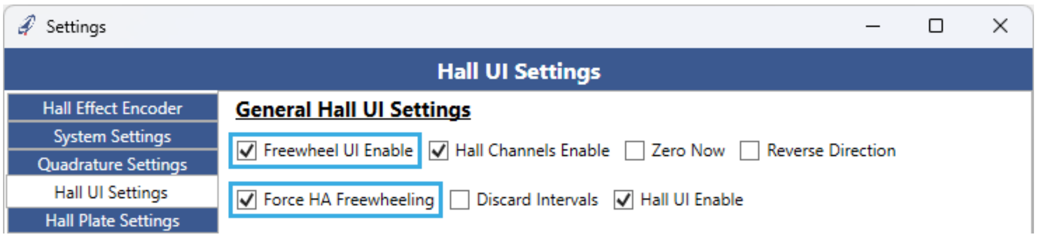
These two behaviours are worth noting because these behaviours are *disabled* when using the freewheeling UI.

7 Configuring Freewheeling UI

The virtual freewheeling feature unifies the Hall-rotation sensing and touch sensing into a UI that emulates a freely-spinning wheel with high inertia, and allows for scrolling through long documents and webpages quickly with a simple flick of the wheel. The IQS7221E tracks the rotation speed of the magnet, and upon detection of a release (i.e. a sudden increase in counts, or a large delta with the touch Direction bit cleared) on the touch channel, freewheeling is activated, and the device clocks out intervals and quadrature pulses as though the wheel is rapidly spinning. This simulated spinning wheel decelerates slowly over time, until it gradually stops. Alternatively, the user can instantly stop the freewheeling by simply touching the touch sensor.

This section describes how to enable and configure the Freewheeling UI. Before attempting to set up freewheeling, the Hall and Button channels must already be configured as described in Sections 5 and 6.

To enable freewheeling, select the **Enable Freewheeling** and **Force HA Freewheeling** options in Hall UI Settings, and enable the **Release UI** in Button Sensor Settings, as shown in Figure 7.1 and Figure 7.2. Note that this will automatically enable **Bi-directional Sensing** on the touch channel.

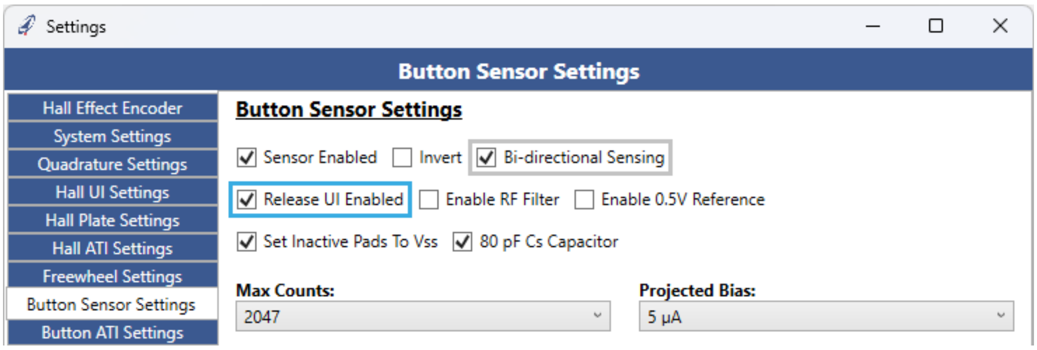


General Hall UI Settings

☑ Freewheel UI Enable ☑ Hall Channels Enable □ Zero Now □ Reverse Direction

☑ Force HA Freewheeling □ Discard Intervals ☑ Hall UI Enable

Figure 7.1: Enable Freewheeling in Hall UI Settings



Button Sensor Settings

☑ Sensor Enabled □ Invert ☑ Bi-directional Sensing

☑ Release UI Enabled □ Enable RF Filter □ Enable 0.5V Reference

☑ Set Inactive Pads To Vss ☑ 80 pF Cs Capacitor

Max Counts: 2047

Projected Bias: 5 µA

Figure 7.2: Enable Release UI in Button Sensor Settings

Now, spin the EV kit's control knob and release it quickly. The angle and interval values should continue changing as though the magnet is continuously spinning, even though the knob may stop moving. This can be seen in Figure 7.3, where the angle value continues to increase long after the actual magnet (raw angle measurement) has stopped moving.

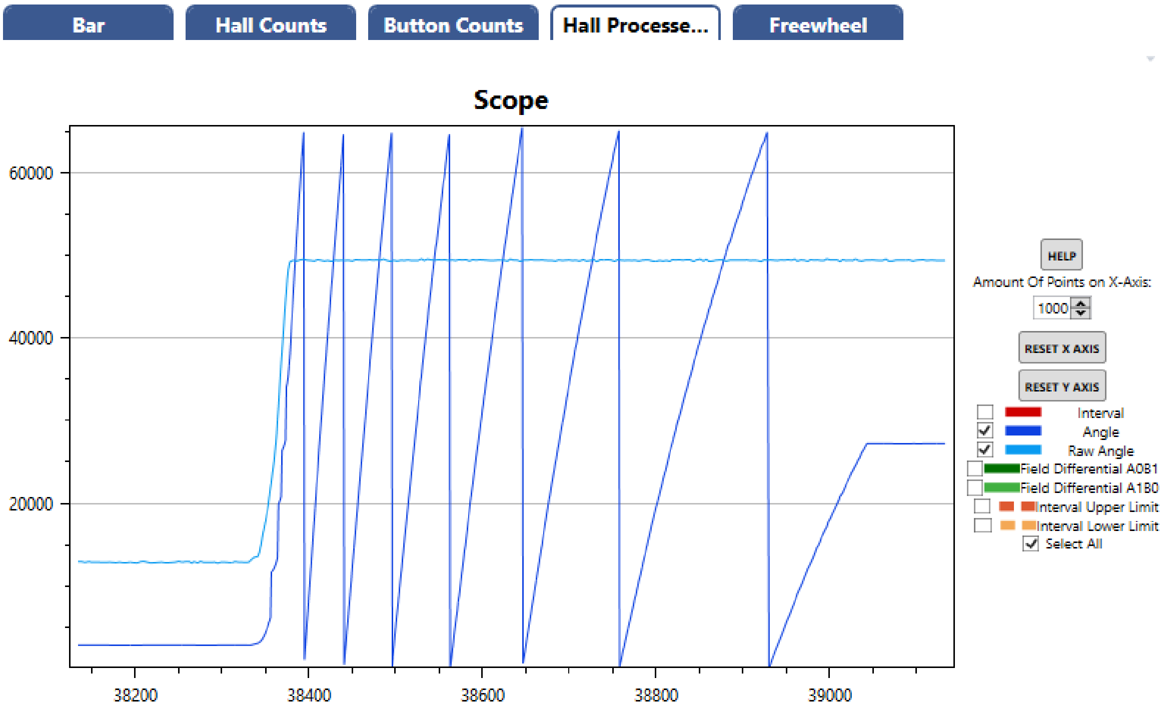
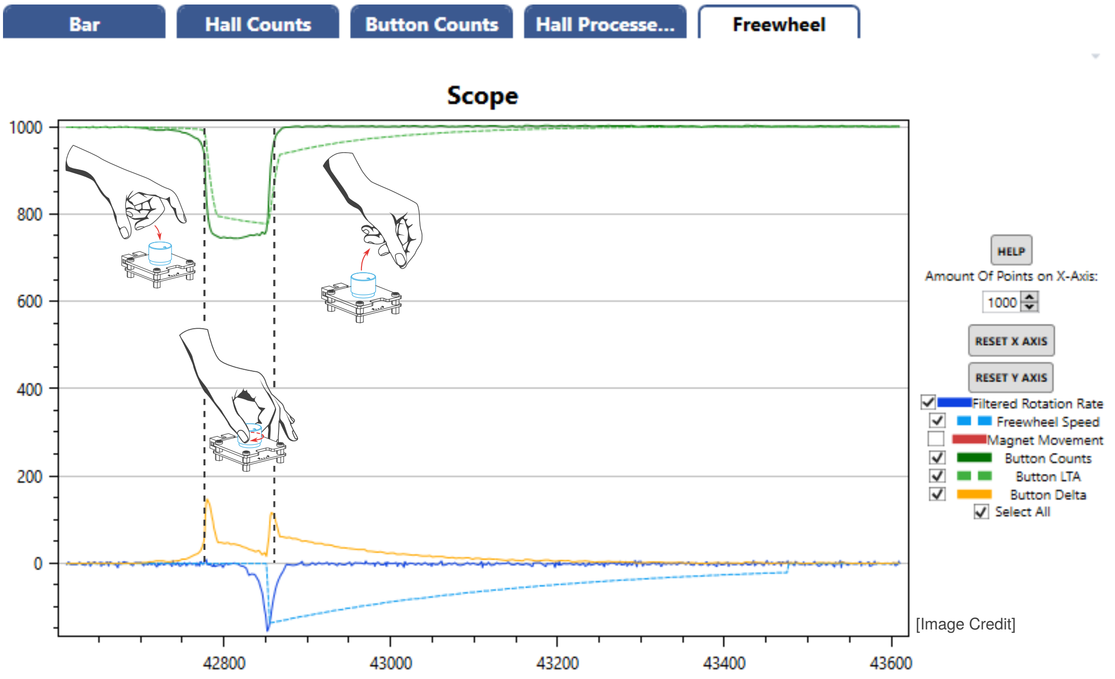


Figure 7.3: Freewheeling Angle Outputs

The freewheeling parameters can be evaluated in the “Freewheel” scope view, as shown in Figure 7.4. The Button Counts and Button LTA plots show the behaviour of the touch channel with Release UI enabled. The LTA is never halted, even when in touch, and always follows the counts value quickly. As a result, when a user performs a touch, there is a brief spike in the delta value which quickly decays away. During the touch (counts < LTA), the direction flag of the touch channel is set. When the touch sensor is released, there is again a brief spike in delta, with the direction flag cleared (counts > LTA).



[Image Credit]

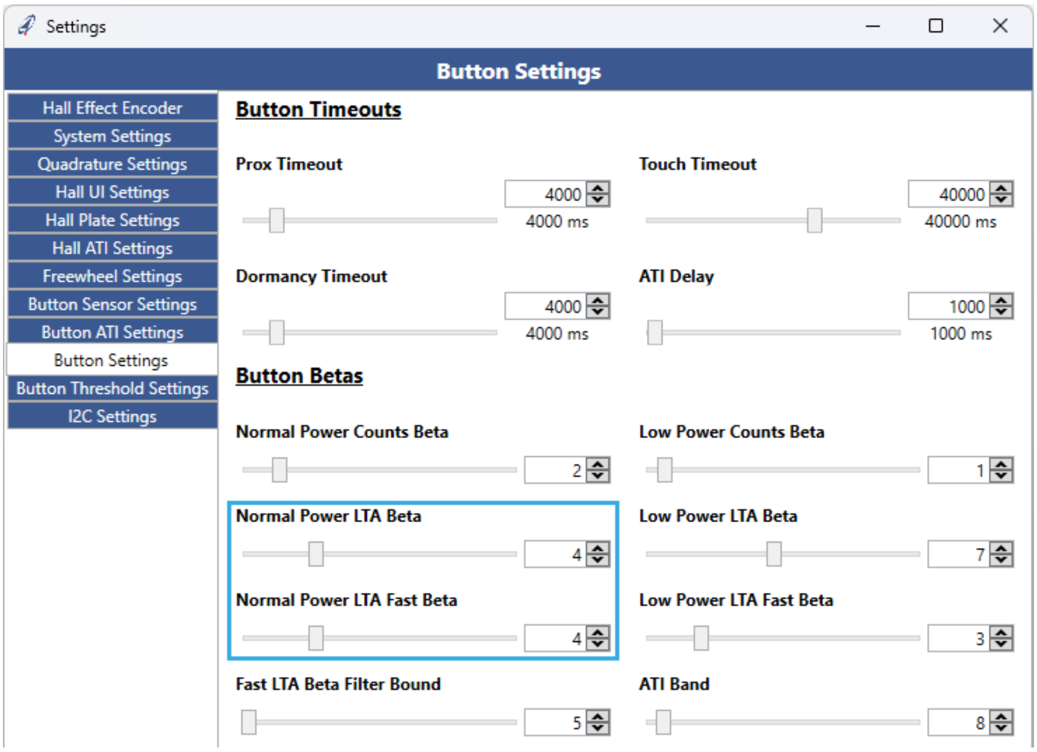
Figure 7.4: Freewheeling Parameters Evaluated in Scope View

A freewheel event is activated if the touch channel is *released* while the magnet is rotating above a threshold speed. Figure 7.4 shows the speed of the magnet's rotation, listed as the **Filtered Rotation Rate**. During the touch release, this value must be larger than the **Freewheel Minimum Start Speed** (see Section 7.2.2).

The **Freewheel Speed** plot shows the simulated speed of the virtual wheel. Upon release, the freewheel speed is set to the current magnet speed, and slowly decays over time. The rate of decay is dependent on the inertia, friction, and damping parameters. Once the simulated freewheel speed drops below the **Freewheel Stop Threshold**, freewheeling stops. The Freewheel Stop Threshold is explained further in Section 7.2.3.

7.1 Release UI Parameters

With the Release UI enabled, the LTA no longer halts during touch events. Instead, the LTA constantly tracks the current counts, and a touch event will only show a large delta for a brief moment, as seen in Figure 7.4. The IQS7221E's release UI uses a combination of the Normal Power LTA Beta and Normal Power Fast LTA Beta to filter the LTA. For now, it is recommended to keep them the same value.



Button Timeouts

Prox Timeout: 4000 — 4000 ms

Touch Timeout: 40000 — 40000 ms

Dormancy Timeout: 4000 — 4000 ms

ATI Delay: 1000 — 1000 ms

Button Betas

Normal Power Counts Beta: 2

Low Power Counts Beta: 1

Normal Power LTA Beta: 4

Low Power LTA Beta: 7

Normal Power LTA Fast Beta: 4

Low Power LTA Fast Beta: 3

Fast LTA Beta Filter Bound: 5

ATI Band: 8

Figure 7.5: Set LTA Betas for Freewheeling

During a rapid wheel flick, the LTA must follow the counts fast enough to generate separate delta "events" for the touch and the release. A starting value of 4 is recommended for the LTA beta values, but this can be reduced further if necessary. However, by reducing the beta value, smaller deltas can be expected during touch and release events, thus the touch thresholds may need to be lowered.

When freewheeling is enabled, instead of using the original Button Sensor touch threshold, the IQS7221E uses the Freewheel Touch Thresholds. These thresholds are:

- **Forward Release Threshold**: Touch threshold during a "forward" magnet rotation to start freewheeling.
- **Reverse Release Threshold**: Touch threshold during a "reverse" magnet rotation to start freewheeling.

- **Stop Threshold**: Touch threshold to stop freewheeling.

The behaviour of these thresholds is shown in Figure 7.6. The location of the settings is shown in Figure 7.7.

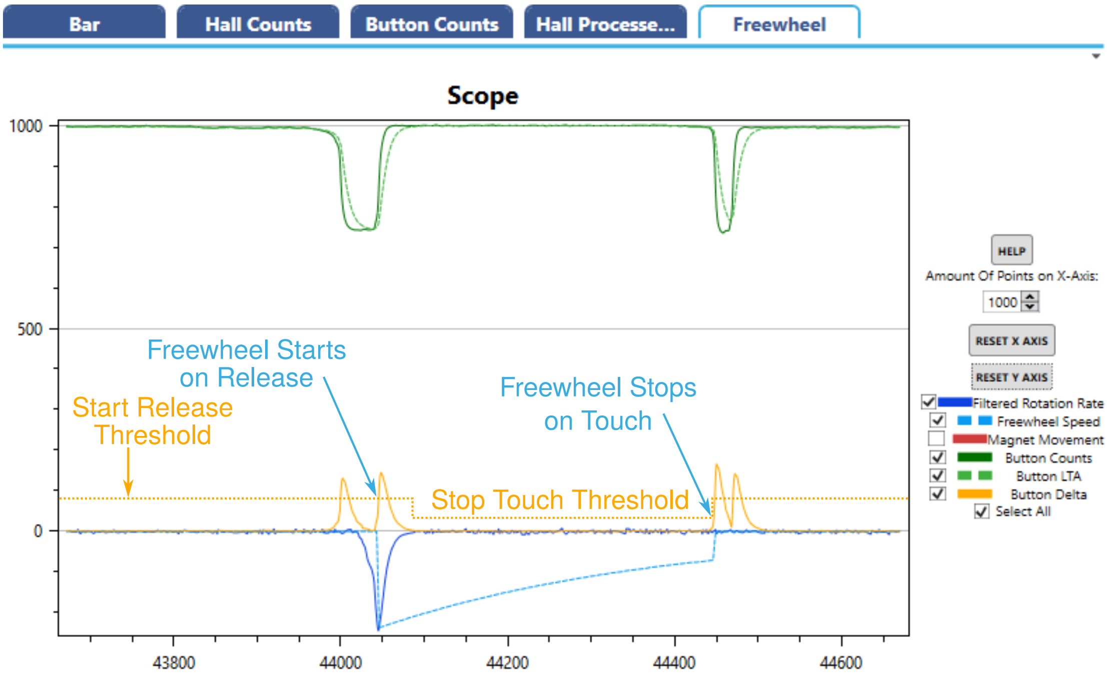
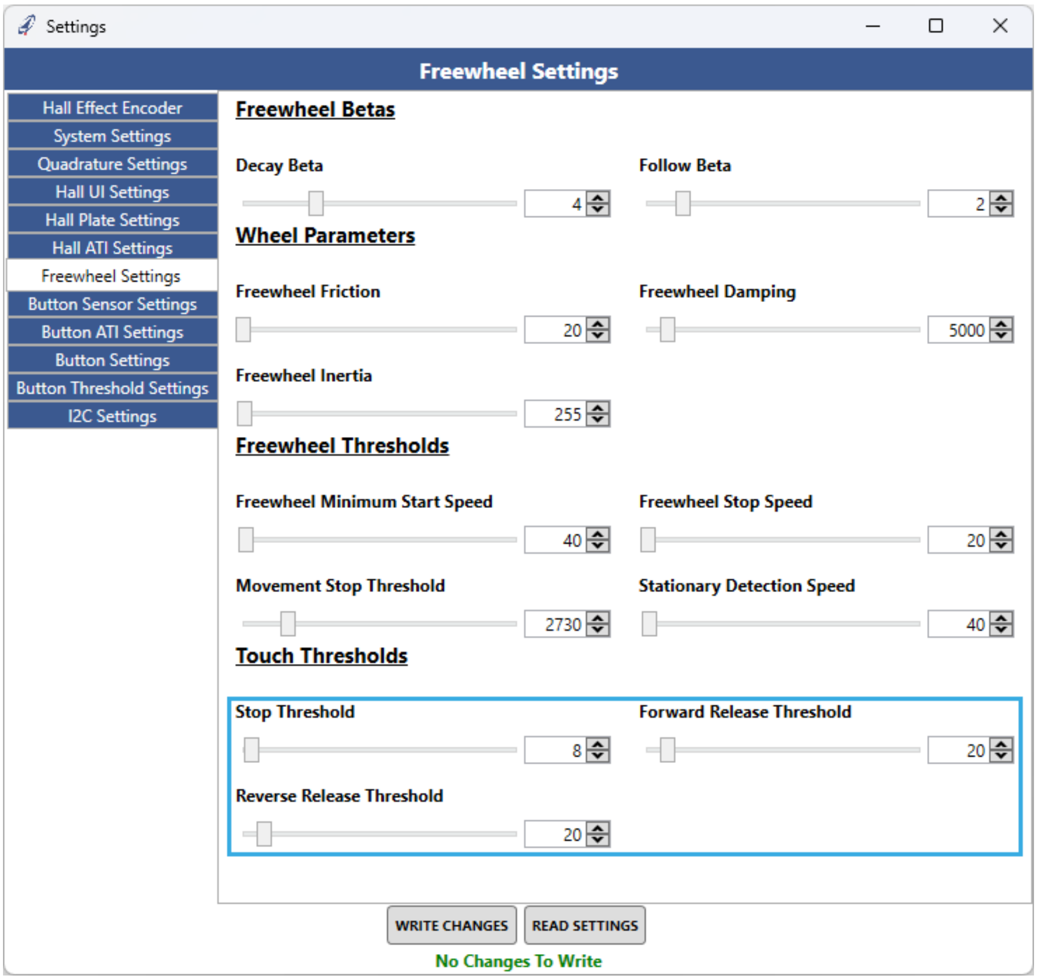


Figure 7.6: Freewheel Touch Thresholds

Having two separate thresholds for forward and reverse rotation is useful in certain applications, such as mouse scroll wheels. When the user flicks backwards to scroll down, the finger moves completely away from the wheel, creating a large delta. When the user flicks forwards to scroll up, the finger tends to stay above the wheel, resulting in a reduced delta, and therefore requiring a smaller threshold. Finally, the stop threshold can be used to change how easily the user can stop the freewheeling.



Freewheel Betas

Decay Beta 4

Follow Beta 2

Wheel Parameters

Freewheel Friction 20

Freewheel Damping 5000

Freewheel Inertia 255

Freewheel Thresholds

Freewheel Minimum Start Speed 40

Freewheel Stop Speed 20

Movement Stop Threshold 2730

Stationary Detection Speed 40

Touch Thresholds

Stop Threshold 8

Forward Release Threshold 20

Reverse Release Threshold 20

No Changes To Write

Figure 7.7: Touch Thresholds Used to Start and Stop Freewheeling

7.1.1 Touch LTA Betas

This section is only applicable to the IQS7221E101 product version, not to IQS7221E001.

When Release UI is enabled, the LTA is filtered with the slower LTA beta value by default. Once a touch event or release event is detected, the LTA temporarily switches to the fast LTA beta, until the delta decays to below the touch threshold. This aims to increase sensitivity to rapid touch-and-release events.

Recommended values for the LTA betas are 8 for the “normal” LTA beta, and 4 or lower for the fast LTA beta.

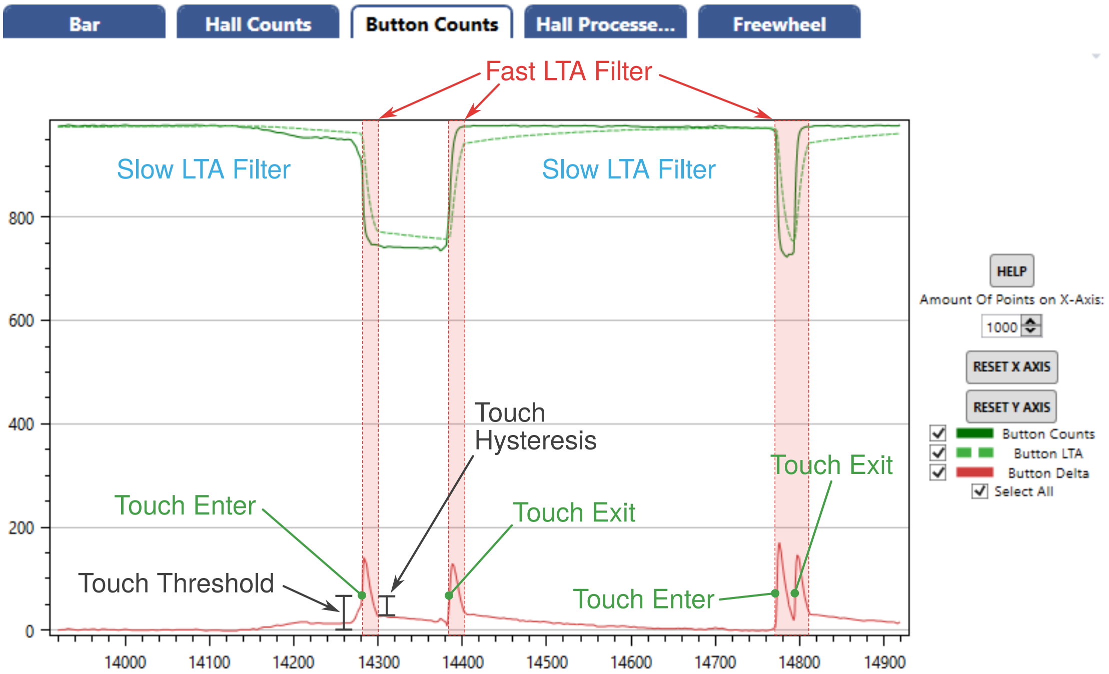
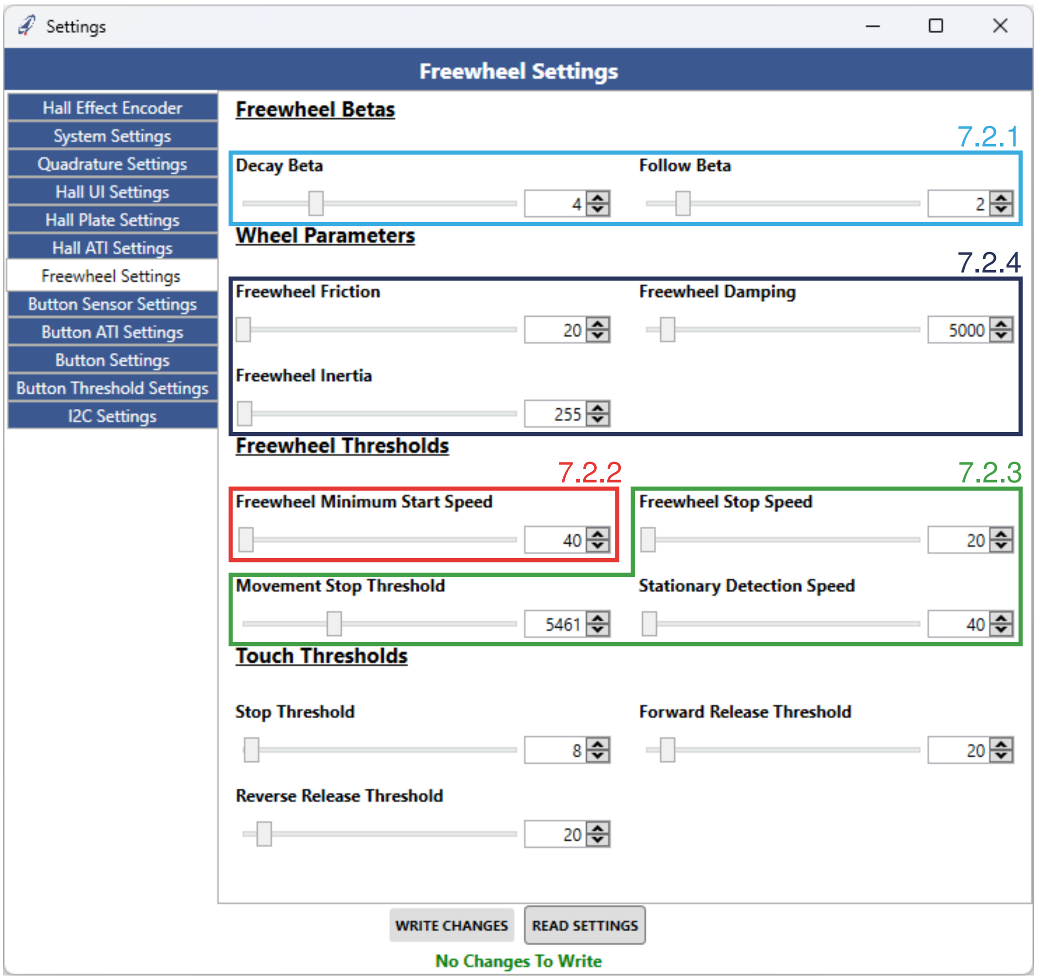


Figure 7.8: Release UI LTA Beta Behaviour

Figure 7.8 shows the operation of the Release UI with an LTA Beta of 8, a Fast LTA Beta of 4, and a Touch Hysteresis of 50 %.

7.2 Freewheel Settings

The following sections describe the settings related to freewheeling speed and filtering. All settings are available in the “Freewheel Settings” tab.



Freewheel Betas

Decay Beta: 4
Follow Beta: 2

Wheel Parameters

Freewheel Friction: 20
Freewheel Damping: 5000
Freewheel Inertia: 255

Freewheel Thresholds

Freewheel Minimum Start Speed: 40
Freewheel Stop Speed: 20
Movement Stop Threshold: 5461
Stationary Detection Speed: 40

Touch Thresholds

Stop Threshold: 8
Forward Release Threshold: 20
Reverse Release Threshold: 20

No Changes To Write

Figure 7.9: Freewheel UI Settings

7.2.1 Filtered Magnet Speed

The IQS7221E tracks the instantaneous speed of the magnet, and filters it to prevent noise from falsely crossing the freewheel start threshold. The filter switches between two beta values:

- **Follow Beta** is a fast filter that tracks *increases* in magnet rotation speed.
- **Decay Beta** is a slower filter that tracks *decreases* in magnet rotation speed. If the magnet stops moving, the filtered speed will slowly decay to zero, keeping the speed above the freewheel start threshold long enough for the user to release the touch sensor and initiate freewheeling.

The output of this filter is the **Filtered Rotation Rate**, shown in the Freewheel scope.

7.2.2 Starting Freewheeling

In order to initiate a freewheeling gesture, this filtered rotation rate must be above the **Freewheel Minimum Start Speed** when the release on the touch sensor occurs. Lowering this minimum speed allows the user to perform slower freewheel gestures. However, noise on the filtered speed must be considered when selecting this threshold.

Once the freewheeling has begun, the simulated freewheeling speed is set to the filtered rotation rate, and will decay until it reaches the Freewheel Stop Speed. This process is shown in Figure 7.10.

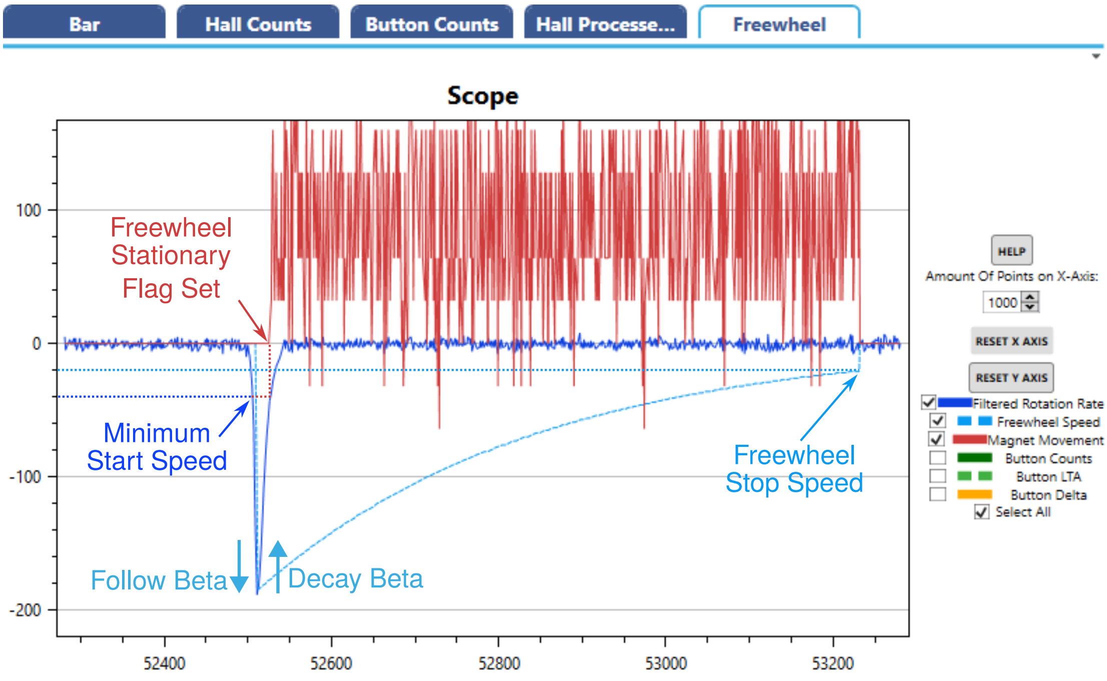


Figure 7.10: Freewheel Speed Thresholds

7.2.3 Stopping Freewheeling

There are three ways to stop a freewheeling event:

- Letting the freewheeling speed decay until it reaches the **Freewheel Stop Speed** threshold.
- Performing a touch on the capacitive sensor.
- Moving the wheel.

As mentioned above, without any additional interaction from the user, freewheeling will continue until the freewheeling speed decays and reaches the Freewheel Stop Speed.

The second method to stop freewheeling is to perform a quick touch on the capacitive sensor. In a mouse application, if the user is scrolling quickly through a document with freewheeling, they can very precisely stop at a desired point simply by tapping lightly on the wheel.

However, the Release UI may not always detect a touch if the user approaches the capacitive sensor slowly. This is due to the LTA that follows the counts quickly. In such a case, freewheeling can also be cancelled if the user rotates the wheel by a certain amount. This movement threshold is the **Movement Stop Threshold**, and is a 16-bit value, related to the final angle. For example, it can be set to the size of one interval. For 12 intervals, this was calculated earlier as 5461.

The IQS7221E starts tracking any additional wheel movement as soon as it detects that the magnet/wheel is stationary. This is signalled with the **Stationary** freewheel flag that gets set when the filtered magnet speed drops below the **Stationary Detection Speed**. (Recall that the Freewheel Stationary flag is different from the Hall UI's Stationary flag.) The **Magnet Movement** graph is available on the Freewheel scope.

7.2.4 Adjusting Freewheel Decay Rate

During freewheeling, the freewheeling speed, ω, slowly decays based on the following variables:

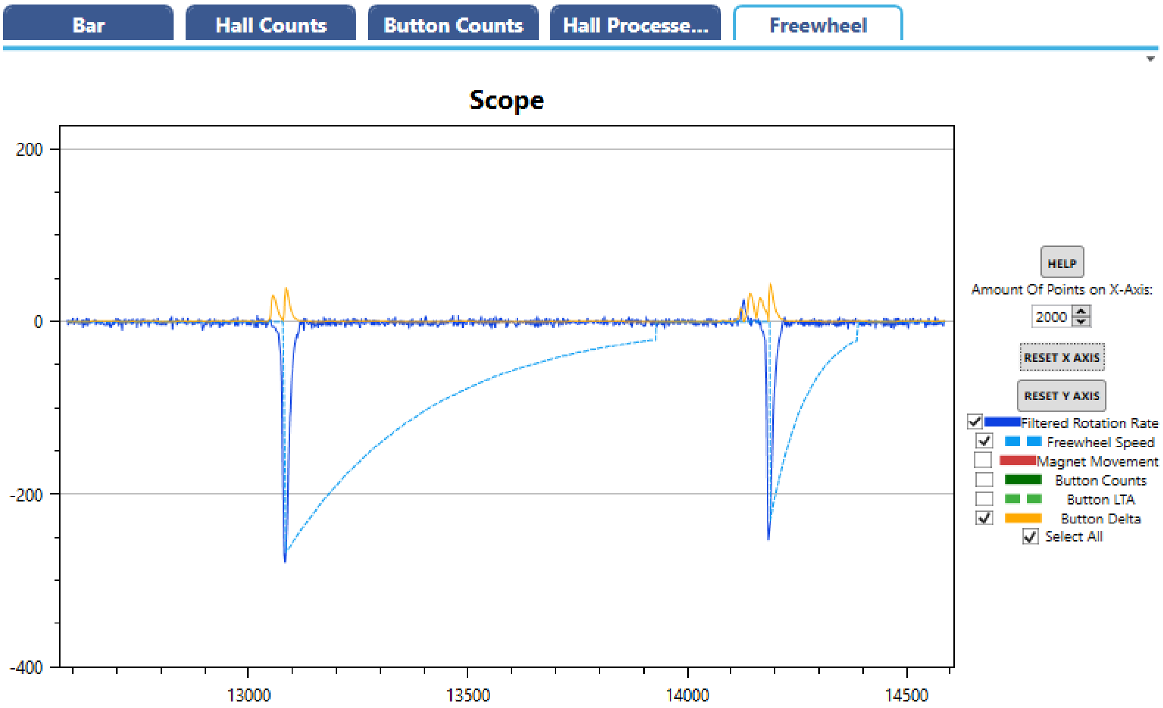
- **Friction**, F, provides a constant deceleration to the freewheeling speed.
- **Damping**, D, applies deceleration proportional to the freewheeling speed. The virtual wheel decelerates more quickly at higher speeds.
- **Moment of inertia**, I, that simulates the mass of the virtual wheel.

The deceleration of the virtual wheel, $\dot{\omega}$, can be expressed over time by the following formula:

$$\dot{\omega}(t) = -\frac{F + D \cdot \omega(t)}{I}$$

Note that the moment of inertia parameter essentially acts as a scaling factor for the two integer values, friction and damping. Thus, a large inertia value can be used to provide resolution, and then only the friction and damping parameters need to be adjusted to change the decay rate of the freewheeling speed.

For example, the scope capture in Figure 7.11 shows the effect of increasing damping from 5000 to 20,000.



Amount Of Points on X-Axis: 2000

☑ Filtered Rotation Rate
☑ Freewheel Speed
☐ Magnet Movement
☐ Button Counts
☐ Button LTA
☑ Button Delta
☑ Select All

Figure 7.11: Freewheeling Angle Outputs

Alternatively, with fixed friction and damping parameters, only the inertia value needs to be changed to change the duration of the freewheeling. Setting a higher inertia value will increase the freewheeling duration, and vice versa.

8 Conclusion

This document provided an overview of how to configure the IQS7221E device for a Hall-rotation application using the IQS7221E Debug and Display Tool(GUI), and provided explanations and recommendations for the most commonly-used settings. The IQS7221E Evaluation Kit was used as an example to show how to configure the IQS7221E in a control knob application that used the Hall-effect sensors for the rotation measurement, and a self-capacitance sensor on the knob to detect user touch.

This document described how to install and navigate the IQS7221E GUI, and explained the functions of the IQS7221E button controls, sensor graphs, and flags. The document then detailed the configuration of the IQS7221E settings, starting with the system settings and power modes. The Hall UI was configured to have sufficient gain on the Hall sensors, an appropriate number of intervals was chosen, the correct auto zero mode was selected, and filter parameters were updated. The Touch channel was configured as a self-capacitance sensor, with appropriate filtering and thresholds. Finally, the document described how the Freewheeling UI combines the Hall UI and Touch channel to provide the effect of a freely-spinning wheel.

For more information regarding any IQS7221E settings not covered in this document, please contact Azoteq.

A Evaluation Kit Details

This section provides a brief overview of the IQS7221E Hall-Rotation Evaluation Kit design.

A.1 Assembly

The IQS7221E EV kit illustrates the design of a control knob that combines two sensors; the magnet in the shaft supplies the diametric magnetic field to measure the angle of the knob, and the metal body of the knob allows it to act as a capacitive touch sensor. The capacitive electrode on the PCB couples to the metal shaft and magnet. A user that touches anywhere on the metal knob adds a large capacitance to earth.

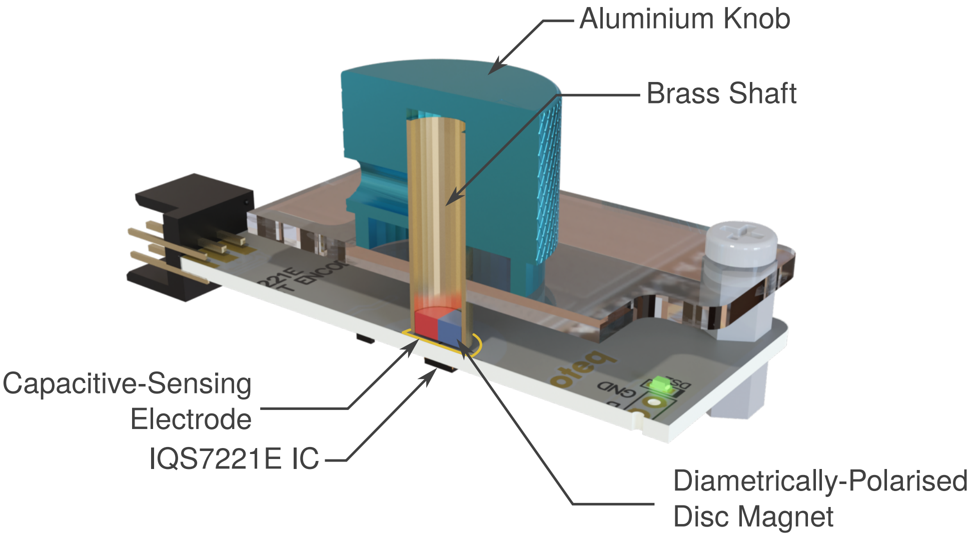

Aluminium Knob
Brass Shaft
Capacitive-Sensing Electrode
IQS7221E IC
Diametrically-Polarised Disc Magnet

Figure A.1: Cross-Section View of IQS7221E EV Kit

This design allows for an incredibly simple assembly, but requires both the IQS7221E IC and the capacitive electrode to be directly below the magnet. The EV kit places the IC on the bottom side of the PCB, away from the magnet, leaving the PCB's top layer free for the capacitive electrode to be directly beneath the magnet for strong capacitive coupling. This provides good sensitivity on the touch channel, but reduces the strength of the measured magnetic field. Not only does this force the Hall plates further away from the magnet, but a drop in measured magnetic field strength of approximately 15 % can be expected when measuring magnet fields from the underside of the IC. This low magnetic field strength is apparent when adjusting the sensitivity of the Hall sensors as described in Section 5.3, as the EV kit requires a large amount of gain to achieve sufficient swing on the Hall counts.

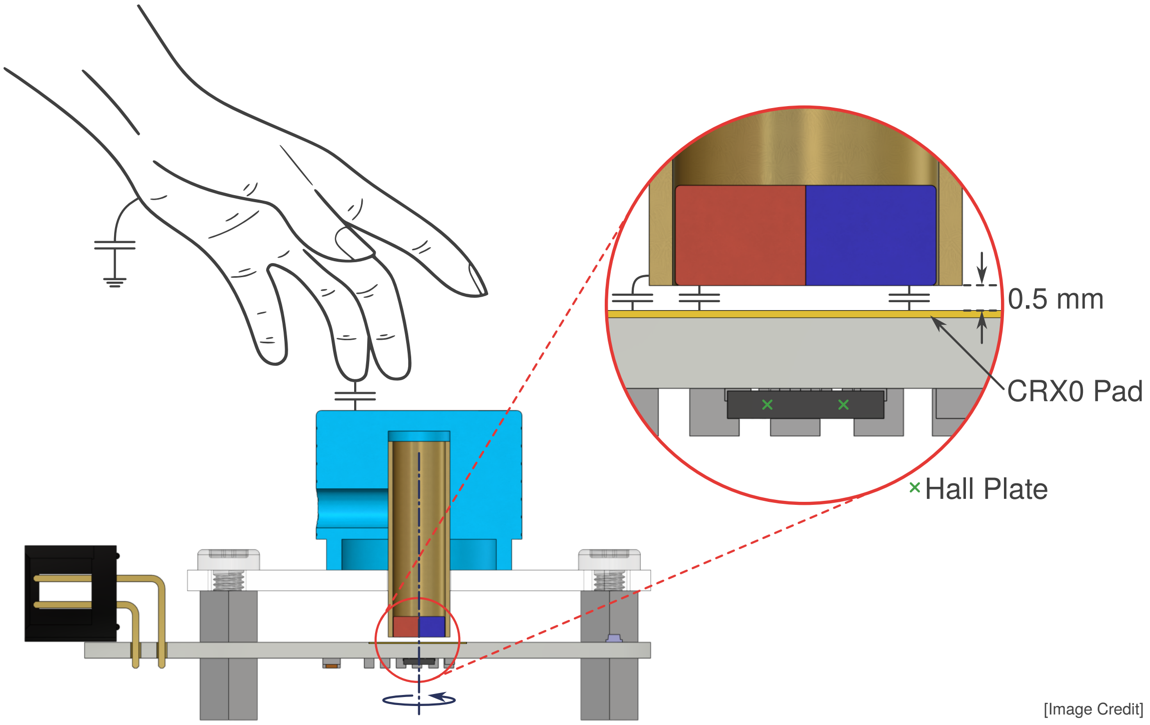


Figure A.2: Cross-Section View of IQS7221E EV Kit

To increase the Hall signals for a higher-resolution application, the design may be altered in the following ways:

- Use a stronger magnet.
- Move the magnet closer to the IC. This can be done by reducing the 0.5 mm gap shown in Figure A.2 or by reducing the PCB thickness. Reducing the gap will also increase the capacitive coupling.
- Place the IC on the top layer of the PCB, on the same side as the magnet. This option will require a different capacitive electrode design, or if it is not needed, the capacitive sensor can be removed altogether.

A.2 Reference Schematic

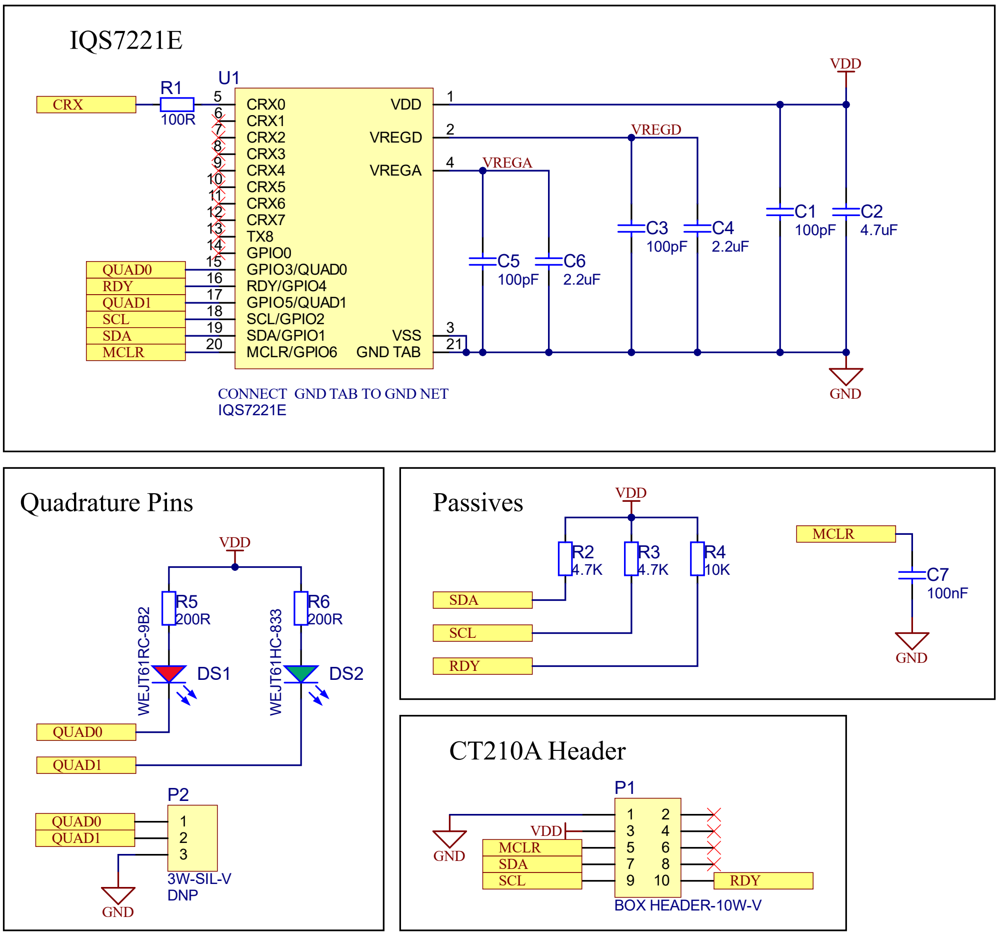


Figure A.3: IQS7221E Evaluation Kit Schematic

Contact Information

	USA	Asia	South Africa
Physical Address	11940 Jollyville Suite 120-S Austin TX 78759 USA	Room 501A, Block A T-Share International Centre Taoyuan Road, Nanshan District Shenzhen, Guangdong, PRC	1 Bergsig Avenue Paarl 7646 South Africa
Postal Address	11940 Jollyville Suite 120-S Austin TX 78759 USA	Room 501A, Block A T-Share International Centre Taoyuan Road, Nanshan District Shenzhen, Guangdong, PRC	PO Box 3534 Paarl 7620 South Africa
Tel	+1 512 538 1995	+86 755 8303 5294 ext 808	+27 21 863 0033
Email	info@azoteq.com	info@azoteq.com	info@azoteq.com

Visit www.azoteq.com
for a list of distributors and worldwide representation.

Patents as listed on www.azoteq.com/patents-trademarks/ may relate to the device or usage of the device.

Azoteq®, Crystal Driver®, IQ Switch®, ProxSense®, ProxFusion®, LightSense™, SwipeSwitch™, and the logo are trademarks of Azoteq.

The information in this Datasheet is believed to be accurate at the time of publication. Azoteq uses reasonable effort to maintain the information up-to-date and accurate, but does not warrant the accuracy, completeness or reliability of the information contained herein. All content and information are provided on an "as is" basis only, without any representations or warranties, express or implied, of any kind, including representations about the suitability of these products or information for any purpose. Azoteq disclaims all warranties and conditions with regard to these products and information, including but not limited to all implied warranties and conditions of merchantability, fitness for a particular purpose, title and non-infringement of any third party intellectual property rights. Azoteq assumes no liability for any damages or injury arising from any use of the information or the product or caused by, without limitation, failure of performance, error, omission, interruption, defect, delay in operation or transmission, even if Azoteq has been advised of the possibility of such damages. The applications mentioned herein are used solely for the purpose of illustration and Azoteq makes no warranty or representation that such applications will be suitable without further modification, nor recommends the use of its products for application that may present a risk to human life due to malfunction or otherwise. Azoteq products are not authorized for use as critical components in life support devices or systems. No licenses to patents are granted, implicitly, express or implied, by estoppel or otherwise, under any intellectual property rights. In the event that any of the abovementioned limitations or exclusions does not apply, it is agreed that Azoteq's total liability for all losses, damages and causes of action (in contract, tort (including without limitation, negligence) or otherwise) will not exceed the amount already paid by the customer for the products. Azoteq reserves the right to alter its products, to make corrections, deletions, modifications, enhancements, improvements and other changes to the content and information, its products, programs and services at any time or to move or discontinue any contents, products, programs or services without prior notification. For the most up-to-date information and binding Terms and Conditions please refer to www.azoteq.com.