iC-TW28 10-BIT SIN/COS INTERPOLATOR WITH AUTO-CALIBRATION AND LINE DRIVER

FEATURES

- Differential PGA inputs for sine, cosine, and index
- Input frequency of up to 700 kHz
- AB output frequency of up to 12.5 MHz
- Automatic compensation of amplitude, offset, and phase errors
- Low latency (typ. 1.5 µs)
- Differential RS422 line driver outputs for ABZ or UVW
- Simultaneous single-ended outputs for ABZ and UVW
- Digital filtering for ultra-low output jitter
- Complete status and fault monitoring capabilities
- Configured by pins or SPI
- In-field re-configuration via Encoder Link interface
- Easy to use with built-in line driver, EEPROM, and oscillator
- Push-button automatic calibration for fast commissioning
- LED intensity control by PWM output
- 10-bit angle data and 14-bit multi-cycle counter available to SPI
- Capture register for coded reference marks and touch-probe applications
- Space-saving 5 x 5 mm QFN package
- Single 3.3 V supply

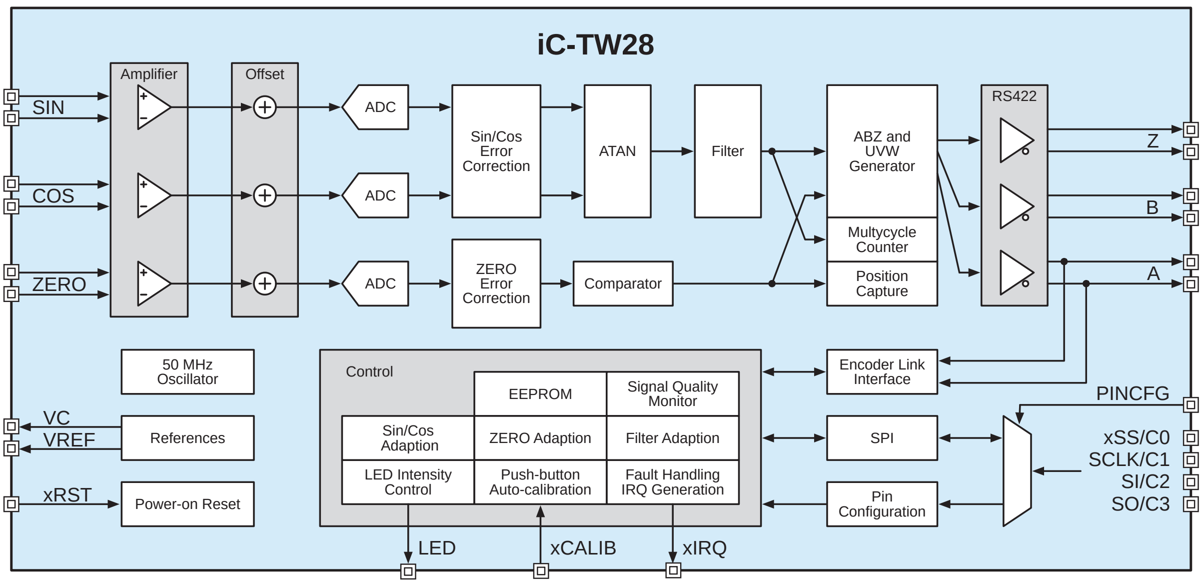
APPLICATIONS

- Rotary and linear encoders
- Magnetic or optical sin/cos sensor interface
- Brushless motor commutation (2...64 poles)
- Embedded motion control

PACKAGES

32-pin QFN
5 mm x 5 mm x 0.9 mm
RoHS compliant

BLOCK DIAGRAM



DESCRIPTION

The iC-TW28 is a general purpose 10-bit application-specific interpolator for sine/cosine signals with automatic calibration and adaption of signal path parameters during operation to maintain minimum angular error and jitter. Angular position is calculated at a programmable resolution of up to 1024 increments per input cycle. Automatic calibration and adaption (correction during operation) of sensor offset, sin/cos amplitude match, and phase quadrature is provided. Additionally, automatic calibration of gain, offset and phase of the zero inputs allows for rapid commissioning.

The iC-TW28 accepts 20 mV to 2 V differential Sin/-Cos input signals directly from magnetic or optical sensors – no external signal conditioning is required in most applications. The differential zero input accepts a wide range of digital and analog index gating sources such as Hall or MR sensor bridges. The Z output width, position relative to the sin/cos inputs, and synchronization to the AB quadrature outputs is fully programmable.

In addition to industry-standard incremental ABZ quadrature output, the iC-TW28 provides optional UVW commutation output modes for 1 to 32 pole-pair motors and SPI angle and multi-cycle readout for embedded applications. The incremental ABZ quadrature output can be generated at a frequency of up to 12 MHz (20 ns edge spacing); the maximum output frequency can also be limited so as not to overwhelm connected counters or controllers.

In SPI mode, the iC-TW28 provides multi-cycle synchronization and reference mark capture functions to support absolute position systems. Higher input signal frequencies are allowed in SPI mode since the ABZ output frequency limitation is not applicable.

The iC-TW28 offers two configuration modes. Pin configuration mode provides simple, static configuration that does not require any programming or complicated calibration. Pin configuration mode uses a subset of the iC-TW28's complete capabilities including ABZ quadrature output, a limited choice of the most commonly used interpolation (resolution) and hysteresis values, and one-button calibration. Eight resistors set voltage levels at four configuration input pins to select all operating parameters, simplifying product assembly. One-button auto calibration sets input gain and compensates sensor offset and sin/cos channel gain match and phase with just a few input cycles and then stores the compensation values to the internal EEPROM.

In more sophisticated applications, serial configuration mode allows access to all iC-TW28 features. Complete device configuration using the bi-directional SPI or Encoder Link ports provides access to all resolutions (including fractional interpolation), fully programmable hysteresis, and advanced noise/jitter filtering, quality monitoring, and fault detection capabilities.

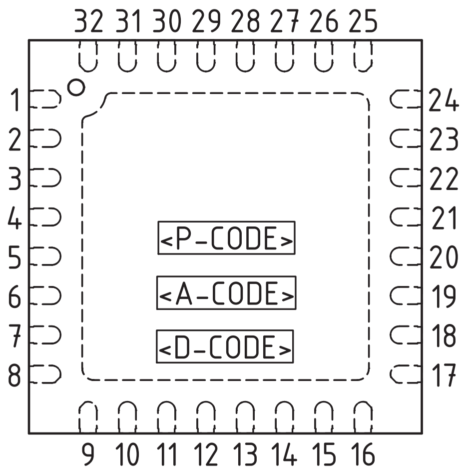
The iC-TW28 requires no external components for operation. An EEPROM for storage of configuration and calibration data, and RS422 line drivers are already integrated on-chip. An integrated power-on reset circuit can be overridden by an external hardware reset signal if necessary.

CONTENTS

PACKAGING INFORMATION

PIN CONFIGURATION QFN32-5x5

(top view)

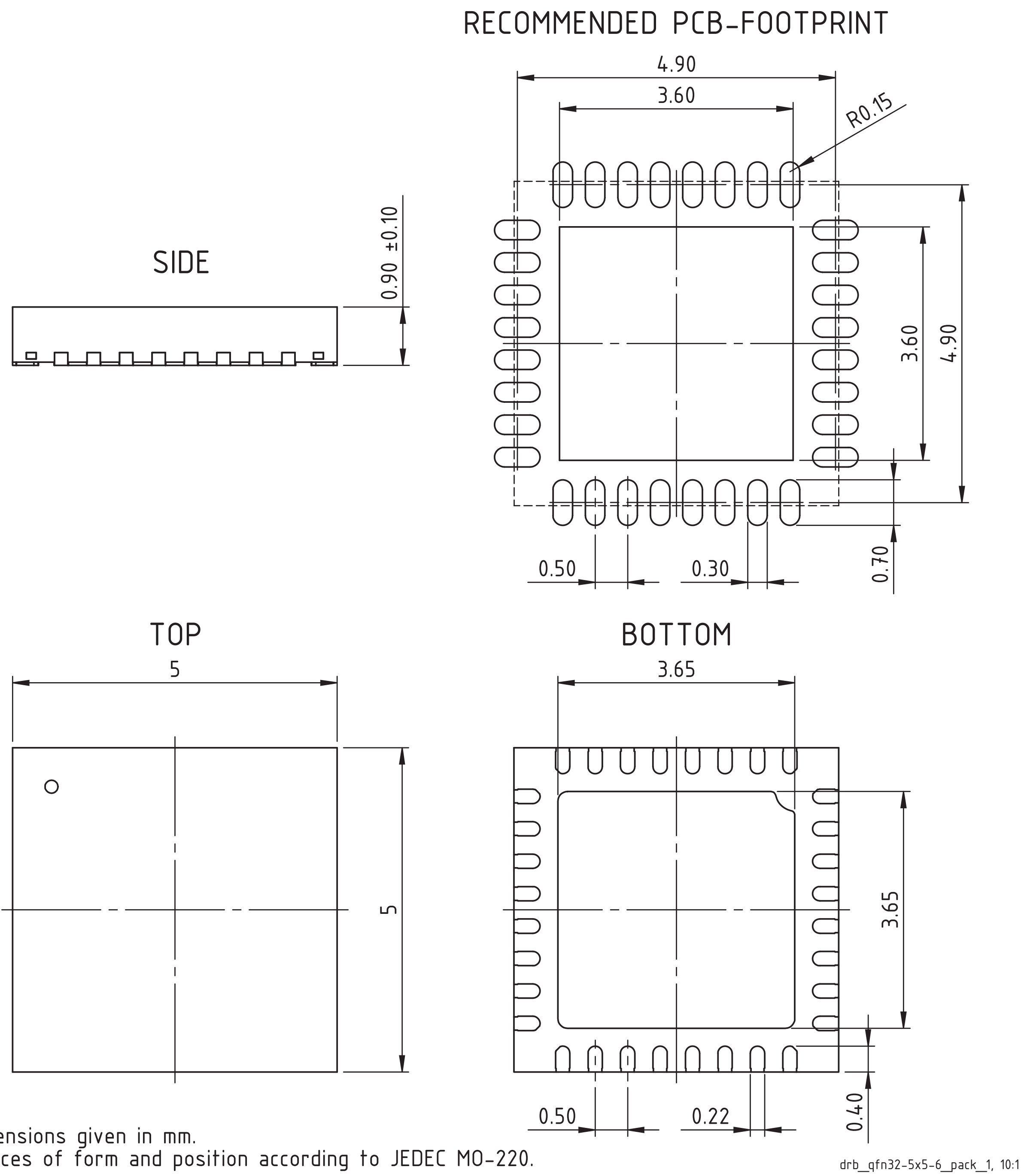


PIN FUNCTIONS

No.	Name	Function
1	SIN+[3]	+ Differential Sine Input
2	SIN–[3]	– Differential Sine Input
3	AVDD	+3.3 V Analog Power Supply Input
4	COS+[3]	+ Differential Cosine Input
5	COS–[3]	– Differential Cosine Input
6	AVSS	Analog Ground
7	ZERO+[4]	+ Differential Zero (Index) Input
8	ZERO–[3]	– Differential Zero (Index) Input
9	VREF	ADC Reference Voltage Output
10	VC	Bias Output (VDD/2)
11	Reserved[1]	
12	Reserved[1]	
13	Reserved[1, 6]	
14	xRST[4]	Reset Input (active low)
15	xCALIB[4]	Auto-Calibration Input (active low)
16	xIRQ[4]	Interrupt Request (active low) or Fault Output
17	Z–	– Differential RS422 Z or W Output
18	Z+	+ Differential RS422 Z or W Output
19	IOVSS	I/O Ground
20	B–	– Differential RS422 B or V Output
21	B+	+ Differential RS422 B or V Output
22	IOVDD	+3.3 V I/O Power Supply Input
23	A–	– Differential RS422 A or U Output
24	A+	+ Differential RS422 A or U Output
25	DVDD	+3.3 V Digital Power Supply Input
26	LED[4]	LED Intensity Control Output
27	DVSS	Digital Ground
28	SO/C3	SPI Slave Output or Configuration Input 3
29	SI/C2[3]	SPI Slave Input or Configuration Input 2
30	SCLK/C1[3]	SPI Clock Input or Configuration Input 1
31	xSS/C0[4]	SPI Slave Select Input or Configuration Input 0
32	PINCFG[5, 6]	Pin Configuration Select Input
	TP[2]	Backside paddle

IC top marking: <P-CODE> = product code, <A-CODE> = assembly code (subject to changes), <D-CODE> = date code (subject to changes);
[1] Must be connected to ground.
[2] Must be connected to a ground plane at AVSS potential. Can also be used to connect DVSS.
[3] Connect to ground via 10 kΩ resistor if not used. Do not allow to float.
[4] Connect to 3.3 V via 10 kΩ resistor if not used. Do not allow to float.
[5] Connect to 3.3 V (to DVDD for pin configuration) or ground (to DVSS for serial configuration). Do not allow to float.
[6] Note on compatibility: iC-TW29 uses pin 13 as GPIO, and pin 32 as BISSEN (high selects BiSS).

PACKAGE DIMENSIONS



All dimensions given in mm.
Tolerances of form and position according to JEDEC MO-220.

PIN FUNCTIONS

No.	Name	I/O	Function	Description
1 2	SIN+ SIN–	Analog in Analog in	Sine Input + Sine Input –	Differential sine signal input. For single ended sensors, SIN– must be biased to an appropriate DC level. Do not allow to float.
3	AVDD	Supply	Analog Power Supply	+3.1 V to +3.6 V supply input for analog circuitry. AVDD should be tied together with DVDD and IOVDD and supplied from a clean source.
4 5	COS+ COS–	Analog in Analog in	Cosine Input + Cosine Input –	Differential cosine signal input. For single ended sensors, COS– must be biased to an appropriate DC level. Do not allow to float.
6	AVSS	Ground	Analog Ground	AVSS must be tied to high quality ground, usually a solid PCB plane.
7 8	ZERO+ ZERO–	Analog in Analog in	Zero Input + Zero Input –	Differential Zero Gating Input. For single ended sensors, the unused input (either ZERO+ or ZERO–) must be biased to an appropriate DC level. Do not allow to float.
9	VREF	Analog out	Bias Output	Decouple with 100 nF capacitor to AVSS. Do not inject noise into this pin as it directly impacts ADC conversion quality.
10	VC	Analog out	Bias Output	Decouple with 100 nF capacitor to AVSS. Do not inject noise into this pin as it directly impacts ADC conversion quality.
11–13	Reserved	Digital inputs	Test Inputs	Reserved pins; must be connected to ground for normal operation.
14	xRST	Digital in, active low	Reset Input	The device is reset as long as xRST is low. An external RC network with at least R populated is recommended. Do not allow to float.
15	xCALIB	Digital in, active low	Calibration Control	Device enters calibration mode on falling edge of CALIB. This pin must be tied high if not used. Do not allow to float.
16	xIRQ	Digital out, active low	IRQ or Fault Output	Interrupt request output to external micro controller. Output can also be used to directly drive a fault LED in stand-alone applications. Can be configured as push-pull or open-drain. Do not allow to float.
17 18	Z– Z+	Digital/RS422 out Digital/RS422 out	Z– or W– Output Z+ or W+ Output	In ABZ output mode these are the differential Z outputs. In UVW output mode these are the W outputs.
19	IOVSS	Ground	I/O Ground	All ground pins must be connected to a high quality ground, usually a solid PCB plane.
20 21	B– B+	Digital/RS422 out Digital/RS422 out	B– or V– Output B+ or V+ Output	In ABZ output mode these are the differential B outputs. In UVW output mode these are the V outputs. In Z calibration mode these show the Z window used to gate the Z output.
22	IOVDD	Supply	Output Drivers Power Supply	+3.1 V to +3.6 V voltage terminal supplying all pin output drivers including the RS422 drivers and LED current. IOVDD and DVDD must be the same voltage level. IOVDD can require up to 100mA depending on loads. It is usually sufficient to tie IOVDD to the same supply as AVDD and DVDD.
23 24	A– A+	Digital/RS422 out Digital/RS422 out	A– or U– Output A+or U+ Output	In ABZ output mode these are the differential A outputs. In UVW output mode this these are the U outputs. In Z calibration mode these show the un-gated Z signal once per input period. With Encoder Link active, A+ is ELCLK input and A– is ELIN input or ELOUT output.
25	DVDD	Supply	Digital Power Supply	+3.1 V to +3.6 V supply voltage terminal for digital circuits. DVDD should be tied together with AVDD and IOVDD to a high quality supply.
26	LED	Digital output	LED PWM Output	Used to supply the illumination LED of optical sensors to maintain constant intensity and constant Sin/Cos sensor amplitude. Can be configured as push-pull or open-drain. Do not allow to float.
27	DVSS	Ground	Digital Ground	Pin must tied to high quality ground, usually a solid PCB plane.
28	SO/C3	Digital out, Analog in	SPI Slave Output, Pin Configuration Input 3	In serial configuration mode, this is the slave output and connects to an SPI master MI pin. In pin configuration mode, this is input C3. Do not allow to float.
29	SI/C2	Digital in, Analog in	SPI Slave Input, Pin Configuration Input 2	In serial configuration mode, this is the slave input and connects to an SPI master MO pin. In pin configuration mode, this is input C2. Do not allow to float.
30	SCLK/C1	Digital in, Analog in	SPI Slave Clock Input, Pin Configuration Input 1	In serial configuration mode, this is the slave clock input and connects to an SPI master clock output pin. In pin configuration mode, this is input C1. Do not allow to float.
31	xSS/C0	Digital in, Analog in	SPI Slave Select Input, Pin Configuration Input 0	In serial configuration mode, this is the slave select input and connects to an SPI master slave select output pin. In pin configuration mode, this is input C0. Do not allow to float.
32	PINCFG	Digital in	Configuration Mode Selection	For serial configuration mode, connect PINCFG to DVSS. For pin configuration mode, connect PINCFG to DVDD.

ABSOLUTE MAXIMUM RATINGS

These ratings do not imply operating conditions; functional operation is not guaranteed. Beyond these values damage may occur.

Item No.	Symbol	Parameter	Conditions	Min.	Max.	Unit
G001	VDD	Voltage at DVDD, AVDD, and IOVDD	Referenced to DVSS, AVSS, and IOVSS respectively	–0.3	4.1	V
G002	Vpin	Pin Voltage at any pin	Referenced to DVSS, AVSS, and IOVSS	–0.3	AVDD + 0.3	V
G003	Ipin	Input Current into any pin		–2	2	mA
G004	Vesd1	ESD Susceptibility	HBM, 100 pF discharged through 1.5 kΩ		4	kV
G005	Tj	Junction Temperature		–40	150	°C
G006	Ts	Storage Temperature		–40	150	°C

THERMAL DATA

Item No.	Symbol	Parameter	Conditions	Min.	Typ.	Max.	Unit
T01	Ta	Operating Ambient Temperature Range		–40		125	°C
T02	Rthja	Thermal Resistance Chip to Ambient	QFN32 surface mounted to PCB according to JEDEC 51		40		K/W

All voltages are referenced to pin AVSS unless otherwise stated.
All currents flowing into the device pins are positive; all currents flowing out of the device pins are negative.

ELECTRICAL CHARACTERISTICS

Operating conditions: AVDD = DVDD = IOVDD = 3.1...3.6 V, Tj = –40...+125 °C, reference point AVSS unless otherwise stated

Item No.	Symbol	Parameter	Conditions	Min.	Typ.	Max.	Unit
Total Device							
001	VDD	Permissible Supply Voltage AVDD, DVDD, IOVDD		3.1		3.6	V
002	I_{AVDD}	Supply Current into AVDD	AVDD, DVDD, IOVDD = 3.3 V, f_{in} = 1 kHz, inter = x256, ABZ and UVW outputs active			15	mA
003	I_{DVDD}	Supply Current into DVDD	AVDD, DVDD, IOVDD = 3.3 V, f_{in} = 1 kHz, inter = x256, ABZ and UVW outputs active			22	mA
004	I_{IOVDD}	Supply Current into IOVDD	RS422 drivers enabled (MAIN_CFG.rs422 = 1); quadrature outputs terminated with 120 Ω quadrature outputs open			 85 2.5	 mA mA
Signal Inputs and Amplifiers: SIN+, SIN–, COS+, COS–							
101	Vin()	Permissible Input Voltage	Refer to Figure 1 Low Input Range (MAIN_CFG.input = 0 or 1) High Input Range (MAIN_CFG.input = 2)	 0.35 0		 AVDD - 1.1 AVDD	 V V
102	Ain()diff	Permissible Differential Input Amplitude, Max(SIN+ – SIN–) or Max(COS+ – COS–)	Refer to Figure 1 Low Input Range (MAIN_CFG.input = 0 or 1) High Input Range (MAIN_CFG.input = 2)	 20 65		 700 2000	 mV mV
103	Vcm()	Permissible Input Common Mode Range, (SIN+ + SIN–)/2 or (COS+ + COS–)/2	Refer to Figures 1 and 2 Minimum gain Maximum gain	 0.7 0.35		 AVDD - 1.45 AVDD - 1.1	 V V
104	fin()	Permissible Input Frequency				700	kHz
105	Vos()	Amplifier Input Offset Voltage				±20	mV
106	Ilk()	Input Leakage Current				±50	nA
108	OFFcorr	Correctable Input Offset Voltage	As percentage of input signal amplitude; input offset voltage is the sum of sensor offset plus amplifier offset (item 105); (step size: 3.9 mV / gain)	±100			%
109	Acorr	Correctable Balance (Amplitude) Mismatch	Max(Asin, Acos) / Min(Asin, Acos), where Asin and Acos are the SIN/COS input amplitudes respectively. (step size 0.2 %)	±25			%
110	PHIcorr	Correctable Phase Error	(step size 0.22°)		±26		°
111	Rin()diff	Differential Input Resistance	Low Input Range (MAIN_CFG.input = 0) Low with Loss Detect. (MAIN_CFG.input = 1) High Input Range (MAIN_CFG.input = 2)	10	1000 0.240 0.670		MΩ MΩ MΩ
Zero Signal Inputs and Amplifier: ZERO+, ZERO-							
201	Vin()	Permissible Input Voltage		0		AVDD	V
202	Vcm()	Permissible Input Common Mode Voltage	Refer to Figures 1 and 2 Minimum gain Maximum gain	 0.7 0.35		 AVDD - 1.45 AVDD - 1.1	 V V
203	Vos()	Input Referenced Offset Voltage				±20	mV
204	Ilk()	Input Leakage Current				±50	nA
205	OFFcorr	Correctable Input Offset Voltage	As percentage of input signal amplitude; input offset voltage is the sum of sensor offset plus amplifier offset (item 105)	±100			%
206	Rin()diff	Differential Input Resistance		10	1000		MΩ
Converter Performance							
303	INL	Integral Nonlinearity	Refer to Figure 4, 1 Vpp-diff SIN/COS input with compensated offset, gain and phase			0.7	°
304	DNL	Differential Nonlinearity	Refer to Figure 4, 1 Vpp-diff SIN/COS input with compensated offset, gain and phase			0.35	°

ELECTRICAL CHARACTERISTICS

Operating conditions: AVDD = DVDD = IOVDD = 3.1...3.6 V, Tj = –40...+125 °C, reference point AVSS unless otherwise stated

Item No.	Symbol	Parameter	Conditions	Min.	Typ.	Max.	Unit
Internal Oscillator							
401	fosc	Oscillator Frequency	Tj = 27 °C; AVDD, DVDD = 3.1 V AVDD, DVDD = 3.6 V	48 49		51 52	MHz MHz
402	TCf	Temperature Coefficient			225		ppm/K
Internal EEPROM							
501	Nwrite	Permissible Number of Write Cycles	Tj = –40...+85 °C; chip release W, W1 [7] chip release V1	50 1000			
502	Tjw	Write Temperature Range		-40		85	°C
503	Tjr	Read Temperature Range		-40		125	°C
504	DRTraw	Raw Data Retension Time		10			years
Reset and Start-Up: xRST							
601	DVDDon	DVDD Power-On Threshold	Increasing voltage at DVDD; xRST tied to DVDD	2.5	2.7	3.0	V
602	DVDDoff	DVDD Undervoltage Reset Threshold	Decreasing voltage at DVDD; xRST tied to DVDD	2.2		2.4	V
603	tstart	Startup Time	Valid EEPROM configuration, until AB output function; START.wait = 0 START.wait = 3 (factory default)		 4.5 14.5		 ms ms
Digital Input Pins: xRST, xCALIB, A+/– (Encoder Link active), SI, SCLK, xSS, PINCFG							
701	Vt()hi	Input Logic Threshold High	DVDD = 3.3 V			1.9	V
702	Vt()lo	Input Logic Threshold Low	DVDD = 3.3 V	0.8			V
703	Ilk()	Input Leakage Current at SI, SCLK, xSS				±50	nA
704	f(SCLK)	Permissible Clock Frequency at SCLK	PINCFG connected to DVSS			20	MHz
Digital Output Pins: xIRQ, SO, A+/A–, B+/B–, Z+/Z– (CMOS drivers enabled: MAIN_CFG.rs422 = 0)							
801	I()max	Permissible Output Current	Per pin, indefinite			±10	mA
803	Vs()hi	Saturation Voltage High	Vs()hi = IOVDD - V(); I() = –4 mA, MAIN_CFG.irqpp = 1 (for xIRQ push-pull)			0.7	V
804	Vs()lo	Saturation Voltage Low	I() = 4 mA			0.7	V
805	Isc()hi	Short-Circuit Current High	Any pin shorted to DVSS	–30	–16		mA
806	Isc()lo	Short-Circuit Current Low	Any pin shorted to DVDD		16	30	mA
807	tr()	Rise Time	DVDD = 3.3 V, CL = 50 pF, 10% → 90% VDD			20	ns
808	tf()	Fall Time	DVDD = 3.3 V, CL = 50 pF, 90% → 10% VDD			20	ns
RS422 Drivers: A+/A–, B+/B–, Z+/Z– (RS422 drivers enabled: MAIN_CFG.rs422 = 1)							
901	Idrv()	Nominal RS422 Driver Current	RL() = 120 Ω between + and – terminals	18		27	mA
902	Isc()hi	Short-Circuit Current High [8]	+ or – pin shorted to IOVSS	–55			mA
903	Isc()lo	Short-Circuit Current Low [8]	+ or – pin shorted to IOVDD			35	mA
904	t_{AB}	Output Phase A vs. B	Refer to Figure 3		25		%
905	t_{whi}	Duty Cycle at Output A, B	Refer to Figure 3		50		%
906	AArel	Relative Angle Accuracy	f_{in} = 1 kHz, ideal waveform, FILT_K.kp = 6, referenced to output period T (see Figure 3); INTER < x50 INTER = x50...x100 INTER > x100			 ±5 ±10 ±15	 % % %
907	t_{MTD}	Time Between AB Edges (Minimum Transition Distance)	Refer to Figure 3; ABLIMIT = 0x00		1/fosc		
LED Output (enabled: LED_CFG = 1)							
A01	I()max	Permissible Output Current	for continuous operation			±15	mA
A02	Vout()hi	Output Voltage High	VDD = 3.3 V, Tj = 27 °C, I() = –10 mA		2.7		V
A03	Vs()hi	Saturation Voltage High	Vs()hi = DVDD – V(LED); I() = –10 mA			1	V

[7] Regarding chip release W1, please refer to the design review on page 77.
[8] Regarding chip release W, please refer to the design review on page 77.

ELECTRICAL CHARACTERISTICS

Operating conditions: AVDD = DVDD = IOVDD = 3.1...3.6 V, Tj = –40...+125 °C, reference point AVSS unless otherwise stated

Item No.	Symbol	Parameter	Conditions	Min.	Typ.	Max.	Unit
A04	Vs()lo	Saturation Voltage Low	I() = 10 mA			1	V
A05	Isc()hi	Short-Circuit Current High	short to GND	-40			mA
A06	Isc()lo	Short-Circuit Current Low	short to VDD			40	mA
Bias Outputs: VC, VREF							
B01	VC	Bias Voltage VC	I(VC) = 0		50		%AVDD
B02	dVREF	ADC Reference Voltage VREF versus VC	dVREF = V(VREF) – V(VC); I(VREF) = 0	–1.1	–1	–0.9	V

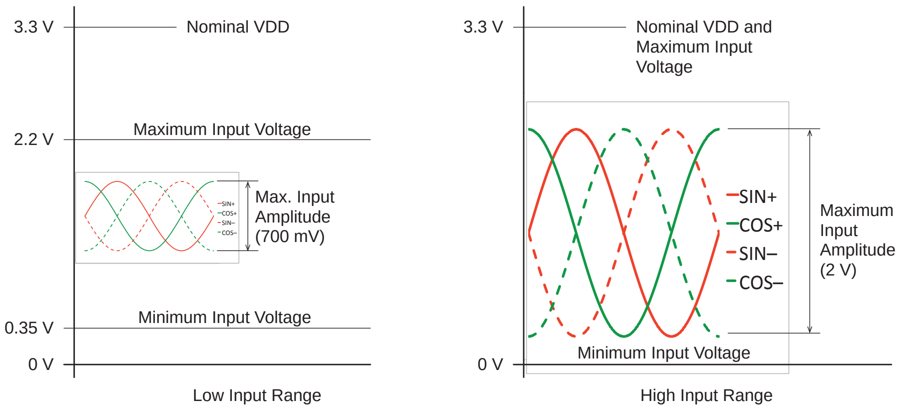


Figure 1: Differential Input Amplitude Ain()diff: Max(SIN+ – SIN–) or Max(COS+ – COS–)

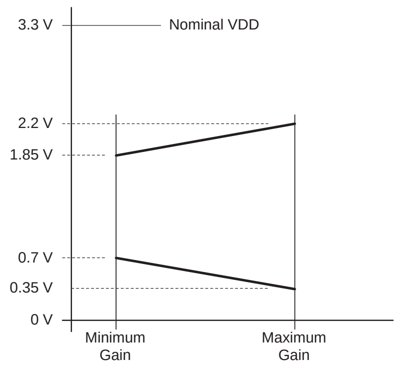


Figure 2: Permissible Input Common Mode Vcm: (SIN+ + SIN–)/2 or (COS+ + COS–)/2

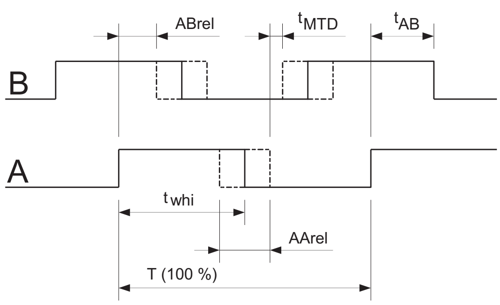


Figure 3: Description of AB Output Signals

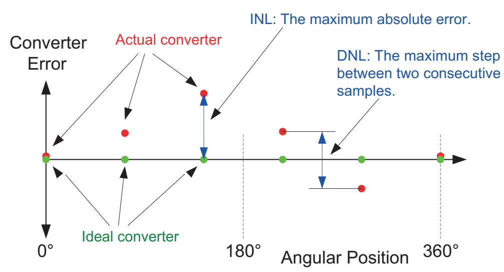


Figure 4: Definition of Integral and Differential Nonlinearity

OPERATING REQUIREMENTS: SPI Interface

Operating conditions: AVDD = DVDD = IOVDD = +3.1...+3.6 V, AVSS = DVSS = IOVSS = 0 V, Tj = –40...125 °C

Item No.	Symbol	Parameter	Conditions	Min.	Max.	Unit
SPI Interface Timing						
I001	t_{C1}	Permissible Clock Cycle Time	see Elec. Char. No.: 704	50		ns
I002	t_{D1}	Clock Signal Lo Level Duration		15		ns
I003	t_{D2}	Clock Signal Hi Level Duration		15		ns
I004	t_{S1}	Setup Time: xSS lo before SCLK lo → hi		80		ns
I005	t_{H1}	Hold Time: xSS lo after SCLK hi → lo		50		ns
I006	t_{W1}	Wait Time: between xSS lo → hi and xSS hi → lo	 with ADC Read as preceding command	200 600		ns ns
I007	t_{S2}	Setup Time: SI stable before SCLK lo → hi		5		ns
I008	t_{H2}	Hold Time: SI stable after SCLK lo → hi		10		ns
I009	t_{P1}	Propagation Delay: SO stable after xSS hi → lo			60	ns
I010	t_{P2}	Propagation Delay: SO high impedance after xSS lo → hi			25	ns
I011	t_{P3}	Propagation Delay: SO stable after SCLK hi → lo			20	ns

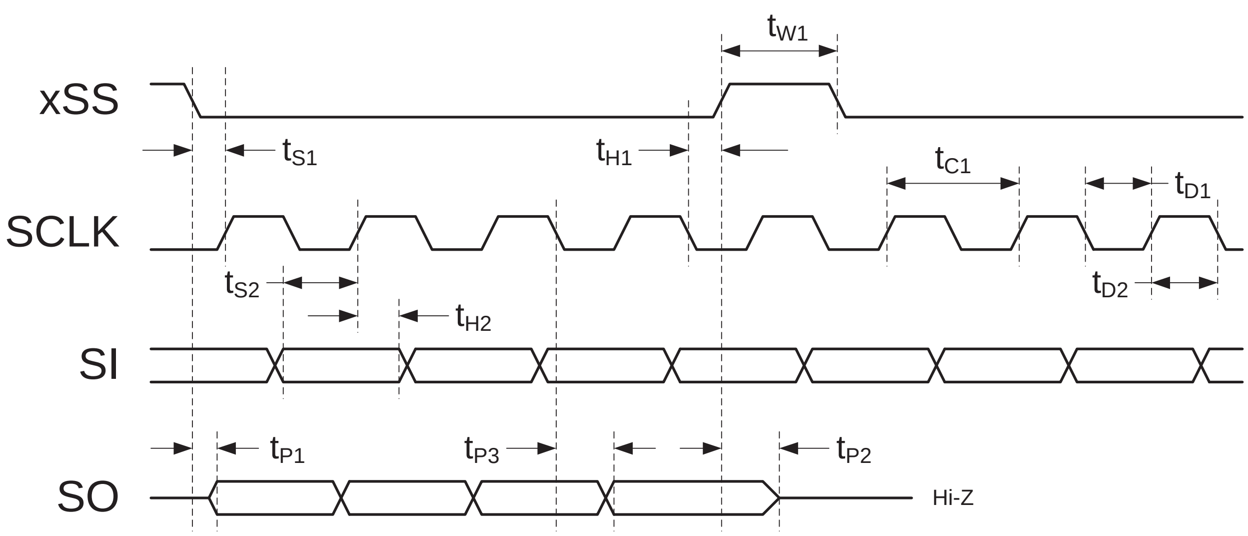


Figure 5: SPI Timing

OPERATING REQUIREMENTS: Encoder Link Interface

Operating Conditions: AVDD = DVDD = IOVDD = +3.1...+3.6 V, AVSS = DVSS = IOVSS = PINCFG = 0 V, Tj = –40...85 °C

Item No.	Symbol	Parameter	Conditions	Min.	Max.	Unit
Encoder Link Activation Sequence Timing						
I101	t1	Activation Sequence Interval 1	A+ > 2.4 V, A– > 2.4 V	0.25	2	ms
I102	t2	Activation Sequence Interval 2	A+ < 0.8 V, A– > 2.4 V	t1 – 10%	t1 + 10%	ms
I103	t3	Activation Sequence Interval 3	A+ < 0.8 V, A– < 0.8 V	t1 – 10%	t1 + 10%	ms
I104	t4	Activation Sequence Interval 4	A+ > 2.4 V, A– < 0.8 V	t1 – 10%	t1 + 10%	ms
Encoder Link Interface Timing (after activation)						
I105	fclk(A+)	ELink Clock Frequency	Signal driven into A+ (operation at > 85 °C requires 500 kHz max.)		1000	kHz
I106	t_{D1}(A+)	ELink Clock Signal Hi Level Duration	Signal driven into A+	200		ns
I107	t_{D2}(A+)	ELink Clock Signal Lo Level Duration	Signal driven into A+	200		ns
I108	t_S(A–)	ELink Input Setup Time	Signal driven into A–	200		ns
I109	t_H(A–)	ELink Input Hold Time	Signal driven into A–	200		ns
I110	t_P(A–)	ELink Output Propagation Delay	Signal driven out on A–		200	ns

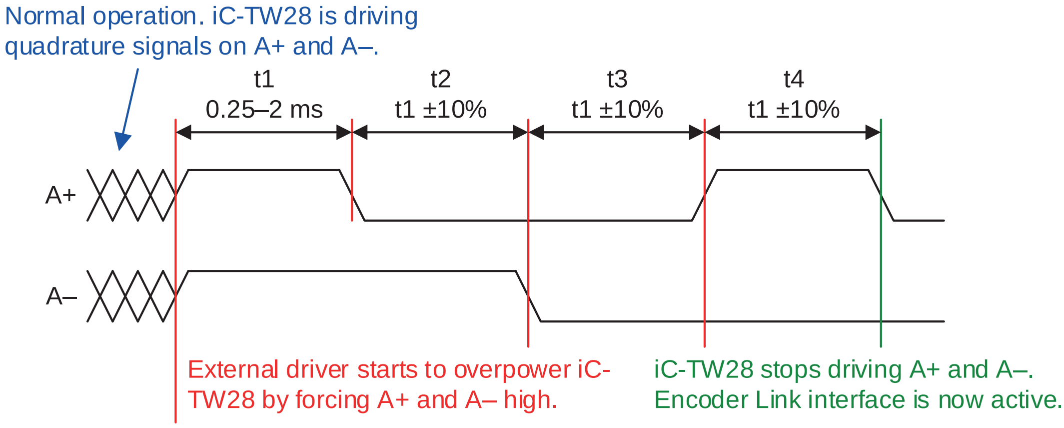


Figure 6: Encoder Link Activation Sequence

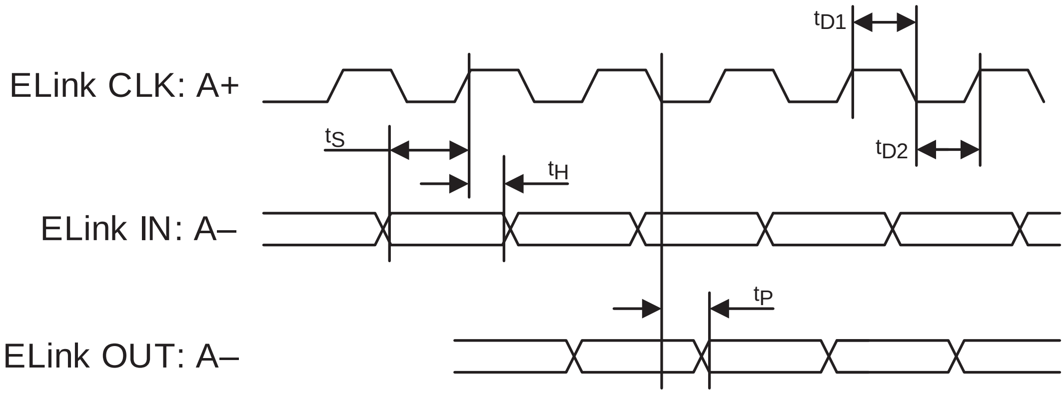


Figure 7: Encoder Link Read and Write Timing

FUNCTIONAL OVERVIEW

The iC-TW28 is a general-purpose 10-bit Sine/Cosine interpolator with sophisticated automatic calibration functions for the ABZ and UVW signal paths, and a built-in RS422-compatible line driver. It accepts differential analog sin/cos input signals from magnetic or optical sensors and calculates (interpolates) the angular position with the sin/cos cycle, as shown in Figure 8. Typical output is industry-standard incremental AB quadrature at programmable resolution and/or UVW commutation signals. Auto calibration means that no complicated signal analysis or calibration procedure is required during product design or production. Auto adaption monitors and adapts signal path error correction values during operation to maintain optimal performance with low error and jitter. An internal EEPROM to store configuration and calibration data and an accurate internal oscillator are included.

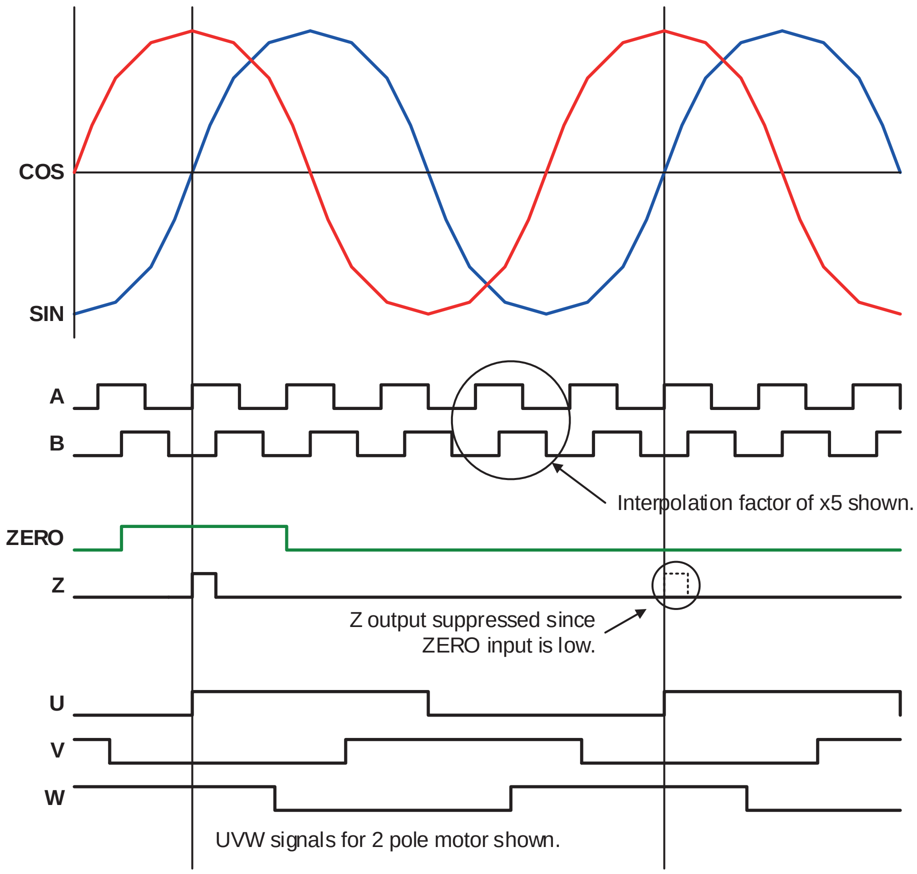


Figure 8: iC-TW28 Functional Overview

Configuration of the iC-TW28 is via dedicated pins, Encoder Link (using the A+ and A– pins), or the SPI interface. In pin-configuration mode, a limited set of the most common operating conditions can be selected for simple stand-alone applications. With SPI or Encoder Link (serial) configuration, complete flexibility and access to all iC-TW28 features is available for more sophisticated applications.

When paired with a local CPU or microcontroller, the iC-TW28's multi-cycle position data and internal status conditions can be monitored. Multiple iC-TW28's can be accessed from a single host CPU or microcontroller for cost-effective synchronized multi-axis applications.

A configurable signal path filter provides dynamic response characteristics, allowing smooth low-jitter output as well as fast response to changing operating conditions. An integrated LED intensity control allows maintaining signal amplitudes of optical sensors in the presence of LED ageing and temperature effects.

FUNCTIONAL BLOCK DIAGRAM

A functional block diagram of the iC-TW28 is shown in Figure 9 and explained on the following pages.

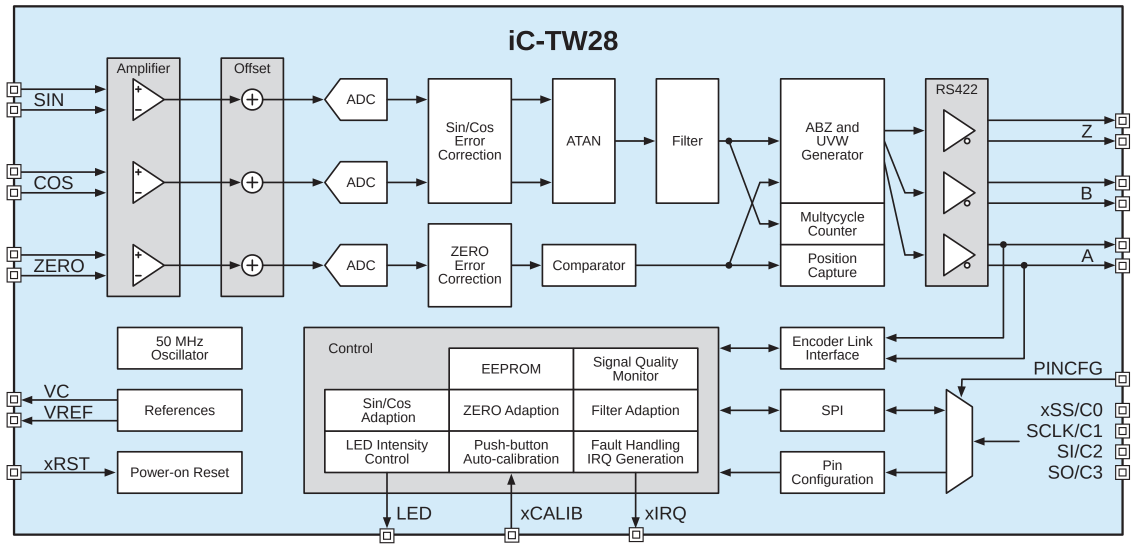


Figure 9: Functional Block Diagram

Reference

The Reference provides reference voltages for the internal analog circuits. These outputs must be capacitively bypassed for proper operation of the iC-TW28.

Oscillator

The Oscillator provides a high-accuracy 50 MHz clock that controls all timing within the iC-TW28.

Power-On Reset

The Power-On Reset (POR) circuit provides orderly startup of the iC-TW28 when power is applied. An external reset source can also be connected to the POR to allow independent control of device startup.

SPI Port/Configuration Pins

When the iC-TW28 is in serial configuration mode, the SPI port is available for use by an external host processor or microcontroller for initial calibration or general communication. In pin configuration mode, the SPI port is disabled and the SPI pins are used to directly configure the iC-TW28.

Encoder Link Interface

In serial configuration mode, the Encoder Link interface provides read/write access to the iC-TW28's internal registers using the A+ and A- outputs. This is useful for field reconfiguration or diagnostics of products incorporating the iC-TW28. Encoder Link can only be used for configuration and diagnostics, it cannot be used to read the multi-cycle counter or captured position values. Encoder Link can be disabled to eliminate tampering with finished products.

Input Stage

Three programmable gain amplifiers are used to amplify the sin, cos, and zero inputs. The sin/cos PGAs also provide configurable 9 dB attenuation to allow high amplitude (rail-to-rail) sensor inputs to be used (from GMR or TMR sensors, for example). A noise filter follows each PGA to reduce high-frequency noise coming from the sin/cos inputs. In Serial configuration mode, two different settings are available for the sin/cos noise filters.

Analog Error Correction

Coarse analog offset and gain correction (6 – 40.5 dB in 1.5 dB steps) is provided to scale and adjust the sin/cos signals prior to A/D conversion. These correction values are determined using auto calibration during initial system calibration.

Analog-to-Digital Converters (ADCs)

10-bit ADCs convert the conditioned analog sin and cos signals into digital values for further processing. An 8-bit ADC is used for the zero signal. The remainder of the signal path is completely digital.

Digital Error Correction
After digital conversion, the sin/cos signals are processed to remove any remaining offset, equalize signal amplitude, and ensure exact 90° shift. Correcting these signal errors increases interpolation accuracy for lowest error and jitter. Initial error correction values are determined automatically by auto calibration during initial system calibration. During operation, these correction values are monitored and automatically adjusted (auto-adaption) to provide high accuracy signals under changing application conditions.

Angle Calculation (Arctan)
The angle within an input cycle indicated by the conditioned digital sin/cos signals is next calculated as arctan(Sin/Cos) with a resolution of 10 bits using a CORDIC algorithm.

Filter
A configurable filter is provided in the signal path to reduce noise and jitter in the AB and UVW outputs. In pin configuration mode, three filter settings (light, medium, and heavy) are available. In Serial configuration mode, additional control over the signal path filter's characteristics is available.

Hysteresis
Hysteresis is available in the AB and UVW signal paths to reduce output dithering (instability) at standstill at the expense of position error on direction reversal. In pin configuration mode, four hysteresis settings are available; Serial configuration mode expands this to 32 settings.

Interpolation Factor
The 10-bit sensor input angle value is scaled to the resolution required by the desired interpolation factor. In pin configuration mode, 24 interpolation factors (12 each binary and decimal) are available. In Serial configuration mode, interpolation factors between 2 and 256 may be selected.

Z Signal Path
The Z signal path is similar to the sin/cos signal path. Analog offset and gain correction are provided to condition the zero signal. Correction values are determined using auto calibration during initial system calibration. An 8-bit ADC and configurable comparator are used to generate the Z gating window. The Z gating window allows producing one and only one Z pulse per revolution in applications where there are multiple input cycles per revolution.

ABZ Generator
The AB generator synthesizes quadrature AB outputs at the interpolated resolution. The output generator also uses the conditioned Z gating window signal from the Z signal path to generate a programmable-width Z output synchronized with the AB outputs.

Automatic phase correction to center the Z output pulse location within the ZERO input gating window is provided. This Z phase correction value is determined automatically using auto calibration during initial system calibration.

Post-AB Divider
An optional programmable divider after the ABZ generator is available. This provides fractional or non-integer AB output resolutions and interpolation factors for special applications.

UVW Signal Path
In addition to the ABZ outputs, the iC-TW28 can generate 3-phase UVW outputs for commutating brushless motors with up to 64 poles (32 pole pairs). Programmable phase correction to properly align the UVW signals with the motor and hysteresis to reduce signal noise (jitter) at standstill are provided.

ABZ/UVW Outputs
The iC-TW28 contains a built-in RS422 compatible differential line driver for driving 120 Ω terminated cables. In Serial configuration mode, the line driver can be bypassed to save power in imbedded or short signal-run applications. Differential ABZ or UVW outputs or simultaneous single-ended ABZ and UVW outputs are available.

Auto Calibration
Auto calibration is used at initial calibration to automatically determine initial offset, gain, and phase compensation values for the sin, cos, and zero channels. Activating the xCALIB input (pin 15) or sending a serial command initiates auto calibration; deactivating xCALIB or a serial command stores the calibrated values to the internal EEPROM.

In pin configuration mode, it is possible to eliminate calibration entirely by permanently grounding xCALIB. This permanently enables auto calibration, and the iC-TW28 will calibrate itself at every startup. In this way, no initial calibration is required, greatly simplifying product manufacturing and deployment.

Auto Adaption
Auto adaption maintains optimal offset, channel balance, and phase compensation values for the sin and cos channels during operation to ensure maximum interpolator accuracy under all operating conditions. In serial configuration mode, auto adaption can be disabled.

Fault Handling
The iC-TW28 provides comprehensive fault handling features and a fault output to notify external systems of faults and warnings during operation. The fault output is the active-low interrupt request output (xIRQ) pin (pin 16). In stand-alone applications, xIRQ can be used to directly drive a fault LED. When the iC-TW28 is used with a host processor or microcontroller, xIRQ is typically used to interrupt the host when a fault occurs. In Serial configuration mode, real-time status and fault information is available over the SPI port.

In pin configuration mode, xIRQ is activated if there is an internal fault in the signal path or if the sensor input amplitude becomes less than 60% or more than 120% of its calibrated value. In Serial configuration mode, additional status conditions (signal path saturation/overflow, excessive input frequency, excessive AB output frequency, excessive adaption, and excessive position lag) can be monitored and programmed to activate xIRQ.

AB Output Frequency Limiter
The iC-TW28 incorporates a programmable AB output frequency limiter that guarantees a minimum separation time between AB edges. This is useful to avoid counting errors with PLCs or counters with input frequency limits less than the 12.5 MHz maximum output frequency of the iC-TW28. Six output frequency limit choices are available in pin configuration mode; Serial configuration mode provides 256 choices.

When AB output frequency is being limited, the AB outputs lag behind the sin/cos inputs. If this condition is temporary or transient, the AB outputs catch up when the limiter is no longer active. If this condition persists, however, a fatal fault is generated and the iC-TW28 stops working. In Serial configuration mode, the AB output frequency limiter can be programmed to activate xIRQ.

Amplitude Monitor
The iC-TW28 continuously monitors the amplitude of the sin/cos input signals by calculating the quantity $\sqrt{sin^2 + cos^2}$. In pin configuration mode, this value is used to activate xIRQ if the input signal amplitude becomes less than 60% or more than 120% of its calibrated value. In serial configuration mode, other amplitude limits can be set.

LED Intensity Control (Serial Only)
In serial configuration mode, the calculated sin/cos amplitude value can also be used to drive the LED output (pin 26) to control the intensity of an optical sensor LED. This maintains the sin/cos signals at their calibrated amplitude in the presence of LED ageing and varying application conditions.

Multi-Cycle Counter (Serial Only)
A 14-bit multi-cycle counter is available in the iC-TW28 to track up to 16,383 input cycles during operation. In serial configuration mode, the multi-cycle counter can be read and written using commands and can be configured to reset on the rising edge of the Z output. The multi-cycle counter value can only be read using the SPI port.

Should input cycle counting beyond the range of the built-in multi-cycle counter be required, the iC-TW28 can be configured to generate an interrupt (activate xIRQ) when there is an imminent overflow of the multi-cycle counter. In this way, a host processor or microcontroller can extend the counter to any arbitrary length.

Position Capture (Serial Only)
The full 24-bit position value (10 bits of interpolated angle within an input cycle plus 14 bits of multi-cycle count) of the iC-TW28 can be captured and read out over the SPI port. This capture can be configured to take place on the rising edge of the Z output or the ZERO input gating window and can also be configured to generate an interrupt. This allows touch-probe or distance-coded index applications to be easily implemented using the iC-TW28.

Filter Adaption (Serial Only)
In serial configuration mode, the signal path filter can be configured to dynamically adjust its bandwidth based on sensor input acceleration. This allows heavy filtering to be used to provide a smooth output at constant speed while still maintaining fast response to changes in input conditions.

Residual Error Calculation (Serial Only)
The iC-TW28 continuously calculates the residual offset, balance, and phase error of the corrected sin/cos signals. These residues represent the uncorrected signal error in the sin and cos channels, and are typically zero when auto adaption is used. In applications where auto adaption cannot be used, these residual values allow sensor signal quality to be monitored in a host processor or microcontroller. Programmable threshold values allow activating xIRQ should any of the residual values become excessive.

EEPROM
The iC-TW28 provides an internal EEPROM to store configuration and initial calibration data for use at startup. In addition to a standard checksum on the EEPROM data, sophisticated data encoding allows detecting and correcting single-bit errors and detecting two-bit errors for enhanced application protection. The EEPROM is usually locked (write protected), but can be unlocked via serial command.

iC-TW28 identification data and serial number are also stored in the EEPROM. Additionally, four bytes of data are available for user information (product ID, serial number, etc.) in the EEPROM.

ELECTRICAL CONNECTIONS

The basic electrical connections for a typical stand-alone application using the iC-TW28 in pin configuration mode are shown in Figure 10. Other than the sin/cos sensor, only a few bypass capacitors and other components are required for operation.

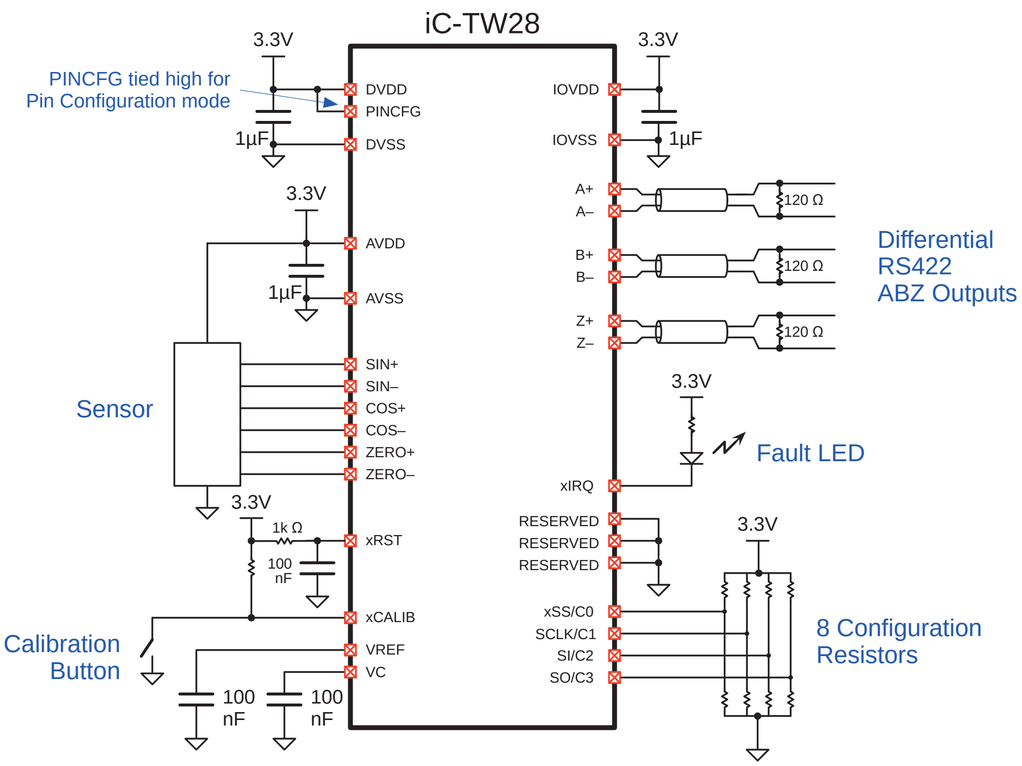


Figure 10: Typical Electrical Connections For Stand-Alone Pin Configuration Application

The basic electrical connections for a typical stand-alone application using the iC-TW28 in serial configuration mode using the SPI port are shown in Figure 11. Other than the sin/cos sensor, only a few bypass capacitors and other components are required for operation.

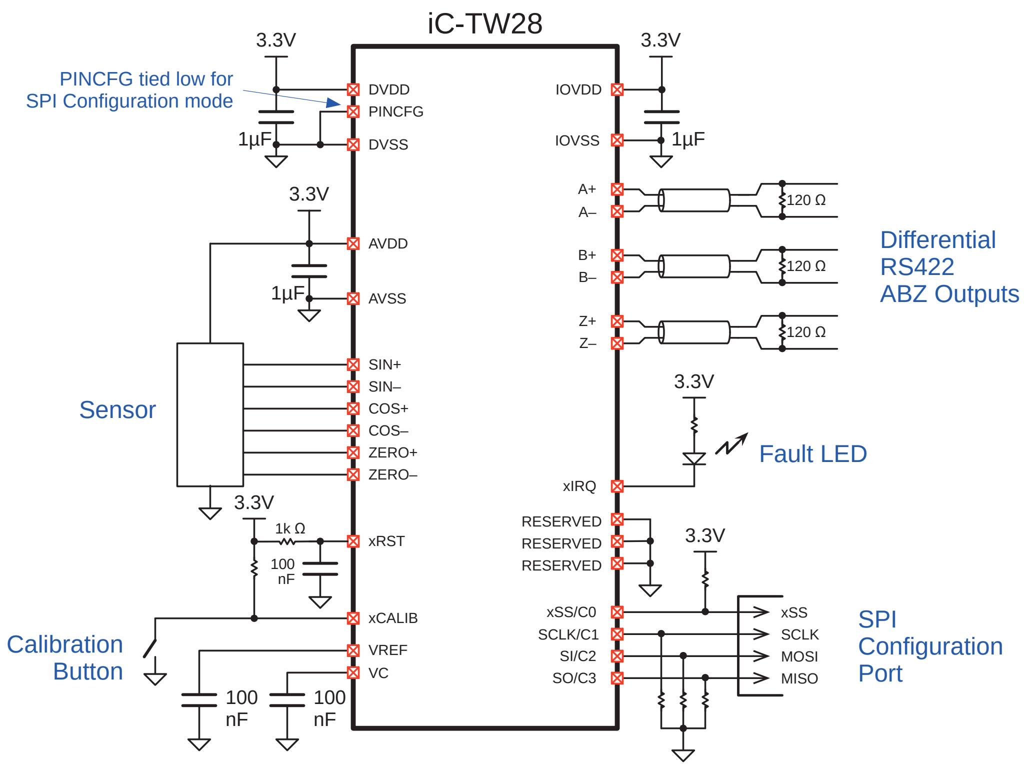


Figure 11: Typical Electrical Connections For Stand-Alone SPI Configuration Application

The basic electrical connections for a hosted application using a single iC-TW28 are shown in Figure 12. Multiple iC-TW28s can also be bussed or chained together using the same SPI port on the host processor or microcontroller. See Bussing Multiple iC-TW28s on page 75 and Chaining Multiple iC-TW28s on page 76 for more information.

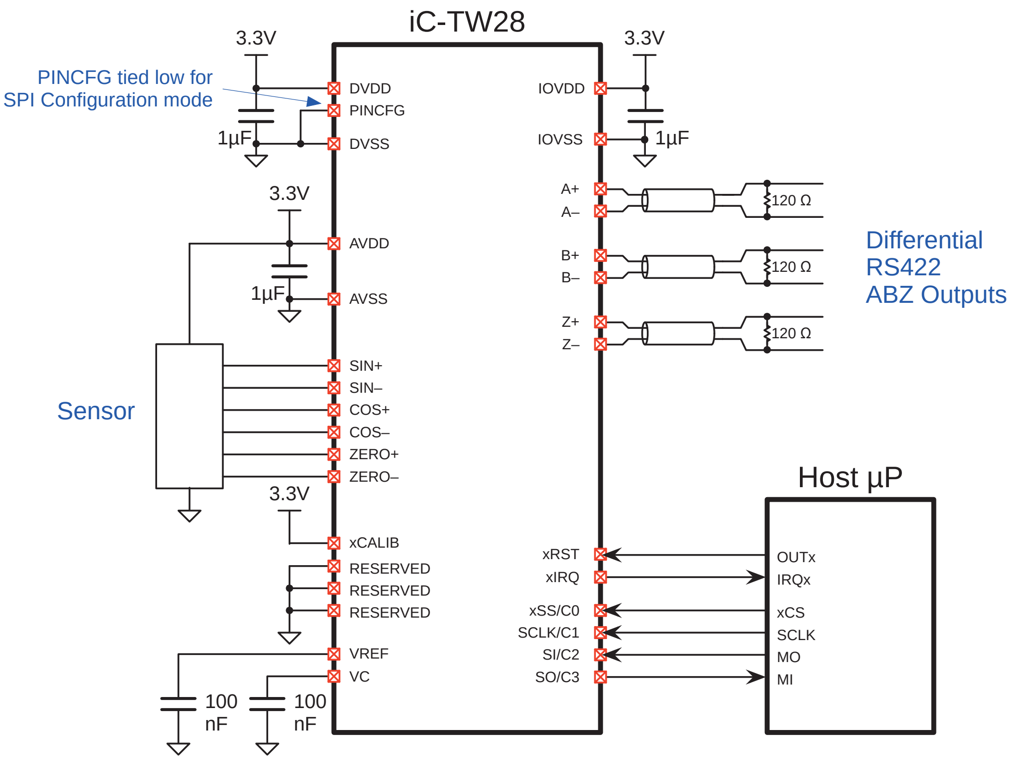


Figure 12: Typical Electrical Connections For Hosted Application

Power and Ground

The iC-TW28 requires a high quality ground and clean 3.3 V power supplies. There are three separate power/ground pin pairs, one each for the analog (AVDD/AVSS), digital (DVDD/DVSS), and I/O (IOVDD/IOVSS) circuitry.

In most cases, it is sufficient to connect all three power pins to the same low-impedance power source, preferably an on-board voltage regulator. Likewise, the three ground pins can usually be connected to the same solid ground plane on the PC board. If necessary, separate voltage regulators can be used to power each section to provide enhanced noise immunity. In all cases, each power pin should have a dedicated 1µF decoupling capacitor placed as close to the iC-TW28 as possible.

Reference Outputs

The reference outputs VREF and VC must each be decoupled to ground with separate 100 nF capacitors placed as close to the iC-TW28 as possible. VC should not be used to bias external circuitry or the sin/cos inputs with single-ended sensors.

xCALIB Input

The active-low xCALIB input is used to activate the auto-calibration feature of the iC-TW28. A push-button and pull-up resistor can be connected to this input as shown for easy manual calibration. xCALIB can also be controlled by a host processor or microcontroller output, if desired.

In pin configuration mode, xCALIB can be permanently tied to ground to eliminate calibration entirely if desired. This permanently enables auto-calibration, and the iC-TW28 will calibrate itself automatically at every start-up.

If push-button calibration is not required, xCALIB should be connected to 3.3 V to avoid spurious calibration.

Note: When switching on the power supply, the voltage at xCALIB must increase with the supply at VDD and must not be kept low by a capacitor.

SIN and COS Inputs

The iC-TW28 connects directly to magnetic (such as iC-SM2L or iC-SM5L) and optical (such as iC-LSHB or iC-PT...H series) sensors providing differential sin/cos outputs, as shown in Figure 13 and Figure 14.

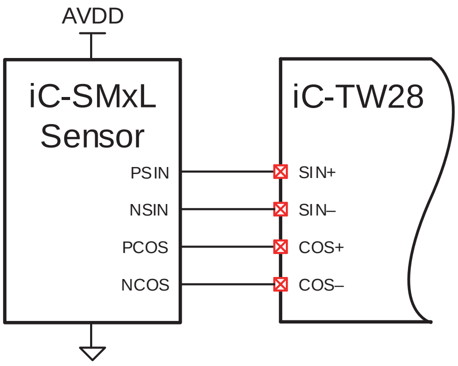


Figure 13: Magnetic Sensor Connection

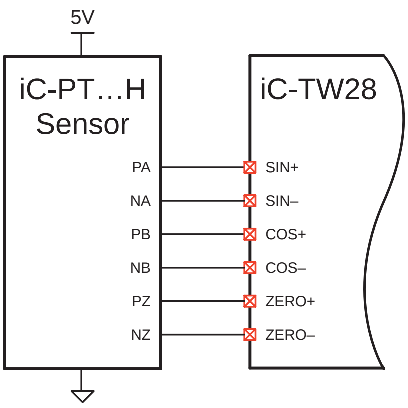


Figure 14: Optical Sensor Connection

Nominal differential signal amplitudes between 20 mV and 2.0 V in two ranges can be accommodated. See Input Configuration and Signal Levels on page 55 for more information.

Single-ended sensors can be connected to the SIN+ and COS+ inputs. In this case, the SIN– and COS– inputs should be connected to a resistive voltage divider to bias them to the zero level of the sensor output (typically half the sensor supply). Do not allow any of the SIN or COS inputs to float.

ZERO Inputs

The iC-TW28 can interface to a wide range of differential or single-ended index or zero sensors to provide a Z output which is synchronized with the AB outputs. Optical sensors usually provide differential zero or index signals along with the sin/cos signals. In magnetic systems, a separate zero sensor is usually required.

Digital zero sensors (Hall, MR, and others) typically provide a single-ended active-low signal via an open-drain output that pulls low in the presence of a magnetic field. Connect active-low (open drain) digital index sensors to the iC-TW28 ZERO– input and connect the ZERO+ input to VC as shown in Figure 15.

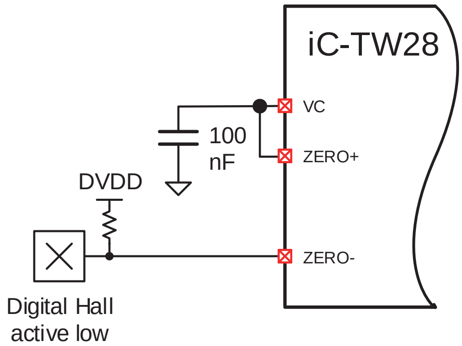


Figure 15: Digital Index Sensor Connection

For active-high (open source) digital index sensors, reverse the ZERO+ and ZERO– connections.

Analog-output zero sensors, such as MR bridges, can also be used with the iC-TW28 as shown in Figure 16.

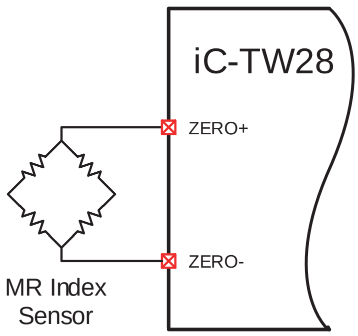


Figure 16: Analog Index Sensor Connection

To produce a Z output once every input cycle, connect ZERO+ to 3.3 V and ZERO– to ground. This is useful in on-axis applications where one input revolution produces only one input cycle.

If no Z output from the iC-TW28 is required, connect ZERO+ to ground and ZERO– to 3.3 V. Do not allow the ZERO+ or ZERO– inputs to float.

ABZ Outputs
The iC-TW28 provides differential ABZ outputs capable of driving 20 mA into a terminated RS422 line. The A+, A–, B+, B–, Z+, and Z– outputs can be directly connected to the RS422 line as shown in Figure 17.

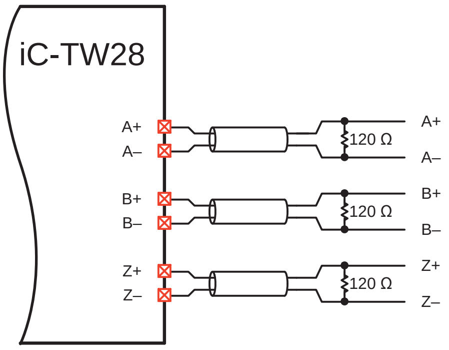


Figure 17: ABZ Output Connection

The three signal pairs should be terminated with a 120 Ω resistor at the far (receiving) end of the cable as shown. In serial configuration mode, the RS422-compatible line driver can be disabled to save power for local or short-run applications. In this case, termination resistors should not be used.

UVW Outputs
In serial configuration mode, the iC-TW28 can be configured to provide differential UVW outputs or simultaneous single-ended ABZ and UVW outputs. See Output Modes, Directions, and Polarities on page 56 for more information.

xRST Input
The iC-TW28 contains a built-in power-on-reset (POR) circuit that controls the safe start-up of the device. In most applications, no external components are required and xRST can be connected directly to 3.3 V.

Alternatively, an RC network with recommended values of 1 kΩ and 100 nF can be connected to the active-low xRST input as shown in Figure 18. This provides a 100 µs delay and guarantees proper start-up under all conditions. Larger resistances can be used to provide proportionally longer delays.

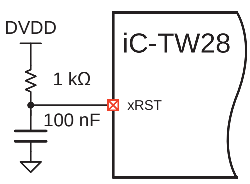


Figure 18: xRST Connection

In stand-alone applications, it is recommended to *always* provide for an RC network on the PC board and only populate the capacitor if required. Without the capacitor, the resistor provides the necessary pull-up. In applications using a host processor or microcontroller, the xRST input is best controlled by the host.

xIRQ Output
In stand-alone applications, xIRQ functions as an active-low fault output. It can be used to directly drive an LED with an appropriate current-limiting resistor for fault indication.

In hosted applications, xIRQ is connected to an interrupt request input on the host processor or microcontroller. In this way, when a warning or fault occurs, the host processor can query the iC-TW28 to determine what action to take. The xIRQ output can also be configured as an open-drain output allowing a wired-OR connection of multiple iC-TW28s to a single interrupt request input on the host processor. See Chaining Multiple iC-TW28s on page 76 for more information.

If the xIRQ output is not used, configure it for push-pull operation so that it does not float.

LED Output
In serial configuration mode, the iC-TW28 can be configured to provide LED intensity control. The LED output functions as a high-current output to drive the illumination LED used with an optical sensor. See LED Intensity Control on page 72 for more information.

PINCFG Input
The PINCFG input is used to select whether the iC-TW28 is in pin configuration or serial configuration mode. Connect PINCFG to 3.3 V to select pin configuration mode. Connect PINCFG to ground to select serial configuration mode. Do not allow the PINCFG input to float.

Configuration Resistors

In pin configuration mode (PINCFG = 3.3 V), the iC-TW28 is configured by applying different voltages to the configuration inputs C0 – C3. Each configuration input recognizes 12 different voltage levels. The desired voltage levels are typically set using a resistive voltage divider on each of the configuration inputs as shown in Figure 19. Thus, only 8 resistors are required to completely configure the iC-TW28.

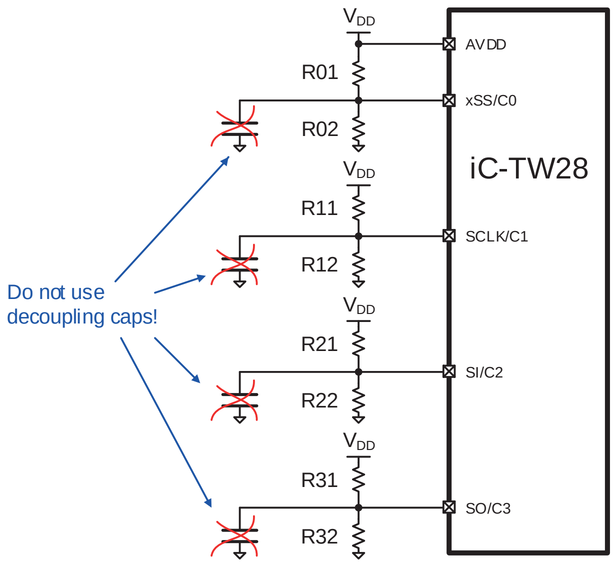


Figure 19: Pin Configuration Resistors

The resistors should be located as close to the configuration input pins as possible and no decoupling capacitors should be used. EIA E48 series 2% resistors or E96 series 1% resistors are recommended to guarantee reliable operation under all conditions.

SPI Port

The iC-TW28 provides a standard SPI (Serial Peripheral Interface) slave port that can be used for device configuration and communication with a host processor or microcontroller in serial configuration mode (PINCFG = 0 V). Connect the SPI port pins to the host processor or microcontroller as shown in Figure 20.

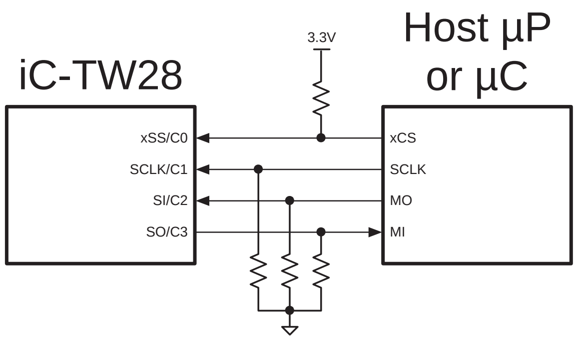


Figure 20: SPI Port Connection

If the host processor or microcontroller can be disconnected, the SPI port pins must be pulled up or down as shown in Figure 20. Do not allow any of the SPI port pins to float.

Although Figure 20 shows a single-device application, multiple iC-TW28's can also communicate over a single SPI channel to a host processor or microcontroller. See Bussing Multiple iC-TW28s on page 75 and Chaining Multiple iC-TW28s on page 76 for more information.

Reserved Pins

All reserved pins must be tied to ground as shown in Figures 10, 11, and 12 for proper operation. Do not allow any of the reserved pins to float.

CONFIGURATION OVERVIEW

The iC-TW28 can be configured for a specific application in one of three ways. For simple, stand-alone applications, a pin configuration interface is available. Pin configuration mode provides easy selection of the most common interpolation factors and operating modes.

For access to all iC-TW28 features in more sophisticated stand-alone or hosted applications, serial configuration mode must be used. Serial configuration can be done through the SPI port or the Encoder Link interface, which uses the A+ and A- outputs for communication. The Encoder Link interface can be used in hosted applications to provide factory configuration independent of the host processor or for field re-configuration.

Pin Configuration Mode

Pin configuration mode is recommended for stand-alone applications since it does not require any programming and is simple and easy to implement for fixed configurations in production.

Note: Encoder Link is not available in pin configuration mode.

To select pin configuration mode, connect the PINCFG pin to 3.3 V and connect eight resistors to the four configuration inputs as shown in Configuration Resistors on page 23.

The recommended resistor values for setting the configuration levels are shown in Table 1.

Configuration Level	Nominal Voltage	Rx1 kΩ	Rx2 kΩ
11	3.30	0.00	∞
10	3.00	10.0	100
9	2.70	11.5	51.1
8	2.40	11.5	30.1
7	2.10	16.2	28.7
6	1.80	9.53	11.5
5	1.50	11.5	9.53
4	1.20	28.7	16.2
3	0.90	30.1	11.5
2	0.60	51.1	11.5
1	0.30	100	10.0
0	0.00	∞	0.00

Table 1: Pin Configuration Resistor Values

EIA E48 series 2% resistors or E96 series 1% resistors are recommended to guarantee reliable operation under all conditions. An open circuit is shown as infinite (∞) resistance and a short circuit is shown as zero resistance in Table 1.

Interpolation Factor

Configuration inputs C0 and C3 are used to select the interpolation factor. Input C3 selects the desired interpolation group I0 or I1 and input C0 selects the desired interpolation factor from within the selected group. Choose the desired interpolation factor from Table 2 and connect the appropriate resistors to configuration input C0 according to Table 1 to set the corresponding configuration level.

C0 Level	Interpolation Group I0	Interpolation Group I1
11	256	250
10	128	200
9	64	180
8	32	125
7	16	100
6	12	80
5	8	50
4	6	40
3	4	25
2	3	20
1	2	10
0	1	5

Table 2: Configuration Input C0

Note: The interpolation factors shown in Table 2 are the number or AB output cycles per sin/cos input cycle. There are four times as many AB output edges per sin/cos input cycle than shown in the table.

Hysteresis and Filtering

Configuration input C1 selects the hysteresis and the amount of filtering (smoothing) of the AB outputs of the iC-TW28. Choose the desired hysteresis and amount of filtering from Table 3 and connect the appropriate resistors to configuration input C1 according to Table 1 to set the corresponding configuration level.

C1 Level	Hysteresis	AB Filtering
11	±2.8°	Heavy filtering
10	±1.4°	
9	±0.7°	
8	±0.35°	
7	±2.8°	Medium filtering
6	±1.4°	
5	±0.7°	
4	±0.35°	
3	±2.8°	Light filtering
2	±1.4°	
1	±0.7°	
0	±0.35°	

Table 3: Configuration Input C1

The hysteresis is shown in sin/cos input cycle degrees.

The amount of filtering is a compromise between fast response and smoothness of the AB outputs. It is recommended to start with light filtering since this gives the fastest response of the AB outputs to changes in the sin/cos inputs. Medium or heavy filtering may be selected if the outputs are noisy or jittery. Experimentation may be necessary to determine the optimal setting.

AB Frequency Limit and Auto Adaption
Configuration input C2 is used to set the AB frequency limit (minimum AB edge separation) and to enable or disable auto adaption. Choose the desired configuration from Table 4 and connect the appropriate resistors to configuration input C2 according to Table 1 to set the corresponding configuration level.

C2 Level	Max. AB Frequency	Min. Edge Separation	Auto Adaption
11	195 kHz	1.28 µs	On
10	390 kHz	640 ns	
9	781 kHz	320 ns	
8	1.56 MHz	160 ns	
7	3.12 MHz	80 ns	
6	6.25 MHz	40 ns	
5	195 kHz	1.28 µs	Off
4	390 kHz	640 ns	
3	781 kHz	320 ns	
2	1.56 MHz	160 ns	
1	3.12 MHz	80 ns	
0	6.25 MHz	40 ns	

Table 4: Configuration Input C2

The highest maximum AB output frequency in pin configuration mode is 6.25 MHz, which is equivalent to an edge separation of 40 ns. Lower maximum output frequencies (higher minimum edge separation) can be selected as shown in Table 4 if devices connected to the iC-TW28 (counters, PLCs, motion controllers, drives, etc.) cannot handle its full output frequency.

Auto adaption maintains optimal offset, channel balance, and phase compensation values for the sin and cos channels during operation to ensure maximum interpolator accuracy under all operating conditions. Unless specifically required otherwise, it is recommended to enable auto adaption and to use the highest maximum AB frequency (configuration level 6).

The sin/cos sensor input frequency *(finput)* that corresponds to a given AB output frequency can be calculated using the following formula:

$$finput = \frac{ABFrequency}{Interpolation}$$

Where *Interpolation* is the interpolation factor set using configuration inputs C0 and C3.

For example, if an interpolation factor of 250 is selected using C0 and C3, and C2 is at configuration level 8 (1.56 MHz), the maximum AB frequency will be reached at a sensor input frequency of

$$finput = \frac{1.56\,MHz}{250} = 0.00624\,MHz = 6.24\,kHz$$

If the sin/cos sensor input exceeds this frequency, the AB output position can no longer keep up with the sensor position. In this case, the iC-TW28 keeps generating output pulses at the maximum AB frequency. If this condition is temporary or transient, the AB outputs catch up when the sin/cos input frequency decreases. If this condition persists and the AB output position falls behind the sensor position by one half (180°) of a sin/cos input cycle, the AB outputs are no longer valid (unexpected direction reversal), a fatal fault is generated (STAT_VAL.lagfatl = 1), and the xIRQ output is activated.

Input Range, Interpolation Group, and Z Calibration
Configuration input C3 selects the input signal range, the interpolation group, and whether Z channel auto calibration is used. Choose the desired setting from Table 5 and connect the appropriate resistors to configuration input C3 according to Table 1 to set the corresponding configuration level.

C3 Level	Input Range	Interpolation Group	Auto Z Calibration
11	Reserved	I1	Yes
10	High		
9	Low		
8	Reserved	I0	
7	High		
6	Low		
5	Reserved	I1	No
4	High		
3	Low		
2	Reserved	I0	
1	High		
0	Low		

Table 5: Configuration Input C3

Select low input range for MR or Hall effect sensors producing differential outputs of up to ±700 mV peak. Select high input range for GMR, TMR, or optical sensors with outputs of up to ±2 V peak. See Input Configuration and Signal Levels on page 55 for more information. Do not select a reserved input range.

Select the interpolation group corresponding to the desired interpolation chosen using configuration input C0.

Auto Z calibration determines whether the zero channel is automatically calibrated along with the sin and cos channels when the xCALIB input is activated. It is recommended to use auto Z calibration in all applications using the ZERO inputs. In applications where the ZERO inputs are not used, select no auto Z calibration.

For example, to configure an interpolation factor of 80, normal input mode, and auto Z calibration, set C0 to configuration level 6 and C3 to configuration level 9.

Values for iC-TW28 registers in pin configuration mode are shown in Table 6 and 7. Pin-configured register values are not stored in EEPROM.

Register.bit(s)	Val.	Description
MAIN_CFG.input		Determined by C3
MAIN_CFG.filter	0	500 kHz max. input freq.
MAIN_CFG.rs422	1	RS422 ABZ outputs
MAIN_CFG.irqpp	0	Open-drain xIRQ output
MAIN_CFG.elinkoff	1	Encoder Link not available
LED_CFG	0	LED PWM disabled
UVW_CFG	0	UVW disabled
INTER		Determined by C0/C3
INTER1.div	0	No post-AB divider
AB.hyst		Determined by C1
UVW.hyst	0	0° UVW hysteresis
FALARM	0	No input freq.alarm
ABLIMIT		Determined by C2
ZERO0.threshold	10	42% of signal amplitude
ZERO0.mode	0	Position capture on Z
ZERO0.clr	0	Never clear mcc
ZERO1.zwidth	0	1AB edge (quad. state)
OUTPUT.apol	0	Normal polarity
OUTPUT.bpol	0	Normal polarity
OUTPUT.zpol	0	Z is active high
OUTPUT.abzdir	0	Normal AB direction
OUTPUT.uvwpol	0	Normal UVW polarity
OUTPUT.uvwdir	0	Normal UVW rotation
OUTPUT.abzen	1	ABZ outputs
OUTPUT.uvwen	0	No UVW outputs
ZPHASE	0	Affected by C3
UVWPH	0	No UVW phase shift
FILT_CFG.auto	0	Static kp
FILT_CFG.fb	0	No feedback delay
FILT_CFG.kpmax	6	Max. dynamic kp limit
FILT_LAG. threshold	12	Filter lag threshold
FILT_K.kp		Determined by C1
FILT_K.ki	3	Maximum ki
STAT_CFG.filter	3	2.5 ms status filter
STAT_CFG	1	IRQ extended by 40 ms
STAT_CFG.enc	0	No IRQ on mcc oflow
STAT_CFG.enz	0	No IRQ on pos. capture
STAT_SEL	0	STAT_VAL for IRQ
STAT_IE.oflow	0	No IRQ on overflow
STAT_IE.falarm	0	No IRQ on FALARM
STAT_IE.laglim	0	No IRQ on excess lag
STAT_IE.inclim	1	IRQ on AB/UVW limit
STAT_IE.lagfatl	1	IRQ on fatal lag

Table 6: Pin Configuration Mode Register Values

Register.bit(s)	Val.	Description
STAT_IE.scamp	1	IRQ on amplitude lo/hi
STAT_IE.adapt	0	No IRQ on excess adapt
STAT_IE.res	0	No IRQ on residuals
STAT_HIZ	0	Outputs always enabled
START.wait	3	10 ms startup wait time
START.mode	1	Same phase startup
START.nostart	0	Automatic startup
ADAPT_CFG0		Determined by C2
ADAPT_CFG1.p	1	Slow auto adaption rate
ADAPT_CFG1.stop	1	Stop auto adaption on SCAMP
ADAPT_CFG1.zcal		Determined by C3
ADAPT_CFG1.xcalee	1	Store to EEPROM when xCALIB de-activates
SC_AMP_TARG	150	Sin/cos amplitude target
SC_AMP_LOW	90	–40% SC_AMP_TARG
SC_AMP_HIGH	180	+20% SC_AMP_TARG
S_OFS_BASE	0	Not used
C_OFS_BASE	0	Not used
SC_OFS_LIM	127	Not used
SC_OFS_TH	0	Not used
SC_BAL_BASE	0	Not used
SC_BAL_LIM	127	Not used
SC_BAL_TH	0	Not used
SC_PH_BASE	0	Not used
SC_PH_LIM	127	Not used
SC_PH_TH	0	Not used
Z_PH_TH	0	Not used
Z_GNA_COR	8	Affected by C3

Table 7: Pin Configuration Mode Register Values

Serial Configuration Mode

In serial configuration mode (PINCFG = 0 V), configuration values for all static registers must be written to the TW28's internal EEPROM using the SPI port or the Encoder Link interface before the device can be used. For stand-alone applications, the easiest way to accomplish this is to use the iC-TW28 demo board and the free Graphical User Interface (GUI) software.

The iC-TW28 TW28_1D evaluation board implements the iC-TW28 and a USB interface for direct communication with the GUI software running on a Windows PC. A functional prototype encoder can thus be quickly assembled and configured. See the TW28_1D evaluation board documentation for more information.

The TW28_1D evaluation board can also be used for prototype development to configure an external iC-TW28 via the SPI or Encoder Link interfaces using the free GUI software. In series production, the evaluation board can be employed to download pre-engineered configurations to iC-TW28s embedded in products.

In hosted applications (where the SPI port is used to communicate with the host processor or microcontroller), the iC-TW28 must be configured via the host or using the Encoder Link interface.

CALIBRATION OVERVIEW

Once the iC-TW28 has been configured, the sin and cos channels must be calibrated to determine proper values for gain, offset correction, channel balance, and phase correction. This is most easily done using the auto calibration feature of the iC-TW28 to automatically determine optimum values for these parameters. If the ZERO inputs are used, the Z channel must also be calibrated for gain, offset correction, and phase. Auto calibration is also available for the Z channel.

Auto calibration can be initiated in hardware using the xCALIB input or via software commands in serial configuration mode using the SPI port or the Encoder Link interface. With auto calibration initiated, provide sensor input of a few hundred sin/cos cycles and the iC-TW28 tunes the correction parameters to provide lowest error and jitter in the interpolated AB and/or UVW outputs. Z channel auto calibration can be accomplished along with AB/UVW auto calibration or separately.

The sensor input used for auto calibration does not need to be at a constant frequency nor must it be unidirectional. A rotary encoder can be calibrated by moving the disc or wheel back and forth by a few revolutions; a linear encoder by moving the sensor back and forth on the scale by a few centimeters. If Z auto calibration is used, input motion must include generation of a ZERO input signal.

After providing sufficient input signals, auto calibration is terminated using the xCALIB input or software commands and **the tuned correction values** are stored to the internal EEPROM for use on subsequent startups.

Application Hint
If no cyclic input signals are present, auto calibration only roughly adjusts gain and offset for non-matching input signals to avoid overloading the ADC.

Hardware Auto Calibration (xCALIB)
Hardware auto calibration can be used in both pin and serial configuration modes and is initiated by pulling the xCALIB input low. A push-button switch and pull-up resistor connected between xCALIB (as shown in Figure 10 on page 18) is an easy way to achieve this in series production. Auto calibration can also tune the Z channel. In pin configuration mode, this is controlled by configuration input C3. In serial configuration mode, auto calibration can be configured to tune or not tune the Z channel and to automatically store or not store the calibrated parameters to EEPROM when xCALIB is released.

The recommended sequence for hardware auto calibration in pin configuration mode is:

1. Pull xCALIB input low.
2. Provide sensor input signals as explained in Calibration Overview.
3. Release the xCALIB input (it is pulled high by the external pull-up resistor) to store all calibrated values to the EEPROM.

The recommended sequence for hardware auto calibration in serial configuration mode is:

1. Ensure all static (configuration) registers have valid values for the desired application, especially SC_AMP_TARG (the recommended value is 150).
2. Pull xCALIB input low.
3. Provide sensor input signals as explained in Calibration Overview.
4. Release the xCALIB input (it is pulled high by the external pull-up resistor). If ADAPT_CFG1.xcalee = 1, **the tuned correction values** are automatically stored to the EEPROM.
5. If ADAPT_CFG1.xcalee = 0, store the calibrated COR register values to the BASE registers by writing 0x12 to the COMMAND register. Then store the calibrated COR and BASE register values to the internal EEPROM by writing 0x11 to the COMMAND register.

Software Auto Calibration
Software auto calibration is available only in serial configuration mode and is accomplished using either the SPI port or the Encoder Link interface to send auto calibration commands to the COMMAND register (see Table 90 on page 50). After sending the appropriate command, provide sensor input signals as for hardware auto calibration.

When calibration is complete, stop auto calibration and write the calibrated values to the EEPROM for use on subsequent startups. After auto calibration, all the error correction residue values (RES registers) should be zero (or near zero). If this is not the case, auto calibration should be repeated.

The recommended sequence for software auto calibration is:

1. Ensure all static (configuration) registers have valid values for the desired application, especially SC_AMP_TARG (the recommended value is 150).
2. Initiate auto calibration by writing 0x23 to the COMMAND register (0x4000).
3. Provide sensor input signals as explained in Calibration Overview.
4. Terminate auto calibration by writing 0x20 to the COMMAND register.
5. Store the calibrated COR register values to the BASE registers by writing 0x12 to the COMMAND register.
6. Store the calibrated COR and BASE register values to the internal EEPROM by writing 0x11 to the COMMAND register.

STARTUP

In operation, the startup sequence is initiated when power is applied to the iC-TW28. However, a startup sequence can also be initiated by external hardware connected to the xRST input, or by a command via the SPI port or the Encoder Link interface in serial configuration mode.

At startup, the iC-TW28's POR circuit monitors the supply voltage and waits until it has reached 2.7...2.9 V. In pin configuration mode, the iC-TW28 then waits 10 ms, reads the configuration data from EEPROM, and starts ABZ/UVW output generation.

In serial configuration mode, the wait time is programmable between 0 and 514 ms and the iC-TW28 can be configured not to start ABZ output generation after reading the configuration data. This is useful in hosted applications to allow the host processor or microcontroller to start the iC-TW28.

Also in serial configuration mode, the state of the AB outputs relative to the Z output at startup is programmable. See Startup Modes on page 58 for more information.

If any errors are detected during the start-up cycle, the iC-TW28 does not enable the outputs but goes into an idle state with xIRQ asserted.

SPI COMMUNICATION

The SPI port is a 4-wire slave interface which operates in CPOL = 0 and CPHA = 0 mode only. This means that the base (resting) value of SCLK is low, SI is sampled on the rising edge of SCLK, and SO is changed on the falling edge of SCLK. The active-low Slave Select input, xSS, is used by the host µP to enable the SPI port to initiate communication.

SPI communication uses an overlapped packet structure where the response to a command is returned while the next command is being sent. Figure 21 shows this for a single-device application, where the host controls a single iC-TW28 slave (see Figure 20). See Bussing Multiple iC-TW28s on page 75 and Chaining Multiple iC-TW28s on page 76 for information on multiple-device applications.

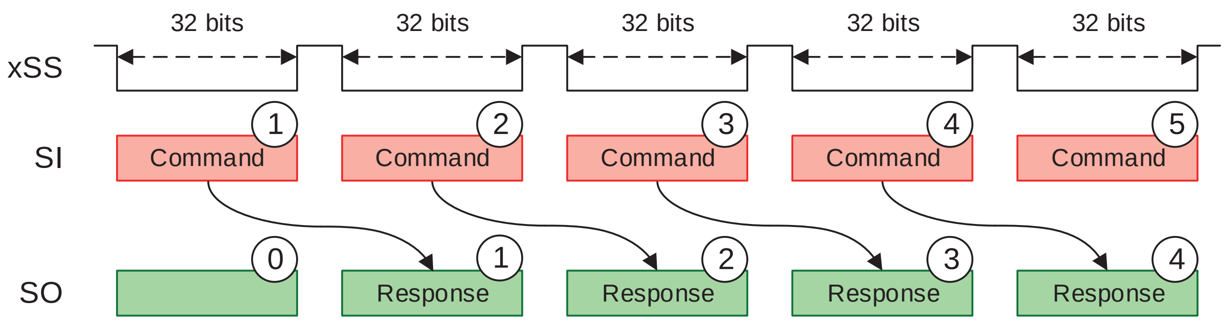


Figure 21: SPI Overlapped Packet Structure

SPI command and response packets are always 32 bits long and sent most-significant bit first. The host initiates communication with the iC-TW28 by driving Slave Select (xSS) low and then clocking a 32-bit command (1) to the Slave Input, SI. The serial clock (SCLK) signal is not shown in Figure 21. The host drives xSS high at the end of the command packet and the iC-TW28 executes the command.

After waiting for the command to be executed, the host again drives xSS low and sends the next command packet (2) to SI while at the same time reading the 32-bit response (1) to the initial command (1) on the Slave Output, SO.

The iC-TW28 always returns a response packet while reading a command packet. The response packet (0) returned while writing the first command packet (1) is not defined.

Note: The host must not pulse xSS low without shifting 32 bits of data into SI. Doing so will cause unpredicatable results and an unstable device. After the host drives xSS low, 32 SCLKs *must* be received by the iC-TW28 before xSS is driven high again.

Note: xSS must also be high when xRST is released.

The available read and write commands are shown in Figure 22 and explained in detail on the following pages.

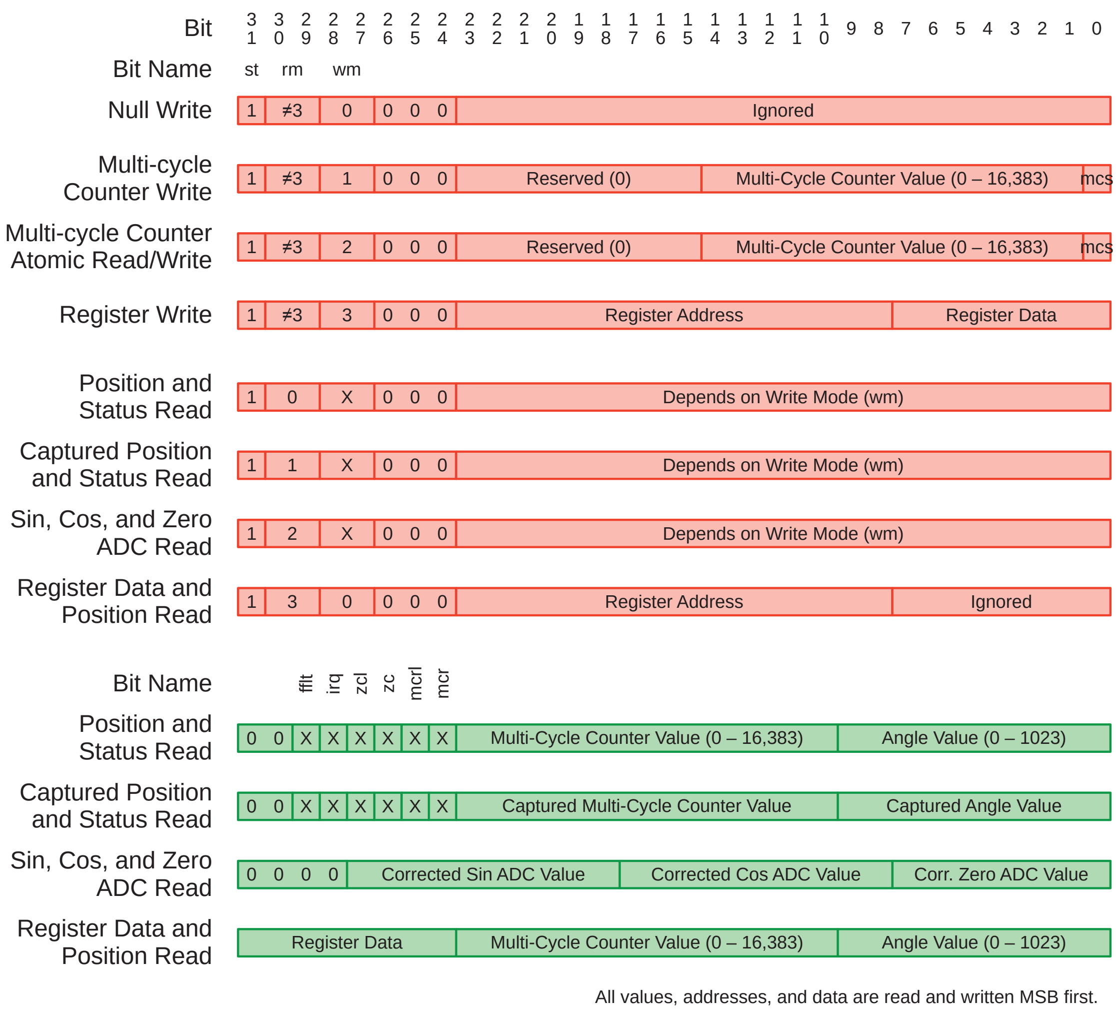


Figure 22: SPI Command Reference

Command Packet Format
Command packets sent to the iC-TW28 by the host are formatted as shown below.

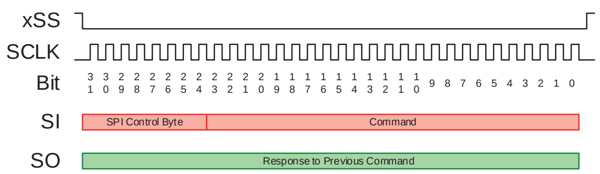


Figure 23: Command Packet Format

Because of the iC-TW28's overlapped packet structure, it is possible to write data to a register or the multiturn counter *and* request read-back of position or other values in the same command. The control byte handles this, as shown below.

Command	SPI Control Byte		
Bit	Bit	Name	Description
31	7	st	Start (must be 1)
30:29	6:5	rm	Read mode
28:27	4:3	wm	Write mode
26:24	2:0	–	Reserved (must be 0)

Table 8: SPI Control Byte

A read-only command is one in which the write mode (wm) is 0. The type of data read is determined by the read mode (rm) encoded in the control byte. A read-/write command is one in which the write mode (wm) is 1, 2, or 3.

The type of data written by an SPI read/write command is determined by the write mode encoded in the control byte of the command packet as shown below.

SPI Write Modes	
wm	Description
0	Null Write (read only)
1	Multi-Cycle Counter Write
2	Multi-Cycle Counter Atomic Read/Write
3	Register Write

Table 9: SPI Write Modes

The type of data returned by an SPI read-only or read-/write command is determined by the read mode encoded in the control byte of the command packet as shown below.

SPI Read Modes	
rm	Description
0	Position and Status Read (20 ns or 320 ns)
1	Captured Position and Status Read
2	Sin, Cos, and Zero ADC Read (320 ns)
3	Register Data and Position Read

Table 10: SPI Read Modes

The requested values are returned in the subsequent response packet. Position (angle) is always returned at the full 10-bit resolution of the iC-TW28, regardless of the interpolation value (INTER).

All values are read on the falling edge of xSS. However, the internal update rates of the various values are different. In all cases, the value read is the most recently updated internal value. See Response Packet Formats on page 32 for more details on the internal update rates.

Null Write (Read Only)

The Null Write command packet is formatted as shown below.

Null Write: wm = 0	
Bits	Description
31:24	SPI Control Byte
23:0	Ignored (Register Address if rm = 3)

Table 11: Null Write Command Packet

The Null Write (Read Only) command does not write any data to the iC-TW28. The data specified by the read mode in the control byte is returned with the next SPI command.

If rm = 0, 1, or 2, the command bits in the command packet are ignored, but must be present to complete the 32-bit packet. If rm = 3 (Register Data and Position Read), the command bits in the command packet are used to specify the register address to read. See Register Data and Position Read on page 32 for more information.

Multi-Cycle Counter Write

The Multi-Cycle Counter Write command packet is formatted as shown below.

Multi-Cycle Counter Write: wm = 1	
Bits	Description
31:24	SPI Control Byte
23:15	Reserved (must be 0)
14:1	Multi-Cycle Counter Value (0 – 16,383)
0	Multi-Cycle Counter Synchronization Bit (mcs)

Table 12: Multi-Cycle Counter Write Command Packet

The specified multi-cycle counter value is written immediately to the multi-cycle counter and the data specified by the read mode in the control byte is returned with the next SPI command.

The multi-cycle counter synchronization bit (mcs) allows synchronization between an external absolute system *and* the multi-cycle counter in the iC-TW28, even when the sin/cos inputs are moving. See Multi-Cycle Counter on page 67 for more information.

Multi-Cycle Counter Atomic Read/Write

The Multi-Cycle Counter Atomic Read/Write command packet is formatted as shown below.

Multi-Cycle Counter Atomic Read/Write: wm = 2	
Bits	Description
31:24	SPI Control Byte
23:15	Reserved (must be 0)
14:1	Multi-Cycle Counter Value (0 – 16,383)
0	Multi-Cycle Counter Synchronization Bit (mcs)

Table 13: Multi-Cycle Counter Atomic Read/Write Command Packet

The Multi-Cycle Counter Atomic Read/Write command is like the Multi-Cycle Counter Write command except that the specified multi-cycle counter value is written to the multi-cycle counter at the same instant as the data specified by the read mode in the control byte is read. Writing of the multi-cycle counter value is delayed until the next SPI command to allow simultaneous reading of the position and writing of the multi-cycle counter value for synchronization confirmation when using external absolute systems. See Multi-Cycle Counter on page 67 for more information.

Register Write

The Register Write command packet is formatted as shown below.

Register Write: wm = 3	
Bits	Description
31:24	SPI Control Byte
23:8	Register Address
7:0	Register Data

Table 14: Register Write Command Packet

The specified register data is written to the register at the specified register address *and* the data specified by the read mode in the control byte is returned with the next SPI command.

Register Data and Position Read

The Register Data and Position Read command packet is formatted as shown below.

Register Data and Position Read: rm = 3, wm = 0	
Bits	Description
31:24	SPI Control Byte
23:8	Register Address
7:0	Ignored

Table 15: Register Data and Position Read Command Packet

The data at the specified register address as well as the multi-cycle counter and angle values are returned with the next SPI command.

Note that the Register Data and Position Read Command requires the write mode in the SPI Control Byte (wm) to be zero. Non-zero wm values result in undefined operation.

Response Packet Formats

The format of the response packet is determined by the read mode specified in the control byte of the previous command packet.

Position and Status Read

The Position and Status Read response packet is formatted as shown below.

Position and Status Read Response: rm = 0	
Bits	Description
31:24	SPI Status Byte
23:10	Multi-Cycle Counter Value (0 – 16,383)
9:0	Angle Value (0 – 1023)

Table 16: Position and Status Read Response Packet

The SPI Status Byte reports the status of the signal path, the most recent capture, and the multi-cycle counter as shown below.

Response	SPI Status Byte		
Bit	Bit	Name	Description
31:30	7:6	–	0 (Reserved)
29	5	fflt	Fatal Fault Occurred
28	4	irq	Interrupt Request Active
27	3	zcl	Zero Capture Lost
26	2	zc	Zero Capture Occurred
25	1	mcrl	Multi-Cycle Counter Rollover Lost
24	0	mcr	Multi-Cycle Counter Rollover Occurred

Table 17: SPI Status Byte

The Fatal Fault (fflt) bit is set if one or more of the bits in the STAT_FATAL register is set, indicating that a fatal fault occurred. The interpolator is disabled after a fatal fault and must be restarted by a serial command or by cycling power.

The Interrupt Request (irq) bit indicates that there is a pending internal interrupt request or fault.

The Zero Capture Occurred (zc) bit is set whenever a zero capture event occurs. This bit is reset when the captured position is read. A zero capture event can also be configured to request an interrupt to the host processor by asserting xIRQ. See STAT_CFG on page 46 for more information.

The Zero Capture Lost (zcl) bit is set whenever a zero capture event occurs while the Zero Capture Event (zc) bit is still active. This condition indicates that the captured position from a previous capture event has been lost. This bit is reset when the captured position is read. See Position Capture on page 69 for more information.

The Multi-Cycle Counter Rollover Occurred (mcr) bit is set whenever the multi-cycle counter passes through a multiple of 4,096 cycles. This bit is reset whenever the position is read. A multi-cycle counter rollover can also be configured to request an interrupt to the host processor by asserting xIRQ. See STAT_CFG on page 46 for more information.

The Multi-Cycle Counter Rollover Lost (mcrl) bit is set whenever the multi-cycle counter passes through a multiple of 4,096 cycles while the Multi-Cycle Counter Rollover (mcr) bit is still active. This condition indicates that a previous multi-cycle counter rollover was not acknowledged. This bit is reset whenever the position is read. See Multi-Cycle Counter on page 67 for more information.

The Multi-Cycle Counter Value is a 14-bit number representing the number of input cycles seen by the iC-TW28 since the iC-TW28 was started (or restarted) or since the multi-cycle counter was reset. If the multi-cycle counter is not used, this value can be ignored. If the ABZ outputs are enabled (OUTPUT.abzen = 1), this value is updated internally every 320 ns. If the ABZ outputs are disabled (OUTPUT.abzen = 0), this value is updated internally every 20 ns, but is only correct if INTER = 0 (interpolation of 256).

The Angle Value is the angular position within an input cycle as indicated by the sin/cos sensor. It is always returned at the full 10-bit resolution of the iC-TW28, regardless of the INTER value. If the ABZ outputs are enabled (OUTPUT.abzen = 1), this value is updated internally every 320 ns. If the ABZ outputs are disabled (OUTPUT.abzen = 0), this value is updated internally every 20 ns, but is only correct if INTER = 0 (interpolation of 256).

Captured Position and Status Read

The Captured Position and Status Read response packet is formatted as shown below.

Captured Position and Status Read Response: rm = 1	
Bits	Description
31:24	SPI Status Byte
23:10	Captured Multi-Cycle Counter Value
9:0	Captured Angle Value (0 – 1023)

Table 18: Captured Position and Status Read Response Packet

The SPI Status Byte reports the status of the signal path, the most recent capture, and the multi-cycle counter as described previously.

The Captured Multi-Cycle Counter Value is a 14-bit number representing the number of input cycles seen by the iC-TW28 since the iC-TW28 was started or restarted as of the occurrence of the last Z pulse or Z gating window as determined by register ZERO0.mode. This value is only correct if INTER = 0 (interpolation of 256).

The Captured Angle Value is the angular position within an input cycle as indicated by the sin/cos sensor as of the occurrence of the last Z pulse or ZERO input gating window signal as determined by ZERO0.mode. This value is only correct if INTER = 0 (interpolation of 256).

See Position Capture on page 69 for more information.

Sin, Cos, and Zero ADC Read

The Sin, Cos, and Zero ADC Values Read response packet is formatted as shown below.

Sin, Cos, and Zero ADC Read Response: rm = 2	
Bits	Description
31:28	Reserved (0)
27:18	Corrected Sin ADC Value
17:8	Corrected Cos ADC Value
7:0	Corrected Zero ADC Value

Table 19: Sin, Cos, and Zero ADC Read Response Packet

The Corrected Sin ADC Value is a signed (2's complement) 10-bit value representing the most recently sampled sine ADC value after offset, gain, and phase correction. The Corrected Cos ADC Value is a signed (2's complement) 10-bit value representing the most recently sampled cosine ADC value after offset, gain, and phase correction. The Corrected Zero ADC Value is a signed (2's complement) 8-bit value representing the most recently sampled zero ADC value after offset and gain correction. The ADC values are updated every 320 ns.

Register Data and Position Read

The Register Data and Position Read response packet is formatted as shown below.

Register Data and Position Read Response: rm = 3, wm = 0	
Bits	Description
31:24	Register Data
23:10	Multi-Cycle Counter Value (0 – 16,383)
9:0	Angle Value (0 – 1023)

Table 20: Register Data and Position Read Response Packet

Register Data contains the value of the register at the address specified in the previous Register Data and Position Read command packet.

The Multi-Cycle Counter Value is a 14-bit number representing the number of input cycles seen by the iC-TW28 since the iC-TW28 was started or restarted. This value is only correct if INTER = 0 (interpolation of 256). If the multi-cycle counter is not used, this value can be ignored. If the ABZ outputs are enabled (OUTPUT.abzen = 1), this value is updated internally every 320 ns. If the ABZ outputs are disabled (OUTPUT.abzen = 0), this value is updated internally every 20 ns, but is only correct if INTER = 0 (interpolation of 256).

The Angle Value is the angular position within an input cycle as indicated by the sin/cos sensor. It is always returned at the full 10-bit resolution of the iC-TW28, regardless of the INTER value. If the ABZ outputs are enabled (OUTPUT.abzen = 1), this value is updated internally every 320 ns. If the ABZ outputs are disabled (OUTPUT.abzen = 0), this value is updated internally every 20 ns, but is only correct if INTER = 0 (interpolation of 256).

ENCODER LINK COMMUNICATION

In serial configuration mode, the Encoder Link interface provides read/write access to the iC-TW28's internal registers using the A+ and A– outputs. This is useful for field reconfiguration or diagnostics of products incorporating the iC-TW28. The Encoder Link interface can be used for configuration and diagnostics; it cannot be used to read the sensor position (angle), the multi-cycle counter, or the captured position values.

Note: Encoder Link is not available in pin configuration mode.

To enable the Encoder Link interface, the iC-TW28 output drivers must be externally over-driven in the activation sequence shown in Figure 24.

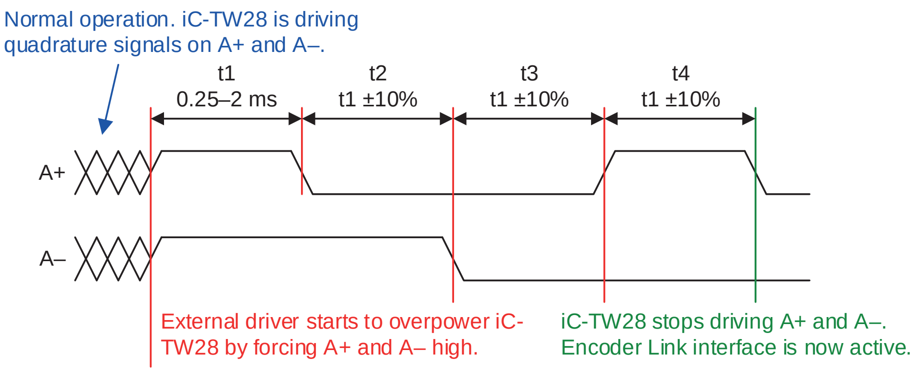


Figure 24: Encoder Link Activation Sequence

The external drivers used to overpower the iC-TW28 must be capable of sourcing and sinking at least 40 mA while driving the A+ and A– outputs to voltage levels < 0.8 V and > 2.4 V. The complete Encoder Link activation sequence takes between 1 and 8 milliseconds (t1 + t2 + t3 + t4) to execute. At the end of the activation sequence, the A+ and A– outputs are both low and the Encoder Link interface is active.

Once activated, the Encoder Link interface provides bidirectional two-wire SPI-like serial communication with the iC-TW28. To deactivate the Encoder Link interface and return to normal operating mode, cycle power to the iC-TW28 or toggle the xRST input.

In SPI Only output mode (where the outputs are floating in a high impedance state) or ABZUVW output mode (where the A+ and A– outputs are independent of each other), it is possible to accidentally activate Encoder Link. In these modes, it is recommended to disable the Encoder Link interface by setting MAIN_CFG.elinkoff = 1. See Output Modes, Directions, and Polarities on page 56 and MAIN_CFG on page 38 for more information.

Encoder Link Write
The Encoder Link write command is 32 bits long and formatted as shown in Figure 25.

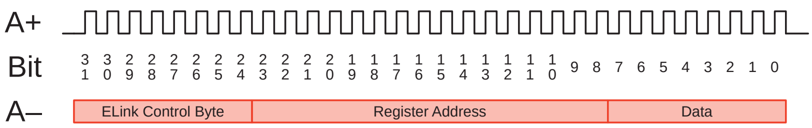


Figure 25: Encoder Link Write

The host supplies a serial clock signal to the A+ pin and writes the command bits (MSB first) to the A– pin. Command bits are latched on the rising edge of A+.

The ELink command byte determines whether the command is a read or a write.

Command Bit	ELink Control Byte Bit	Name	Description
31	7	st	Start (must be 1)
30	6	wr	Write
29:24	5:0	–	Reserved (must be 0)

Table 21: ELink Control Byte

A write command is one in which the write bit (wr) is 1. The data byte specified in the write command (bits 7:0) is written to the register address specified in the command (bits 23:8).

Encoder Link Read
The Encoder Link read command is 40 bits long and formatted as shown in Figure 26.

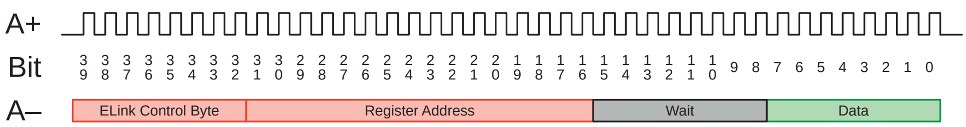


Figure 26: Encoder Link Read

A read command is one in which the write bit (wr) of the ELink control byte is 0. As with the write command, the host supplies a serial clock signal to the A+ pin and writes the command bits (MSB first) to the A– pin. Command bits are latched on the rising edge of A+.

During the wait byte, the host must stop driving A–. The iC-TW28 then drives the data from the register address specified in the read command (bits 31:16) to the A- pin (bits 7:0). The host must sample the data on the rising edge of A+. At the end of the read command, the A– pin reverts to being an input and the host must drive it low to be ready to send the next command.

CONFIGURATION PARAMETERS

Register Map
The iC-TW28 register map is shown in Tables 22 and 23. Register features are shown in the Type column as follows:

E indicates that the register value is stored in the internal EEPROM and restored at startup or restart. Unless otherwise stated, these registers may be written by the user via the SPI or Encoder Link interfaces in serial configuration mode, but the modified values are not automatically stored to the EEPROM.

D indicates that the register is dynamic; its value may be modified by the iC-TW28 during operation. The user may also write to these registers to override the calculated value.

R indicates that the register is dynamic and read-only. Its value may be modified by the iC-TW28 during operation but cannot be modified by the user.

Registers without a code in the Type column may be read and written by the user. Registers not shown are reserved and must not be accessed.

Address	Register Name	Description								Type
		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	
0x0000	MAIN_CFG	0	0	elinkoff	irqpp	rs422	noise	input		E
0x0001	LED_CFG	buffer		freq		auto	odrain	pol	en	E
0x0002	LED_START	Initial (Starting) LED PWM Value								E
0x000B	TEST	0	0	0	0	0	z	0	we	
0x0010	LED_PWM	Actual LED PWM Value								R
0x0100	UVW_CFG	0	0	0	pairs					E
0x0101	INTER0	interlsb								E
0x0102	INTER1	0	0	0	div			intermsbs		E
0x0103	AB	0	0	0	hyst					E
0x0104	UVW	0	0	0	hyst					E
0x0105	FALARM	input frequency alarm level								E
0x0106	ABLIMIT	AB frequency limit								E
0x0107	ZERO0	mode	clr	threshold						E
0x0108	ZERO1	0	0	0	0	zwidth				E
0x0109	OUTPUT	uvwen	abzen	uvwdir	uvwpol	abzdir	zpol	bpol	apol	E
0x010A	ZPHASE	msb								E, D
0x010B	UVWPH	msb								E
0x010C	PHASE_LSB	0	0	0	0	uvw		z		E, D
0x020C	Z_ADC	Zero ADC								D
0x021A	S_ADC	Sine ADC MSB								D
0x021C	C_ADC	Cosine ADC MSB								D
0x0300	FILT_CFG	0	0	0	kpmax			fb	auto	E
0x0301	FILT_LAG	0	0	0	threshold					E
0x0302	FILT_K	0	0	ki			kp			E, D
0x0400	STAT_CFG	0	0	enz	enc	long	filter			E
0x0401	STAT_SEL	res	adapt	scamp	lagfatl	inclim	laglim	falarm	oflow	E
0x0402	STAT_IE	res	adapt	scamp	lagfatl	inclim	laglim	falarm	oflow	E
0x0403	STAT_HIZ	res	adapt	scamp	lagfatl	inclim	laglim	falarm	oflow	E
0x0404	STAT_VAL	res	adapt	scamp	lagfatl	inclim	laglim	falarm	oflow	D
0x0405	STAT_LATCH	res	adapt	scamp	lagfatl	inclim	laglim	falarm	oflow	D
0x0406	STAT_FATAL						interr	ee2bit	eechk	D
0x0601	EE_ADDR	EEPROM address to read or write								
0x0602	EE_DATA	EEPROM data								D
0x0603	EE_STAT	Validity of read EEPROM data								D

Table 22: iC-TW28 Register Map

Address	Register Name	Description								Type
		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	
0x4000	COMMAND	Command register								D
0x4001	START	0	0	nostart	mode		wait			E
0x4002	ADAPT_CFG0	zphase	zgain	zofs	scgain	scofsa	scph	scbal	scofs	E
0x4003	ADAPT_CFG1	0	0	xcalee	zcal	stop	p			E
0x4004	SC_AMP_TARG	Sin/Cos amplitude monitor target (0...180)								E
0x4005	SC_AMP_LOW	Sin/Cos amplitude monitor low limit (0...SC_AMP_TARG)								E
0x4006	SC_AMP_HIGH	Sin/Cos amplitude monitor high limit (SC_AMP_TARG...180)								E
0x4007	S_OFS_BASE	Sin channel digital offset base value (±127)								E
0x4008	C_OFS_BASE	Cos channel digital offset base value (±127)								E
0x4009	SC_OFS_LIM	Sin and Cos channels digital offset limit (0...127)								E
0x400A	SC_OFS_TH	Sin and Cos channels offset residue threshold (0...127)								E
0x400B	SC_BAL_BASE	Sin/Cos balance base value (±127)								E
0x400C	SC_BAL_LIM	Sin/Cos balance limit (0...127)								E
0x400D	SC_BAL_TH	Sin/Cos balance residue threshold (0...127)								E
0x400E	SC_PH_BASE	Sin/Cos phase base value (±127)								E
0x400F	SC_PH_LIM	Sin/Cos phase limit (0...127)								E
0x4010	SC_PH_TH	Sin/Cos phase residue (0...127)								E
0x4011	Z_PH_TH	Z channel phase residue (0...127)								E
0x4012	S_OFS_COR	Sin channel digital offset correction value (±127)								E, D
0x4013	S_OFSA_COR	Sin channel analog offset correction value (±31)								E, D
0x4014	C_OFS_COR	Cos channel digital offset correction value (±127)								E, D
0x4015	C_OFSA_COR	Cos channel analog offset correction value (±31)								E, D
0x4016	SC_BAL_COR	Sin/Cos balance correction value (±127)								E, D
0x4017	SC_GN_COR	Sin/Cos digital gain correction value (0...255)								E, D
0x4018	SC_GNA_COR	Sin/Cos analog gain correction value (0...23)								E, D
0x4019	SC_PH_COR	Sin/Cos phase correction value (±127)								E, D
0x401A	Z_OFSA_COR	Z channel analog offset correction value (±31)								E, D
0x401B	Z_GNA_COR	Z channel analog gain correction value (0...23)								E, D
0x401C	WATCHDOG	Watchdog (31)								D
0x4020	SC_AMP	Sin/Cos amplitude (0...255)								R
0x4021	S_AMP	Sin channel amplitude (0...255)								R
0x4022	C_AMP	Cos channel amplitude (0...255)								R
0x4023	S_OFS_RES	Sin channel offset residue (±127)								R
0x4024	C_OFS_RES	Cos channel offset residue (±127)								R
0x4025	SC_BAL_RES	Sin/Cos balance residue (±127)								R
0x4026	SC_PH_RES	Sin/Cos phase residue (±127)								R
0x4028	Z_PH_RES	Z channel phase residue (±127)								R
0xE000	DEVICE_ID	iC-TW28 Device ID								R
0xE002	REV0	iC-TW28 Revision LSB								R
0xE003	REV1	iC-TW28 Revision MSB								R

Table 23: iC-TW28 Register Map (continued)

MAIN_CFG
MAIN_CFG.input configures the input stage of the sin/-cos inputs.

MAIN_CFG.input (0x0000 Bits 1:0)	
Value	Description
0	Low range signals (0 dB)
1	Low range signals with sensor loss detection
2	High range signals (-9 dB) with sensor loss detection
3	Reserved (do not use)

Table 24: Input Stage Configuration

See Input Configuration and Signal Levels on page 55 for more information.

MAIN_CFG.noise selects the amount of noise filtering applied to the input signals, which also affects the latency of the iC-TW28.

MAIN_CFG.noise (0x0000 Bit 2)	
Value	Description
0	Less filtering, 1.5 µs latency
1	More filtering, 1.9 µs latency

Table 25: Input Noise Filter

Latency in the amount of time it takes for a change in the sin/cos inputs to show up in the AB or UVW outputs. See Filter Configuration on page 70 for more information.

MAIN_CFG.rs422 enables and disables the RS422-compatible line driver on the ABZ/UVW outputs.

MAIN_CFG.rs422 (0x0000 Bit 3)	
Value	Description
0	Line driver disabled (standard digital outputs)
1	Line driver enabled

Table 26: RS422 Line Driver Enable

Application Hint
If RS422 drivers are operated without termination, they cannot handle pull-ups or pull-downs and provide varying edge shapes.

When the line driver is disabled, the ABZ/UVW outputs are standard digital outputs. See Output Modes, Directions, and Polarities on page 56 for more information.

MAIN_CFG.irqpp determines whether the xIRQ output is open-drain or push-pull.

MAIN_CFG.irqpp (0x0000 Bit 4)	
Value	Description
0	xIRQ output is open-drain
1	xIRQ output is push-pull

Table 27: Interrupt Output Configuration

Normally, the xIRQ output is push-pull. In multidevice chained applications, the xIRQ output can be set to open-drain to allow wire-ORing multiple iC-TW28 xIRQ outputs using a pull-up resistor. See Chaining Multiple iC-TW28s on page 76 for more information.

MAIN_CFG.elinkoff disables the Encoder Link interface.

MAIN_CFG.elinkoff (0x0000 Bit 5)	
Value	Description
0	Encoder Link interface available
1	Encoder Link interface disabled

Table 28: Encoder Link Interface Disable

When MAIN_CFG.elinkoff = 1, the Encoder Link interface cannot be activated by the sequence shown in Figure 24. Set MAIN_CFG.elinkoff = 1 when operating in SPI Only or ABZUVW output modes. See Output Modes, Directions, and Polarities on page 56 for more information.

LED_CFG
LED_CFG is used to configure the LED intensity control feature of the iC-TW28. See LED Intensity Control on page 72 for more information on using this feature.

LED_CFG.en enables or disables the LED pin.

LED_CFG.en (0x0001 Bit 0)	
Value	Description
0	LED pin disabled
1	LED pin enabled

Table 29: LED Output Enable

If LED_CFG.en = 0, the LED output pin is in the state shown below.

LED Pin State (LED_CFG.en = 0)		
LED_CFG.odrain	LED_CFG.pol	State
0	0	Low
0	1	High
1	X	Hi-Z

Table 30: LED Pin State (LED_CFG.en = 0)

LED_CFG.pol selects the polarity of the LED PWM signal.

LED_CFG.pol (0x0001 Bit 1)	
Value	Description
0	Positive polarity
1	Negative polarity

Table 31: PWM Polarity

With positive LED polarity, a larger PWM value results in more high time in the LED PWM signal. With negative LED polarity, a larger PWM value results in more low time in the LED PWM signal. If LED_CFG.en = 0, LED_CFG.pol has no effect.

LED_CFG.odrain selects whether the LED output is open-drain or push-pull.

LED_CFG.odrain (0x0001 Bit 2)	
Value	Description
0	Push-pull LED output
1	Open-drain LED output

Table 32: LED Output Configuration

If LED_CFG.en = 0, LED_CFG.odrain has no effect.

LED_CFG.auto enables and disables the LED intensity control.

LED_CFG.auto (0x0001 Bit 3)	
Value	Description
0	LED intensity control off
1	LED intensity control on

Table 33: LED Intensity Control

If LED_CFG.en = 0, LED_CFG.auto has no effect. If LED_CFG.en = 1 and LED_CFG.auto = 0, the signal on the LED output is the value of LED_START.

LED_CFG.freq selects the DSM (Delta-Sigma Modulation) frequency of the LED intensity control output. The corresponding PWM frequency is shown for use in selecting filter components.

LED_CFG.freq (0x0001 Bits 5:4)		
Value	DSM Frequency	PWM Frequency
0	3.125 MHz	12.2 kHz
1	6.25 MHz	24.4 kHz
2	12.5 MHz	48.8 kHz
3	25 MHz	97.6 kHz

Table 34: LED Modulation Frequency

LED_CFG.buffer is used to set the hysteresis of the LED intensity control.

LED_CFG.buffer (0x0001 Bits 7:6)	
Value	Buffer
0	8
1	16
2	24
3	32

Table 35: LED Control Loop Buffer

If LED_CFG.en = 1 and LED_CFG.auto = 1, the LED intensity control increases the duty cycle of the LED PWM signal whenever SC_AMP < (SC_AMP_LOW + LED_CFG.buffer) and decreases it whenever SC_AMP > (SC_AMP_HIGH – LED_CFG.buffer). For proper operation,

$$LED\_CFG.buffer < \frac{SC\_AMP\_HIGH - SC\_AMP\_LOW}{2}$$

If LED_CFG.en = 0, or if LED_CFG.en = 1 and LED_CFG.auto = 0, LED_CFG.buffer has no effect.

LED_START
LED_START contains the initial (starting) value for the duty cycle of the LED PWM output.

LED_START (0x0002)	
Value	Description
0...255	Starting LED PWM duty cycle

Table 36: Starting LED PWM Duty Cycle

If LED_CFG.en = 1, LED_START is loaded from the internal EEPROM at startup and used as the initial value for the LED PWM output. If LED_CFG.en = 1 and LED_CFG.auto = 0, the LED output stays at this value. If LED_CFG.en = 1 and LED_CFG.auto = 1, the LED output is adjusted by the LED intensity control to maintain optimum illumination of an optical sin/cos sensor.

If LED_CFG.en = 0, LED_START has no effect. See LED Intensity Control on page 72 for more information.

The LED_START value is also used during calibration (xCALIB input low or command 0x21 – 0x23 active) regardless of the settings of the LED_CFG register bits.

LED_PWM
LED_PWM contains the actual current operating duty cycle of the LED PWM output.

LED_PWM (0x0010)	
Value	Description
0...255	Actual LED PWM duty cycle

Table 37: Actual LED PWM Duty Cycle

If LED_CFG.en = 1, LED_PWM is loaded from LED_START at startup. If LED_CFG.en = 1 and LED_CFG.auto = 0, the LED_PWM stays equal to LED_START and the LED output stays at this value. If a new value is written to LED_PWM, the LED output is immediately set to this level.

If LED_CFG.en = 1 and LED_CFG.auto = 1, LED_PWM (and thus the LED output) is adjusted by the LED intensity control to maintain optimum illumination of an optical sin/cos sensor. If LED_CFG.en = 0, LED_PWM has no effect. See LED Intensity Control on page 72 for more information.

TEST
TEST is used to unlock the internal EEPROM and enable Z test mode. The TEST register value is *not* stored in EEPROM and is set to 0 at every startup.

TEST.we (0x000B Bit 0)	
Value	Description
0	EEPROM locked (write protection enabled)
1	EEPROM unlocked (write prot. disabled)

Table 38: EEPROM Unlock

The EEPROM must be unlocked (TEST.we = 1) to write to it using COMMAND register commands. The EEPROM can be written at the rising edge of the xCALIB input regardless of the value of TEST.we. See Hardware Auto Calibration (xCALIB) on page 27 for more information.

TEST.z (0x000B Bit 2)	
Value	Description
0	Normal operating mode
1	Z test mode

Table 39: Zero Test Mode

In Z test mode, the A outputs show the un-gated Z signal once per input period, the B outputs show the internal Z gating signal derived from the ZERO inputs, and the Z outputs show the gated Z output pulse. Z test mode is only available if OUTPUT.abzen = 1. See Z Test Mode and Calibration on page 65 for more information.

UVW_CFG
UVW_CFG is used to configure the UVW outputs (if used).

UVW_CFG.pairs (0x0100 Bits 4:0)	
Value	Description
0	32 UVW cycles per input cycle
1...31	1...31 UVW cycles per input cycle

Table 40: UVW Pole Pairs

If OUTPUT.uvwen = 0, UVW_CFG.pairs has no effect.

INTER0
INTER0 contains the least significant byte of the interpolator resolution, INTER.

INTER0.interlsb (0x0101)	
Value	Description
0...255	Interpolator resolution LSB (INTER [7:0])

Table 41: Interpolator Resolution LSB

INTER1
INTER1 contains the most significant bits of the Interpolator resolution and the post-AB divider.

INTER1.msbs (0x0102 Bits 1:0)	
Value	Description
0...3	Interpolator resolution MSBs (INTER [9:8])

Table 42: Interpolator Resolution MSB

The interpolator resolution, INTER, is calculated as

$$INTER = 256 \times INTER1.msbs + INTER0.lsb$$

and is the number of AB output edges per sin/cos input cycle.

INTER (9:0)	
Value	Description
0	*inter* = 256
1...7	Reserved (do not use)
8...1023	*inter* = INTER/4

Table 43: INTER (9:0)

inter is the interpolation factor in AB output cycles per sin/cos input cycle.

Application Hint
The chip must be restarted after changing *inter* to guarantee proper Z output alignment.

INTER1.div (0x0102 Bits 4:2)	
Value	Description
0	Post-AB divider disabled
1...7	Minimum...maximum post-AB divider

Table 44: Post-AB Divider

The actual value used by the post-AB divider, *div*, is calculated as

$$div = INTER1.div + 1$$

When using the post-AB divider, the effective interpolation factor, *intereff*, is

$$intereff = \frac{inter}{div}$$

The AB output frequency limit specified by ABLIMIT applies to the AB frequency *prior* to the post-AB divider. See Post-AB Divider on page 74 for more information.

AB
AB is used to set the hysteresis of the AB outputs.

AB.hyst (0x0103 Bits 4:0)	
Value	Description
0 – 31	Minimum – maximum AB hysteresis

Table 45: AB Output Hysteresis

The hysteresis in sin/cos input degrees, *abhyst*, is calculated as

$$abhyst[°] = \pm AB.hyst \times \frac{360°}{2048}$$

The equivalent hysteresis in output AB edges is a function of the interpolation factor and the AB divider.

UVW
UVW is used to set the hysteresis of the UVW outputs.

UVW.hyst (0x0104 Bits 4:0)	
Value	Description
0 – 31	Minimum – maximum UVW hysteresis

Table 46: UVW Output Hysteresis

The hysteresis in sin/cos input degrees, *uvwhyst*, is calculated as

$$uvwhyst = \pm UVW.hyst \times \frac{360°}{2048}$$

FALARM

FALARM is used to set the level of the input frequency alarm.

FALARM (0x0105)	
Value	Description
0	Reserved (do not use)
1 – 128	Min. – max. input frequency alarm level
129 – 255	Reserved (do not use)

Table 47: Input Frequency Alarm Level

An input frequency alarm (STAT_VAL.falarm) is activated if the sin/cos input frequency, *finput*, exceeds

$$finput[MHz] = 1.56 \times \frac{FALARM}{256}$$

FALARM is intended as a high input frequency alarm; not for accurate detection of input frequency.

ABLIMIT

ABLIMIT is used to set the level of the AB output frequency limiter.

ABLIMIT (0x0106)	
Value	Description
0 – 255	Max. – min. AB output frequency limit

Table 48: AB Output Frequency Limit

The actual AB output frequency limit *fab*, is calculated as

$$fab[MHz] = \frac{12.5\,MHz}{(ABLIMIT + 1) \times div}$$

The equivalent minimum time between AB edges, *tedge*, is calculated as

$$tedge[ns] = 20(ABLIMIT + 1) \times div$$

The AB output frequency limit specified by ABLIMIT applies to the AB frequency *prior* to the post-AB divider (*div*). See Post-AB Divider on page 74 for more information.

ZERO0

ZERO0 is used to set the threshold of the Z channel comparator, the capture mode, and the multi-cycle counter mode.

ZERO0.threshold (0x0107 Bits 5:0)	
Value	Description
±31	Z channel comparator threshold

Table 49: Z Comparator Threshold

ZERO0.threshold is a signed, 2's complement value used to define the width of the internal Z gating window. The internal Z gating window is active whenever the value of the conditioned ZERO input is greater than four times the comparator threshold. See Z Test Mode and Calibration on page 65 for more information.

ZERO0.clr (0x0107 Bit 6)	
Value	Description
0	Multi-cycle counter is never cleared (reset)
1	Multi-cycle counter cleared on Z output

Table 50: Multi-Cycle Counter Clear Mode

See Multi-Cycle Counter on page 67 for more information.

ZERO0.mode (0x0107 Bit 7)	
Value	Description
0	Position captured on Z output
1	Position captured on Z gating window

Table 51: Position Capture Mode

If ZERO0.mode = 0, the 24-bit position (multi-cycle counter plus angle) is captured whenever the Z outputs are activated. If ZERO0.mode = 1, the 24-bit position (multi-cycle counter plus angle) is captured whenever the internal Z gating window is activated. See Position Capture on page 69 for more information.

ZERO1

ZERO1 is used to set the width of the Z output pulse.

ZERO1.zwidth (0x0108 Bits 3:0)	
Value	Description
0 – 15	Min. – max. Z pulse width

Table 52: Z Pulse Width

The actual width of the Z output pulse, *zwidth*, in AB output edges, is calculated as

$$zwidth[edges] = \frac{ZERO1.zwidth + 1}{div}$$

ZERO1.zwidth must be less than INTER. See Z Test Mode and Calibration on page 65 for more information.

OUTPUT

OUTPUT is used to enable and disable and set the polarity and direction of the ABZ/UVW outputs. See Output Modes, Directions, and Polarities on page 56 for more information on setting the output mode.

OUTPUT.apol determines whether the polarity of the A output is normal or inverted. OUTPUT.apol also determines the state of the A output when the Z output is active. See Startup Modes on page 58 for more information.

OUTPUT.apol (0x0109 Bit 0)	
Value	Description
0	Normal A polarity
1	Inverted A polarity

Table 53: A Output Polarity

If OUTPUT.abzen = 0, OUTPUT.apol has no effect.

OUTPUT.bpol determines whether the polarity of the B output is normal or inverted. OUTPUT.bpol also determines the state of the B output when the Z output is active. See Startup Modes on page 58 for more information.

OUTPUT.bpol (0x0109 Bit 1)	
Value	Description
0	Normal B polarity
1	Inverted B polarity

Table 54: B Output Polarity

If OUTPUT.abzen = 0, OUTPUT.bpol has no effect.

OUTPUT.zpol determines whether the polarity of the Z output is normal or inverted.

OUTPUT.zpol (0x0109 Bit 2)	
Value	Description
0	Normal Z+ polarity (active high)
1	Inverted Z+ polarity (active low)

Table 55: Z Output Polarity

If OUTPUT.abzen = 0, OUTPUT.zpol has no effect.

OUTPUT.abzdir determines the counting direction of the AB outputs.

OUTPUT.abzdir (0x0109 Bit 3)	
Value	Description
0	Normal counting direction
1	Reversed counting direction

Table 56: AB Counting Direction

If OUTPUT.abzen = 0, OUTPUT.abzdir has no effect.

Application Hint
The chip must be restarted after changing *.apol, .bpol* to guarantee proper Z output alignment.

OUTPUT.uvwpol determines whether the polarity of the UVW outputs is normal or inverted.

OUTPUT.uvwpol (0x0109 Bit 4)	
Value	Description
0	Normal UVW polarity
1	Inverted UVW polarity

Table 57: UVW Output Polarity

If OUTPUT.uvwen = 0, OUTPUT.uvwpol has no effect.

OUTPUT.uvwdir determines the rotation direction (phase sequence) of the UVW outputs.

OUTPUT.uvwdir (0x0109 Bit 5)	
Value	Description
0	Normal UVW phase sequence
1	Reversed UVW phase sequence

Table 58: UVW Rotation Direction

If OUTPUT.uvwen = 0, OUTPUT.uvwdir has no effect.

OUTPUT.abzen enables or disables the ABZ outputs.

OUTPUT.abzen (0x0109 Bit 6)	
Value	Description
0	ABZ outputs disabled
1	ABZ outputs enabled

Table 59: ABZ Output Enable

See Output Modes, Directions, and Polarities on page 56 for more information.

OUTPUT.abzen also determines the internal update rate of the angle and multi-cycle counter values read via the SPI port. See Response Packet Formats on page 32 for more information.

OUTPUT.uvw enables or disables the UVW outputs.

OUTPUT.uvwen (0x0109 Bit 7)	
Value	Description
0	UVW outputs disabled
1	UVW outputs enabled

Table 60: UVW Output Enable

See Output Modes, Directions, and Polarities on page 56 for more information.

ZPHASE
ZPHASE contains the most significant byte of the Z output phase (Z output location within an input cycle).

ZPHASE.msb (0x010A)	
Value	Description
0 – 255	Z phase MSB (ZPHASE [9:2])

Table 61: Z Phase MSB

UVWPH
UVWPH contains the most significant byte of the UVW output phase shift relative to the sin/cos input cycle.

UVWPH.msb (0x010B)	
Value	Description
0 – 255	UVW phase MSB (UVWPH [9:2])

Table 62: UVW Phase MSB

PHASE_LSB
PHASE_LSB contains the least significant bits of the Z and UVW phase.

PHASE_LSB.z (0x010C Bits 1:0)	
Value	Description
0 – 3	Z phase LS bits (ZPHASE [1:0])

Table 63: Z Phase LSB

The Z output phase, ZPH, is calculated as

$$ZPH = 4 \times ZPHASE.msb + PHASE\_LSB.z$$

The actual Z channel phase, *zphase*, in sin/cos input cycle degrees, is calculated as

$$zphase[^\circ] = ZPH \times \frac{360^\circ}{1024}$$

If Z auto calibration is used, ZPH is tuned by the iC-TW28 during auto calibration.

PHASE_LSB.uvw (0x010C Bits 3:2)	
Value	Description
0 – 3	UVW phase LS bits (UVWPH [1:0])

Table 64: UVW Phase LSB

The UVW phase shift, UVWPS, is calculated as

$$UVWPS = 4 \times UVWPH.msb + PHASE\_LSB.uvw$$

The actual UVW phase, *uvwph*, in sin/cos input cycle degrees, is calculated as

$$uvwph[^\circ] = uvwph \times \frac{360^\circ}{1024}$$

S_ADC
S_ADC contains the MSB of the 12-bit sin channel ADC value.

S_ADC (0x021A)	
Value	Description
0 – 255	Min. – max. sin ADC value

Table 65: Sin ADC Value

In serial configuration mode, this value can be read using the Encoder Link interface to check the rough value of the sin channel ADC for diagnostic purposes. It is updated at an irregular interval of approximately 50 µs. When using the SPI interface, use the dedicated Sin, Cos, and Zero ADC Read command (see page 33).

C_ADC
C_ADC contains the MSB of the 12-bit cos channel ADC value.

C_ADC (0x021C)	
Value	Description
0 – 255	Min. – max. cos ADC value

Table 66: Cos ADC Value

In serial configuration mode, this value can be read using the Encoder Link interface to check the rough value of the cos channel ADC for diagnostic purposes. It is updated at an irregular interval of approximately 50 µs. When using the SPI interface, use the dedicated Sin, Cos, and Zero ADC Read command (see page 33).

Z_ADC
Z_ADC contains the 8-bit Z channel ADC value.

Z_ADC (0x020C)	
Value	Description
0 – 255	Min. – max. Z ADC value

Table 67: Z ADC Value

In serial configuration mode, this value can be read using the Encoder Link interface to check the rough value of the Z channel ADC for diagnostic purposes. It is updated at an irregular interval of approximately 50 µs. When using the SPI interface, use the dedicated Sin, Cos, and Zero ADC Read command (see page 33).

FILT_CFG
FILT_CFG is used to configure the signal path filter. See Filter Configuration on page 70 for more information.

FILT_CFG.auto (0x0300 Bit 0)	
Value	Description
0	Filter kp is static
1	Filter kp is dynamic

Table 68: Dynamic Filtering Enable

FILT_CFG.fb (0x0300 Bit 1)	
Value	Description
0	No feedback loop delay (normal lag)
1	600 ns feedback loop delay (reduced lag)

Table 69: Lag Reduction

FILT_CFG.kpmax (0x0300 Bits 4:2)	
Value	Description
0 – 2	Reserved (do not use)
3 – 6	Minimum - maximum dynamic kp limit
7	Reserved (do not use)

Table 70: Dynamic Filter Limit

FILT_LAG
FILT_LAG is used to set the lag threshold of the signal path filter when dynamic kp is used (FILT_CFG.auto = 1). See Filter Configuration on page 70 for more information.

FILT_LAG.threshold (0x0301 Bit 4:0)	
Value	Description
0	Reserved
1 – 31	Minimum – maximum filter lag threshold

Table 71: Dynamic Filtering Adaption Threshold

When dynamic kp is used, if filter lag is greater than the filter lag threshold, kp is reduced to make the filter more responsive. If filter lag is less than the filter lag threshold, kp is increased to increase filter smoothing.

FILT_K
FILT_K is used to set the filter coefficients. See Filter Configuration on page 70 for more information.

FILT_K.kp (0x0302 Bits 2:0)	
Value	Description
0 – 6	Minimum - maximum filter kp
7	Reserved (do not use)

Table 72: Filter P-Coefficient

When dynamic kp is used, FILT_K.kp is dynamic and updated by the iC-TW28 based on sin/cos input acceleration.

FILT_K.ki (0x0302 Bits 5:4)	
Value	Description
0 – 3	Minimum – maximum filter ki

Table 73: Filter I-Coefficient

STAT_CFG
STAT_CFG is used to configure the status/fault monitoring features of the iC-TW28.

STAT_CFG.filter determines how long a status condition must persist before the corresponding bit in the STAT_VAL register is set.

STAT_CFG.filter (0x0400 Bits 2:0)	
Value	Description
0	0 (none)
1	10 µs
2	150 µs
3	2.5 ms
4	40 ms
5 – 7	Reserved (do not use)

Table 74: Status Event Filtering

STAT_CFG.long determines how long xIRQ is active when a configured status condition or fault occurs.

STAT_CFG.long (0x0400 Bit 3)	
Value	Description
0	xIRQ active for duration of condition
1	xIRQ prolonged by 40 ms

Table 75: Interrupt Extension

Prolonging xIRQ is useful when it is used to drive a fault LED to ensure that transient conditions are visible.

STAT_CFG.enc determines whether or not an overflow of the multi-cycle counter activates xIRQ.

STAT_CFG.enc (0x0400 Bit 4)	
Value	Description
0	No interrupt on counter overflow
1	Interrupt on multi-cycle counter overflow

Table 76: Interrupt Enable for Counter Overflow

STAT_CFG.enc does not affect the Multi-Cycle Counter Rollover Occurred (mcr) or Multi-Cycle Counter Rollover Lost (mcrl) bits in the SPI Status Byte. See Multi-Cycle Counter on page 67 for more information.

STAT_CFG.enz determines whether or not a position capture event (as configured in ZERO0.mode) activates xIRQ.

STAT_CFG.enz (0x0400 Bit 5)	
Value	Description
0	No interrupt on position capture
1	Interrupt on position capture

Table 77: Interrupt Enable for Position Capture

STAT_CFG.enz does not affect the Zero Capture Occurred (zc) or Zero Capture Lost (zcl) bits in the SPI Status Byte. See Position Capture on page 69 for more information.

STAT_VAL
STAT_VAL contains bits that indicate the status of the signal path. These bits are active for the duration of the specified condition.

STAT_VAL (0x0404)		
Bit	Name	Description
0	oflow	Signal path overflow
1	falarm	Input frequency alarm
2	laglim	Excessive position lag
3	inclim	Output frequency limited
4	lagfatl	Fatal position lag
5	scamp	Input amplitude out of range
6	adapt	Adaption limit exceeded
7	res	Correction residue threshold exceeded

Table 78: Status Register

STAT_VAL.oflow indicates that the signal path is saturated somewhere, most likely due to ADC overflow. This condition is not fatal, but does result in reduced interpolation accuracy.

STAT_VAL.falarm indicates that the sin/cos input frequency is above the limit set in the FALARM register.

STAT_VAL.laglim indicates that the AB outputs are lagging behind the input by more than 45° of a sin/cos input cycle. In this condition, the filter is disabled, but the AB outputs are still valid. This condition can be avoided by reducing input acceleration or by reducing the FILT_K.kp value.

STAT_VAL.inclim indicates that either the AB output frequency is being limited to the AB output frequency limit set in the ABLIMIT register, or the UVW output frequency is greater than 8.33 MHz. This condition is not fatal and the AB/UVW outputs are still valid, although if it persists, it will eventually cause a fatal lag limit (STAT_VAL.lagfatl) condition.

STAT_VAL.lagfatl indicates a fatal lag condition and the AB and UVW outputs are not valid (unexpected direction reversal). This occurs if a STAT_VAL.inclim condition persists and the AB/UVW output position falls behind the sensor position by one half (180°) of a sin/cos input cycle.

STAT_VAL.scamp indicates that the sin/cos signal amplitude as calculated by $\sqrt{sin^2 + cos^2}$ is outside the limits specified by the amplitude limit registers.

STAT_VAL.scamp		
Sin/Cos Amplitude Register	Amplitude Low Limit Register	Amplitude High Limit Regester
SC_AMP	SC_AMP_LOW	SC_AMP_HIGH

Table 79: Input Amplitude Out of Range

STAT_VAL.adapt indicates that one or more of the correction parameters has deviated from its base value by more than its specified limit.

STAT_VAL.adapt		
Correction Register	Base Register	Limit Register
S_OFS_COR C_OFS_COR	S_OFS_BASE C_OFS_BASE	SC_OFS_LIM
SC_BAL_COR	SC_BAL_BASE	SC_BAL_LIM
SC_PH_COR	SC_PH_BASE	SC_PH_LIM

Table 80: Adaption Limit Exceeded

STAT_VAL.res indicates that one or more of the correction residue values has exceeded its residue threshold.

STAT_VAL.res	
Residue Register	Residue Threshold Register
S_OFS_RES C_OFS_RES	SC_OFS_TH
SC_BAL_RES	SC_BAL_TH
SC_PH_RES	SC_PH_TH
Z_PH_RES	Z_PH_TH

Table 81: Correction Residue Threshold Exceeded

STAT_LATCH
The STAT_LATCH register contains the same status bits as the STAT_VAL register, except that the bits are latched and stay active until cleared.

STAT_LATCH (0x0405)		
Bit	Name	Description
0	oflow	Signal path overflow
1	falarm	Input frequency alarm
2	laglim	Excessive position lag
3	inclim	Output frequency limited
4	lagfatl	Fatal position lag
5	scamp	Input amplitude out of range
6	adapt	Adaption limit exceeded
7	res	Correction residue threshold exceeded

Table 82: Latched Status Register

Latched status bits are cleared by writing 0 to the bit. Writing 1 to a bit does nothing, allowing bits to be cleared individually.

STAT_SEL

STAT_SEL is used to select whether the status value bits in STAT_VAL or the latched status bits in STAT_LATCH are used to generate an interrupt (activate xIRQ).

STAT_SEL (0x0401)		
Bit	Name	Description
0	oflow	Signal path overflow
1	falarm	Input frequency alarm
2	laglim	Excessive position lag
3	inclim	Output frequency limited
4	lagfatl	Fatal position lag
5	scamp	Input amplitude out of range
6	adapt	Adaption limit exceeded
7	res	Correction residue threshold exceeded

Table 83: Latched Status Selection

If a given STAT_SEL bit is zero, the corresponding STAT_VAL condition is used to generate the interrupt. If a given STAT_SEL bit is one, the corresponding STAT_LATCH condition is used to generate the interrupt. See Status and Fault Logic on page 59 for more information.

STAT_IE

The STAT_IE register is used to enable the bits selected by the STAT_SEL register to actually generate an interrupt (activate xIRQ).

STAT_IE (0x0402)		
Bit	Name	Description
0	oflow	Signal path overflow
1	falarm	Input frequency alarm
2	laglim	Excessive position lag
3	inclim	Output frequency limited
4	lagfatl	Fatal position lag
5	scamp	Input amplitude out of range
6	adapt	Adaption limit exceeded
7	res	Correction residue threshold exceeded

Table 84: Interrupt Enable

If a bit in the STAT_IE register is 1, the corresponding bit selected by the STAT_SEL register generates an interrupt when active. If a bit in the STAT_IE register is 0, the corresponding bit selected by the STAT_SEL register does not generate an interrupt when active. See Status and Fault Logic on page 59 for more information.

STAT_HIZ

STAT_HIZ is used to enable the bits selected by the STAT_SEL register to disable the ABZ/UVW outputs.

STAT_HIZ (0x0403)		
Bit	Name	Description
0	oflow	Signal path overflow
1	falarm	Sin/cos input frequency too high
2	laglim	Filter lag limit exceeded
3	inclim	AB or UVW frequency limit exceeded
4	lagfatl	Fatal lag condition
5	scamp	Sin/cos amplitude out of range
6	adapt	Adaption limit exceeded
7	res	Correction residue threshold exceeded

Table 85: Output Disable

If a bit in the STAT_HIZ register is 1, the corresponding bit selected by the STAT_SEL register disables the ABZ/UVW outputs when active. When disabled, the ABZ/UVW outputs are in a high-impedance state. If a bit in the STAT_HIZ register is 0, the corresponding bit selected by the STAT_SEL register does not disable the ABZ/UVW outputs when active. See Status and Fault Logic on page 59 for more information.

STAT_FATAL

STAT_FATAL is a read-only register containing bits that indicate fatal errors.

STAT_FATAL (0x0406)		
Bit	Name	Description
0	eechk	EEPROM checksum error
1	ee2bit	EEPROM read double bit error
2	interr	Internal error
3 – 7		Reserved

Table 86: Fatal Errors

STAT_FATAL.eechk indicates an error in the checksum of the internal EEPROM.

STAT_FATAL.ee2bit indicates a double bit error occurred when reading the internal EEPROM.

STAT_FATAL.interr indicates that a fatal error occurred in the iC-TW28.

Any of these errors will inhibit startup of the iC-TW28 or stop it during operation, requiring a power cycle to reset. Fatal errors activate xIRQ and disable the ABZ and UVW outputs. See Status and Fault Logic on page 59 for more information.

EE_ADDR
EE_ADDR is used to store the internal EEPROM address to read or write using commands 0x13 and 0x14 respectively. Eight user-accessible bytes in the EEPROM are available.

EEPROM Memory Map	
Address	Description
0x00 – 0x03	Device serial number (do not write)
0x04 – 0x3B	Reserved (do not write)
0x3C – 0x3F	User data (read/write)
0x40 – 0xFF	Reserved (do not write)

Table 87: EEPROM Memory Map

See Device Serial Number and User Data on page 64 for more information.

Do not write to EE_ADDR while the EEPROM is being accessed (commands 0x11, 0x14, or 0x17) or the EEPROM may be corrupted.

EE_DATA
EE_DATA is used to store the data read from or written to the internal EEPROM using commands 0x13 and 0x14 respectively. See Device Serial Number and User Data on page 64 for more information.

EE_DATA (0x0602)	
Value	Description
0...255	EEPROM data

Table 88: EEPROM Data

EE_STAT
EE_STAT indicates the validity of the data read from the internal EEPROM.

After an EEPROM read command (0x13), the value of the EE_STAT register indicates the validity of the data in the EE_DATA register.

EE_STAT (0x0603)	
Value	Description
0	No error, EE_DATA valid
1	Single-bit error corrected, EE_DATA valid
2	Double-bit error, EE_DATA invalid
3 – 255	Reserved

Table 89: EEPROM Data Validity Status

See Device Serial Number and User Data on page 64 for more information.

COMMAND

The COMMAND register is used to start or stop the iC-TW28, save the configuration parameters to EEPROM, individually perform any of the auto calibration routines, etc. To execute a command, write the appropriate value to the COMMAND register. When the command has been executed, the COMMAND register is reset to 0x00 by the iC-TW28 and a new command may be sent.

COMMAND (0x4000)	
Value	Description
0x00	Command register ready/idle
0x01	Start/restart interpolation
0x02	Stop interpolation
0x03 – 0x0F	Reserved (do not use)
0x10	Load configuration and COR registers from EEPROM
0x11	Write configuration and COR register values to EEPROM*
0x12	Copy COR values to BASE registers
0x13	Read EEPROM address
0x14	Write EEPROM address*
0x15	Reserved (do not use)
0x16	Load configuration and COR registers from EEPROM and start interpolation
0x17	Write COR register values to EEPROM*
0x18 – 0x1F	Reserved (do not use)
0x20	Stop auto calibration
0x21	Auto calibrate sin and cos parameters
0x22	Auto calibrate Z channel parameters
0x23	Auto calibrate all parameters
0x24 – 0xFF	Reserved (do not use)

Table 90: Command Register

*These commands do nothing if the EEPROM is locked (TEST.we = 0). Unlock the EEPROM (TEST.we = 1) before executing these commands. See TEST on page 40 for more information. Also, do not write to EE_ADDR while these commands are active or the EEPROM may be corrupted.

Command 0x01 starts or restarts the interpolator using the currently loaded configuration values.

Command 0x02 stops the interpolator. When the interpolator is stopped, the ABZ and UVW are in a high impedance state and the LED output is deactivated as if LED_CFG.en = 0. The xIRQ output remains operational.

Command 0x10 loads the configuration and COR registers from the internal EEPROM but does not start the interpolator.

Command 0x11 writes the values of the configuration registers to the internal EEPROM. The value of the LED register and the values of the correction parameter registers (0x4012 - 0x401B) are also written to the EEPROM, but the value of the TEST register is not. This command may take up to 1 second to complete. Do not write to EE_ADDR while this commands is active or the EEPROM may be corrupted.

Command 0x11 does nothing if the EEPROM is locked (TEST.we = 0). Unlock the EEPROM (TEST.we = 1) before executing this command. See TEST on page 40 for more information.

Command 0x12 copies the values in the correction parameter registers to the corresponding base registers as shown below.

Command 0x12	
Correction Register	Base Register
S_OFS_COR (0x4012)	S_OFS_BASE (0x4007)
C_OFS_COR (0x4014)	C_OFS_BASE (0x4008)
SC_BAL_COR (0x4016)	SC_BAL_BASE (0x400B)
SC_PH_COR (0x4019)	SC_PH_BASE (0x400E)

Table 91: Command 0x12

Command 0x13 reads the EEPROM address specified in register EE_ADDR and returns the value in register EE_DATA and the status (validity) of the data in register EE_STAT. The EEPROM read command requires a minimum of 1 ms to load the EE_DATA and EE_STAT registers. See Device Serial Number and User Data on page 64 for more information.

Command 0x14 writes the data in register EE_DATA to the EEPROM address specified in EE_ADDR. The EEPROM write command requires a minimum of 20 ms to complete. See Device Serial Number and User Data on page 64 for more information. Do not write to EE_ADDR while this commands is active or the EEPROM may be corrupted.

Command 0x14 does nothing if the EEPROM is locked (TEST.we = 0). Unlock the EEPROM (TEST.we = 1) before executing this command. See TEST on page 40 for more information.

Command 0x16 loads the configuration and COR registers from the internal EEPROM and starts the interpolator.

Command 0x17 writes the values of the correction parameter registers to the internal EEPROM. Do not write to EE_ADDR while this commands is active or the EEPROM may be corrupted.

Command 0x17 does nothing if the EEPROM is locked (TEST.we = 0). Unlock the EEPROM (TEST.we = 1) before executing this command. See TEST on page 40 for more information.

Command 0x20 stops auto calibration initiated by command 0x21, 0x22, or 0x23.

Command 0x21 initiates auto calibration of the sin/cos offset, gain, balance, and phase correction parameters. This command must be manually terminated after calibration is complete by writing 0x20 to the COMMAND register. Command 0x11 or 0x17 can then be used to store the calibrated values to EEPROM.

Command 0x22 initiates auto calibration of the Z channel offset, gain, and phase correction parameters. This command must be manually terminated after calibration is complete by writing 0x20 to the COMMAND register. Command 0x11 or 0x17 can then be used to store the calibrated values to EEPROM.

Command 0x23 initiates auto calibration of the sin/cos offset, gain, balance, and phase correction parameters, as well as the Z channel offset, gain, and phase correction parameters. This command must be manually terminated after calibration is complete by writing 0x20 to the COMMAND register. Command 0x11 or 0x17 can then be used to store the calibrated values to EEPROM.

START
START is used to set the startup wait time and the startup mode.

START.wait (0x4001 Bits 2:0)	
Value	Startup Wait Time
0	0 ms
1	1 ms
2	3 ms
3	10 ms
4	30 ms
5	100 ms
6	300 ms
7	Reserved (do not use)
Note	Additional waiting time after configuration is complete.

Table 92: Startup Wait Time

START.mode determines how the AB and Z outputs behave at startup.

START.mode (0x4001 Bits 4:3)	
Value	Description
0	Relative startup mode
1	Same phase startup mode
2	Absolute burst startup mode
3	Reserved (do not use)

Table 93: Startup Mode

See Startup Modes on page 58 for more information.

START.nostart determines whether or not interpolation starts automatically at startup.

START.nostart (0x4001 Bit 5)	
Value	Description
0	Start interpolation at startup
1	Do not start interpolation at startup

Table 94: Interpolation Startup

Inhibiting interpolation at startup is useful in hosted applications to allow the host processor to control interpolation startup.

ADAPT_CFG0
ADAPT_CFG0 is one of the two registers used to configure auto adaption.

ADAPT_CFG0 (0x4002)		
Bit	Name	Description
0	scofs	S/C Offset adaption
1	scbal	S/C Balance adaption
2	scph	S/C Phase adaption
3	scofsa	S/C Analog offset adaption
4	scgain	S/C Gain adaption
5	zofs	Z Offset adaption
6	zgain	Z Gain adaption
7	zphase	Z Phase adaption

Table 95: Auto Adaption Configuration

Sin/cos analog offset and gain adaption as well as all Z channel adaption are meant to be used only for initial calibration and not during operation. It is therefore recommended to set ADAPT_CFG0 $\leq$ 7.

ADAPT_CFG1

ADAPT_CFG1 is the other register used to configure auto adaption.

ADAPT_CFG1.p sets the auto adaption rate.

ADAPT_CFG1.p (0x4003 Bits 2:0)	
Value	Description
0	1/32 COR increment per sin/cos input period
1	1/16 x error per sin/cos input period
2	1/8 x error per sin/cos input period
3	1/4 x error per sin/cos input period
4	1/2 x error per sin/cos input period
5	Reserved (do not use)
6	1 COR increment per sin/cos input period
7	Reserved (do not use)

Table 96: Adaption Rate

Higher ADAPT_CFG1.p values result in faster adaption and calibration. Lower ADAPT_CFG1.p values require more input cycles for calibration. If ADAPT_CFG1.p = 6, the associated COR register is incremented or decremented by 1 as required. If ADAPT_CFG1.p = 0, 32 steps in the same direction are required before the associated COR register is incremented or decremented by 1. This provides very slow correction, but is more immune to noise.

ADAPT_CFG1.stop determines whether or not auto adaption continues when the sensor input amplitude is out of range.

ADAPT_CFG1.stop (0x4003 Bit 3)	
Value	Description
0	Auto adaption always active
1	Auto adaption stopped when STAT_VAL.scamp = 1

Table 97: Adaption Fault Mode

ADAPT_CFG1.zcal determines whether or not the Z channel is calibrated when pin xCALIB is active.

ADAPT_CFG1.zcal (0x4003 Bit 4)	
Value	Description
0	Sin/cos calibration only when xCALIB active
1	Sin/cos and Z calibration when xCALIB active

Table 98: Auto Z Calibration

Disable Z calibration (ADAPT_CFG1.zcal = 0) if the ZERO inputs are not used.

ADAPT_CFG1.xcalee determines whether or not the calibrated correction values are stored to the internal EEPROM when pin xCALIB is deactivated.

ADAPT_CFG1.xcalee (0x4003 Bit 5)	
Value	Description
0	Configuration values not stored to EEPROM when xCALIB deactivated
1	Configuration values stored to EEPROM when xCALIB deactivated

Table 99: xCALIB Storage Function

SC_AMP_TARG

SC_AMP_TARG is used to set the desired sensor amplitude $\sqrt{sin^2 + cos^2}$ value (target) for sin/cos gain calibration.

SC_AMP_TARG (0x4004)	
Value	Description
0 – 180	Min. – max. sin/cos amplitude target
150	Recommended sin/cos amplitude target
181 – 255	Reserved (do not use)
Note	The allowed maximum (180) avoids ADC overload conditions. It is reported in the GUI as 100 %.

Table 100: Sin/Cos Amplitude Calibration Target

SC_AMP_LOW

SC_AMP_LOW is used to set the low limit for the sin/-cos amplitude monitor and LED intensity control during operation.

SC_AMP_LOW (0x4005)	
Value	Description
0 – 180	Min. – max. sin/cos amplitude low limit
181 – 255	Reserved (do not use)

Table 101: Sin/Cos Amplitude Low Limit

During operation, STAT_VAL.scamp is active whenever SC_AMP < SC_AMP_LOW. See Sin/Cos Amplitude Monitor on page 62 and LED Intensity Control on page 72 for more information.

SC_AMP_HIGH

SC_AMP_HIGH is used to set the high limit for the sin/-cos amplitude monitor and LED intensity control during operation.

SC_AMP_HIGH (0x4006)	
Value	Description
0 – 180	Min. – max. sin/cos amplitude high limit
181 – 255	Reserved (do not use)

Table 102: Sin/Cos Amplitude High Limit

During operation, STAT_VAL.scamp is active whenever SC_AMP > SC_AMP_HIGH. See Sin/Cos Amplitude Monitor on page 62 and LED Intensity Control on page 72 for more information.

COR Registers
The 10 COR registers contain the error correction parameters. The correction values are determined when auto calibration is performed and are updated during operation by auto adaption as configured in ADAPT_CFG0.

Correction Registers		
Address	Name	Description
0x4012	S_OFS_COR	Sin offset correction
0x4013	S_OFSA_COR	Sin analog offset corr.
0x4014	C_OFS_COR	Cos offset correction
0x4015	C_OFSA_COR	Cos analog offset corr.
0x4016	SC_BAL_COR	Sin/cos balance corr.
0x4017	SC_GN_COR	Sin/cos gain correction
0x4018	SC_GNA_COR	S/C analog gain corr.
0x4019	SC_PH_COR	Sin/cos phase corr.
0x401A	Z_OFSA_COR	Z analog offset corr.
0x401B	Z_GNA_COR	Z analog gain corr.

Table 103: Correction Registers

Values for the COR registers can also be written directly if auto calibration or auto adaption is not used. This is also useful to provide initial values for auto calibration to provide faster calibration.

BASE Registers
The four BASE registers are used to set the base levels of the sin/cos error correction parameters for detection of an excessive adaption status condition (STAT_VAL.adapt).

Base Registers		
Address	Name	Description
0x4007	S_OFS_BASE	Sin offset base value
0x4008	C_OFS_BASE	Cos offset base value
0x400B	SC_BAL_BASE	Sin/cos balance base
0x400E	SC_PH_BASE	Sin/cos phase base

Table 104: Base Registers

Base register values are signed, 2's complement numbers. See Excessive Adaption Detection on page 63 for more information.

LIM Registers
The three LIM registers are used to set the deviation limits of the sin/cos error correction parameters for detection of an excessive adaption status condition (STAT_VAL.adapt). Auto adaption of the corresponding error correction parameter stops when its limit is reached.

Limit Registers		
Address	Name	Description
0x4009	SC_OFS_LIM	Sin/cos offset limit
0x400C	SC_BAL_LIM	Sin/cos balance limit
0x400F	SC_PH_LIM	Sin/cos phase limit

Table 105: Limit Registers

Limit register values are positive integers. See Excessive Adaption Detection on page 63 for more information.

RES Registers
The five RES registers contain the error correction residue (uncorrected error) used for detection of an excessive error status condition (STAT_VAL.res). After auto calibration, all residue values should be zero (or near zero). Auto adaption (as configured in ADAPT_CFG0) keeps the corresponding residue values at or near zero during operation.

Residue Registers		
Address	Name	Description
0x4023	S_OFS_RES	Sin offset residue
0x4024	C_OFS_RES	Cos offset residue
0x4025	SC_BAL_RES	Sin/cos balance residue
0x4026	SC_PH_RES	Sin/cos phase residue
0x4028	Z_PH_RES	Z phase residue

Table 106: Residue Registers

Residue register values are signed, 2's complement numbers. See Excessive Error Detection on page 62 for more information.

TH Registers

The four TH registers contain the error correction residue thresholds used for detection of an excessive error status condition (STAT_VAL.res).

Residue Threshold Registers		
Address	Name	Description
0x400A	SC_OFS_TH	Sin/cos offset threshold
0x400D	SC_BAL_TH	Sin/cos balance threshold
0x4010	SC_PH_TH	Sin/cos phase threshold
0x4011	Z_PH_TH	Z phase threshold

Table 107: Residue Threshold Registers

Residue threshold register values are positive integers. See Excessive Error Detection on page 62 for more information.

WATCHDOG

The WATCHDOG register is continuously updated by the iC-TW28 while it is operating correctly.

WATCHDOG (0x401C)	
Value	Description
0 – 30	iC-TW28 not OK
31	iC-TW28 OK
32 – 255	iC-TW28 not OK

Table 108: Watchdog

Clear the watchdog register by writing 0 to it. After a minimum wait time of 1 ms, it should read 31 (0x1F) if the iC-TW28 is operating correctly. Any other value indicates a serious internal malfunction.

SC_AMP

SC_AMP contains the current sin/cos sensor input amplitude $\sqrt{sin^2 + cos^2}$. It is updated every 500 µs.

SC_AMP (0x4020)	
Value	Description
0 – 255	Minimum – maximum sin/cos amplitude

Table 109: Sin/Cos Amplitude

S_AMP

S_AMP contains the current sin channel amplitude. It is updated once per input cycle, or less at high input speeds.

S_AMP (0x4021)	
Value	Description
0 – 255	Min. – max. sin channel amplitude

Table 110: Sin Amplitude

C_AMP

C_AMP contains the current cos channel amplitude. It is updated once per input cycle, or less at high input speeds.

C_AMP (0x4022)	
Value	Description
0 – 255	Min. – max. cos channel amplitude

Table 111: Cos Amplitude

DEVICE_ID

DEVICE_ID is a read-only register containing the iC-TW28 device (chip) ID for identification purposes.

DEVICE_ID (0xE000)	
Value	Description
0x0C	iC-TW28 Device ID

Table 112: iC-TW28 Device ID

REV0 and REV1

REV0 and REV1 are read-only registers which together contain the iC-TW28 device (chip) revision level.

REV0 (0xE002) and REV1 (0xE003)		
REV1	REV0	Description
0x11	0x14	iC-TW28 X
0x11	0x15	iC-TW28 W and W1
0x04	0x17	iC-TW28 V1

Table 113: iC-TW28 revision

INPUT CONFIGURATION AND SIGNAL LEVELS

The iC-TW28's analog sin/cos inputs accept sensor signals with differential peak amplitudes between 20 mV and 2.0 V in two ranges. It is important to configure the input properly to ensure best performance of the device. Available input configurations are shown in Table 114.

iC-TW28 Input Configurations					
Input Range	Differential Input Amplitude (peak)	Input Loss Detection	Input Impedance	Configuration Mode Pin C3 Level	Configuration Mode Serial MAIN_CFG.input
Low	20 mV – 700 mV	No	>10 MΩ	0, 3, 6, 9	0
		Yes	220 kΩ	—	1
High	65 mV – 2.0 V	Yes	620 kΩ	1, 4, 7, 10	2

Table 114: iC-TW28 Input Configurations

Low input range accepts 20 – 700 mV peak amplitude differential sin/cos signals (10 – 350 mV peak amplitude per + or – input), as shown in Figure 27.

In pin configuration mode, low input range is selected via configuration input C3 level 0, 3, 6, or 9. In serial configuration mode, low input range is selected by setting MAIN_CFG.input = 0 or 1.

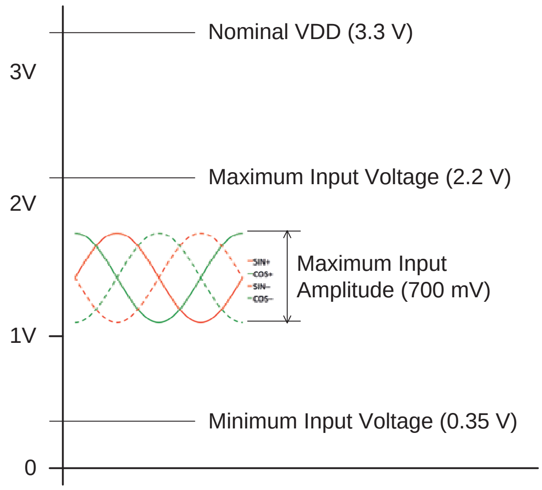


Figure 27: Low Input Range Signals

High input range accepts 65 mV – 2.0 V peak amplitude differential sin/cos signals (32.5 mV – 1.0 V peak amplitude per + or – input) as shown in Figure 28.

In pin configuration mode, high input range is selected via configuration input C3 level 1, 4, 7, or 10. In serial configuration mode, high input range is selected by setting MAIN_CFG.input = 2.

High input range is implemented in the iC-TW28 using a resistive attenuator before the programmable gain amplifier. Thus, input impedance of the sin/cos inputs is different for the two ranges.

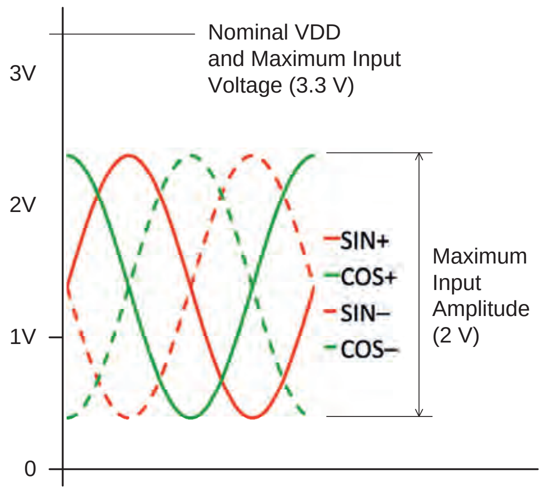


Figure 28: High Input Range Signals

The common-mode (DC) voltage of the individual sensor inputs must not exceed the specified common mode range (spec. item 103) or be such that the peaks of the sin/cos signals exceed the maximum input voltages shown in Figure 27 and Figure 28.

Note: 1 V_{pp} signals with 2.5 V common mode cannot be connected directly to the iC-TW28 as the common mode voltage is too high.

Sensor loss detection is accomplished by internal high value resistors which are placed between the + and – inputs of the sin and cos channels.

In this way, floating inputs are pulled together causing a loss of signal amplitude. This results in a sin/cos amplitude out of range error (STAT_VAL.scamp = 1).

Sensor outputs are typically differential, but can also be single-ended. In this case, the SIN– and COS– inputs must be biased to VDD/2 using an external voltage divider. Do not use VC (pin 10) for this purpose as the sensor input attenuator will draw too much current.

OUTPUT MODES, DIRECTIONS, AND POLARITIES

The iC-TW28 is capable of operating in six different output modes, as shown below.

iC-TW28 Output Modes					
OUTPUT		MAIN_CFG	Output		
.uvwen	.abzen	.rs422	Mode	Format	Driver
0	0	X	SPI Only	—	Hi-Z
0	1	0	ABZ	Differential	CMOS
0	1	1	ABZLD	Differential	RS422
1	0	0	UVW	Differential	CMOS
1	0	1	UVWLD	Differential	RS422
1	1	X	ABZUVW	Single-ended	CMOS

Table 115: iC-TW28 Output Modes

SPI Only
In SPI Only output mode, all six output drivers are disabled and the output pins are in a high impedance state. In this mode, position (angle) must be read via the SPI interface. See SPI Only Output Mode on page 71 for more information.

The ABZ outputs should be tied low as shown in Figure 29 to minimize noise pickup and to avoid accidentally enabling the Encoder Link interface.

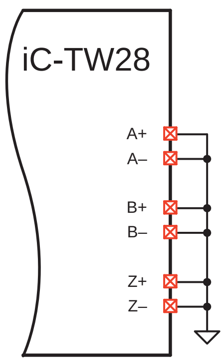


Figure 29: SPI Only Output Connections

SPI Only output mode allows higher sensor input frequencies because the AB output frequency limit (ABLIMIT) is no longer in effect. The input frequency alarm (STAT_VAL.falarm), however, is still in effect in SPI Only output mode. FALARM should be set to 128 to allow maximum input frequency. Other FALARM settings can be used according to the application requirements. STAT_IE.inclim and STAT_HIZ.inclim should be set to 0 to avoid spurious interrupt requests. Finally, **INTER must be set to 0 (interpolation of 256) to allow the multi-cycle counter to be used.**

Position is updated every 20 ns in SPI Only output mode, compared to 320 ns in other modes.

ABZ and ABZLD
In ABZ and ABZLD output modes, the apol, bpol, zpol, and abzdir bits in the OUTPUT register can be used to set the output polarities and directions to match the application requirements. Note that setting apol or bpol (but not both) also reverses the counting direction of the AB outputs. In this case, set the abzdir bit to restore the original AB output counting direction. Setting both apol and bpol does not change the AB output counting direction. See Figure 17 on page 22 for ABZ output connections.

UVW and UVWLD
In UVW and UVWLD output modes, the uvwpol and uvwdir bits in the OUTPUT register can be used to set the output polarity and phase sequence direction of the UVW outputs to match the application requirements. Note that setting uvwpol shifts the phase of the UVW outputs by 180°.

In UVWLD mode, connect the ABZ outputs as shown in Figure 30.

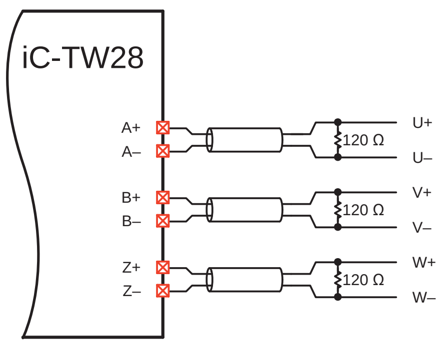


Figure 30: UVW Output Connections

The three signal pairs should be terminated with a 120 Ω resistor at the far (receiving) end of the cable as shown.

In UVW output mode, the connections are the same but termination resistors should not be used.

ABZUVW

In ABZUVW mode, both ABZ and UVW signals are available as single-ended standard CMOS outputs as shown in Figure 31.

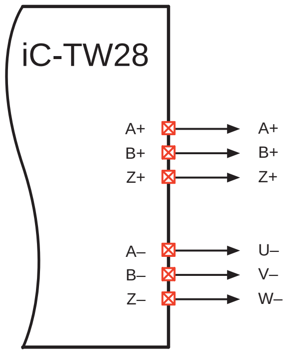


Figure 31: ABZUVW Output Connections

Note that the UVW outputs are inverted; set the uvw-pol bit to provide normal UVW output polarity. The apol, bpol, zpol, abzdir, and uvwdir bits can be used as explained previously to set the desired polarities and directions of the ABZ and UVW outputs.

It is strongly recommended to disable the Encoder Link interface (MAIN_CFG.elinkoff = 1) in ABZUVW mode to avoid accidentally triggering the Encoder Link activation sequence.

STARTUP MODES

In serial configuration mode, the START.mode register value determines how the AB outputs behave at startup. START.mode is only effective if the incremental ABZ outputs are enabled (OUTPUT.abzen = 1). If OUTPUT.abzen = 0, the START.mode value has no effect.

START.mode (0x4001 Bits 4:3)	
Value	Description
0	Relative startup mode
1	Same phase startup mode (always used in pin configuration mode)
2	Absolute burst startup mode
3	Reserved (do not use)

Table 116: Startup Mode

In relative startup mode, the state of the A+ and B+ outputs is always the same after startup, regardless of the sin/cos inputs (sensor angle). In same phase startup mode, the state of the A+ and B+ outputs at a given sensor angle is always the same after every startup. This is similar to the operation of a non-interpolated encoder. Absolute burst startup mode is like same phase, except that the sensor angle within an input cycle is counted out on the A and B outputs at startup.

For example, Figure 32 shows the AB output behavior in the three different startup modes with *inter* = 9, OUTPUT.apol = 0, OUTPUT.bpol = 0, and a sensor angle of 70° with no motion at startup.

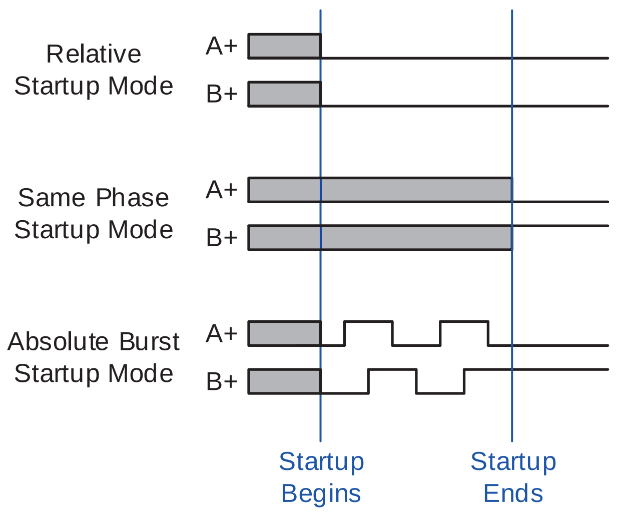


Figure 32: iC-TW28 Startup Modes

Prior to startup, the AB outputs are in a high-impedance state regardless of the startup mode.

In relative startup mode, the A+ and B+ outputs are both 0 after startup because OUTPUT.apol = 0 and OUTPUT.bpol = 0.

In same phase startup mode, the A+ output is low and the B+ output is high after startup since these are the states that correspond to an input sensor angle of 70° with *inter* = 9.

In absolute burst startup mode, the A+ and B+ outputs are both 0 at the beginning of startup (because OUTPUT.apol = 0 and OUTPUT.bpol = 0) and then 7 AB edges are generated, leaving the A+ output low and the B+ output high (the same as in same phase startup mode). With *inter* = 9, each AB edge represents 360°/(9×4) = 10° of the input sin/cos cycle. Thus 70°/10° = 7 AB edges are required to represent the sin/cos sensor startup angle of 70°.

In all startup modes, the actual state of the outputs after startup can be changed using OUTPUT.apol and OUTPUT.bpol. See Output Modes, Directions, and Polarities on page 56 for more information.

Application Hint
Same phase startup mode should only be used if INTER1.div = 0, 1, 3, or 7 (post-AB divider disabled or a power of 2), and *inter* is an integer. If INTER1.div = 2, 4, 5, or 6, or *inter* is not an integer, then the state of the A+, B+, and Z+ outputs will not always be the same after every startup. In addition, the state of the A+ and B+ outputs when the Z+ output is active will not always be the same during operation.

In absolute burst startup mode, a maximum of 2×*inter* AB edges (±180°) may be generated during startup. These edges are output at a rate determined by the value of the ABLIMIT register.

In pin configuration mode, same phase startup mode with OUTPUT.apol = 0 and OUTPUT.bpol = 0 is used, as shown in Figure 32.

STATUS AND FAULT LOGIC

The status and fault logic for a single condition is shown schematically in Figure 33.

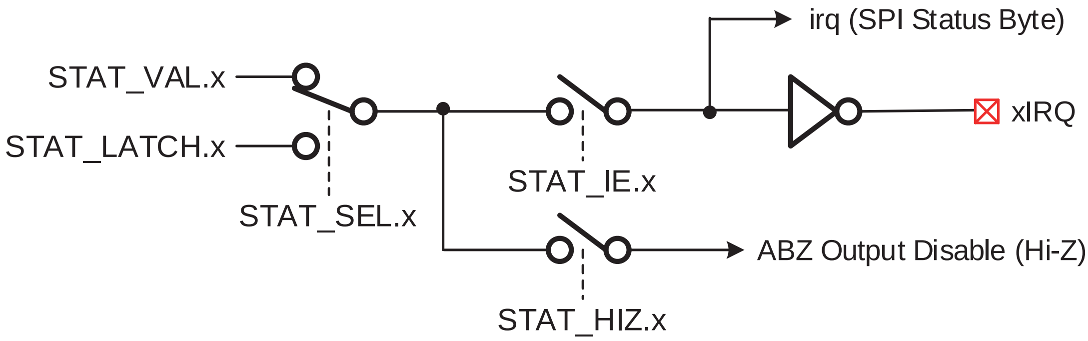


Figure 33: Single Condition Status and Fault Logic

The iC-TW28 continuously monitors 8 internal conditions and reports these status values in the STAT_VAL register. Before activating a bit in the STAT_VAL register, however, the specific condition must have been continuously active for the time specified by STAT_CFG.filter. This filtering avoids nuisance tripping of the status bits. See STAT_VAL on page 47 for more information on the individual status conditions.

The eight status values are also latched into the STAT_LATCH register where they remain active even if the specific condition is no longer active. The host processor or microcontroller can read and then individually clear these latched status bits by writing to the STAT_LATCH register.

Status conditions can also be individually configured to indicate a fault and/or interrupt the host processor or microcontroller when activated. The STAT_SEL register selects whether the dynamic bits in the STAT_VAL register or the latched bits in the STAT_LATCH register are used to generate an interrupt request. In general, select the STAT_VAL register in stand-alone applications (to avoid latching up the fault LED) and STAT_LATCH in hosted applications where the iC-TW28 status is polled (to avoid missing an event).

The STAT_IE register enables the selected status conditions to actually activate the interrupt request output, xIRQ. The internal interrupt request is also available as part of the SPI status byte in the SPI command response packet. See Response Packet Formats on page 32 for more information.

Finally, the STAT_HIZ register provides an independent selection of which of the selected status conditions disable the ABZ outputs by putting them in a high impedance state. Thus each status condition can be individually configured to interrupt the host processor or disable the ABZ outputs, do neither, or do both.

The complete status and fault logic of the iC-TW28 is shown in Figure 34. Bits shown with an "X" indicate that these bits function as switches or gates as shown in Figure 33.

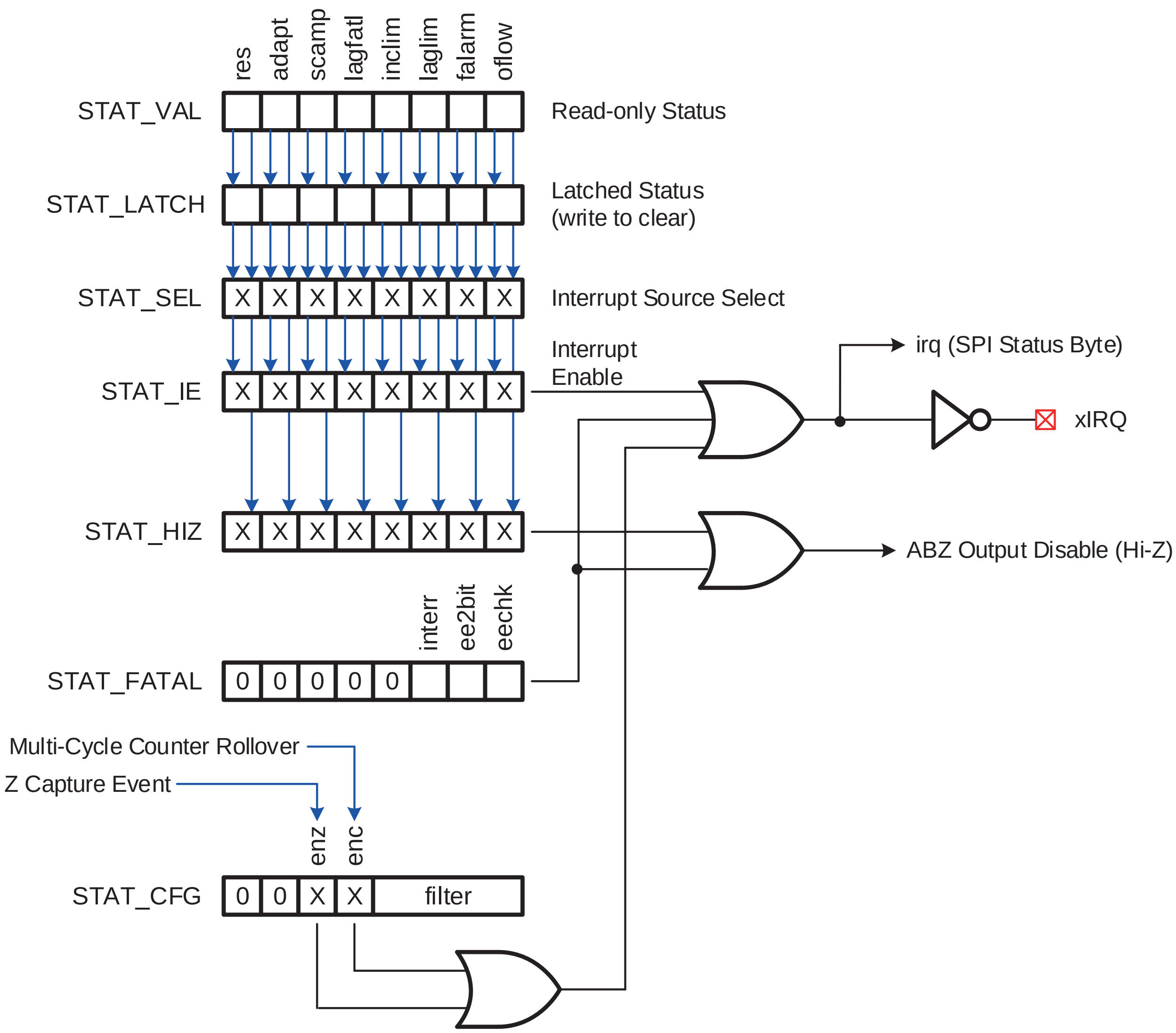


Figure 34: iC-TW28 Status and Fault Logic

The STAT_FATAL register contains three bits that indicate fatal internal conditions (interrupt error, eeprom double-bit error, eeprom checksum error). If any of these conditions are active, the iC-TW28 activates the interrupt request bit (irq), activates the interrupt request output (xIRQ), and disables the ABZ outputs (output shutdown, output tristate).

In stand-alone applications, these fatal fault conditions must be cleared by cycling power to the iC-TW28. In hosted applications, the host processor can reset the iC-TW28 by toggling the reset input (xRST) or by using the start/restart command. See COMMAND on page 50.

A position capture event and/or multi-cycle counter rollover can also be configured to interrupt the host processor. STAT_CFG.enc enables an interrupt when the multi-cycle counter rolls over; STAT_CFG.enz enables an interrupt on a position capture event (as configured by ZERO0.mode).

The status and fault logic of the iC-TW28 in pin configuration mode is shown in Figure 35.

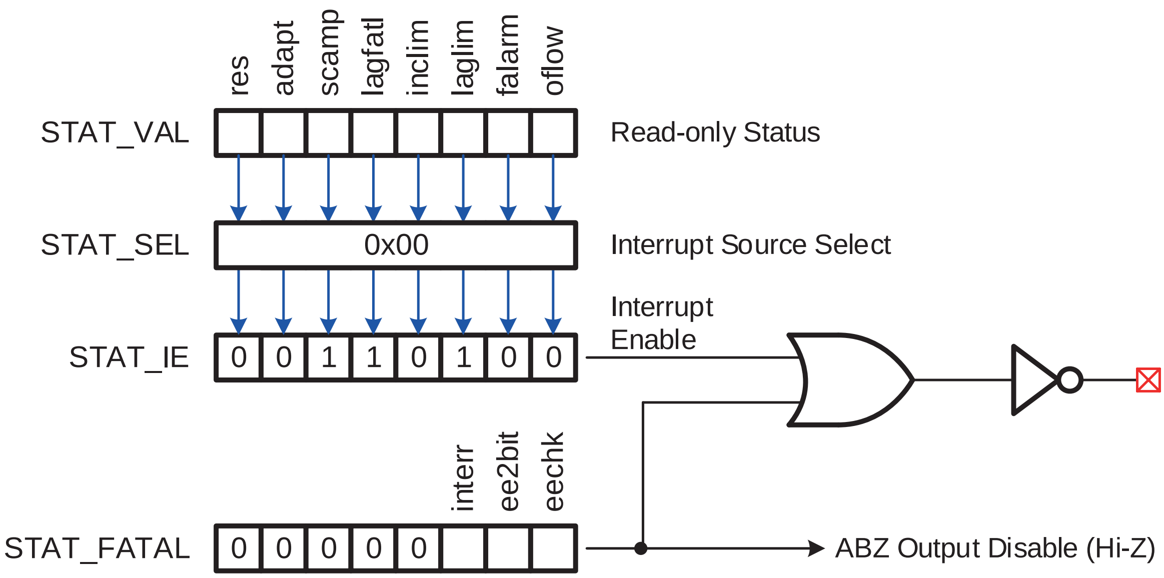


Figure 35: iC-TW28 Pin Configuration Mode Status and Fault Logic

In pin configuration mode, xIRQ is typically used to drive a fault LED. STAT_SEL = 0x00, selecting the STAT_VAL register as the fault source for all conditions. Faults are enabled for scamp, lagfatl, and laglim; any fatal condition also activates xIRQ.

Thus, the fault LED is activated only when the input amplitude is out of bounds or there is a fatal condition. For STAT_HIZ = 0x00 only a fatal condition disables the ABZ outputs.

SIN/COS AMPLITUDE MONITOR

The iC-TW28 continuously monitors the amplitude of the sin/cos input signals by calculating the quantity $\sqrt{sin^2 + cos^2}$. In pin configuration mode, this value is used to activate xIRQ if the input signal amplitude becomes less than 60% or more than 120% of its calibrated value. In serial configuration mode, the amplitude monitor must be specifically configured and calibrated.

To configure the amplitude monitor, enter a value for the desired sin/cos amplitude $\sqrt{sin^2 + cos^2}$ in the SC_AMP_TARG register. During auto calibration, the iC-TW28 adjusts the sin/cos analog and digital gain values to provide this sin/cos level after analog-to-digital conversion. The recommended value is SC_AMP_TARG = 150 (0x96).

Next, enter values for the amplitude limits in the SC_AMP_LOW and SC_AMP_HIGH registers. These values should be the lowest and highest acceptable sin/cos amplitudes respectively. For example, to set the amplitude monitor limits at ±10% with an amplitude target of 150, enter the following values:

$$SC\_AMP\_HIGH = 150 + 10\% = 165$$

$$SC\_AMP\_LOW = 150 - 10\% = 135$$

In operation, the scamp bit (bit 5) in the STAT_VAL and STAT_LATCH registers is set whenever the measured sin/cos amplitude, SC_AMP, is outside these limits:

$$135 > SC\_AMP > 165$$

Configure the desired action to take when this condition occurs using the scamp bit (bit 5) in the STAT_SEL, STAT_IE, and STAT_HIZ registers.

Finally, calibrate the sin/cos amplitude to the target value using auto calibration: activate the xCALIB pin or send an auto calibrate sin/cos (0x21) or auto calibrate all (0x23) command. After calibration, verify that SC_AMP = SC_AMP_TARG.

EXCESSIVE ERROR DETECTION

The iC-TW28 continuously calculates the residual offset, balance, and phase error of the corrected sin/cos signals. These residues represent the uncorrected signal error in the sin and cos channels, and are typically zero when auto adaption is used.

In applications where auto adaption cannot be used, the error residue values allow sensor signal quality to be monitored in a host processor or microcontroller. Programmable threshold values allow activating xIRQ should any of the residue values become excessive. This excessive error condition is indicated by the res bits (bit 7) in the STAT_VAL and STAT_LATCH registers.

STAT_VAL.res	
Residue Register	Residue Threshold Register
S_OFS_RES	SC_OFS_TH
C_OFS_RES	
SC_BAL_RES	SC_BAL_TH
SC_PH_RES	SC_PH_TH
Z_PH_RES	Z_PH_TH

Table 117: Correction Residue Threshold Exceeded

To configure excessive error detection, enter values for the four threshold (TH) registers. These values are the maximum error residue that should be allowed during operation.

In operation, STAT_VAL.res and STAT_LATCH.res are set whenever the absolute value of one of the residues exceeds its corresponding threshold. Specifically, whenever

$$|S\_OFS\_RES| > SC\_OFS\_TH$$

$$|C\_OFS\_RES| > SC\_OFS\_TH$$

$$|SC\_BAL\_RES| > SC\_BAL\_TH$$

$$|SC\_PH\_RES| > SC\_PH\_TH$$

$$|Z\_PH\_RES| > Z\_PH\_TH$$

Configure the desired action to take when this condition occurs using the res bit (bit 7) in the STAT_SEL, STAT_IE, and STAT_HIZ registers. Store all these values to the internal EEPROM using the write configuration to EEPROM command (0x11).

EXCESSIVE ADAPTION DETECTION

In serial configuration mode, the iC-TW28 can be configured to detect when one or more of the error correction parameters changes too much due to auto-adaption during operation. This excessive adaption condition is indicated by the adapt bits (bit 6) in the STAT_VAL and STAT_LATCH registers.

To configure excessive adaption detection, values for the four base registers and three limit registers must be entered and stored in the EEPROM.

STAT_VAL.adapt		
Correction Register	Base Register	Limit Register
S_OFS_COR C_OFS_COR	S_OFS_BASE C_OFS_BASE	SC_OFS_LIM
SC_BAL_COR	SC_BAL_BASE	SC_BAL_LIM
SC_PH_COR	SC_PH_BASE	SC_PH_LIM

Table 118: Adaption Limit Exceeded

After auto calibration has been used to set the nominal error correction values in the correction (COR) registers, use the write COR to BASE command (0x12) to copy these values to the corresponding base registers. Then enter values for the three limit (LIM) registers representing the maximum error correction parameter deviation that should be allowed.

In operation, STAT_VAL.adapt and STAT_LATCH.adapt are set whenever the absolute value of the difference between one of the correction values and its corresponding base value exceeds the corresponding limit. Specifically, whenever

$$|S\_OFS\_COR - S\_OFS\_BASE| > SC\_OFS\_LIM$$

$$|C\_OFS\_COR - C\_OFS\_BASE| > SC\_OFS\_LIM$$

$$|SC\_BAL\_COR - SC\_BAL\_BASE| > SC\_BAL\_LIM$$

$$|SC\_PH\_COR - SC\_PH\_BASE| > SC\_PH\_LIM$$

Configure the desired action to take when this condition occurs using the adapt bit (bit 6) in the STAT_SEL, STAT_IE, and STAT_HIZ registers. Store all these values to the internal EEPROM using the write configuration to EEPROM command (0x11).

DEVICE SERIAL NUMBER AND USER DATA

Each iC-TW28 comes from iC-Haus programmed with a unique serial number. This can be used for tracking devices or when contacting iC-Haus for support.

The device serial number is a four-byte value stored at addresses 0x00 - 0x03 in the internal EEPROM. In serial configuration mode, this value may be read via the SPI port or the Encoder Link interface using the following sequence:

1. Write 0x00 (first address of four-byte serial number) to EE_ADDR (0x0601).
2. Write 0x13 (EEPROM read command) to COMMAND (0x4000).
3. Wait 1 ms or until COMMAND = 0.
4. Read serial number byte value from EE_DATA (0x0602).
5. Read EE_STAT (0x0603) to determine if EE_DATA is valid. If EE_STAT $\leq$ 1, the value in EE_DATA is valid. If EE_STAT > 1, the value in EE_DATA is not valid. See EE_STAT on page 49 for more information.

Repeat this sequence for all four bytes, incrementing the address written to EE_ADDR in step 1 each time.

Do not write to the serial number bytes 0x00 - 0x03 in the internal EEPROM.

Four additional bytes are available in the internal EEPROM for storing user data. These can be used to store the product model and serial number, manufacturing date, etc.

The four user data bytes are stored at addresses 0x3C - 0x3F in the internal EEPROM. In serial configuration mode, these values can be written via the SPI port or the Encoder Link interface using the following sequence:

1. Write 0x01 to the TEST register (0x000B) to unlock the EEPROM.
2. Write 0x3C (address of first user data byte) to EE_ADDR (0x0601).
3. Write the desired user data byte value to EE_DATA (0x0602).
4. Write 0x14 (EEPROM write command) to COMMAND (0x4000).
5. Wait 20 s or until COMMAND = 0.

Repeat this sequence for all four bytes, incrementing the address written to EE_ADDR in step 1 each time. After writing the user data bytes, lock the EEPROM by writing 0x00 to the TEST register (0z000B) or by cycling the xRST input.

In serial configuration mode, these four user data bytes may be read via the SPI port or the Encoder Link interface using the following sequence:

1. Write 0x3C (address of first user data byte)to EE_ADDR (0x0601).
2. Write 0x13 (EEPROM read command) to COMMAND (0x4000).
3. Wait 1 ms or until COMMAND = 0.
4. Read user data byte value from EE_DATA (0x0602).
5. Read EE_STAT (0x0603) to determine if EE_DATA is valid. If EE_STAT $\leq$ 1, the value in EE_DATA is valid. If EE_STAT > 1, the value in EE_DATA is not valid. See EE_STAT on page 49 for more information.

Repeat this sequence for all four bytes, incrementing the address written to EE_ADDR in step 1 each time.

Z TEST MODE AND CALIBRATION

When a zero or index sensor is used, the Z channel must be properly configured and calibrated to ensure one and only one Z output pulse per input revolution,. Configuration consists of setting the Z channel comparator threshold; calibration is performed automatically using auto calibration. See Calibration Overview on page 27 for more information.

In serial configuration mode, the Z channel comparator threshold is set via ZERO0.threshold and the resulting internal Z gating window can be observed on the B outputs in Z test mode. The width of the Z output pulse is set via ZERO1.zwidth. In pin configuration mode, the Z channel comparator threshold is fixed at ZERO0.threshold = 10 (42% of the amplitude of the ZERO input signal) and ZERO1.zwidth = 0 (Z output is 1 AB edge wide).

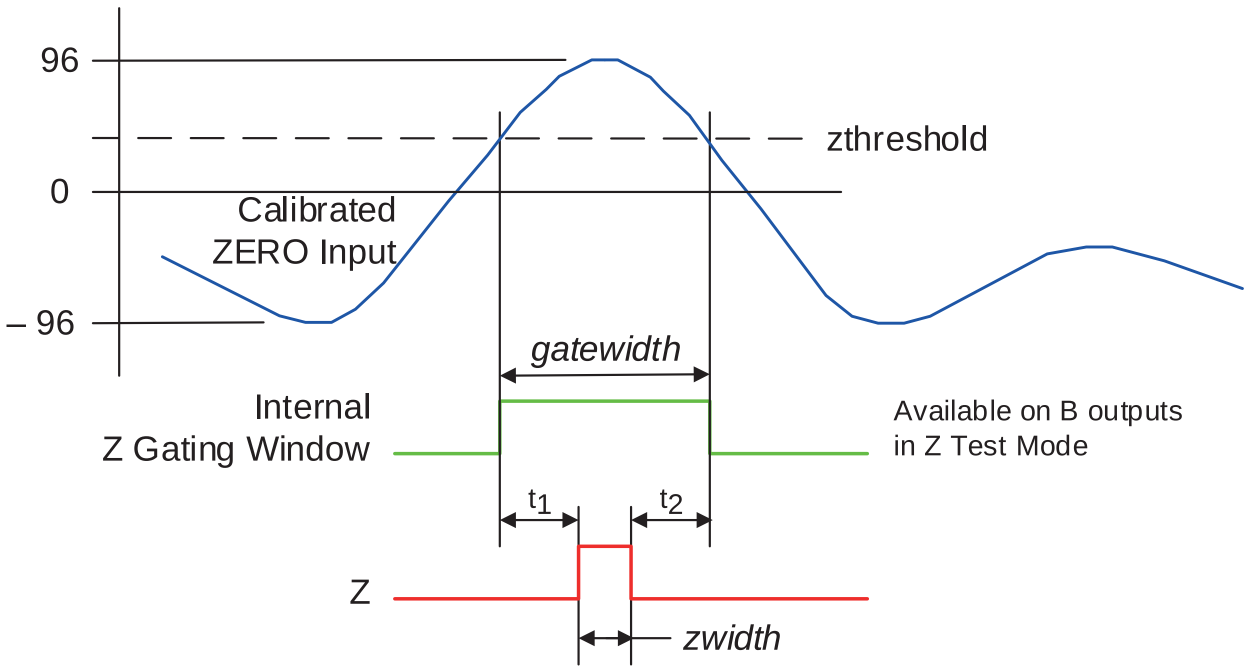


Figure 36: Z Calibration

To configure the Z channel in serial configuration mode, enter values for ZERO0.threshold and ZERO1.zwidth using the SPI port or the Encoder Link interface. The recommended starting values are 10 and 0 respectively – the same values used in pin configuration mode.

Calibrate the Z channel by initiating auto calibration and providing sin/cos and ZERO inputs as explained in Calibration Overview on page 27. During Z auto calibration, the iC-TW28 tunes the Z offset (Z_OFSA_COR) and gain (Z_GNA_COR) so that the calibrated ZERO input signal (blue) spans 75% of the 8-bit Z channel ADC range. This results in a digitized amplitude of 96 as shown in Figure 36.

Auto Z calibration also tunes *zphase* (not shown) to center the Z output pulse in the Z gating window such that $t_1 = t_2$ as shown in Figure 36. This provides the widest possible margin for changes in the width of the Z gating window *(gatewidth)* during operation due to temperature and other operating conditions. See PHASE_LSB on page 44 for more information on *zphase*. When calibration is complete, terminate auto calibration as explained in Calibration Overview on page 27.

As shown in Figure 36, the internal Z gating window is high when the calibrated ZERO input signal is above *zthreshold* and low otherwise.

$$zthreshold = 4 \times ZERO0.threshold$$

The actual threshold in percent of the amplitude of the ZERO signal is

$$\frac{4 \times ZERO0.threshold}{96} \times 100\%$$

Thus, if ZERO0.threshold = 10, *zthreshold* = 40 and the actual Z threshold is at 42% of the ZERO signal.

zthreshold – along with the form of the ZERO input signal – determines the width of the internal Z gating window *(gatewidth)*. To guarantee one and only one Z output pulse per revolution, *gatewidth* must be at least as wide as the desired width of the Z output pulse and no longer than two input cycles minus the width of the Z output pulse under all operating conditions as shown in Figure 37.

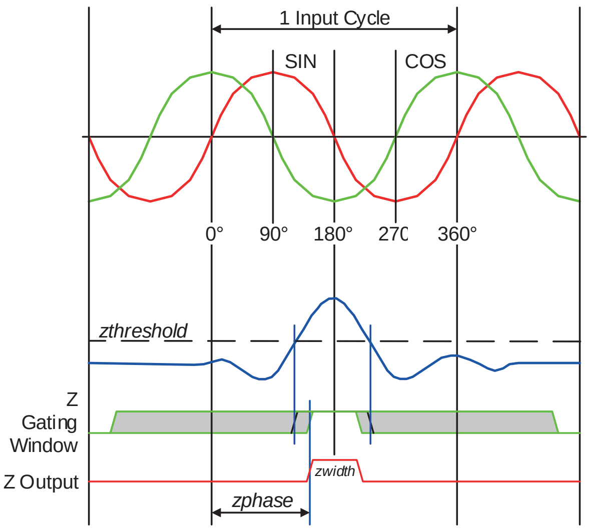


Figure 37: Z Gating Window Requirements

The Z gating window transitions may occur anywhere in the gray area shown. Specifically,

$$zwidth[°] < gatewidth° < 720° - zwidth[°]$$

where *zwidth*[°] is the width of the Z output pulse and *gatewidth*[°] is the width of the internal Z gating window, both in input cycle degrees. Note that the width of the Z output pulse is configured in AB edges in the iC-TW28 and must be converted to input cycle degrees for use in the formula above.

$$zwidth[°] = \frac{90°}{inter} zwidth$$

To ensure that the internal Z gating window is the proper width, it can be observed on the B outputs using Z test mode. Invoke Z test mode by setting TEST.z = 1 using the SPI port or the Encoder Link interface. Adjust the ZERO0.threshold value for the desired *gatewidth*[°] and then run auto calibration for the Z channel again and verify that the Z output pulse is centered in the Z gating window. It is important to always run Z channel auto calibration after making any changes to ZERO0.threshold to guarantee proper centering of the Z output pulse in the Z gating window. Terminate Z test mode by setting TEST.z = 0. See TEST on page 40 for more information.

When a zero or index sensor is not used and a Z output once per input cycle is required, connect ZERO+ to 3.3 V and ZERO- to ground. This is useful in on-axis applications where one input revolution produces only one input cycle. In this case, the z gating window is always high (*gatewidth* = 360°), automatic Z calibration cannot be used, and *zphase* must be set manually to place the Z output in the desired relationship to the sin and cos inputs. See PHASE_LSB on page 44 for more information on *zphase*.

In same phase or absolute burst startup modes, the state of the AB outputs when the Z output is active is determined by the polarity bits OUTPUT.apol and OUTPUT.bpol. The polarity of the Z output is determined by OUTPUT.zpol. See Startup Modes on page 58 and Output Modes, Directions, and Polarities on page 56 for more information.

MULTI-CYCLE COUNTER

The iC-TW28's 14-bit multi-cycle counter continuously tracks up to 16,383 input cycles during operation. In serial configuration mode, the multi-cycle counter can be read and written via the SPI port and can be configured to reset on the rising edge of the Z output. The multi-cycle counter cannot be read using the Encoder Link interface.

To use the multi-cycle counter, select whether or not the multi-cycle counter is to be reset whenever the Z output is activated using ZERO0.clr. If ZERO0.clr = 0, the multi-cycle counter is never cleared. If ZERO0.clr = 1, the multi-cycle counter is cleared (set to zero) whenever the Z outputs are activated. Clearing the counter on the Z output is useful to ensure that multi-cycle counter never rolls over and is always in sync with the external system.

The multi-cycle counter value is a 14-bit number representing the number of input cycles seen by the iC-TW28 since it was started or restarted or since the multi-cycle counter was reset. The multi-cycle counter value and the 10-bit interpolated angle within an input cycle are always read together as a 24-bit position value. See Response Packet Formats on page 32 for more information.

The multi-cycle counter rollover occurred (mcr) bit in the SPI status byte is set whenever the multi-cycle counter passes through a multiple of 4,096 cycles. For example, with continuous positive rotation, the mcr bit is set when the counter passes through values of 4,096, 8,192, 12,288, and 0. Hysteresis of 4,096 cycles is employed to avoid setting the mcr bit multiple times due to direction reversals. For example, after the mcr bit is set at 4,096 cycles with positive rotation, if the direction of rotation reverses, mcr will not be set again passing through 4,096, but will be set again when passing through 0. The multi-cycle counter rollover occurred bit is reset whenever the position is read.

If multiple rollovers occur before the position is read, the multi-cycle counter rollover lost (mcrl) bit is set. This indicates that a previous multi-cycle counter rollover was not acknowledged. This bit is reset whenever the position is read.

The host microprocessor or microcontroller can poll the multi-cycle counter rollover occurred (mcr) bit in the SPI status byte to determine when a counter rollover has occurred. A multi-cycle counter rollover can also be configured to interrupt the host processor by asserting xIRQ. See STAT_CFG on page 46 for more information.

The multi-cycle counter can be preset by writing a value to it. This is useful to synchronize the multi-cycle counter with an external absolute system, for example. Write the new value for the multi-cycle counter and the multi-cycle counter synchronization bit using the multi-cycle counter write command via the SPI port. See Multi-Cycle Counter Write on page 31 for more information.

The multi-cycle counter synchronization bit is used to ensure proper updating of the multi-cycle counter when the sin/cos inputs are moving or if the external absolute system in misaligned. It indicates in which sector (half period) of an input cycle the sensor angle is expected to be when the multi-turn position is updated. This allows correcting the new multi-turn counter value if the external absolute system in misaligned by up to $\pm 90^\circ$ of an input cycle or if the sin/cos inputs move during the presetting of the multi-cycle counter.

Multi-Cycle Counter Synchronization Bit: mcs	
Value	Description
0	$0^\circ (0) \leq$ Sensor Angle $< 180^\circ$ (511)
1	$180^\circ (0) \leq$ Sensor Angle $< 360^\circ$ (1023)

Table 119: Multi-Cycle Counter Synchronization Bit

If the sin/cos input sector indicated by the multi-cycle counter synchronization bit matches the actual sin/cos input sector, the multi-turn counter is preset to the value in the command. If not, the multi-turn counter is preset to the value in the command ± 1 because the input has moved to the next (or previous) cycle.

Specifically, if mcs = 0 and 270° (768) $\leq$ Sensor Angle $< 360^\circ$ (1023), the multi-cycle counter is preset to the value in the command – 1. If mcs = 1 and 0° (0) $\leq$ Sensor Angle $\leq 90^\circ$ (256), the multi-cycle counter is preset to the value in the command + 1. Otherwise, the multi-cycle counter is preset with the value in the command, as shown in Figure 38.

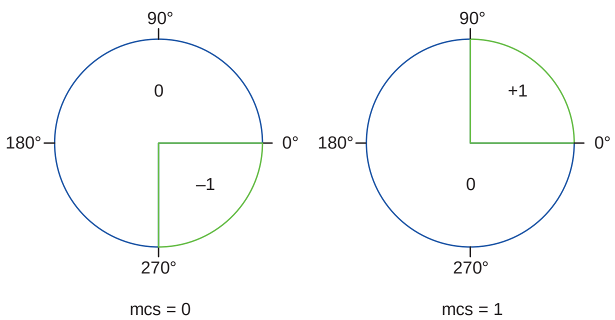


Figure 38: Multi-Cycle Counter Synchronization

For example, if the sensor angle is between 0° and 180° when the multi-cycle counter write command is initiated, then mcs is set to 0 in the command. When the command is executed by the iC-TW28 and the multi-cycle counter is preset, if the sensor angle is between 0° and 270°, it must have moved clockwise across 0° and into the previous input cycle. Thus, the multi-cycle counter is preset to one cycle less than the commanded value (-1).

Therefore, it is necessary to know the sensor angle sector (half-period) prior to executing the multi-cycle counter write command. In the absence of an external absolute system, the sin/cos input sector can be determined by the MSB of the 10-bit sensor angle. An external absolute system must supply its own sector information.

The complete sequence for presetting the multi-cycle counter using the multi-cycle counter write command in the absence of an external absolute system is as follows:

1. Host sends a null write (wm = 0) command with rm = 0 (position and status read) or rm = 3 (register value and angle read) in the SPI control byte.
2. Host sends another command while reading back the response packet.
3. Host sends a multi-cycle counter write command (wm = 1) with the new multi-cycle counter value and mcs = step 2 response packet bit 9.

See SPI Communication on page 29 for more information. If an external absolute system supplies the multi-cycle counter synchronization bit, steps 1 and 2 above are omitted.

The multi-cycle counter atomic read/write command presets the multi-turn counter and reads back the sensor/angle at the same instant to verify that the multi-cycle counter was properly preset.

The complete sequence for presetting the multi-cycle counter using the multi-cycle counter atomic read/write command in the absence of an external absolute system is as follows:

1. Host sends a null write (wm = 0) command with rm = 0 (position and status read) or rm = 3 (register value and position read) in the SPI control byte.
2. Host sends another command while reading back the response packet.
3. Host sends a multi-cycle counter atomic write command (wm = 2) with the new multi-cycle counter value and mcs = step 2 response packet bit 9.
4. Host sends a null write (wm = 0) command with rm = 0 (position and status read) or rm = 3 (register value and position read) in the SPI control byte. The multi-cycle counter is preset at the same instant as the position is read.
5. Host sends another command while reading back the response packet. The returned sensor angle is the angle indicated by the sin/cos inputs when the multi-turn counter was preset.
6. Host verifies that the sin/cos input sector (half-period) when the multi- turn counter was preset was correct. If not, the multi-cycle counter must be preset again.

See SPI Communication on page 29 for more information. If an external absolute system supplies the multi-cycle counter synchronization bit, steps 1 and 2 above are omitted.

POSITION CAPTURE

In serial configuration mode, the full 24-bit position value (10 bits of interpolated angle within an input cycle plus 14 bits of multi-cycle count) of the iC-TW28 can be captured on the rising edge of the Z output or the ZERO input gating window and read out over the SPI port.

Select the desired position capture event using ZERO0.mode. If ZERO0.mode = 0, the position is captured whenever the Z outputs are activated. If ZERO0.mode = 1, it is captured whenever the internal Z gating window is activated. Capturing position on the Z output is useful with distance-coded index quasi-absolute systems or to ensure that the distance (angle) between Z pulses is correct and consistent. Inconsistent distance or angle between successive Z pulses may indicate a fault in the external system. Capturing position on the Z gating window allows the ZERO inputs to be used for touch probe or registration applications.

The position value captured is the most-recently updated internal position when the selected capture event occurs. The position value is updated internally every 320 ns, synchronously with the activation of the Z outputs. Thus, if ZERO0.mode = 1 (position capture on Z gating window active), the captured position may be in error by the distance traveled during the update time. If ZERO0.mode = 0 (position capture on Z outputs active), there is no uncertainly in the captured position.

After the configured capture event has occurred, the captured position can be read via the SPI port. See SPI Communication on page 29 for more information. Captured position cannot be read using the Encoder Link interface.

The zero capture occurred (zc) bit in the SPI status byte indicates that a capture event has occurred and that the captured position is valid. This bit is reset after the captured position is read.

If multiple capture events occur before the captured position is read, the zero capture lost (zcl) bit in the SPI status byte is set. This indicates that the captured position from one or more previous zero capture events has been lost. This bit is also reset after the captured position is read.

The host microprocessor or microcontroller can poll the zero capture (zc) bit in the SPI status byte to determine when a zero capture event has occurred. A zero capture event can also be configured to interrupt the host processor by asserting xIRQ by setting STAT_CFG.enz = 1. See STAT_CFG on page 46 for more information.

Application Hint
The position capture feature of the iC-TW28 does not depend on the interpolation factor *inter* (see page 41).

FILTER CONFIGURATION

The signal path filter is used to reduce noise and jitter in the AB and UVW outputs and the position value read via the SPI port. It can also reduce angle lag at constant speed due to interpolator latency. In pin configuration mode, three filter settings (light, medium, and heavy) are available. In serial configuration mode, the filter must be configured.

The filter is implemented as a PI servo loop with optional feedback path delay as shown below.

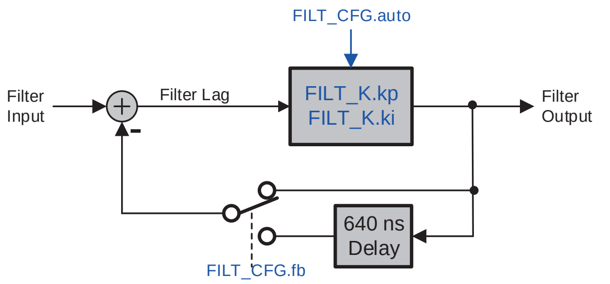


Figure 39: Signal Path Filter

FILT_K.kp determines the proportional (P) gain of the loop, which sets the noise and jitter bandwidth of the filter. Lower FILT_K.kp values provide higher P gain and higher filter bandwidth (less filtering). Higher values provide more filtering and less output noise and jitter.

FILT_K.ki determines the integral (I) gain of the loop, which affects the amount of filter lag under acceleration. Any non-zero FILT_K.ki value also provides zero filter lag at constant sensor input velocity. Lower FILT_K.ki values provide higher I gain and less filter lag under acceleration. However, lower FILT_K.ki values can also produce position overshoot on deceleration. FILT_K.ki = 3 is the recommended value.

Note that FILT_K.kp = 0 with FILT_K.ki = 1 is unstable and therefore should not be used. Also, if FILT_K.ki = 0, FILT_K.kp must also be 0.

FILT_CFG.fb determines whether or not the loop feedback path includes the 640 ns delay or not. This delay is used to reduce the position lag at constant sensor input velocity due to interpolator latency. The intrinsic interpolator latency of the iC-TW28 depends on the value of MAIN_CFG.noise. If FILT_CFG.fb = 1, position lag at constant input velocity due to interpolator latency is reduced by 40% when MAIN_CFG.noise = 0 and by 22% when MAIN_CFG.noise = 1. Note that when feedback delay is used (FILT_CFG.fb = 1), FILT_K.kp $\geq$ 3 is required, otherwise the filter may become unstable.

FILT_CFG.auto determines whether the P gain of the filter is static or dynamic. The normal setting is FILT_CFG.auto = 0, in which case the P gain of the filter loop is static and as set by FILT_K.kp.

In general, filter configuration is a compromise between fast response to sensor input changes and smoothness of the outputs. It is recommended to start with maximum filter bandwidth (least filtering) since this gives the fastest response of the outputs to changes in the sin/-cos inputs. Filtering can then be increased as required to reduce output noise and jitter. Experimentation may be necessary to determine the optimal configuration.

To configure the signal path filter for fastest response (least filtering), use the settings shown below.

Fastest Response Filter Configuration		
Parameter	Value	Description
FILT_K.kp	0	Maximum P gain
FILT_K.ki	3	Nominal I gain
FILT_CFG.fb	0	No feedback path delay
FILT_CFG.auto	0	Static filter kp

Table 120: Fastest Response Filter Configuration

Increase FILT_K.kp as required to reduce output noise and jitter. Enable the feedback path delay (FILT_CFG.fb = 1) as required to reduce the position lag at constant sensor input velocity due to interpolator latency.

The recommended filter configurations that correspond to light, medium, and heavy filtering in pin configuration mode are shown below.

Recommended Filter Configurations				
Filtering	FILT_K.kp	FILT_K.ki	FILT_K.fb	FILT_K.auto
Light	2	3	0	0
Medium	4	3	0	0
Heavy	6	3	0	0

Table 121: Recommended Filter Configurations

SPI ONLY OUTPUT MODE

In SPI Only output mode, the ABZ/UVW outputs are disabled and position (angle) is read via the SPI interface. See SPI Only on page 56 for information on configuring SPI only output mode.

SPI only output mode is useful in embedded applications because the AB output frequency limit (12.5 MHz) is no longer in effect, enabling higher input frequencies to be used. In addition, the integrated multi-cycle counter allow the host processor or microcontroller to sample the position less frequently than would otherwise be necessary while still preserving directional information.

Position is read via the SPI port by sending an SPI command with rm = 0 or 3 in the SPI control byte. See SPI Communication on page 29 for more information. Because of the overlapped packet structure used by the SPI port, it takes two commands to read the position: one command to request the position, and another command to read it back, as shown in Figure 40.

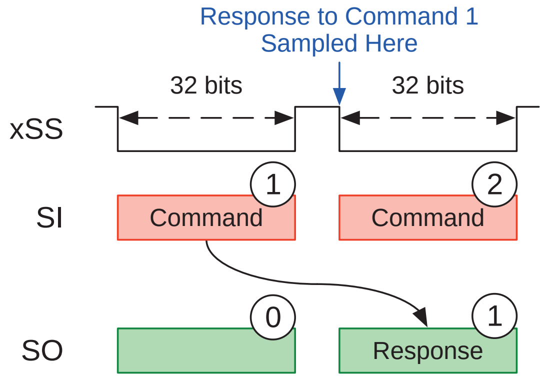


Figure 40: SPI Position Read

Note that the data in response packet 1 is sampled on the falling edge of xSS at the beginning of command packet 2, *not* command packet 1. Therefore, the communication latency in reading position via the SPI port is the length of one 32-bit SPI command, not two.

When multiple iC-TW28's are chained together with a single host microprocessor or microcontroller, the extended communication packet structure provides simultaneous position sampling for all devices. See Chaining Multiple iC-TW28s on page 76 for more information.

In SPI only output mode, the 24-bit position is internally updated every 20 ns. Thus, the position value read via the SPI port may be up to 20 ns old at any given read. Therefore, sequential position values may exhibit a jitter equivalent to the distance (angle) moved by the sin/cos inputs in 20 ns.

In most applications, position needs to be read back by the host microprocessor or microcontroller at a fixed rate. This is most easily accomplished using a sequence of null write (read only) commands with rm = 0. At maximum SPI clock frequency, a new command can be sent – and a new position value read – every 2 µs.

If the sin/cos inputs are moving at 500 kHz, they cover a distance of 1 input cycle per SPI sample.

$$\frac{2\,\mu s}{SPISample} \cdot 500,000 \frac{InputCycles}{Second} = 1.0 cycles$$

The position uncertainty due to the internal position update rate is 0.01 input cycles (3.6°).

$$\frac{20\,ns}{Update} \cdot 500,000 \frac{InputCycles}{Second} = 0.01 cycles$$

Thus the sampled position values will have a jitter of 3.6° or 0.01 · 100% = 1%. Slower input speeds and/or lower SPI sampling rates provide a proportionally lower jitter percentage.

Note that the ADC values are internally updated only every 320 ns. Thus, when reading a sin, cos, and zero ADC read response packet, the jitter is 0.16 cycles (57.6°) or 0.16 · 100% = 16%.

$$\frac{320\,ns}{Update} \cdot 500,000 \frac{InputCycles}{Second} = 0.16 cycles$$

LED INTENSITY CONTROL

In serial configuration mode, the calculated sin/cos amplitude value, SC_AMP, can also be used to drive the LED output to control the intensity of an optical sensor LED. This maintains the sin/cos signals at their calibrated amplitude in the presence of LED ageing and varying application conditions.

To use LED intensity control, enable the LED output by setting LED_CFG.en = 1 and enable LED intensity control by setting LED_CFG.auto = 1. Set LED_START to provice a nominal LED current at startup as explained following. Configure the amplitude monitor as explained in SIN/COS AMPLITUDE MONITOR on page 62 to set the desired amplitude limits for LED control. Set LED_CFG.buffer as required to provide the desired LED intensity hysteresis as explained following.

In operation, the LED intensity control increases the duty cycle of the LED PWM signal whenever

$$SC\_AMP < (SC\_AMP\_LOW + LED\_CFG.buffer)$$

and decreases it whenever

$$SC\_AMP > (SC\_AMP\_HIGH - LED\_CFG.buffer)$$

Select an LED_CFG.buffer value to provide the desired hysteresis for the LED intensity as shown in Figure 41. See LED_CFG.buffer on page 39 for more information on the available buffer values.

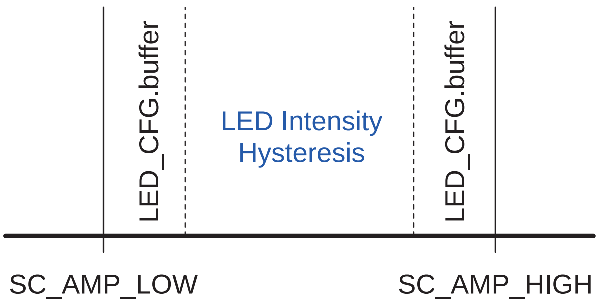


Figure 41: LED Intensity Hysteresis

In operation, the LED intensity control changes the LED PWM value by ±1 at a rate of 500 Hz as required to maintain SC_AMP within the LED intensity hysteresis area shown in Figure 41.

Specifically, the LED intensity hysteresis is

$$SC\_AMP\_HIGH - SC\_AMP\_LOW - 2 \cdot LED\_CFG.buffer$$

For example, if SC_AMP_LOW = 135 and SC_AMP_HIGH = 165, a CFG_LED.buffer value of 0 provides LED intensity hysteresis of

$$165 - 135 - 2 \cdot 8 = 14$$

In this case, LED intensity is controlled so that

$$SC\_AMP = \frac{165 + 135}{2} \pm \frac{14}{2} = 150 \pm 7.$$

Note that the calculated LED intensity hysteresis must be a positive value, otherwise the buffer zones overlap resulting in undefined operation of the LED intensity control. Thus, for proper operation,

$$LED\_CFG.buffer < \frac{SC\_AMP\_HIGH - SC\_AMP\_LOW}{2}$$

Continuing the example above, LED_CFG.buffer = 0 is the only possible choice, since

$$\frac{165 - 135}{2} = 15$$

To use a larger buffer value, SC_AMP_LOW must be decreased, SC_AMP_HIGH must be increased, or both.

If maximum LED current is less than 15 mA under all operating conditions and can be supplied by a 3.3 V supply, connect the LED directly to the iC-TW28 as shown in Figure 42. This is typically the case for red LEDs.

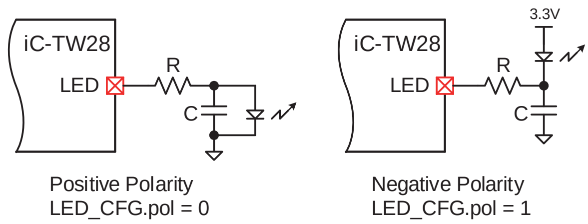


Figure 42: Direct LED Connection

Configure the LED output as open-drain by setting LED_CFG.odrain = 1. Configure the LED polarity to match the specific connection as shown.

If maximum LED current is more than 15 mA or the maximum LED current cannot be supplied by a 3.3 V supply, use an external FET to connect the LED to the iC-TW28 as shown in Figure 43. This is typically the case for blue LEDs due to their higher V_f.

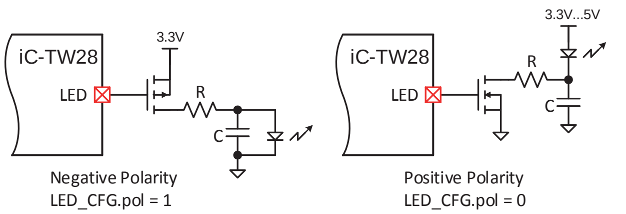


Figure 43: High Current/Voltage LED Connection

Configure the LED output as push-pull by setting LED_CFG.odrain = 0. Configure the LED polarity to match the specific connection as shown.

Select the desired LED PWM frequency using LED_CFG.freq (see LED_CFG.freq on page 39). In general, use the highest PWM frequency (97.6 kHz) to minimize filter capacitor size by selecting LED_CFG.freq = 3. Lower PWM frequencies may be selected if interference is a problem.

Size the current-limiting resistor and filter capacitor using the following formulas:

$$R[k\Omega] = \frac{3.3 - V_f[volts]}{I_{max}[mA]}$$

$$C[\mu F] > \frac{1}{2^{(LED\_CFG.freq+3)} \times R[k\Omega]}$$

Where V_f is the forward voltage drop of the LED and I_{max} is the maximum forward current of the LED or 15 mA, whichever is less. This gives a filter cutoff one decade below the PWM frequency. For example, with the iC-SD85 infrared LED (V_f = 1.4 V and I_{max} = 20 mA), R = 127 Ω and C > 120 nF for LED_CFG.freq = 3.

Once the current-limiting resistor has been chosen to enforce the maximum LED current, LED_START can be calculated to provide the desired nominal LED operating current as

$$LED\_START = INT\left(\frac{I_{nom}}{I_{max}} \cdot 256\right)$$

For example, for a nominal LED current of 8 mA with the iC-SD85,

$$LED\_START = INT\left(\frac{8}{15} \cdot 256\right) = 137$$

Note: As the iC-TW28 runs at 3.3 V, the LED output may not be able to power a blue LED because V_f may exceed 3 V. To operate the LED from 5 V, use an external n-channel FET as shown in Figure 43 (right side). This circuit also avoids an overvoltage at pin LED (V(LED) < AVDD + 0.3 V is required).

POST-AB DIVIDER

The iC-TW28 includes an optional divider after the internal ABZ output generator that can be used to reduce (divide) the configured interpolation by a factor of 1-8. This is useful when the desired output resolution is not an integer multiple of the input resolution.

For example, with an input resolution of 24 sin/cos cycles per revolution, it is impossible to achieve an output resolution of 1,024 AB cycles per revolution (CPR) without the post-AB divider since $(4 \times 1024)/24$ is not an integer. The required interpolation factor in this case is $1024/24 = 42.\overline{6}$. This value can be achieved, however, using an interpolation factor of 128 and a post-AB divider value of 3 since $128/3 = 42.\overline{6}$. Thus, an output resolution of 512 CPR is possible with an input resolution of 24 using *inter* = 128 (INTER(9:0) = 512) and *div* = 3 (INTER1.div = 2). See INTER1 on page 41 for more information.

The post-AB divider can also be used to achieve low value and fractional interpolation factors that are not native to the iC-TW28. The lowest interpolation factor that can be directly programmed is *inter* = 2 (INTER(9:0)) = 8). Lower effective interpolation values *(intereff)* can be achieved as shown below.

intereff				
Value	*inter*	*div*	INTER(9:0)	INTER1.div
0.25	2	8	8	7
0.50	2	4	8	3
0.75	3	4	12	3
1.00	2	2	8	1
1.25	5	4	20	3
1.50	3	2	12	1
1.75	7	4	28	3

Table 122: Effective Interpolation

Note that the maximum AB output frequency *(fab)* is inversely proportional to the post-AB divider value *(div)*. See INTER1 on page 41 and ABLIMIT on page 42 for more information.

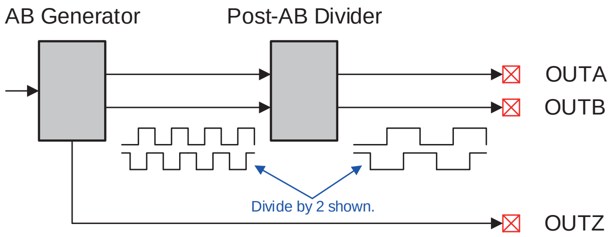


Figure 44: Post-AB Divider

As shown in Figure 44, the Z output bypasses the post-AB divider. Therefore, any configured synchronization of the Z output to the A and B outputs may be lost when using the post-AB divider. See Startup Modes on page 58 for more information. Note also that the actual width of the Z output *(zwidth)* in AB edges is inversely proportional to the post-AB divider value *(div)*. See ZERO1 on page 42 for more information.

BUSSING MULTIPLE iC-TW28s

Multiple iC-TW28 slaves can be used with a single SPI host in a traditional SPI bus connection. In this case, SCLK, SI, and SO on all devices are connected together and each device uses a separate Slave Select (xSS) signal, as shown below.

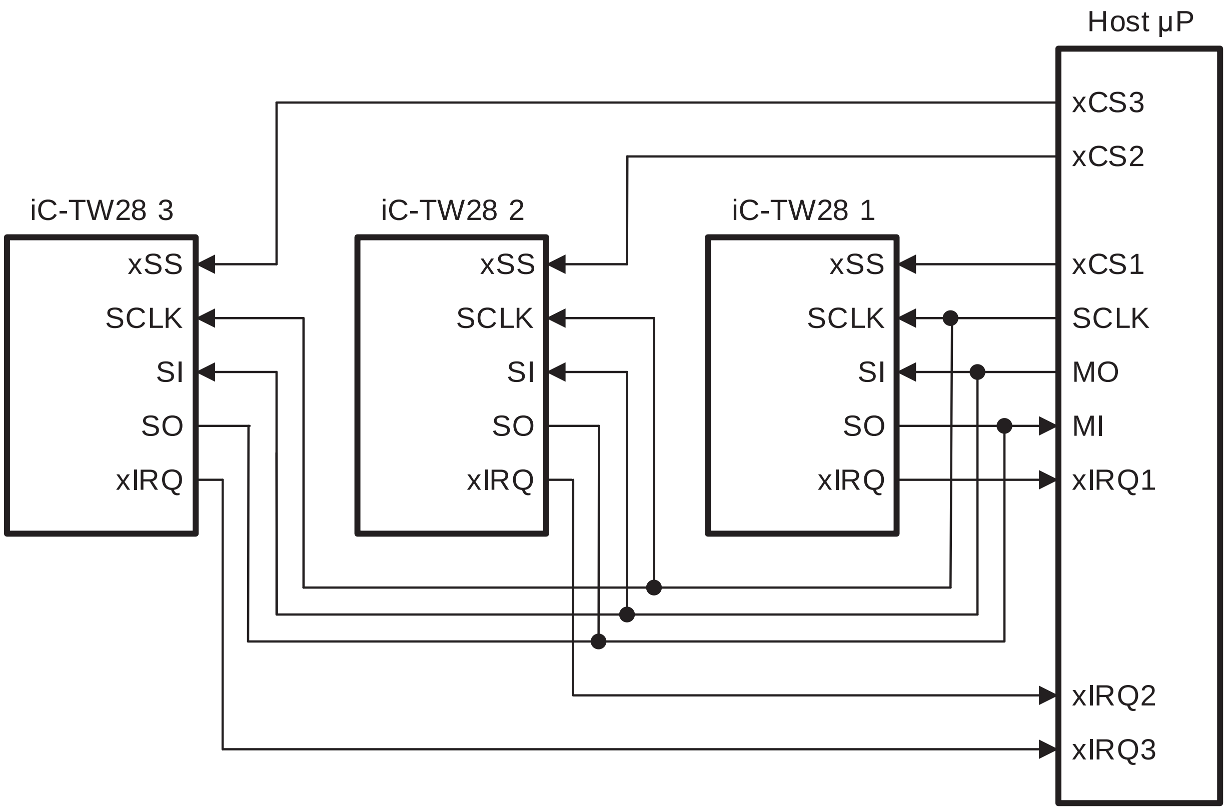


Figure 45: SPI Bus Connection of Multiple iC-TW28s

In operation, the host initiates communication with one of the iC-TW28s by activating the appropriate chip select (xCS) and then clocking a 32-bit SPI command to the Slave Input, SI, while at the same time reading the 32-bit response to the previous command on the Slave Output, SO. This behavior is the same as with a single iC-TW28. Note that with a bussed connection, the host communicates with only one iC-TW28 slave at a time.

As shown, the interrupt request outputs (xIRQ pins) of the bussed iC-TW28s are connected to their own interrupt request input on the host processor. It is recommended to use push-pull xIRQ outputs (MAIN_CFG.irqpp = 1) with bussed iC- TW28s.

CHAINING MULTIPLE iC-TW28s

Multiple iC-TW28 slaves can also be chained together using a single SPI host. In this case, all devices are accessed together as a group and all data is read back together by the host in an extended response packet.

In a chained configuration, SCLK and xSS on all devices are connected together while SI and SO, are linked from one device to the next, as shown below. The xIRQ outputs are configured as open-drain and wire-OR'd together to a common interrupt request input on the host.

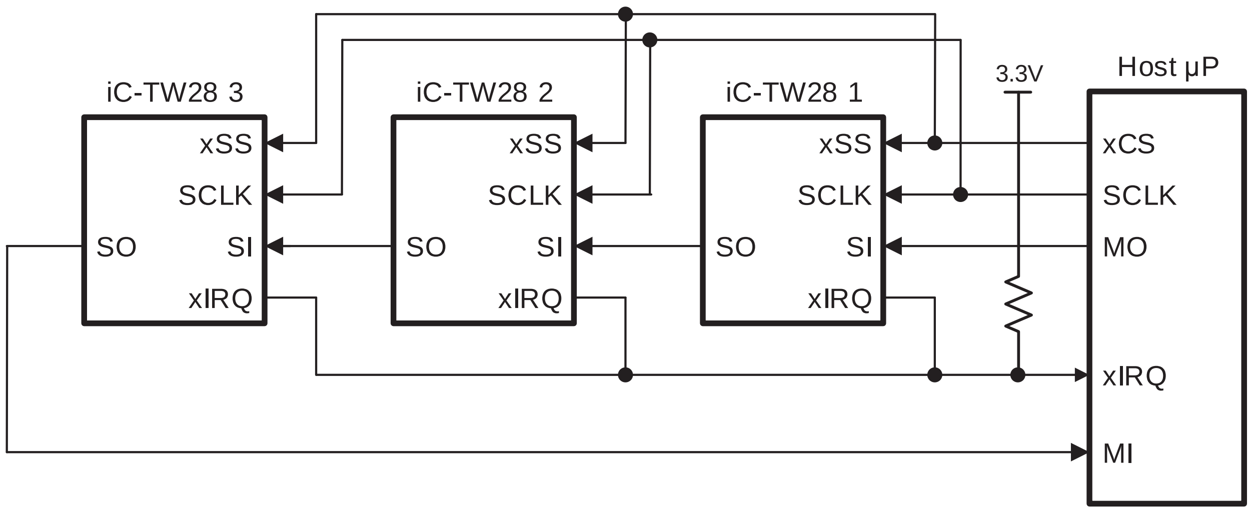


Figure 46: Chained Connection of Multiple iC-TW28s

In operation, the host initiates communication with all of the iC-TW28s by activating the chip select output (xCS) and sending three consecutive SPI commands by clocking $3\times 32 = 96$ bits to the beginning of the SI/SO chain (MO), while at the same time reading the 96-bit response to the previous command from the end of the SI/SO chain (MI). As long as the xSS input of the iC-TW28s is held low, data is shifted through the chained devices from SI to SO. After all commands have been loaded to the chained devices, the host de-activates xSS to execute the commands simultaneously. This extended packet communication structure is shown in Figure 47.

With the interrupt request outputs wire-OR'd as shown in Figure 46, the xIRQ outputs of all iC-TW28s must be configured as open-drain (MAIN_CFG.irqpp = 0) and an external pull-up resistor is required, as shown.

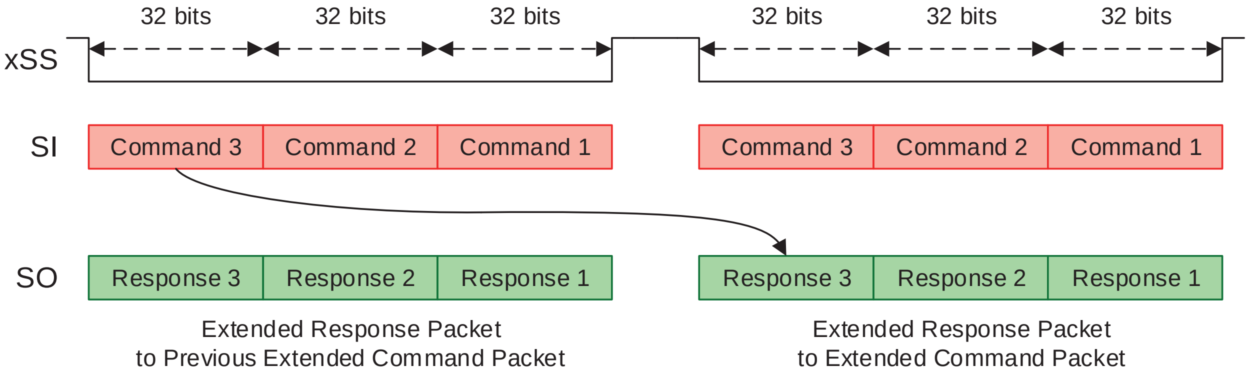


Figure 47: Extended Communication Packet Structure With Chained iC-TW28s

DESIGN REVIEW: Function Notes

iC-TW28 X		
No.	Function, Parameter/Code	Description and Application Notes
		Refer to iC-TW28 datasheet release B1, 2016.

Table 123: Notes on chip functions regarding iC-TW28 chip release X.

iC-TW28 W		
No.	Function, Parameter/Code	Description and Application Notes
1	Power Supply Rise Time	Extending the power-on reset by RC components at pin xRST is recommended for all designs (see Fig. 10 and 11). If pin xRST is directly connected to VDD, VDD rise time (from 0 V to 3.3 V) should be no less than 40 µs.
2	Elec. Characteristics: Items 902, 903	The specified short-circuit current limits can be exceeded at low/high temperatures.

Table 124: Notes on chip functions regarding iC-TW28 chip release W.

iC-TW28 W1		
No.	Function, Parameter/Code	Description and Application Notes
1	Power Supply Rise Time	Extending the power-on reset by RC components at pin xRST is recommended for all designs (see Fig. 10 and 11). If pin xRST is directly connected to VDD, VDD rise time (from 0 V to 3.3 V) should be no less than 40 µs.
2	Elec. Characteristics: Item 501	The permissible number of write cycles is limited to 16 cycles.

Table 125: Notes on chip functions regarding iC-TW28 chip release W1.

iC-TW28 V1		
No.	Function, Parameter/Code	Description and Application Notes
		None at time of printing.

Table 126: Notes on chip functions regarding iC-TW28 chip release V1.

REVISION HISTORY

Rel.	Rel. Date[9]	Chapter	Modification	Page
A1	2015-11-13	All	Initial release	all

Rel.	Rel. Date[9]	Chapter	Modification	Page
B1	2016-03-03		Refer to datasheet release B1.	

Rel.	Rel. Date[9]	Chapter	Modification	Page
C1	2016-07-20		Refer to datasheet release C1.	

Rel.	Rel. Date[9]	Chapter	Modification	Page
D1	2017-04-28		Refer to datasheet release D1.	

Rel.	Rel. Date[9]	Chapter	Modification	Page
D2	2017-10-10		Refer to datasheet release D2.	

Rel.	Rel. Date[9]	Chapter	Modification	Page
D3	2018-04-25		Preliminary marking removed.	all
		ELECTRICAL CHARACT.	Item 901: min limit	9
		CONFIGURATION OVERVIEW	Clarified chip response to exceeding the AB frequency limit.	25
		STARTUP	Correction of wait time up to 514 ms	28
		CONFIGURATION PARAMETERS	Register Map: description updated for type of registers; Table 22, 23: updated for type of register, Z_ADC added; Table 47: added that 0 is not a legal value Table 65, 66: notes added about slow update rates of ADC registers and recommendation to use the dedicated SPI command. Table 67 added on Z_ADC STAT_VAL.lagfatl: Clarified chip response to exceeding the AB frequency limit. Table 96 and description: clarified operation of ADAPT_CFG1.p	36, 37, 42, 44, 45, 47, 52
		STARTUP MODES	Table 116: pin config mode mentioned.	58
		ORDERING INFORMATION	P/O code corrected to QFN32-5x5	80

Rel.	Rel. Date[9]	Chapter	Modification	Page
D4	2019-05-20	OPERATING REQUIREMENTS: Encoder Link Interface	Operating conditions: PINCFG = 0 V added	13
		FUNCTIONAL BLOCK DIAGRAM	Added analog gain range and step size	15
		SPI COMMUNICATION	2 note boxes added	29
		CONFIGURATION PARAMETERS	Note box added on chaning *inter*	41
		Various	Encoder Link not available in pin configuration mode.	15, 24, 26, 27, 28, 34, 36, 44, 45, 64.

Rel.	Rel. Date[9]	Chapter	Modification	Page
D5	2022-11-24	PACKAGING INFORMATION	Footnote 6 added	5
		ELECTRICAL CHARACT.	Item 603: correction of typ. values	9ff
		OPERATING REQUIREMENTS: Encoder Link Interface	Headline and item I105 adapted for 85 °C max.	13
		ELECTRICAL CONNECTIONS	Note box added for xCALIB	20
		CONFIGURATION OVERVIEW	Added note to Tables 6 and 7 that pin-configured values not stored to EEPROM.	26
		CALIBRATION OVERVIEW	Application hint added	27
		CONFIGURATION PARAMETERS	SC_OFS_LIM, SC_OFS_TH, SC_BAL_LIM, SC_BAL_TH, SC_PH_LIM, SC_PH_TH, Z_PH_LIM: value range corrected to 0...127; Table 26: application hint added Table 43 cf: definition of INTER and inter as resolution and factor, application hints added Table 92: correction of waiting times, footnote added Table 100: note added Table 113: chip release V1 added	37, 38, 41, 43, 51, 52, 54
		STARTUP	Application hint added	58
		POSITION CAPTURE	Application hint added	69

[9] Release Date format: *YYYY-MM-DD*

ORDERING INFORMATION

Type	Package	Options	Order Designation
iC-TW28	QFN32, 5 mm x 5 mm		iC-TW28 QFN32-5x5
Evaluation Board	PCB, approx. 68 mm x 102 mm		iC-TW28 EVAL TW28_1D
iC-TW28 GUI		Evaluation software for Windows PC (entry of IC parameters, file storage, and transfer to DUT)	For download link refer to www.ichaus.com/tw28

Please send your purchase orders to our order handling team:

Fax: +49 (0) 61 35 - 92 92 - 692
E-Mail: dispo@ichaus.com

For technical support, information about prices and terms of delivery please contact:

iC-Haus GmbH
Am Kuemmerling 18
D-55294 Bodenheim
GERMANY

Tel.: +49 (0) 61 35 - 92 92 - 0
Fax: +49 (0) 61 35 - 92 92 - 192
Web: https://www.ichaus.com
E-Mail: sales@ichaus.com

Appointed local distributors: https://www.ichaus.com/sales_partners