iC-TW8 16-BIT SIN/COS INTERPOLATOR WITH AUTO-CALIBRATION

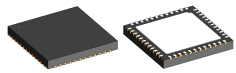
FEATURES

- Input frequency of up to 125 kHz
- Binary/decimal interpolation factors from x0.25 to x16384
- Post-AB divider [1/1 to 1/32] allows fractional resolution
- Differential sine/cosine input signal range of 20 mV to 1.4 V peak-peak
- Automatic offset, gain match and phase correction
- Simple automatic one-pin calibration
- Easy configuration: by static pins (for generic ABZ output)
- Advanced configuration: 1-wire interface, 3 and 4-wire SPI (32 MHz), serial I^2C EEPROM
- PWM or ABZ quadrature encoder output signals
- Incremental ABZ output to 8 MHz (32 MHz edge separation)
- Position and velocity read-out (32 bit SPI)
- Sophisticated error handling and signal monitoring
- Static 64 position LUT to compensate for arbitrary sensor distortions
- Supply voltage range of 3.1 V to 5.5 V
- Extended temperature range of -40 to +125 °C

APPLICATIONS

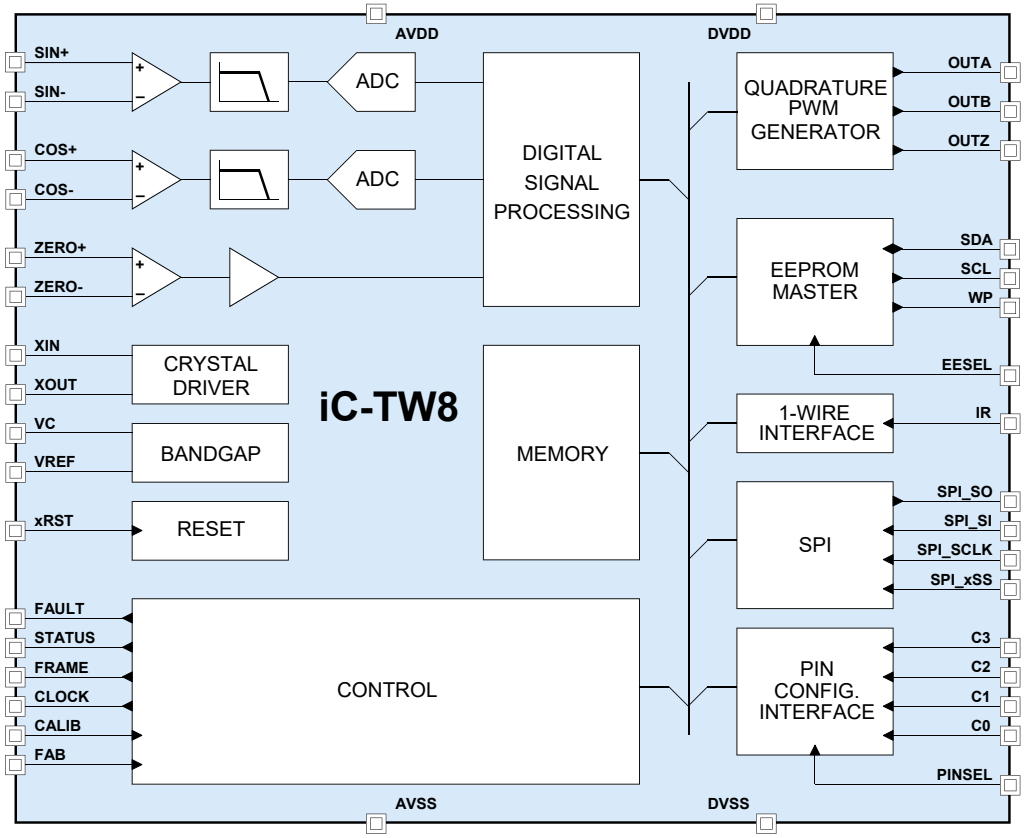
- Sine/cosine interpolation
- Signal conditioning with auto calibration
- Linear and rotary encoders
- Flexible incremental encoder systems

PACKAGES



48-pin QFN
7 mm x 7 mm
RoHS compliant

BLOCK DIAGRAM



DESCRIPTION

The iC-TW8 is a 16-bit application-specific DSP 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 65,536 increments per input cycle. Automatic calibration and adaption (correction during operation) of sensor offset, Sin/Cos amplitude match, and phase quadrature is provided. Additionally, a 64-byte lookup table (LUT) is available to correct for residual harmonic distortion in the sensor signals.

The iC-TW8 accepts 10 mV to 700 mV differential Sin/-Cos input signals directly from magnetic or optical sensors – no external signal conditioning is required in most applications. The differential zero (reference) 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-TW8 provides optional PWM (potentiometer) and serial-only (SPI or 1-wire) output modes for special or imbedded applications. The incremental ABZ quadrature output can be generated at a frequency of up to 8 MHz (31 ns edge spacing); the maximum output frequency can also be limited so as not to overwhelm connected counters.

In PWM output mode, the iC-TW8 provides a differential pulse-width modulated signal proportional to the calculated angle (position) of the Sin/Cos inputs. An RC filter may be used to provide an analog voltage output for potentiometer applications, or to drive an external ADC for simple single-turn absolute applications. Synchronous digital filters may be implemented using an external microcontroller or FPGA for more sophisticated or imbedded applications.

In serial-only output mode, the iC-TW8's internal 32-bit position counter and instantaneous angular velocity register can be read via the built-in SPI or 1-wire serial ports at clock rates of up to 32 MHz. Higher input signal frequencies are allowed in serial-only output mode since the ABZ output frequency limitation is not applicable.

The iC-TW8 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-TW8'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 EEPROM.

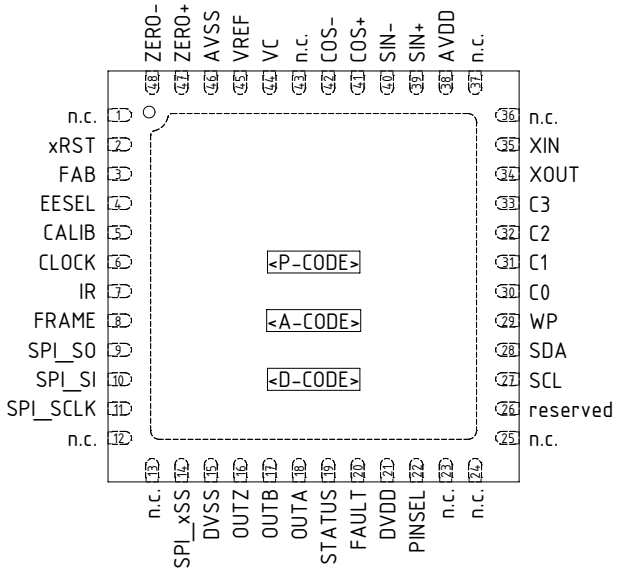
In more sophisticated applications, serial configuration mode allows access to all iC-TW8 features. Complete device configuration using the bi-directional SPI or 1-wire serial ports provides access to all resolutions (including fractional interpolation), the sensor distortion LUT, fully programmable hysteresis, and advanced noise/jitter filtering, quality monitoring, and fault detection capabilities.

The iC-TW8 requires only two external components for operation, a serial EEPROM for storage of configuration and calibration data, and a clock oscillator or crystal. An internal RC oscillator is provided for cost-sensitive applications. Split analog and digital power supply inputs allow optimal filtering for noisy industrial environments. An integrated power-on reset circuit can be overridden by an external hardware reset signal if necessary.

CONTENTS

PACKAGING INFORMATION

PIN CONFIGURATION QFN48-7x7 (topview)

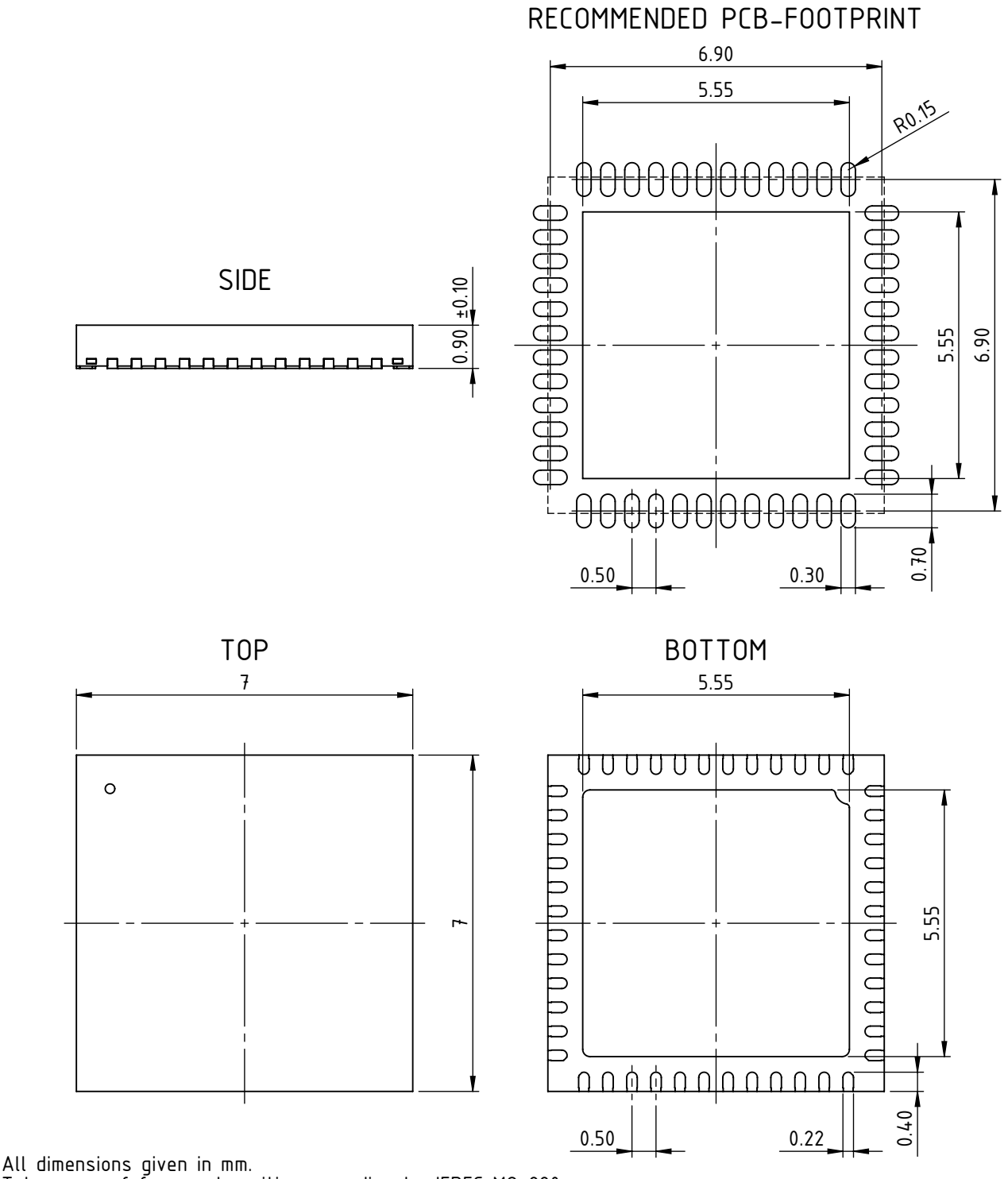


PIN FUNCTIONS

No.	Name	Function
38	AVDD	+3.1 V to 5.5 V Analog Supply Voltage AVDD and DVDD can be powered either mutually or separately but must be the same voltage.
46	AVSS	Analog Ground AVSS must be connected to DVSS externally.
21	DVDD	+3.1 V to 5.5 V Digital Supply Voltage
15	DVSS	Digital Ground
	TP	Connect the backside paddle to a ground plane which must have AVSS potential.
...	other	Refer to the Pin Function table for a detailed description.

IC top marking: <P-CODE> = product code, <A-CODE> = assembly code (subject to changes), <D-CODE> = date code (subject to changes)

PACKAGE DIMENSIONS



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

drb_qfn48-7x7-2_pack_1, 8:1

PIN FUNCTIONS

No.	Name	I/O	Function	Description
1	nc			Pin has no connection to die. Connect to DVSS on PCB.
2	xRST	Digital in, low active, 40 kΩ pull-up	Reset Input	The device is held in reset as long as xRST is low. For operation at 5 V, connect 68 kΩ to DVSS (changes the power-on-reset threshold to approx. 4 V). Refer to section Power-on-Reset and Startup for more information on reset.
3	FAB	Digital in, 5 kΩ pull-down	Test Enable Input	Fabrication test is enabled when pin is high during reset. This pin must be connected to DVSS for normal operation.
4	EESEL	Digital in, 150 kΩ pull-down	Selection Input	Pin must be tied low.
5	CALIB	Digital in, 10 kΩ pull-up, hysteresis	Calibration Control	Device enters calibration mode on falling edge of CALIB. Adaptation parameters are written to the external eeprom on rising edge of CALIB. This pin should be left floating or connected to DVDD if not used.
6	CLOCK	Digital in/out, 150 kΩ pull-down	Clock Output	The pin is a programmable clock output that can be used for PWM synchronization. A connection to DVSS is advisable when not in use.
7	IR	Digital in/out, 150 kΩ pull-up	1-Wire Interface I/O	Pin is bi-directional. Refer to Programmer's Reference for more details. Do not load pin IR by a pull-down. Connecting a pull-up of 1... 4.7 kΩ is recommended for in-field programming using PC adapter MB3U-I2C.
8	FRAME	Digital in/out, 150 kΩ pull-down	Sync Output	This pin outputs the internal ADC sampling clock, which can be used to synchronize downstream circuits. A connection to DVSS is advisable when not in use.
9	SPI_SO	Digital out	SPI Slave Output	Pin directly connects to an SPI master MI pin.
10	SPI_SI	Digital in	SPI Slave Input	Pin directly connects to SPI master MO pin. This pin should be tied to DVSS if the SPI is not used.
11	SPI_SCLK	Digital in, with hysteresis	SPI Slave Clock Input	Pin connects to SPI master clock output. The input implements hysteresis to avoid double triggering. This pin must be tied to DVSS if the SPI is not used.
12	nc			These pins have no connection to die. Connect to DVSS on PCB.
13	nc			
14	SPI_xSS	Digital in, low active	SPI Slave Select Input	In 4-pin SPI mode this pin directly connects to the SPI master slave select output. In case the SPI is operated in 3-pin mode, SPI_xSS must be tied low to DVSS. This pin is not debounced or filtered. A noise-free ground connection is essential to avoid SPI_SO tri-stating during communication. If the SPI is not used, this pin should be tight to DVDD or DVSS.
15	DVSS	Ground	Digital Ground	Pin must tie to high quality ground, usually a solid PCB plane.
16	OUTZ	Digital out	Z Output	Quadrature interface reference output.
17	OUTB	Digital out	B Output PWM- Output Z Window	In quadrature mode this is output B. In PWM mode this is PWM-, the inverted output of OUTA. In Z calibration mode (bit RB_TEST1.z_test = 1) this is the Z window seen just after the input comparator.
18	OUTA	Digital out	A Output PWM+ Output Z Window	In quadrature mode this is output A. In PWM mode this is PWM+. In Z calibration mode (bit RB_TEST1.z_test = 1) this is the Z window used to gate the Z output.
19	STATUS	Digital out	PWM Status Output	This pin provides proportional status information. Pin can drive a 10 mA LED and is widely configurable. Refer to section Monitoring Interpolation Quality for details.
20	FAULT	Digital out	Error Status Output	Pin is low on error and is capable of driving a 10 mA LED. The error response can be configured as detailed in section Fault Handling for details.
21	DVDD	Supply	Digital Power Supply	+3.1 V to +5.5 V supply voltage terminal. DVDD and AVDD must be the same voltage level (5 V or 3.3 V).
22	PINSEL	Digital in, 150 kΩ pull-down	Configuration Mode Selection	Tie pin to DVSS to enable serial configuration mode. Tie pin to DVDD to select pin configuration mode.
23	nc			These pins have no connection to die. Connect to DVSS on PCB.
24	nc			
25	nc			
26	reserved			Connect this pin to DVSS on PCB.

PIN FUNCTIONS

No.	Name	I/O	Function	Description
27	SCL	Digital in/out, 10 kΩ pull-up	EEPROM Clock Line	This pin connects to the EEPROM SCL pin. No external I2C pull-up resistor is required as 10 kΩ is integrated.
28	SDA	Digital in/out, 10 kΩ pull-up	EEPROM Data Line	This pin connects to the EEPROM SDA pin. No external I2C pull-up resistor is required as 10 kΩ is integrated.
29	WP	Digital out	EEPROM Write Protection	This pin acts as the write protect signal and connects to the EEPROM WP pin. No external pull-up is required as this pin is a push-pull output actively driving low and high.
30 31 32 33	C0 C1 C2 C3	Analog in	Configuration Inputs	If the pin configuration mode is used (pin PINSEL tied high), each pin functions as 12-level configuration input. If serial configuration mode is used (pin PINSEL tied low), connect these pins to DVSS on PCB.
34	XOUT	Analog in/out	Crystal Terminal	Pin should be tied to AVSS if no crystal is used. An external oscillator or other square wave clock source can be used to drive this pin. Refer to Providing a Clock.
35	XIN	Analog in	Crystal Terminal	Pin must be tied to AVSS if no crystal is used.
36 37	nc nc			These pins have no connection to die. Connect to DVSS on PCB.
38	AVDD	Supply	Analog Power Supply	+3.1 V to +5.5 V supply voltage terminal. Keep it clean! DVDD and AVDD must be the same voltage level (5 V or 3.3 V).
39 40	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.
41 42	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.
43	nc			Pin has no connection to die. Connect to DVSS on PCB.
44	VC	Analog out	Bias Output	Decouple with 1 µF capacitor to AVSS. Do not inject noise into this pins as it directly impacts ADC conversion noise.
45	VREF	Analog out	Bias Output	Decouple with 1 µF capacitor to AVSS. Do not inject noise into this pin as it directly impacts ADC conversion noise.
46	AVSS	Ground	Analog Ground	Pin must be tied to high quality ground, usually a solid PCB plane.
47 48 ...	ZERO+ ZERO-	Analog in Analog in	Zero Input + Zero Input -	Differential Zero Gating Input. If single-ended signal sources are used, the unused terminal (either ZERO+ or ZERO-) must be tied to an appropriate DC bias. Note that tying both inputs to ground does not work.

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	DVDD, AVDD	Voltage at DVDD, AVDD	referenced to AVSS	-0.3	6	V
G002	dVDD	Difference Voltage between DVDD and AVDD		-0.3	+0.3	V
G003	dVSS	Difference Voltage between DVSS and AVSS		-0.3	+0.3	V
G004	Vpin	Pin Voltage at any pin versus AVSS	referenced to AVSS	-0.3	AVDD + 0.5	V
G005	Ipin	Input Current at any pin, except AVDD, AVSS, DVDD, DVSS		-10	10	mA
G006	Vesd1	ESD Susceptibility Of Signal Input Pins	HBM, 100 pF discharged through 1.5 kΩ; pins SIN+, SIN-, COS+, COS-, ZERO+, ZERO-	2		kV
G007	Vesd2	ESD Susceptibility Of Remaining Pins	HBM, 100 pF discharged through 1.5 kΩ	2		kV
G008	Tj	Junction Temperature		-40	150	°C
G009	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	QFN48 surface mounted to PCB according to JEDEC 51		30		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 = 3.1...5.5 V, Tj = -40...+125 °C, reference point AVSS unless otherwise stated

Item No.	Symbol	Parameter	Conditions	Min.	Typ.	Max.	Unit
Total Device							
001	AVDD, DVDD	Permissible Supply Voltage AVDD, DVDD		3.1		5.5	V
002	IDD	Total Supply Current in AVDD and DVDD	fin = 1 kHz, Increments 4096, Post-AB Divider 1, error-free boot-up from EEPROM; VDD = 5 V, 32 MHz crystal VDD = 3.3 V, 20 MHz crystal		 25 12	 35 20	 mA mA
Signal Inputs and Amplifiers: SIN+, SIN-, COS+, COS-							
101	Vin()	Permissible Input Voltage Range	refer to Figure 1	1.4		AVDD - 1.0	V
102	V_{IN}()	Permissible Input Amplitude	refer to Figure 1			700	mVpp
103	V_{IN}()diff	Permissible Input Amplitude, differential				1.4	Vpp
104	Vos()	Amplifier Input Offset Voltage				±15	mV
105	Iin()	Input Leakage Current				±50	nA
107	OFFcorr	Correctable Input Offset Voltage	as percentage of input signal amplitude; input offset voltage is the sum of sensor offset plus amplifier offset (item 104);	±100			%
108	Acorr	Correctable SIN/COS Amplitude Mismatch	max(Asin, Acos) / min(Asin, Acos), whereas Asin and Acos are the SIN/COS input amplitudes respectively.	1.24			
109	PHIcorr	Correctable SIN/COS Phase Error	(step size 0.052 degree)		±53		deg
110	Rpu()	Pull-Up Resistor at SIN+, COS+	MAIN_CFG.pull = 1		3		MΩ
111	Rpd()	Pull-Down Resistor at SIN-, COS-	MAIN_CFG.pull = 1		3		MΩ
Index Signal Inputs and Amplifier: ZERO+, ZERO-							
201	Vin()	Permissible Input Voltage		0		AVDD	V
202	Vos()	Input Referred Offset Voltage				±15	mV
203	Iin()	Input Leakage Current				±50	nA
Converter Performance							
301	INL	Integral Nonlinearity	refer to Figure 3, 1 Vpp-diff SIN/COS input with compensated offset, gain and phase			0.08	deg
302	DNL	Differential Nonlinearity	refer to Figure 3, 1 Vpp-diff SIN/COS input with compensated offset, gain and phase			0.02	deg
303	N	Output Angle Noise	1 Vpp-diff SIN/COS input, fin = 0 Hz			0.08	deg

305	tp()io	Input-to-Output Latency	fosc = 32 MHz;
			MAIN_FLTR.fb = 0 (lag recovery disabled)
			MAIN_FLTR.fb = 1 (lag recovery enabled)
			
			fosc = 24 MHz;
			MAIN_FLTR.fb = 0 (lag recovery disabled)
			MAIN_FLTR.fb = 1 (lag recovery enabled)
					
					24
					4
					
					32
					5.3
							
							µs
							µs
							
							µs
							µs

Clock: XIN, XOUT							
401	fin()xtl	Permissible External Frequency Input to XOUT	AVDD, DVDD = 5 V AVDD, DVDD = 3.3 V	6 6		32 24	MHz MHz
402	Vin()xtl	Required AC Input to XOUT	With DC levels close to ground or up to VDD.	300			mV
403	fosc	Internal Oscillator Frequency	AVDD, DVDD = 5 V, permissible maximum Tj = 27 °C, MAIN_CLOCK.freq = 0 AVDD, DVDD = 3.3 V, permissible maximum Tj = 27 °C, MAIN_CLOCK.freq = 0	 15 14	 20 18	32 26 24 24	MHz MHz MHz MHz
404	TCosc	Internal Oscillator Temperature Dependancy	AVDD, DVDD = 5 V AVDD, DVDD = 3.3 V		-0.07 -0.09		%/K %/K

ELECTRICAL CHARACTERISTICS

Operating conditions: AVDD = DVDD = 3.1...5.5 V, Tj = -40...+125 °C, reference point AVSS unless otherwise stated

Item No.	Symbol	Parameter	Conditions	Min.	Typ.	Max.	Unit
405	VCosc	Internal Oscillator Power Supply Dependancy			1.18		MHz/V
Reset and Start-Up: xRST							
601	DVDDon	Reset Threshold (at DVDD)	increasing voltage at DVDD; xRST tied to DVDD xRST connected through 68 kΩ to DVSS		 2.7 4.0		 V V
602	DVDD()hys	Reset Threshold Hysteresis			150		mV
603	Vt()hi	Input Logic Threshold High	increasing voltage at xRST			2.7	V
604	Vt()lo	Input Logic Threshold Low		0.8			V
605	Rpu()	Pull-Up Resistor	V() = 0 ... DVDD - 1 V		40		kΩ
606	tstart	Startup Time: Pin Configuration	internal clock, EEPROM connected; AVDD, DVDD = 5 V AVDD, DVDD = 3.3 V		 100 107	 140 150	 ms ms
607	tstart	Startup Time: Ready for Serial Configuration	with EEPROM, w/o read errors (MAIN_OUT.start = 0x00); AVDD, DVDD = 5 V AVDD, DVDD = 3.3 V no EEPROM, SDA = AVSS no EEPROM, SDA = AVDD		 250 265 10 100	 340 360	 ms ms ms ms
608	td()lo	Low-Level Duration for Reset Triggering	V(xRST) < Vt(xRST)lo	10			ns
Digital Input Pins: EESEL, CALIB, CLOCK, IR, FRAME, SPI_SI, SPI_SCLK, SPI_xSS, PINSEL, SCL, SDA, FAB							
701	Vt()hi	Input Logic Threshold High	DVDD = 5 V DVDD = 3.3 V			2.2 2.2	V V
702	Vt()lo	Input Logic Threshold Low	DVDD = 5 V DVDD = 3.3 V	0.8 0.8			V V
703	Vt()hi	Input Logic Threshold High CALIB, SPI_SCLK	DVDD = 5 V DVDD = 3.3 V			3.75 2.47	V V
704	Vt()lo	Input Logic Threshold Low CALIB, SPI_SCLK	DVDD = 5 V DVDD = 3.3 V	0.75 0.49			V V
706	Ilk()	Input Leakage Current at SPI_SI, SPI_SCLK, SPI_xSS				±50	nA
707	Rpu()	Pull-Up Resistor at IR			150		kΩ
708	Rpu()	Pull-Up Resistor at CALIB, SCL, SDA			10		kΩ
709	Rpd()	Pull-Down Resistor at EESEL, CLOCK, FRAME, PINSEL			150		kΩ
710	Rpd()	Pull-Down Resistor at FAB			4		kΩ
Digital Output Pins: CLOCK, IR, FRAME, SPI_SO, OUTA, OUTB, OUTZ, STATUS, FAULT, SCL, SDA, WP							
801	Vs()hi	Output Voltage High	pins SCL, SDA excluded; DVDD = 5 V, IOUT = 4 mA DVDD = 3.3 V, IOUT = 4 mA	 4.4 2.7			 V V
802	Vs()lo	Output Voltage Low	DVDD = 5 V, IOUT = -4 mA DVDD = 3.3 V, IOUT = -4 mA			0.45 0.45	V V
803	Idc()max	Permissible Output DC Load	per pin			±10	mA
804	Idcmax	Permissible Total Output DC Load	for all output pins in aggregate			±60	mA
805	tr()	Rise Time	DVDD = 5 V, CL = 50 pF DVDD = 3.3 V, CL = 50 pF	50 30			ns ns
806	tf()	Fall Time	DVDD = 5 V, CL = 50 pF DVDD = 3.3 V, CL = 50 pF	50 30			ns ns
807	twhi	Duty Cycle at OUTA, OUTB	referred to period T, see Fig. 2		50		%
808	tAB	Output Phase OUTA vs. OUTB	referred to period T, see Fig. 2		25		%
809	tMTD	Minimum Edge Distance at OUTA vs. OUTB	fcore = fxtl or fosc, AB_CFG1.div = 0, AB_VTOP = 0, see Fig. 2		1/fcore		

ELECTRICAL CHARACTERISTICS

Operating conditions: AVDD = DVDD = 3.1...5.5 V, Tj = -40...+125 °C, reference point AVSS unless otherwise stated

Item No.	Symbol	Parameter	Conditions	Min.	Typ.	Max.	Unit
Bias Outputs: VC, VREF							
901	VC	Bias Voltage at VC	I(VC) = 0		50		%VDD
902	dVREF	Bias Voltage at VREF versus VC	dVREF = V(VREF) - V(VC); I(VREF) = 0		-1		V
Configuration Inputs: C0, C1, C2, C3							
A01	Ilk()	Input Leakage Current				±1	µA
EEPROM Interface (SDA, SCL, WP): Additional Specifications							
B01	fclk(SCL)	SCL Clock Frequency (I2C)	EEPROM access on power up (at 5 V for typ.) after configuration, fin()xtl (fosc) = 32 MHz after configuration, fin()xtl (fosc) = 24 MHz		32	50 200 150	kHz kHz kHz
B02	tread	Parameter Read Time	duration of I2C activity after xRST lo → hi; AVDD, DVDD = 5 V, PINSEL = lo AVDD, DVDD = 3.3 V, PINSEL = lo AVDD, DVDD = 5 V, PINSEL = high AVDD, DVDD = 3.3 V, PINSEL = high		 217 240 20 22		 ms ms ms ms
B03	tstore	Parameter Storage Time	writing of 150 bytes after CALIB lo → hi; for EEPROM with write time of 5 ms for EEPROM with write time of 2 ms			 770 320	 ms ms

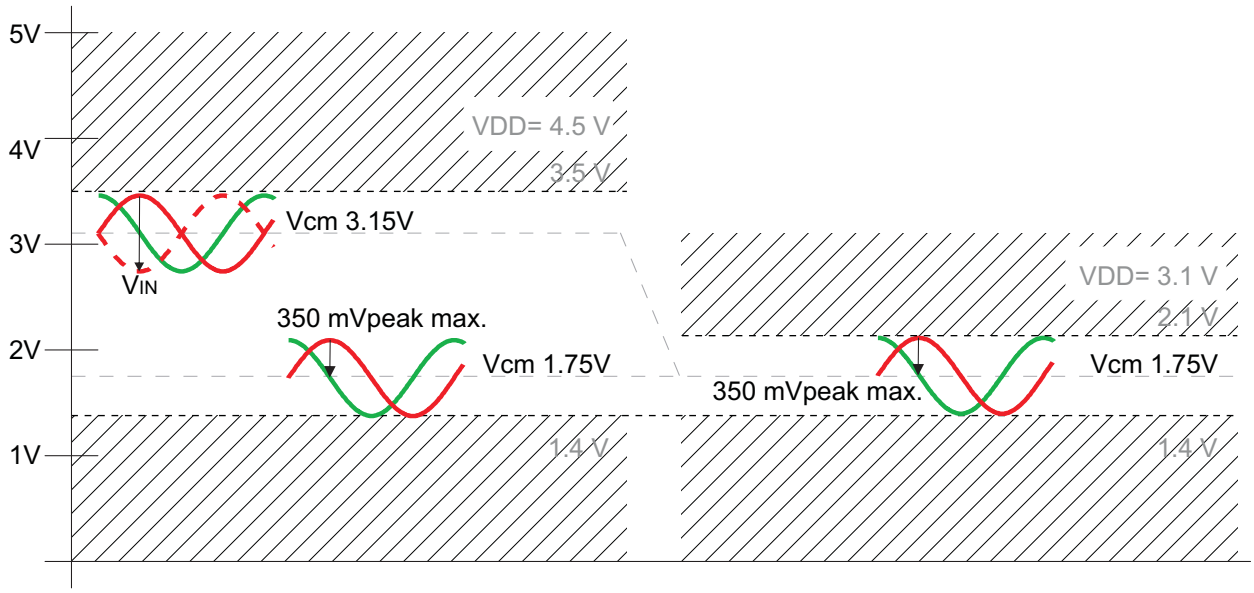


Figure 1: Permissible input voltage range (@5 V -10%, and @3.3 V -5%)

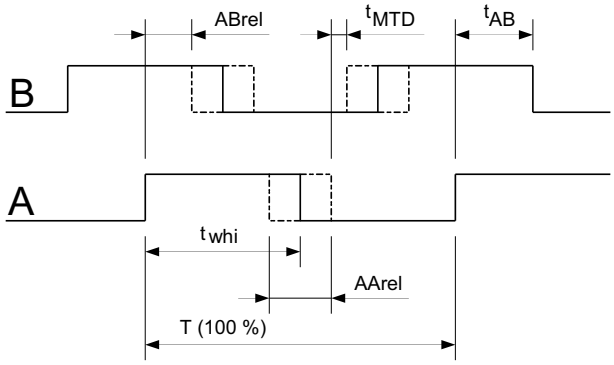


Figure 2: Description of AB output signals.

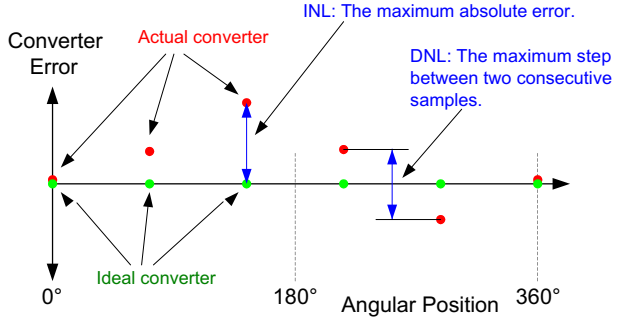


Figure 3: Definition of integral and differential nonlinearity.

OPERATING REQUIREMENTS: SPI Interface

Operating Conditions: AVDD = DVDD = +3.1 ... +5.5 V, AVSS = DVSS = 0 V, Tj = -40...125 °C

Item No.	Symbol	Parameter	Conditions	Min.	Max.	Unit
SPI Interface Timing						
I001	txSS_set	Setup Time: SPI_xSS hi → lo before SPI_SCLK lo → hi		15		ns
I002	txSS_hold	Hold Time: SPI_xSS lo after SPI_SCLK lo → hi		25		ns
I003	txSS_hi	SPI_xSS Signal Duration hi		15		ns
I004	tSI_set	Setup Time: SPI_SI stable before SPI_SCLK lo → hi		5		ns
I005	tSI_hold	Hold Time: SPI_SI stable after SPI_SCLK lo → hi		10		ns
I006	f(SCLK)	SPI_SCLK Clock Frequency	AVDD = DVDD = +4.5 ... +5.5 V, fosc = 32 MHz AVDD = DVDD = +3.1 ... +5.5 V, fosc = 24 MHz		24 20	MHz MHz
I007	tC_hi	SPI_SCLK Signal Duration hi		10		ns
I008	tC_lo	SPI_SCLK Signal Duration lo		10		ns
I009	tSO_p	Propagation Delay: SPI_SO stable after SPI_SCK hi → lo			20	ns
I010	tSO_p2	Propagation Delay: SPI_SO high impedance after SPI_xSS lo → hi			20	ns

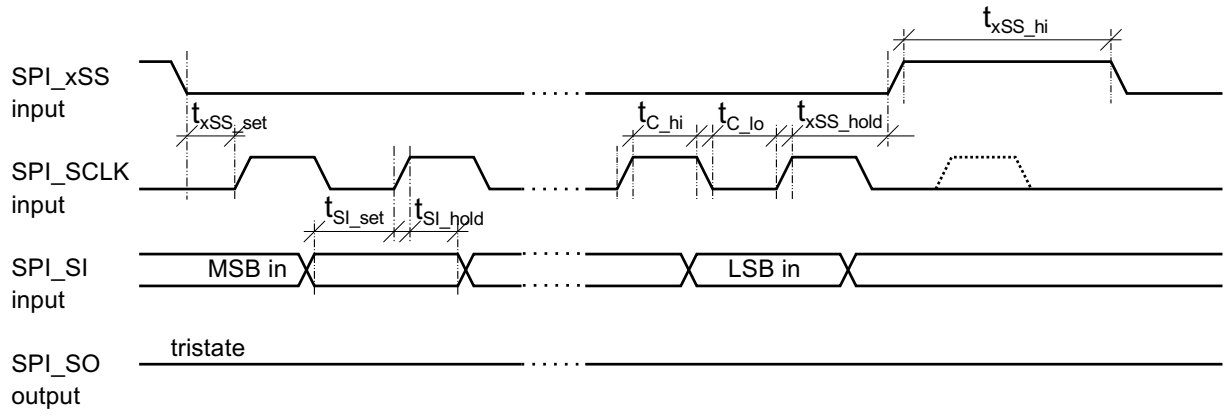


Figure 4: SPI Write Timing.

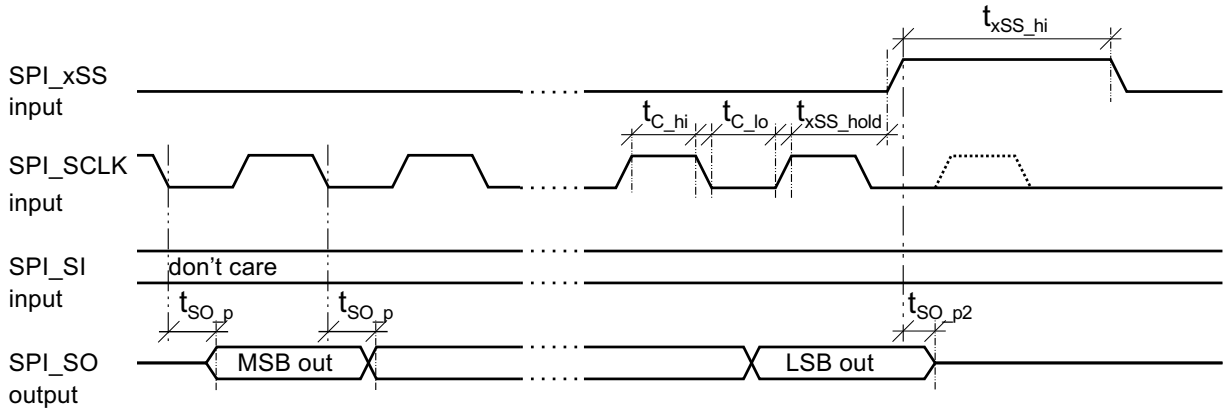


Figure 5: SPI Read Timing.

OPERATING REQUIREMENTS: 1-Wire Interface

Operating Conditions: AVDD = DVDD = +3.1 ... +5.5 V, AVSS = DVSS = 0 V, Tj = -40...125 °C

Item No.	Symbol	Parameter	Conditions	Min.	Max.	Unit
1W-Interface Timing						
I101	t_{start}	Low Time Start Condition		2		ms
I102	t_{long}	Unit Time Long	 unprogrammed EEPROM	50 100	500	µs µs
I103	t_{short}	Unit Time Short	 unprogrammed EEPROM	25 50	t_{long} - 25	µs µs
I104	t_{idle}	Interface Idle Before New Access		100		µs
I105	t_{delay_ic}	Memory Read Delay by IC	iC-TW8 returning data	t_{short} / 2	t_{short}	
I106	t_{long_ic}	Unit Time Long by IC	iC-TW8 returning data	t_{long} / 2	t_{long}	
I107	t_{short_ic}	Unit Time Short by IC	iC-TW8 returning data	t_{short} / 2	t_{short}	

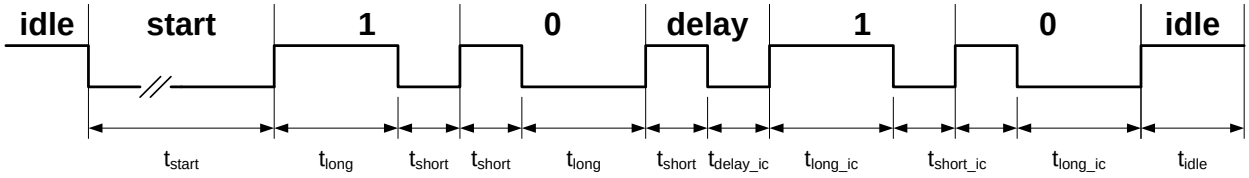


Figure 6: 1-Wire interface timing (PWM bit stream).

PIN CONFIGURATION MODE

Introduction

This section describes the pin configuration mode of the iC-TW8 Interpolator. Pin-configuration mode provides simple, static configuration of the device without any programming or complicated calibration.

Pin configuration allows selection of only the most commonly used features of the device. Access to all features of the iC-TW8 is available in serial configuration mode, described in a separate section.

A design tool for rotary encoders using the iC-TW8 is available. This tool is in the form of an Excel spreadsheet that allows entering application parameters in engineering units and then provides performance limits and graphs, configuration resistor values, and jumper settings for use with the iC-TW8 TW8_1D demonstration board. This greatly simplifies and speeds up the design process of encoders using the iC-TW8.

Functional Overview

The iC-TW8 is a general-purpose 16-bit Sin/Cos interpolator with sophisticated auto-calibration functions. It accepts 20mV to 1.4V peak-to-peak differential analog input signals from magnetic or optical sensors and calculates (interpolates) the angular position. Output is industry-standard incremental AB quadrature at programmable resolution. Simple pin configuration and one-step automatic calibration requires no complicated signal analysis or calibration procedure during product design or production.

In addition to the chosen sensor, the iC-TW8 requires an external EEPROM to store its configuration and calibration data. An external crystal can be used instead of the internal oscillator of the iC-TW8 for superior frequency stability. A differential line driver can be connected directly to the iC-TW8 quadrature outputs to make a complete encoder.

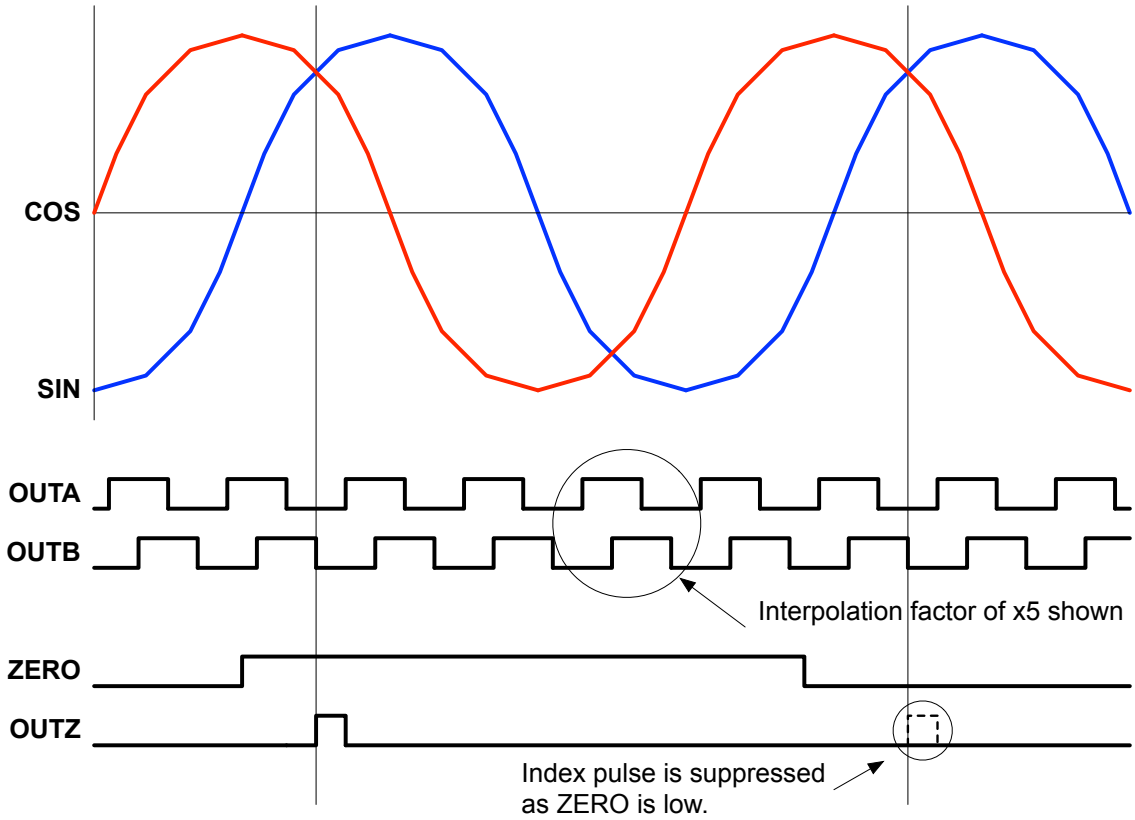


Figure 7: iC-TW8 Functional Overview

Electrical Connections

The basic electrical connections for an encoder using the iC-TW8 are shown in Figure 8. Other than the analog sensor, only an EEPROM for parameter storage, eight configuration resistors, and a few bypass capacitors are required for operation. PINSEL (pin 22) must be connected to DVDD to select pin configuration mode. A voltage regulator to supply clean power to the device and a line driver are recommended. A crystal or oscillator (clock driver) is also recommended.

The iC-TW8 requires a high quality ground and clean power supplies. Analog ground (AVSS = pin 46) and digital ground (DVSS = pin 15) must both be connected on the PCB to a solid ground plane.

Analog power (AVDD = pin 38) and digital power (DVDD = pin 21) inputs must be connected to a low impedance 5V or 3.3V power source, preferably an on-board voltage regulator. In addition, AVDD and DVDD should each have a dedicated 1µF decoupling capacitor placed as close to the power pins on the device as possible.

If the iC-TW8 is used with a line driver (such as the iC-HD7 or iC-HX) or a high frequency FPGA, it is especially important to isolate the iC-TW8 power inputs (AVDD and DVDD) against external noise. In such applications, a dual voltage regulator such as the iC-DC is highly recommended. Note that both AVDD and DVDD must be the same voltage level (5V or 3.3V).

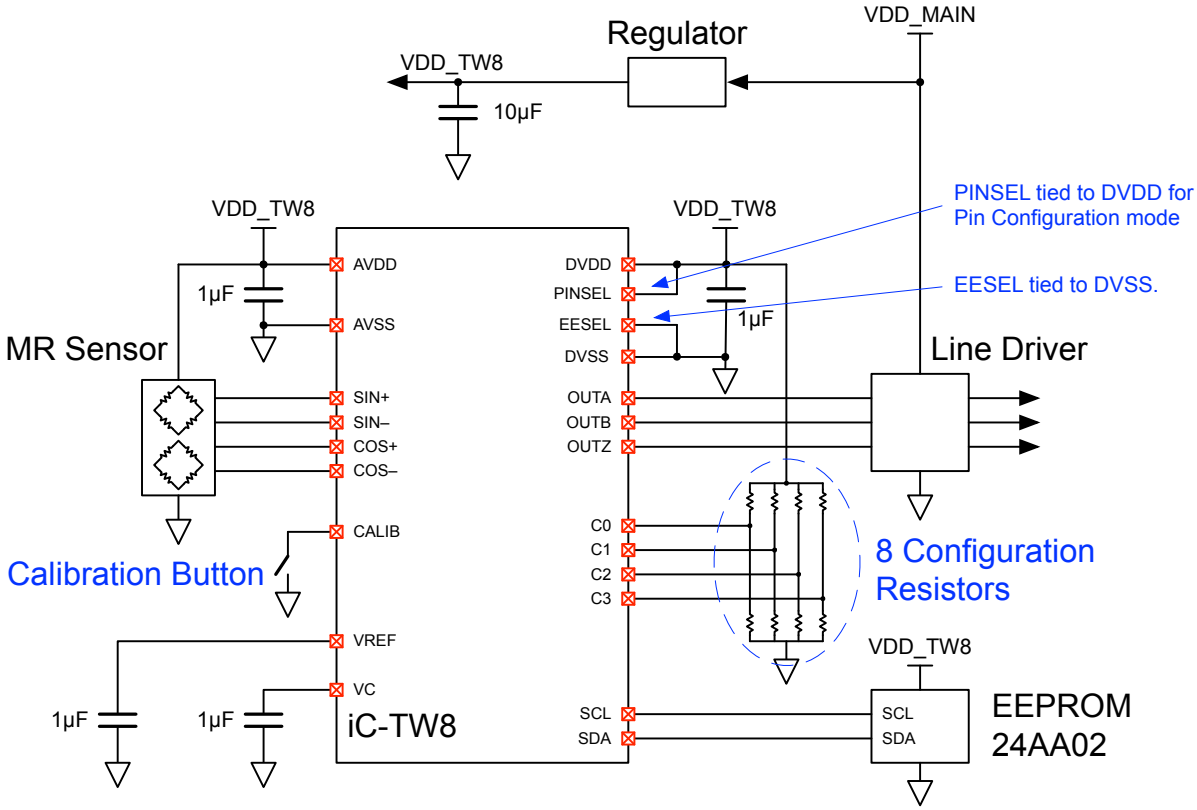


Figure 8: iC-TW8 Electrical Connections (Pin Configuration Mode)

Both pin VC (pin 44) and VREF (pin 45) must be decoupled to ground with 1µF each as shown. The EEPROM and the line driver (if used) must be decoupled according to their respective requirements. The calibration button provides an easy means to activate the auto-calibration feature of the iC-TW8 to set the data path parameters (offset, gain, and phase) to their optimal values. All pins not explicitly shown in Figure 8 should be connected to appropriate levels as indicated in PIN FUNCTIONS.

Connecting the Incremental Sensor

The iC-TW8 connects directly to magnetic (such as the iC-SM2L or iC-SM5L) or optical sensors providing differential Sin/Cos outputs as shown in Figure 9. Signal amplitude (A) from all four sensor channels must be in the range of 5 – 350mV as shown in Figure 10 for proper operation with the iC-TW8.

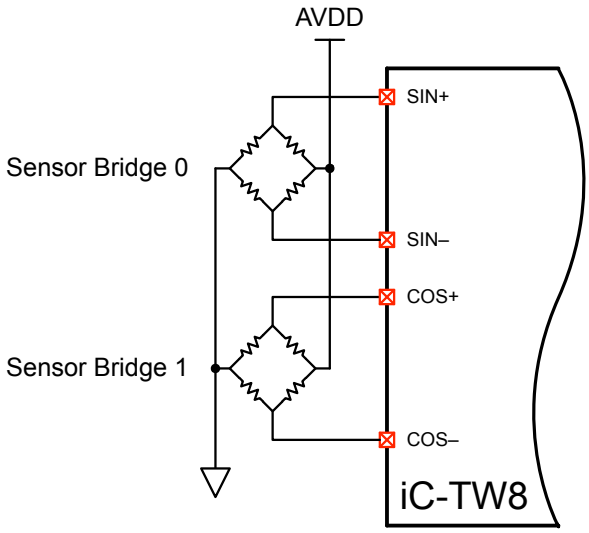


Figure 9: Differential Sensor Connection

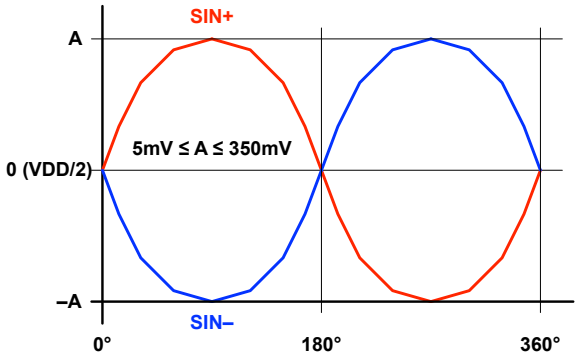


Figure 10: Differential Sensor Signal Inputs

Single-ended Sin/Cos inputs can also be connected to the iC-TW8 by connecting the SIN– and COS– pins to the proper bias voltage as shown in Figure 11. Since the iC-TW8 provides high-impedance signal inputs, a simple resistive voltage divider can be used to generate the required bias voltage. For AMR sensors, it is recommended that the resistors of the bias generator match the AMR sensor bridge resistance to improve power supply noise rejection. No decoupling capacitor should be used.

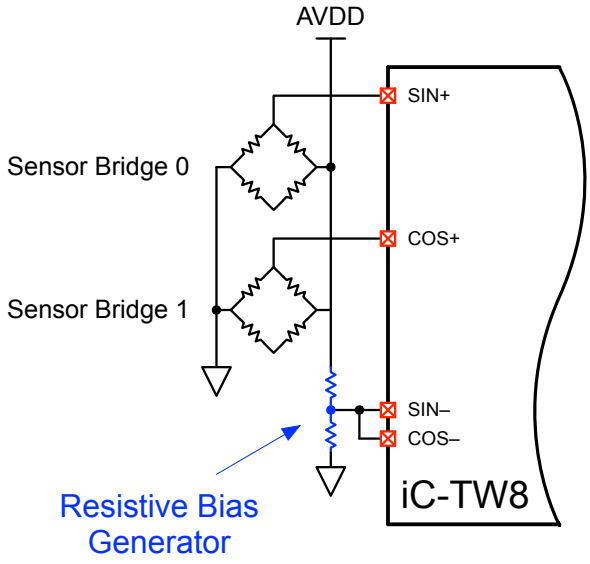


Figure 11: Single-ended Sensor Connection

Differential inputs should be used whenever possible to reject common mode distortions and provide increased signal amplitude and improved signal to noise ratio (SNR).

Regardless of input configuration, the auto-calibration feature of the iC-TW8 is sufficient in most cases to compensate for any signal offset, gain, and phase distortions without requiring any additional external components.

Connecting an Index Sensor (If Used)
The iC-TW8 can interface to a wide range of index sensors or gating sources to provide an index or Z output which is synchronized with the AB outputs. Digital Hall sensors are often used for this purpose, as shown in Figure 12.

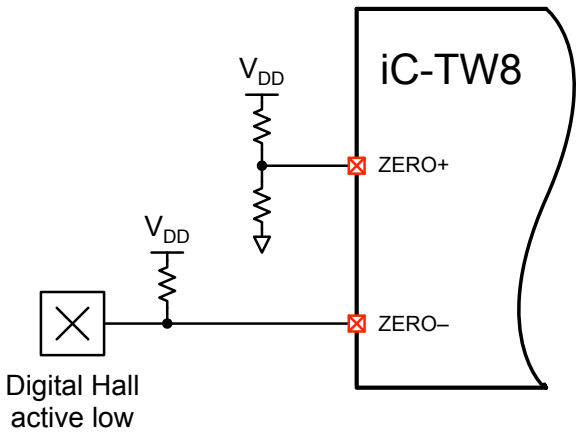


Figure 12: Index Sensor Connection

Digital sensors (Hall, MR, and others) typically provide an active-low signal via an open-drain output that pulls low in the presence of a magnetic field. Connect an active-low (open drain) digital index sensor to the iC-TW8 ZERO– input (pin 48) and connect the ZERO+ input (pin 47) to a resistive voltage divider set to the midpoint of the index sensor output voltage swing to provide good noise rejection. For active-high (open source) digital index sensors, reverse the ZERO+ and ZERO– connections.

To guarantee one and only one Z output from the iC-TW8 over multiple Sin/Cos input cycles, the ZERO input signal must be roughly centered on (in phase with) the Cos+ incremental sensor signal and no larger than 1.5 Sin/Cos input cycles under all conditions.

If no index (Z) output is required, connect ZERO+ (pin 47) to ground and ZERO– (pin 48) to AVDD.

Analog-output index sensors can also be used with the iC-TW8, but only in serial configuration mode.

Connecting the EEPROM
The iC-TW8 requires a 24xx02 through 24xx16 family I2C EEPROM for storage of sensor calibration data (offset, gain, etc.). The selected EEPROM must operate down to 1.8V; the Microchip 24A02 is recommended.

The EEPROM connects directly to the iC-TW8 via a dedicated I^2C communication channel using SCL (pin 27) and SDA (pin 28) as shown in Figure 13. No external pull-up resistors are required. EESEL (pin 4) of the TW8 must also be connected to DVSS as shown.

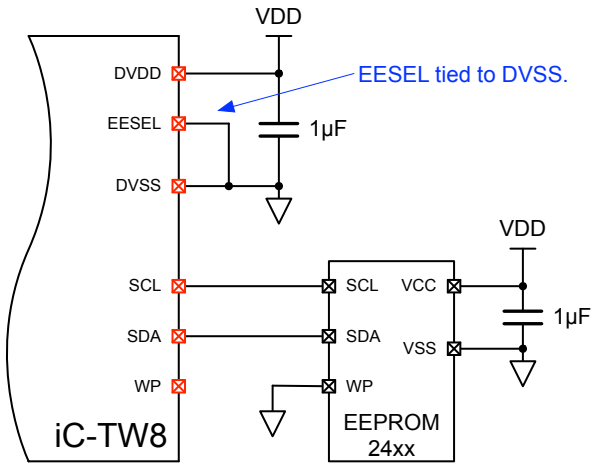


Figure 13: EEPROM Connection

In pin configuration mode, the Write Protect (WP) input of the EEPROM is not controlled by the iC-TW8. WP must be grounded (disabled) during configuration to allow the iC-TW8 to write the auto-adapted parameters to the EEPROM. In operation, WP can be left grounded (write protection disabled) as shown in Figure 13, or connected to DVDD by additional circuitry (such as a jumper) for parameter protection.

Power-on-Reset
The iC-TW8 contains a built-in power-on-reset (POR) circuit that controls the safe start-up of the device. The internal POR can be configured for two thresholds, 2.7V for use with 3.3V supplies or 4V, which is recommended when operating the iC-TW8 at 5V. Connect xRST (pin 2) to DVDD or leave it unconnected (xRST has an internal 40kΩ pullup resistor to DVDD) to select a power-on-reset threshold of 2.7V. Connect xRST to ground through a 68kΩ resistor to select a threshold of 4V. An external reset source (such as an RC circuit) can also be connected to xRST (pin 2) to directly control the POR behavior.

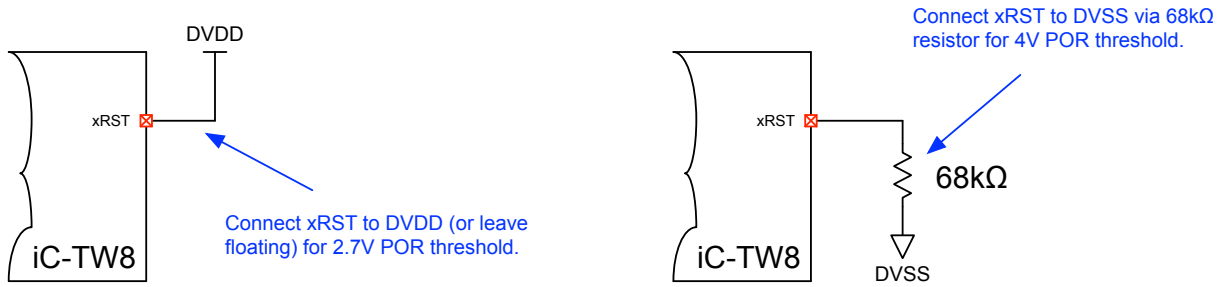


Figure 14: Setting the Power-On-Reset Threshold

Providing a Clock
The iC-TW8 supports three clocking modes as indicated in Figure 15. While the iC-TW8 provides an internal oscillator, it cannot be tuned in pin configuration mode. Thus, an external crystal or other clock source is recommended for all but the most cost-sensitive applications.

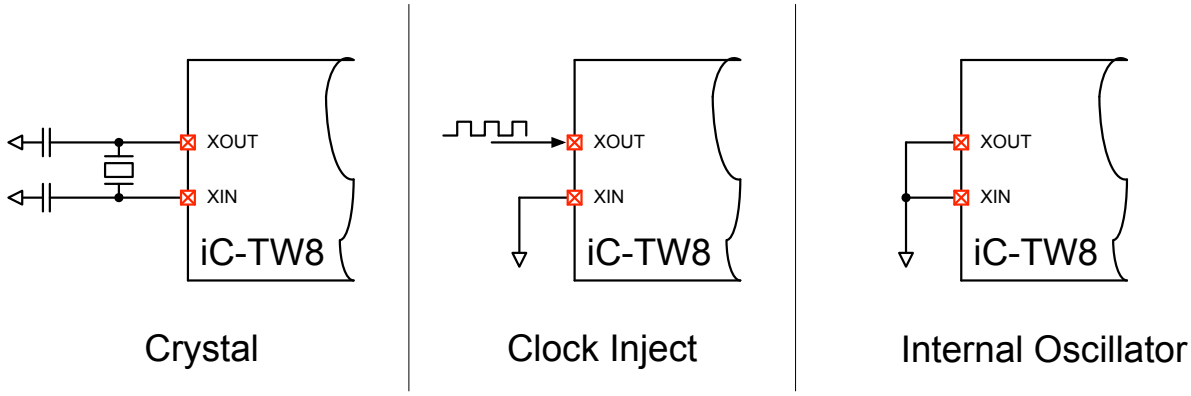


Figure 15: Clocking

An external crystal or oscillator provides the most stable clock signal over time and temperature variations, resulting in the most consistent long-term iC-TW8 performance. Connect an external crystal to XIN (pin 35) and XOUT (pin 34) as shown in Figure 15. Depending on the crystal characteristics, two capacitors to ground might be necessary for stable oscillation. To use an oscillator or ceramic resonator, connect XIN (pin 35) to ground and the oscillator output to XOUT (pin 34) as shown in Figure 15. Maximum crystal frequency (f_{clock}) is 32 MHz with 5V supplies and 24 MHz with 3.3V supplies. Minimum frequency is 6 MHz regardless of supply voltage.

To use the internal oscillator, connect both XIN (pin 35) and XOUT (pin 34) to ground as shown in Figure 15. The iC-TW8's internal oscillator has a nominal frequency of 20 MHz with 5V supplies and 16 MHz with 3.3V supplies. However, manufacturing tolerances and changes in temperature can cause large variations in the frequency of the internal oscillator.

Connecting the ABZ Outputs

In pin configuration mode, the iC-TW8 provides industry-standard quadrature incremental outputs at OUTA (pin 18) and OUTB (pin 17). If an index sensor is connected to the ZERO inputs of the iC-TW8, a Z output synchronous with the AB outputs is also available on OUTZ (pin 16). These outputs can be directly connected to an encoder counter such as the iC-MD.

Alternatively, the ABZ outputs can be connected to a differential line driver (such as the iC-HD7 or iC-HX) as shown in Figure 16. When using a line driver it is especially important to isolate the iC-TW8 from external switching noise using a regulator, as shown.

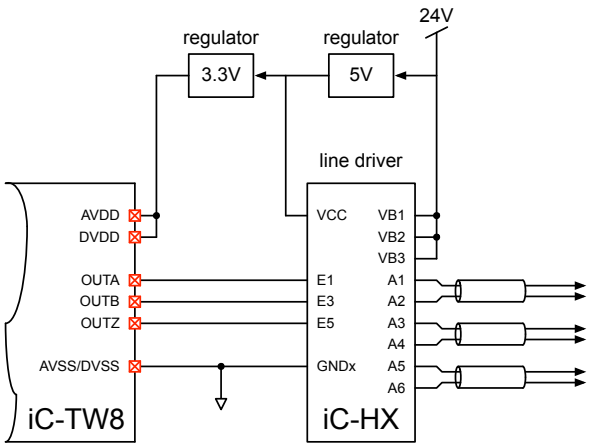


Figure 16: Line Driver Connection

Power-Up Sequence

After power-up, the iC-TW8's POR circuit monitors the supply voltage and waits until it has reached the appropriate threshold (as determined by the xRST connections). The iC-TW8 then starts up using the internal oscillator (even if a crystal or external clock source is connected) and executes its start-up cycle as outlined below.

1. Read parameter values from EEPROM.
2. Set proper clock source and frequency.
3. Read configuration inputs and enable signal path.
4. Wait for all analog circuitry to settle.
5. Clear all faults.
6. Start ABZ output generation.

If any faults are detected during the start-up cycle, the iC-TW8 does not enable the AB outputs but goes into an idle state with the FAULT output (pin 20) asserted (low).

The amount of time the iC-TW8 waits for the analog circuitry to settle (*twait*) depends on the clock frequency as shown by the formula below.

$$twait[\text{ms}] = \frac{1048}{f_{clock}[\text{MHz}]}$$

For example, using a 32 MHz crystal, the iC-TW8 waits 33 ms for the analog circuitry to settle.

Pin Configuration

In pin configuration mode, the iC-TW8 is configured by applying different voltages to configuration inputs C0, C1, C2, and C3 (pins 30, 31, 32, and 33 respectively). Each configuration input recognizes 12 different voltage levels which together select one of 124 = 20,736 separate operational configurations for the device.

The desired voltage levels are typically set using a resistive voltage divider on each of the configuration inputs as shown in Figure 17. Thus, only 8 resistors are required to completely configure the iC-TW8.

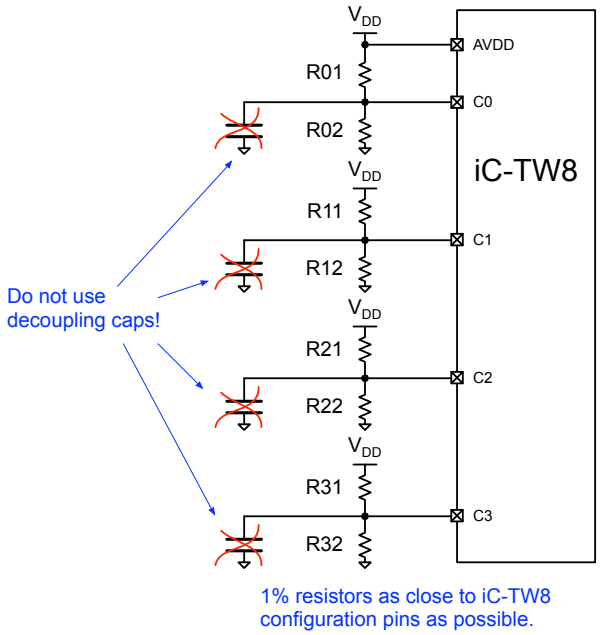


Figure 17: Pin Configuration Resistors

The resistors should be located as close to the configuration input pins as possible and no decoupling capacitors should be used.

Table 1 shows the recommended resistor values for setting the configuration levels. The resistors must be 1% (or better) tolerance to guarantee reliable operation under all conditions.

iC-TW8 Configuration Resistors			
Configuration Level	**Divider Ratio**	**Rx1 kΩ**	**Rx2 kΩ**
11	0.87	0.00	∞
10	0.80	3.01	12.1
9	0.73	4.02	11.0
8	0.67	4.99	10.0
7	0.60	6.04	9.09
6	0.54	6.98	8.06
5	0.46	8.06	6.98
4	0.40	9.09	6.04
3	0.33	10.0	4.99
2	0.27	11.0	4.02
1	0.20	12.1	3.01
0	0.13	∞	0.00

Table 1: Pin Configuration Resistor Values

An open circuit is shown as infinite (∞) resistance and a short circuit is shown as zero resistance in Table 1.

Setting the Interpolation Factor

Configuration inputs C0 and C1 are used to select the interpolation factor. Input C1 selects the desired interpolation group I0, I1, or I2 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 (pin 30) according to Table 1 to set the corresponding configuration level.

iC-TW8 Configuration Input C0			
C0 Level	**Interpolation Group I0**	**Interpolation Group I1**	**Interpolation Group I2**
11	x8192	x1600	x10000[1]
10	x4096	x800	x5000[2]
9	x2048	x400	x2500
8	x1024	x200	x1250
7	x512	x160	x1000
6	x256	x100	x500
5	x128	x80	x250
4	x64	x50	x125
3	x32	x40	x62.5
2	x16	x20	x25
1	x8	x10	x12.5[3]
0	x4	x5	x6.25[3]

Table 2: Configuration Input C0

[1] This interpolation factor can only be used with heavy filtering (C2 configuration levels 8 – 11) and lag recovery enabled (C3 configuration levels 6 – 11).

[2] This interpolation factor can only be used with normal or heavy filtering (C2 configuration levels 4 – 11) and lag recovery enabled (C3 configuration levels 6 – 11).

[3] Z output is not synchronous with AB outputs when using fractional interpolation factors.

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 as shown in the table.

Note that while the iC-TW8 is a true 16-bit interpolator, x10000 is the maximum interpolation factor available in pin configuration mode. Full 16-bit (x16384) interpolation is available in serial configuration mode.

Setting the Auto-Adaption Mode

Configuration input C1 is used to select the desired auto-adaption mode and the interpolation group corresponding to the interpolation factor selected via configuration input C0. Choose the desired auto-adaption mode and interpolation group from Table 3 and connect the appropriate resistors to configuration input C1 (pin 31) according to Table 1 to set the corresponding configuration level.

iC-TW8 Configuration Input C1		
C1 Level	**Interpolation Group**	**Auto-adaption Mode**
11 10 9	I2 I1 I0	A3: Full analog and digital auto-adaption.
8 7 6	I2 I1 I0	A2: Full digital auto-adaption.
5 4 3	I2 I1 I0	A1: Digital offset and gain auto-adaption.
2 1 0	I2 I1 I0	A0: No auto-adaption.

Table 3: Configuration Input C1

Three auto-adaption modes are available in pin configuration mode. Mode A3 provides full analog and digital auto-adaption. In this mode, the iC-TW8 continuously adjusts analog gain, analog offset, digital offset, digital gain, and digital phase during operation to maintain optimum performance and lowest error and jitter.

Auto-adaption mode A2 provides full digital auto-adaption and is the recommended mode for most applications. In this mode, analog gain and offset are fixed at the values established during device calibration (see Auto Calibration on page 25) and the iC-TW8 continuously adjusts only the digital offset, digital gain match, and digital phase during operation to maintain optimum performance and lowest error and jitter.

Auto-adaption mode A1 is similar to mode A2 except that digital phase is fixed at the value established during device calibration and not adjusted during operation.

In mode A0, auto-adaption is disabled and parameter values are not adjusted from their initial calibration values during operation.

For example, to configure an interpolation factor of x80 and no auto-adaption, set C0 to configuration level 5 (R01 = 8.06k, R02 = 6.98k) and C1 to configuration level 1 (R11 = 12.1k, R12 = 3.01k). To configure an interpolation factor of x250 and full auto-adaption, set C0 to

configuration level 5 (R01 = 8.06k, R02 = 6.98k) and C1 to configuration level 11 (R11 = 0).

Setting the Filter Mode and Hysteresis

Configuration input C2 selects the amount of filtering (smoothing) and the hysteresis of the AB outputs of the iC-TW8. Choose the desired filter mode and hysteresis from Table 4 and connect the appropriate resistors to configuration input C2 (pin 32) according to Table 1 to set the corresponding configuration level.

iC-TW8 Configuration Input C2		
C2 Level	**Hysteresis (AB Edges)**	**AB Filter Mode**
11 10 9 8	1 2 4 8	Heavy Filtering. Smooth output at the expense of fast response time.
7 6 5 4	1 2 4 8	Normal filtering. A good balance between smoothness and response time.
3 2 1 0	1 2 4 8	Minimal filtering. Outputs respond quickly to sensor inputs (and noise).

Table 4: Configuration Input C2

The choice of filter mode is a compromise between fast response, smoothness of the AB outputs, and position lag at constant speed. It is recommended to start with minimal filtering since this gives the fastest response of the AB outputs to changes in the Sin/Cos inputs. Normal or heavy filtering may be selected if the outputs are noisy or jittery. Experimentation may be necessary to determine the optimal value.

Note that configuration level C2 also sets the AB output hysteresis in edges. The corresponding hysteresis in degrees of a Sin/Cos input can be calculated using the following formula.

$$Hysteresis[°] = \frac{90}{Interpolation} \cdot Hysteresis[\text{Edges}]$$

Where *Interpolation* is the interpolation factor set using configuration inputs C0 and C1. For example, if *Interpolation* is set to x100 and hysteresis is set to 2 AB edges, the hysteresis in degrees of a Sin/Cos input cycle is

$$Hysteresis[°] = \frac{90}{100} \cdot 2 = 1.8°$$

Setting the Maximum AB Frequency and Lag Recovery

Configuration input C3 sets the maximum AB output frequency and determines whether lag recovery is used or not. Choose the desired maximum AB frequency and lag recovery setting from Table 5 and connect the appropriate resistors to configuration input C3 (pin 33) according to Table 1 to set the corresponding configuration level.

iC-TW8 Configuration Input C3		
C3 Level	**Maximum (AB Frequency)**	**Lag Recovery**
11 10 9 8 7 6	$f_{clock}/8$ $f_{clock}/16$ $f_{clock}/32$ $f_{clock}/64$ $f_{clock}/128$ $f_{clock}/256$	Lag recovery enabled.
5 4 3 2 1 0	$f_{clock}/8$ $f_{clock}/16$ $f_{clock}/32$ $f_{clock}/64$ $f_{clock}/128$ $f_{clock}/256$	Lag recovery disabled.

Table 5: Configuration Input C3

In pin configuration mode, the highest maximum output frequency of the AB outputs is $f_{clock}/8$, where f_{clock} is the clocking frequency of the iC-TW8 (see Providing a Clock on page 19). Lower maximum output frequencies can be selected as shown in Table 5 if devices connected to the iC-TW8 (counters, PLCs, motion controllers, drives, etc.) cannot handle its full output frequency.

The incremental sensor input frequency (f_{input}) which corresponds to a given AB output frequency can be calculated using the following formula:

$$f_{input} = \frac{\text{AB Frequency}}{Interpolation}$$

Where *Interpolation* is the interpolation factor set using configuration inputs C0 and C1.

For example, if f_{clock} = 32 MHz, an interpolation factor of 2500 is selected using C0 and C1, and C3 is at configuration level 8 ($f_{clock}/64$), the maximum AB frequency will be reached at a sensor input frequency of

$$f_{input} = \frac{\frac{32}{64}}{2500} = 0.0002\,\text{MHz} = 200\,\text{Hz}$$

If the incremental sensor input exceeds this frequency, the AB output position can no longer keep up with the sensor position. In this case, the iC-TW8 keeps generating output pulses at the maximum AB frequency until the AB output position either catches up to the sensor position or falls behind the sensor position by half a Sin/Cos input cycle, at which point the FAULT output (pin 20) is activated (latched low). This fault must be cleared by cycling power to the iC-TW8.

Lag recovery can be used to eliminate most of the position lag at constant speed caused by the AB output filter, at the expense of possible position overshoot on stopping. With lag recovery enabled, the AB output position "catches up" to the sensor position during constant speed motion. However, enabling lag recovery also causes the AB output filter response to become under-damped which can result in position over- and undershoot with fast sensor input position changes. This effect is worse with more AB output filtering (configuration input C2). At standstill, the AB output position is always equal to the sensor position regardless of the AB output filter and lag recovery settings.

Unless specifically required otherwise, it is recommended to use the highest maximum AB frequency and to disable lag recovery by setting C3 to configuration level 5.

Serial Interfaces

The SPI and 1-wire serial interfaces of the iC-TW8 are fully active while in pin configuration mode. It is possible to read and write to memory and observe the current configuration. However, any changes made to the configuration are only temporary. After cycling power, the configuration reverts to that set by the configuration inputs C0 – C3.

Auto Calibration
Once the iC-TW8 has been configured using the configuration inputs, the signal path must be calibrated to determine proper values for gain, offset correction, gain match, and phase correction. This is most easily done using the Auto Calibration feature of the iC-TW8 to automatically determine optimum values for these parameters.

Auto Calibration is initiated by pulling CALIB (pin 5) low. A pushbutton switch connected to CALIB (as shown in Figure 9 on page 17) is an easy way to achieve this in series production. With CALIB held low, provide sensor input of a few hundred Sin/Cos cycles and the iC-TW8 "tunes" the analog gain, analog offset correction, digital offset correction, digital gain match, and digital phase correction parameters to provide lowest error and jitter in the interpolated AB outputs.

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 a few revolutions; a linear encoder by moving the sensor back and forth on the scale by a few centimeters.

After providing sufficient input signals, release the CALIB input (it is pulled high by an internal pull-up resistor) and the iC-TW8 immediately stores all calibration parameters in the external EEPROM. These auto-calibrated values are then used whenever the iC-TW8 is powered up.

iC-TW8 Rotary Encoder Design Tool
A design tool in the form of an Excel spreadsheet is available to simplify configuration of the iC-TW8 in rotary encoder applications. Only 10 values (in engineering units) are required to completely specify the desired iC-TW8 configuration. The required configuration resistors values and performance limits and graphs are then available for implementation and characterization of the encoder.

In the design tool spreadsheet, cells with a blue background are used for user entry. Start by entering the operating voltage (Vdd) as 5.0 or 3.3 and the frequency of the crystal you are using. The operating voltage and crystal frequency affect the current drawn by the iC-TW8, which is calculated and shown for reference. The crystal frequency also affects the maximum speed and the filter response. In general, use the highest possible frequency unless there is a need to minimize current.

Continue by entering the resolution of the magnetic wheel or optical disc. This value is the number of Sin/Cos cycles per revolution of the wheel or disc produced by the sensor connected to the iC-TW8. Next enter the desired output resolution of the encoder as an integer equal to the number of AB cycles per revolution (CPR). This is also sometimes called PPR (pulses per revolution) and is equivalent to the number of "lines" in a non-interpolated optical encoder. Fractional output resolutions are not possible in pin-configuration mode.

With these values entered, the design tool calculates and displays the required interpolation factor. Interpolation factors not available with the iC-TW8 (or not selectable in pin-configuration mode) are flagged by a warning in red "Output resolution not possible with given input resolution!"

Next, enter the desired maximum speed (speed limit) for the encoder in RPM. The spreadsheet calculates and displays the closest speed limit that can be implemented using the iC-TW8 in pin-configuration mode. The calculated actual speed limit is shown in RPM, AB channel frequency, AB edge frequency, and as minimum time between edges. Exceeding the actual speed limit activates the iC-TW8's FAULT output (pin 20).

For an enclosed encoder the maximum speed is usually dictated by the bearings used. For a modular encoder (without bearings) or to use the full speed capability of the iC-TW8, enter an arbitrarily large value (such as 1,000,000 RPM). Note that he calculated actual speed limit is affected by the chosen crystal frequency and the interpolation factor used.

Next, enter the desired level of filtering for the AB outputs and whether lag recovery is desired. Performance graphs of encoder lag in AB edges and response to a trapezoidal motion profile with constant acceleration, velocity, and deceleration are displayed to allow observing the effect of the different filter settings. Scroll to the right on small computer screens to see the performance graphs.

A single parameter (maximum acceleration) is used to generate the filter response graphs. Enter a value in radians per second^2 that represents the maximum expected acceleration of the encoder. The design tool calculates the equivalent acceleration in units of kRPM (1000 RPM) per second and time to reach the chosen maximum speed, which may be more easily understood. The acceleration value is used only to generate the performance graphs, it does not affect the configuration of the iC-TW8. The motion profile is generated by accelerating for 1000 ADC samples at the entered acceleration, running at the attained speed for 1000 ADC samples, and decelerating at the entered acceleration for 1000 ADC samples.

The maximum acceleration (α_m) of a motor in radians per second^2 may be calculated knowing its stall torque (T_s) in Newton meters and its rotor inertia (J_r) in kilogram meters^2 using the following formula.

$$\alpha_m = \frac{T_s}{J_r}$$

For example, the Maxon EC 90 flat 90W motor with the 24V winding (order number 323772) has a rotor inertia of 3,060 gcm^2 and a stall torque of 4,670 mNm. Thus its no-load maximum acceleration in radians per second^2 is

$$\alpha_m = \frac{T_s}{J_r} = \frac{4670}{3060} \cdot 10,000 = 15,261$$

Using this value for the maximum acceleration parameter in the design tool will then show the performance of the iC-TW8 when used in an encoder for feedback with this motor.

The graphs below show typical encoder lag and motion profile performance of the iC-TW8 using normal filtering and lag recovery at 15,261 radians per second^2 maximum acceleration and a resolution of 16,384 AB cycles per revolution.

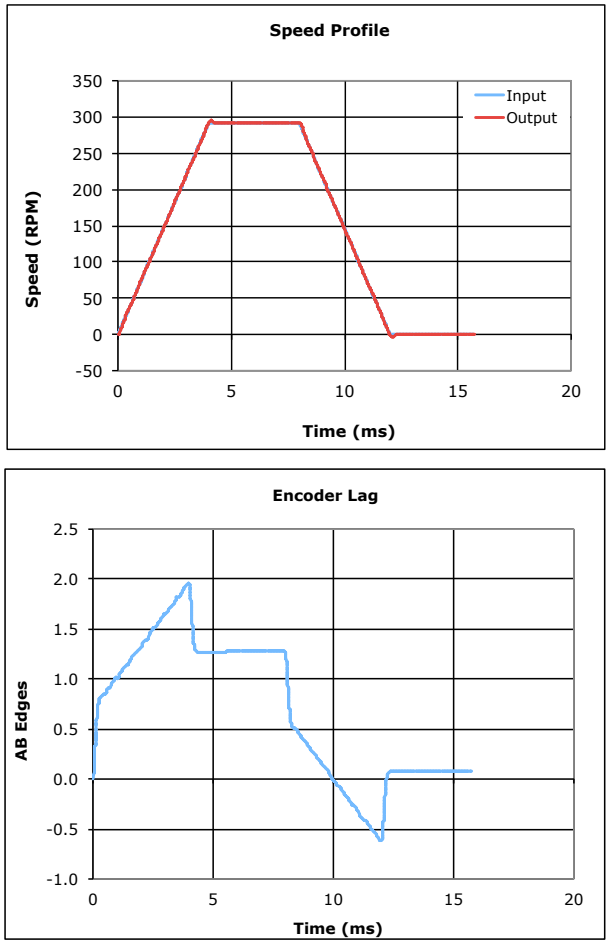


Figure 18: Typical Encoder Performance Graphs

The design tool also calculates the encoder lag in AB edges per kRPM and mechanical degrees (°m) of encoder rotation at the chosen maximum speed for the selected filter configuration. For a given filter and lag recovery setting, encoder lag is directly proportional to encoder speed.

The design tool also calculates the position and speed overshoot due to the AB output filter. Position overshoot (shown in AB edges and mechanical degrees of encoder rotation) is proportional to the entered maximum acceleration and is typically noticed on stopping (deceleration). At low acceleration values and/or low encoder resolutions, position overshoot may be less than one AB edge and thus totally undetectable during encoder operation, even with heavy filtering.

Speed overshoot is calculated as a percent of speed. Speed overshoot is typically noticed at the end of the acceleration ramp (the beginning of the constant velocity portion of the motion profile). Speed overshoot as a percent of speed is affected only by the selected AB filter level and whether or not lag recovery is used. The

actual amount of speed overshoot in RPM is proportional to encoder speed.

Next, enter the desired AB output hysteresis in AB edges. The design tool calculates the equivalent hysteresis in degrees of a Sin/Cos input cycle. Hysteresis is used to prevent instability (dithering) of the AB outputs at standstill due to noise. In general, use the lowest hysteresis that produces stable AB outputs with no sensor motion.

Next enter Yes or No as to whether an index sensor is used. If an index sensor is not used, the ZERO inputs of the iC-TW8 should be connected to AVDD and AVSS as explained on page 18. If an index sensor is used the width of the Z output pulse is shown. The Z output pulse is always one AB edge (one quadrature state) wide in pin configuration mode. The Z output is synchronous with A and B low when an integer interpolation factor is used. Not using an index sensor allows higher speeds with high interpolation factors.

Enter the desired auto-adaption mode for the iC-TW8. Auto-adaption refers to the iC-TW8's ability to track and change (adapt) sensor gain, offset, and phase during encoder operation to maintain low AB output error and jitter under all operating conditions. One of four auto-adaption modes can be selected and the design tool confirms which iC-TW8 parameters are adapted during operation. Regardless of the selected mode, auto-adaption is disabled above the displayed speed. In pin configuration mode, the maximum auto-adaption speed is fixed and depends on the selected maximum speed, crystal frequency, and interpolation factor.

At this point, the iC-TW8 configuration is completely specified and the configuration resistor values necessary to achieve this configuration are shown. See Figure 17 on page 21 for a schematic of the electrical connection of the resistors to the iC-TW8. Also shown is a jumper map for setting the TW8_1D demo board to the selected configuration.

Finally, the design tool shows the conditions which activate or latch the FAULT output (pin 20). Latched faults are cleared by cycling power to the iC-TW8.

Pin Configuration Reference

At power-on, the four configuration input levels (C0 – C3) are read and used to set internal variables of the iC-TW8 to the proper values. Other internal variables are also set to specific values and cannot be changed using the configuration inputs. Table 6 provides a list of the internal variable values used in pin configuration mode as a reference for trouble-shooting or new applications. Variables not listed in the table are set to 0. Detailed information on all iC-TW8 internal variables is available in the iC-TW8 Programmer's Reference.

iC-TW8 Pin Configuration Reference	
iC-TW8 Variable	**Value**
MAIN_CFG	wp = 1
MAIN_INTER	Determined by C0 and C1.
MAIN_HYST	Determined by C2.
MAIN_FLTR	Determined by C2 and C3.
MAIN_OUT	start = 10 (0x0A)
MAIN_CLOCK	xtal = 1, div = 3, xforce = 1
MAIN_Z	reset = 0
AB_CFG0	startmode = 1
AB_VTOP	Determined by C3.
ADPT_CFG	Determined by C1.
ADPT_DETAIL	flimit = 6, tbase = 4
ADPT_CORR	prop = 2, offtol = 2, gaintol = 2
MON_OFF	limit = 10
MON_GAIN	limit = 10
MON_PHASE	limit = 10
FLT_CFG	vwarn = 1, long = 1
FLT_EN	ee = 1, vfatal = 1

Table 6: Pin Configuration Reference

SERIAL CONFIGURATION MODE

Introduction

This section describes the serial configuration mode of the iC-TW8 Interpolator. Serial configuration mode requires configuration using the SPI or 1-wire communication ports and allows access to all the powerful features of the iC-TW8 for creating sophisticated applications. For simple applications, pin-configuration mode provides simple, static configuration of the device without any programming or complicated calibration. Pin configuration mode is described in the previous section.

A design tool for rotary encoders using the iC-TW8 is available. This tool is in the form of an Excel spreadsheet that allows entering application parameters in engineering units and then provides performance limits and graphs, configuration register values, and configuration documentation for the application. This greatly simplifies and speeds up the design process of encoders using the iC-TW8.

Documentation Conventions

Throughout this section, hexadecimal values are written in C-style where the hex value is preceded by "0x". For example, $80CA_{16}$ is written as 0x80CA.

Internal iC-TW8 registers are named using all CAPITAL letters. Individual bits or bit groups within the register use lower case Roman lettering. For example, MAIN_CFG.wp refers to the wp bit (bit 0) in the MAIN_CFG register at address 0x8044. Likewise, MAIN_OUT.mode refers to the 2-bit group mode (bits 4–5) of register MAIN_OUT at address 0x804B. See the iC-TW8 Programmer's Reference for complete information on all iC-TW8 registers.

Derived variables not explicitly available as register values are shown in all lower-case italics. For example, the formula

$$inter = \frac{\text{MAIN\_INTER}}{4}$$

shows that internal variable *inter* is derived from the value in the MAIN_INTER register.

Internal variable units-of-measure are shown in brackets following the variable name. For instance *adcs*[V] is the sin channel ADC input level in volts.

Functional Overview

The iC-TW8 is a general-purpose 16-bit Sin/Cos interpolator with sophisticated auto-calibration functions. It accepts 10 – 700mV differential analog Sin/Cos 45 input signals from magnetic or optical sensors and calculates (interpolates) the angular position. Output is industry-standard incremental AB quadrature at programmable resolution. Automatic calibration requires no complicated signal analysis or calibration procedure during product design or production. Auto-adaption monitors and adapts parameter values during operation to maintain optimal performance and low error and jitter.

In addition to the chosen sensor, the iC-TW8 requires an external EEPROM to store its configuration and calibration data. An external crystal can be used instead of the internal oscillator of the iC-TW8 for superior frequency stability. A differential line driver can be connected directly to the iC-TW8 quadrature outputs to make a complete encoder.

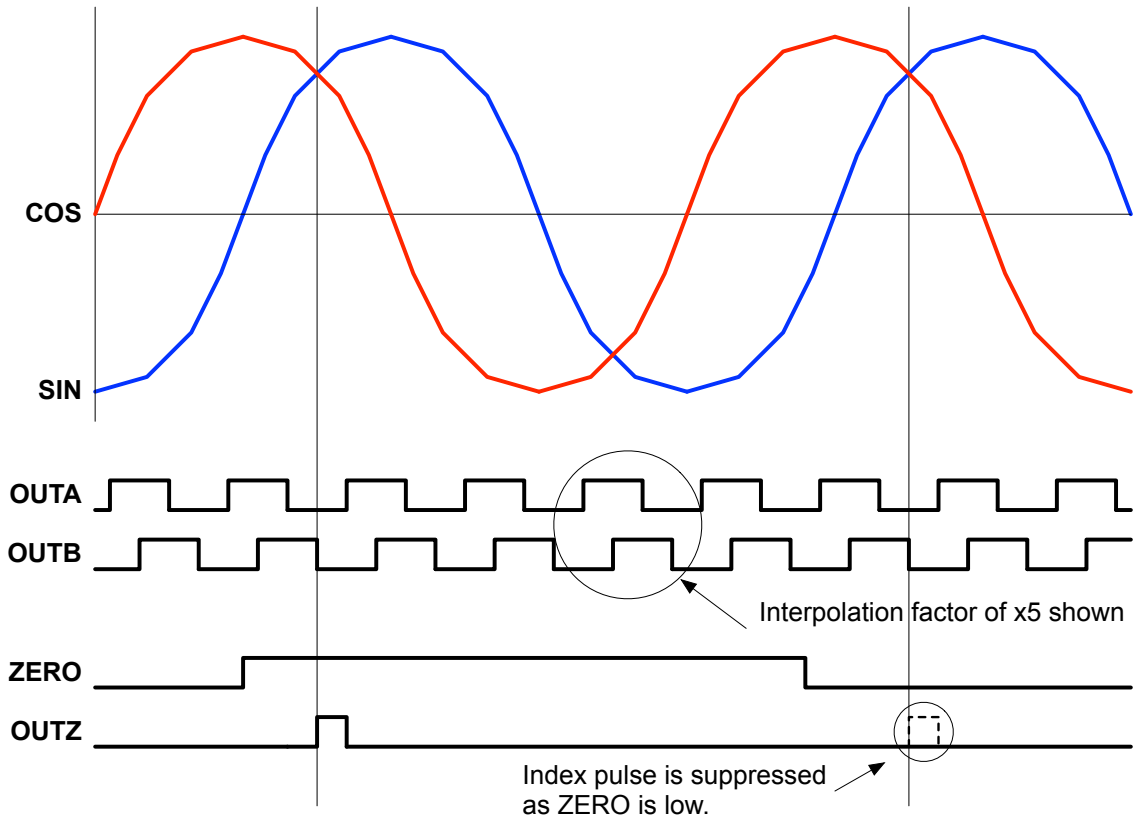


Figure 19: iC-TW8 Functional Overview

Functional Block Diagram

A functional block diagram of the iC-TW8 is shown in Figure 20 and explained on the following pages.

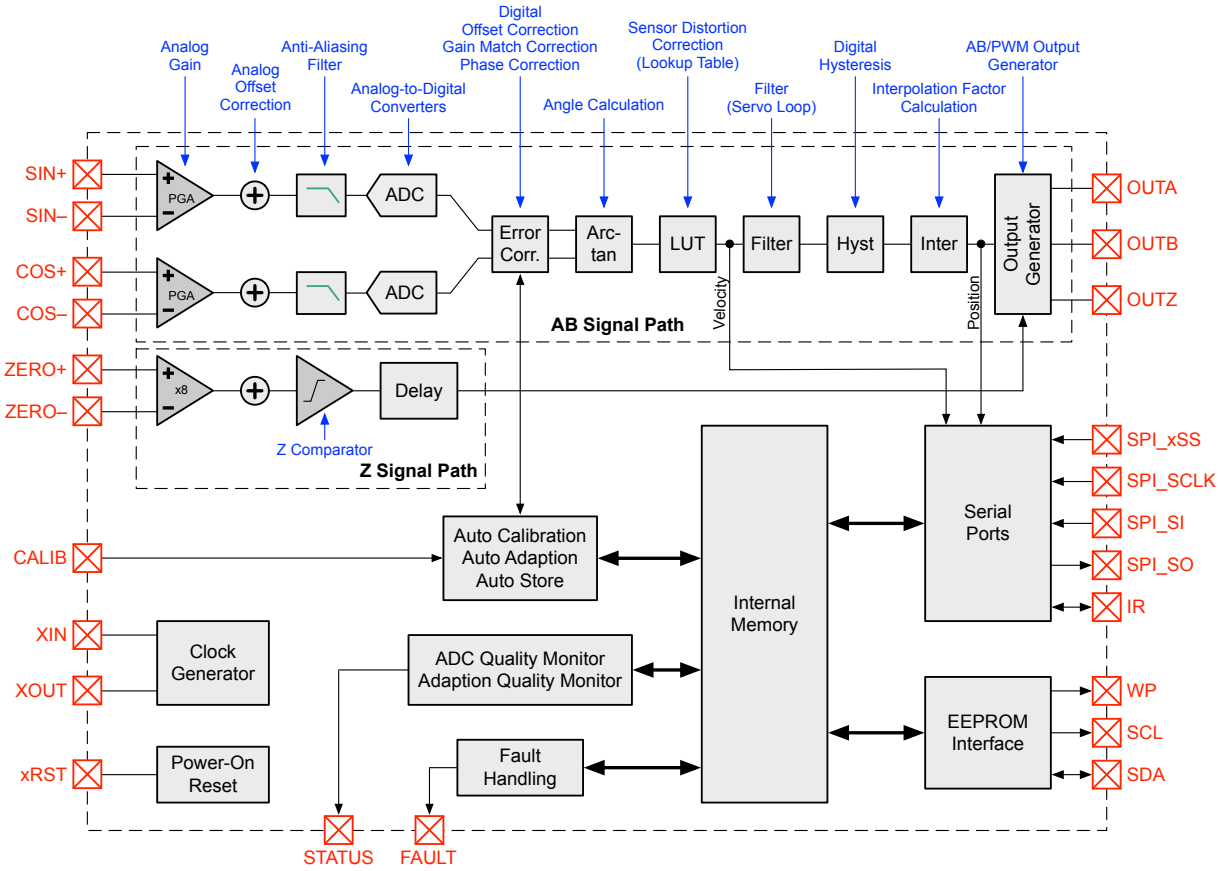


Figure 20: iC-TW8 Block Diagram

Clock Generator

The Clock Generator uses an external crystal or other clock source to generate the internal core clock (f_{core}) for the iC-TW8. An internal oscillator is also provided which can be used if an external crystal is not desired. The internal core clock controls all timing within the iC-TW8.

Power-On Reset

The Power-On Reset (POR) circuit provides orderly startup of the iC-TW8 when power is applied. An external reset source can also be connected to the POR to allow independent control of device startup.

Serial Ports

The iC-TW8 contains two serial ports that may be used for communication. These are a standard SPI (Serial Peripheral Interface) slave port and a 1-wire port, both of which utilize the same communication protocol (see Serial Communication Ports in the Programmer's Reference). Both ports may be used simultaneously.

EEPROM Interface

A 24xx02 through 24xx16 family I^2C EEPROM is required for storage of sensor calibration data (offset, gain, etc.). The EEPROM connects directly to the iC-TW8 via the dedicated I^2C EEPROM interface.

Internal Memory

The iC-TW8 provides 32K bytes of internal memory divided into eight functional blocks that can be accessed by the user via either of the serial ports. See Internal Memory in the Programmer's Reference for more information.

AB Signal Path

The AB Signal Path consists of the blocks between the analog PGAs and the output generator. The front end of the AB signal path is shown below.

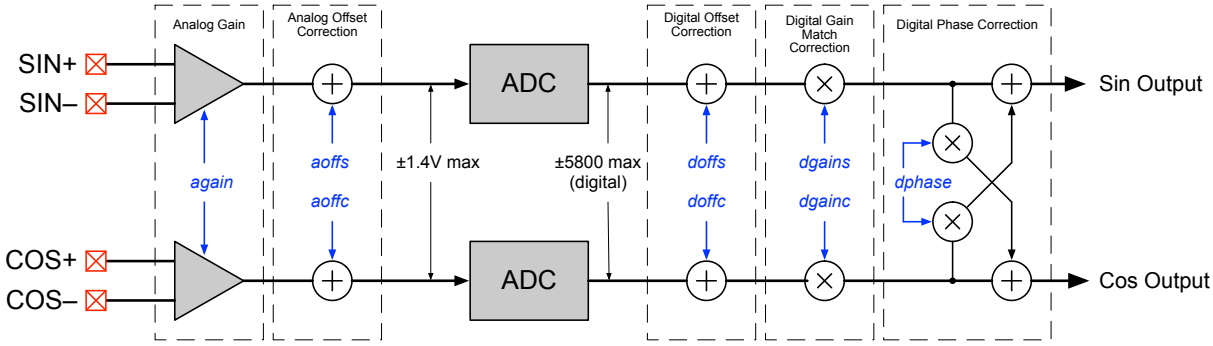


Figure 21: AB Signal Path Front End

Analog Gain

Analog gain is applied to the raw Sin/Cos sensor inputs by two programmable gain amplifiers (PGAs) with a range of 6–45 dB in 3dB steps. The actual analog gain, *again*, is determined by the parameter AGAIN in the PAR block in internal memory. The same gain is applied to both channels.

Analog Offset Correction

Individual analog offset correction is applied to the output of the two analog PGAs This is a coarse correction with a range of ±3.1V in 100mV steps. The actual analog offset corrections for the Sin and Cos channels, *aoffs* and *aoffc* respectively, are determined by the parameters AOFFS and AOFFC in the PAR block in internal memory.

The effective analog offset correction referenced to the input Sin/Cos signals, *aoffins* and aoffinc is a function of the analog gain and can be calculated as

$$aoffins = \frac{aoffs}{again} \qquad aoffinc = \frac{aoffc}{again}$$

Analog-to-Digital Converters (ADCs)

The two ADCs convert the conditioned analog Sin and Cos signals into 14-bit digital values for further processing. The remainder of the AB signal path is completely digital.

The ADCs in the iC-TW8 operate best with inputs limited to ±1.4V maximum, which corresponds to an output value of approximately ±5800 ADC increments. Above these values, ADC distortion increases and produces a corresponding reduction in interpolation accuracy.

Digital Offset Correction

Individual fine offset correction is next applied to the digital Sin and Cos ADC outputs. A range of fine offset correction of ±125mV in 244µV steps at the ADC inputs is available. The actual digital offset corrections for the Sin and Cos channels, *doffs* and *doffc* respectively, are determined by the parameters DOFFS and DOFFC in the PAR block in internal memory.

The effective digital offset correction referenced to the input Sin/Cos signals can be calculated as

$$doffins = \frac{doffs}{again} \qquad doffinc = \frac{doffc}{again}$$

Digital Gain Match Correction

Next, digital gain match correction is applied to either the Sin or Cos channel to correct any mismatch between the two signal amplitudes. The actual digital gain values for the Sin and Cos channels, *dgains* and *dgainc* respectively, are determined by the DGAIN parameter in the PAR block in internal memory. A range of ±1.25 in steps of 0.000244 at the ADC inputs is available.

Digital Phase Correction

Lastly, phase correction is applied to the Sin and Cos channels to guarantee 90° phase shift between the two signals. The actual digital phase correction, *dphase*, is determined by the DPH parameter in the PAR block in internal memory. A range of ±53° in steps of 0.056° is available.

Angle Calculation (Arctan)

The angle within an input cycle indicated by the conditioned digital Sin and Cos signals is next calculated

as *arctan*(Sin/Cos) with a resolution of 14 bits using a CORDIC algorithm.

Sensor Distortion LUT
An optional 64-byte lookup table (LUT) is provided to allow correction of any remaining sensor distortion (see Using the LUT on page 59). The LUT can be bypassed if not required.

Filter
A filter is provided in the AB signal path that adds two bits of resolution to the angle calculation (giving 16-bit angle resolution), reduces noise and jitter in the AB outputs, and can reduce angle lag at constant speed due to interpolator latency. The filter is implemented as a PI servo loop with feedback path delay as shown below.

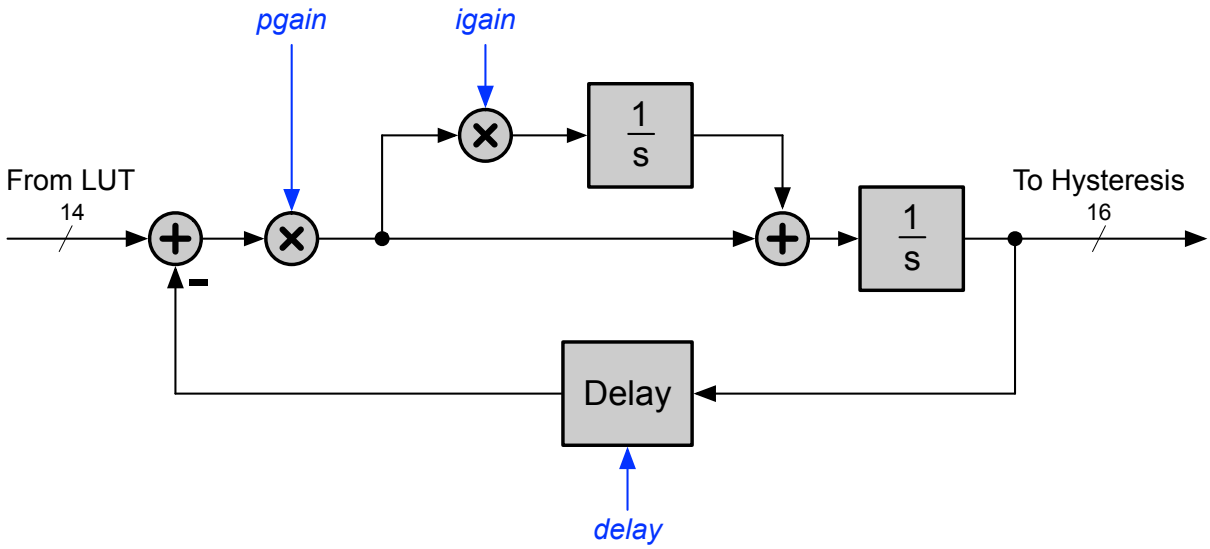


Figure 22: AB Signal Path Filter

The filter parameters *pgain*, *igain*, and *delay* are determined by variables in the MAIN_FLTR register in the CFG block in internal memory. The MAIN_FLTR register value necessary to realize a given filter configuration is called its "instance".

Because of the performance tradeoffs related to the selection of the filter parameters, it is helpful to define three filter modes to make parameter selection easier. Experimentation may be necessary to determine optimal filter parameters.

iC-TW8 Filter Modes	
Mode	Description
P	Low-pass filter with lag.
PI	Critically-damped low-pass filter with no filter lag.
Lag Recovery	Under-damped low-pass filter with no filter lag and reduced latency lag.

Table 7: iC-TW8 Filter Modes

P Mode is the simplest filter mode and uses only *pgain* (*igain* = *delay* = 0). In P mode, the signal path filter configuration reduces to

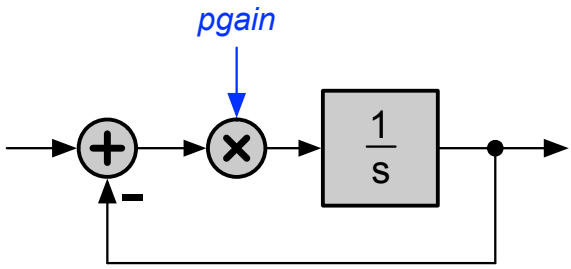


Figure 23: P Mode Filter Configuration

In P mode, the *pgain* value determines the noise and jitter bandwidth of the filter. Position lag at constant sensor input velocity is directly proportional to the velocity and inversely proportional to *pgain*. There is no position or speed overshoot in P mode.

The following P mode filter instances* and approximate filter bandwidths (–3dB) are available.

P Mode Filter Instances	
Instance	Noise and Jitter Bandwidth [kHz]
112	2.5 · f_{core} [MHz] (Minimum lag)
114	2.5/3 · f_{core} [MHz]
116	2.5/7 · f_{core} [MHz]
118	2.5/15 · f_{core} [MHz]
120	2.5/31 · f_{core} [MHz]
122	2.5/63 · f_{core} [MHz]
124	2.5/127 · f_{core} [MHz] (Maximum lag)

Table 8: P Mode Filter Instances

* The filter instance is the MAIN_FLTR register value necessary to realize the given filter.

In general, use the maximum filter bandwidth (instance 112); this gives the fastest response and minimum filter lag. Another instance may be used if additional noise and jitter filtering is required.

PI Mode adds *igain* to the P mode filter, eliminating filter lag at constant sensor input velocity at the expense of critically damped filter response. In PI mode, the signal path filter configuration is

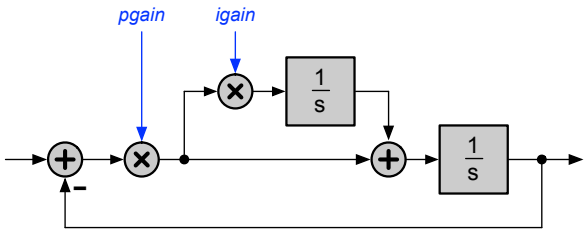


Figure 24: PI Mode Filter Configuration

In PI mode, filter lag at constant sensor input velocity is eliminated, but position and speed overshoot are inversely proportional to filter bandwidth.

The following PI mode filter instances and approximate filter bandwidths (–3dB) are recommended (instances not shown may cause unstable filter response).

Recommended PI Mode Filter Instances	
Instance	Noise and Jitter Bandwidth [kHz]
52	2.5/7 · f_{core} [MHz] (Minimum overshoot)
70	2.5/15 · f_{core} [MHz]
88	2.5/31 · f_{core} [MHz]
106	2.5/63 · f_{core} [MHz] (Maximum overshoot)

Table 9: PI Mode Filter Instances

In general, use the maximum filter bandwidth (instance 52); this gives the fastest response and minimum position and speed overshoot. Another instance may be used if additional noise and jitter filtering is required.

Lag Recovery Mode adds delay to the feedback path of the PI mode filter, reducing latency-induced position lag at constant sensor input velocity at the expense of under-damped filter response. In lag recovery mode, the signal path filter configuration is as shown in Figure 22 on page 32.

In lag recovery mode, position lag at constant sensor input velocity due to interpolator latency is reduced, but filter response is under-damped and position and speed overshoot are inversely proportional to filter bandwidth. The following lag recovery mode filter instances and approximate filter bandwidths (–3dB) are recommended (instances not shown may cause unstable filter response).

Recommended Lag Recovery Mode Filter Instances	
Instance	Noise and Jitter Bandwidth [kHz]
71	2.5/15 · f_{core} [MHz] (Minimum overshoot)
89	2.5/31 · f_{core} [MHz]
107	2.5/63 · f_{core} [MHz] (Maximum overshoot)

Table 10: PI Mode Filter Instances

In general, use the maximum filter bandwidth (instance 71); this gives the fastest response and minimum position and speed overshoot. Another instance may be used if additional noise and jitter filtering is required.

Digital Hysteresis

Digital hysteresis may be added to the interpolated sensor input angle to reduce AB output dithering (instability) at standstill at the expense of position error on direction reversal. A hysteresis range of 0–22.5° of an input cycle in 0.044° steps is available. The actual hysteresis, *hyst*[°], is determined by the MAIN_HYST register in the CFG block in internal memory.

Interpolation Factor

The 16-bit sensor input angle value is scaled to the resolution required by the desired interpolation factor. A range of 4 – 65,536 AB output edges per sensor Sin/Cos input cycle in steps of 1 edge is available. This is equivalent to a interpolation range of x1.00 – x16,384.00 AB output cycles per input cycle in steps of 0.25 AB output cycle. The actual interpolation factor, *inter*, is determined by the MAIN_INTER register in the CFG block in internal memory.

Output Generator

The output generator generates quadrature AB outputs or a pulse-width modulated (PWM) bit stream output depending on the output mode. The output mode is determined by MAIN_OUT.mode in the CFG block in internal memory. The output generator also uses the conditioned ZERO signal from the Z signal path to generate a Z output that can be synchronized to the AB outputs.

The output generator contains an optional divider that can be used in AB output mode to produce fractional interpolation factors. This is useful in situations where the desired output resolution is not an integer multiple of the input resolution.

Z Signal Path

The Z signal path is similar to the front end of the AB signal path except that a 1-bit ADC (comparator) is used.

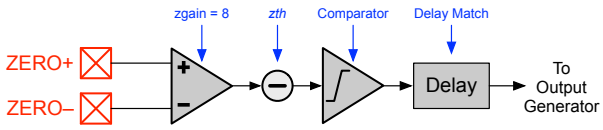


Figure 25: Z Signal Path

A fixed analog gain of 8 is applied to the differential ZERO sensor input. A threshold value is then subtracted from the amplified ZERO signal to set the switching threshold of the comparator. The actual analog threshold level, *zth*, is determined by MAIN_Z in the CFG block in internal memory and has a range of ±450 mV in 30 mV steps. The effective ZERO input threshold therefore has a range of ±56.25 mV in 3.75 mV steps. Note that a negative value must be programmed to MAIN_Z.th to raise the switching threshold.

To ensure that the digital ZERO signal stays in synchronization with the digital sensor angle at all sensor speeds, a delay matching the inherent latency of the AB signal path is introduced into the Z signal path. The delayed and conditioned digital ZERO signal is then sent to the output generator where the Z output is generated and can be synchronized to the AB outputs.

Auto Calibration and Auto Adaption

The iC-TW8 provides sophisticated automatic calibration and adaption features to allow optimal parameter values for the AB signal path to be set and maintained during operation. Auto calibration is used to determine initial parameter values when the iC-TW8 is first commissioned. The CALIB input (pin 5) or commands over one of the serial interfaces may be used to initiate auto calibration.

Auto adaption is used to maintain (adapt) optimal parameter values for the AB signal path during operation. Auto adaption can adjust analog gain, analog offset correction, digital offset correction, digital gain match, and digital phase correction to ensure maximum interpolator accuracy under all operating conditions. Auto adaption is controlled by variables in the ADPT_CORR and ADPT_DETAIL registers in the CFG block of internal memory.

Auto Store

The iC-TW8's Auto Store feature automatically stores adapted parameter values to the EEPROM during operation for use as the startup values at the next power-up. A double buffer is used which ensures a consistent set of AB signal path parameters at every startup. Auto store is controlled by variables in the ADPT_STORE register in the CFG block of internal memory.

Fault Handling

The iC-TW8 provides comprehensive fault handling features and a hardware FAULT output (pin 20) to notify external systems of faults and warnings during operation. Real-time fault status is also available over the serial ports.

The following signal path conditions are monitored and can be configured to activate the FAULT output.

Signal Path Fault/Status Conditions
ADC overflow
ADC underflow
Excessive adaption
Speed limit exceeded
Overspeed warning
Fatal overspeed fault
Excessive filter lag
Excessive AB output lag
1-wire interface time-out
External crystal fault

Table 11: Signal Path Fault/Status Conditions

These conditions can be monitored in real time over either of the serial ports by reading the STAT_SP register in the VAR block in internal memory.

The following EEPROM conditions are monitored and can be configured to activate the FAULT output.

EEPROM Fault/Status Conditions
Communication timeout
Hardware fault
Wrong EEPROM ID
CFG block checksum error
PAR Block 0 checksum error
PAR Block 1 checksum error
PAR_BASE block checksum error
LUT block checksum error
Wrong configuration mode

Table 12: EEPROM Fault/Status Conditions

These conditions can be monitored in real time over either of the serial ports by reading the STAT_EE register in the VAR block in internal memory.

The conditions that activate or latch the FAULT output are determined by bits in the FLT_CFG and FLT_EN registers in the CFG block in internal memory.

ADC and Adaption Quality Monitors

The iC-TW8 provides a sophisticated quality evaluation feature with a hardware STATUS output (pin 19) to monitor signal amplitude and auto adaption during operation. Operation of the quality monitors is controlled by bits in the MON_CFG register in the CFG block in internal memory. Real-time quality status is also available over the serial ports.

The ADC Quality Monitor continuously evaluates the amplitude of the sensor signal at the output of the analog-to-digital converters in the AB signal path and provides a pulse-width modulated signal on the STATUS output that is inversely proportional to relative signal amplitude. Thus the STATUS output can be used to drive an LED whose brightness or color is proportional to sensor signal amplitude for early warning of a possible future failure. The ADC quality monitor levels are determined by variables in the MON_ADC register in the CFG block in internal memory.

The Adaption Quality Monitor continuously evaluates the deviation of the adapted AB signal path parameters from their base values and provides a pulse-width modulated signal on the STATUS output that is proportional to the amount of deviation. Thus the STATUS output can be used to drive an LED whose brightness or color is proportional to the relative amount of adaption that has occurred for early warning of a possible future failure. The adaption quality monitor levels are determined by variables in the MON_OFF, MON_GAIN, and MON_PHASE registers in the CFG block in internal memory.

Electrical Connections

The basic electrical connections for an encoder using the iC-TW8 are shown in Figure 26. Other than the analog sensor, only an EEPROM for parameter storage and a few bypass capacitors are required for operation. PINSEL (pin 22) must be connected to DVSS to select serial configuration mode and either the SPI or 1-wire serial ports must be connected to an appropriate master for configuration. A voltage regulator to supply clean power to the device and a line driver are recommended. A crystal or oscillator (clock driver) is recommended for all but the most cost-sensitive applications.

The iC-TW8 requires a high quality ground and clean power supplies. Analog ground (AVSS = pin 46) and digital ground (DVSS = pin 15) must both be connected on the PCB to a solid ground plane.

Analog power (AVDD = pin 38) and digital power (DVDD = pin 21) inputs must be connected to a low impedance 5V or 3.3V power source, preferably an on-board voltage regulator. In addition, AVDD and DVDD should each have a dedicated 1µF decoupling capacitor placed as close to the power pins on the device as possible.

If the iC-TW8 is used with a line driver (such as the iC-HD7 or iC-HX) or a high frequency FPGA, it is especially important to isolate the iC-TW8 power inputs (AVDD and DVDD) against external noise. In such applications, a dual voltage regulator such as the iC-DC is highly recommended. Note that both AVDD and DVDD must be the same voltage level (5V or 3.3V).

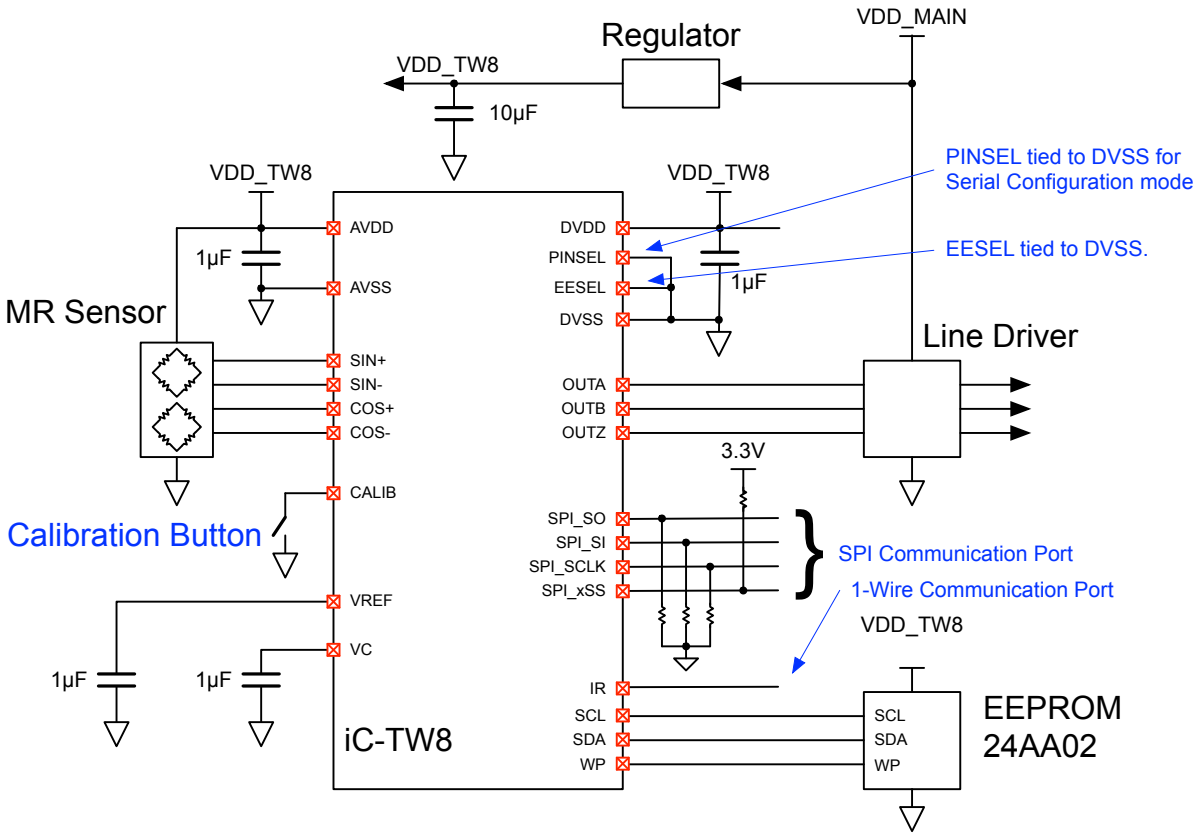


Figure 26: iC-TW8 Electrical Connections (Serial Configuration Mode)

Both pin VC (pin 44) and VREF (pin 45) must be decoupled to ground with 1µF each as shown. The EEPROM and the line driver (if used) must be decoupled according to their respective requirements. The calibration button provides an easy means to activate the auto-calibration feature of the iC-TW8 to set the data path parameters (offset, gain, and phase) to their optimal values. All pins not explicitly shown in Figure 26 should be connected to appropriate levels as indicated in PIN FUNCTIONS.

Providing a Clock

The iC-TW8 supports three clocking modes as indicated in Figure 27. While the iC-TW8 provides an internal oscillator that, while it can be tuned, is not particularly stable over temperature. Thus, an external crystal or other clock source is recommended for all but the most cost-sensitive applications.

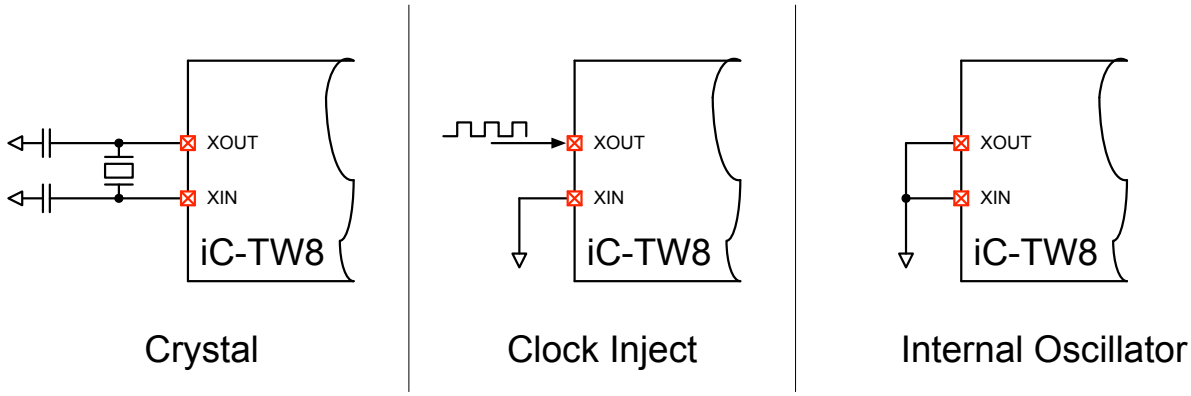


Figure 27: Clocking

An external crystal or oscillator provides the most stable clock signal over time and temperature variations, resulting in the most consistent long-term iC-TW8 performance. Connect an external crystal to XIN (pin 35) and XOUT (pin 34) as shown in Figure 27. Depending on the crystal characteristics, two capacitors to ground might be necessary for stable oscillation. To use an oscillator or ceramic resonator, connect XIN (pin 35) to ground and the oscillator output to XOUT (pin 34) as shown in Figure 27. Maximum crystal frequency (f_{clock}) is 32 MHz with 5V supplies and 24 MHz with 3.3V supplies. Minimum frequency is 6 MHz regardless of supply voltage.

To use the internal oscillator, connect both XIN (pin 35) and XOUT (pin 34) to ground as shown in Figure 27. The iC-TW8's internal oscillator has a nominal frequency of 20 MHz with 5V supplies and 16 MHz with 3.3V supplies and can be tuned (see page 58). However, manufacturing tolerances and changes in temperature can cause large variations in the frequency of the internal oscillator.

Connecting the EEPROM

The iC-TW8 requires a 24xx02 through 24xx16 family I2C EEPROM for storage of sensor calibration data (offset, gain, etc.). The selected EEPROM must operate down to 1.8V; the Microchip 24A02 is recommended.

The EEPROM connects directly to the iC-TW8 via a dedicated I^2C communication channel using SCL (pin 27) and SDA (pin 28) as shown in Figure 28. No external pull-up resistors are required. EESEL (pin 4) of the TW8 must also be connected to DVSS as shown.

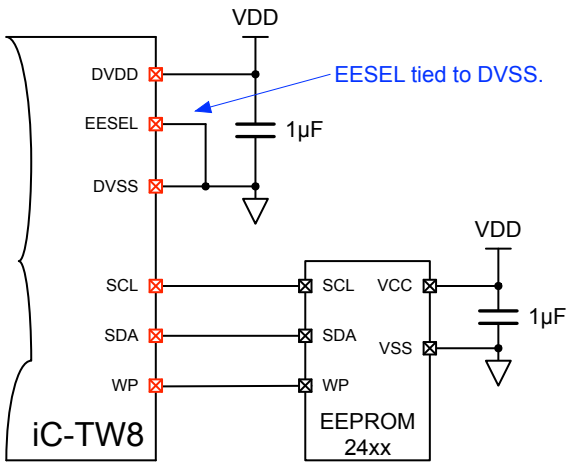


Figure 28: EEPROM Connection

In serial configuration mode, the Write Protect (WP) input of the EEPROM is controlled by the iC-TW8. WP must be disabled via the serial port during calibration to allow the iC-TW8 to write parameter values to the EEPROM. In operation, WP can be left enabled or disabled as required by the application.

Connecting the Incremental Sensor

The iC-TW8 connects directly to magnetic (such as the iC-SM2L or iC-SM5L) or optical sensors providing differential Sin/Cos outputs as shown in Figure 29. Signal amplitude (A) from all four sensor channels must be in the range of 5–350mV as shown in Figure 30 for proper operation with the iC-TW8.

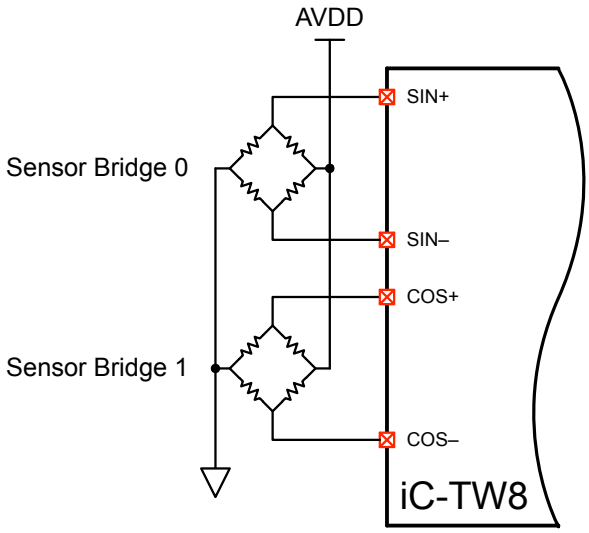


Figure 29: Differential Sensor Connection

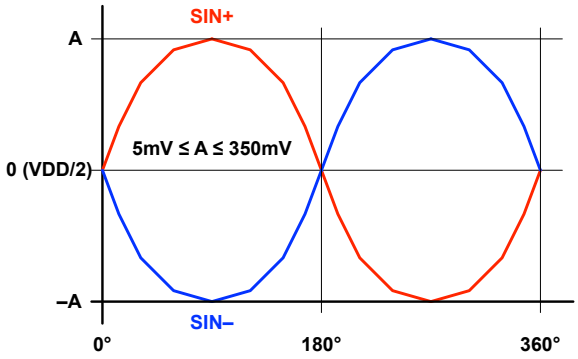


Figure 30: Differential Sensor Signal Inputs

Single-ended Sin/Cos inputs can also be connected to the iC-TW8 by connecting the SIN– and COS– pins to the proper bias voltage as shown in Figure 31. Since the iC-TW8 provides high-impedance signal inputs, a simple resistive voltage divider can be used to generate the required bias voltage. For AMR sensors, it is recommended that the resistors of the bias generator match the AMR sensor bridge resistance to improve power supply noise rejection. No decoupling capacitor should be used.

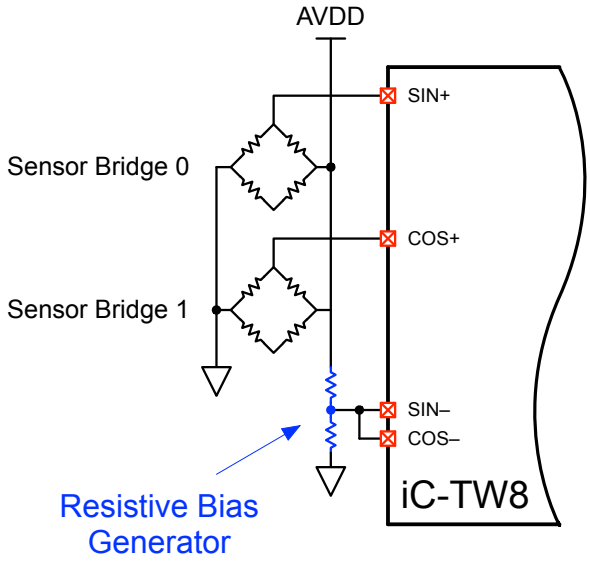


Figure 31: Single-ended Sensor Connection

Differential inputs should be used whenever possible to reject common mode distortions and provide increased signal amplitude and improved signal to noise ratio (SNR).

Regardless of input configuration, the auto-calibration feature of the iC-TW8 is sufficient in most cases to compensate for any signal offset, gain, and phase distortions without requiring any additional external components.

Note: To produce a Z output once every input cycle, connect ZERO+ to AVDD 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-TW8 is required, connect ZERO+ to ground and ZERO- to AVDD (do not allow inputs to float).

Connecting an Index Sensor (If Used)
The iC-TW8 can interface to a wide range of index sensors or gating sources to provide an index or Z output which is synchronized with the AB outputs. Digital Hall sensors are often used for this purpose, as shown in Figure 32.

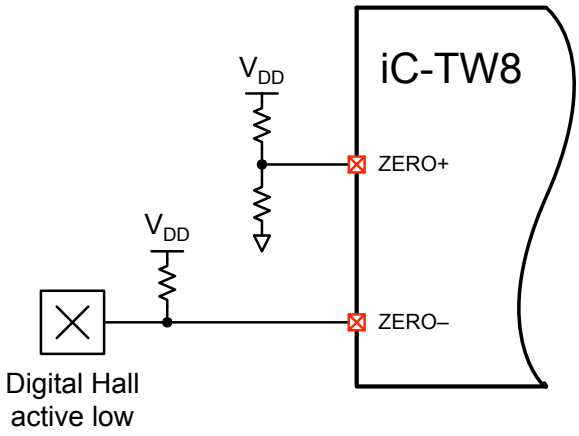


Figure 32: Index Sensor Connection

Digital sensors (Hall, MR, and others) typically provide an active-low signal via an open-drain output that pulls low in the presence of a magnetic field. Connect an active-low (open drain) digital index sensor to the iC-TW8 ZERO– input (pin 48) and connect the ZERO+ input (pin 47) to a resistive voltage divider set to the midpoint of the index sensor output voltage swing to provide good noise rejection. For active-high (open source) digital index sensors, reverse the ZERO+ and ZERO– connections.

Analog-output index sensors, such as MR bridges, can also be used with the iC-TW8 in serial configuration mode as shown in Figure 33.

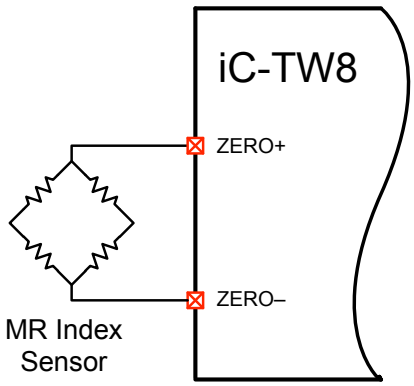


Figure 33: Analog Index Sensor Connection

If no index (Z) output is required, connect ZERO+ (pin 47) to ground and ZERO– (pin 48) to AVDD.

Connecting the ABZ Outputs
In pin configuration mode, the iC-TW8 provides industry-standard quadrature incremental outputs at OUTA (pin 18) and OUTB (pin 17). If an index sensor is connected to the ZERO inputs of the iC-TW8, a Z output synchronous with the AB outputs is also available on OUTZ (pin 16). These outputs can be directly connected to an encoder counter such as the iC-MD.

Alternatively, the ABZ outputs can be connected to a differential line driver (such as the iC-HD7 or iC-HX) as shown in Figure 34. When using a line driver it is especially important to isolate the iC-TW8 from external switching noise using a regulator, as shown.

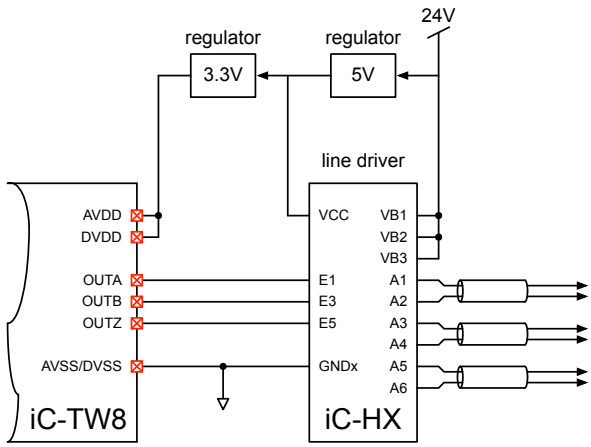


Figure 34: Line Driver Connection

Connecting the Serial Ports
The iC-TW8 contains two serial ports that may be used to access internal registers for configuration. These are a standard SPI (Serial Peripheral Interface) slave port and a 1-wire port, both of which utilize the same communication protocol (see Serial Communication Ports in the Programmer's Reference for the protocol definition). Both ports may be used simultaneously.

The SPI slave port uses a 3-wire or 4-wire full-duplex interface which operates in CPOL = 0 and CPHA = 0 mode only. This means that the base (resting) value of SPI_SCLK (pin 27) is low, SPI_SI (pin 10) is sampled on the rising edge of SPI_SCLK, and SPI_SO (pin 9) is changed on the falling edge of SPI_CLCK.

When using a 4-wire SPI interface, the SPI_xSS input (pin 14) is used to enable the SPI port whenever communication is required. When using a 3-wire SPI interface, the SPI_xSS input must be tied to DVSS.

The 1-wire port uses a single bi-directional data line connected to IR (pin 7) for communication. The bit stream is pulse-width modulated: a 1-bit is encoded as a long high level followed by a short low level, a 0-bit is encoded as a short high level followed by a long low level. A timeout circuit activates a status bit in the STAT_SP register (see Programmer's Reference) in the event of an illegal transaction.

Power-on-Reset
The iC-TW8 contains a built-in power-on-reset (POR) circuit that controls the safe start-up of the device. The internal POR can be configured for two thresholds, 2.7V for use with 3.3V supplies or 4V, which is recommended when operating the iC-TW8 at 5V. Connect xRST (pin 2) to DVDD or leave it unconnected (xRST has an internal 40kΩ pullup resistor to DVDD) to select a power-on-reset threshold of 2.7V. Connect xRST to ground through a 68kΩ resistor to select a threshold of 4V. An external reset source (such as an RC circuit) can also be connected to xRST (pin 2) to directly control the POR behavior.

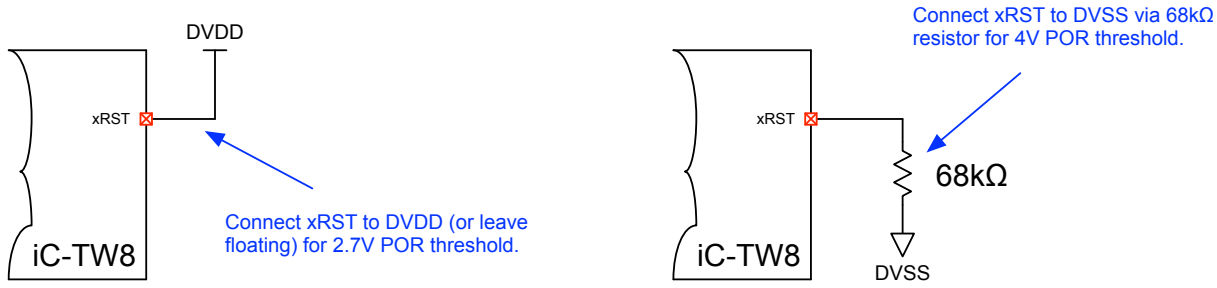


Figure 35: Setting the Power-On-Reset Threshold

Startup
In operation, the startup sequence is initiated when power is applied to the iC-TW8. However, a startup sequence can also be initiated by external hardware connected to the xRST input (pin 2), or by the Restart command (write 0x40 to 0x80B0). See the iC-TW8 Programmer's Reference for more information on executing commands.

After power-up, the iC-TW8's POR circuit monitors the supply voltage and waits until it has reached the appropriate threshold (as determined by the xRST connections). The iC-TW8 then starts up using the internal oscillator (even if a crystal or external clock source is connected) and executes its start-up cycle as outlined here.

1. Read configuration and parameter values from EEPROM.
2. Set proper clock source and frequency.
3. Enable the signal path.
4. Wait for all analog circuitry to settle.
5. Clear all errors.
6. Start ABZ output generation.

If any errors are detected during the start-up cycle, the iC-TW8 does not enable the outputs but goes into an idle state with ERR (pin 20) asserted (low). The amount of time the iC-TW8 waits for the analog circuitry to settle (twait) is configured using the internal variable MAIN_OUT.start.

Serial Configuration Overview

In serial configuration mode, values for all configuration variables must be written to the TW8's internal CFG block registers before the device can be used. The easiest way to accomplish this is to use the free encoder design tool along with the iC-TW8 demo board and the free Graphical User Interface (GUI) software. The encoder design tool—in the form of an Excel spreadsheet—allow entering application parameters in engineering units and then provides configuration register values that can be directly written to the iC-TW8 using the GUI software. The encoder design tool also provides performance limits and graphs and configuration documentation.

The iC-TW8 TW8_1D evaluation board implements the iC-TW8 along with a 24AA02 EEPROM, 24MHz crystal, and an SPI interface (with 3.3V to 5V level shifter if required) to the iC-MB3U-I2C PC adapter for direct communication with the GUI software. A functional prototype can thus be quickly assembled by connecting only a sensor and optional line driver. See the TW8_1D evaluation board documentation for more information. Once the iC-TW8 has been configured, the AB signal path must be calibrated to determine proper values for gain, offset correction, gain match correction, and phase correction. This is most easily done using the Auto Calibration feature of the iC-TW8 to automatically determine optimum parameter values. Alternatively, initial parameter values may be written over the serial interface and allowed to adapt. The adapted parameter values can then be written to the EEPROM.

Rotary Encoder Design Tool

A design tool in the form of an Excel spreadsheet is available to simplify configuration of the iC-TW8 in rotary encoder applications. The design tool requires answers to questions and input of desired application specifications (in engineering units) to produce the desired iC-TW8 configuration (CFG block register values). These values can then be written directly to the iC-TW8 over the SPI or 1-wire serial interfaces to configure the device.

In the design tool spreadsheet, cells with a blue background are for user entry. Cells without a blue background are fixed or calculated by the spreadsheet and cannot be edited. Depending on answers to previous questions, certain blue background cells may be labeled "Ignore". In this case, the entered value has no effect on device operation. Register values and internal variables calculated from user input are shown in column F or I (typically) on the spreadsheet tabs. These correlate directly to the register and variable descriptions in the iC-TW8 Programmer's Reference.

When possible, legal responses to questions are provided (in parentheses). Entering values other than the given responses may cause "Error" to appear for the corresponding variable value in column H (typically). In this case, the Excel error "VALUE!" is displayed in the corresponding CFG register in the CFG Block tab.

General Configuration
Click on the General Configuration tab and start by entering the operating voltage (Vdd) as 5.0 or 3.3. If you are using an external crystal or other clock source (recommended), enter its frequency. The operating voltage and crystal frequency affect the current drawn by the iC-TW8, which is calculated and shown for reference. The signal path clock frequency (f_{core})—which affects the maximum encoder speed and the filter response—is shown for reference. In general, use the highest possible frequency unless there is a need to minimize current.

If you are not using an external oscillator or clock source, enter the desired Clock Divider and Clock Tuning Value for the internal oscillator.

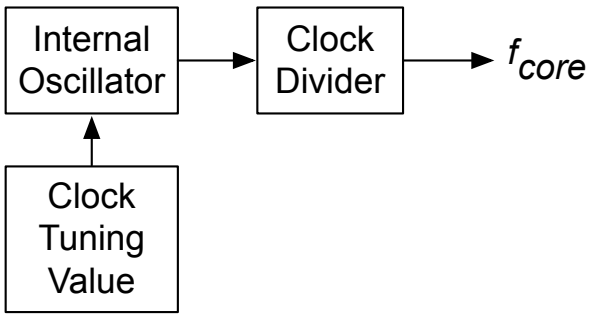


Figure 36: Internal Oscillator Configuration

The nominal frequency of the internal oscillator, shown for reference, depends on the operating voltage. The actual frequency, however, varies from unit to unit and over temperature. To compensate for this, the oscillator frequency may be reduced by a factor of 2 or 4 using the Clock Divider and also tuned (raised) using the Clock Tuning Value (see page 58). For initial configuration, enter the desired divider and a tuning value of 0. The resulting signal path clock frequency (f_{core}) is shown for reference. Changing the Clock Tuning Value changes the nominal calculated f_{core}. The actual f_{core} in any given chip with the given Clock Tuning Value will vary from the nominal calculated value due to normal production tolerances.

The iC-TW8 can be configured to use pull-up and pull-down resistors on the sensor Sin/Cos inputs to enhance fault detection in the event of a floating sensor input.

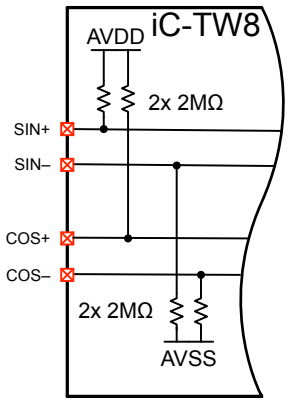


Figure 37: Input Pull-up Resistors

When input pull-ups are used and if a sensor input becomes disconnected, that input is immediately pulled to a level that causes an ADC fault (see page 35).

If the iC-TW8 WP (Write Protect) output (pin 29) is connected to the EEPROM WP input (see Connecting the EEPROM on page 37), then the EEPROM can be configured to be locked (protected) or unlocked during operation. Usually the EEPROM should be locked during operation. To temporarily unlock the EEPROM to store configuration and calibration data, disable EEPROM Write Protection in the Miscellaneous pane of the Settings tab in the iC-TW8 GUI software or set the Unlock EEPROM bit in the RB_TEST1 register in the RB Block of internal memory using serial port commands (see Programmer's Reference).

Next, enter the desired Output Mode. AB is the usual setting and configures the iC-TW8 to produce industry-standard AB quadrature outputs. Serial and PWM are alternate output modes that are described on pages 55 and 56.

AB Startup Mode determines how the AB outputs behave when power is applied to the iC-TW8.

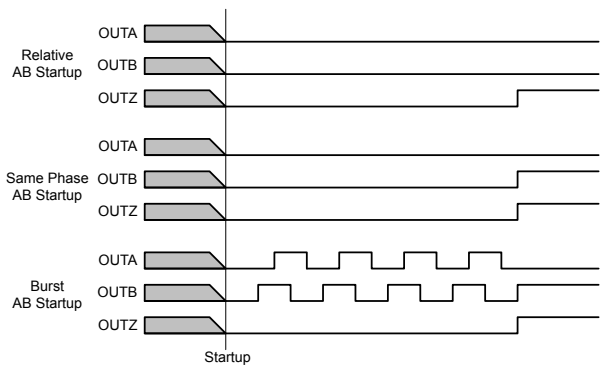


Figure 38: AB Startup Mode

In Relative startup mode, the A and B outputs are always in the same states on startup regardless of the sensor input position (angle). In Same Phase startup mode, the phase relationship of the A, B, and Z outputs is the same after every startup as long as the post-AB divider is not used and the actual interpolation factor, *inter*, is an integer. Absolute Burst startup mode is like Same Phase, except that the sensor position (angle) within one input cycle is counted out on the A and B outputs at startup. Same Phase startup mode is the usual selection and provides operation similar to that of a non-interpolated encoder.

Enter the desired Counting Direction (phase relationship) for the AB outputs when sensor angle is increasing. While it may be impossible to predict the correct value during design, the counting direction can always be changed during testing to ensure correct phasing.

Enter the desired amount of time for the iC-TW8 to wait during startup for its analog circuitry to settle (see Startup on page 40). The design tool returns the closest possible (actual) startup delay using the current crystal frequency.

The CLOCK (pin 6) and FRAME (pin 8) outputs can be activated if necessary for special applications (see Using PWM Output Mode on page 56). In most cases, disable the CLOCK and FRAME outputs.

AB Configuration

In the AB Configuration tab, enter the input resolution produced by the wheel or magnet and sensor. This value should be the number of Sin/Cos cycles per mechanical revolution of the wheel, magnet, or optical disc produced by the sensor connected to the iC-TW8. Next enter the desired output resolution of the encoder as an integer equal to the number of AB cycles per revolution (CPR). This is also sometimes called PPR (pulses per revolution) and is equivalent to the number of "lines" in a non-interpolated optical encoder. Fractional output resolutions are possible using the iC-TW8, but are not supported by the design tool.

With these values entered, the design tool calculates and displays the minimum required iC-TW8 interpolation factor, the minimum post-AB divider, and the effective overall interpolation factor. In most applications, the post-AB divider is not required, and the Minimum Divider value should be 1. In general, this will be true if the Desired Output Resolution is an integer multiple of the Input Resolution. Specifically, Minimum Divider = 1 if

$$4 \leq \frac{4 \cdot \text{Desired Output Resolution}}{\text{Input Resolution}} \leq 16\,384$$

and

$$\frac{4 \cdot \text{Desired Output Resolution}}{\text{Input Resolution}} \text{ is an integer.}$$

For example, with an input resolution of 32 Sin/Cos cycles per revolution, a desired output resolution of 16,384 CPR results in a minimum interpolation factor of 512 and a minimum divider of 1 since

$$\frac{4 \cdot 16\,384}{32} = 2\,048$$

In this case enter 1 for the divider override factor and then actual interpolation = effective interpolation = minimum interpolation and actual output resolution = desired output resolution. This is the usual case.

If it is desired to achieve the same output resolution using an input resolution of 24 cycles per revolution, however, a minimum interpolation of 2 048 and a minimum divider of 3 are required since

$$\frac{4 \cdot 16\,384}{24} = 2\,730.66...$$

If the minimum divider ≠ 1, it is not possible to synchronize the Z output with the states of the A and B outputs.

See Using the Post-AB Divider on page 57. Impossible combinations of input and output resolution are flagged with red warnings.

Next, enter the desired maximum speed (speed limit) for the encoder in RPM. The spreadsheet calculates and displays the closest actual speed limit that can be implemented for the chosen output resolution using the current crystal. The calculated actual speed limit is shown in RPM, AB channel frequency, and AB edge frequency. Exceeding the actual speed limit can be configured to activate the FAULT output (pin 20). See Fault Configuration on page 47 for more information.

For an enclosed encoder the maximum speed is usually dictated by the bearings used. For a modular encoder (without bearings) or to use the full speed capability of the iC-TW8, enter an arbitrarily large value (such as 1,000,000 RPM) and the design tool will default to the highest possible speed. The calculated actual speed limit is affected by the crystal frequency and output resolution.

Because edge separation (the time between two consecutive AB edges) is the reciprocal of the AB output frequency, the design tool also calculates this value. Note that if the post-AB divider is used, the minimum edge separation at standstill (dither) is different than the minimum edge separation when running at speed.

Next, enter the desired maximum adaption speed in RPM. This is the speed above which auto adaption of the AB signal path parameters ceases. The actual maximum adaption speed depends on the input resolution and crystal frequency. In general, set the desired maximum adaption speed to an arbitrarily high value (such as 1,000,000 RPM) to enable auto adaption at all speeds. See Auto Adaption Configuration on page 47 for more information.

The warning fault speed depends on crystal frequency and input resolution and can be configured to activate the FAULT output. The warning fault speed is always half the fatal fault speed. The fatal fault speed is the ultimate speed limit of the iC-TW8 at the selected input resolution using the current crystal. Exceeding the fatal fault speed always activates the FAULT output. See Fault Configuration on page 47 for more information.

The design tool shows the resulting speed/resolution performance in a chart at columns M – T. The X (Resolution) axis shows the range of output resolutions that are possible using the selected input resolution. The chart shows the resulting maximum speed, maximum adaption speed, warning fault speed and fatal fault speed at a given output resolution.

Finally, enter the desired hysteresis in equivalent AB output edges to prevent output dithering at standstill. The design tool calculates and displays the closest possible (actual) hysteresis with the current input and output resolution. Also displayed is the equivalent hysteresis in AB output cycles, mechanical degrees (°m) and arc-minutes of rotation, and input Sin/Cos cycle degrees. Desired hysteresis can be less than 1 AB output edge. For best accuracy, choose the smallest value that produces stable AB outputs with no sensor motion. Experimentation may be required to determine the optimal value.

The complete range of output resolutions using the selected input resolution is shown numerically in cells B39 – B41. A complete list of output resolutions and the corresponding interpolation factor required to achieve them is shown in the All Resolutions tab for easy reference with respect to completed designs. For example, with an input resolution of 32 cycles per revolution, an output resolution of 16 384 CPR requires an interpolation factor of 512, as shown previously. Using this same input resolution, the All Resolution tab shows that an output resolution of 10 000 is also possible by choosing an interpolation factor of 312.50 (row 1257).

Filter Configuration
In the Filter Configuration tab, enter the desired filter mode for the AB signal path filter and then enter one of the recommended filter instances. The design tool confirms the actual filter mode and approximate noise and jitter bandwidth for the selected filter instance at the current crystal frequency. If "Not Defined" appears for the actual filter mode, the entered instance is not one of the recommended ones. Non-recommended instances may cause unstable filter response. See Filter on page 32 for details on filter modes and instances.

The following filter modes are available.

iC-TW8 Filter Modes	
Mode	Description
P	Low-pass filter with lag.
PI	Critically-damped low-pass filter with no filter lag.
Lag Recovery	Under-damped low-pass filter with no filter lag and reduced latency lag.

Table 13: iC-TW8 Filter Modes

P Mode is the simplest mode; position lag at constant sensor input velocity is directly proportional to the sensor velocity and the filter instance value. There is no position or speed overshoot in P mode.

PI Mode eliminates filter lag at constant sensor input velocity at the expense of critically damped filter response. In PI mode, filter lag at constant velocity is eliminated, but position and speed overshoot are proportional to the filter instance value.

Lag Recovery Mode reduces latency-induced position lag at constant sensor input velocity at the expense of under-damped filter response. In lag recovery mode, position lag at constant velocity due to interpolator latency is reduced, but filter response is under-damped and position and speed overshoot are proportional to the filter instance value.

In general, use the lowest instance value for the desired filter type. This gives the fastest response, minimum lag, and minimum position and speed overshoot. Another instance may be used if additional noise and jitter filtering is required.

The design tool shows the encoder lag in AB edges per kRPM and mechanical degrees (°m) of encoder rotation at the chosen maximum speed for the selected filter instance. For a given filter instance, lag is directly proportional to encoder speed.

The design tool also shows the position and speed overshoot resulting from the selected filter instance. Position overshoot (shown in AB edges and mechanical degrees of encoder rotation) is proportional to the entered maximum acceleration and is typically noticed on stopping (deceleration). At low acceleration values and/or low encoder resolutions, position overshoot may be less than one AB edge and thus totally undetectable during encoder operation.

Speed overshoot is calculated as a percent of speed. The actual amount of speed overshoot in RPM is proportional to encoder speed. Speed overshoot is typically noticed at the end of the acceleration ramp (the beginning of the constant velocity portion of the motion profile).

Performance graphs of encoder lag in AB edges and response to a trapezoidal motion profile with constant acceleration, velocity, and deceleration are shown to allow observing the effect of the different filter instances. Scroll to the right on small computer screens to see the performance graphs.

A single parameter (maximum acceleration) is used to generate the filter response graphs. Enter a value in radians per second^2 that represents the maximum expected acceleration of the encoder. The design tool calculates the equivalent acceleration in units of kRPM (1000 RPM) per second and time to reach the chosen maximum speed, which may be more easily understood. The acceleration value is used only to generate the performance graphs, it does not affect the configuration of the iC-TW8. The motion profile is generated by accelerating for 1000 ADC samples at the entered acceleration, running at the attained speed for 1000 ADC samples, and decelerating at the entered acceleration for 1000 ADC samples.

The maximum acceleration (α_m) of a motor in radians per second^2 may be calculated knowing its stall torque (T_s) in Newton meters and its rotor inertia (J_r) in kilogram meters^2 using the following formula.

$$\alpha_m = \frac{T_s}{J_r}$$

For example, the Maxon EC 90 flat 90W motor with the 24V winding (order number 323772) has a rotor inertia of 3,060 gcm^2 and a stall torque of 4,670 mNm. Thus its no-load maximum acceleration in radians per second^2 is

$$\alpha_m = \frac{T_s}{J_r} = \frac{4670}{3060} \cdot 10,000 = 15,261$$

Using this value for the maximum acceleration parameter in the design tool will then show the performance of

the iC-TW8 when used in an encoder for feedback with this motor.

The graphs below show typical encoder lag and motion profile performance of the iC-TW8 using normal filtering and lag recovery at 15,261 radians per second^2 maximum acceleration and a resolution of 16,384 AB cycles per revolution.

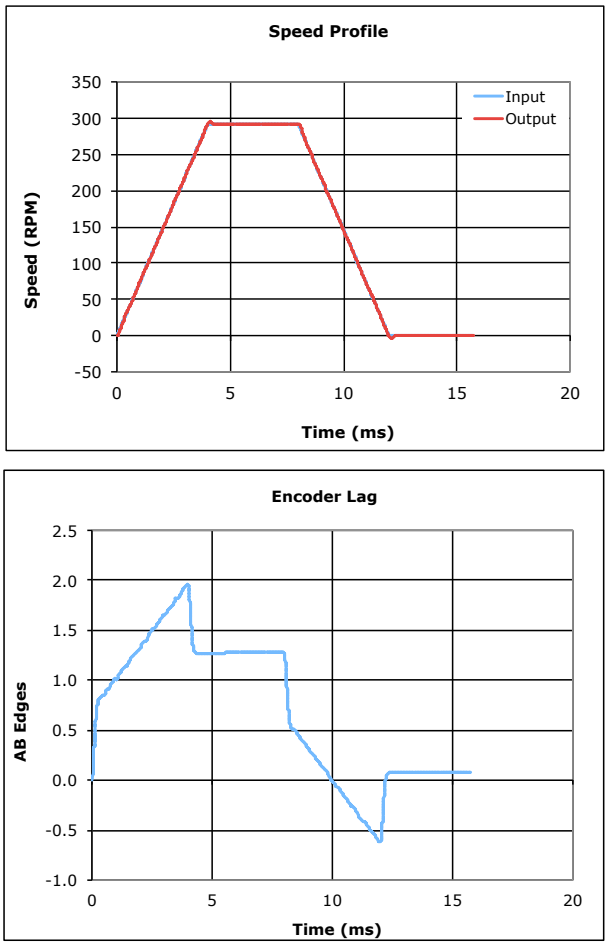


Figure 39: Typical Encoder Performance Graphs

Z Configuration

In the Z Configuration tab, enter Yes or No as to whether an index sensor is connected to the ZERO+ and ZERO- inputs. If an index sensor is not used, the ZERO inputs of the iC-TW8 should be connected to AVDD and AVSS as explained on page 39 and the remaining entries on this sheet can be ignored. Not using an index sensor allows higher speeds with high interpolation factors.

If an index sensor is used, enter the desired width of the Z output pulse in AB edges. The actual Z output pulse width is confirmed; it is always equal to the desired width unless the post-AB divider is used (see page 57) or a fractional desired width is entered. The design tool shows the equivalent Z output pulse width in AB output cycles, mechanical degrees of rotation (°m), arc-minutes of rotation, pre-divider edges, and sensor input cycle degrees for reference.

Next, enter the desired polarity for the Z output as either active high or active low. If the post-AB divider is not used and the actual interpolation factor, *inter*, is an integer, the states of the A and B outputs at the beginning of the Z output (*zpos*) can be entered to specify the desired synchronization of the A, B, and Z outputs.

Set both the Z position (*zpos*) and desired ZERO threshold to 0 initially. Proper values for these parameters are determined when the Z signal path is calibrated (see page 51).

Finally, configure the iC-TW8's internal 32-bit position counter to be reset to 0 by the Z output or not at all. If "On zpos" is selected, the counter is reset at *zpos* whenever the ZERO output is active.

Auto Adaption Configuration
In the Auto Adaption Configuration tab, select which parameters are automatically adjusted (adapted) during operation to maintain low interpolation error and jitter. Analog gain, analog offset, digital offset, digital gain match, and phase may be individually selected although there are certain dependencies that cause the design tool to override the entered selections. In this case, a red warning is displayed by the design tool.

It is recommended to disable analog gain and offset adaption and enable only digital offset, gain match, and phase adaption during operation. In this case, analog gain and offset remain at their calibrated settings during operation. This prevents changes in analog gain and offset (which can be quite large) from causing disturbances in the interpolated output during operation. Analog gain and offset adaption may be enabled if large changes in sensor signal amplitude or offset are expected during operation.

If digital offset adaption is enabled, enter the desired digital offset tolerance and the design tool confirms the closest actual tolerance that can be achieved. Digital offset tolerance is the amount of un-corrected offset that the iC-TW8 tolerates before auto adaption corrects it. Small tolerance values must be used for highest accuracy, but also cause offset corrections to occur more frequently during operation and thus may lead to increased output jitter. Digital offsets of less than the entered tolerance are not corrected. In general, set the digital offset tolerance to the smallest value that prevents dithering of the digital offset correction value during normal operation.

If digital gain match adaption is enabled, enter the desired digital gain match tolerance and the design tool confirms the closest actual tolerance that can be achieved. Digital gain match tolerance is the amount of un-corrected gain mismatch between the sensor Sin and Cos channels that the iC-TW8 tolerates before auto adaption corrects it. Small tolerance values must be used for highest accuracy, but also cause gain corrections to occur more frequently during operation and may lead to increased output jitter. Digital gain mismatch of less than the entered tolerance is not corrected. In general, set the digital gain match tolerance to the smallest value that prevents dithering of the digital gain match correction value during normal operation.

The Correction Configuration controls how the auto adaption corrections are applied to the AB signal path parameters during operation. First, select linear or exponential correction mode. Linear correction mode is the recommended selection and provides the smoothest correction; parameter corrections are applied one increment per correction cycle. This results in the least disturbance to the interpolator output when the auto adaption corrections are made.

Faster correction is available in exponential correction mode which allows selection of the desired correction step size. Small correction steps correct 25% of the parameter error every correction cycle; medium correction steps correct 50% of the parameter error every correction cycle; large correction steps correct 75% of the parameter error every correction cycle. Experimentation may be required to determine the optimal correction configuration.

The correction timebase determines the rate at which auto adaption corrections are applied. In general, enter 0 for the desired correction timebase and the design tool confirms the closest possible actual timebase using the current crystal. This results in auto adaption corrections being applied as quickly as possible. Larger timebase values provide slower response. Lastly, select whether or not the auto adapted parameter values are written to the EEPROM during operation (auto store) and used at the next restart. If auto store is enabled, enter the desired digital offset and gain match thresholds; the design tool confirms the closest available (actual) values. These thresholds are the levels of change in the respective parameters that must be achieved by auto adaption to cause new values to be written to EEPROM. Since most EEPROMS are only guaranteed for a finite number of write cycles, these thresholds must be set high enough to minimize unnecessary EEPROM writes. If auto store is disabled, these thresholds are ignored.

FAULT Pin Configuration
In the FAULT Pin Configuration tab, select whether the FAULT output (pin 20) is active high or active low. Active low is the recommended since this is the startup default of the iC-TW8. If active high is chosen, the FAULT output will be active (high) at startup until configured to active low polarity during the startup process.

The FAULT output can be configured to stay active for a time after a fault condition has cleared to enhance observation of transient fault conditions. The amount of time by which the output is prolonged is fixed and inversely proportional to crystal frequency. Next, choose whether or not the AB outputs should be stopped when the FAULT output is active.

iC-TW8 fault conditions can be configured to either activate or latch the FAULT output. Faults which do not latch the FAULT output only activate the FAULT output for the time during which the condition is active (subject to prolonging, as explained above). Faults which latch the FAULT output are cleared at restart or may

be cleared by writing to the STAT_SP, STAT_EE, or FLT_STAT registers using one of the serial ports (see Programmer's Reference).

A crystal fault is active if the iC-TW8 is configured to use an external crystal or clock source (see page 19) and the expected external signal is not present. In this case, the iC-TW8 reverts to using its internal oscillator. Once an external clock signal becomes available, the iC-TW8 switches back to using the external oscillator and clears the crystal fault. Clearing the crystal fault also de-activates the FAULT output if the crystal fault is not configured to latch.

An EEPROM fault is active if the EEPROM has not been initialized, there is a hardware or communication problem with the EEPROM, or if any of the internal checksums are invalid. It is recommended to latch EEPROM faults as they can result in un-defined startup conditions.

A fatal operational fault is active if the instantaneous sensor input velocity is greater than the fatal fault speed shown on the AB Configuration tab (see page 43), the filter lag is too large (see page 32), or the AB output is more than half an input cycle behind the sensor input position due to prolonged operation above the maximum speed shown on the AB Configuration tab (see page 43). It is recommended to configure fatal operational faults to latch the FAULT output as they can result in erroneous interpolator output.

An operational warning is active if the instantaneous sensor input velocity is greater than the maximum speed or warning fault speed shown on the AB configuration tab (see page 43). Interpolator output is still correct while an operational warning is active. Maximum speed is determined by the desired speed limit (see page 43) while the warning fault speed is always half the fatal fault speed. If operational warnings are treated as faults, they can be configured to activate or latch the FAULT output. If operational warnings are not treated as faults, they are ignored. It is recommended to configure operational warnings to be treated as faults but not to latch the FAULT output so that the FAULT output is de-activated when the condition is resolved.

An ADC fault is active if the signal input level to the analog-to-digital converters in the AB signal path (see page 31) is outside defined operational limits. The upper signal level (called ADC overflow) is fixed at 1.5V, which is 106% of the nominal signal level into the ADC.

The lower signal level (called ADC underflow) can be set between 10% and 100% of nominal signal level or left at the default (10%). Enter the desired ADC underflow level and the design tool confirms the closest possible (actual) level. Less resolution for the ADC underflow level is available if the ADC quality monitor is used (see page 49).

ADC overflow and underflow faults are not fatal, but interpolation accuracy is reduced when operating under these conditions. Therefore, it is recommended not to latch ADC faults, but to stop auto adaption when an ADC fault is active. This prevents auto adaption from overcompensating for an out-of-range input signal and de-activates the FAULT output when the input signal is back in range.

If adaption faults are used, enter the desired maximum adaption values for offset, gain match, and phase. The design tool confirms the closest possible (actual) values. When any of these levels is exceeded, adaption of the corresponding parameter stops and an adaption fault is activated.

If adaption faults are not used, the design tool sets the adaption limits to their highest possible (actual) values to avoid ever stopping auto adaption. Since an adaption fault always latches the FAULT output and stops auto adaption if adaption faults are used, it is recommended not to use adaption faults without precise knowledge of how the parameters adapt under all operating conditions. Adaption faults are cleared at restart when valid startup values for the parameters are read from the EEPROM.

Finally, the design tool summarizes the conditions that activate the FAULT output, the conditions that latch the FAULT output, and the conditions that are ignored. Latched faults are cleared by restarting the iC-TW8 or by writing to the STAT_SP, STAT_EE, or FLT_STAT registers using one of the serial ports (see Programmer's Reference).

ADC Quality Monitor
If Set ADC Underflow? = No in the FAULT Pin Configuration tab, the ADC Quality Monitor tab can be ignored in its entirety. If Set ADC Underflow? = Yes in the FAULT Pin Configuration tab, use the ADC Quality Monitor tab to enable or disable the ADC quality monitor. If the ADC quality monitor is not used, the remainder of the ADC Quality Monitor tab may be ignored.

If the ADC quality monitor is used, enter the desired ADC quality threshold level. This is the ADC level in percent of nominal signal input amplitude that is considered to be the boundary between acceptable and marginal signal level for the application. The design tool shows the closest possible (actual) ADC quality threshold level and the desired and actual ADC underflow fault levels from the FAULT Pin Configuration tab.

In operation, the iC-TW8 continuously calculates the variable QM_ADC as a value between 0 (ADC level > ADC quality threshold) and 255 (ADC level $\leq$ ADC underflow level). Between the quality threshold level and the underflow level, QM_ADC is inversely proportional to the ADC level as shown in the design tool graph. Thus, QM_ADC indicates the relative quality (signal amplitude) of the input signal. QM_ADC is a component of the PWM signal that drives the STATUS output (pin 19) for external quality monitoring; it may also be read directly from the VAR block in internal memory via the serial ports (see Programmer's Reference).

Adaption Quality Monitor
If Use Adaption Faults? = No in the FAULT Pin Configuration tab, the Adaption Quality Monitor tab can be ignored in its entirety. If Use Adaption Faults? = Yes in the FAULT Pin Configuration tab, use the Adaption Quality Monitor tab to enable or disable the adaption quality monitor. If the adaption quality monitor is not used, the remainder of the Adaption Quality Monitor tab may be ignored.

If the adaption quality monitor is used, enter the desired offset, gain match, and phase adaption quality threshold levels. These levels are the cumulative amount of adaption that is considered to be the boundary between acceptable and marginal operation for the application. The design tool shows the closest possible (actual) quality threshold levels and the desired and actual fault levels from the FAULT Pin Configuration tab.

In operation, the iC-TW8 continuously calculates the variable QM_ADAPT as

$$\text{QM\_ADAPT} = \mathit{moffs} + \mathit{moffc} + \mathit{mgain} + \mathit{mphase}$$

where *moffs*, *moffc*, *mgain*, and *mphase* are values between 0 (total adaption < adaption quality threshold) and 255 (total adaption $\geq$ adaption quality threshold) of the Sin channel offset, Cos channel offset, gain match, and phase adaption quality monitors respectively. Between the quality threshold level and the fault level, *moffs*, *moffc*, *mgain*, and *mphase* are proportional to the total cumulative amount of adaption of the respective parameter as shown in the design tool graphs.

QM_ADAPT is the sum of these four values and thus indicates the relative quality (amount) of parameter adaption. QM_ADAPT is a component of the PWM signal that drives the STATUS output (pin 19) for external quality monitoring; it may also be read directly from the VAR block in internal memory via the serial ports (see Programmer's Reference). When QM_ADAPT reaches 255, an adaption fault is activated.

STATUS Pin Configuration
If neither the ADC quality monitor nor the adaption quality monitor is used, the STATUS Pin Configuration tab may be ignored in its entirety. If either of the quality monitors is used, use the STATUS Pin Configuration tab to set the polarity and mode of the STATUS output (pin 19). The STATUS output is a pulse-width modulated signal whose duty cycle is proportional to the outputs of the enabled quality monitors. PWM polarity determines whether the duty cycle increases of decreases for an increasing value from the quality monitors while PWM mode determines whether the STATUS output response is linear or logarithmic. Logarithmic mode is useful when driving an LED from the STATUS output so that the intensity or color is proportional to the PWM duty cycle. The design tool graph shows the configured response.

LUT Configuration
The sensor distortion lookup table (LUT) can be enabled or disabled in the LUT Configuration tab. If the LUT is not used, the remainder of the LUT Configuration tab may be ignored. See Using the LUT on page 59 for information on configuring the LUT.

CFG Block
The CFG Block tab shows the complete iC-TW8 configuration (as determined by the preceding tabs) as stored in the CFG Block in internal memory (see Programmer's Reference). CFG Block register values are shown in decimal, hexadecimal, and binary as a reference for troubleshooting.

GUI Hex
The GUI Hex tab shows the complete iC-TW8 configuration (as determined by the preceding tabs) in the format used by the Graphical User Interface software Hex Editor. After using the design tool to configure the iC-TW8, type these values into the corresponding address locations in the GUI Hex Editor tab to transfer the configuration to the actual device.

Auto Calibration
Once the iC-TW8 has been configured, the signal path must be calibrated to determine proper values for gain, offset correction, gain match, and phase correction. This is most easily done using the Auto Calibration feature of the iC-TW8 to automatically determine optimum values for these parameters. Alternatively, initial parameter values may be written to the parameter registers using the serial interface and allowed to adapt. The adapted parameter values must then be written to the EEPROM for use after subsequent startups.

Auto calibration is initiated by pulling CALIB (pin 5) to DVSS. A pushbutton switch connected between CALIB and circuit ground (as shown in Figure 26 on page 36) is an easy way to achieve this in series production. With CALIB held low, provide sensor input of a few hundred Sin/Cos cycles and the iC-TW8 "tunes" the analog gain, analog offset correction, digital offset correction, digital gain match, and digital phase correction parameters to provide lowest error and jitter in the interpolated AB outputs.

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.

After providing sufficient input signals, release the CALIB input (it is pulled high by an internal pull-up resistor) and the iC-TW8 immediately stores all calibration parameters in the external EEPROM. These auto-calibrated values are then used for subsequent startups (see page 40).

Auto Calibration Using the GUI
The iC-TW8 can also be auto calibrated using the GUI software instead of the CALIB input. With the GUI running and connected to the iC-TW8, make sure the Enable Write Immediately checkbox (in the lower left corner of the GUI screen) is checked so that changes made in the GUI are immediately written to the chip. In the Monitor tab, make sure the ADC Quality Monitor Enable and Adapt Quality Monitor Enable checkboxes are unchecked. This disables the quality monitors so they do not interfere with initial calibration.

In the Signal Path tab, make sure the Enable Continuous Parameter Read checkbox in the Signal Path pane is checked so that the parameter values can be observed during auto calibration. In the Auto Adaption pane, set the Adaption Gain Tolerance and Adaption Offset Tolerance to their lowest values, 1.0005 (0x00) and 0.49 mV (0x00) respectively, to ensure the most accurate calibration. Finally, check the OFF/ON checkbox at the top of the Auto Adaption pane to enable auto adaption of all parameters.

Now, provide sensor input of a few hundred Sin/Cos cycles and the iC-TW8 "tunes" the analog gain, analog offset correction, digital offset correction, digital gain match, and digital phase correction parameters to provide lowest error and jitter in the interpolated AB outputs. 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.

After the parameters in the Signal Path pane have converged, click the Write EEPROM button in the lower right corner of the GUI screen to store all the calibrated parameter values in the external EEPROM. These auto-calibrated values are then used for subsequent startups (see page 40). After calibration, the auto adaption enable and tolerance parameters in the Auto Adaption pane of the Signal Path tab may be set back to their desired operational values. The quality monitors may also be enabled and the thresholds and limits set as desired in the Monitor tab.

Calibrating the Z Signal Path

The Z signal path must be calibrated to generate a Z output pulse at the proper location within a sensor input cycle and that is properly synchronized to the AB outputs. The Z output is configured using the design tool, but the Z signal path must be calibrated manually, if required. When the post-AB divider is not used and the interpolation factor, *inter*, is an integer, once the Z signal path is calibrated, one and only one Z output pulse, aligned as desired with the A and B outputs, is produced per revolution. For example, if *zwidth* = 1 (edge), OUTA is high at *zpos*, and OUTB is low at *zpos*, the following output synchronization results.

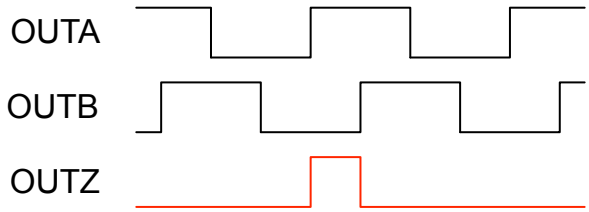


Figure 40: ABZ Output Synchronization

If the post-AB divider is used or the actual interpolation, *inter*, is not an integer, the configured ABZ output synchronization is not maintained at every Z output. See page 57 for information on using the Z output with the post-AB divider.

In general, there are three cases, depending on the type of index sensor connected to the ZERO inputs of the iC-TW8:

1. No index sensor.
2. Digital output index sensor.
3. Analog output index sensor.

These three cases each are discussed separately following.

No Index Sensor

The simplest case is where no index sensor is used. This is useful in applications where the incremental sensor produces only one input cycle per revolution, such as with an on-axis analog Hall effect device. In this case, one Z output is produced for every input cycle and it is only necessary to program the desired location, length, and synchronization of the Z output pulse using the design tool (see page 46).

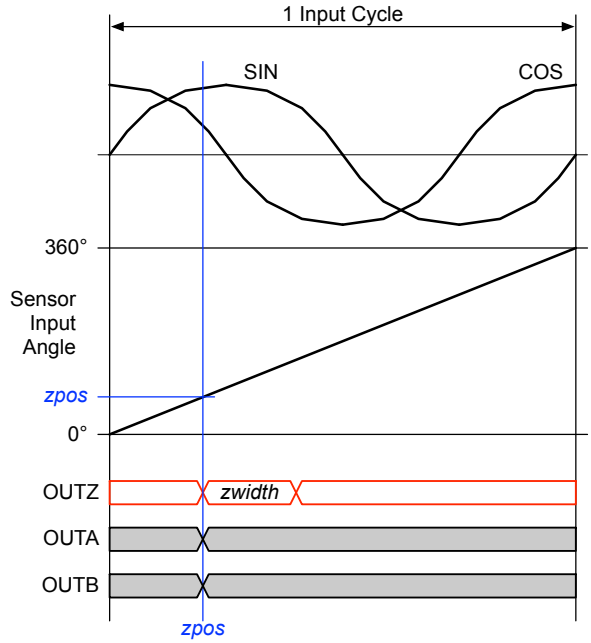


Figure 41: Z Output with No Index Sensor

In the design tool Z Configuration tab, answer "Yes" to the "Use Index Sensor?" question even though the ZERO inputs are tied to AVDD and AVSS (see page 39) and set the Desired ZERO Threshold to 0. Configure the remaining parameters as required and load the complete CFG bock to the iC-TW8 using the GUI or serial port commands (see Programmer's Reference).

Digital Output Index Sensor

In applications where the incremental sensor produces multiple input cycles per revolution—such as optical and MR magnetic sensors—an index sensor must be connected to the ZERO inputs to qualify or "gate" the Z output to ensure one and only one Z output pulse per revolution. In this case, it is necessary to align the Z output pulse with the ZERO input signal and to program the desired length and synchronization of the Z output pulse using the design tool (see page 46).

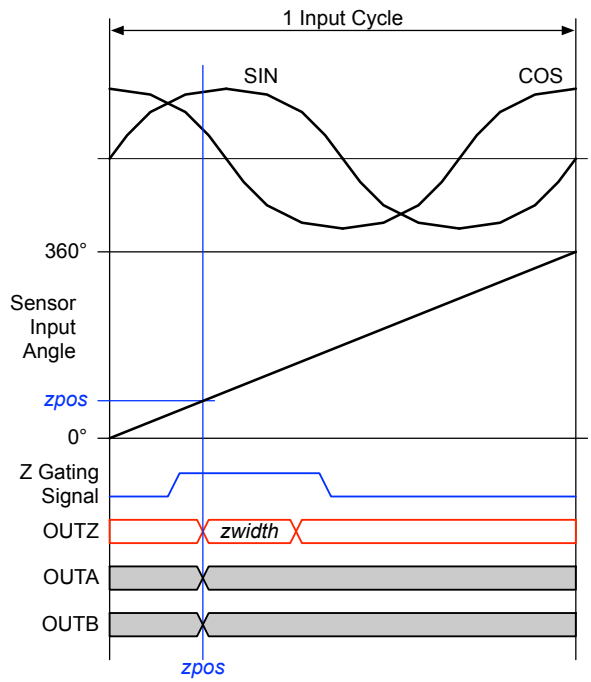


Figure 42: Z Output With Digital Index Sensor

First, connect a digital output index sensor to the iC-TW8 as explained on page 52. Then, in the Z Configuration tab of the design tool, set the Desired ZERO Threshold to 0, configure the remaining parameters as required (see page 46), and load the complete CFG bock to the iC-TW8 using the GUI or serial port commands (see Programmer's Reference).

To calibrate *zpos*, it is necessary to observe both the Z output and the internal Z gating signal simultaneously. To observe the internal gating signal, enable Index Gating Test Mode in the Test Mode pane of the Settings tab in the GUI or set the Enable Z Test Mode bit (ztest) in the RB_TEST1 register in the RB Block of internal memory using serial port commands (see Programmer's Reference). This forces the internal Z gating signal onto OUTA (pin 18), as shown following, where it can be observed using an oscilloscope.

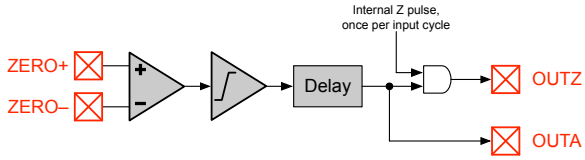


Figure 43: Observing the Z Gating Signal

Calibrate the Z signal path by incrementing MAIN_ZPOS (the *zpos* location within an input cycle) until the Z output is centered within the Z gating signal, as shown in Figure 42. MAIN_ZPOS can be accessed using the GUI or by writing to the MAIN_ZPOS register in the CFG block in internal memory via the serial ports (see Programmer's Reference). Finally, disable Index Gating Test Mode in the Test Mode pane of the Settings tab in the GUI or reset the Enable Z Test Mode bit (ztest) in the RB_TEST1 register and then store the updated MAIN_ZPOS register value to EEPROM using a Write All command via one of the serial ports or the Write EEPROM button in the GUI.

In general, the synchronization of the index gating signal to the Sin and Cos input signals is fixed. It is determined by the physical alignment of the incremental and Z tracks on the encoder disc or wheel and the physical alignment of the incremental and index sensors to each other. Since these physical alignments cannot typically be easily adjusted, it is important that they be carefully controlled from unit to unit to allow interchangeability of encoder wheels and sensors in series production.

To guarantee one and only one Z output pulse per revolution, the ZERO input signal from the index sensor 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. Specifically,

$$zwidth\,[°] < indexwidth\,[°] < 720° - zwidth\,[°]$$

where *indexwidth* [°] is the width of the index sensor signal in input cycle degrees.

For example, assume it is desired to have the Z output pulse centered within the input cycle. In this case, *zpos* should be set to a nominal value of

$$zpos\,[°] = 180° - \frac{zwidth\,[°]}{2}$$

and the ZERO input signal must be approximately physically aligned (in-phase) with the COS– input. As shown in Figure 44, the Z gating signal will meet the index

width requirement if its transitions fall anywhere in the gray areas of the waveform.

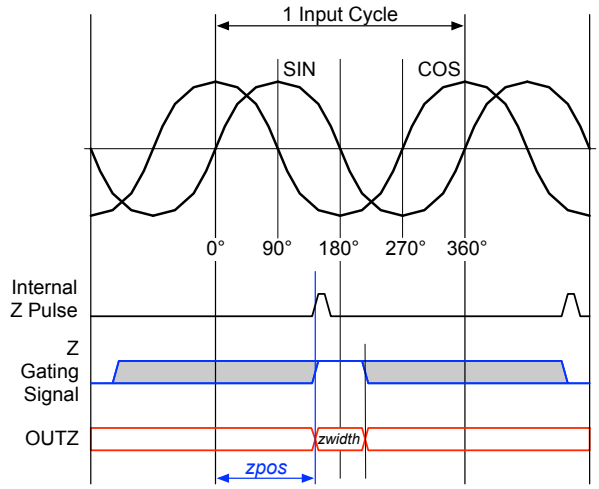


Figure 44: Digital Index Sensor Example

Any deviations from ideal synchronization of the index signal to the COS– signal due to production tolerances can then be compensated by adjusting the nominal *zpos* value, if necessary.

Analog Output Index Sensor

In applications where the incremental sensor produces multiple input cycles per revolution—such as optical and MR magnetic sensors—an index sensor must be connected to the ZERO inputs to qualify or "gate" the Z output to ensure one and only one Z output pulse per revolution. In this case, it is necessary to set the switching threshold of the Z signal path comparator, align the Z output pulse with the Z gating signal, and program the desired length and synchronization of the Z output pulse using the design tool (see page 46).

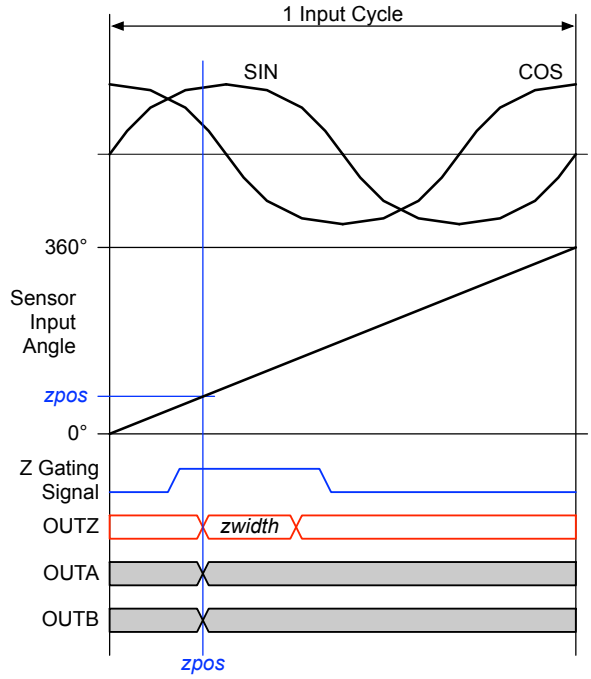


Figure 45: Z Output With Analog Index Sensor

First, connect an analog output index sensor to the iC-TW8 as explained on page 39. Then, in the Z Configuration tab of the design tool, set the initial Desired ZERO Threshold to 0, configure the remaining parameters as required (see page 46), and load the complete CFG bock to the iC-TW8 using the GUI or serial port commands.

To calibrate the ZERO threshold level, it is necessary to observe the internal Z gating signal. To observe the internal gating signal, enable Index Gating Test Mode in the Test Mode pane of the Settings tab in the GUI or set the Enable Z Test Mode bit (ztest) in the RB_TEST1 register in the RB Block of internal memory using serial port commands (see Programmer's Reference). This forces the internal Z gating signal onto OUTA (pin 18), as shown following, where it can be observed using an oscilloscope.

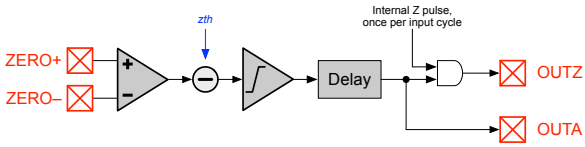


Figure 46: Observing the Z Gating Signal

Set the Z comparator switching threshold, *zth*, by adjusting the value of MAIN_Z.th to produce a single pulse at the Z gating signal output (OUTA) as shown.

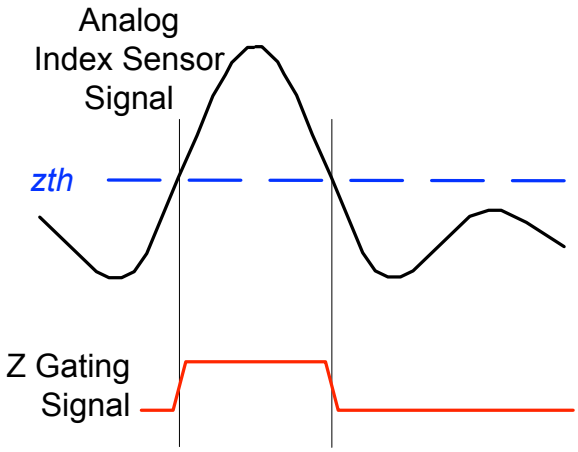


Figure 47: Setting the Z Comparator Threshold

Variable MAIN_Z.th can be accessed using the GUI or by writing to the MAIN_Z register in the CFG block in internal memory via the serial ports (see Programmers Reference). Note that since *zth* is subtracted from the amplified ZERO signal, negative MAIN_Z.th values produce positive *zth* values and vice versa.

Analog sensors often have side lobes on their output signals as shown in Figure 47. It is important that these side lobes not cross the comparator threshold under any operating condition otherwise multiple Z outputs may result.

The Z comparator switching threshold, *zth*, has a range of ±450 mV in ±30 mV steps (see Figure 46). However, since the ZERO input amplifier has a fixed gain of 8, the actual switching threshold range referenced to the ZERO input signal level is ±56.25 mV in steps of 3.75 mV.

To calibrate *zpos*, it is necessary to observe both the Z output and the internal Z gating signal simultaneously. Calibrate *zpos* by incrementing MAIN_ZPOS (the *zpos* location within an input cycle) until the Z output is centered within the Z gating signal, as shown in Figure 45. MAIN_ZPOS can be accessed using the GUI or by writing to the MAIN_ZPOS register in the CFG block in internal memory via the serial ports (see Programmer's Reference). Finally, disable Index Gating Test Mode in the Test Mode pane of the Settings tab in the GUI or reset the Enable Z Test Mode bit (ztest) in the RB_TEST1 register and then store the updated MAIN_ZPOS register value to EEPROM using a Write All command via one of the serial ports or the Write EEPROM button in the GUI.

In general, the synchronization the of analog index sensor signal to the Sin and Cos input signals is fixed. It is determined by the physical alignment of the incremental and Z tracks on the encoder disc or wheel and the physical alignment of the incremental and index sensors to each other. Since these physical alignments cannot typically be easily adjusted, it is important that they be carefully controlled from unit to unit to allow interchangeability of encoder wheels and sensors in series production.

To guarantee one and only one Z output pulse per revolution, the width of Z gating signal as set by *zth* 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. Specifically,

$$zwidth\,[°] < indexwidth\,[°] < 720° - zwidth\,[°]$$

where *indexwidth* [°] is the width of the Z gating signal in input cycle degrees.

For example, assume it is desired to have the Z output pulse centered within the input cycle. In this case, *zpos* should be set to a nominal value of

$$zpos\,[°] = 180° - \frac{zwidth\,[°]}{2}$$

and the analog index sensor signal must be roughly in-phase with the COS– input. As shown in Figure 48, the Z gating signal will meet the index width requirement if its transitions fall anywhere in the gray areas of the waveform.

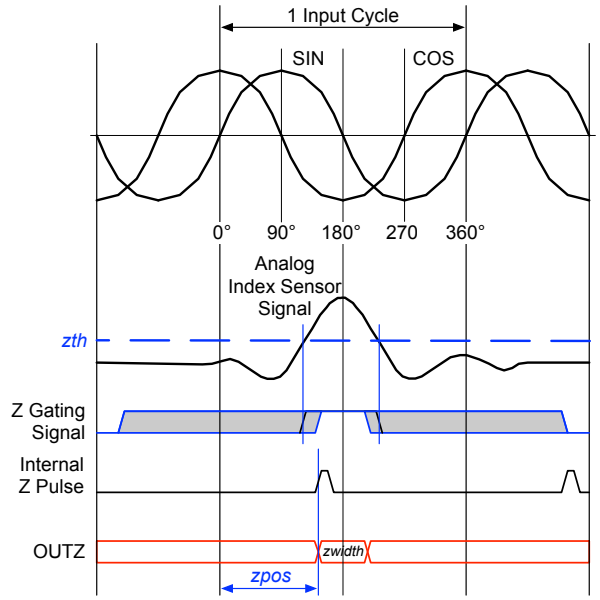


Figure 48: Analog Index Sensor Example

Any deviations from ideal synchronization of the index signal to the COS– signal due to production tolerances can then be compensated by adjusting the nominal *zpos* value, if necessary.

Using Serial-Only Output Mode

The iC-TW8 contains a 32-bit sensor position (angle) counter that can be accessed through either of the serial ports. Instantaneous sensor velocity can also be accessed via the serial ports. While the serial ports can be used to read position and velocity in all output modes, serial-only output mode removes the maximum AB output frequency limit, f_{ab}, allowing higher sensor speeds. Enter Serial for the Output Mode in the General Configuration tab of the design tool and the A, B, and Z outputs are disabled.

Serial-only output mode is not explicitly handled in the design tool, although the design tool may still be used for iC-TW8 configuration. See the iC-TW8 Programmer's Reference for information on the serial ports, the 32-bit position variable POS, and the sensor velocity variable VEL in the WM block in internal memory.

The 32-bit position is updated at the ADC sampling clock frequency, f_{adc}.

$$f_{\mathrm{adc}} = \frac{f_{\mathrm{core}}}{128}$$

To avoid jitter and aliasing when making continuous position reads, the ADC sampling frequency and the SPI sampling frequency must be synchronized externally. This can be accomplished either by synchronizing the iC-TW8 to the external CPU that handles SPI communication with the TW8 or by synchronizing the CPU to the iC-TW8.

To synchronize the iC-TW8 to an external CPU, the CPU must generate a signal derived from its own clock to drive the iC-TW8's clock input (pin XOUT). The CPU must then read the 32-bit position from the iC-TW8 at an integer divisor of f_{adc}.

To synchronize an external CPU to the iC-TW8, set MAIN_CFG.clockout = 1 to output the ADC sampling clock on the FRAME pin. This signal can then be used as an interrupt to the CPU to trigger an SPI position read. Alternatively, the iC-TW8's DSM clock on the CLOCK output pin can be used to directly clock the external CPU. In this case, the CPU must then read the 32-bit position from the iC-TW8 at an integer divisor of f_{adc}.

Using PWM Output Mode

As an alternative to the standard AB quadrature output, the iC-TW8 can provide a pulse-width modulated (PWM) output proportional to sensor angle. This is useful as a direct digital interface between the interpolator and a subsequent microcontroller or FPGA in imbedded control applications. Note that the PWM output is generated using a delta-sigma modulator so the signal does not look like a tradition PWM waveform. The interpolation factor *inter* must be less than or equal to 1024 in PWM output mode.

Enter PWM for the Output Mode in the General Configuration tab of the design tool. The A and B outputs (pins 18 and 17 respectively) become the differential PWM output; the Z output (pin 16) remains unchanged.

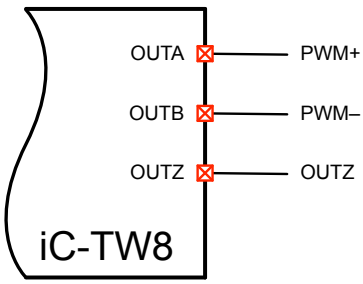


Figure 49: PWM Output Mode

Selecting PWM output mode in the design tool reveals the PWM configuration selections. Enter the desired frequency for the PWM output; the design tool shows the closest actual frequency available using the selected crystal. Finally, select whether the PWM output is clamped or not.

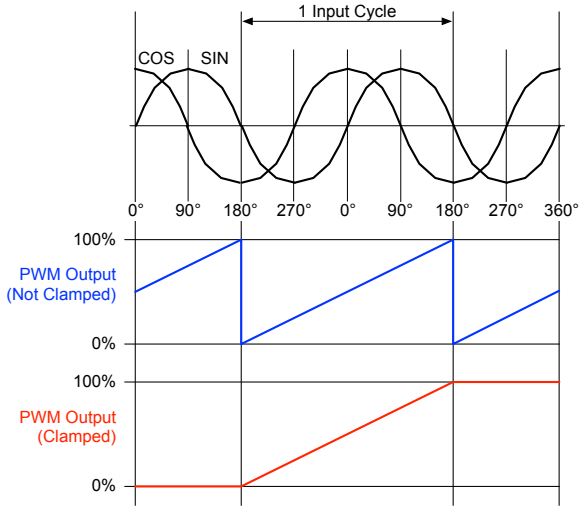


Figure 50: PWM Output Clamp

The typical configuration is not clamped, in which case the PWM output duty cycle represents the angle of the sensor Sin/Cos inputs over multiple input cycles. The clamped configuration is useful for single-turn absolute applications where there is only one Sin/Cos input cycle per revolution.

An external first or second order low-pass filter can be used to convert the PWM output to a voltage, which in turn can be sampled by an ADC. Since the PWM output is generated by a first order delta-sigma modulator (DSM), a second order filter is recommended. However, a first order analog RC filter may be sufficient in simple applications.

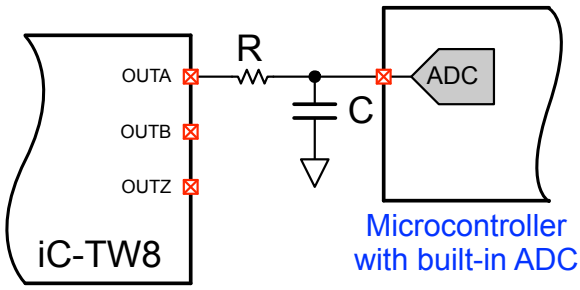


Figure 51: Analog Low-Pass PWM Filter

Alternatively, the PWM output can be directly sampled and digitally filtered by a microcontroller. FPGA, or PLD. In this case, enable the CLOCK and FRAME outputs in the General Configuration tab of the design tool. The CLOCK output (pin 6) can then be used to synchronously sample the PWM output.

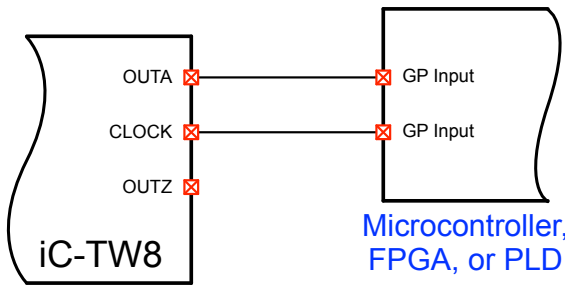


Figure 52: Digitally Sampling the PWM Output

This allows an all-digital decimation filter to be implemented. Since a first order DSM is used to generate the PWM output, a second order restructuring filter is required for best signal-to-noise performance. As with the analog filter, however, a first order filter may be sufficient for many applications.

Using the Post-AB Divider

The iC-TW8 includes an optional divider after the internal AB output generator that can be used to reduce (divide) the configured resolution by a factor of 1 – 32. This post-AB divider is useful when the desired output resolution is not an integer multiple of the input resolution.

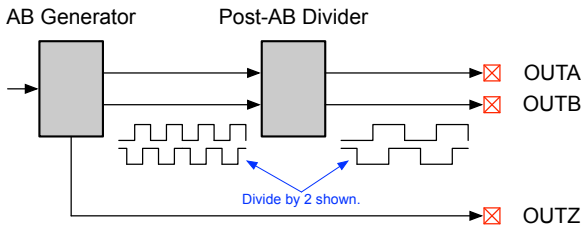


Figure 53: Post-AB Divider

When using the rotary encoder design tool to configure the iC-TW8, the post-AB divider is automatically enabled if necessary. The AB Configuration tab in the design tool calculates the minimum divider necessary to achieve the desired output resolution using the configured input resolution. This minimum value may be overridden if necessary by entering a divider override factor.

As shown in Figure 53, the Z output bypasses the post-AB divider. Therefore, any configured synchronization of the Z output to the A and B outputs is lost when using the post-AB divider.

For example, with an input resolution of 24 Sin/Cos cycles per revolution, it is impossible to achieve an output resolution of 16,384 AB cycles per revolution (CPR) without the post-AB divider since $4 \cdot 16\,384/24$ is not an integer (see page 43). However, entering these values into the design tool gives a minimum divider value of 3 and an effective interpolation, *intereff*, of 682.66667. Thus, an output resolution of 16,384 CPR is possible with an input resolution of 24 by using the post-AB divider.

Note that the maximum speed and maximum adaption speed are inversely proportional to the post-AB divider value. Higher post-AB divider values produce lower maximum speed and maximum adaption speed. These speed values are shown in the AB Configuration tab of the design tool.

The post-AB divider is also useful to allow a complete range of output resolutions to be achieved using a single input resolution. For example, with an input resolution of 128 Sin/Cos cycles per revolution, only binary output resolutions can be achieved without using the post-AB divider. As shown in cell B49 of the AB Configuration tab of the design tool, resolutions between 128 and 2,097,152 CPR in increments of 32 CPR can be achieved with this configuration. The actual resolutions available in this configuration (128, 160, 192, 224, 256, etc.) are shown in the All Resolutions tab in the design tool.

Entering a divider override factor of 16 in this configuration provides an actual divider of 16 and reduces the resolution increment to 2 CPR. Now, all even resolutions between 8 and 131,072 CPR may be achieved. Increasing the divider override factor to 32 allows all integer resolutions (4, 5, 6, 7, 8, etc.) between 4 and 65,536 to be achieved using an input resolution of 128 Sin/Cos cycles per revolution.

Tuning the Internal Oscillator

The iC-TW8's internal oscillator has a nominal frequency of 20 MHz with 5 V supplies and 16 MHz with 3.3V supplies. However, manufacturing tolerances and changes in temperature can cause large variations in the actual internal oscillator frequency of any given device.

The clock tuning value in the MAIN_CLOCK.freq variable allows tuning the frequency of the internal oscillator. In addition, the clock divider value in the MAIN_CLOCK.div variable allows dividing the oscillator frequency by 1, 2, or 4. These two parameters allow setting the internal oscillator of any specific device to the desired frequency. The MAIN_CLOCK register can be accessed using the serial ports (see Programmer's Reference) or via the Settings tab in the GUI.

To observe the internal oscillator output, enable Internal Oscillator Test Mode in the Test Mode pane of the Settings tab in the GUI or set the Enable Clock Test Mode bit (adctest) in the RB_TEST1 register in the RB Block of internal memory using serial port commands (see Programmer's Reference). This forces $f_{core}/2$ onto the FRAME output (pin 8), as shown below, where it can be observed using an oscilloscope or a frequency counter.

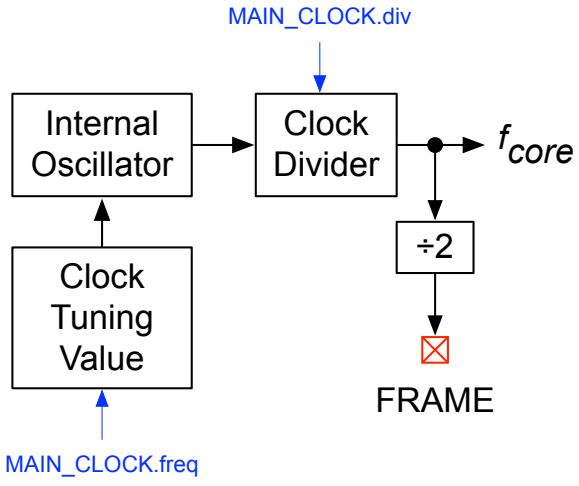


Figure 54: Internal Oscillator Tuning

First, set the clock divider to the desired value, usually 1 (MAIN_CLOCK.div = 3), and the clock tuning value to its lowest value (MAIN_CLOCK.freq = 0). Increment the clock tuning value until the desired oscillator frequency is obtained, remembering that the output frequency on the FRAME pin is half the actual oscillator frequency. Finally, disable Internal Oscillator Test Mode in the Test Mode pane of the Settings tab in the GUI or reset the Enable Clock Test Mode bit (adctest) in the RB_TEST1 register and then store the updated MAIN_CLOCK register value to EEPROM using a Write All command via one of the serial ports or the Write EEPROM button in the GUI.

Typical internal oscillator frequency of the iC-TW8 with 5V supplies and a clock divider of 1 at room temperature (20°C) is shown below. Other clock divider values produce appropriately scaled-down frequencies.

iC-TW8 Typical Internal Oscillator Frequency		
Clock Tuning Value	**FRAME Output Frequency [MHz]**	**f_{core} Frequency [MHz]**
0	10.6	21.2
1	11.0	22.0
2	11.5	23.0
3	11.9	23.8
4	12.5	25.0
5	13.0	26.0
6	13.6	27.2
7	14.3	28.6
8	15.1	30.2
9	15.8	31.6
10	16.8	33.6
11	17.7	35.4
12	18.9	37.8
13	20.2	40.4
14	21.6	43.2
15	23.3	46.6

Table 14: Typical Internal Oscillator Frequency

Note that the nominal oscillator frequency is achieved with a clock tuning value of 0 and that the range of the clock tuning value allows setting the internal oscillator frequency above the maximum allowed for proper operation of the device (shown in red). Setting the internal oscillator frequency higher than the specified maximum may result in undefined operation.

In general, set the internal oscillator frequency as close to—but not greater than—the maximum frequency for the supply voltage used (32 MHz at 5V, 24MHz at 3.3V). The temperature coefficient of the internal oscillator is negative, so oscillator frequency decreases with rising temperature and increases with decreasing temperature. In applications where low temperature operation is expected, the room temperature oscillator frequency should be reduced accordingly. See Electrical Characteristics for the actual value of the oscillator's temperature coefficient.

Using the LUT

The iC-TW8 contains a 64-byte sensor distortion look-up table (LUT) that can be used to apply an arbitrary correction function to the calculated sensor position (angle). To use the LUT, the correction values must first be calculated and loaded into the table using the design tool, the GUI software, or commands sent via the serial ports. Then the LUT must be enabled by setting ADPT_CFG.lut = 1 in the CFG block in internal memory. When the LUT is enabled, the maximum SPI port clock frequency is reduced to $f_{core}/4$. If ADPT_CFG.lut = 0, the LUT is bypassed and no correction is applied.

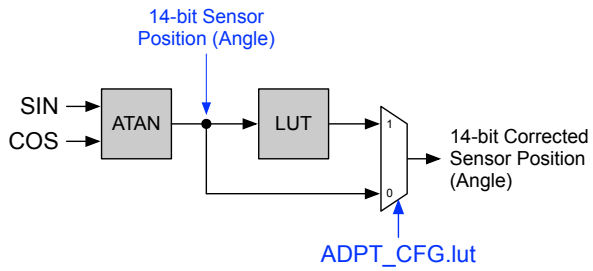


Figure 55: Sensor Distortion LUT

The LUT divides the sensor Sin/Cos input cycle into 64 segments of 5.625° each and the table values define a 64-point piecewise linear correction curve. Correction values for sensor angles within a LUT table segment are linearly interpolated between the two table values on either end of the segment. In operation, the calculated correction value is added to the sensor angle.

LUT table values have a range of ±127. The actual correction applied to the 14-bit sensor angle is 4 times the table value, giving the LUT a correction range of

$$\frac{4 \cdot \pm 127}{16\,384} \cdot 360° = \pm 11.16°$$

The resolution of the LUT values is

$$\frac{4}{16\,384} \cdot 360° = 0.088° = 5.3 \text{ arc minutes}$$

The simplest LUT is a null table, where all values are zero. This is equivalent to bypassing the LUT and applies no correction to the sensor signal, but serves as a starting point for understanding LUT operation.

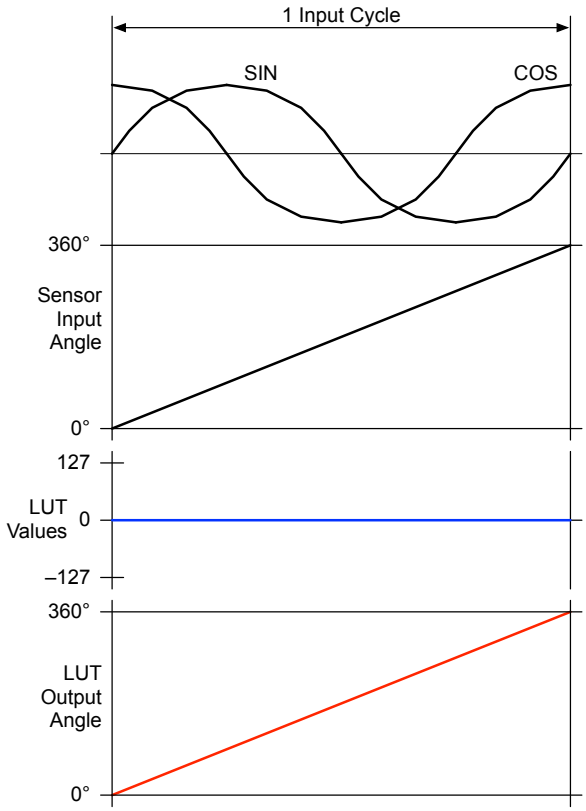


Figure 56: Null LUT

A constant LUT is one in which all the values are the same. This adds a constant phase shift to the sensor angle, shifting the 0° position left for positive LUT values and right for negative values.

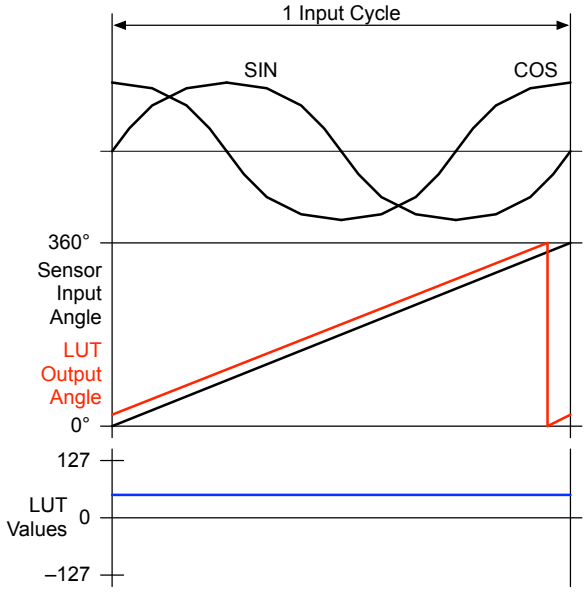


Figure 57: Constant LUT

A constant LUT is useful in single turn absolute applications such as synchronous AC motor commutation where the feedback signal must be aligned to the motor poles. For example, to shift the interpolator 0° position 3.8° to the right, the constant LUT value is calculated as

$$-\frac{3.8}{360} \cdot \frac{16\,384}{4} = -43$$

More complex LUTs require careful FFT measurement of sensor distortion. When sensor distortion harmonics are known, the design tool can be used to calculate a LUT that corrects for up to two sensor harmonics which are identical on both Sin and Cos input channels.

In the LUT Configuration tab of the design tool, enter up to two sensor distortion harmonic to be corrected. The harmonic number, amplitude in percent, and phase in degrees are required. To correct for only one distortion harmonic enter 0 amplitude for the unused harmonic. The design tool shows the resultant angular error due to the distortion and the residual error left after correction. The residual angular error is due to the finite (8 bit) resolution of the LUT.

For example, if FFT analysis of a particular sensor showed 2% second harmonic distortion at 0° phase shift and 1% third order harmonic distortion at 180° phase shift, the following angular error results.

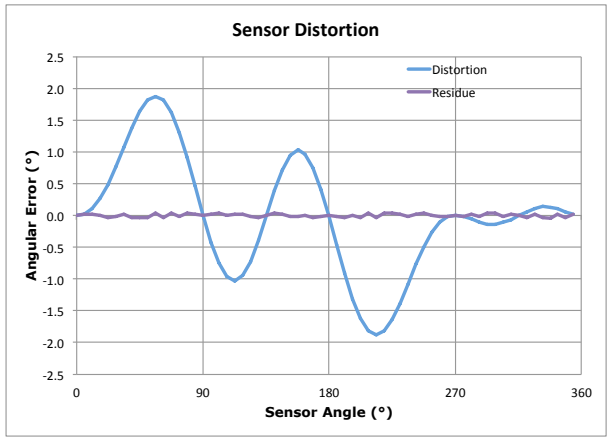


Figure 58: Sensor Distortion

The design tool shows that this sensor distortion produces peak angular error of 1.88 input cycle degrees. After correction by the LUT, 0.04° (15.76 arc minutes) of angular error remain.

The GUI LUT tab in the design tool shown the calculated LUT as a graph and LUT values. Save the GUI LUT tab as a .csv file for direct import into the GUI software. The LUT entries are also copied into the GUI Hex tab of the design tool.

ADDITIONAL INFORMATION

Documentation

- Programmer's Reference: available on request, please contact iC-Haus applications engineering.
- Evaluation Board Description iC-TW8 EVAL TW8_1D: refer to iC-Haus website for download.
- GUI Software: refer to iC-Haus website for download.

Design Tools

- Rotary Encoder Design Tool (Excel spreadsheet):
  available on request, please contact iC-Haus applications engineering.

Application Notes

- (under preparation)

DESIGN REVIEW: Function Notes

iC-TW8 C1		
No.	Function, Parameter/Code	Description and Application Notes
1	Look-up Table: ADPT_CFG.lut	The distortion look-up table is not featured by chip release C1; mandatory programming: ADPT_CFG.lut = 0

Table 15: Notes on chip functions regarding iC-TW8 chip release C1.

iC-TW8 D1, D2, D3		
No.	Function, Parameter/Code	Description and Application Notes
1	Adaption Quality Monitoring: MON_CFG.adapt	If enabling adaption quality monitoring, note that this reduces the permissible adaption limits and quality thresholds for MON_OFF, MON_GAIN, and MON_PHASE (allowed maximum value is 7 instead of 12). Refer to the Programmer's Reference for further details.
2	ADC Underflow: STAT_SP.adcuf	This status error is initially raised on startup because the signal amplitude (first ADC_AMP value) has not been calculated yet.
3	ADC Fault Enable: FLT_EN.adc	Do not enable latching (freezes ADC underflow status raised on startup). If latching ADC overflow and underflow conditions is essential to the system, use an external MCU to set the FLT_EN.adc after startup has been completed.
4	32-bit Position Counter Reset: MAIN_Z.reset	To reset the 32-bit position counter by an index signal, use MAIN_Z.reset = 1 to reset whenever the Z output is active. Do not use MAIN_Z.reset = 2 to reset by the ZERO input signal (as described in previous datasheet releases), as this may cause invalid data for SPI output and spurious AB output bursts.
5	Reset during EEPROM read access: STAT_EE.id: wrong EEPROM ID	Refer to the description # 1 for chip release D4 below.

Table 16: Notes on chip functions regarding iC-TW8 chip releases D1, D2, D3

iC-TW8 D4, D4H		
No.	Function, Parameter/Code	Description and Application Notes
1	Reset during EEPROM read access: STAT_EE.id: wrong EEPROM ID	If the iC-TW8 is reset during I^2C communication, SCL and SDA are immediately pulled high (default state). In this case, the external EEPROM does not get a proper stop condition, and the first read (address 0) fails when the reset is released causing a wrong EEPROM ID. Subsequent addresses are read successfully, as proper stop conditions return the EEPROM to its idle state. To avoid this situation, do one of the following: 1. Do not reset the iC-TW8 while EEPROM access is active. Delay the reset until startup is complete or keep the reset pin low during power-up. 2. Connect SDA to the host processor and drive it low before resetting the iC-TW8. 3. Eliminate the EEPROM and use the host processor to configure the iC-TW8 after reset.

Table 17: Notes on chip functions regarding iC-TW8 chip release D4, and D4H.

REVISION HISTORY

Rel.	Rel. Date*	Chapter	Modification	Page
C3	2017-10-27		Refer to datasheet C3 for previous changes to configuration sections.	

Rel.	Rel. Date*	Chapter	Modification	Page
D1	2019-02-05	PACKAGING INFORMATION	Note addded to ZERO+/- inputs	7
		ELECTRICAL CHARACTERISTICS	Item 402: item added; item 404: min/max values added; Items 606: splitted into 606 and 607, values adapted Block B introduced: item 807 moved to B01 and values adapted, item B02 added, item 711 moved to B03 and name changed	9, 10, 11

Rel.	Rel. Date*	Chapter	Modification	Page
D2	2019-08-19	PIN FUNCTIONS	Note added for pin IR	7
		PIN and SERIAL CONFIGURATION MODES	Appendices PC and SC manuals taken up into main document Correction of Figure 46 (page 54) for zth having the incorrect sign (polarity). Correction of zth range (pages 34 and 54)	15-27, 28-60

Rel.	Rel. Date*	Chapter	Modification	Page
D3	2021-10-05	SERIAL CONFIGURATION MODE	Note added on MAIN_Z.th programming Note added on ZERO input wiring	34, 38
		DESIGN REVIEW: Function Notes	Entry added; chip revision D4H added	61cf

* Release Date format: *YYYY-MM-DD*

ORDERING INFORMATION

Type	Package	Options	Order Designation
iC-TW8	48-pin QFN, 7 mm x 7 mm, thickness 0.9 mm, RoHS compliant		iC-TW8 QFN48-7x7
Evaluation Board	PCB, 120 mm x 100 mm		iC-TW8 EVAL TW8_1D
PC-USB Adapter		Only required if the evaluation board needs to be operated by the GUI.	iC-MB3 iCSY MB3U-I2C
iC-TW8 GUI		Evalutation software for Windows PC (entry of IC parameters, file storage, and transfer to DUT)	For download link refer to www.ichaus.com/tw8
Encoder Design Tool		Excel-based rotary encoder design tool for parameter definition.	Check www.ichaus.com/tw8 for download link or contact iC-Haus applications engineering.

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: http://www.ichaus.com
E-Mail: sales@ichaus.com

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