TDC-GPX2

4-Channel Time-to-Digital Converter

General Description

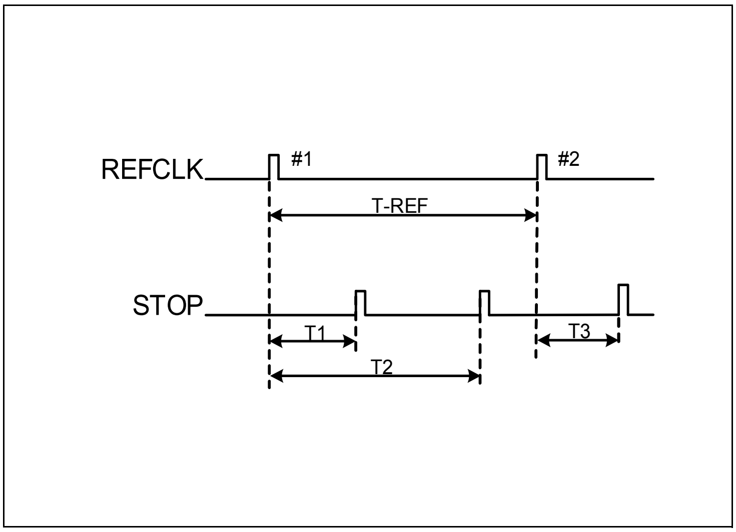
The GPX2 is a high performance time-to-digital converter (TDC) frontend device.

Highest measurement performance and highest data throughput is achieved with LVDS stop inputs and LVDS serial outputs for each channel. Current saving operation is also possible with CMOS inputs and SPI readout.

High configuration flexibility and unlimited measurement range cover many applications. They range from portable handheld laser range equipment to ambitious time-of-flight measurements of highest performance, as e.g. done in medical imaging applications.

GPX2 operates without any locked loop technologies. GPX2 calculates all stop measurements inside, proportional to the applied reference clock. Combinations of best single shot accuracy of 10ps with lowest pulse-to-pulse spacing <5ns and maximum data throughput rate of 70MSPS per stop input are possible.

Figure 1:
Time Interval Measurements



Ordering Information and Content Guide appear at end of datasheet.

Key Benefits & Features

The benefits and features of this device are listed below:

Figure 2:
Added Value of Using TDC-GPX2

Benefits	Features

• Simple data post-processing thanks to calibrated results	• 4 stop channels with serial
	  • 20ns pulse-to-pulse spacing
	  • Maximum 35MSPS
	• 2 combined channels with
	  • 5ns pulse-to-pulse spacing
	  • Maximum 70MSPS
	• Single shot accuracy
	  • 20ps rms single shot resolution per channel
	  • 10ps rms with high resolution option
	• Unlimited measuring range 0s to 16s

• Event assignment thanks to reference clock index simplifies coincidence measurements • Easy pulse width measurements • High efficiency thanks high sample rate	• Differential reference clock input 2MHz to 12.5MHz, optional with quartz • Inputs optional with LVDS or CMOS level • Readout with LVDS or SPI • 16-stage FIFO per channel • Automatic calibration to reference clock (no PLL or DLL) • SPI compatible 4-wire interface for configuration
• Compact design thanks to small package and low number of external components • Reduced cooling thanks to low power consumption	• Supply voltage 3.3V • Power dissipation 60mW to 450mW • Standby current 60µA • QFN64 (9mm x 9mm) or QFP64 (12mm x 12mm)

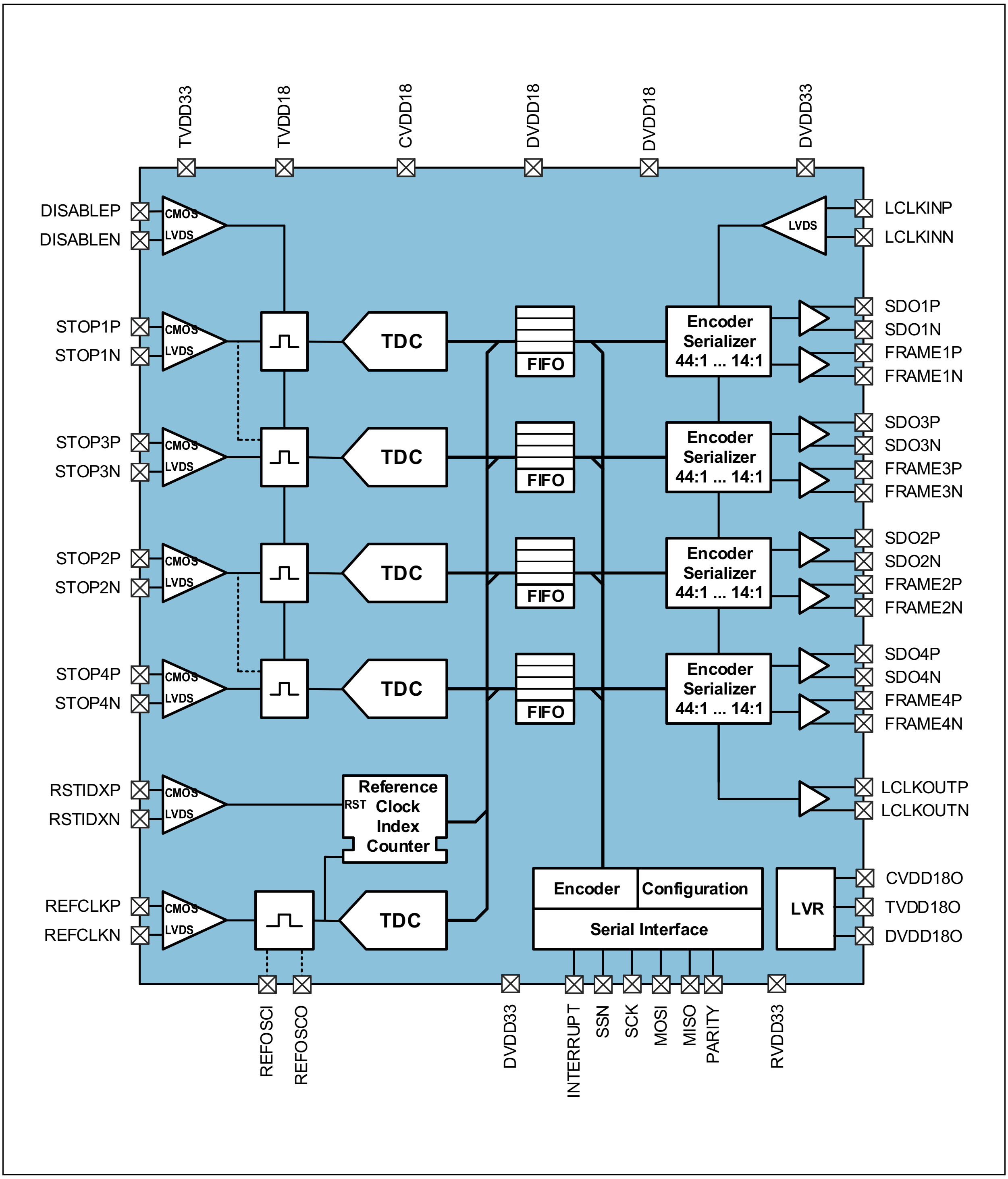
Applications

- Automated Test Equipment
- Laser Range Measurement
- Medical Imaging
- Time-of-Flight Measurement
- Particle Physics
- Lidar, Radar, Sonar

Block Diagram

The functional blocks of this device are shown below:

Figure 3:
Functional Blocks of TDC-GPX2

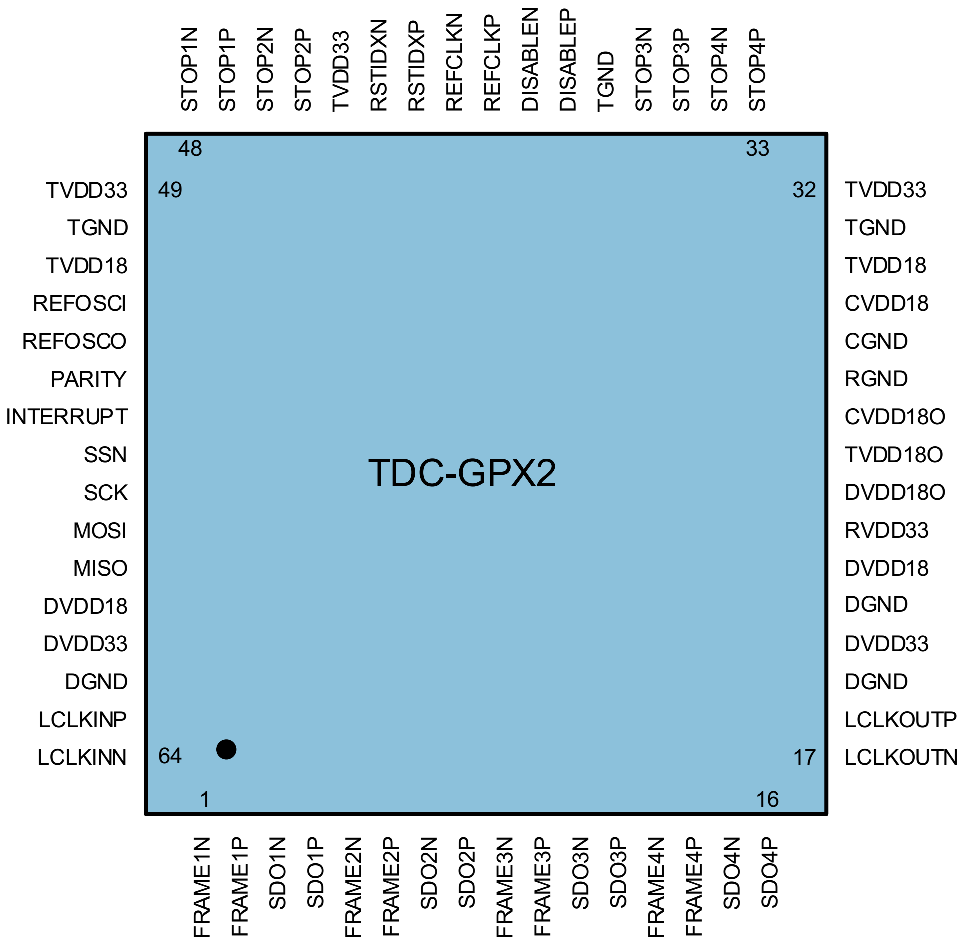


Pin Assignments

The TDC-GPX2 is shipped in QFN64 or QFP64 plastic packages with the following pin assignment.

Pin Diagram

Figure 4:
Pin Diagram of TDC-GPX2



Pin Description

Figure 5:
Pin Description of TDC-GPX2

Pin No.	Pin Name	Description	Type	Not Used
1	FRAME1N	Negative frame signal of stop channel 1	LVDS Output	Open
2	FRAME1P	Positive frame signal of stop channel 1	LVDS Output	Open
3	SDO1N	Negative serial data output of stop channel 1	LVDS Output	Open
4	SDO1P	Positive serial data output of stop channel 1	LVDS Output	Open
5	FRAME2N	Negative frame signal of stop channel 2	LVDS Output	Open
6	FRAME2P	Positive frame signal of stop channel 2	LVDS Output	Open
7	SDO2N	Negative serial data output of stop channel 2	LVDS Output	Open
8	SDO2P	Positive serial data output of stop channel 2	LVDS Output	Open
9	FRAME3N	Negative frame signal of stop channel 3	LVDS Output	Open
10	FRAME3P	Positive frame signal of stop channel 3	LVDS Output	Open
11	SDO3N	Negative serial data output of stop channel 3	LVDS Output	Open
12	SDO3P	Positive serial data output of stop channel 3	LVDS Output	Open
13	FRAME4N	Negative frame signal of stop channel 4	LVDS Output	Open
14	FRAME4P	Positive frame signal of stop channel 4	LVDS Output	Open
15	SDO4N	Negative serial data output of stop channel 4	LVDS Output	Open
16	SDO4P	Positive serial data output of stop channel 4	LVDS Output	Open
17	LCLKOUTN	Negative serial clock output	LVDS Output	Open
18	LCLKOUTP	Positive serial clock output	LVDS Output	Open
19, 21, 62	DGND	Ground for digital and IO units	Power Supply	
20, 61	DVDD33	3.3V supply for digital and IO units	Power Supply	
22, 60	DVDD18	1.8V supply for digital and IO units	Power Supply	
23	RVDD33	3.3V supply for linear voltage regulator	Power Supply	
24	DVDD18O	1.8V supply voltage for digital and IO units	Regulator Out	Open
25	TVDD18O	1.8V supply voltage for time frontend	Regulator Out	Open
26	CVDD18O	1.8V supply voltage for time digital converter	Regulator Out	Open
27	RGND	Ground for linear voltage regulator	Power Supply	
28	CGND	Ground for TDC	Power Supply	

Pin No.	Pin Name	Description	Type	Not Used
29	CVDD18	1.8V positive supply for TDC	Power Supply	
30, 51	TVDD18	1.8V positive supply for time front-end	Power Supply	
31, 37, 50	TGND	Ground for 1.8V time front-end supply	Power Supply	
32, 44, 49	TVDD33	3.3V positive supply for time front-end	Power Supply	
33	STOP4P	Positive stop input for channel 4	CMOS/LVDS Input	TVDD33
34	STOP4N	Negative stop input for channel 4	LVDS Input	TVDD33
35	STOP3P	Positive stop input for channel 3	CMOS/LVDS Input	TVDD33
36	STOP3N	Negative stop input for channel 3	LVDS Input	TVDD33
38	DISABLEP	Positive disabling pin for stop channels	CMOS/LVDS Input	TVDD33
39	DISABLEN	Negative disabling pin for stop channels	LVDS Input	TVDD33
40	REFCLKP	Positive clock signal of reference clock	CMOS/LVDS Input	TVDD33
41	REFCLKN	Negative clock signal of reference clock	LVDS Input	TVDD33
42	RSTIDXP	Positive reference index reset signal	CMOS/LVDS Input	TVDD33
43	RSTIDXN	Negative reference index reset signal	LVDS Input	TVDD33
45	STOP2P	Positive stop input for channel 2	CMOS/LVDS Input	TVDD33
46	STOP2N	Negative stop input for channel 2	LVDS Input	TVDD33
47	STOP1P	Positive stop input for channel 1	CMOS/LVDS Input	TVDD33
48	STOP1N	Negative stop input for channel 1	LVDS Input	TVDD33
52	REFOSCI	Input for quartz as reference clock	XOSC Driver In	Open
53	REFOSCO	Output for quartz as reference clock	XOSC Driver Out	Open
54	PARITY	Parity of all configuration registers	LVTTL Output	Open
55	INTERRUPT	SPI interrupt	LVTTL Output	Open
56	SSN	SPI slave select not + interface reset	LVTTL Input	
57	SCK	SPI serial clock	LVTTL Input	
58	MOSI	SPI serial data master out, slave In	LVTTL Input	
59	MISO	SPI serial data master in, slave Out	LVTTL Tristate	
63	LCLKINP	Positive serial clock in	LVDS Input	DVDD33
64	LCLKINN	Negative serial clock in	LVDS Input	DVDD33

Note(s):

1. A small dot on the package indicates the pin 1. There is no need to connect the exposed pad to GND (internally not connected). Connecting it may be helpful for heat dissipation. The package is RoHS compliant and does not contain any Pb.

Absolute Maximum Ratings

Stresses beyond the Absolute Maximum Ratings may cause permanent damages to the device. Exposure to any Absolute Maximum Rating condition for extended periods may also affect device reliability and lifetime.

Figure 6:
Absolute Maximum Ratings

Symbol	Parameter	Min	Max	Units	Comments
Electrical Parameters					
VDD33	3.3V Supply Voltage to Ground	-0.5	4.0	V	Pins DVDD33, TVDD33, RVDD33
VDD18	1.8V Supply Voltage to Ground	-0.5	2.2	V	Pins DVDD18, TVDD18, CVDD18
	Voltage between ground pins	-0.3	+0.3	V	Pins DGND, TGND, RGND, CGND
V_{iLVDS}	Voltage at differential input pins	-0.3	VDD33 + 0.3	V	Pins STOP1, STOP2, STOP3, STOP4, REFCLK, REFRES, DISABLE, LCLKIN
V_{osc}	Voltage at input of oscillator cell	-0.3	VDD18 +0.3	V	Pin REFOSCIN
Electrostatic Discharge					
ESD_{HBM}	Electrostatic Discharge HBM	± 1000		V	JS-001-2014
Temperature Ranges and Storage Conditions					
T_J	Operating Junction Temperature	-40	125	°C	
T_{STRG}	Storage Temperature Range	-65	150	°C	
T_{BODY}	Package Body Temperature		260	°C	The reflow peak soldering temperature (body temperature) is specified according to IPC/JEDEC J-STD-020 “Moisture/Reflow Sensitivity Classification for Non-hermetic Solid State Surface Mount Devices.” The lead finish for Pb-free leaded packages is “Matte Tin” (100% Sn)
RH_{NC}	Relative Humidity (non-condensing)	5	85	%	
MSL	Moisture Sensitivity Level	3			Maximum floor life time of 168 hours

Recommended Operation Conditions

Recommended operating ratings indicate conditions for which the device is functional, but do not guarantee specific performance limits. Test conditions for guaranteed specification are expressly denoted.

Figure 7:
Recommended Operating Conditions

Symbol	Pin	Description	Min	Typ	Max	Unit
Power-Supply						
VDD33	DVDD33, TVDD33, RVDD33	Supply Voltage	2.4	3.3	3.6	V
VDD18	DVDD18, TVDD18, CVDD18	Core Supply Voltage powered by integrated regulator, pins DVDD18O, TVDD18O, CVDD18O	1.7	1.8	1.9	V
Temperature						
T_A		Operating free air temperature [(1)]	-40		125	°C
Reference & Stop Inputs						
$V_{ID,LVDS}$	STOP1, STOP2, STOP3, STOP4, REFCLK, RSTIDX, DISABLE,	LVDS Differential Input Voltage	200			mV
$V_{IC,LVDS}$		LVDS Common Mode Input Voltage	$V_{ID}/2$	1.25	2.2 – $V_{ID}/2$	V
$V_{IL,CMOS}$		CMOS Input Low Voltage			0.4	V
$V_{IH,CMOS}$		CMOS Input High Voltage	VDD33 - 0.4			V
SPI-Interface						
V_{IL}	SCK, MOSI, SSN	Digital Input LOW Voltage			0.8	V
V_{IH}		Digital Input HIGH Voltage	0.7 * VDD33			V
C_{LOAD}	INTERRUPT, MISO, PARITY	Load Capacitance to Ground			20	pF

Symbol	Pin	Description	Min	Typ	Max	Unit
LVDS-Interface						
$V_{ID,LVDS}$	LCLKIN	LVDS Differential Input Voltage	200			mV
$V_{IC,LVDS}$		LVDS Common Mode Input Voltage		1.25		V
R_{TERM}	SDO1, SDO2, SDO3, SDO4, FRAME1, FRAME2, FRAME3, FRAME4, LCLKOUT	Differential Termination Resistor for LVDS Outputs		100		Ω
C_{LOAD}		Load Capacitance to Ground			5	pF

Note(s):

1. Recommended Operating Ratings indicate conditions for which the device is functional, but do not guarantee specific performance limits. Test conditions for guaranteed specification are explicitly denoted.

The following test levels apply to all following characteristics:

Figure 8:
Test Levels

Test Level	Description
I	100% production tested.
II	100% production tested at 25 °C and guaranteed by design and characterization testing
III	Parameter is guaranteed by design and characterization testing
IV	Sample tested
V	Parameter is a typical value only.

Converter Characteristics

General Conditions: VDD33 = 3.3V; VDD18 = 1.8V;
T_A = 0°C to 80°C.

Figure 9:
Converter Characteristics

Symbol	Description	Condition	TL	Min	Typ	Max	Unit
Accuracy of Time Measurement							
RMS	Single-shot RMS resolution	High_Resolution = 0 (off) High_Resolution = 1 (2x) High_Resolution = 2 (4x)	IV		20 15 10	30 20 15	ps
INL	Integral non-linearity		IV			20	ps
DNL	Differential non-linearity		V		5		ps
	No missing code	At time quantization level	III	Assured			
	Channel to channel isolation	At same times measured	IV		20	100	ps
	Offset error	High_Resolution = 0 (off) High_Resolution = 1 (2x) High_Resolution = 2 (4x)	V		100 150 200		ps
	Offset error temperature drift	High_Resolution = 0 (off) High_Resolution = 1 (2x) High_Resolution = 2 (4x)	IV		0.5 1 1.5	3	ps/K
Switching Performance							
t_{CONV}	Converter latency	High_Resolution = 0 (off) High_Resolution = 1 (2x) High_Resolution = 2 (4x)	III			20 50 100	ns
	Peak conversion rate	High_Resolution = 0 (off) High_Resolution = 1 (2x) High_Resolution = 2 (4x)	III			50 20 10	MSPS
	Maximum read-out rate LVDS: 44 Bit/14 Bit SPI: Opcode + 48 Bit/16 Bit	SDR / 250MHz DDR / 250MHz SPI / 50MHz	III	5.6 11.3 0.9		17.8 35.7 2.1	MSPS

Power Supply Characteristic

General Conditions: VDD33 = 3.3V; VDD18 = 1.8V;
T_A = 0°C to 80°C

Figure 10:
Power Supply Characteristics

Symbol	Description	Condition	TL	Min	Typ	Max	Unit
		Supply Voltage					
t_{VDD18O}	Delay from power-up of RVDD33 to TVDD18O, CVDD18O, DVDD18O stable	C_{load} = 100µF	V			100	ms
$P_{TOT,MIN}$	Minimum total power dissipation	CMOS inputs and SPI read f_{REFCLK} = 5MHz conversion rate 1MSPS	V		60		mW
$P_{TOT,MAX}$	Maximum total power	LVDS inputs and outputs f_{REFCLK} = 10MHz $f_{STOP1..4}$ = 50MHz f_{LCLK} = 300MHz	V		450		mW
		Detailed Current Consumption					
$I_{DVDD18,REFCLK}$	Core current into REFCLK	f_{REFCLK} = 5MHZ	V		2		mA
$I_{DVDD18,STOP}$	Current per stop channel		V		0.5		mA
I_{CVDD18}	Current with activated TDC core		V		14		mA
$I_{TVDD18,REFOSC}$	Quartz oscillator current if used	f_{REFOSC} = 4MHZ	III		2		mA
$I_{DVDD33,LVDS-IN}$ $I_{TVDD33,LVDS-IN}$	Current per LVDS input buffer		III		2	6	mA
$I_{DVDD33,LVDS-OUT}$	Current per LVDS output buffer	RTERM = 100Ω	III		5	10	mA
I_{DDQ}	Quiescent current mainly by I_{RVDD33}	LVDS inputs tied to VDD33	II		60	100	µA
I_{LKG}	Input leakage current	LVDS, CMOS, Digital, REFOSCI	II	-5		1	µA

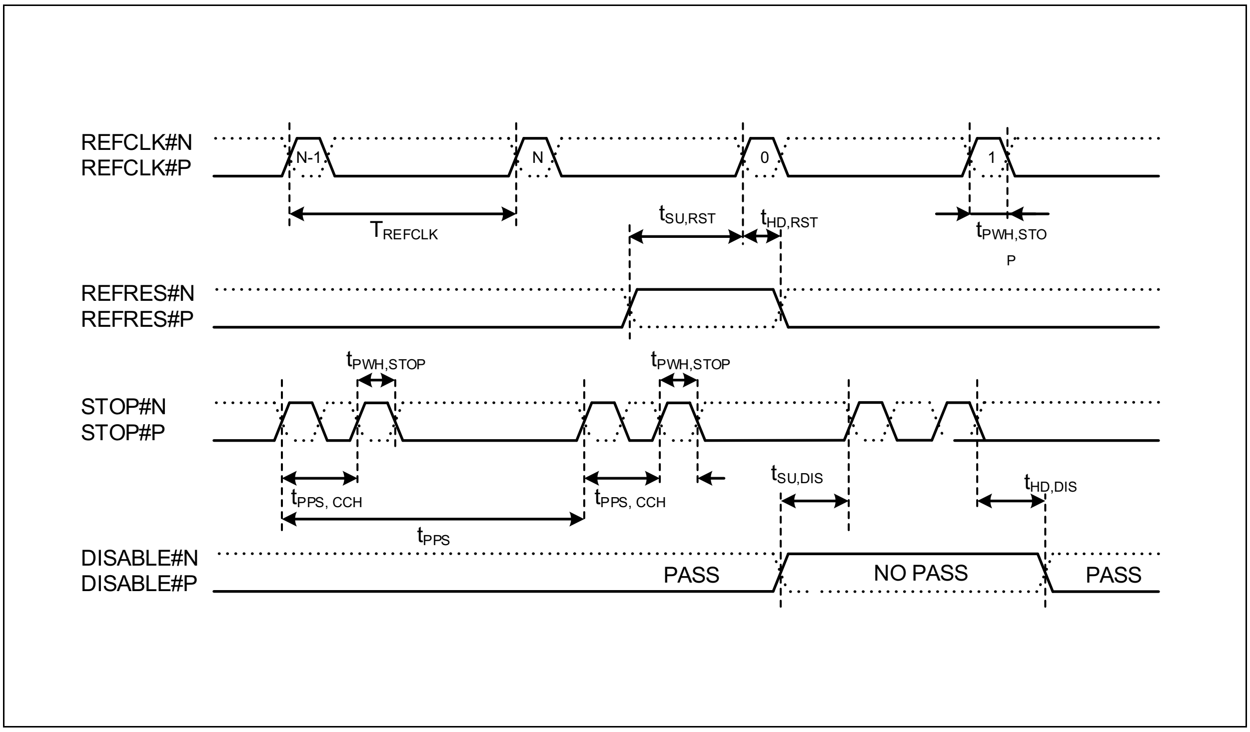
Reference Clock and Stop Input Requirements

General Conditions: VDD33 = 3.3V; VDD18 = 1.8V;
T_A = 0°C to 80°C; V_{ID} = 200mV; V_{IC} = 1.25V; V_{IL} = 0V; V_{IH} = 3.3V

Figure 11:
Clock and Input Characteristics

Symbol	Description	Condition	TL	Min	Typ	Max	Unit
f_{REFCLK}	Reference clock frequency	High_Resolution = 0 (off) High_Resolution = 1 (2x) High_Resolution = 2 (4x)	III	2 2 2	5 5 5	12.5 12.5 10.0	MHz
f_{REFOSC}	Reference oscillator frequency at pin 52,53	High_Resolution = 0 (off) High_Resolution = 1 (2x) High_Resolution = 2 (4x)	III	2 2 2	5 5 5	12.5 12.5 10.0	MHz
T_{REFCLK}	Reference clock period		III	83	200	500	ns
	Reference clock jitter		V			100	ps
	Reference clock stability	No requirement					
$t_{PWH,STOP}$	Minimum pulse width	LVDS CMOS	III	2 10			ns
t_{PPS}	Minimum pulse-to-pulse spacing	High_Resolution = 0 (off) High_Resolution = 1 (2x) High_Resolution = 2 (4x)	III	20 50 100			ns
$t_{PPS,CCH}$	Minimum pulse-to-pulse spacing	CHANNEL_COMBINE = 1 For a single pair of pulses.	III	5			ns
$t_{SU,RST}$	Setup time from RSTIDX to REFCLK		III	5			ns
$t_{HD,RST}$	Hold time from RSTIDX to REFCLK		III	5			ns
$t_{SU,DIS}$	Setup time from STOP to DISABLE		III	5			ns
$t_{HD,DIS}$	Hold Time from STOP to DISABLE		III	5			ns
t_{PIN_ENA}	Pin activation time from configuration of PIN_ENA… to valid data	Pins: RSTIDX, DISABLE, REFCLK, STOP1…4	III	200			µs
t_{POR}	Delay between power-on or initialization reset and next communication	Power-up, opcodes spiopc_power & spiopc_init, pin	III	100			µs

Figure 12:
Timing Symbols and Parameters



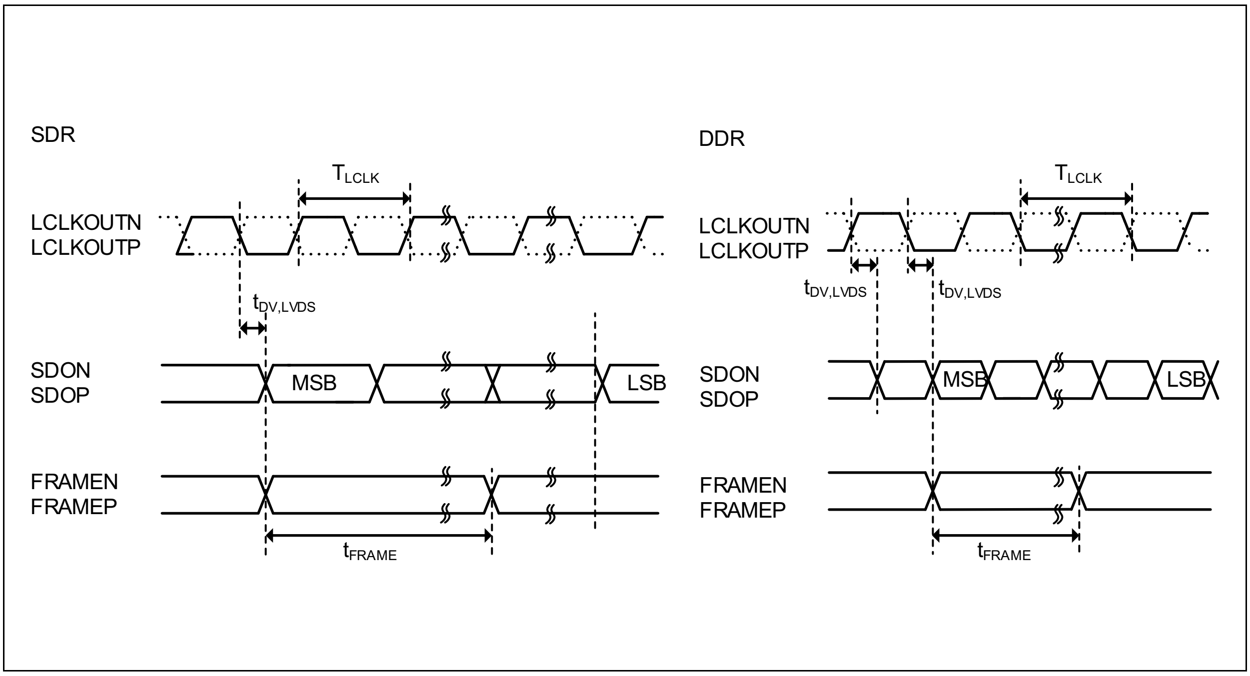
LVDS Data Interface Characteristics

General Conditions: VDD33 = 3.3V, VDD18 = 1.8V, T_A = 0°C to 80°C, V_{ID} = 200 mV, V_{IC} = 1.25V

Figure 13:
Interface Characteristics

Symbol	Description	Condition	TL	Min	Typ	Max	Unit
Electrical Characteristics							
$V_{OD,LVDS}$	LVDS differential output voltage	R_L = 100Ω, C_L = 5pF	III	200			mV
$V_{OC,LVDS}$	LVDS common mode output voltage	R_L = 100Ω, C_L = 5pF	III	1.125	1.25	1.375	V
t_{PIN_ENA}	Pin activation time from configuration PIN_ENA_LVDS to valid data at pin	Pins: LCLKIN, LCLKOUT, SDO1…4, FRAME1…4	III			200	µs
Timing Characteristics							
t_{SYNC}	Synchronization latency	SDR DDR	III		6 3		Clock
t_{FRAME}	Frame length	SDR DDR	III		8 4		Clock
f_{LCLK}	LVDS clock frequency SDR/DDR		III	10		250	MHz
	LVDS clock duty cycle		III	45	50	55	%
	Path delay LCLKIN to LCLKOUT, SDO1…4, FRAME1…4		III		5	10	ns
$t_{DV,LVDS}$	Data valid after active clock edge	lvds_data_valid _ adjust = 1	III		0		ns

Figure 14:
LVDS Timing Symbols and Parameters



Serial Communication Interface

General Conditions: VDD33 = 3.3V; VDD18 = 1.8V;
T_A = 0°C to 80°C; V_{IL} = 0V; V_{IH} = 3.3V

Figure 15:
Serial Communication Interface

Symbol	Description	Condition	TL	Min	Typ	Max	Unit
Electrical Characteristics							
V_{OL}	Digital output LOW voltage	I_O = 2mA	III			0.3	V
V_{OH}	Digital output HIGH voltage	I_O = 2mA	III	DVDD +0.3			V
Timing Characteristics							
f_{SCK}	Serial clock frequency	C_L = 5pF	III			50	MHz
$t_{PWH,SCK}$	Serial clock pulse width HI state		III	10			ns
$t_{PWL,SCK}$	Serial clock pulse width LO state		III	10			ns
$t_{PWH,SSN}$	SSN pulse width between write cycles		III	10			ns
$t_{SU,SSN}$	SSN setup time after SCK falling		III	20			ns
$t_{HD,SSN}$	SSN hold time before SCK rising		III	20			ns
$t_{SU,MOSI}$	Data setup time prior to clock edge		III	5			ns
$t_{HD, MOSI}$	Data hold time after clock edge		III	5			ns
$t_{DV,MISO}$	Data valid after rising clock edge		III	8			ns
$t_{ZX,MISO}$	HighZ to output time		III	8			ns
$t_{XZ,MISO}$	Output to HighZ time		III	8			ns

Figure 16:
Write and Incremental Write

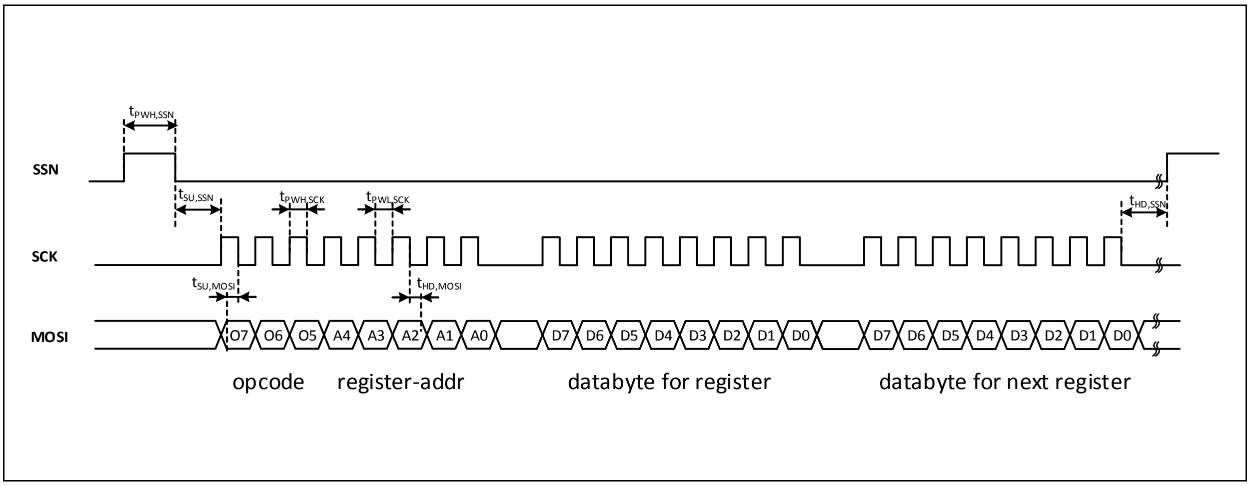
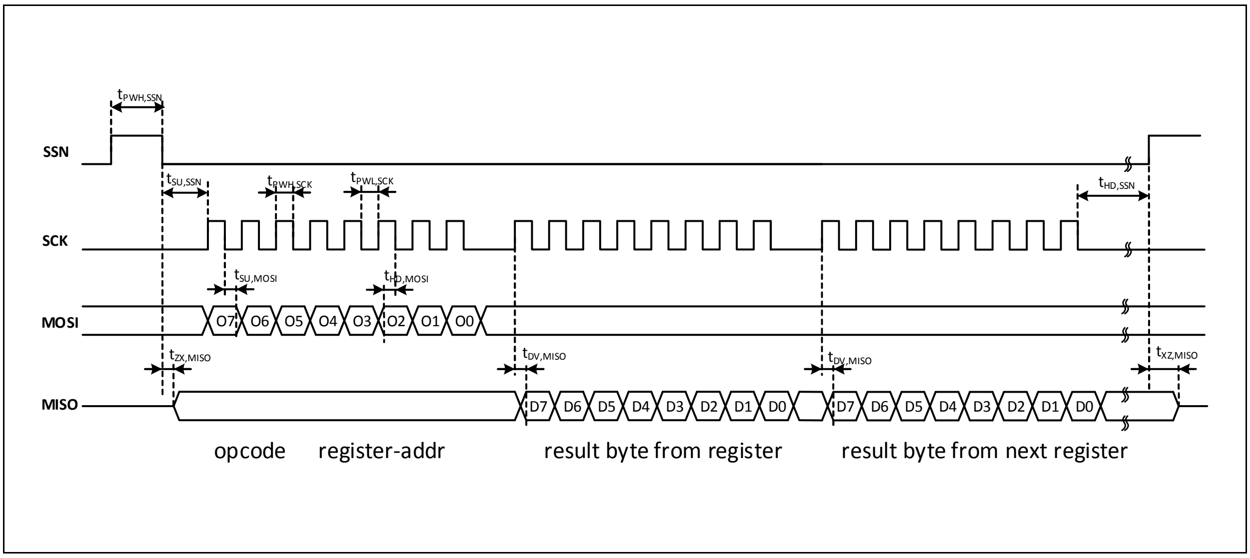


Figure 17:
Read and Incremental Read



Typical Operating Characteristics

Histograms

Figure 18:
STOP2, FWHM, Histogram 100000 Values

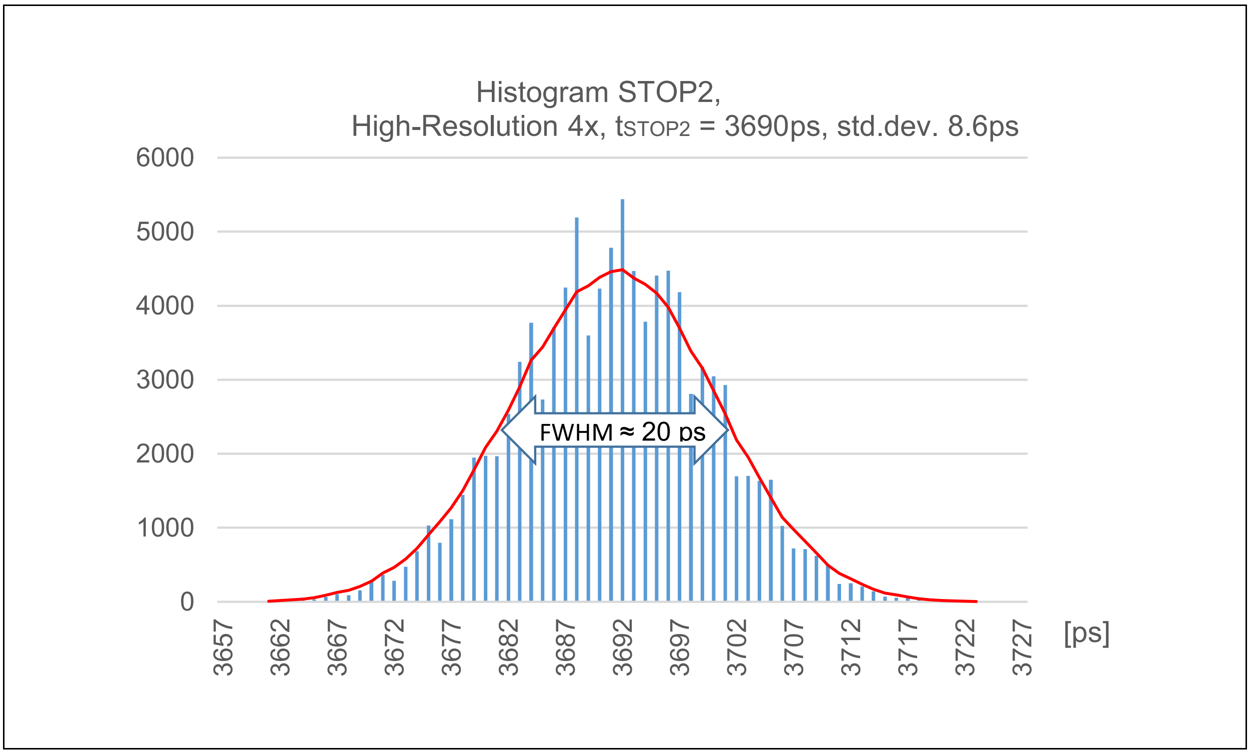


Figure 19:
STOP2 – STOP1, HIGHRES 4x, Histogram 100000 Values

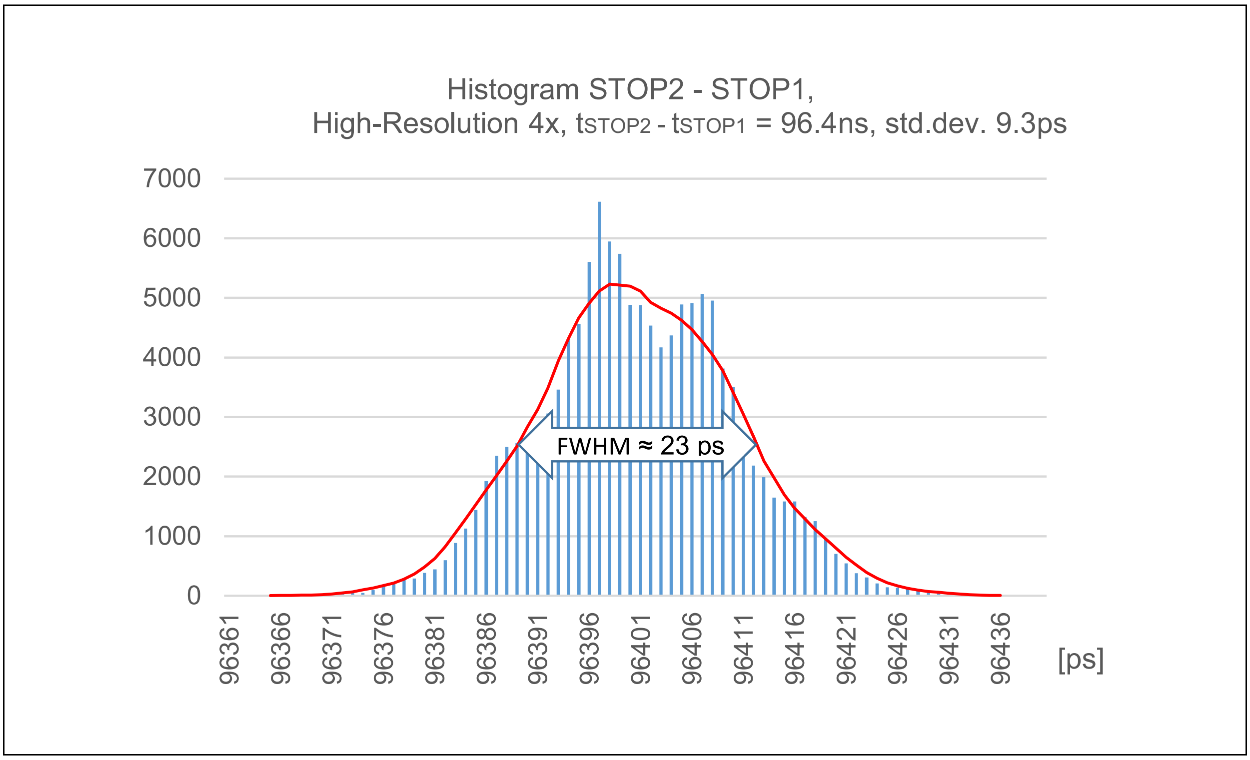


Figure 20:
STOP2 – REFCLK, HIGHRES 0, Histogram 100000 Values

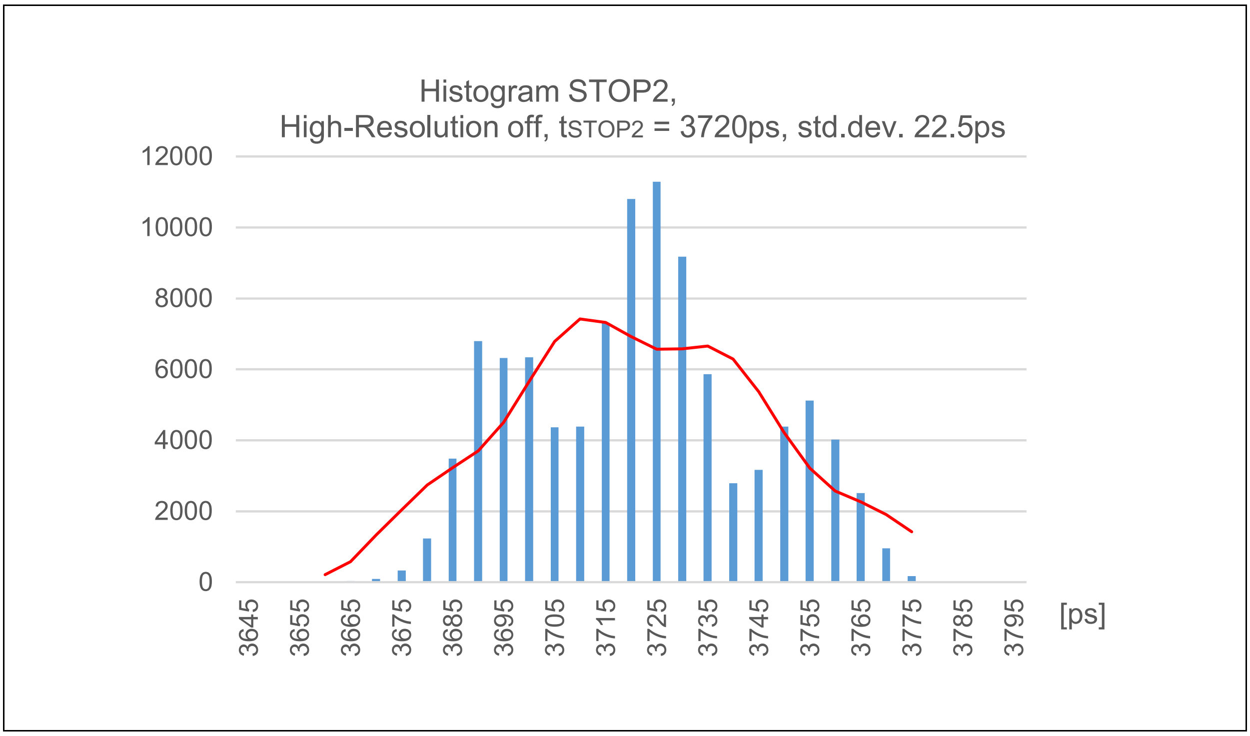
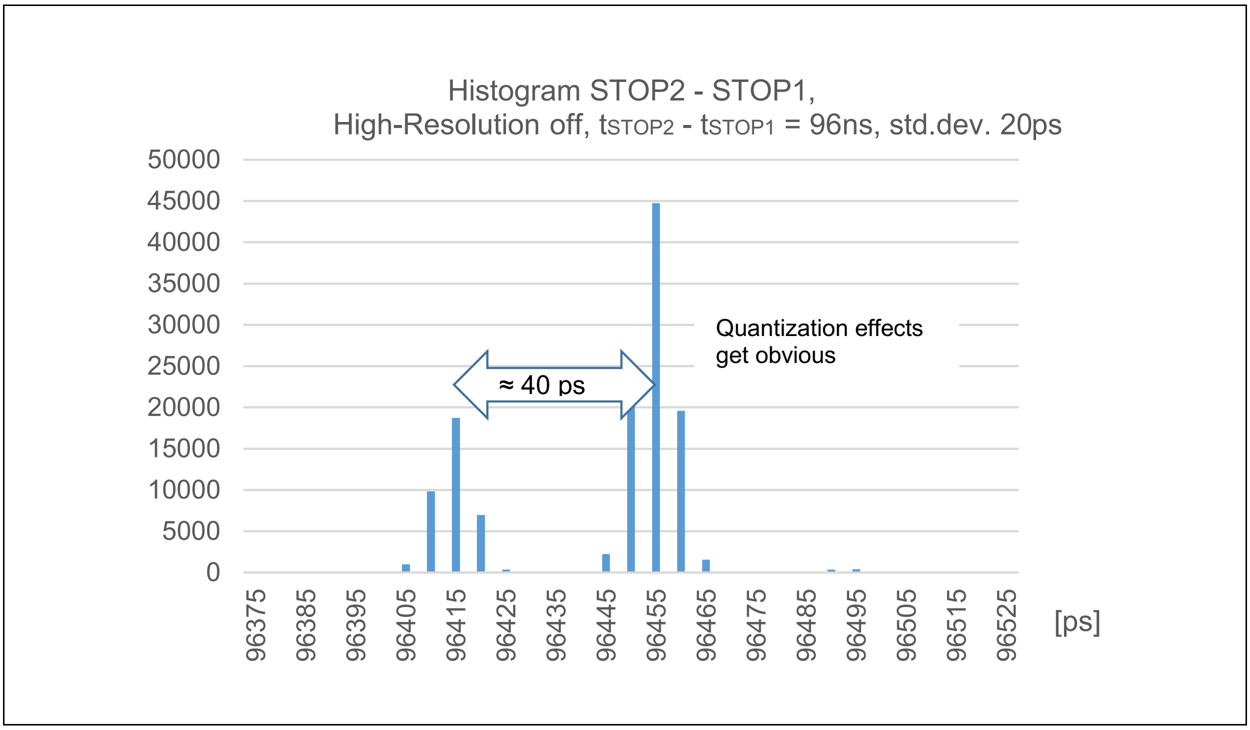
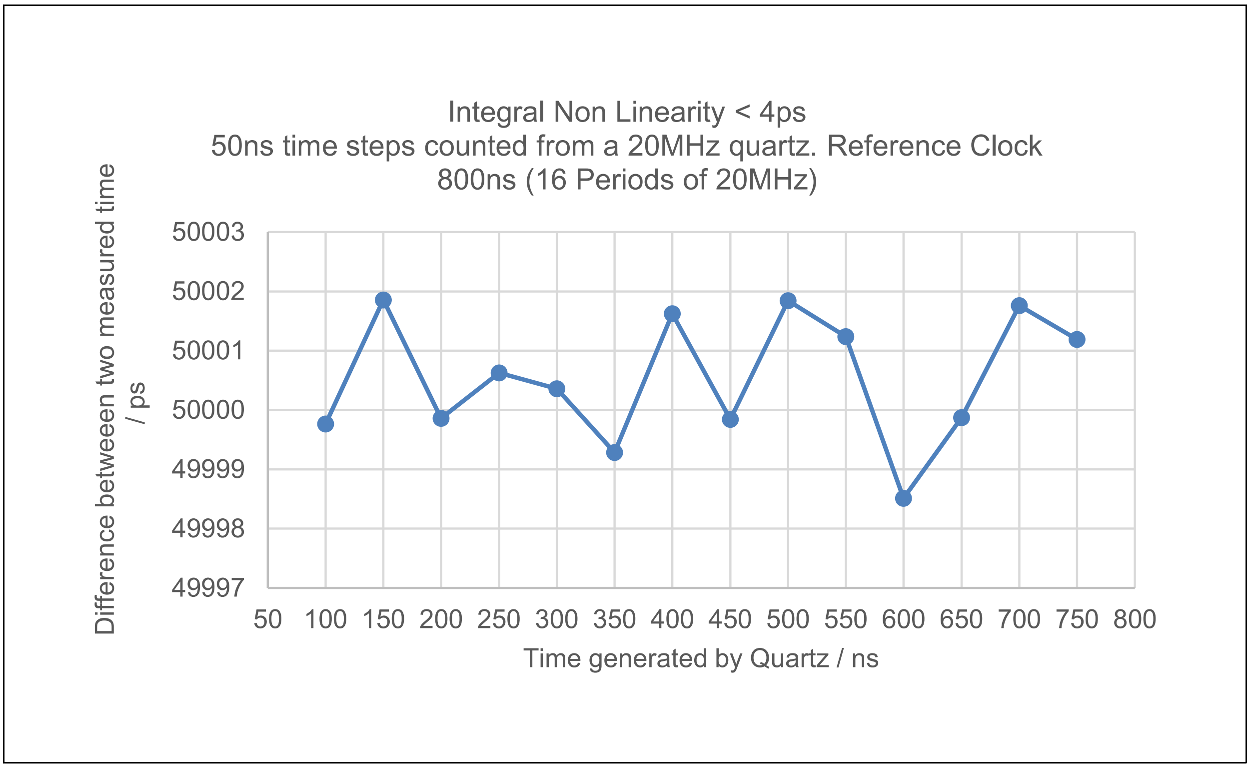


Figure 21:
STOP2 – STOP1, HIGHRES 0, Histogram 100000 Values



Integral Non-Linearity

Figure 22:
Integral Non-Linearity



Register Description

Configuration Register Overview

The configuration registers are organized in 17 addresses of one byte. All configuration registers are accessible via the SPI interface. They can be read and written individually or with an incremental access. For monitoring the chip it is possible to observe at the PARITY pin whether the sum of all set bits is even or odd.

Figure 23:
Configuration Register Overview

Addr	<D7>	<D6>	<D5>	<D4>	<D3>	<D2>	<D1>	<D0>
0	PIN_ENA_RSTIDX	PIN_ENA_DISABLE	PIN_ENA_LVDS_OUT	PIN_ENA_REFCLK	PIN_ENA_STOP4	PIN_ENA_STOP3	PIN_ENA_STOP2	PIN_ENA_STOP1
1	HIGH_RESOLUTION		CHANNEL_COMBINE		HIT_ENA_STOP4	HIT_ENA_STOP3	HIT_ENA_STOP2	HIT_ENA_STOP1
2	BLOCKWISE_ FIFO_READ	COMMON_FIFO_ READ	LVS_DOUBLE_DATA_ RATE	STOP_DATA_BITWIDTH		REF_INDEX_BITWIDTH		
3	REFCLK_DIVISIONS (Lower byte)							
4	REFCLK_DIVISIONS (Middle byte)							
5	Fixed value*: (0000b)				REFCLK_DIVISIONS (Upper bits)			
6	Fixed value*: (110b)			LVDS_ TEST_ PATTERN	Fixed value*: (0000b)			
7	REFCLK_BY _ XOSC	Fixed value*: (1b)	LVDS_DATA_VALID_ ADJUST		Fixed Value*: (0011b)			
8	Fixed value*: (10100001b)							
9	Fixed value*: (00010011b)							
10	Fixed value*: (00000000b)							
11	Fixed value*: (00001010b)							

Addr	<D7>	<D6>	<D5>	<D4>	<D3>	<D2>	<D1>	<D0>
12	Fixed value*: (11001100b)							
13	Fixed value*: (11001100b)							
14	Fixed value*: (11110001b)							
15	Fixed value*: (01111101b)							
16	Fixed value*: (00000b)					CMOS_ INPUT	Fixed value*: (00b)	

The fixed values are assigned by ams: Unless otherwise suggested, they should be set as shown in this table.

Detailed Configuration Register Description

All registers are read/write with 0 as default value, besides registers 13, 14 with 5 as default value.

Figure 24:
Configuration Register 0

Addr: 0		Pin Enable Register
Bit	**Bit Name**	**Bit Description**
The PIN_ENA registers activate the LVDS input or output drivers of the related pins. Main purpose of PIN_ENA is cutting of current consumption of differential LVDS buffers to nearly zero. But also with CMOS input levels the pins have to be activated accordingly. Unused inputs has to be tied to VDD33.		
0 to 3	PIN_ENA1 to PIN_ENA4	Activation on stop event input pins STOP1 to STOP4 0: Stop input pins not active 1: Stop input pins active
4	PIN_ENA_ REFCLK	0: REFCLK input pins not active 1: REFCLK input pins active
5	PIN_ENA_ LVDS_OUT	0: All LDVS output pins disabled 1: Activation of LCLK and LCLKOUT pins. Activation of SDO1…4 and FRAME1…4, depends further on CHANNEL_COMBINE and PIN_ENA
6	PIN_ENA_ DISABLE	0: Stop disable pin is not active. The stop measurement on all channels is always active according to configuration. 1: Stop disable pin is active. The stop measurements are disabled if the DISABLE pin on the PCB is set to HIGH
7	PIN_ENA_ RSTIDX	0: Deactivation of reference clock index counter reset pin 1: Activation of reference clock index counter reset pin

Figure 25:
Configuration Register 1

Addr: 1		Content
Bit	**Bit Name**	**Bit Description**
0 to 3	HIT_ENA1 to HIT_ENA4	0: Stop events are internally rejected. The pin enabling of STOP1…4 is not affected. 1: Stop events are internally accepted and processed. Normal working condition
4, 5	CHANNEL_ COMBINE	The four stop channels may be combined for improved pulse pair resolution or higher conversion rate. 00b: Normal operation with four independent stop channels 01b: "Pulse distance" Stop events at STOP1 are measured alternatingly by stop channels 1 & 3 Stop events at STOP2 are measured alternatingly by stop channels 2 & 4 10b: "Pulse width" The rising edges at STOP1 are measured by stop channel 1 The falling edges at STOP1 are measured by stop channel 3 The rising edges at STOP2 are measured by stop channel 2 The falling edges at STOP2 are measured by stop channel 4
6, 7	HIGH_ RESOLUTION	A stop event is internally delayed, measured several times and summed up in order to one result to increase the time resolution. = 0 (off): Off, standard resolution with minimal pulse-to-pulse spacing. = 1 (2x): A stop event is measured twice = 2 (4x): A stop event is measured four times

Figure 26:
Configuration Register 2

Addr: 2		Data Output
Bit	**Bit Name**	**Bit Description**
0 to 2	REF_INDEX_ BITWIDTH	Bit width of reference clock index in LVDS output (not applicable to SPI data readout) 000b: 0Bit, no data out 001b: 2Bits 010b: 4Bits 011b: 8Bits 100b: 16Bits 101b: 24Bits 110b: 6Bits 111b: 12Bits
3, 4	STOP_DATA_ BITWIDTH	Bit width of the stop result in LVDS output. Bit width should be sufficient to represent the REFCLK_DIVISIONS configuration value (not applicable to SPI data readout) 00b: 14Bits → max of REFCLK_DIVISIONS = 2^{14}-1 01b: 16Bits → max of REFCLK_DIVISIONS = 2^{16}-1 10b: 18Bits → max of REFCLK_DIVISIONS = 2^{18}-1 11b: 20Bits → max of REFCLK_DIVISIONS = 2^{20}-1
5	LVDS_ DOUBLE _ DATA_RATE	0: Single Data Read (SDR): The LVDS data clocked out on rising edges of LCLK-OUT 1: Double Data Read (DDR): The LVDS data are clocked on both edges of LCLK-OUT
6	COMMON_ FIFO_READ	0: LVDS: Operation with four independent stop channels SPI: INTERUPT pin is set to zero, as soon as one FIFOs does have a value. OFF, operation with four independent stop channels 1: LVDS: All active frame pins are set simultaneous as soon as all related FIFOs have values. SPI: INTERUPT pin is set to zero, as soon as all active FIFOs have value. In combination with BLOCKWISE_READ this option guaranties successive measurements in parallel on all stop channels
7	BLOCKWISE_ FIFO_READ	0: OFF, Operation with standard FIFO function 1: Data output (LVDS or SPI) is not started before a channel FIFO is full. Once FIFO is full, measurement is not restarted before FIFO is completely read-out. This option guaranties successive measurements at high stop event rate or slow read-out speeds (e.g. SPI)

Figure 27:
Configuration Register 3, 4, 5

Addr: 3, 4, 5		Reference Clock Divider
Bit	**Bit Name**	**Bit Description**
0 to 7 0 to 7 0 to 3	REFCLK_ DIVISIONS	Defines a LSB at the output interface as fraction of the reference clock period. The most convenient way is applying a LSB of 1ps by configuring REFCLK_DIVISIONS to the picosecond value of the reference clock period address 3 lower 8bits, address 4 middle 8bits, address 5 upper 4bits

Figure 28:
Configuration Register 6

Addr: 6		Content
Bit	**Bit Name**	**Bit Description**
4	LVDS_TEST_ PATTERN	0: Normal operation of LVDS outputs 1: LVDS interface continuously outputs the following test patterns. All stop events are ignored. Reference index = 111100001100110010101010bin (=15781034dec) Stop result = 00001010101011001111 0000bin (=699632dec) Depending on the configuration of the output format width (REF_INDEX_BITWIDTH, STOP_DATA_BITWIDTH) only the corresponding lower bits are transmitted
5 to 7	Fixed value	110b: Defined by **ams**

Figure 29:
Configuration Register 7

Addr: 7		Pin Enable Register
Bit	**Bit Name**	**Bit Description**
0 to 3	Fixed value	0011b: Defined by **ams**
4, 5	LVDS_DATA_ VALID_ ADJUST	Adjustment of the data valid time at the LVDS output interface. 000b: - 160ps 001b: 0ps 010b: +160ps 011b: +320ps
6	Fixed value	1b: Defined by **ams**
7	REFCLK_BY_ XOSC	0: Reference pulses have to be applied at REFCLK pins. The circuit for driving the external quartz is not in use. 1: The reference clock is generated by a quartz which is connected to the GPX2; REFCLK pins are not in use and should be disabled with PIN_ENA_REFCLK.

For registers 8 to 15 use the default fixed values as shown in the register overview.

Figure 30:
Configuration Register 16

Addr: 16		Pin Enable Register
Bit	**Bit Name**	**Bit Description**
0 to 1	Fixed value	00b: Defined by **ams**
2	CMOS_INPUT	Input voltage levels of STOP1 to STOP4, REFCLK, RSTIDX and DISABLE are selected as CMOS or LVDS 0: Differential LVDS input level. 1: Single ended CMOS input level Also with CMOS input level the pins have to be activated with according PIN_ENA-configuration
3 to 7	Fixed value	00000b: Defined by **ams**

Read Register Overview

All read registers are accessible via SPI Interface. Incremental read may start at any register address.

Figure 31:
Read Register Overview

Addr	Name	<D6>
0	Status	n.c.
1		n.c.
2		n.c.
3		n.c.
4		n.c.
5		n.c.
6		n.c.
7		n.c.
8	Channel1	REFERENCE INDEX CH1 BYTE #3
9		REFERENCE INDEX CH1 BYTE #2
10		REFERENCE INDEX CH1 BYTE #1
11		STOP RESULT CH1 BYTE #3
12		STOP RESULT CH1 BYTE #2
13		STOP RESULT CH1 BYTE #1
14	Channel2	REFERENCE INDEX CH2 BYTE #3
15		REFERENCE INDEX CH2 BYTE #2
16		REFERENCE INDEX CH2 BYTE #1
17		STOP RESULT CH2 BYTE #3
18		STOP RESULT CH2 BYTE #2
19		STOP RESULT CH2 BYTE #1
20	Channel3	REFERENCE INDEX CH3 BYTE #3
21		REFERENCE INDEX CH3 BYTE #2
22		REFERENCE INDEX CH3 BYTE#1
23		STOP RESULT CH3 BYTE #3
24		STOP RESULT CH3 BYTE #2
25		STOP RESULT CH3 BYTE #1

Addr	Name	<D6>
26	Channel4	REFERENCE INDEX CH4 BYTE #3
27		REFERENCE INDEX CH4 BYTE #2
28		REFERENCE INDEX CH4 BYTE #1
29		STOP RESULT CH4 BYTE #3
30		STOP RESULT CH4 BYTE #2
31		STOP RESULT CH4 BYTE #1

Detailed Description

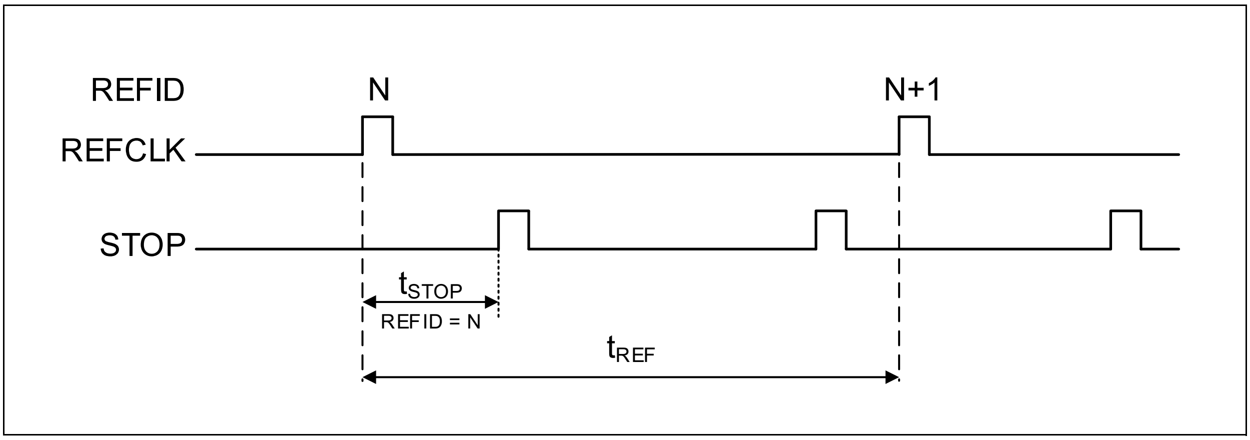
Time Measurements and Results

Measurements of TDC-GPX2

The reference clock is the framework for all time measurements. The clock pulses are measured continuously by the TDC as time reference point for stop pulses and as internal reference period. The measurement of the stop events always refers to the preceding reference clock. Additionally, the reference clock is counted continuously and the actual count is assigned as reference index to a stop pulse.

- **t_{REF}** is the internal TDC measurement of the reference clock period
- **t_{STOP}** is the internal TDC measurement of a stop to the preceding reference clock
- **REFID** is the index of reference period where the measured stop occurred

Figure 32:
TDC-GPX2 Time Measurement



Output Results

Each stop generates a dataset which consists of two values **TSTOP** and **REFID**:

REFID is the reference index of the preceding reference clock pulse to TSTOP. The reference index is necessary to indicate the relationship of stop pulses which belong to different reference clock periods. The maximum length of the reference index is 24 bits.

TSTOP is the ratio of the internal measured times of t_{STOP} over t_{REF} scaled by the configured REFCLK_DIVISONS. The readout result TSTOP is always less than configured REFCLK_DIVISONS. The resulting LSB at the output interface has to be chosen much lower than the single shot resolution of GPX2. For details see chapter "Coding of Results". Suitable values are e.g. 1ps, 5ps or 10ps.

Figure 33:
t_{STOP} Calculation

$$t_{STOP} = \overbrace{\underbrace{\frac{t_{STOP}}{t_{REF}}}_{\text{Ratio of internal time measurements}} \times REFCLK\_DIVISIONS}^{TSTOP} \times \overbrace{\frac{t_{REFCLK\text{-}PERIOD}}{REFCLK\_DIVISIONS}}^{LSB}$$

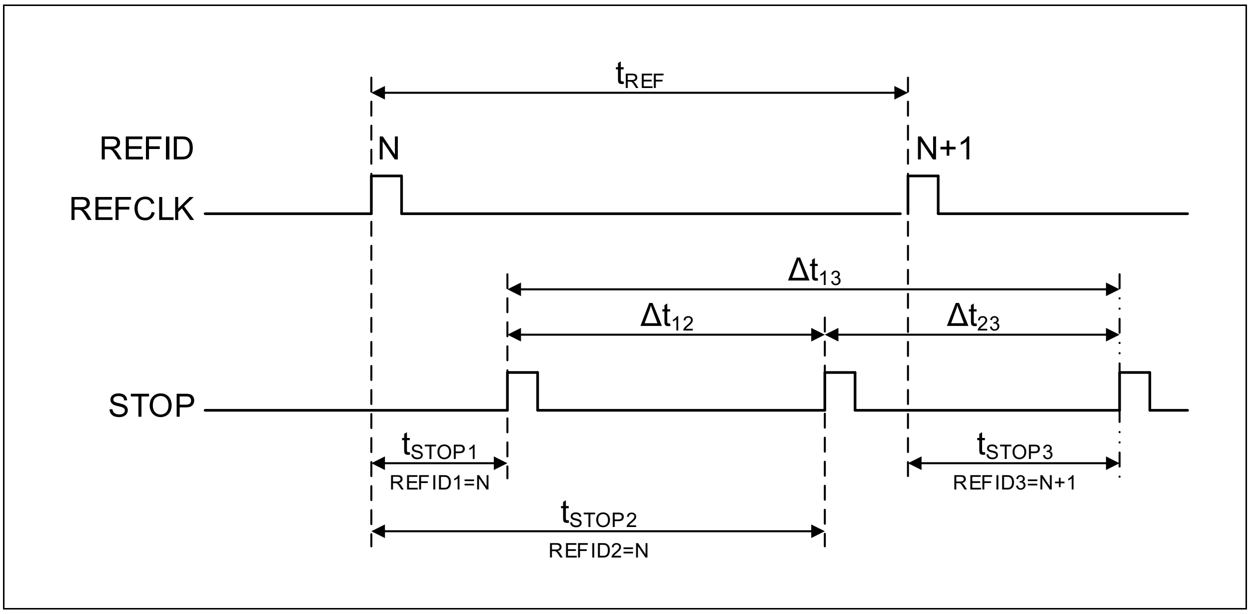
Internal calculated result for read-out

LSB resulting by the period of the applied reference clock and by the configured REFCLK_DIVISIONS

Calculation of Time Differences

The results of the GPX2 are the time intervals from stop event pulses to the preceding reference clock pulses. In many applications the time difference between stop event pulses is desired. This happens e.g. in case of a quartz as a reference clock. Depending on the application and the measurement setup, several approaches are possible to calculate the time between two stops in the connected microprocessor or FPGA.

Figure 34:
Calculating Time Differences



General Approach

On the output interface, either SPI or LVDS, both data REFID and TSTOP are available. With these data it is possible to calculate time differences between stops. The maximum time difference depends on the bit width of the reference index (see also chapter "Maximum Time Differences" between stops depending on the reference index bit width)

Δt_{13} = (TSTOP3 - TSTOP1) + (REFID3 - REFID1) * REFCLK_DIVISIONS

In two special cases it is not necessary to readout the REFID:

Stops in the Same Reference Clock Period

In applications where stops occur always in the same reference period (e.g. STOP1 & STOP2), it is not necessary to read out the reference index. It is sufficient just read out the stop results and to calculate the difference:

Δt_{12} = TSTOP2 - TSTOP1; REFID2 = REFID1

Time Difference Smaller Than Reference Clock

In applications where the measured time difference Δt is always smaller than the reference clock period T_{REF} but not necessarily in the same reference clock period (e.g. STOP2 & STOP3), it is often sufficient to read out just the stop results without the reference index by distinguishing positive and negative time difference:

If TSTOP3 - TSTOP2 > 0

- Δt_{23} = (TSTOP3 – TSTOP2)

If TSTOP3 - TSTOP2 < 0 and ΔT < REFCLK_DIVISIONS

- Δt_{23} = (TSTOP3 – TSTOP2) + REFCLK_DIVISIONS

Resolution

RMS-Resolution Versus Effective Resolution

The RMS resolution of a TDC is the root-mean-square-value of a set of single shot time measurements. TDC do not have an obvious full scale definition, as the time they are measuring is unlimited. Therefore, the definition of an effective resolution in number of bits likewise in ADC is not feasible.

High Resolution

For achieving best single-shot RMS resolution, GPX2 offers a complete integrated solution. During the initial sampling the stop event is internally delayed and sampled again, after the first sample was stored in the FIFO. All samples of one stop event are averaged inside of the GPX2 and occur as one result with lower conversion noise at the output interface. With HIGH_RESOLUTION it is possible to configure internal 2 or 4 samples of one event. Due of the internal delay and the multiple samples the conversion latency t_{conv} and the pulse-to-pulse spacing t_{PPS} increase as well as the maximum FIFO_DEPTH decreases. In order to compensate these drawbacks, it is possible to use HIGH_RESOLUTION with both CHANNEL_COMBINATION modes and to achieve the excellent pulse-to-pulse spacing of channel combination mode, doubled FIFO depth per stop input and higher resolution.

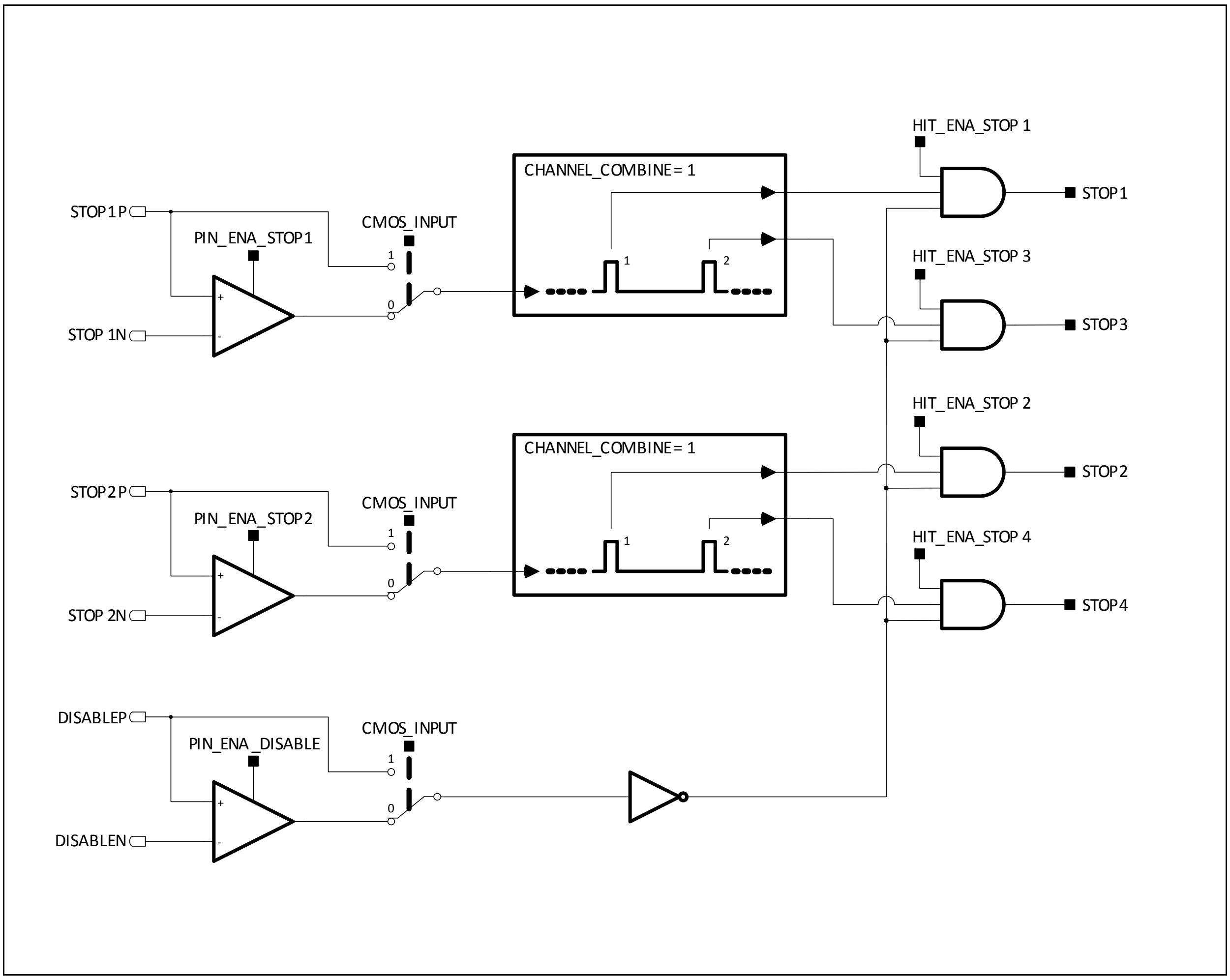
Combining Two Stop Channels

Channel Combination for Low Pulse-to-Pulse Spacing

With CHANNEL_COMBINE set to "PULSE_SPACING", two stop channels 1 & 3 (and 2 & 4) are connected to one input pin STOP1 (and STOP2). The stop events at the input pin are distributed alternatingly between the combined channels. Readout is indicated via FRAME or INTERRUPT pins when both channels have results in their FIFO. The advantage of combining channels lies in improved pulse-to-pulse spacing

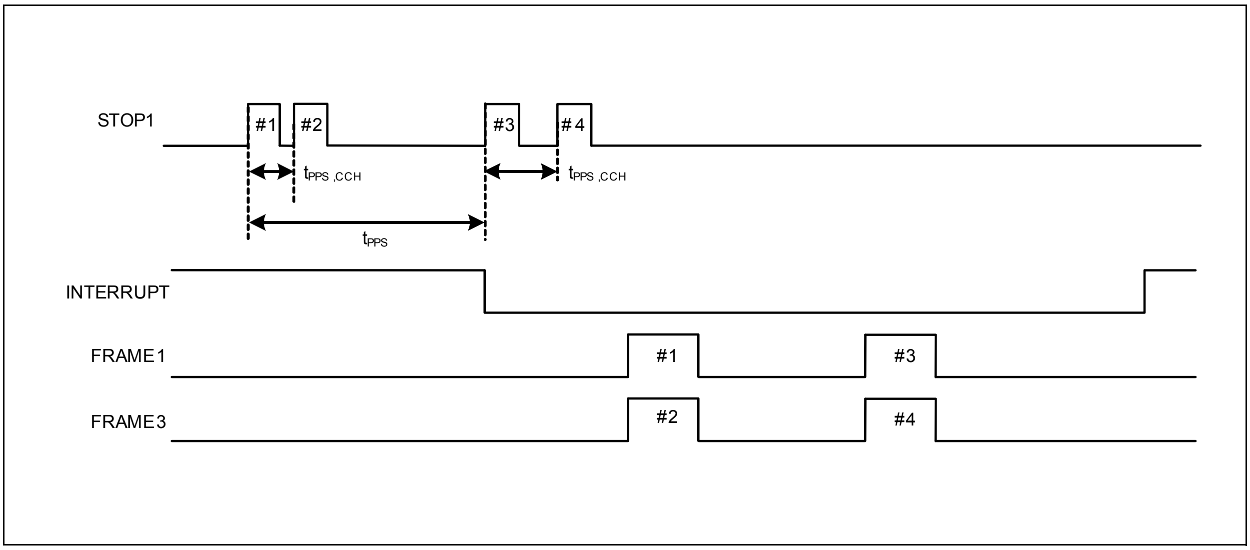
- Excellent pulse-to-pulse spacing
- Doubled FIFO depth per stop input pin
- Higher burst storage capability
- Doubled LVDS readout rate per stop input pin
- HIGH_RESOLUTION is applicable

Figure 35:
Channel Combination Low Pulse-to-Pulse Spacing



The outstanding low pulse-to-pulse spacing $t_{PPS,CCH}$ is achievable only for a single pulse pair. After a pulse pair, the regular pulse-to-pulse spacing t_{PPS} must be awaited, before capturing the next pulse becomes possible. Measurements with HIGH_RESOLUTION will increase the regular pulse-to-pulse spacing but the low pulse-to-pulse spacing $t_{PPS,CCH}$ is not affected.

Figure 36:
Channel Combination Low Pulse-to-Pulse Spacing



Note(s):

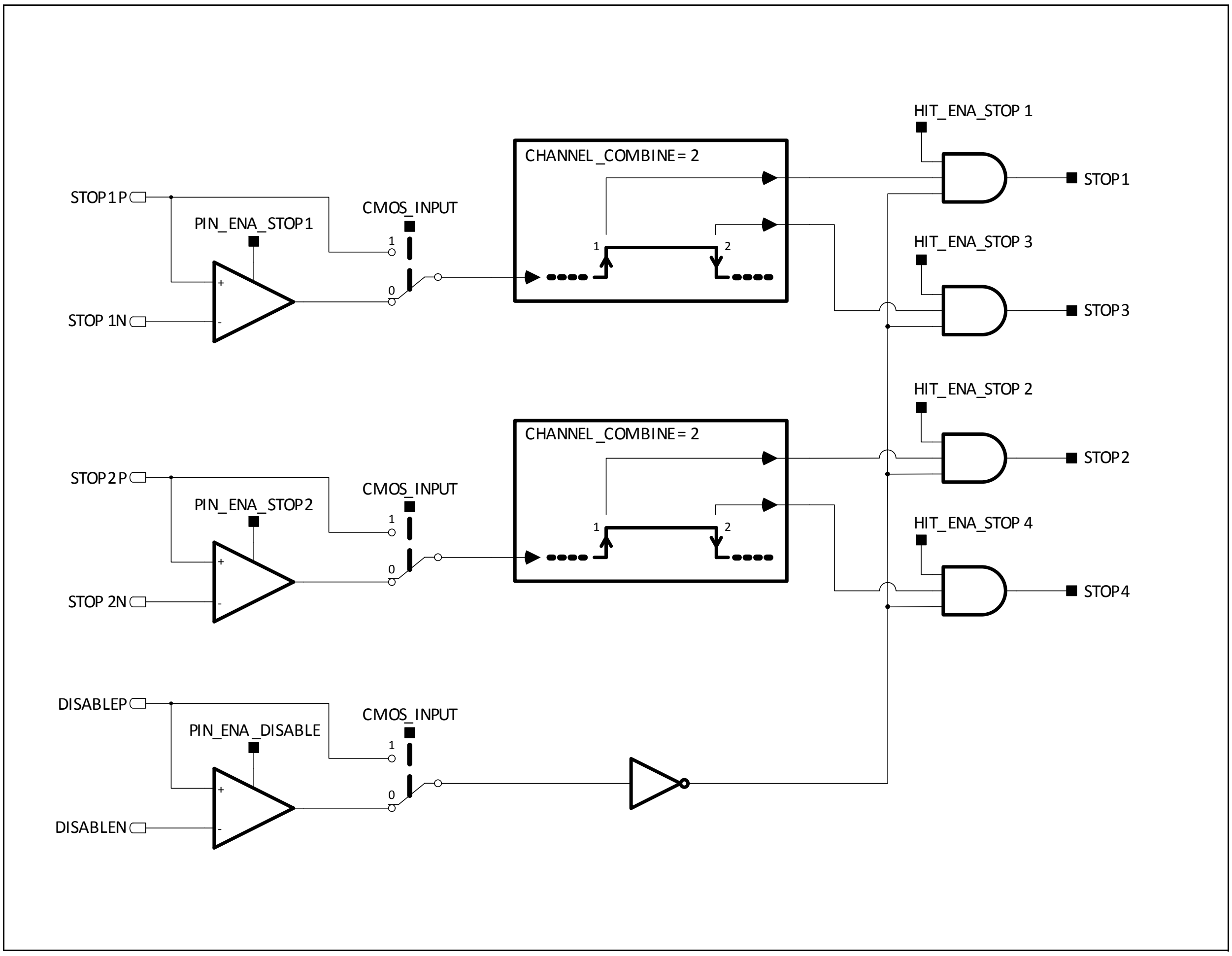
- With LVDS outputs the FRAME pins of combined channels are active together
- SPI readout of combined channel pairs is permitted only pairwise like ch1-ch3-ch1-ch3-… or ch2-ch4-ch2-ch4-… . Also incremental readout like ch1-ch2-ch3-ch4… is possible. But it is not permitted to read one channel twice like ch1-ch1-ch3-ch3-.. or ch2-ch2-ch4-ch4….

Channel Combination for Pulse Width Measurement

With CHANNEL_COMBINE set to “PULSE_WIDTH” two internal stop channels 1&3 (and 2&4) are connected to one input pin STOP1 (and STOP2). The rising edges are measured by channel 1 (2), falling edges are measured by channel 3 (4). Readout starts on both channels simultaneous when a rising and falling edge was measured.

- HIGH_RESOLUTION or COMMON_FIFO_READ is fully applicable

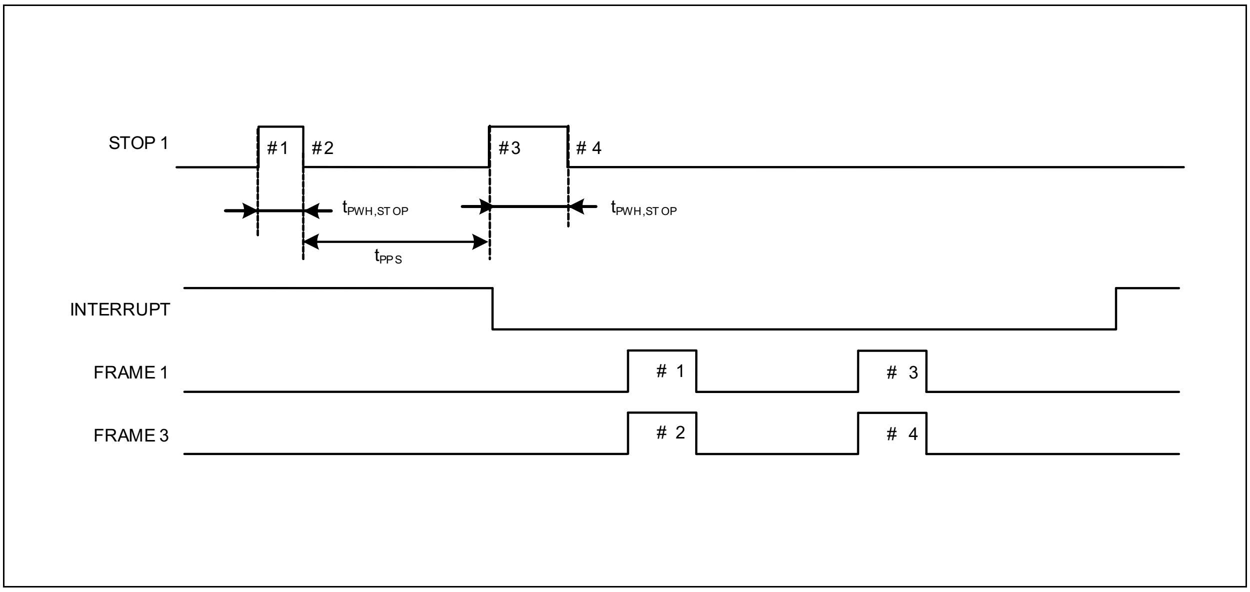
Figure 37:
Channel Combination for Pulse Width Measurement



Note(s):

1. For internal processing reasons, after the conversion latency t_{PPS} must be waited before capturing the next pulse. Measurements with HIGH_RESOLUTION will increase the conversion latency but minimum pulse width $t_{PWH,STOP}$ is not affected.

Figure 38:
Channel Combination Pulse Width Measurement



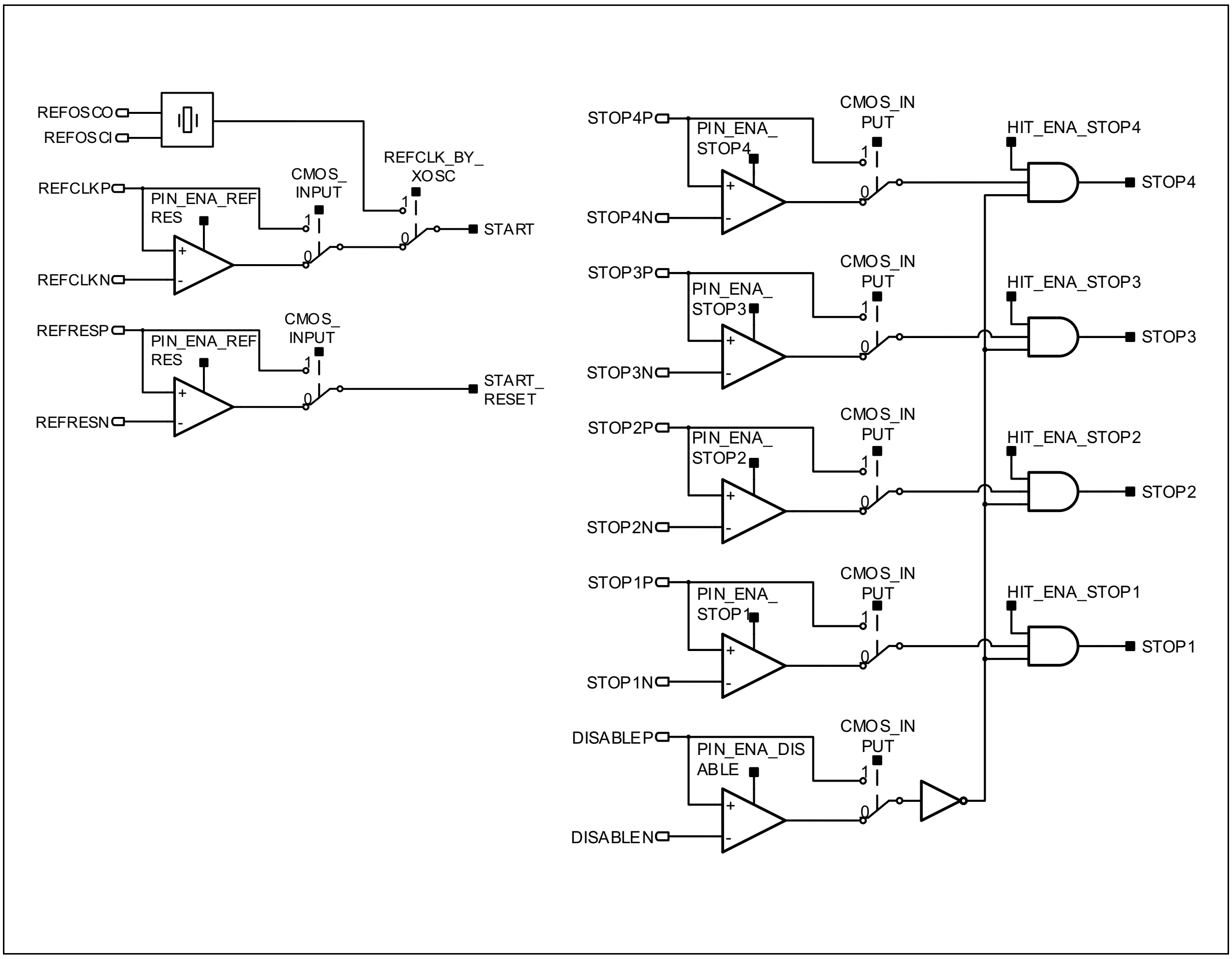
Note(s):

- With LVDS output the FRAME pins of combined channels are active together
- SPI readout of combined channel pairs is permitted only pairwise like ch1-ch3-ch1-ch3-… or ch2-ch4-ch2-ch4-…. Also incremental readout like ch1-ch2-ch3-ch4… is possible. But it is not permitted to read one channel twice like ch1-ch1-ch3-ch3-.. or ch2-ch2-ch4-ch4….

Input Pins for Time Measurement

The following diagram show the relevant input pins for the reference and the stops.

Figure 39:
Input Circuitry



REFCLKP/N: Reference Clock Input

The reference clock serves as universal time base. Due to internal averaging, the phase jitter of the reference clock is non-critical. The accuracy and drift of the reference clock also does not affect the proper working of GPX2 itself. But it will directly affect the quality of the time measurement results.

REFOSCI/O: Quartz Driver as Reference Clock

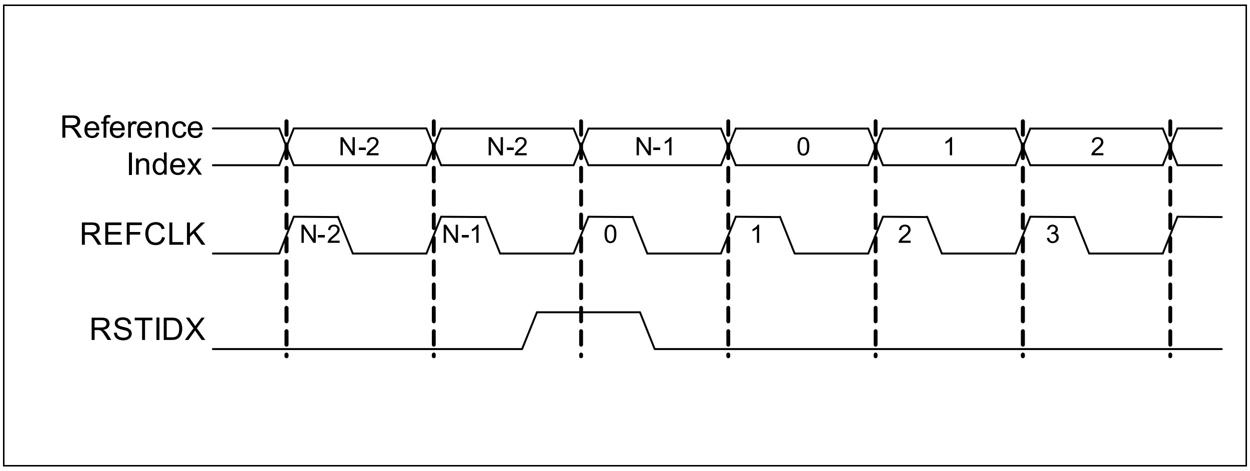
Note(s): The quartz is not mandatory for operation of GPX2.

The quartz is just an optional source for the reference clock. It can be used instead of a clock signal at the reference clock pin. Therefore REFCLK pins should be disabled. With a quartz as reference clock usually the time difference between stops channels is relevant (see chapter "Calculation of Time Differences"). The use of COMMON_FIFO_READ and BLOCKWISE_FIFO_READ can help to measure and read out associated stop results together.

RSTIDXP/N: Reference Index Counter Reset

With pin RSTIDX the internal counter for the reference index is set back to zero. This option may simply the overview on the reference index in the output data stream. RSTIDX is applied synchronously to the reference clock for a single period. Therefore one reference clock cycle passes, before stop events are assigned with zero as reference index. The pin has to be activated with PIN_ENA_RSTIDX.

Figure 40:
Reference Index Counter Reset



STOP1...STOP4P/N: Stop Channels

Inputs for the stop signals. The positive edges of the stop signals are measured versus the preceding reference clock edge.

The chip has four independent stop channels. With CHANNEL_COMBINE variations of this normal operation mode can be achieved.

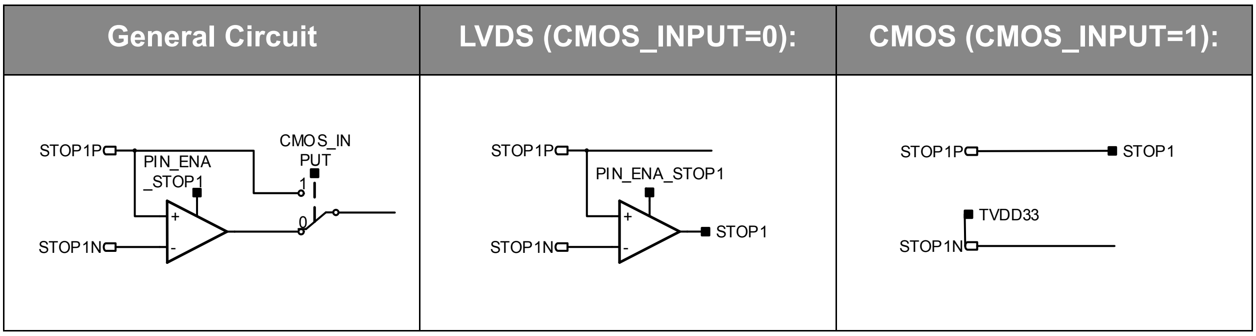
DISABLE/N: Stop Disable

With setting stop disable pin to HIGH, the measurement on all four stops is disabled. The reference clock is not affected and internal reference measurements are continued. The DISABLE should meet the timing requirement with regards to a stop event. The pin has to be activated by configuring PIN_ENA_DISABLE to 1.

Input Levels, CMOS or LVDS

All input pins, STOP1 to STOP4, REFCLK, RSTIDX and DISABLE, can be switched in common to CMOS input levels with CMOS_INPUT configuration. Tie the unused negative inputs to TVDD33.

Figure 41:
CMOS-LVDS



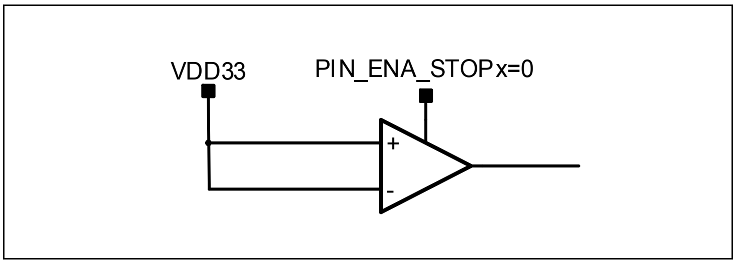
Termination of Differential LVDS Input Pin

Integrated termination is not provided. It is necessary to place termination resistors on the PCB near to the input pins. The default termination for LVDS signals is to have single 100Ω resistors between the differential lines.

Connection of Unused LVDS Inputs

Any kind of unused LVDS inputs (e.g. STOP1 to STOP4, REFCLK, RSTIDX, DISABLE, LCLKIN) have to be pulled up to VDD33 and disabled by setting PIN_ENA to zero. Unused channels should also be switched off with HIT_ENA_STOP1…4.

Figure 42:
Unused LVDS



Software Enable (HIT_ENA_STOP1…4)

Setting the configuration bits HIT_ENA_STOP1 to HIT_ENA_STOP4 applies a software enable for stop channels 1 to 4.

Pin Enable (PIN_ENA_xxx)

The pin enable registers PIN_ENA_STOP1 to PIN_ENA_STOP4, PIN_ENA_REFCLK, PIN_ENA_RSTIDX and PIN_ENA_DISABLE activate the LVDS input or output drivers of the related pins. Main purpose of PIN_ENA is cutting of current consumption of unused differential LVDS buffer to nearly zero. But also with CMOS_INPUTs the pin need to be activated. In case of the LVDS output interface, PIN_ENA_STOP1 to PIN_ENA_STOP4 enable also the according LVDS output drivers.

LVDS Output Interface

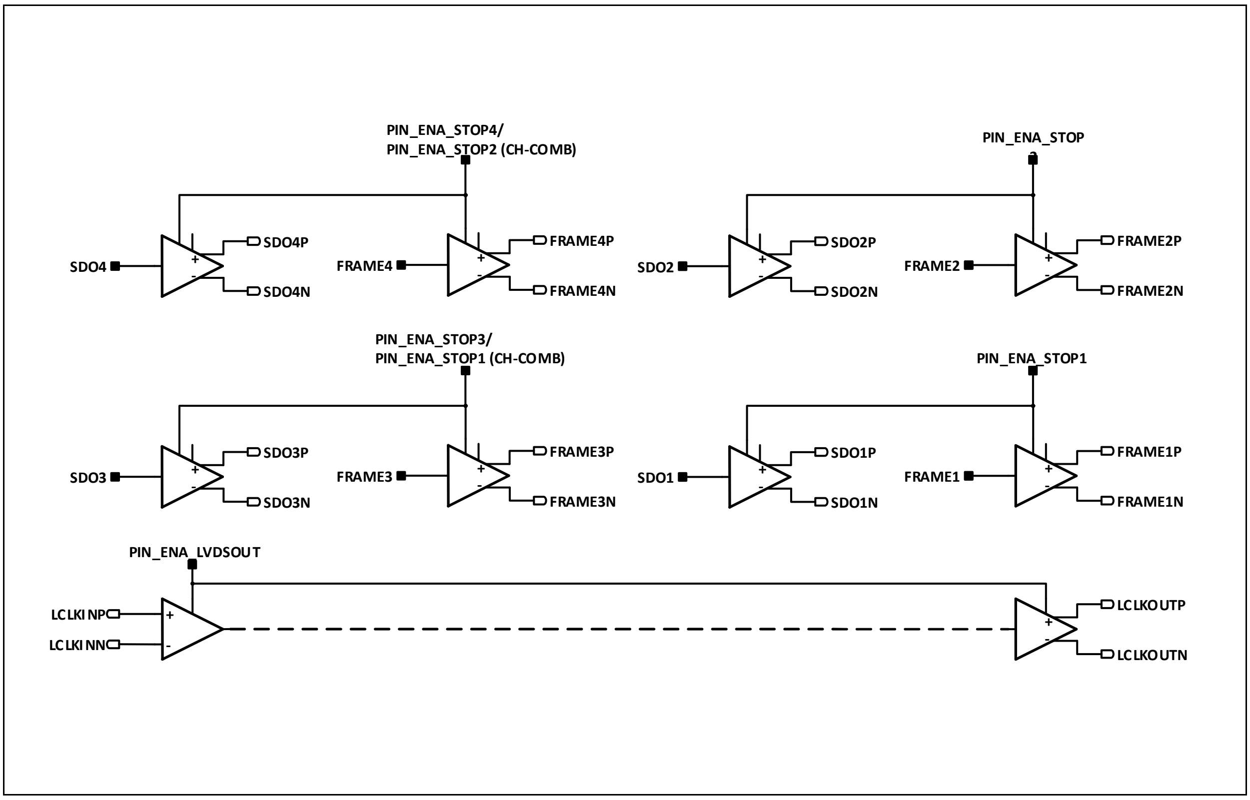
Digital Output Interface

Each stop channel has its own serial interface with a data output SDO pin and a FRAME pin to indicate the MSB. Data output is supported on falling edges (SDR, single data read) or rising and falling edges (DDR, double data read). The operating clock is looped from LCLKIN through the chip to LCLKOUT pin. The data at SDO and FRAME pins have stable timing relation a $t_{DV,LVDS}$ to LCLKOUT. The FRAME indicate the first 8bits of an output sequence. On the SDO pin the reference index is output first, and the stop result follows that. The bit width of both results is configurable by STOP_DATA_BITWIDTH and REF_INDEX_BITWIDTH. With careful configuration data overhead can be avoided in favor of higher conversion rates.

Output Setup and Configuration:

LVDS output interface is activated configuring LVDS_ENA_LVDSOUT =1. The clock at the input LCLKIN is looped through the chip to pins LCLKOUT. The phase of SDO and FRAME pins are in stable relation to LCLKOUT. The SDO and FRAME pins needed for output are activated according to the configuration of PAD_ENA_STOP1 to PAD_ENA_STOP4 and CHANNEL_COMBINE.

Figure 43:
LVDS Outputs



LVDS Output Buffers

The LVDS output buffers SDO1 to SDO4, FRAME1 to FRAME4, and LCLKOUT are designed for 200mV voltage swing with external 100Ω termination.

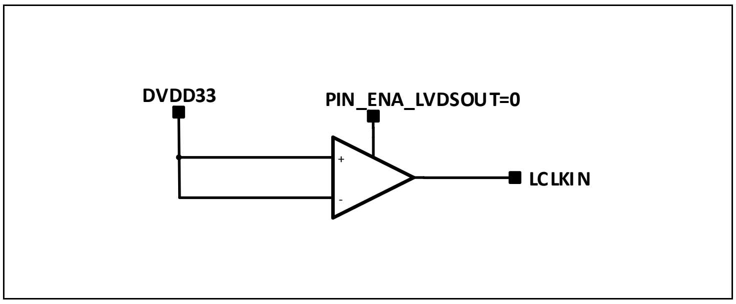
Unused LVDS output buffers can be left open.

Differential LCLKIN Input

Termination: No integrated termination resistors are provided. A termination resistor of 100Ω should be placed near the input pin.

Connection of unused LCLKIN input: LCLKIN input has to be pulled up to VDD33 and disabled by configuring PIN_ENA_LVDS to zero.

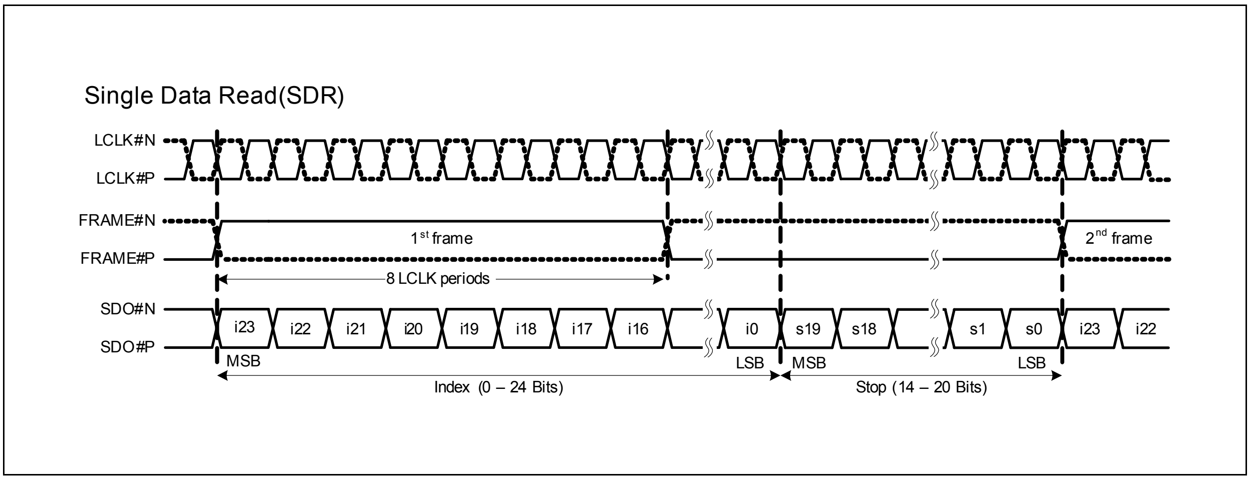
Figure 44:
LCLIN Input



LVDS Single Data Read Output Interface (SDR)

In single data read mode (LVDS_DOUBLE_DATA_RATE = 0) the data and frame bits are clocked on the falling edge of LVDS output clock LCLKOUT. The data bits are stable during the following rising edge of LCLKOUT.

Figure 45:
LVDS Outputs



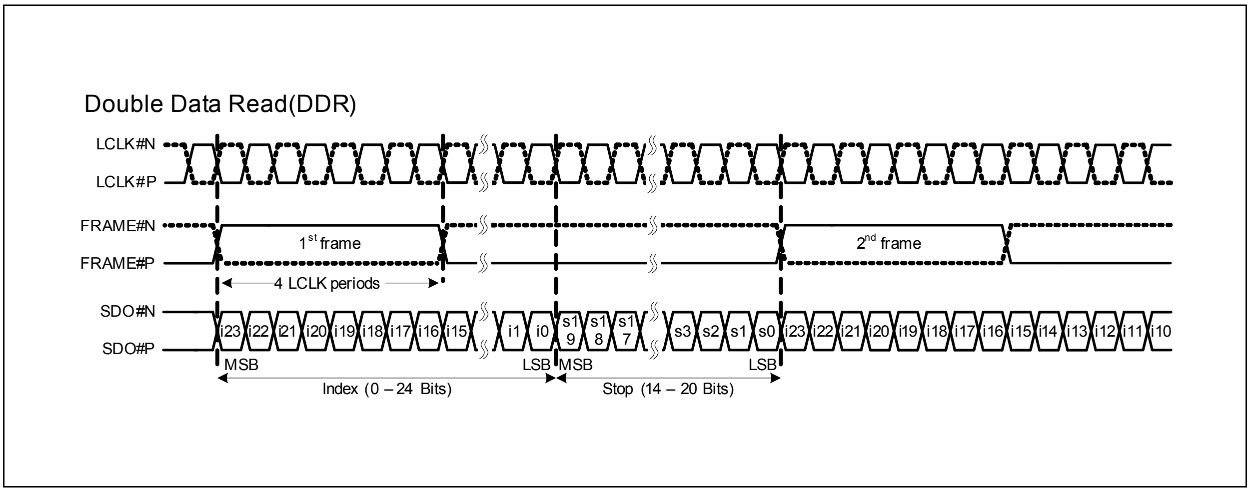
Note(s):

1. Bit width of the reference index and the stop result is configured by STOP_DATA_BITWIDTH and REF_INDEX_BITWIDTH

LVDS Double Data Read Output Interface (DDR)

With double data read mode the readout rate is doubled or alternatively the LVDS clock frequency can be halved with constant readout rate. The data and frame bits are clocked on rising and falling edges of LCLKOUT. Both bits, data and frame, are delayed by $t_{DV,LVDS}$ to LCLKOUT in order to grant sufficient hold time for the receiving device. With configuration parameter LVDS_DATA_VALID_ADJUST the delay can be adjusted for all LVDS outputs in common.

Figure 46:
LVDS Outputs



Note(s):

1. Bit width of the reference index and the stop result is configured by STOP_DATA_BITWIDTH and REF_INDEX_BITWIDTH

LVDS Output Test Pattern

Setting LVDS_TEST_PATTERN = 1 the interface continuously outputs the following fixed test patterns. All stop events are ignored.

Reference index = 111100000110011001010101010bin (=15781034dec)

Stop result = 00001010101011001111 0000bin (=699632dec)

Depending on the configuration of the output format width (REF_INDEX_BITWIDTH, STOP_DATA_BITWIDTH) only the corresponding lower bits of the reference index and the stop result are transmitted.

SPI Communication Interface

General

The SPI interface is implemented to

- Reset the chip to power on state
- Write configuration registers
- Verify configuration or status registers
- Initialize and restart measurements
- Byte-wise readout of results from the read registers (see Figure 31) via SPI instead via serial LVDS outputs

The serial interface is compatible with the 4-wire SPI standard in Motorola specification:

- Clock Phase Bit = 1
- Clock Polarity Bit = 0

Detailed Pin Description

Pin SSN
The 'Slave Select Not' line is the HIGH-active reset for the serial interface. When set to LOW, the interface is ready for serial shift of data into or out of the device. Each access POR, INIT, READ or WRITE has to start with a positive pulse on SSN.

Pin SCK
The 'Serial Clock' line is the driving clock which starts at LOW level and expects HIGH active pulses.

Pin MOSI
The 'Master Out Slave In' line is the serial data input of the device. Data takeover is done with the falling edge of SCK. The MSB is sent first.

Pin MISO
At 'Master In Slave Out' line, the serial data are clocked out of the chip with the rising edge of SCK. When SSN is set to HIGH, then the data output pin MISO is in high-Z state. The MSB is sent first.

Pin INTERRUPT
A low level at the interrupt pin indicates to the receiving device that data are available.

Pin PARITY
Monitoring the chip is possible by observing the PARITY pin. It indicates whether the sum of all configuration bits is even (Parity = 0) or odd (Parity = 1).

Communication Commands (Opcodes)

Figure 47:
Opcodes Overview

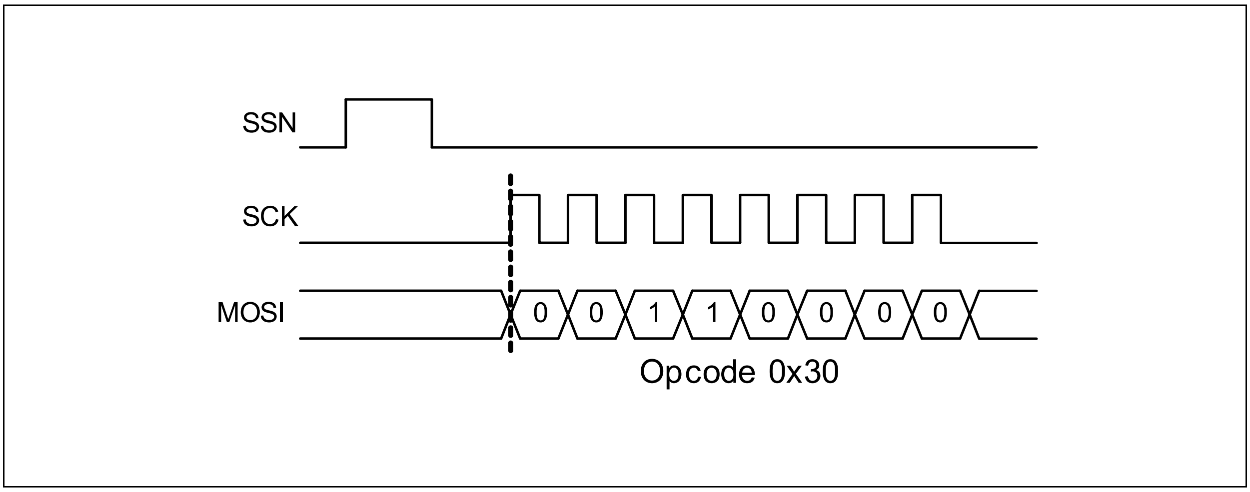
Opcode	HEX / BIN	Description
spiopc_power	0x30 = 0b00110000	Power on reset and stop measurement
spiopc_init	0x18 = 0b00011000	Initializes Chip and starts measurement
spiopc_write_config	0x80 = 0b100XXXXX	Write to configuration register X=0..17
spiopc_read_results	0x60 = 0b011XXXXX	Read opcode for result and status register X=8..31
spiopc_read_config	0x40 = 0b010XXXXX	Readout of configuration register X=0..17

Detailed Command Description

Power-ON Reset

After stabilization of all VDD33 and VDD18 the device expects the opcode spiopc_power = 0x30 to be sent via the SPI interface for power on reset. After the last bit of the opcode the reset remains active during $t_{HD,SSN}$ before the device is ready for the next read or write access. After the reset, the measurement is stopped and the configuration registers are set to internal defaults of the chip.

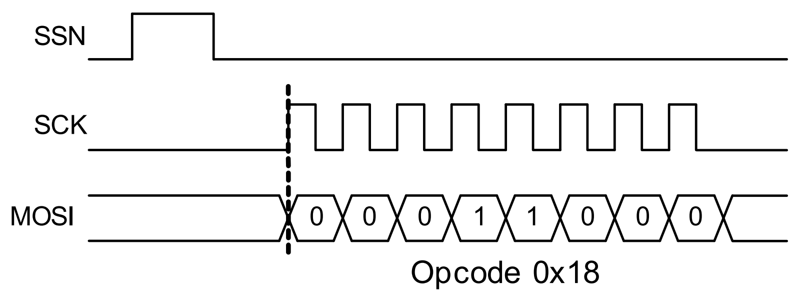
Figure 48:
Power-On Reset Opcode



Initialization Reset

After the configuration, the initialization opcode spiopc_init=0x18 resets again the chip to power on state, but preserves the configuration and starts the measurement. The initialization reset can be send while the reference clock or stops are applied. It takes 16 pulses of the reference clock before the stop channels are opened internally. After the initialization reset the delay t_{POR} has to be waited before next communication. The initialization reset can be applied also during measurements to restart the chip, but preserves measured data in FIFOs.

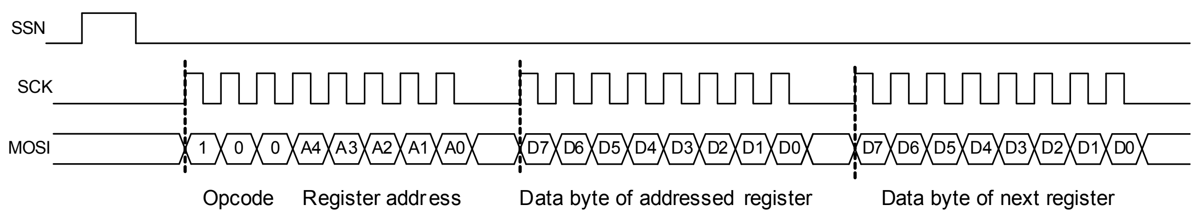
Figure 49:
Initialization Reset Opcode



Write / Incremental Write

Write access is permitted to the configuration registers exclusively. The access starts by sending the opcode spiopc_write_config = 0x80 after a positive SSN pulse. The register address is just added to spiopc_write_config. The data are sent after the opcode. Incremental write access to the successive registers is possible by sending the next data bytes. A complete configuration starts normally at register 0, followed by all register data bytes.

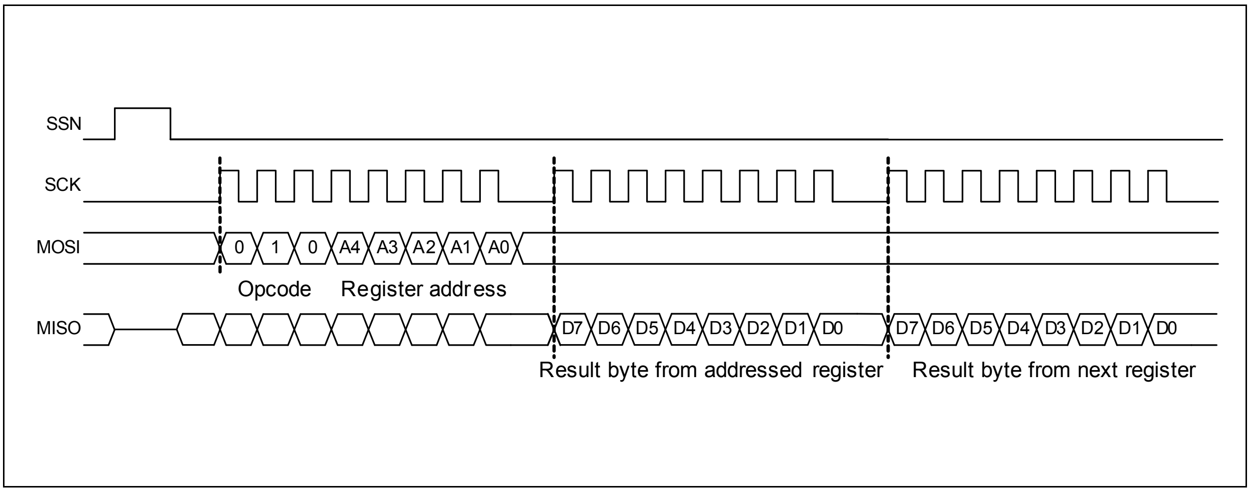
Figure 50:
SPI Incremental Write



Read / Incremental Read

The read access to registers starts by sending the opcodes spiopc_read_results =0x60 or spiopc_read_config = 0x40 after a positive SSN pulse. The register address is just added to the opcode. After the opcode the data are clocked out at the MISO line. Incremental read access to following registers is possible by continuously reading bytes. Each register is suitable as start address for incremental access.

Figure 51:
SPI Incremental Read



Using SPI Interface for Read-Out of Stop Results

Reading results byte-wise from TDC-GPX2 e.g. by an external microcontroller is fully supported. While using the SPI interface, data read by LVDS has to be suppressed by setting PIN_ENA_LVDS_OUT to zero or at least by not applying a clock at LCLKIN.

When reading an empty channel the results of REFINDEX and STOPRESULT are marked with 0xFFFFFF. Typically, the measurement rate of TDC-GPX2 is much higher than the readout rate possible with SPI. In this case using COMMON_FIFO_READ and BLOCKWISE_FIFO_READ is helpful to get sequential results which were measured in parallel in TDC-GPX2.

REF_INDEX_BITWIDTH and STOP_DATA_BITWIDTH are not relevant for reading via SPI.

Coding of Results

Configuration of LSB by REFCLK_DIVISIONS

The reference clock period is divided into subdivisions by REFCLK_DIVISIONS for the definition of the LSB of the stop results at the output interface. One subdivision corresponds to the LSB and the stop results are scaled into multiples of this LSB. In order to avoid quantization artefacts of the output interface, the resulting LSB has to be much smaller than the single shot resolution of GPX2. The most convenient way is choosing an LSB of 1ps by configuring REFCLK_DIVISIONS to the picosecond value of the reference clock period. Other LSB settings are possible as well, like LSB of 5ps or 10ps.

Figure 52:
LSB Configuration

Reference Clock Period	Reference Clock Frequency	REFCLK_DIVISIONS LSB = 1ps	REFCLK_DIVISIONS LSB = 5ps	REFCLK_DIVISIONS LSB = 10ps
500ns	2MHz	500000	100000	50000
250ns	4MHz	250000	50000	25000
200ns	5MHz	200000	40000	20000
100ns	10MHz	100000	20000	10000
50ns	20MHz	50000	10000	5000

Note(s):

1. For LVDS output, REFCLK_DIVISIONS must not exceed the result bit width defined by STOP_DATA_BITWIDTH

Examples for Codes of Time Measurements Results

Figure 53:
LSB Configuration

Readout of Stop Result		Resulting Stop Time with An Assumed LSB of			Note	
Hexadecimal	Decimal	LSB = 1ps	LSB = 5ps	LSB = 10ps		
0x0	0	0ps	0ps	0ps		
0x1	1	1ps	5ps	10ps		
0x2	2	2ps	10ps	20ps		
0xA	10	10ps	50ps	100ps		
0x64	100	100ps	500ps	1000ps		
0x3E8	1000	1000ps	5000ps	10000ps		
0x2710	10000	10000ps	50000ps	100000ps		
0x61A7	24999	24999ps	124995ps	249990ps	refclk-period t_{REFCLK} =250ns	
0xC34F	49999	49999ps	249995ps [2]	See note (1)		
0x3D08F	249999	249999ps [2]	See note (1)	See note (1)		
0x1869F	99999	99999ps	499995ps	See note (1)	refclk-period t_{REFCLK} =500ns	
0x30D3F	199999	199999ps	See note (1)	See note (1)		
0xF423F	999999	See note (1)	See note (1)	See note (1)		
0x3FFF	16383	16383ps	81915ps	163830ps	LVDS: Max readout with stop_data_ bitwidth=	14Bit
0xFFFF	65335	65335ps	326675ps	653350ps		16Bit
0x3FFFF	262143	262143ps	See note (1)	See note (1)		18Bit
0xFFFFF	1048575	1048575ps	See note (1)	See note (1)		20Bit
0x0FFFFF	1048575	1048575ps	See note (1)	See note (1)	SPI: Max readout with 20Bit [3]	

Note(s):
1. Time difference exceed GPX2 specification for reference clock period
2. REFCLK_DIVISIONS decreased by one is the highest possible readout value
3. With SPI read-out the four upper bits are unused

Maximum Time Differences

The following table shows the maximum possible time differences between stops, depending on the reference index bit width.

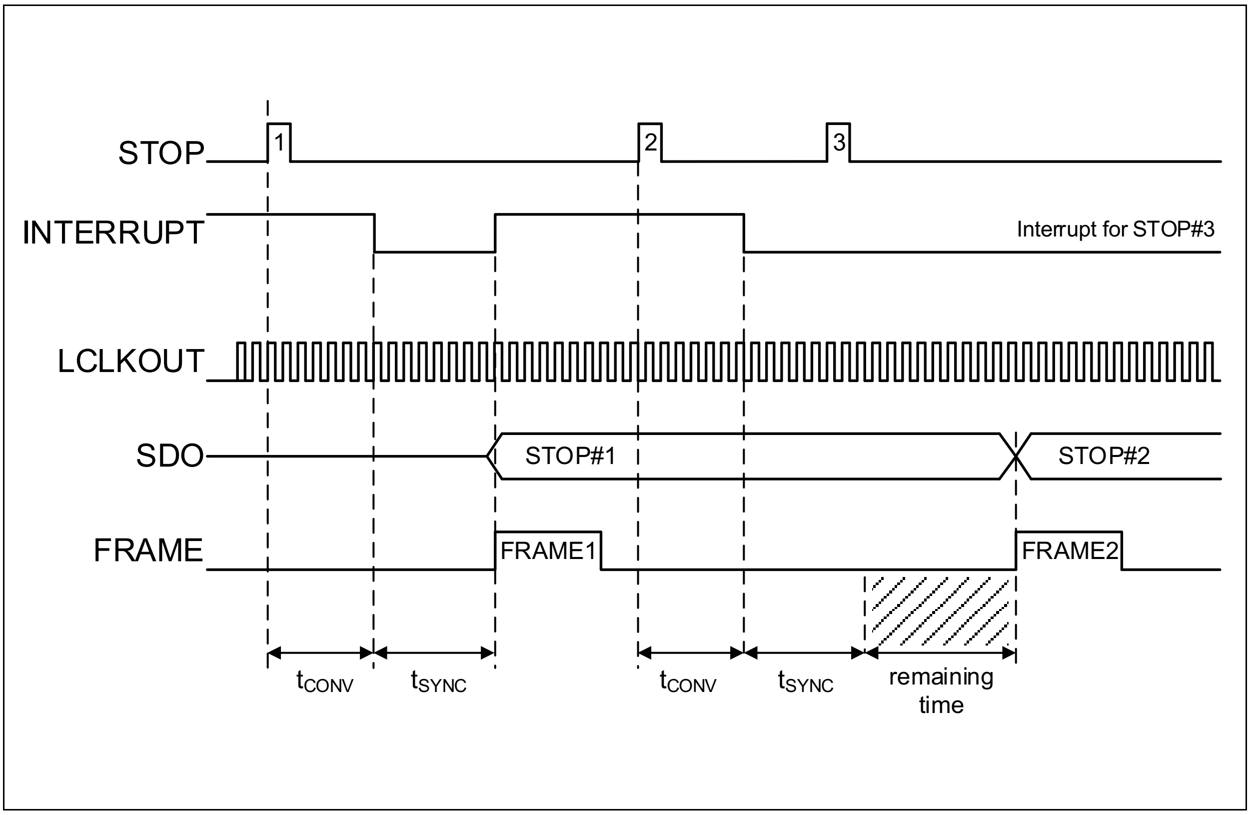
Figure 54:
LSB Configuration

REF_INDEX_ BITWIDTH	Mode	Maximum Readout Hexadecimal	Maximum Readout Decimal	Max Time Difference with Reference Clock		
				f_{REFCLK} = 2MHz	f_{REFCLK} = 5MHz	f_{REFCLK} = 10MHz
0Bit	LVDS/SPI	No read-out	No read-out	0.5µs	200ns	100ns
2Bit	LVDS	0x3	3	2µs	800ns	400ns
4Bit	LVDS	0xF	15	8µs	3.2µs	1.6µs
8Bit	LVDS/SPI	0xFF	255	128µs	51.2µs	25.6µs
16Bit	LVDS/SPI	0xFFFF	65335	32ms	13.0ms	6.5ms
24Bit	LVDS/SPI	0xFFFFFF	16777215	8s	3.2s	1.6s
6Bit	LVDS	0x3F	63	31µs	12.6µs	6.3µs
12Bit	LVDS	0xFFF	4095	2ms	800µs	400µs

Conversion Latency and Conversion Rate

The conversion latency t_{conv} is the time need when an event at a stop input pin occurs until it is processed and ready for output through the interface. With LVDS instead of SPI output an additional synchronization latency to the LCLK is applied.

Figure 55:
Conversion Latency



The conversion and synchronization latency is only applied to single events. During an output sequence of several events the conversion latency is processed in parallel during the remaining time.

Converter Latency

The conversion latency t_{CONV} is the time needed when an event at a stop input pin occurs until it is processed. Once a stop event is recognized, it has to be converted into the results of TSTOP and REFID. The basic conversion latency t_{CONV} is the same for SPI or LVDS readout. After the conversion latency has passed, the INTERRUPT pin is set to zero (if not already zero from a previous stop) and the stop result is ready for readout via the SPI interface. The conversion latency depends also on the HIGH_RESOLUTION configuration.

LVDS Synchronization Latency

For both LVDS output modes, DDR+SDR, an additional synchronization latency t_{SYNC} has to be processed before the output sequence starts. With LVDS reading an additional latency t_{SYNC} for synchronization to the LCLK is applied. t_{SYNC} is counted in LVDS clock cycles and the output is indicated by setting the frame output pin.

Conversion Rate

Conversion rate is the rate where stop events can be measured. It is determined or limited by the peak input conversion rate or the read-out rate. The conversion rate of the stop events at the input can be higher or also lower than the read-out rate output interface. In any case, the FIFO will adapt a variable peak stop event rate and to the read-out rate.

Peak Conversion Rate

The peak input conversion rate is limited by the ability of GPX2 to sample, convert and store stop events in the FIFOs. The maximum peak conversion rate is limited minimal pulse-to-pulse-spacing t_{PPS} of the chosen measuring mode. The number of conversions at peak conversion rate is given by the FIFO depth and to a certain extent by the read out rate of the interface.

Read-Out Rate

The maximum read-out rate is reached when the output interface (either SPI or LVDS) is continuously in use for outputting the measurement results. The configured code length (LVDS: STOP_DATA_BITWIDTH and REF_INDEX_ BITWIDTH, SPI: readout bytes) and the frequency define the readout capabilities.

Average Conversion Rate

The average conversion rate is determined either by the

- **Peak Input Conversion Rate:** If the read-out rate is higher than peak input conversion rate no time event is getting lost because of a full FIFO. This is typically the case when reading out with LVDS.
- **Read-Out Rate:** If read-out rate is always slower than the input conversion rate then time measurements necessarily are getting lost because the FIFO may be full. This is typically the case when reading out via SPI. In this case the configuration of BLOCKWISE_FIFO_READ and COMMON_FIFO_READ is an option even to get measured a sequence of successive stops

Examples for Read-Out Rate with LVDS

The conversion rate of measured stop events can be calculated by dividing the bus frequency by the number of bits, which are readout reference index and stop result. The number of bits is configured by STOP_DATA_BITWIDTH and REF_INDEX_BITWIDTH.

Figure 56:
Example Data Average Conversion Rate

STOP_DATA_ BITWIDTH	REF_INDEX_ BITWIDTH	Sum of Bits	LCLK	SDR Throughput Rate	DDR Throughput Rate
00 (14Bit)	000 (0Bit)	14	300MHz	21MSPS	42MSPS
00 (14Bit)	010 (4Bit)	18	300MHz	16MSPS	32MSPS
01 (16Bit)	000 (0Bit)	16	300MHz	18MSPS	37MSPS
01 (16Bit)	011 (8Bit)	24	300MHz	12MSPS	25MSPS
10 (18Bit)	000 (0Bit)	18	200MHz	11MSPS	22MSPS
10 (18Bit)	100 (16Bit)	32	200MHz	6MSPS	12MSPS
11 (20Bit)	000 (0Bit)	20	100MHz	5MSPS	10MSPS
11 (20Bit)	101 (24Bit)	44	100MHz	2MSPS	4MSPS

Note(s):

1. Maximal throughput rate is only reached when the stop event rate at input is high enough
2. With CHANNEL_COMBINE = 1 ("Pulse Distance") the throughput rate per stop input pin is doubled, as the stop events of one input pin are alternatively measured and readout by two channels.

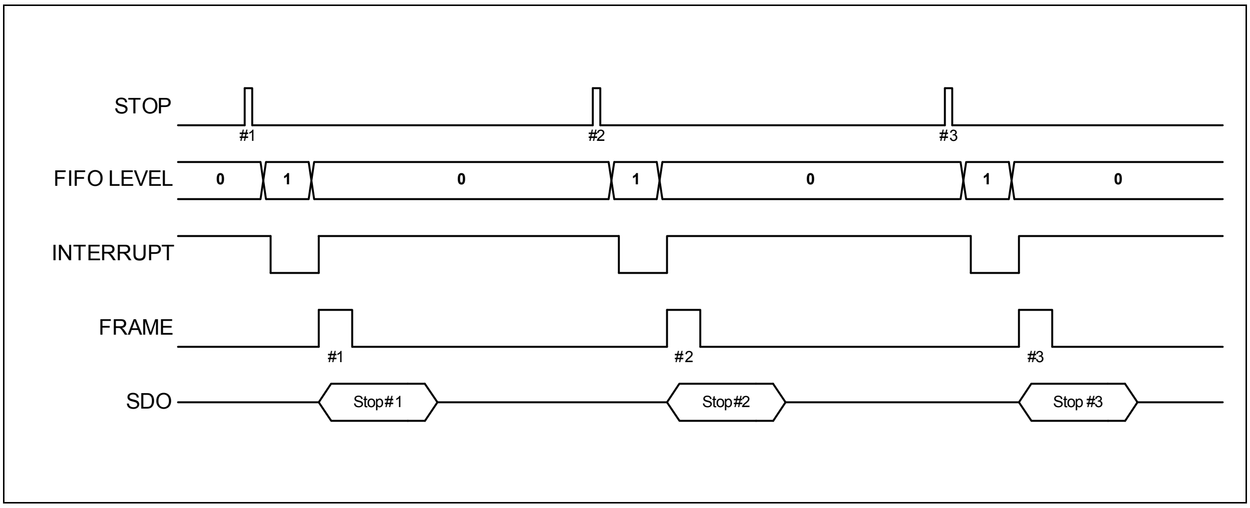
FIFOs for Adapting Peak and Average Conversion Rate

Each channel of GPX2 has a First-In-First-Out data buffer (FIFO). Generally, GPX2 is capable of measuring the incoming stops faster than the length of an output sequence. The FIFO is capable of storing up to data of 16 stop events until the data are read out. Up to a certain degree, the FIFO prevents rejection of stop events for a short time when the input stop event rate is higher than the read-out rate. But when the input data rate is constantly higher than the read-out rate, then the FIFO gets full and stop events are rejected. After a full FIFO was read out and empty space is available for stop measurement further two stops are needed to restart the FIFO ($t_{FIFO\_RESTART}$).

The maximum FIFO depth is 16, 8 or 4 stages, depending on the HIGH_RESOLUTION configuration (off, 2x, 4x).

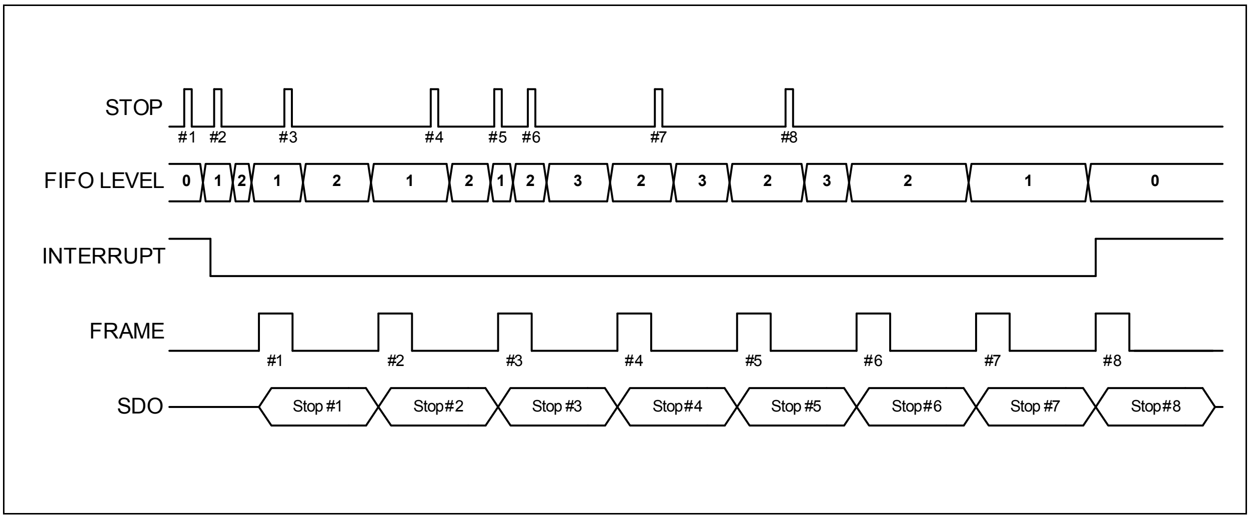
The following figures illustrate the typical dependencies between stop event rate and the read out rate. They are applicable for both SPI and LVDS readout. The INTERRUPT pin indicates that the result is available for read-out through the SPI interface. For SPI a continuous readout is assumed as long as the interrupt is on low level. For LVDS output the FRAME indicates the beginning of data output at SDO line. The interrupt goes back to HIGH when all FIFOs are empty even if output is LVDS. In the figures FIFO_DEPTH = 4 is assumed. The FIFO LEVEL indicates the stop event buffered in the FIFO. A stop event will increase FIFO LEVEL by one, reading out will decrease the FIFO LEVEL.

Figure 57:
Input Stop Event Rate is Lower than the Readout Rate



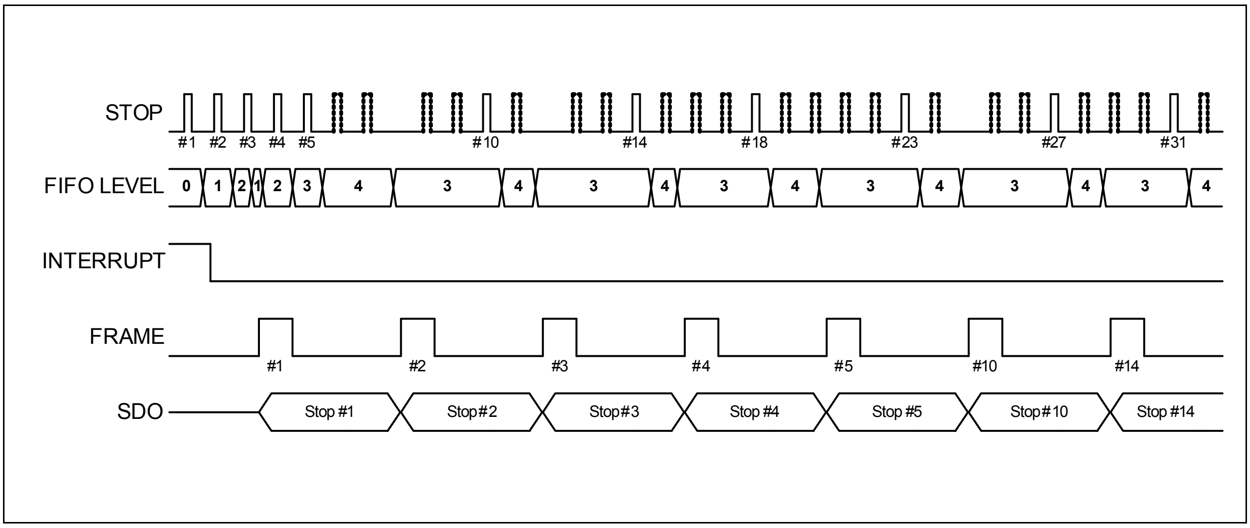
- Enough time for complete readout of first stop before the next stop event arises
- Interrupt goes back to high because the FIFO is empty after read-out
- In this example, no stop events are rejected. All stops are measured and read out

Figure 58:
Average Stop Event Rate is Lower, but Peak Stop Event Rate is Higher than the Readout Rate



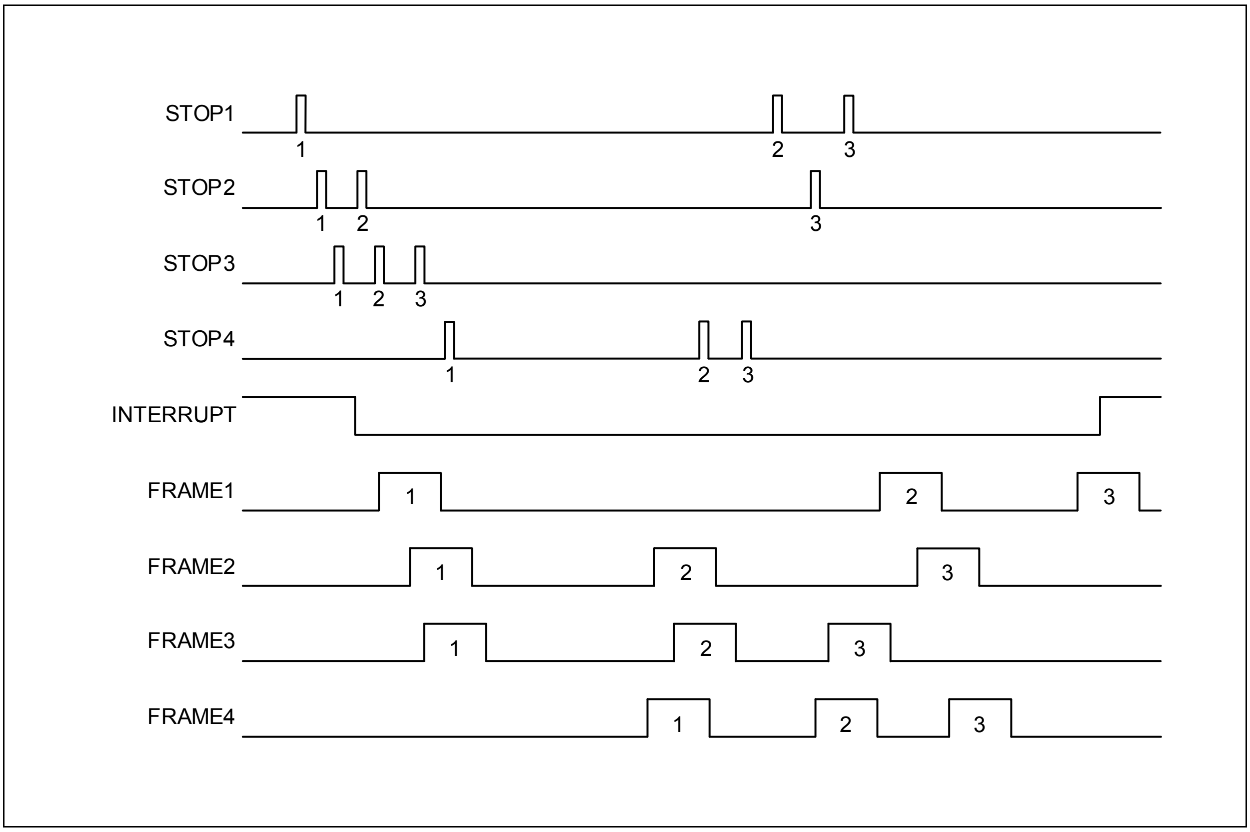
- Stop events during read-out are stored in FIFO
- Stop events buffer up to FIFO LEVEL 3
- In this example, no stop events are rejected. All stops are measured and read out.
- Interrupt goes back to high when all data are readout and the FIFO is empty.
- Maximal FIFO_DEPTH and HIGH_RESOLUTION limits the peak event storage

Figure 59:
Stop Event Rate is Higher than the Readout Rate



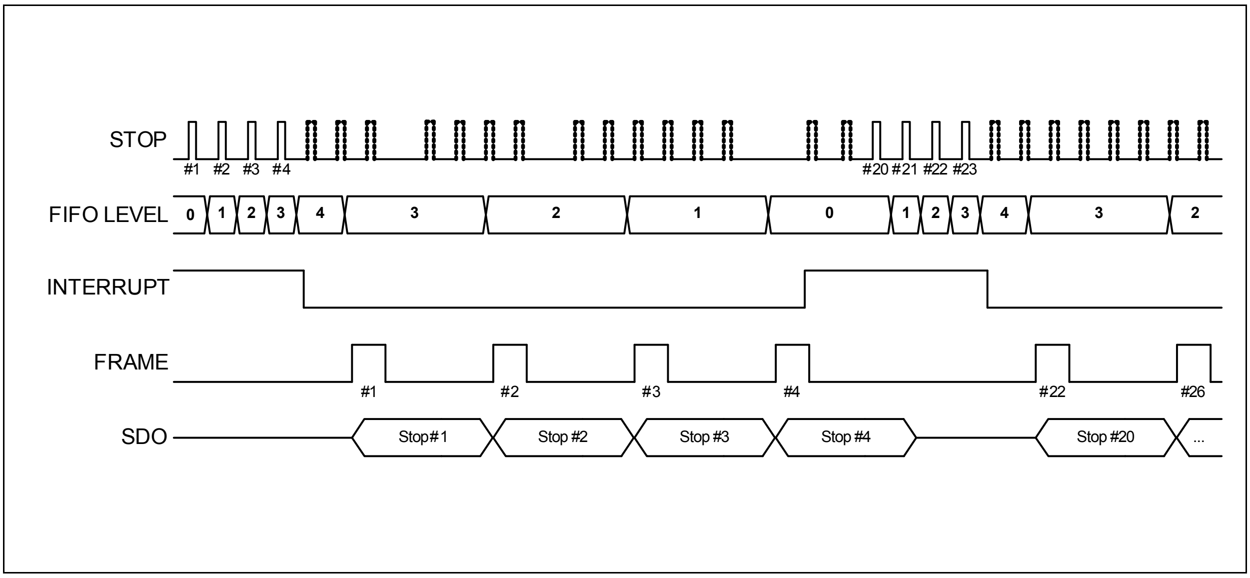
- During read-out stop events (dots) are ignored when FIFO full at FIFO LEVEL 4.
- After reading a result from a full FIFO the next two stops events (dashed) are still ignored but used to restart the FIFO
- Interrupt is always zero because the FIFO never gets empty.

Figure 60:
Stops on All Four Channels



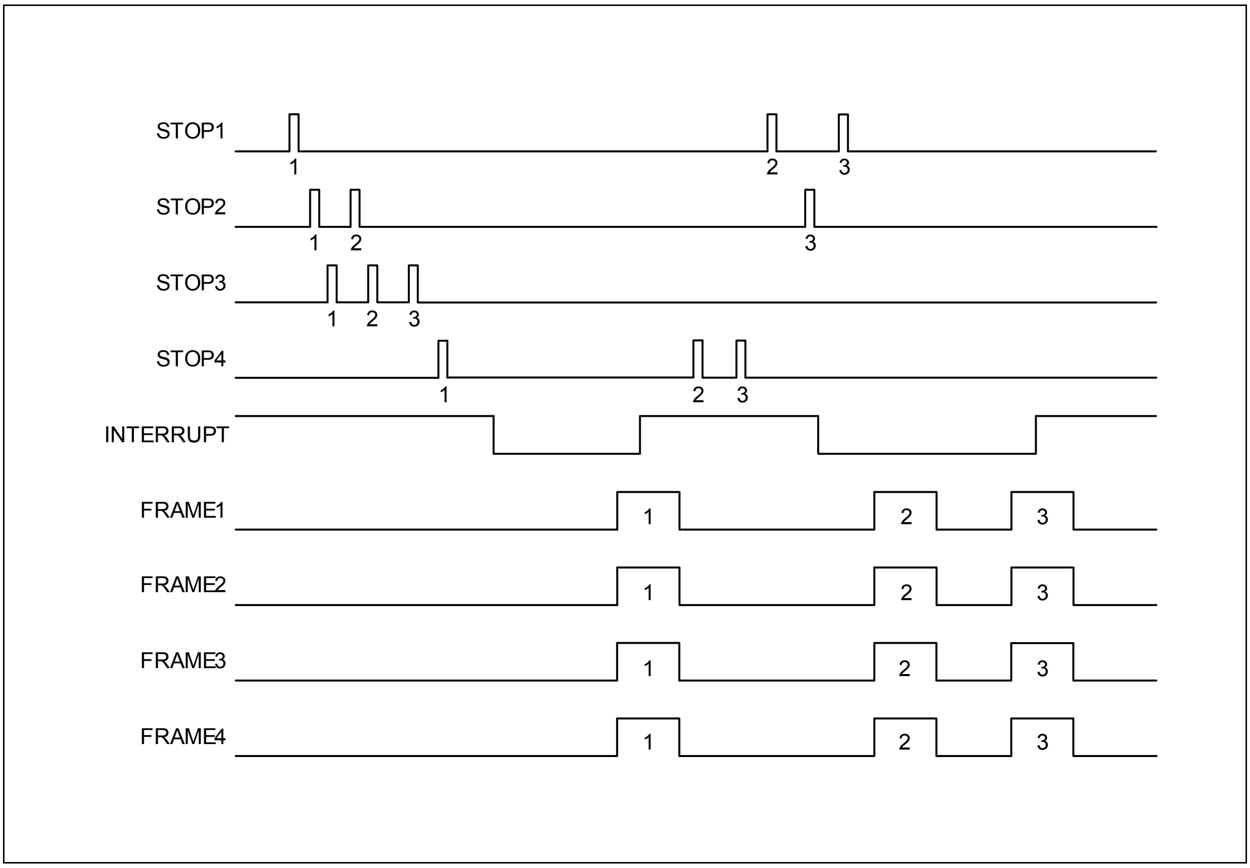
- All four channels are completely independent from each other (COMMON_FIFO_READ=0)
- In this example no stop events are rejected, because FIFOs never get full
- Interrupt remains zero as long as at least one FIFO has a valid data, interrupt gets high when all FIFO are empty

Figure 61:
BLOCKWISE_FIFO_READ



- A block of successive stop events are measured in a block before readout
- Readout of FIFO starts not before the FIFO is full.
- During read-out stop events (dots) are ignored when FIFO full at FIFO level 4…1.
- After reading all result from the FIFO the next two stops events (dashed) are still ignored but used to restart the FIFO
- Measurement starts not before the FIFO is empty.
- COMMON_FIFO_READ is applicable.

Figure 62:
COMMON_FIFO_READ



- All active FRAME pins are set simultaneously, as soon as all active FIFOs have value (COMMON_FIFO_READ = 1)
- As long as one FIFO has no valid data, no readout is done
- Interrupt doesn't fall to low before all active FIFOs have valid data
- In this example no stop events are rejected, because FIFOs never get full.
- BLOCKWISE_FIFO_READ is fully applicable
- SPI readout only successively of all active FIFOs (1, 2, 3, 4 …). It is not permitted to read one channel twice (e.g 1 & 1, 2 & 2 …)

Application Information

Configuration Examples

Typical Configuration for LVDS

org ROM_ADD_CFG; config_default.cfg saved on 19.09.2016 11:58

equal 0x401F0131; Register 3, 2, 1, 0

equal 0x53C0030D; Register 7, 6, 5, 4

equal 0x0A0013A1; Register 11, 10, 9, 8

equal 0x7DF1CCCC; Register 15, 14, 13, 12

equal 0x00000004; Register 19, 18, 17, 16

equal 0x00000000; Register 23, 22, 21, 20

Example C++ Code

The following C++ code is provided to give an overview about how to organize the initial steps of a microprocessor, to be able to conduct a typical time measurement task with GPX2.

```
#include <uProcessor.h>              // This is an imaginary header file
                                     // defined to support this example code
// ------------------------------------------------------------------------------------------
// *** uProcessor.h ***
// ------------------------------------------------------------------------------------------
// Almost every microprocessor has a specific C++ libraries (header files) that introduce
// specific commands for data readout.
// Therefore, this imaginary header data is given to support this example code.
// The intention of each virtual function on this header is clearly explained as follows.
// In real projects, instead of these functions,
// the user should use the similar functions of the micro-processor which is used with GPX2.
//
// Virtual functions:
//    send_byte_to_SPI( Var1 );   : send Var1 (8 Bits) through the SPI
//
//    read_byte_from_SPI( Var1 ); : read 1 Byte data from SPI and write it to Var1
//
// Virtual pin variables:
//    GPIO_SSN :              Variable (1 Bit) to control the output pin which is
//                            supposed to be connected the SSN pin of the GPX2
//
//    GPIO_INTERRUPT :        Variable (1 Bit) to monitor the input pin which is
//                            supposed to be connected INTERRUPT pin of the GPX2
// ------------------------------------------------------------------------------------------
// *** Configuration Registers ***
// ------------------------------------------------------------------------------------------
const char config_register[16] = { 0x31, 0x01, 0x1F, 0x40, 0x0D, 0x03, 0xC0, 0x53,
                                   0xA1, 0x13, 0x00, 0x0A, 0xCC, 0xCC, 0x31, 0x8E, 0x04 };
// A typical config settings  = { config00, config01, … , config16 }
```

```
// ------------------------------------------------------------------------------------------------
// *** SPI Opcodes ***
// ------------------------------------------------------------------------------------------------
const char spiopc_power          = 0x30;     // opcode for "Power on Reset"
const char spiopc_init           = 0x18;     // opcode for "Initialize Chip and Start Measurement"
const char spiopc_write_config  = 0x80;     // opcode for "Write Configuration"
const char spiopc_read_config   = 0x40;     // opcode for "Read Configuration"
const char spiopc_read_results  = 0x60;     // opcode for "Read Measurement Results"

// ------------------------------------------------------------------------------------------------
// *** SPI Addresses ***
// ------------------------------------------------------------------------------------------------
const char reference_index_ch1_byte3 = 8;    //
const char reference_index_ch1_byte2 = 9;
const char reference_index_ch1_byte1 = 10;
const char stopresult_ch1_byte3 = 11;
const char stopresult_ch1_byte2 = 12;
const char stopresult_ch1_byte1 = 13;
//  . . . .
const char stopresult_ch4_byte3 = 29;
const char stopresult_ch4_byte2 = 30;
const char stopresult_ch4_byte1 = 31;

// ------------------------------------------------------------------------------------------------
// *** Other Variables ***
// ------------------------------------------------------------------------------------------------
int  Buffer                = 0;            // buffer variable used to copy the SPI data
char i                     = 0;            // counter for for-loops
int  reference_index[4]    = 0;            // reference index data array {Ch1, Ch2, Ch3, Ch4}
int  stopresult[4]         = 0;            // stop result data array {Ch1, Ch2, Ch3, Ch4}
bool config_error          = false;        // flag that indicates if the config registers
                                           // are not written correctly

// ------------------------------------------------------------------------------------------------
// *** Main body of the software ***
// ------------------------------------------------------------------------------------------------
int main(void)
{
  // ----------------------------------------------------------------------------------------------
  // *** Power on reset ***
  // ----------------------------------------------------------------------------------------------
  GPIO_SSN = 1;                            // Reset the SPI interface and select the slave device
   GPIO_SSN = 0;

   send_byte_to_SPI( spiopc_power );       // Opcode for "Power On Reset" is sent over SPI
```

```
// -------------------------------------------------------------------------------------------
// *** Writing the configuration registers ***
// -------------------------------------------------------------------------------------------
GPIO_SSN = 1;                                   // Reset the SPI interface and select the slave device
GPIO_SSN = 0;

config_error = false;

send_byte_to_SPI( spiopc_write_config + 00 );  // Opcode for "Write Configuration"
                        // and config address (00) are sent over SPI

for ( i = 0; i < 17; i++)                       // Send all 17 config registers via SPI
    send_byte_to_SPI( config_register[i] );

// -------------------------------------------------------------------------------------------
// *** Verification of config registers ***
// -------------------------------------------------------------------------------------------
GPIO_SSN = 1;                                   // Reset the SPI interface and select the slave device
GPIO_SSN = 0;

send_byte_to_SPI( spiopc_read_config + 00 );   // Opcode for "Read Configuration"
                                                // and config address (00) are sent over SPI
for ( i = 0; i < 17; i++)
{
   read_byte_from_SPI( Buffer );                // read one byte from SPI to Buffer variable

   if ( config_register[i] != Buffer ) config_error = true;
                                                // if there was a failure in writing the config
                                                // registers, then the config_error flag is raised.
}                                               //

// -------------------------------------------------------------------------------------------
// *** Initialize and start the measurement ***
// -------------------------------------------------------------------------------------------
if (config_error == false )
{
   GPIO_SSN = 1;                                // Reset the SPI interface and select the slave device
   GPIO_SSN = 0;

   send_byte_to_SPI( spiopc_init );             // Opcode for "Initialize" is sent over SPI
                                                // This is required to start measuring process

   // ***************************************************************************************
   // End of the configuration settings. After now the time measurement will start.
   // This code is designed to use SPI to read the measurement data from GPX2.
   // Using LVDS as a output interface requires additional hardware and code.
   // ***************************************************************************************
```

```
// ------------------------------------------------------------------------------------------
// *** Readout of measurement data via SPI ***
// ------------------------------------------------------------------------------------------
   while( GPIO_INTERRUPT != 0 );                 // wait till the Interrupt pin is low

   GPIO_SSN = 1;                                 // Reset the SPI interface and select the slave device
   GPIO_SSN = 0;

   send_byte_to_SPI( spiopc_read_results + reference_index_ch1_byte3 );
                                                 // Opcode for "Read Result" and data address are sent

   for ( i = 0; i < 4; i++)
   {
     read_byte_from_SPI( Buffer );               // read one byte from SPI to Buffer
     reference_index[i] = reference_index[i]     // Data is shifted 16 Bits to the left
                + ( Buffer << 16 );              // and added to the reference_index

     read_byte_from_SPI( Buffer );               // read one byte from SPI to Buffer
     reference_index[i] = reference_index[i]     // Data is shifted 8 Bits to the left
                + ( Buffer << 8 );               // and added to the reference_index

     read_byte_from_SPI( Buffer );               // read one byte from SPI to Buffer
     reference_index[i] = reference_index[i]     // Data is directly added to reference_index
                + Buffer;
                                                 // The complete reference index (3 Bytes)
                                                 // has been received.

     read_byte_from_SPI( Buffer );               // Same process as reference_index
     stopresult[i] = stopresult[i]               // is repeated for stop results
              + ( Buffer << 16 );

     read_byte_from_SPI( Buffer );
     stopresult[i] = stopresult[i] + ( Buffer << 8 );

     read_byte_from_SPI( Buffer );
     stopresult[i] = stopresult[i] + Buffer;
                                                 // The complete stopresult (3 Bytes)
                                                 // has been received
   }
   // In this point the software has obtained
   // the reference_index and stopresult data for all channels,
   // the rest of the codes should be designed depending on the user’s application.
   // . . .
  }
   // . . .
}
```

Schematic

The following figures show a typical circuits with power supply and line termination.

Figure 63:
Schematics for LVDS Inputs and Outputs

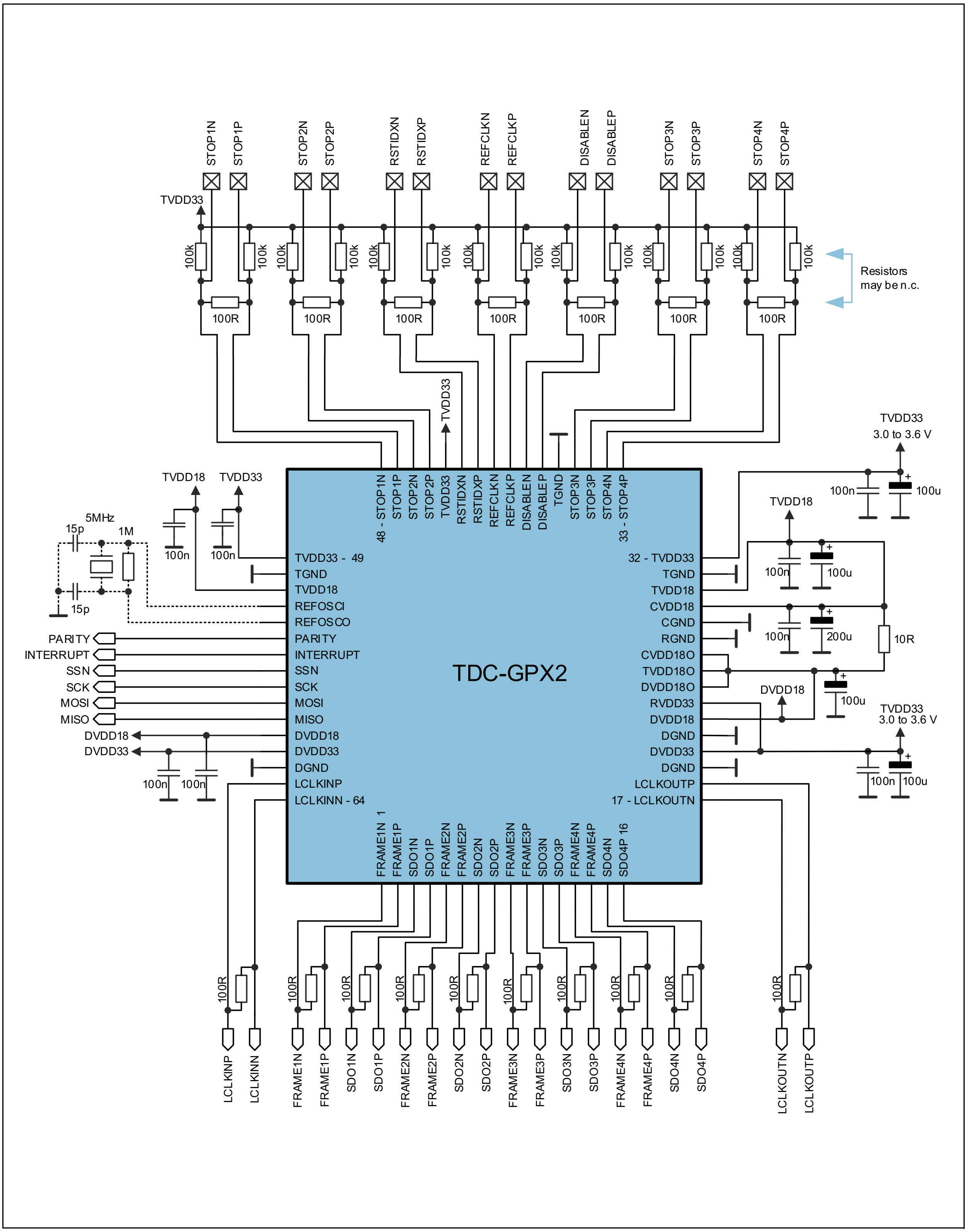
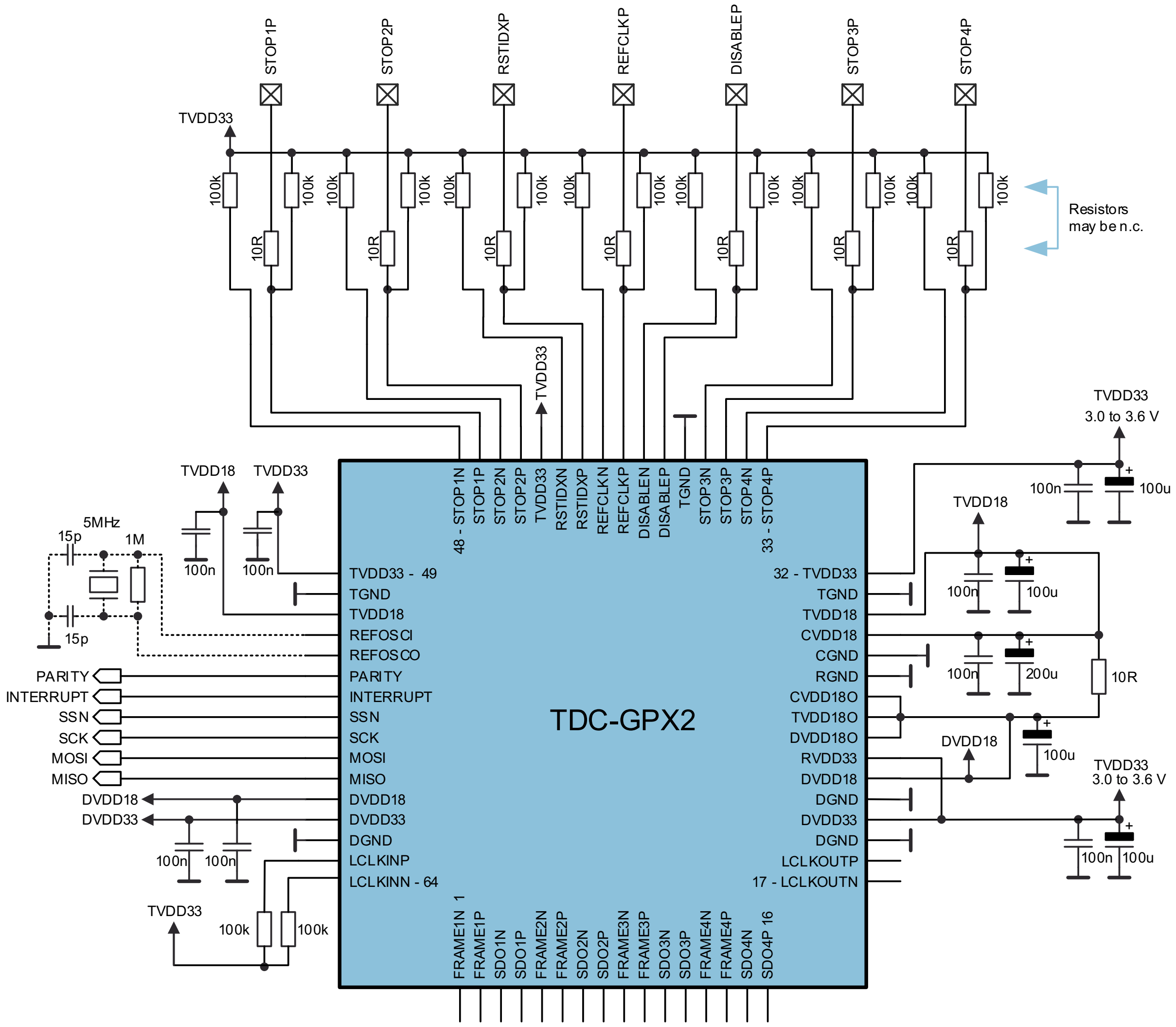


Figure 64:
Schematics for CMOS Inputs and SPI Communication



External Components

Supply Decoupling: GPX2 provides 6 power supply domains. Careful buffering is recommended. Small decoupling capacitors (e.b 100nF) with minimal ESL and ESR help to filter external power supply noise when placed near to the power supply pins.

The optimum number of decoupling capacitors depends on the actual application.

It is recommended to use separate supplies for time-analog (TVDD33) and digital (DVDD33, RVDD33) supply pins to isolate digital switching noise from sensitive circuitry. In case only a single (digital) supply is available, it should be routed to DVDD33 and RVDD33. It can then be tapped and isolated with a resistor (10Ω) to TVDD33. Grounding: A single ground plane is sufficient to give optimum performance, provided the analog, digital and clock sections of the board are cleanly partitioned. Refer to the GPX2 Evaluation PCB for an example on board layout schemes.

Signal lines: Even though LVDS signalizing on input and output reduces ground bounding during its transition, the positive and negative signal path has to be well matched and their trace should be kept as short as possible. Time-analog signal path like quartz oscillator or single ended (CMOS) stop inputs must be treated as a transmission line and should have a solid ground return path with a small loop. A serial resistor (10Ω) in single ended (CMOS) signal lines further help to damp reflections.

PCB Layout

Please refer to our GPX2-EVA-KIT

Package Drawings & Markings

The TDC-GPX2 comes in QFN64 or QFP64 package.

Figure 65:
QFN64 Package Drawings

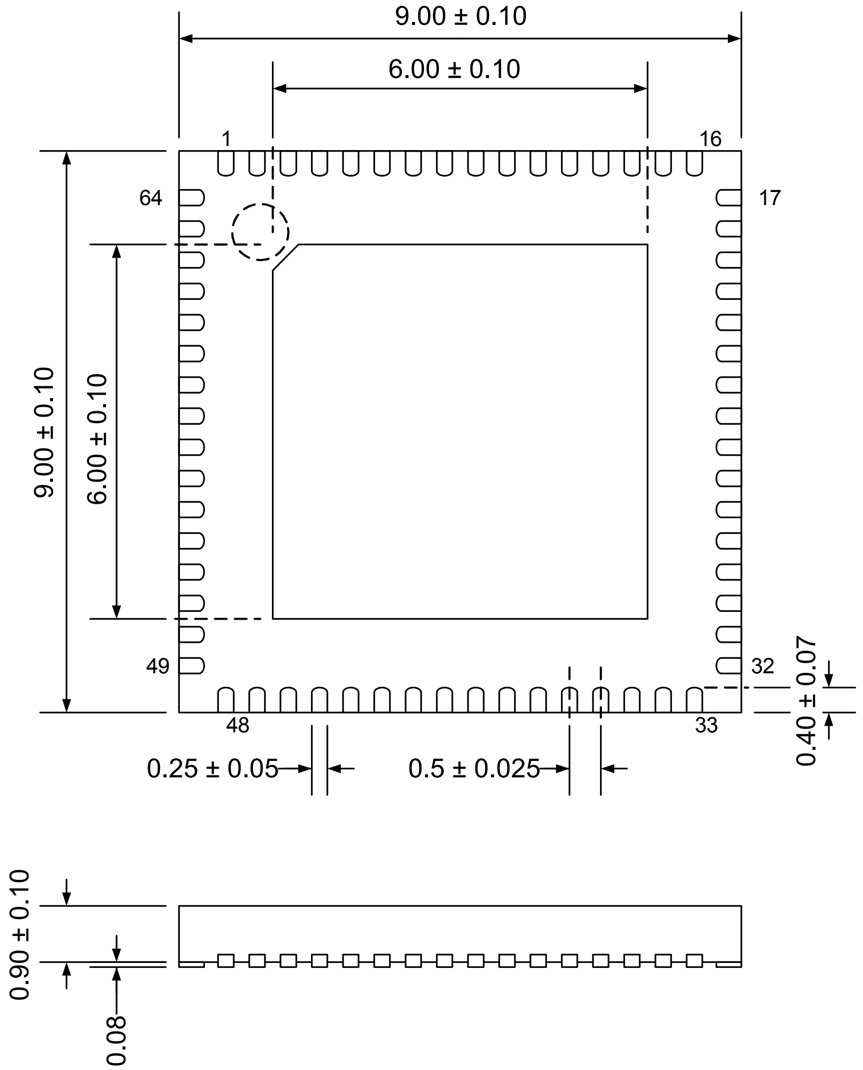


Figure 66:
QFN64 Package Marking

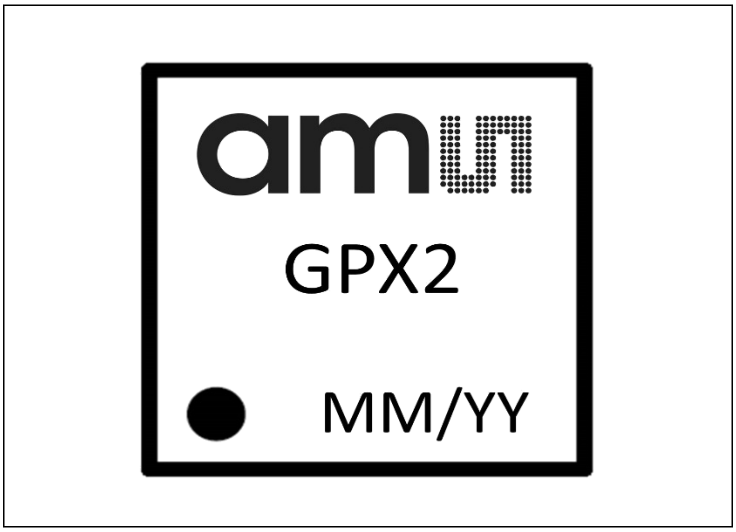
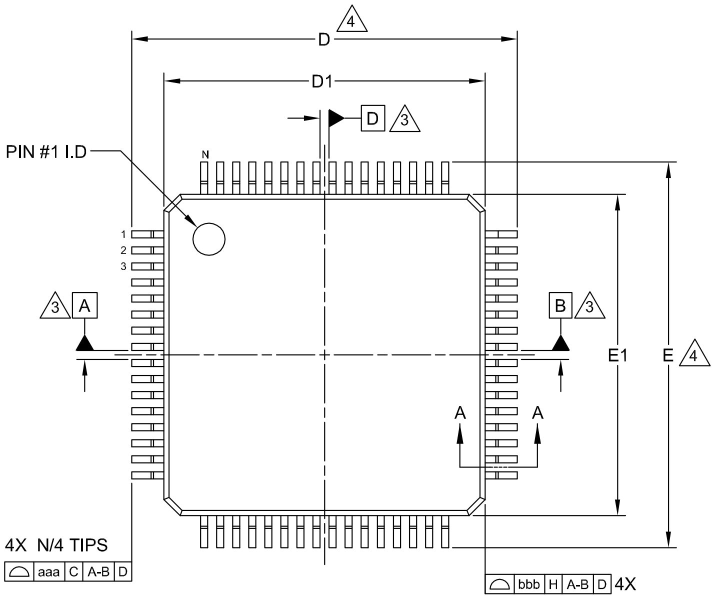
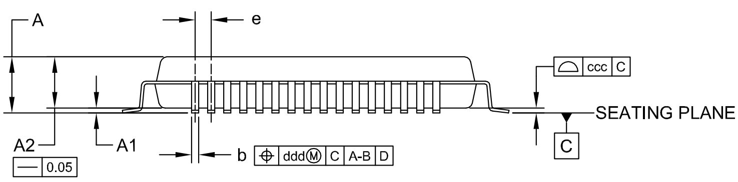


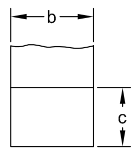
Figure 67:
QFN64 Package Code

MM	YY
Manufacturing month	Last two digits of the manufacturing year

Figure 68:
QFP64 Package Drawings

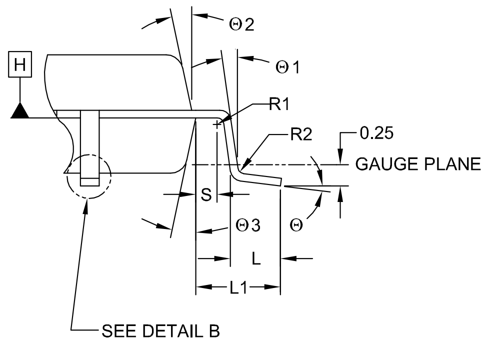






DETAIL B

REF.	MIN	NOM	MAX
A	-	-	1.60
A1	0.05	-	0.15
A2	1.35	1.40	1.45
b	0.17	0.22	0.27
c	0.09	-	0.20
D		12.00 BSC	
D1		10.00 BSC	
E		12.00 BSC	
E1		10.00 BSC	
e		0.50 BSC	
L	0.45	0.60	0.75
L1		1.00 REF	
R1	0.08	-	-
R2	0.08	-	0.20
S	0.20	-	-
Θ	0°	3.5°	7°
Θ1	0°	-	-
Θ2	11°	12°	13°
Θ3	11°	12°	13°
aaa	-	0.20	-
bbb	-	0.20	-
ccc	-	0.08	-
ddd	-	0.08	-
N		64	



SECTION A-A

Note(s):

1. Dimensioning and tolerancing conform to ASME Y14.5M-1994.
2. All dimensions are in millimeters (angles are in degrees).
3. Datums A-B and D to be determined at datum plane H.
4. To be determined at seating datum plane C.

Figure 69:
QFP64 Package Marking

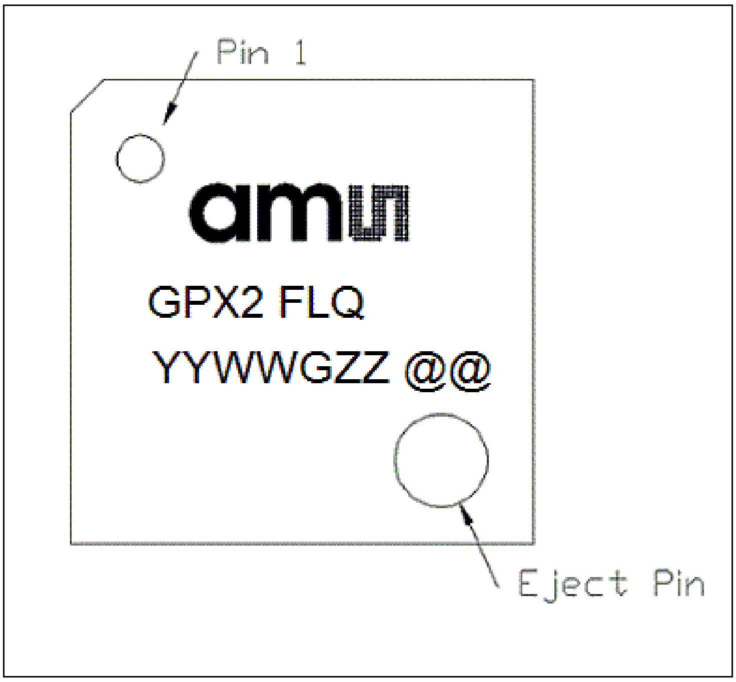


Figure 70:
QFP64 Package Code

YY	WW	G	ZZ	@@
Year	Manufacturing week	Assembly	Free choice	Sublot ID

Mechanical Data

QFN64

The QFN64 package has 9mm x 9mm outline. The solder pitch is 0.5mm. Package dimensions do not include mold flash, protrusions, burrs or metal smearing. All dimensions are given in millimeters.

QFN64 Tray Information

JEDEC NHBG09091.510266 Rev. A

10x 26 = 260 pieces

QFP64

The QFP64 package has 10mm x 10mm body size, with pins 12mm x 12mm outline. The solder pitch is 0.5mm. Package dimensions do not include mold flash, protrusions, burrs or metal smearing. All dimensions are given in millimeters.

QFP64 Tape & Reel Information

1 reel = 250 pcs

Soldering & Storage Information

Center-pad can be connected to ground or left open. Through-connections (vias) in the area between the center-pad and the pins should be avoided.

Ordering & Contact Information

Figure 71:
Ordering Information

Ordering Code	Package	Marking	Delivery Form	Delivery Quantity
TDC-GPX2 TRA	QFN64	GPX2	Tray	260 pcs/tray
TDC-GPX2 FLQM	QFP64	GPX2	Tape & Reel	250 pcs/reel

Technical Support is available at:
www.ams.com/Technical-Support

Provide feedback about this document at:
www.ams.com/Document-Feedback

For further information and requests, e-mail us at:
ams_sales@ams.com

For sales offices, distributors and representatives, please visit:
www.ams.com/contact

Headquarters
ams AG
Tobelbader Strasse 30
8141 Premstaetten
Austria, Europe

Tel: +43 (0) 3136 500 0

Website: www.ams.com

RoHS Compliant & ams Green Statement

RoHS: The term RoHS compliant means that ams AG products fully comply with current RoHS directives. Our semiconductor products do not contain any chemicals for all 6 substance categories, including the requirement that lead not exceed 0.1% by weight in homogeneous materials. Where designed to be soldered at high temperatures, RoHS compliant products are suitable for use in specified lead-free processes.

ams Green (RoHS compliant and no Sb/Br): ams Green defines that in addition to RoHS compliance, our products are free of Bromine (Br) and Antimony (Sb) based flame retardants (Br or Sb do not exceed 0.1% by weight in homogeneous material).

Important Information: The information provided in this statement represents ams AG knowledge and belief as of the date that it is provided. ams AG bases its knowledge and belief on information provided by third parties, and makes no representation or warranty as to the accuracy of such information. Efforts are underway to better integrate information from third parties. ams AG has taken and continues to take reasonable steps to provide representative and accurate information but may not have conducted destructive testing or chemical analysis on incoming materials and chemicals. ams AG and ams AG suppliers consider certain information to be proprietary, and thus CAS numbers and other limited information may not be available for release.

Copyrights & Disclaimer

Document Status

Document Status	Product Status	Definition
Product Preview	Pre-Development	Information in this datasheet is based on product ideas in the planning phase of development. All specifications are design goals without any warranty and are subject to change without notice
Preliminary Datasheet	Pre-Production	Information in this datasheet is based on products in the design, validation or qualification phase of development. The performance and parameters shown in this document are preliminary without any warranty and are subject to change without notice
Datasheet	Production	Information in this datasheet is based on products in ramp-up to full production or full production which conform to specifications in accordance with the terms of ams AG standard warranty as given in the General Terms of Trade
Datasheet (discontinued)	Discontinued	Information in this datasheet is based on products which conform to specifications in accordance with the terms of ams AG standard warranty as given in the General Terms of Trade, but these products have been superseded and should not be used for new designs

Revision Information

Changes from 1-02 (2017-Nov-07) to current revision 1-03 (2017-Dec-18)	Page
Updated Figure 2	2
Updated text under Pin Assignments	4
Updated note under Figure 5	5
Updated text under Package Drawings & Markings	65
Updated title of Figure 65	65
Updated title of Figures 66 & 67	66
Added Figure 68	67
Added Figures 69 & 70	68
Updated Mechanical Data section	69
Updated Figure 72	71

Note(s):

1. Page and figure numbers for the previous version may differ from page and figure numbers in the current revision.
2. Correction of typographical errors is not explicitly mentioned.

Content Guide