



## Datasheet

DS000640

# AS6500

## 4-Channel Time-to-Digital Converter

v4-00 • 2020-Feb-28

---

This product was initially sold under ams AG is now owned by SciSense B.V. .

Only the ownership has changed. The specification and properties of the product remain unchanged.

# Content Guide

<b>1</b>	<b>General Description .....</b>	<b>3</b>	8.1	Time Measurements and Results.....	28
1.1	Key Benefits & Features.....	3	8.2	Resolution .....	30
1.2	Applications .....	4	8.3	Combining Two Stop Channels .....	31
1.3	Block Diagram .....	4	8.4	Input Pins for Time Measurement .....	33
<b>2</b>	<b>Ordering Information .....</b>	<b>5</b>	8.5	SPI Communication Interface.....	35
<b>3</b>	<b>Pin Assignment .....</b>	<b>6</b>	8.6	Coding of Results .....	38
3.1	Pin Diagram.....	6	8.7	Conversion Latency and Conversion Rate.	40
3.2	Pin Description .....	6	8.8	Conversion Rate.....	41
<b>4</b>	<b>Absolute Maximum Ratings .....</b>	<b>9</b>	<b>9</b>	<b>Application Information.....</b>	<b>45</b>
<b>5</b>	<b>Recommended Operation Conditions.....</b>	<b>10</b>	9.1	Configuration Examples .....	45
<b>6</b>	<b>Typical Characteristics .....</b>	<b>11</b>	9.2	Example C++ Code .....	45
6.1	Converter Characteristics.....	11	9.3	Schematic .....	48
6.2	Power Supply Characteristics .....	12	9.4	External Components .....	49
6.3	Reference Clock and Stop Input Requirements .....	13	<b>10</b>	<b>Package Drawings &amp; Markings... 51</b>	
6.4	Serial Communication Interface .....	14	<b>11</b>	<b>Reel Information .....</b>	<b>53</b>
6.5	Typical Operating Characteristics .....	16	<b>12</b>	<b>Soldering &amp; Storage Information 54</b>	
<b>7</b>	<b>Register Description .....</b>	<b>19</b>			
7.1	Register Overview .....	19			
7.2	Detailed Register Description.....	21			
<b>8</b>	<b>Detailed Description.....</b>	<b>28</b>			

# 1 General Description

The AS6500 is a high performance time-to-digital converter (TDC) frontend device. It is a derivative of TDC-GPX2, with CMOS inputs and serial SPI output only. It comes in a QFN40 package. AS6500 achieves high measurement performance and high data throughput. High configuration flexibility and unlimited measurement range cover many applications, ranging from portable handheld laser range equipment to ambitious time-of-flight measurements of high performance.

AS6500 calculates calibrated stop measurements, referenced to the applied reference clock. Combinations of single shot accuracy of 10ps with lowest pulse-to-pulse spacing <10ns and maximum data input burst rate of 70 MSPS per stop input are possible. Total data throughput via SPI is about 1.5 MSPS.

## 1.1 Key Benefits & Features

The benefits and features of AS6500, 4-Channel Time-to-Digital Converter, are listed below:

**Figure 1:**  
**Added Value of Using AS6500**

Benefits	Features
<ul style="list-style-type: none"> <li>Simple data post-processing thanks to calibrated results</li> </ul>	<ul style="list-style-type: none"> <li>4 stop channels with 20 ns pulse-to-pulse spacing</li> <li>2 combined channels with 10 ns pulse-to-pulse spacing</li> <li>Maximum 1.5 MSPS via SPI</li> <li>Single shot accuracy 20 ps RMS resolution per channel, 10 ps RMS with high resolution option</li> <li>Unlimited measuring range 0 s to 16 s</li> </ul>
<ul style="list-style-type: none"> <li>Event assignment thanks to reference clock index simplifies coincidence measurements</li> <li>Easy pulse width measurements</li> <li>High efficiency thanks high sample rate</li> </ul>	<ul style="list-style-type: none"> <li>Differential reference clock input 2 MHz to 12.5 MHz</li> <li>Inputs CMOS level</li> <li>Readout with SPI, used also for configuration</li> <li>16-stage FIFO per channel</li> <li>Automatic calibration to reference clock (no PLL or DLL)</li> </ul>
<ul style="list-style-type: none"> <li>Compact design thanks to small package and low number of external components</li> <li>Reduced cooling thanks to low power consumption</li> </ul>	<ul style="list-style-type: none"> <li>Supply voltage 3.3 V</li> <li>Power dissipation 60 mW</li> <li>Standby current 60 <math>\mu</math>A</li> <li>QFN40 package (6 mm x 6 mm)</li> </ul>

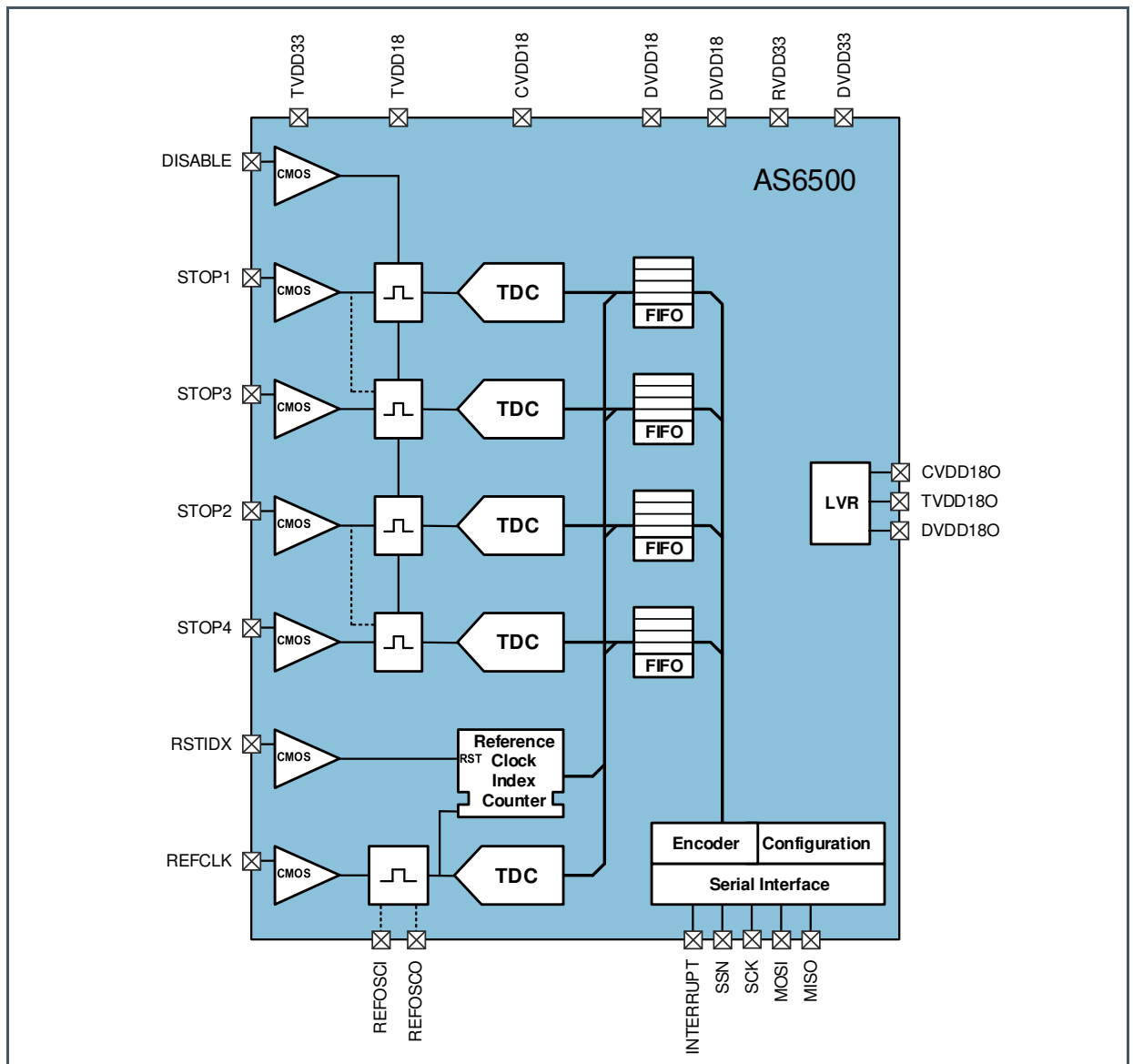
## 1.2 Applications

- Laser range measurement
- Time-of-flight measurement
- Lidar

## 1.3 Block Diagram

The functional blocks of this device are shown below:

**Figure 2:**  
**Functional Blocks of AS6500**



---

## 2 Ordering Information

---

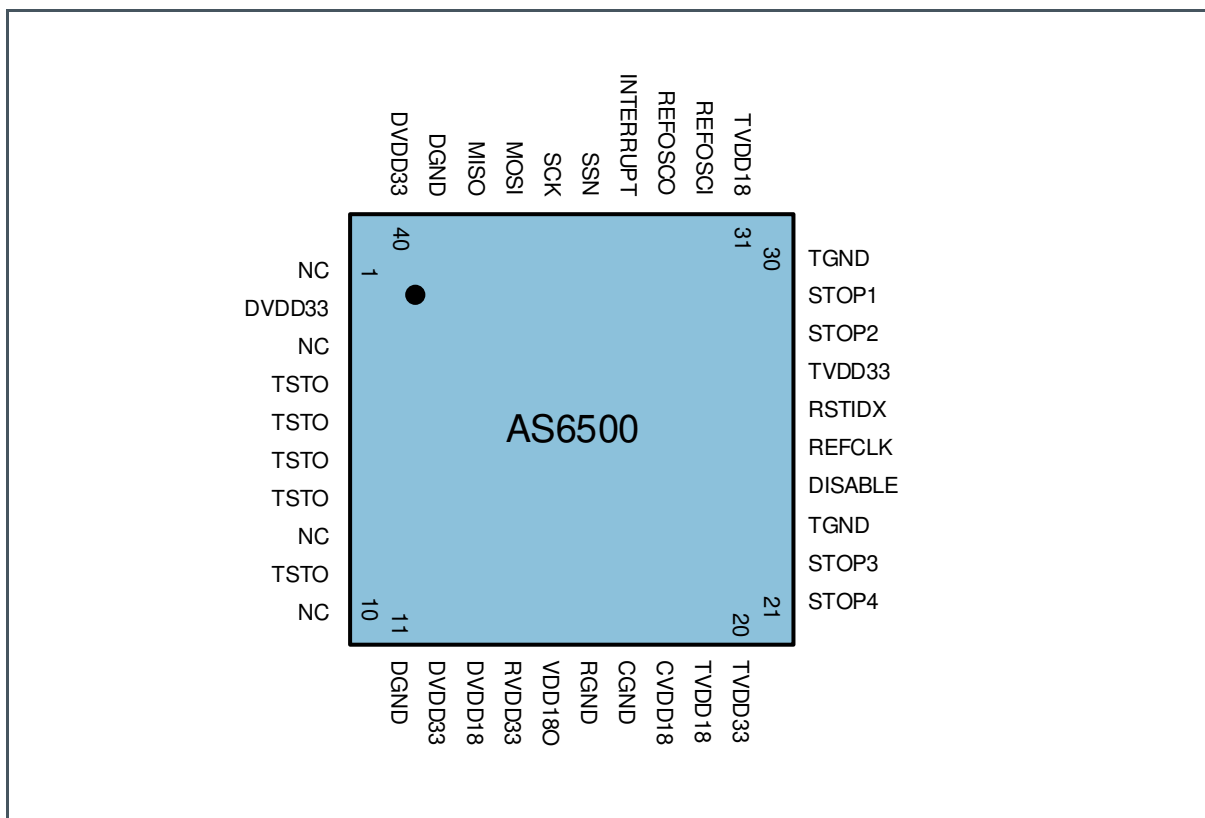
Ordering Code	Package	Marking	Delivery Form	Delivery Quantity
AS6500-FQFM	QFN40	AS6500 -FQF	T&R	500 pcs/reel

---

## 3 Pin Assignment

### 3.1 Pin Diagram

Figure 3:  
AS6500 QFN40



### 3.2 Pin Description

Figure 4:  
Pin Description of AS6500 QFN40

Pin Number	Pin Name	Pin Type	Description	Not Used
1	NC		Not connected	
2	DVDD33	Power Supply	3.3V supply for digital and IO units	
3	NC		Not connected	

Pin Number	Pin Name	Pin Type	Description	Not Used
4	TSTO		Test pin	
5	TSTO		Test pin	
6	TSTO		Test pin	
7	TSTO		Test pin	
8	NC		Not Connected	
9	TSTO		Test pin	
10	NC		Not Connected	
11	DGND	Power Supply	Ground for digital and IO units	
12	DVDD33	Power Supply	3.3V supply for digital and IO units	
13	DVDD18	Power Supply	1.8V supply for digital and IO units	
14	RVDD33	Power Supply	3.3V supply for linear voltage regulator	
15	VDD18O	Regulator Output	1.8V supply for digital and IO units, regulator output	
16	RGND	Power Supply	Ground for linear voltage regulator	
17	CGND	Power Supply	Ground for TDC	
18	CVDD18	Power Supply	1.8V positive supply for TDC	
19	TVDD18	Power Supply	1.8V positive supply for time front-end	
20	TVDD33	Power Supply	3.3V positive supply for time front-end	
21	STOP4	CMOS Input	Positive stop input for channel 4	
22	STOP3	CMOS Input	Positive stop input for channel 3	
23	TGND	Power Supply	Ground for 1.8V time front-end supply	
24	DISABLE	CMOS Input	Positive disabling pin for stop channels	TVDD33
25	REFCLK	CMOS Input	Negative clock signal of reference clock	TVDD33
26	RSTIDX	CMOS Input	Positive reference index reset signal	TVDD33
27	TVDD33	Power Supply	3.3V positive supply for time front-end	
28	STOP2	CMOS Input	Positive stop input for channel 2	
29	STOP1	CMOS Input	Positive stop input for channel 1	
30	TGND	Power Supply	Ground for TDC	
31	TVDD18	Power Supply	1.8V positive supply for time front-end	Open
32	REFOSCI	XOSC Driver In	Input for quartz as reference clock	Open
33	REFOSCI	XOSC Driver Out	Output for quartz as reference clock	Open
34	INTERRUPT	CMOS output	SPI interrupt	Open
35	SSN	LVTTTL Input	SPI slave select not + interface reset	

Pin Number	Pin Name	Pin Type	Description	Not Used
36	SCK	LVTTL Input	SPI serial clock	
37	MOSI	LVTTL Input	SPI serial data master out , slave In	
38	MISO	LVTTL Tristate	SPI serial data master in, slave Out	
39	DGND	Power Supply	Ground for digital and IO units	
40	DVDD33	Power Supply	3.3V supply for digital and IO units	DVDD33



## 4 Absolute Maximum Ratings

Stresses beyond those listed under “Absolute Maximum Ratings” may cause permanent damage to the device. These are stress ratings only. Functional operation of the device at these or any other conditions beyond those indicated under “Operating Conditions” is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

**Figure 5:**  
**Absolute Maximum Ratings of AS6500**

Symbol	Parameter	Min	Max	Unit	Comments
<b>Electrical Parameters</b>					
VDD33	3.3V Supply Voltage to Ground	-0.5	4.0	V	Pin: DVDD33, TVDD33, RVDD33
VDD18	1.8V Supply Voltage to Ground	-0.5	2.2	V	Pin: DVDD18, TVDD18, CVDD18
	Voltage between ground pins	-0.3	+0.3	V	Pin: DGND, TGND, RGND
V <sub>osc</sub>	Voltage at input of oscillator cell	-0.3	VDD18 + 0.3	V	Pin: REFOSCIN
<b>Electrostatic Discharge</b>					
ESD <sub>HBM</sub>	Electrostatic Discharge HBM	± 1000		V	JS-001-2014
<b>Temperature Ranges and Storage Conditions</b>					
T <sub>J</sub>	Operating Junction Temperature	-40	125	°C	
T <sub>STRG</sub>	Storage Temperature Range	- 65	150	°C	
T <sub>BODY</sub>	Package Body Temperature		260	°C	IPC/JEDEC J-STD-020 <sup>(1)</sup>
RH <sub>NC</sub>	Relative Humidity (non-condensing)	5	85	%	
MSL	Moisture Sensitivity Level	3			Maximum floor life time of 168h

- (1) The reflow peak soldering temperature (body temperature) is specified according to IPC/JEDEC J-STD-020 “Moisture/Reflow Sensitivity Classification for Nonhermetic Solid State Surface Mount Devices.” The lead finish for Pb-free leaded packages is “Matte Tin” (100 % Sn)

## 5 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 6:**  
**Recommended Operation Conditions of AS6500**

Symbol	Pin	Description	Min	Typ	Max	Unit
<b>Power-Supply</b>						
VDD33	DVDD33, TVDD33, RVDD33	Supply Voltage	2.4	3.3	3.6	V
VDD18	DVDD18, TVDD18, CVDD18	Core Supply Voltage powered by integrated regulator, pin VDD18O	1.7	1.8	1.9	V
<b>Temperature</b>						
T <sub>A</sub>		Operating free air temperature <sup>(1)</sup>	-40		125	°C
<b>Reference &amp; Stop Inputs</b>						
V <sub>IL,CMOS</sub>	STOP1, STOP2, STOP3, STOP4,	CMOS Input Low Voltage			0.4	V
V <sub>IH,CMOS</sub>	REFCLK, RSTIDX, DISBALE	CMOS Input High Voltage	VDD33 - 0.4			V
<b>SPI-Interface</b>						
V <sub>IL</sub>		Digital Input LOW Voltage			0.8	V
V <sub>IH</sub>	SCK, MOSI, SSN	Digital Input HIGH Voltage	0.7 × VDD33			V
C <sub>LOAD</sub>	INTERRUPT, MISO	Load Capacitance to Ground			20	pF

(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.

## 6 Typical Characteristics

The following test levels apply to all following characteristics:

**Figure 7:**  
**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.

### 6.1 Converter Characteristics

General Conditions: VDD33 = 3.3 V; VDD18 = 1.8 V; T<sub>A</sub> = 0 °C to 80 °C.

**Figure 8:**  
**Converter Characteristics**

Symbol	Description	Condition	TL	Min	Typ	Max	Unit
<b>Accuracy of Time Measurement</b>							
RMS	Single-shot RMS resolution	High_Resolution = 0 (off)			20	30	
		High_Resolution = 1 (2x)	IV		15	20	ps
		High_Resolution = 2 (4x)			10	15	
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)			100		
		High_Resolution = 1 (2x)	V		150		ps
		High_Resolution = 2 (4x)			200		

Symbol	Description	Condition	TL	Min	Typ	Max	Unit
	Offset error temperature drift	High_Resolution = 0 (off)			0.5	3	ps/K
		High_Resolution = 1 (2x)	IV		1		
		High_Resolution = 2 (4x)			1.5		
<b>Switching Performance</b>							
t <sub>CONV</sub>	Converter latency	High_Resolution = 0 (off)				20	ns
		High_Resolution = 1 (2x)	III			50	
		High_Resolution = 2 (4x)				100	
	Peak conversion rate	High_Resolution = 0 (off)				50	MSPS
		High_Resolution = 1 (2x)	III			20	
		High_Resolution = 2 (4x)				10	
	Maximal read-out rate	SCK = 50MHz SPI : Opcode + 16 Bit	III	0.9			MSPS
		Opcode + 16-Bit				2.1	

## 6.2 Power Supply Characteristics

General Conditions: VDD33 = 3.3 V; VDD18 = 1.8 V; T<sub>A</sub> = 0 °C to 80 °C.

**Figure 9:**  
**Power Supply Characteristics**

Symbol	Description	Condition	TL	Min	Typ	Max	Unit
<b>Supply Voltage</b>							
t <sub>VDD18O</sub>	Delay from power-up of RVDD33 to TVDD18, CVDD18, DVDD18 stable	C <sub>load</sub> = 100 μF	V			100	ms
P <sub>TOT,MIN</sub>	Minimum total power dissipation	CMOS inputs and SPI read f <sub>REFCLK</sub> = 5 MHz conversion rate 1MSPS	V		60		mW
<b>Detailed Current Consumption</b>							
I <sub>DVDD18,REFCLK</sub>	Core current into REFCLK	f <sub>REFCLK</sub> = 5 MHz	V		2		mA
I <sub>DVDD18,STOP</sub>	Current per stop channel	Stop rate = 0.5 MHz	V		0.5		mA
I <sub>CVDD18</sub>	Current with activated TDC core		V		14		mA
I <sub>TVDD18,REFOSC</sub>	Quartz oscillator current if used	f <sub>REFOSC</sub> = 4 MHz	III		2		mA

Symbol	Description	Condition	TL	Min	Typ	Max	Unit
I <sub>DDQ</sub>	Quiescent current mainly by I <sub>RVDD33</sub>		II		60	100	μA
I <sub>LKG</sub>	Input leakage current	CMOS, Digital	II	-10		1	μA

## 6.3 Reference Clock and Stop Input Requirements

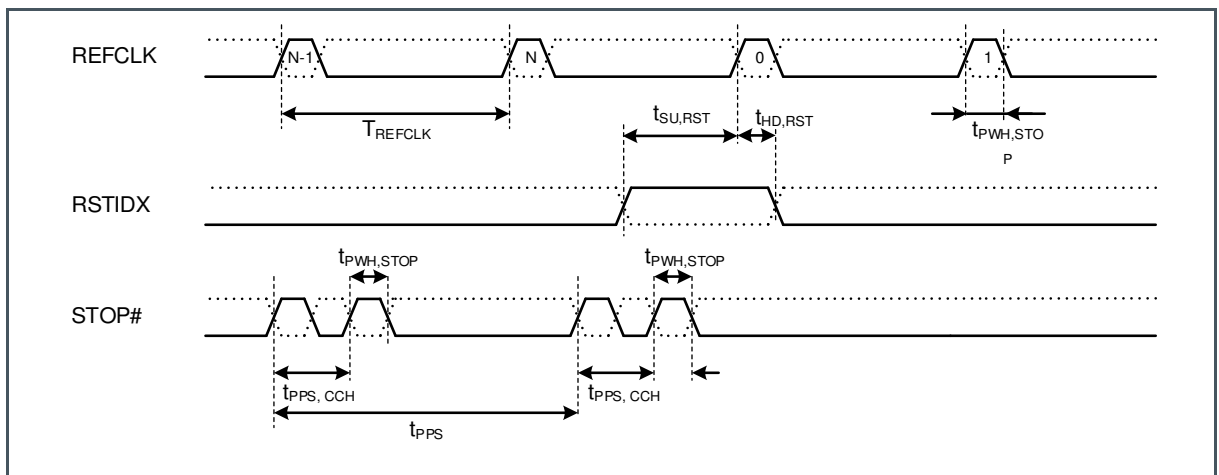
General Conditions: V<sub>DD33</sub> = 3.3 V; V<sub>DD18</sub> = 1.8 V; T<sub>A</sub> = 0 °C to 80 °C; V<sub>ID</sub> = 200mV; V<sub>IC</sub> = 1.25 V; V<sub>IL</sub> = 0 V; V<sub>IH</sub> = 3.3 V

**Figure 10:**  
**Clock and Input Characteristics**

Symbol	Description	Condition	TL	Min	Typ	Max	Unit
f <sub>REFCLK</sub>	Reference clock frequency	High_Resolution = 0 (off) High_Resolution = 1 (2x) High_Resolution = 2 (4x)	II III II	2 2 2	5 5 5	12.5 12.5 10.0	MHz
T <sub>REFCLK</sub>	Reference clock period		III	83	200	500	ns
	Reference clock jitter		V			100	ps
	Reference clock stability	No requirement					
t <sub>PWH,STOP</sub>	Minimum pulse width	CMOS	III	5	10		ns
t <sub>PPS</sub>	Minimum pulse-to-pulse spacing	High_Resolution = 0 (off) High_Resolution = 1 (2x) High_Resolution = 2 (4x)		20 50 100			ns
t <sub>PPS,CCH</sub>	Minimum pulse-to-pulse spacing for a single pair of pulses	CHANNEL_COMBINE = 1	III	20			ns
t <sub>SU,RST</sub>	Setup Time from RSTIDX to REFCLK		III	5			ns
t <sub>HD,RST</sub>	Hold Time from RSTIDX to REFCLK		III	5			ns
t <sub>PIN_ENA</sub>	Pin Activation Time from configuration of PIN_ENA... to valid data	Pins: RSTIDX, REFCLK, STOPA/B	III	200			μs

Symbol	Description	Condition	TL	Min	Typ	Max	Unit
$t_{POR}$	Delay between power-on or initialization reset and next communication	Delay between power-on or initialization reset and next communication	III	100			$\mu s$

**Figure 11:**  
**Timing Symbols and Parameters**



## 6.4 Serial Communication Interface

General Conditions:  $V_{DD33} = 3.3\text{ V}$ ;  $V_{DD18} = 1.8\text{ V}$ ;  $T_A = 0\text{ }^\circ\text{C}$  to  $80\text{ }^\circ\text{C}$ ;  $V_{IL} = 0\text{ V}$ ;  $V_{IH} = 3.3\text{ V}$

**Figure 12:**  
**Serial Communication Interface Characteristics**

Symbol	Description	Condition	TL	Min	Typ	Max	Unit
<b>Electrical Characteristics</b>							
$V_{OL}$	Digital Output LOW Voltage	$I_o = 2\text{ mA}$	III			0.3	mV
$V_{OH}$	Digital Output HIGH Voltage	$I_o = 2\text{ mA}$	III	$DV_{DD33} - 0.3$			mV
<b>Timing Characteristics</b>							
$f_{SCK}$	Serial clock frequency	$C_L = 5\text{ pF}$	III			50	MHz
$t_{PWH,SCK}$	Serial clock pulse width HI state		III	10			ns

Symbol	Description	Condition	TL	Min	Typ	Max	Unit
$t_{PWL,SCK}$	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 13:**  
**Write and Incremental Write**

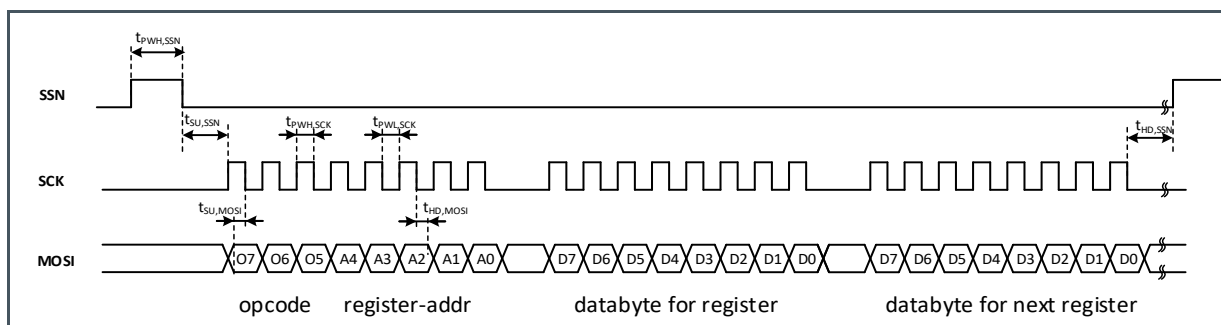
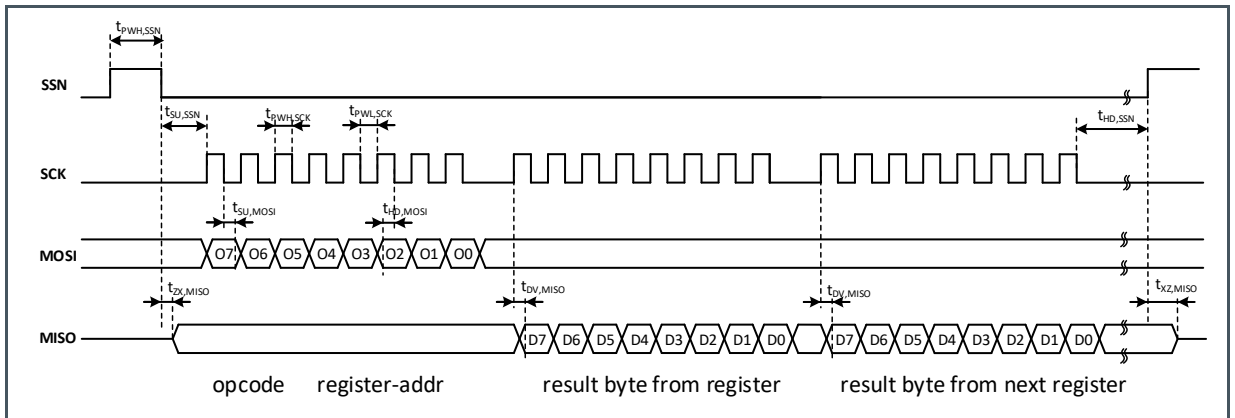


Figure 14:  
Read and Incremental Read



## 6.5 Typical Operating Characteristics

Figure 15:  
STOP, HIGHRES 4x, Histogram 100000 Values

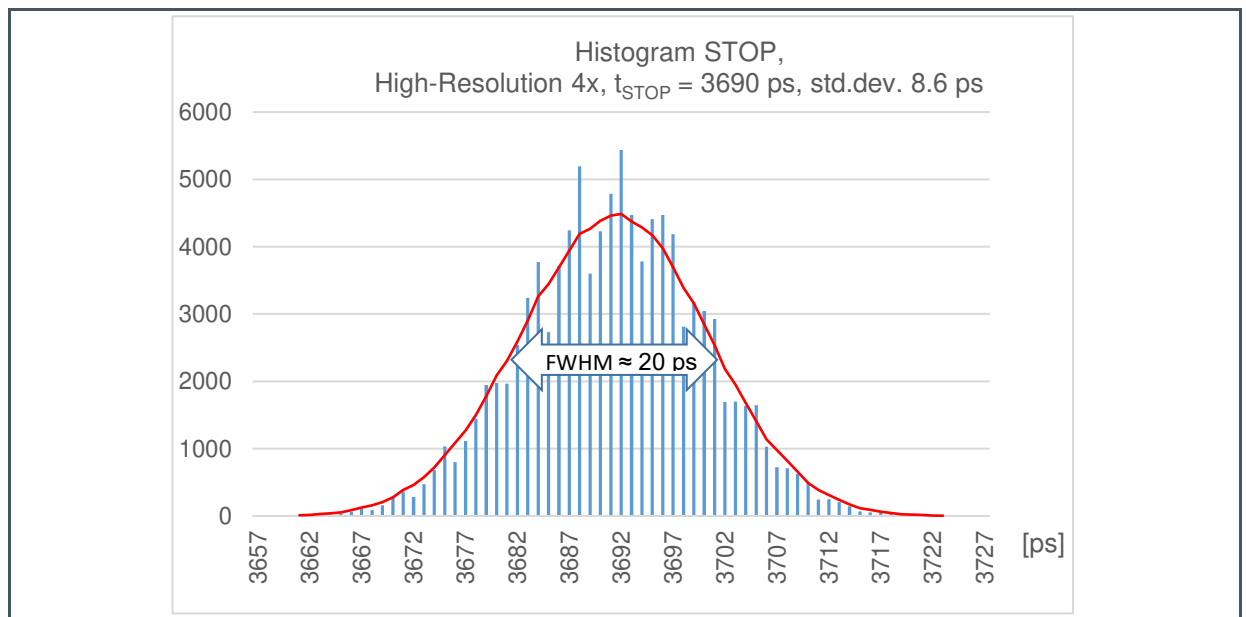




Figure 16:  
STOPB – STOPA, HIGHRES 4x, Histogram 100000 Values

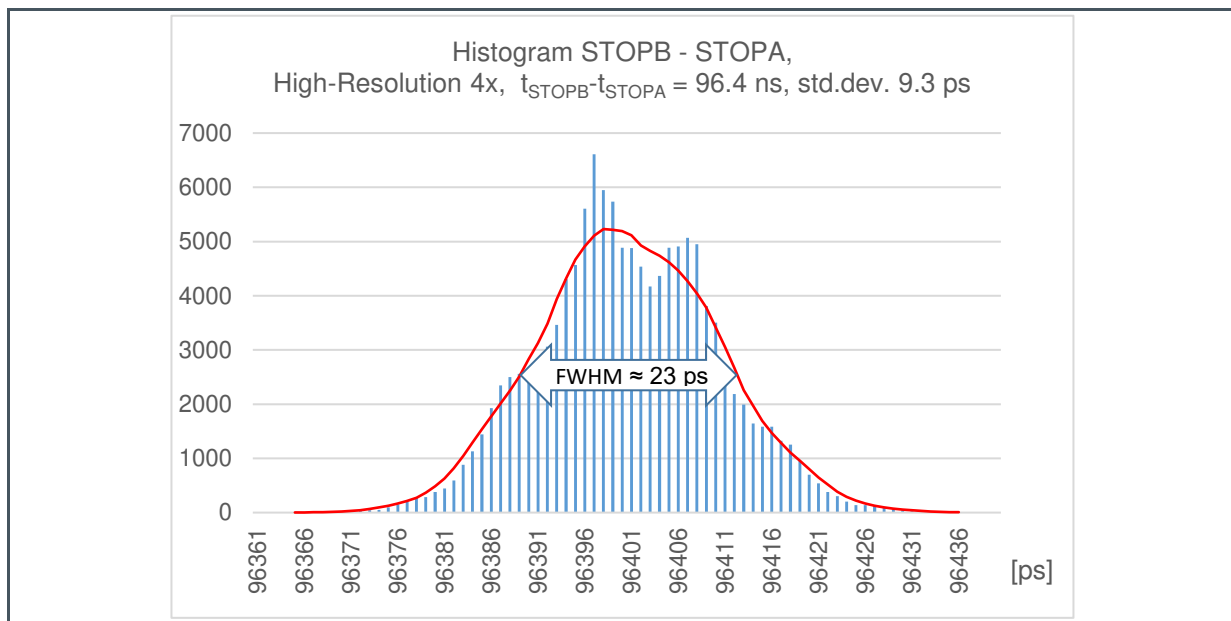


Figure 17:  
STOP – REFCLK, HIGHRES Off, Histogram 100000 Values

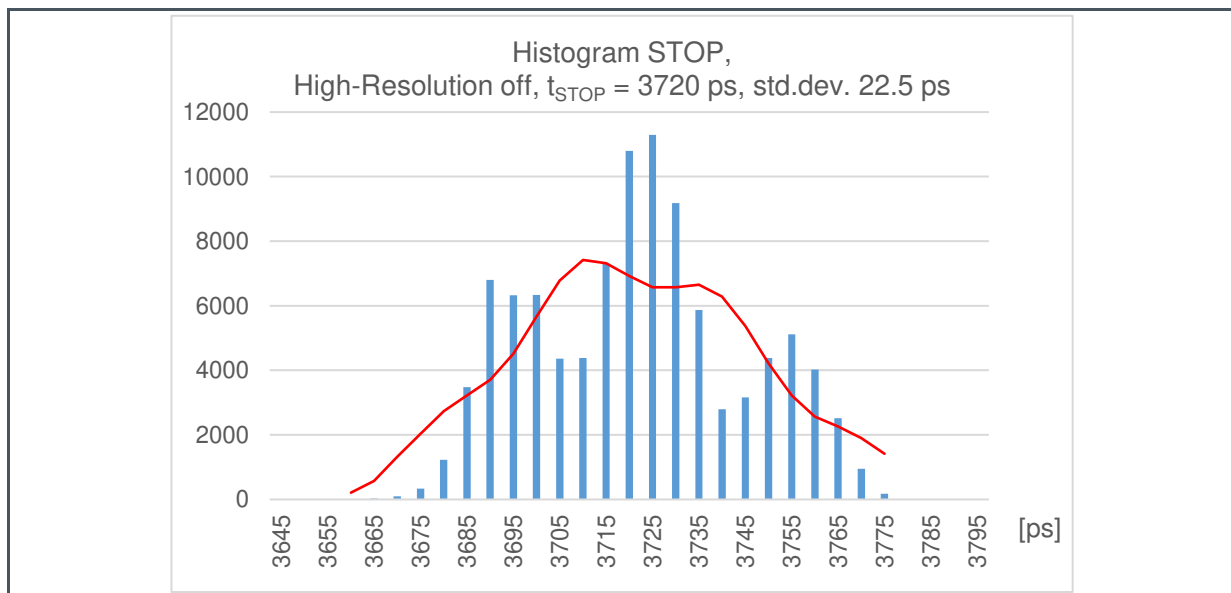


Figure 18:  
STOPB – STOPA, HIGHRES Off, Histogram 100000 Values

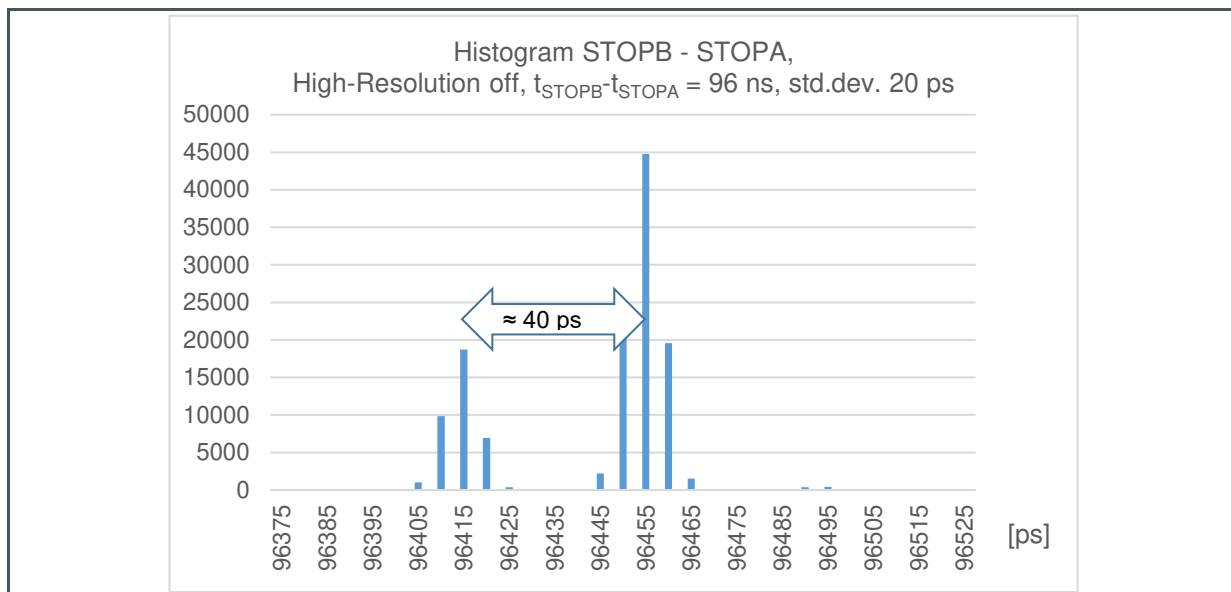
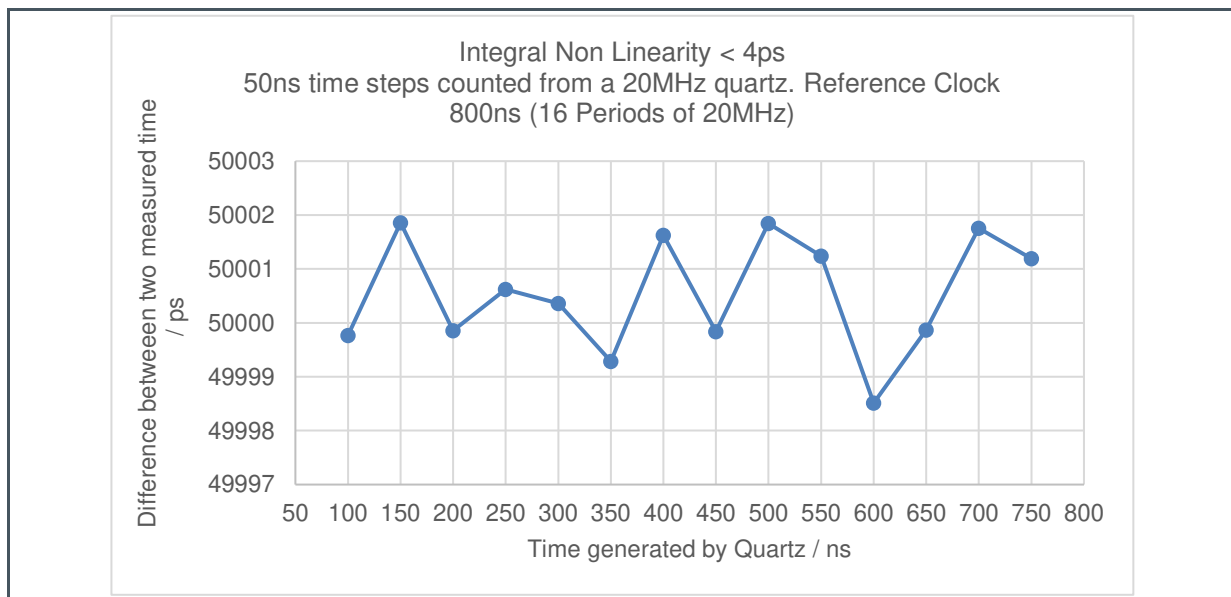


Figure 19:  
Integral Non-Linearity



# 7 Register Description

## 7.1 Register Overview

The configuration registers are organized in 17 addresses of one byte. All configuration registers are accessible via the SPI interface with the `spiopc_write_config` and `spiopc_read_config`. The result read registers are organized in 12 addresses of one byte. All result read registers are accessible via the SPI interface with `spiopc_read_result`. Users can read and write register individually or with an incremental access.

**Figure 20:**  
**Configuration Register Overview**

Addr	Name	<D7>	<D6>	<D5>	<D4>	<D3>	<D2>	<D1>	<D0>	
0	CFG0	PIN_ENA_RSTIDX	PIN_ENA_DISABLE	Fixed value: (0b)	PIN_ENA_REFCLK	PIN_ENA_STOP4	PIN_ENA_STOP3	PIN_ENA_STOP2	PIN_ENA_STOP1	
1	CFG1	HIGH_RESOLUTION		CHANNEL_COMBINE		HIT_ENA_STOP4	HIT_ENA_STOP3	HIT_ENA_STOP2	HIT_ENA_STOP1	
2	CFG2	BLOCK_WISE_FIFO_READ	COMMON_FIFO_READ	Fixed value: (000000b)						
3	CFG3	REFCLK_DIVISIONS (Lower byte)								
4	CFG4	REFCLK_DIVISIONS (Middle byte)								
5	CFG5	Fixed value: (0000b)				REFCLK_DIVISIONS (Upper bits)				
6	CFG6	Fixed value: (11000000b)								
7	CFG7	REFCLK_BY_XOSC	Fixed value: (10100011b)							
8	CFG8	Fixed value: (10100001b)								
9	CFG9	Fixed value: (00010011b)								
10	CFG10	Fixed value: (00000000b)								
11	CFG11	Fixed value: (00001010b)								
12	CFG12	Fixed value: (11001100b)								
13	CFG13	Fixed value: (00000101b)								
14	CFG14	Fixed value: (11110001b)								
15	CFG15	Fixed value: (01111101b)								
16	CFG16	Fixed value: (00000100b)								

All register are read/write with 0 as default value, besides registers 13 and 14 with 5 as default value. The fixed values are assigned by **ams**: Unless otherwise suggested, they should be set as shown in this table.

**Figure 21:**  
**Result Register Overview**

Addr	Name	<D7>
0 to 7		n.c.
8	CHANNEL1	REFERENCE INDEX CHANNEL 1 BYTE #3
9		REFERENCE INDEX CHANNEL 1 BYTE #2
10		REFERENCE INDEX CHANNEL 1 BYTE #1
11		STOP RESULT CHANNEL 1 BYTE #3
12		STOP RESULT CHANNEL 1 BYTE #2
13		STOP RESULT CHANNEL 1 BYTE #1
14	CHANNEL2	REFERENCE INDEX CHANNEL A BYTE #3
15		REFERENCE INDEX CHANNEL A BYTE #2
16		REFERENCE INDEX CHANNEL A BYTE #1
17		STOP RESULT CHANNEL A BYTE #3
18		STOP RESULT CHANNEL A BYTE #2
19		STOP RESULT CHANNEL A BYTE #1
20	CHANNEL3	REFERENCE INDEX CHANNEL 3 BYTE #3
21		REFERENCE INDEX CHANNEL 3 BYTE #2
22		REFERENCE INDEX CHANNEL 3 BYTE #1
23		STOP RESULT CHANNEL 3 BYTE #3
24		STOP RESULT CHANNEL 3 BYTE #2
25		STOP RESULT CHANNEL 3 BYTE #1
26	CHANNEL4	REFERENCE INDEX CHANNEL 4 BYTE #3
27		REFERENCE INDEX CHANNEL 4 BYTE #2
28		REFERENCE INDEX CHANNEL 4 BYTE #1
29		STOP RESULT CHANNEL 4 BYTE #3
30		STOP RESULT CHANNEL 4 BYTE #2
31		STOP RESULT CHANNEL 4 BYTE #1

## 7.2 Detailed Register Description

### 7.2.1 CFG0 Register (Address 0)

Figure 22:  
CFG0 Register

Addr: 0		CFG0		
Bit	Bit Name	De- fault	Ac- cess	Bit Description
0	PIN_ENA_STOP1	0	RW	Activation on stop event input pin STOP1 0:= Stop input pins not active 1:= Stop input pins active
1	PIN_ENA_STOP2	0	RW	Activation on stop event input pin STOP2 0:= Stop input pins not active 1:= Stop input pins active
2	PIN_ENA_STOP3	0	RW	Activation on stop event input pin STOP3 0:= Stop input pins not active 1:= Stop input pins active
3	PIN_ENA_STOP4	0	RW	Activation on stop event input pin STOP4 0:= Stop input pins not active 1:= Stop input pins active
4	PIN_ENA_REFCLK	0	RW	0:= REFCLK input pins not active 1:= REFCLK input pins active
5	Fixed value:	0	RW	(0b)
6	PIN_ENA_DISABLE	0	RW	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	RW	0:= Deactivation of reference clock index counter reset pin 1:= Activation of reference clock index counter reset pin

## 7.2.2 CFG1 Register (Address 1)

Figure 23:  
CFG1 Register

Addr: 1		CFG1		
Bit	Bit Name	De- fault	Ac- cess	Bit Description
0	HIT_ENA_STOP1	0	RW	0:= Stop events are internally rejected. The pin enabling of STOP1 is not affected. 1:= Stop events are internally accepted and processed. Normal working condition
1	HIT_ENA_STOP2	0	RW	0:= Stop events are internally rejected. The pin enabling of STOP2 is not affected. 1:= Stop events are internally accepted and processed. Normal working condition
2	HIT_ENA_STOP3	0	RW	0:= Stop events are internally rejected. The pin enabling of STOP3 is not affected. 1:= Stop events are internally accepted and processed. Normal working condition
3	HIT_ENA_STOP4	0	RW	0:= Stop events are internally rejected. The pin enabling of STOP4 is not affected. 1:= Stop events are internally accepted and processed. Normal working condition
4, 5	CHANNEL_COMBINE	0	RW	The two stop channels may be combined for improved pulse pair resolution or higher conversion rate. 00b := Normal operation with two independent stop channels 01b := "Pulse distance" Stop events at STOPA are measured alternatingly by stop channels A & B 10b := "Pulse width" The rising edges at STOPA are measured by stop channel A, the falling edges at STOPA are measured by stop channel B
6, 7	HIGH_RESOLUTION	0	RW	A stop event is internally delayed, measured several times and summed up in order to one result to increase the time resolution. 00b := 0 (Off): standard resolution with minimal pulse-to-pulse spacing. 01b := 1 (2x): A stop event is measured twice 10b := 2 (4x): A stop event is measured four times

### 7.2.3 CFG2 Register (Address 2)

Figure 24:  
CFG2 Register

Addr: 2		CFG2		
Bit	Bit Name	De- fault	Ac- cess	Bit Description
[5:0]	Fixed value:	0	RW	(000000b)
6	COMMON_FIFO_READ	0	RW	0:= Off INTERRUPT pin is set to zero, as soon as one FIFOs does have a value.OFF. 1:= On INTERRUPT 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	RW	0:= OFF, Operation with standard FIFO function 1:= Data output 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

### 7.2.4 CFG3, CFG4, CFG5 Registers (Addresses 3 to 5)

These registers combine for a 20-bit value.

Figure 25:  
CFG3, CFG4, CFG5 Registers

Addr:		CFGRG3		
Bit	Bit Name	De- fault	Ac- cess	Bit Description
0 to 7	REFCLK_DIVISIONS Lower 8 bits	0	RW	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
Addr: 4		CFGRG4		
0 to 7	REFCLK_DIVISIONS Middle 8 bits	0	RW	See above

Addr: 5		CFGRG5		
0 to 3	REFCLK_DIVISIONS High 4 bits	0	RW	See above
4 to 7	Fixed value:	0	RW	(0000b)

## 7.2.5 CFG6 Register (Address 6)

Figure 26:  
CFG6 Register

Addr: 6		CFG6		
Bit	Bit Name	De- fault	Ac- cess	Bit Description
0 to 7	Fixed value:	0	RW	(11000000b)

## 7.2.6 CFG7 Register (Address 7)

Figure 27:  
CFG7 Register

Addr: 7		CFG7		
Bit	Bit Name	De- fault	Ac- cess	Bit Description
0 to 6	Fixed value:	0	RW	(100011b)
7	REFCLK_BY_XOSC	0	RW	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 AS6500; REFCLK pins are not in use and should be disabled with PIN_ENA_REFCLK.

## 7.2.7 CFG8 to CFG15 Register (Addresses 8 to 15)

For registers 8 to 15 use the default fixed values as shown in the Register Overview.



## 7.2.8 CFG16 Register (Address 16)

Figure 28:  
CFG16 Register

Addr: 16		CFG16		
Bit	Bit Name	De- fault	Ac- cess	Bit Description
0 to 7	Fixed value:	0	RW	(00000100b)

## 7.2.9 CHANNEL1 Result Register (Addresses 8 to 13)

Channel1 register is made of 6 bytes. Three bytes for the reference index REFID, three bytes for the time stamp:

Figure 29:  
CHANNEL1 Register

Address	Name	Description	Format
8	REFID1	REFERENCE INDEX CH1 BYTE #3	$REFID = 2^{16} \times \text{BYTE}\#3 + 2^8 \times \text{BYTE}\#2 + \text{BYTE}\#1$
9		REFERENCE INDEX CH1 BYTE #2	
10		REFERENCE INDEX CH1 BYTE #1	
11	TSTOP1	STOP RESULT CH1 BYTE #3	$TSTOP = 2^{16} \times \text{BYTE}\#3 + 2^8 \times \text{BYTE}\#2 + \text{BYTE}\#1$
12		STOP RESULT CH1 BYTE #2	
13		STOP RESULT CH1 BYTE #1	

REFID is the reference index of the preceding reference clock edge.

TSTOP is the ratio of the internal measured times of  $t_{STOP}$  over  $t_{REF}$  scaled by the configured REFCLK\_DIVISONS. For details see section Time Measurements and Results.

## 7.2.10 CHANNEL2 Result Register (Addresses 14 to 19)

Channel2 register is made of 6 bytes. Three bytes for the reference index REFID, three bytes for the time stamp:

**Figure 30:**  
**CHANNEL2 Register**

Address	Name	Description	Format
14	REFID2	REFERENCE INDEX CH2 BYTE #3	$REFID = 2^{16} \times \text{BYTE}\#3 + 2^8 \times \text{BYTE}\#2 + \text{BYTE}\#1$
15		REFERENCE INDEX CH2 BYTE #2	
16		REFERENCE INDEX CH2 BYTE #1	
17	TSTOP2	STOP RESULT CH2 BYTE #3	$TSTOP = 2^{16} \times \text{BYTE}\#3 + 2^8 \times \text{BYTE}\#2 + \text{BYTE}\#1$
18		STOP RESULT CH2 BYTE #2	
19		STOP RESULT CH2 BYTE #1	

REFID is the reference index of the preceding reference clock edge.

TSTOP is the ratio of the internal measured times of  $t_{STOP}$  over  $t_{REF}$  scaled by the configured REFCLK\_DIVISIONS. For details see section Time Measurements and Results.

### 7.2.11 CHANNEL3 Result Register (Addresses 20 to 25)

Channel3 register is made of 6 bytes. Three bytes for the reference index REFID, three bytes for the time stamp:

**Figure 31:**  
**CHANNEL3 Register**

Address	Name	Description	Header row left aligned
20	REFID3	REFERENCE INDEX CH3 BYTE #3	$REFID = 2^{16} \times \text{BYTE}\#3 + 2^8 \times \text{BYTE}\#2 + \text{BYTE}\#1$
21		REFERENCE INDEX CH3 BYTE #2	
22		REFERENCE INDEX CH3 BYTE #1	
23	TSTOPB	STOP RESULT CH3 BYTE #3	$TSTOP = 2^{16} \times \text{BYTE}\#3 + 2^8 \times \text{BYTE}\#2 + \text{BYTE}\#1$
24		STOP RESULT CH3 BYTE #2	
25		STOP RESULT CH3 BYTE #1	

REFID is the reference index of the preceding reference clock edge.

TSTOP is the measured time as ratio of the internal measured times of  $t_{STOP}$  over  $t_{REF}$  scaled by the configured REFCLK\_DIVISIONS. For details see section Time Measurements and Results.

### 7.2.12 CHANNEL4 Result Register (Addresses 26 to 31)

Channel4 register is made of 6 bytes. Three bytes for the reference index REFID, three bytes for the time stamp:

**Figure 32:**  
**CHANNEL4 Register**

Address	Name	Description	Header row left aligned
26	REFID4	REFERENCE INDEX CH4 BYTE #3	REFID = $2^{16} \times \text{BYTE\#3} + 2^8 \times \text{BYTE\#2} + \text{BYTE\#1}$
27		REFERENCE INDEX CH4 BYTE #2	
28		REFERENCE INDEX CH4 BYTE #1	
29	TSTOP4	STOP RESULT CH4 BYTE #3	TSTOP = $2^{16} \times \text{BYTE\#3} + 2^8 \times \text{BYTE\#2} + \text{BYTE\#1}$
30		STOP RESULT CH4 BYTE #2	
31		STOP RESULT CH4 BYTE #1	

REFID is the reference index of the preceding reference clock edge.

TSTOP is the measured time as ratio of the internal measured times of  $t_{\text{STOP}}$  over  $t_{\text{REF}}$  scaled by the configured REFCLK\_DIVISIONS. For details see section Time Measurements and Results.

## 8 Detailed Description

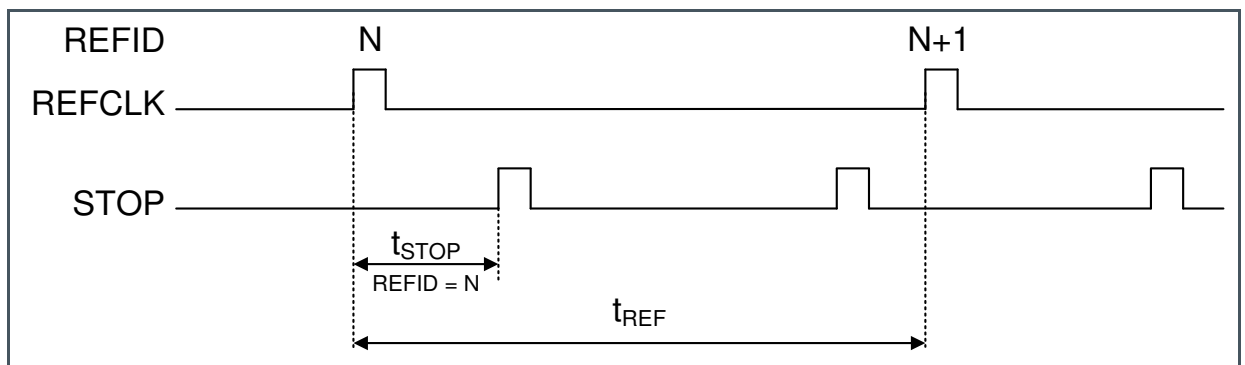
### 8.1 Time Measurements and Results

#### 8.1.1 Measurements of AS6500

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 33:**  
**AS6500 Time Measurement**



#### 8.1.2 Output Results

Each stop generates a dataset that 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 that belong to different reference clock periods. The maximum length of the reference index is 20 bits.

TSTOP is the ratio of the internal measured times of  $t_{STOP}$  over  $t_{REF}$  scaled by the configured REFCLK\_DIVISIONS. The readout result TSTOP is always less than configured REFCLK\_DIVISIONS. The resulting LSB at the output interface should be chosen much lower than the single shot resolution of AS6500. For details, see chapter Coding of Results. Suitable values are e.g. 1 ps, 5 ps or 10 ps.

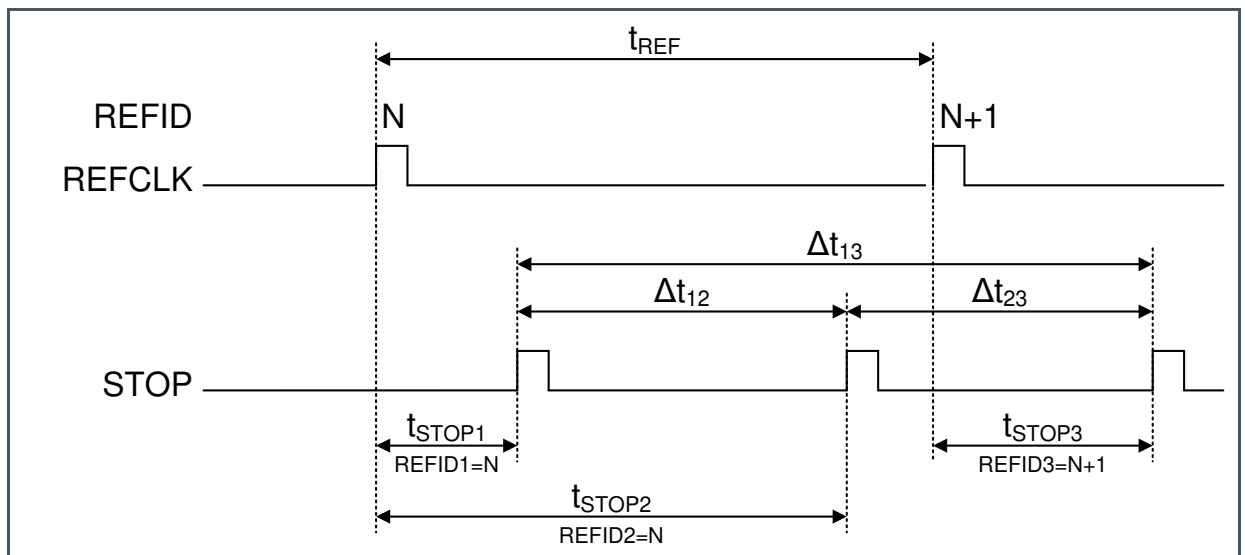
Figure 34:  
Time Calculation

$$t_{STOP} = \underbrace{\frac{t_{STOP}}{t_{REF}} \times REFCLK\_DIVISIONS}_{\substack{\text{Ratio of internal} \\ \text{time measurements} \\ \text{Internal calculated result for read-out}}} \times \underbrace{\frac{t_{REFCLK-PERIOD}}{REFCLK\_DIVISIONS}}_{\substack{\text{LSB resulting by the} \\ \text{period of the applied} \\ \text{reference clock} \\ \text{and by the configured} \\ \text{REFCLK\_DIVISIONS}}}$$

### 8.1.3 Calculation of Time Differences

The results of the AS6500 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 35:  
Calculating Time Differences



## GENERAL APPROACH

Both data REFID and TSTOP are available on the output interface. 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).

$$\Delta t_{13} = (TSTOP3 - TSTOP1) + (REFID3 - REFID1) * REFCLK\_DIVISIONS$$

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, it is not necessary to read out the reference index. It is sufficient to read out just the stop results and to calculate the difference:

$$\Delta t_{12} = TSTOP2 - TSTOP1 \quad \text{if } REFID2 = REFID1$$

- **TIME DIFFERENCE SMALLER THAN REFERENCE CLOCK**

In applications where the measured time difference  $\Delta t$  is always smaller than the reference clock period TREF but not necessarily in the same reference clock period, it is often sufficient to read out just the stop results without the reference index by distinguishing positive and negative time difference:

$$\Delta t_{23} = (TSTOP3 - TSTOP2) \quad \text{if } TSTOP3 - TSTOP2 > 0$$

$$\Delta t_{23} = (TSTOP3 - TSTOP2) + REFCLK\_DIVISIONS \quad \text{if } TSTOP3 - TSTOP2 < 0$$

## 8.2 Resolution

### 8.2.1 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.

### 8.2.2 High Resolution

For achieving best single-shot RMS resolution, AS6500 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 AS6500 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.

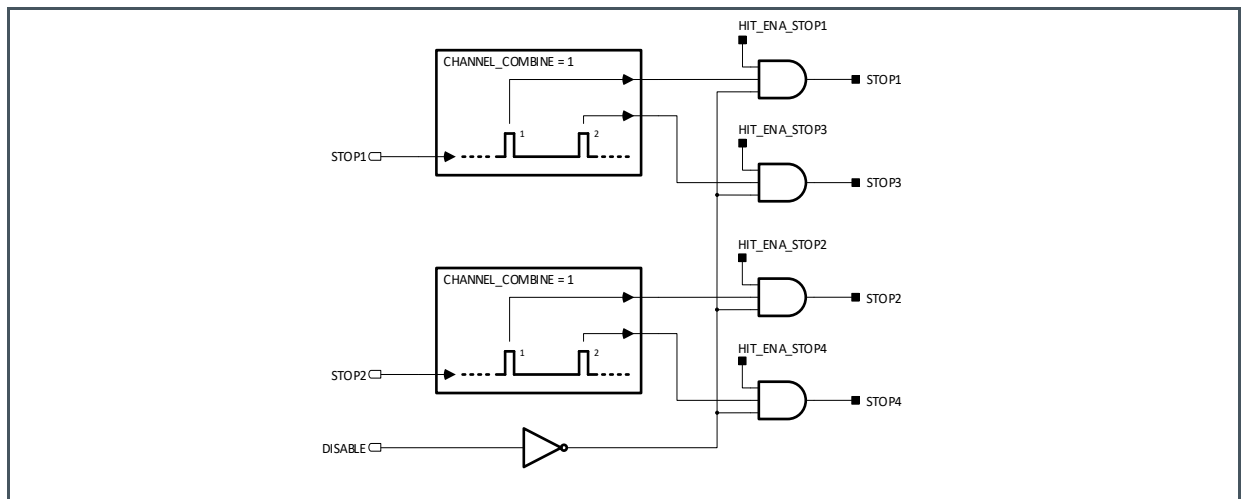
## 8.3 Combining Two Stop Channels

### 8.3.1 Channel Combination for Low Pulse-to-Pulse Spacing

With CHANNEL\_COMBINE set to “PULSE\_SPACING”, the two stop channels A & B are connected to one input pin STOPA. The stop events at the input pin are distributed alternately 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

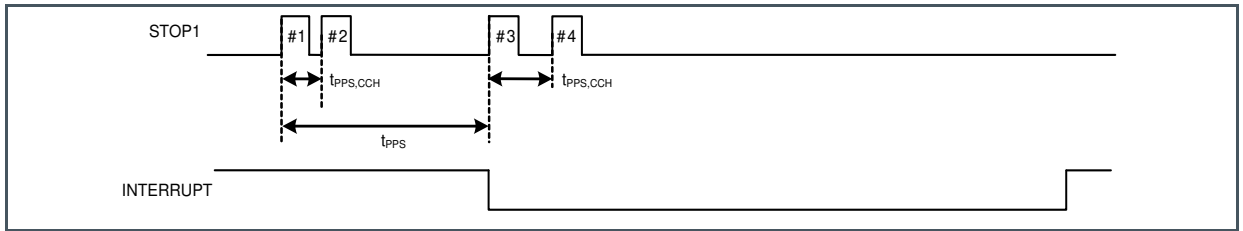
- Excellent pulse-to-pulse spacing
- Doubled FIFO depth per stop input pin
- Higher burst storage capability
- HIGH\_RESOLUTION is applicable

**Figure 36:**  
**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 37:**  
**Channel Combination Low Pulse-to-Pulse Spacing**



**Information**

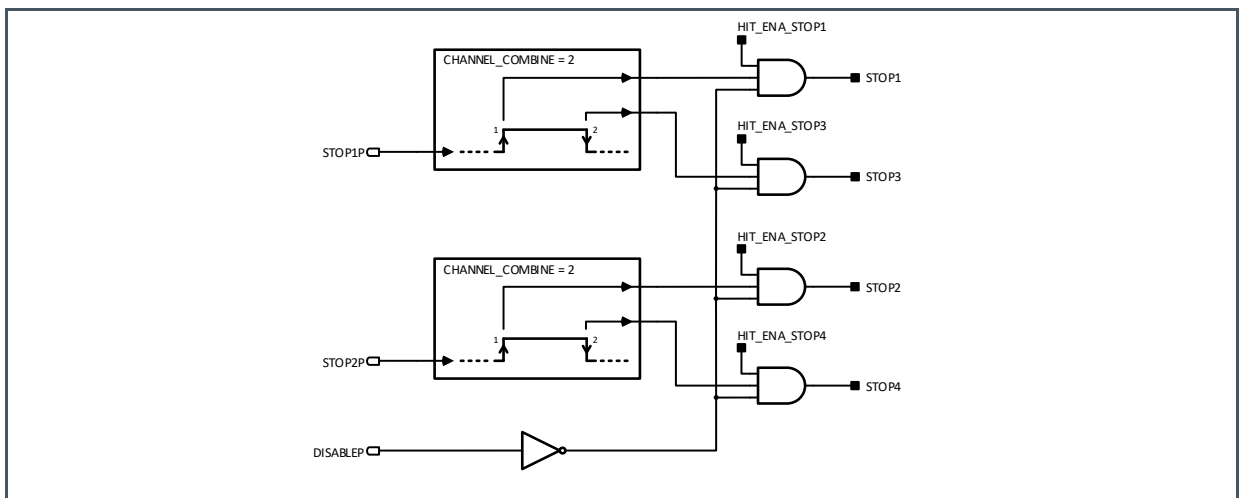
SPI readout of combined channel pairs is permitted only pairwise like ch1-ch3-ch1-ch3- or ch2-ch4-ch2-ch4-... . Also, incremental readout like ch1-ch2-ch3-ch4... is possible. But it is not permitted to read one channel twice like ch1-ch1-ch3-ch3- or ch2-ch2-ch4-ch4.

**8.3.2 Channel Combination for Pulse Width Measurement**

With CHANNEL\_COMBINE set to “PULSE\_WIDTH” the two internal stop channels 1 & 3 are connected to one input pin STOPA. The rising edges are measured by channel A, falling edges are measured by channel B. Readout starts on both channels simultaneous when a rising and falling edge was measured.

HIGH\_RESOLUTION or COMMON\_FIFO\_READ is fully applicable

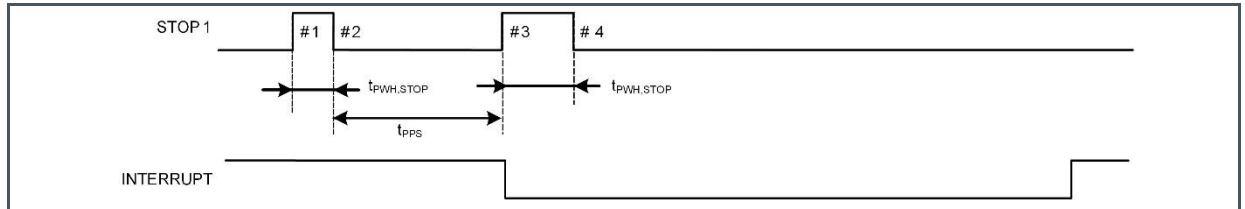
**Figure 38:**  
**Channel Combination Pulse Width Measurement**





**Note:** For internal processing reasons, after the conversion latency  $t_{PPS}$  the next pulses can be captured earliest. Measurements with HIGH\_RESOLUTION will increase the conversion latency but minimum pulse width  $t_{PWH,STOP}$  is not affected.

**Figure 39:**  
**Channel Combination Pulse Width Measurement**



### Information

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.

## 8.4 Input Pins for Time Measurement

Besides the STOP inputs, there are other inputs relevant for the time measurement:

- REFCLK
- REFOSC
- REFRES
- DISABLE

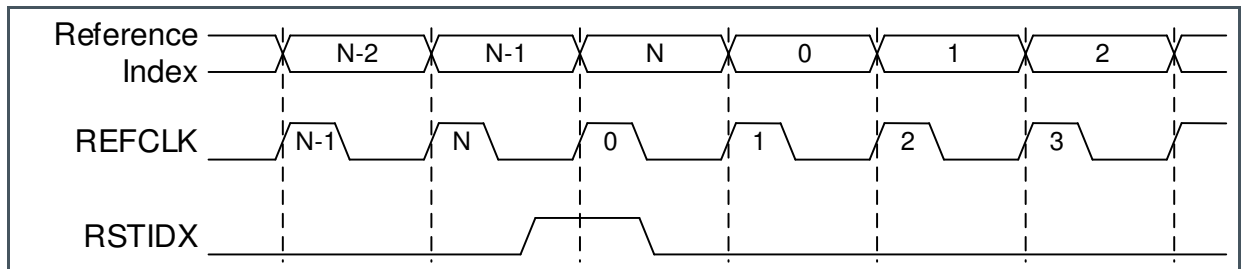
### 8.4.1 REFCLK: Reference Clock Input

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 AS6500 itself. But it will directly affect the quality of the time measurement results.

### 8.4.2 RSTIDX: 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 at least a single period. After release of RSTIDX, 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**



### 8.4.3 STOP1 to STOP4: 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.

### 8.4.4 DISABLE: 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.

#### 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 input drivers of the related pins.

## 8.5 SPI Communication Interface

### 8.5.1 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

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

Clock Phase Bit = 1    Clock Polarity Bit = 0

### 8.5.2 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.

### 8.5.3 Communication Commands (Opcodes)

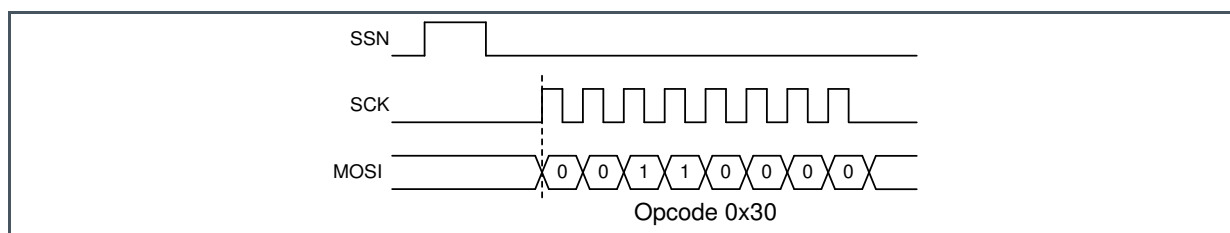
**Figure 41:**  
**Opcodes Overview**

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	0x60 = 0b011XXXXX
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

- 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 stops and the configuration registers are set to internal defaults of the chip.

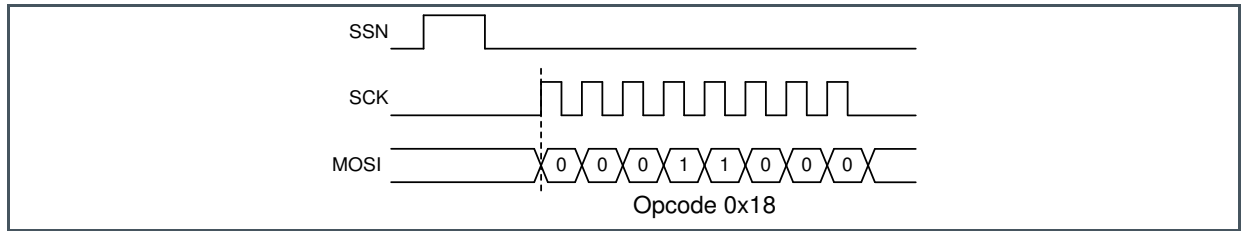
**Figure 42:**  
**Power-On Reset Opcode**



- 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 opens 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.

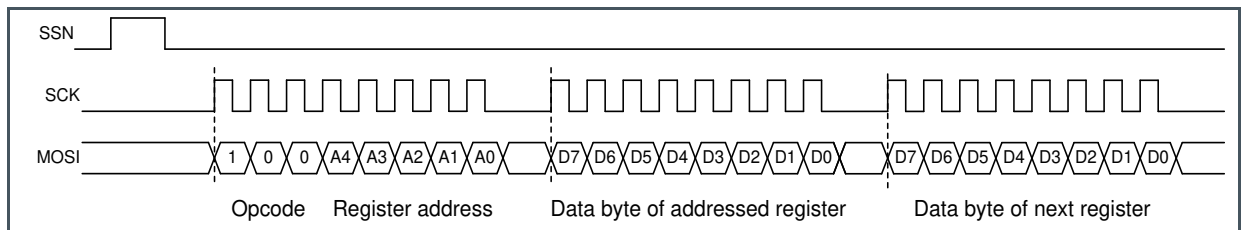
**Figure 43:**  
**Init Reset Opcode**



- 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.

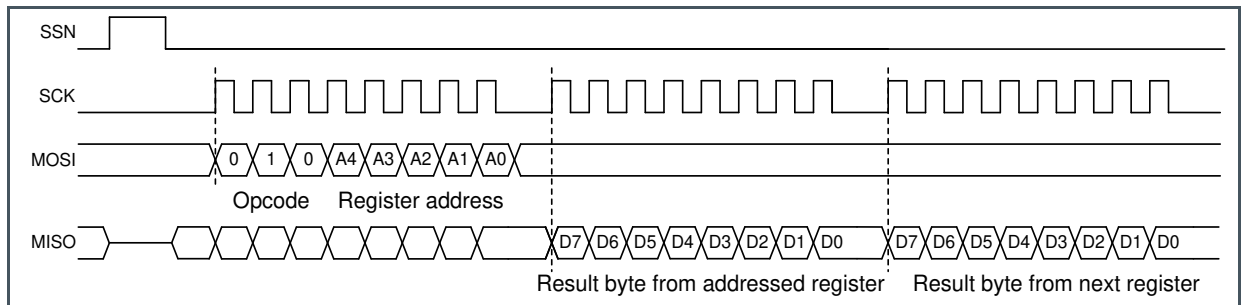
**Figure 44:**  
**SPI Incremental Write**



- 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 45:**  
**SPI Incremental Read**



## 8.5.4 Data Readout via SPI Interface

Reading results byte-wise from AS6500 e.g. by an external microcontroller is fully supported.

When reading an empty channel the results of REFINDEX and STOPRESULT are marked with 0xFFFFFFFF. Typically, the measurement rate of AS6500 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 AS6500.

REF\_INDEX\_BITWIDTH and STOP\_DATA\_BITWIDTH are not relevant for reading via SPI.

## 8.6 Coding of Results

### 8.6.1 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 scale 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 AS6500. The most convenient way is choosing an LSB of 1 ps by configuring REFCLK\_DIVISIONS to the picosecond value of the reference clock period. Other LSB settings are possible as well, like LSB of 5 ps or 10 ps.

**Figure 46:**  
**LSB Configuration**

Reference Clock Period	Reference Clock Frequency	REFCLK_DIVISIONS LSB = 1 ps	REFCLK_DIVISIONS LSB = 5 ps	REFCLK_DIVISIONS LSB = 10 ps
500 ns	2 MHz	500000	100000	50000
250 ns	4 MHz	250000	50000	25000
200 ns	5 MHz	200000	40000	20000
100 ns	10 MHz	100000	20000	10000
80 ns	12.5 MHz	80000	16000	8000

## 8.6.2 Examples for Codes of Time Measurements Results

**Figure 47:**  
**LSB Configuration**

Readout of Stop Result		Resulting Stop Time with an Assumed LSB of			Note
Hexadecimal	Decimal	LSB = 1ps	LSB = 5ps	LSB = 10ps	
0x1	1	1 ps	5 ps	10 ps	
0x2	2	2 ps	10 ps	20 ps	
0xA	10	10 ps	50 ps	100 ps	
0x64	100	100 ps	500 ps	1000 ps	
0x3E8	1000	1000 ps	5000 ps	10000 ps	
0x2710	10000	10000 ps	50000 ps	100000 ps	
0x61A7	24999	24999 ps	124995 ps	249990 ps	refclk-period t <sub>REFCLK</sub> = 250ns
0xC34F	49999	49999 ps	249995 ps <sup>(2)</sup>	<sup>(1)</sup>	
0x3D08F	249999	249999 ps <sup>(2)</sup>	<sup>(1)</sup>	<sup>(1)</sup>	
0x1869F	99999	99999 ps	499995 ps	<sup>(1)</sup>	refclk-period t <sub>REFCLK</sub> = 1μs
0x30D3F	199999	199999 ps	<sup>(1)</sup>	<sup>(1)</sup>	
0xF423F	999999	<sup>(1)</sup>	<sup>(1)</sup>	<sup>(1)</sup>	
0x3FFF	16383	16383 ps	81915 ps	163830 ps	
0xFFFF	65335	65335 ps	326675 ps	653350 ps	
0x3FFFF	262143	262143 ps	<sup>(1)</sup>	<sup>(1)</sup>	
0x0FFFFF	1048575	1048575 ps	<sup>(1)</sup>	<sup>(1)</sup>	SPI: max readout with 20Bit <sup>(2)</sup>

- (1) Time difference exceed AS6500 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

### 8.6.3 Maximum Time Differences

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

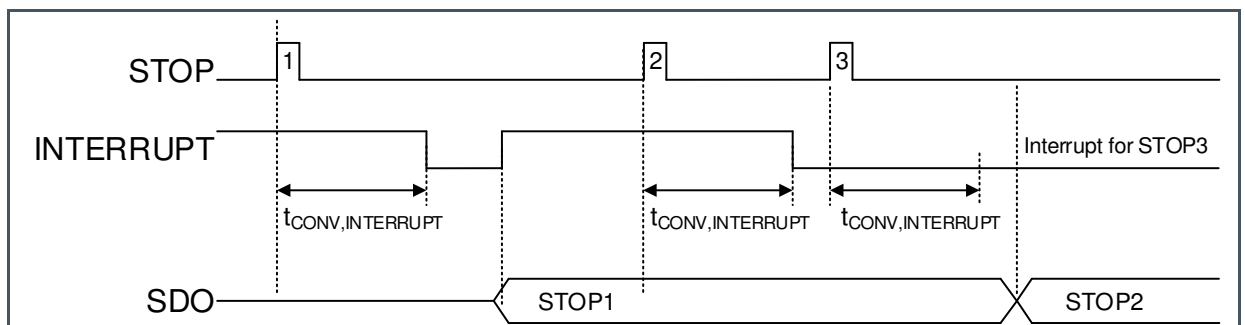
**Figure 48:**  
**Maximum Time Differences**

REF_INDEX_BITWIDTH	Mode	Maximum Readout Hexadecimal	Maximum Readout Decimal	Max. Time Difference with		
				$f_{REFCLK} = 2 \text{ MHz}$	$f_{REFCLK} = 5 \text{ MHz}$	$f_{REFCLK} = 10 \text{ MHz}$
0 Bit	SPI	No read-out	No read-out	0.5 $\mu\text{s}$	200 ns	100 ns
8 Bit	SPI	0xFF	255	128 $\mu\text{s}$	51.2 $\mu\text{s}$	25.6 $\mu\text{s}$
16 Bit	SPI	0xFFFF	65335	32 ms	13.0 ms	6.5 ms

## 8.7 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.

**Figure 49:**  
**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. 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.

## 8.8 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.

### 8.8.1 Peak Conversion Rate

The peak input conversion rate is limited by the ability of AS6500 to sample, convert and store stop events in the FIFOs. The maximum peak conversion rate is limited minimal pulse-to-pulse-spacing tPPS 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.

### 8.8.2 Read-Out Rate

If read-out rate is slower than the input conversion rate, then time measurements necessarily are getting lost because the FIFO may be full. 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

### 8.8.3 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.
- 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

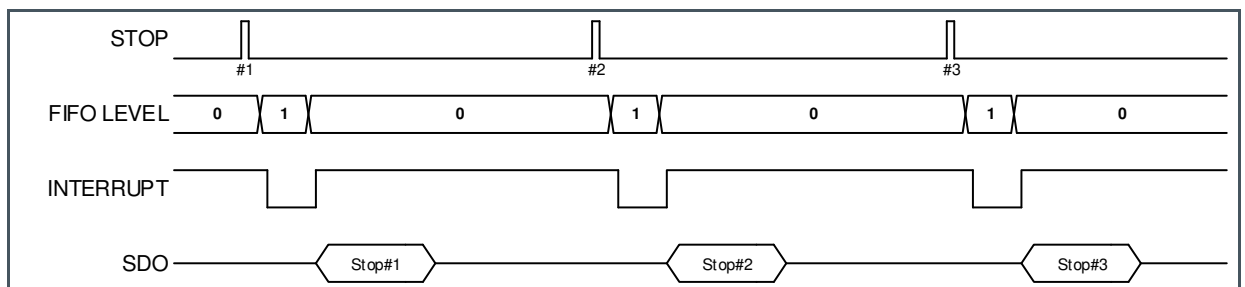
### 8.8.4 FIFOs for Adapting Peak and Average Conversion Rate

Each channel of AS6500 has a First-In-First-Out data buffer (FIFO). Generally, AS6500 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 ( $t_{\text{FIFO\_RESTART}}$ ).

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

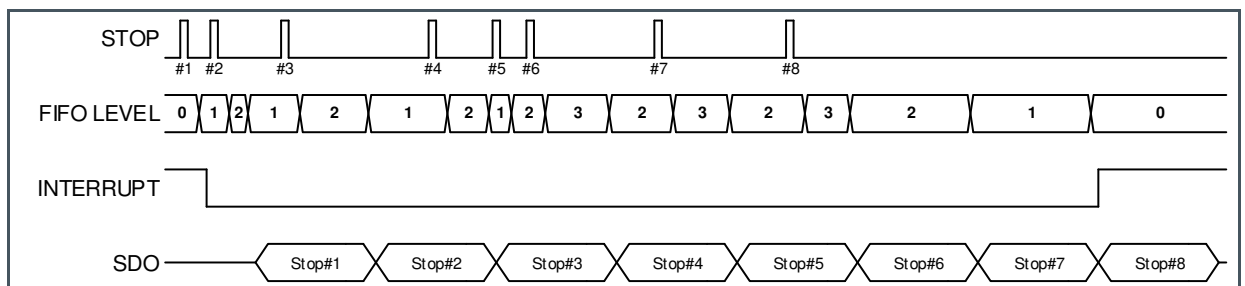
The following figures illustrate the typical dependencies between stop event rate and the read out rate. The INTERRUPT pin indicates that the result is available for read-out through the SPI interface. For SPI a continuous readout is assumed as long as the interrupt is on low level. The interrupt goes back to HIGH when all FIFOs are empty. 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 50:**  
**Input Stop Event Rate Is Lower than the Readout Rate**



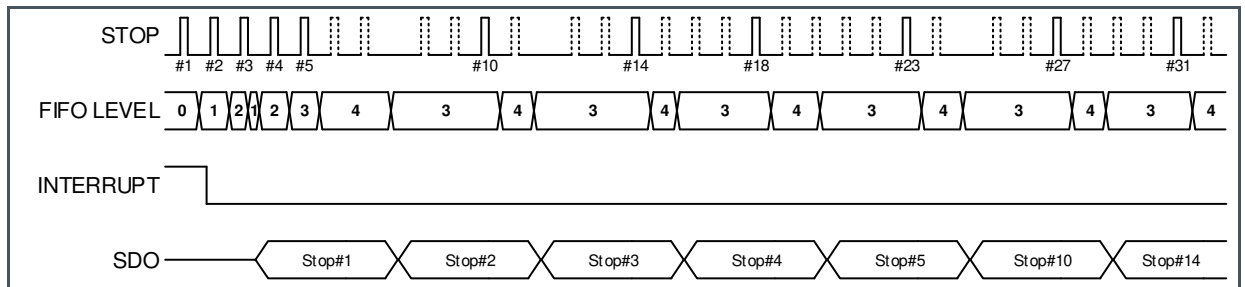
- Enough time for complete readout of first stop before the next stop event arises
- Interrupt goes back to high because the FIFO is empty after read-out
- In this example, no stop events are rejected. All stops are measured and read out

**Figure 51:**  
**Average Stop Event Rate Is Lower, but Peak Stop Event Rate Is Higher than the Readout Rate**



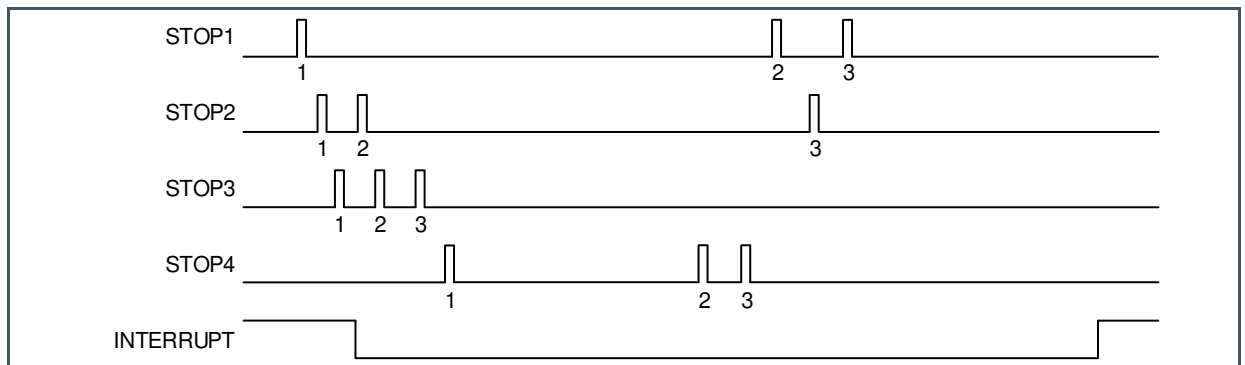
- Stop events during read-out are stored in FIFO
- Stop events buffer up to FIFO LEVEL 3
- In this example, no stop events are rejected. All stops are measured and read out.
- Interrupt goes back to high when all data are readout and the FIFO is empty.
- Maximal FIFO\_DEPTH and HIGH\_RESOLUTION limits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

**Figure 52:**  
**Stop Event Rate Is Higher than the Readout Rate**



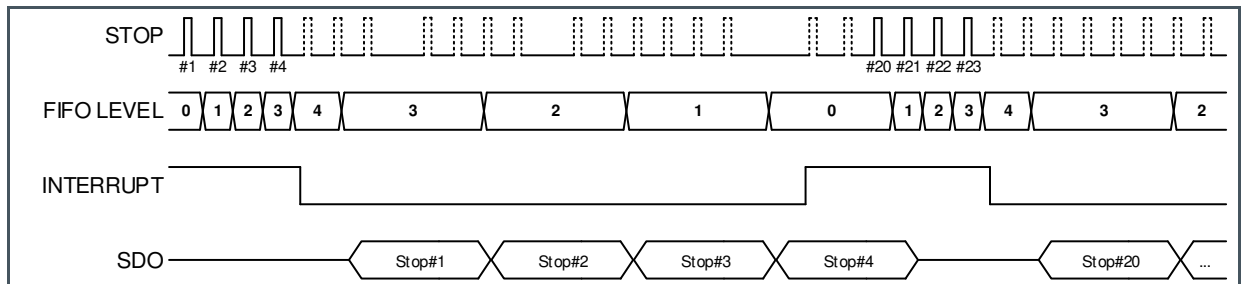
- During read-out stop events (dots) are ignored when FIFO full at FIFO LEVEL 4.
- After reading a result from a full FIFO the next two stops events (dashed) are still ignored but used to restart the FIFO
- Interrupt is always zero because the FIFO never gets empty.

**Figure 53:**  
**Stops On All Four Channels**



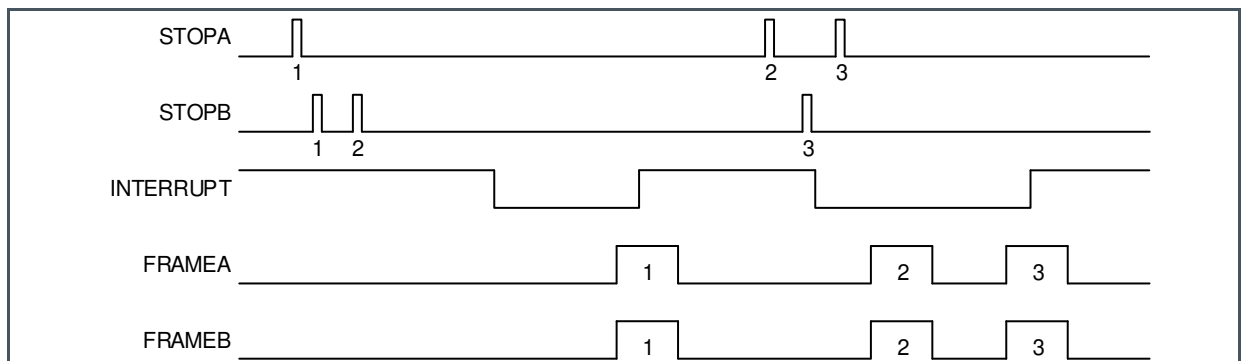
- All four channels are completely independent from each other (COMMON\_FIFO\_READ=0)
- In this example no stop events are rejected, because FIFOs never get full
- Interrupt remains zero as long as at least one FIFO has a valid data, interrupt gets high when all FIFO are empty

**Figure 54:**  
**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 55:**  
**COMMON\_FIFO\_READ**



- 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
- 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 ...)

## 9 Application Information

### 9.1 Configuration Examples

Typical configuration:

```

1  org ROM_ADD_CFG      // AS6500_default_01.cfg saved on 19.11.2018
2  equal 0x40DF0303     // Register 3, 2, 1, 0
3  equal 0xD3C0030D     // Register 7, 6, 5, 4
4  equal 0x0A0013A1     // Register 11, 10, 9, 8
5  equal 0x7DF1CCCC     // Register 15, 14, 13, 12
6  equal 0x00000004     // Register 19, 18, 17, 16
7  equal 0x00000000     // Register 23, 22, 21, 20

```

### 9.2 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 AS6500.

```

8  #include <uProcessor.h>
9  // This is an imaginary header file defined to support this example code
10 // -----
11 // *** uProcessor.h ***
12 // -----
13 // Almost every microprocessor has a specific C++ libraries (header files)
14 // that introduce
15 // specific commands for data readout. Therefore, this imaginary header data
16 // is given to support this example code.
17 // The intention of each virtual function on this header is clearly explained
18 // as follows.
19 // In real projects, instead of these functions, // the user should use the
20 // similar functions of the micro-processor which is used with AS6500.
21 // Virtual functions:
22 //   send_byte_to_SPI( Var1 ); : send Var1 (8 Bits) through the SPI
23 //   read_byte_from_SPI( Var1 ); : read 1 Byte data from SPI and write it
24 //   to Var1
25 // Virtual pin variables:
26 //   GPIO_SSN : Variable (1 Bit) to control the output pin which is
27 //             supposed to be connected the SSN pin of the AS6500
28 //   GPIO_INTERRUPT: Variable (1 Bit) to monitor the input pin which is
29 //             supposed to be connected INTERRUPT pin of the AS6500
30 // -----
31 // *** Configuration Registers ***

```

```

32 // -----
33 const char config_register[16] = {03, 03, DF, 40, 0D, 03, C0, D3, A1, 13, 00,
    0A, CC, CC, F1, 7D, 04, 00, 00, 00};
34 // A typical config settings = { config00, config01, ... , config16 }
35 // -----
36 // *** SPI Opcodes ***
37 // -----
38 const char spiopc_power = 0x30; // opcode for "Power on Reset"
39 const char spiopc_init = 0x18; // opcode for "Initialize Chip and Start
40 // Measurement"
41 const char spiopc_write_config = 0x80; // opcode for "Write Configuration"
42 const char spiopc_read_config = 0x40; // opcode for "Read Configuration"
43 const char spiopc_read_results = 0x60; // opcode for "Read Measure Results"
44 // -----
45 // *** SPI Addresses ***
46 // -----
47 const char reference_index_ch1_byte3 = 8;
48 const char reference_index_ch1_byte2 = 9;
49 const char reference_index_ch1_byte1 = 10;
50 const char stopresult_ch1_byte3 = 11;
51 const char stopresult_ch1_byte2 = 12;
52 const char stopresult_ch1_byte1 = 13;
53 // . . .
54 const char stopresult_ch4_byte3 = 29;
55 const char stopresult_ch4_byte2 = 30;
56 const char stopresult_ch4_byte1 = 31;
57 // -----
58 // *** Other Variables ***
59 // -----
60 int Buffer = 0; // buffer variable used to copy the SPI data
61 char i = 0; // counter for for-loops
62 int reference_index[4] = 0; // reference index data array {Ch1, Ch2, Ch3,
63 // Ch4}
64 int stopresult[4] = 0; // stop result data array {Ch1, Ch2, Ch3, Ch4}
65 bool config_error = false; // flag that indicates if the config
66 // registers are not written correctly
67 // -----
68 // *** Main body of the software ***
69 // -----
70 int main(void)
71 {
72 // -----
73 // *** Power on reset ***
74 // -----
75 GPIO_SSN = 1; // Reset the SPI interface and select the slave device
76 GPIO_SSN = 0;
77

```

```

78     send_byte_to_SPI( spiopc_power ); // Opcode for "Power On Reset" is sent
79                                     // over SPI
80 // -----
81 // *** Writing the configuration registers ***
82 // -----
83     GPIO_SSN = 1; // Reset the SPI interface and select the slave device
84     GPIO_SSN = 0;
85
86     config_error = false;
87
88     send_byte_to_SPI( spiopc_write_config + 00 );
89 // Opcode for "Write Configuration" and config address (00) are sent over SPI
90
91     for ( i = 0; i < 17; i++) // Send all 17 config registers via SPI
92         send_byte_to_SPI( config_register[i] );
93 // -----
94 // *** Verification of config registers ***
95 // -----
96     GPIO_SSN = 1; // Reset the SPI interface and select the slave device
97     GPIO_SSN = 0;
98
99     send_byte_to_SPI( spiopc_read_config + 00 );
100 // Opcode for "Read Configuration" and config address (00) are sent over SPI
101
102     for ( i = 0; i < 17; i++)
103     {
104         read_byte_from_SPI( Buffer ); // read byte from SPI to Buffer variable
105
106         if ( config_register[i] != Buffer ) config_error = true;
107 // if there was a failure in writing the config registers, then the
108 // config_error flag is raised.
109     }
110 // -----
111 // *** Initialize and start the measurement ***
112 // -----
113     if (config_error == false )
114     {
115         GPIO_SSN = 1; // Reset SPI interface and select the slave device
116         GPIO_SSN = 0;
117
118         send_byte_to_SPI( spiopc_init );
119 // Opcode for "Initialize" is sent over SPI. This is required to start
120 // measuring process
121 // -----
122 // End of the configuration settings. After now the time measurement will
123 // start. This code is designed to use SPI to read the measurement data from
124 // AS6500.

```

```

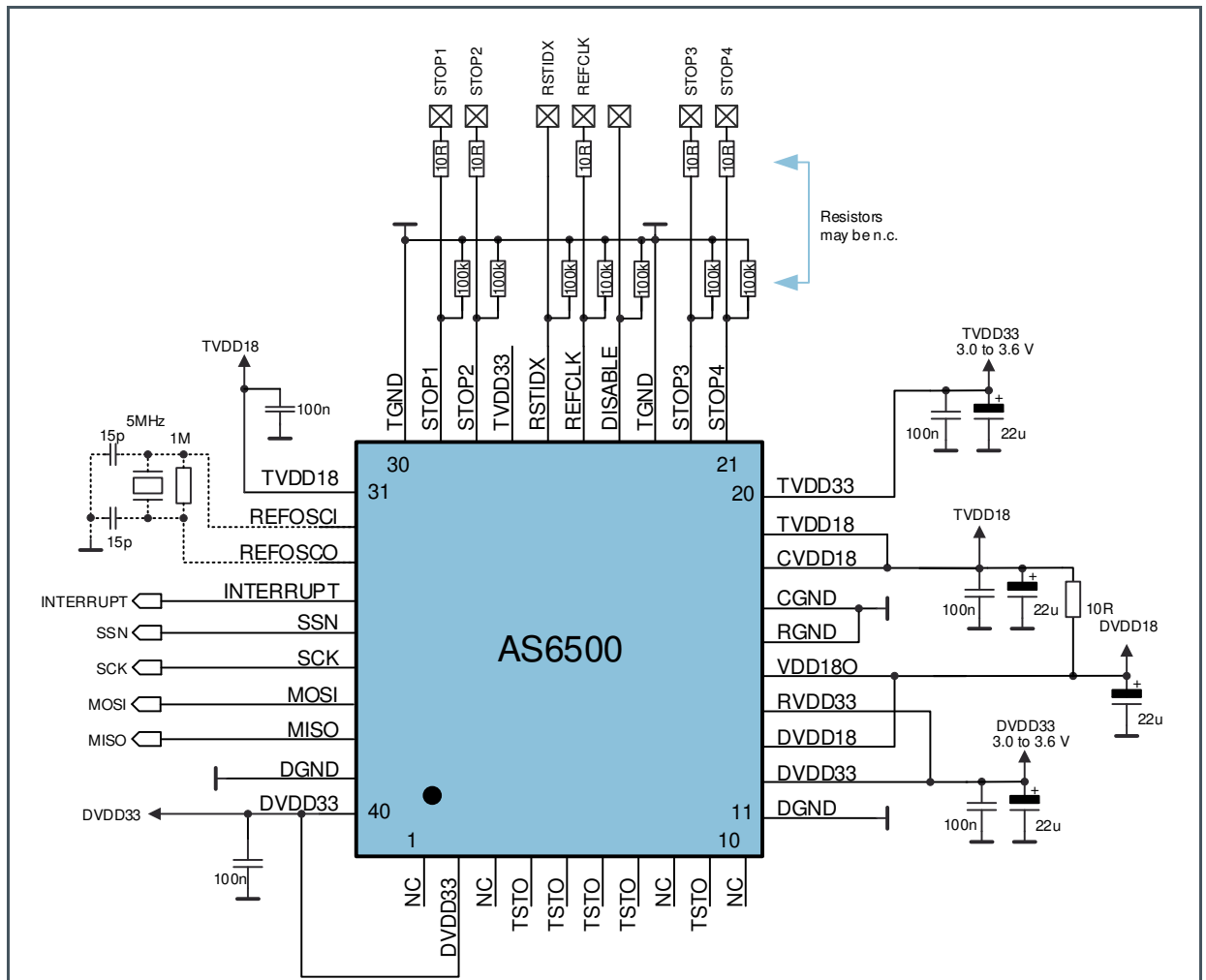
125 // -----
126 // *** Readout of measurement data via SPI ***
127 // -----
128     while( GPIO_INTERRUPT != 0 ); // wait till the Interrupt pin is low
129
130     GPIO_SSN = 1;    // Reset SPI interface and select the slave device
131     GPIO_SSN = 0;
132
133     send_byte_to_SPI( spiopc_read_results + reference_index_ch1_byte3 );
134     // Opcode for "Read Result" and data address are sent
135     for ( i = 0; i < 4; i++)
136     {
137         read_byte_from_SPI( Buffer ); // read one byte from SPI to Buffer
138         reference_index[i] = reference_index[i] + ( Buffer << 16 );
139     // Data is shifted 16 Bits to the left and added to the reference_index
140         read_byte_from_SPI( Buffer ); // read one byte from SPI to Buffer
141         reference_index[i] = reference_index[i] + ( Buffer << 8 );
142     // Data is shifted 8 Bits to the left and added to the reference_index
143         read_byte_from_SPI( Buffer ); // read one byte from SPI to Buffer
144         reference_index[i] = reference_index[i] + Buffer;
145     // Data is directly added to reference_index
146     // The complete reference index (3 Bytes) has been received.
147         read_byte_from_SPI( Buffer ); // Same process as reference_index
148         stopresult[i] = stopresult[i] // is repeated for stop results
149             + ( Buffer << 16 );
150
151         read_byte_from_SPI( Buffer );
152         stopresult[i] = stopresult[i] + ( Buffer << 8 );
153
154         read_byte_from_SPI( Buffer );
155         stopresult[i] = stopresult[i] + Buffer;
156     // The complete stop result (3 Bytes) has been received
157     }
158     // In this point the software has obtained
159     // the reference_index and stopresult data for all channels,
160     // the rest of the codes should be designed depending on the user's
161     // application.
162     // . . .
163 }
164 // . . .
165 }
    
```

## 9.3 Schematic

The following figure shows a typical circuit with power supply and line termination.



**Figure 56:**  
**Schematics for AS6500**



## 9.4 External Components

**Supply Decoupling:** AS6500 provides 6 power supply domains. Careful buffering is recommended. Small decoupling capacitors (e.g. 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.

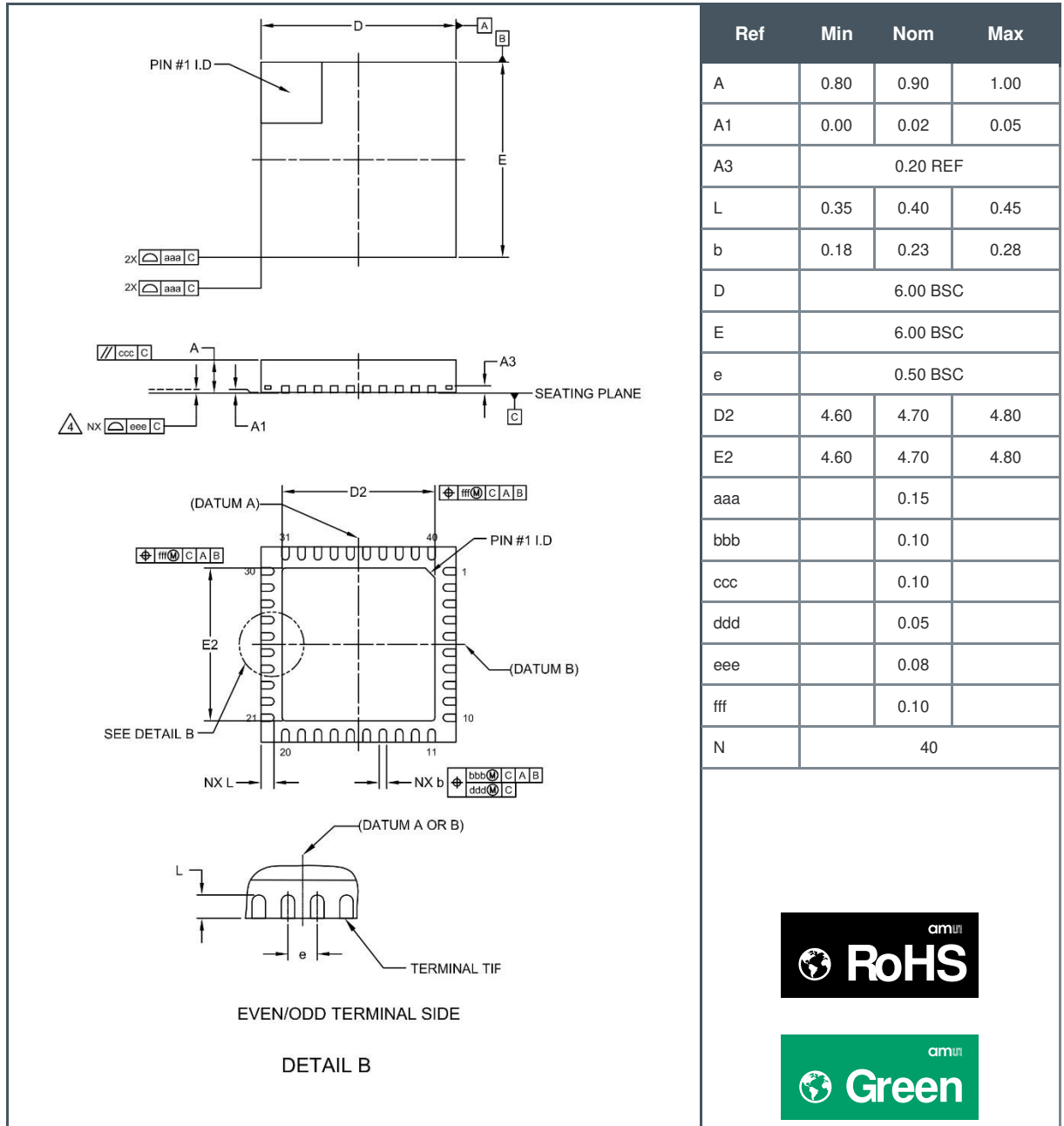
It is recommended to use separate supplies for time-analog (TVDD33) and digital (DVDD33, RVDD33) supply pins to isolate digital switching noise from sensitive circuitry. In case only a single (digital) supply is available, it should be routed to DVDD33 and RVDD33. It can then be tapped and isolated with a resistor (10 Ohm) 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.

**Signal Lines:** 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 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 (10 Ohm) in single ended CMOS signal lines further help to damp reflections.

# 10 Package Drawings & Markings

Figure 57:  
QFP48 Package Outline Drawing



- (1) All dimensions are in millimeters. Angles in degrees.
- (2) Dimensioning and tolerancing conform to ASME Y14.5M-1994.
- (3) N is the total number of terminals.
- (4) This package contains no lead (Pb).
- (5) This drawing is subject to change without notice.

Figure 58:  
QFP48 Package Marking/Code



YY	Manufacturing Year
WW	Manufacturing Week
X	Assembly Plant Identifier
ZZ	Assembly Traceability Code
@	Sublot Identifier

---

## 11 Reel Information

---

The device will be shipped in a JEDEC 7" tape.