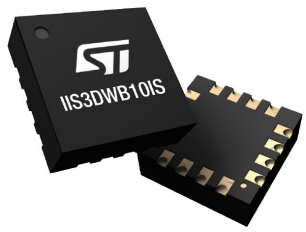
IIS3DWB10IS

Datasheet

Ultrawide bandwidth, low-noise, 3-axis digital vibration sensor with ISPU - intelligent sensor processing unit



LGA-16L
4.5 x 4.5 x 1.5 mm

Product status link
IIS3DWB10IS

Product summary	
Order code	IIS3DWB10ISTR
Temperature range [°C]	-40 to +125
Package	LGA-16L
Packing	Tape and reel

Product resources
AN6497 (device application note)
TN0018 (handling, mounting, and soldering guidelines)

Product label
LONGEVITY 10 YEARS COMMITMENT

Features

- 3-axis vibration sensor with digital output
- User-selectable full scale: ±50 / ±100 / ±200 *g*
- Ultrawide and flat frequency response range: from DC to above 10 kHz (±3 dB point)
- Equalization and filtering guarantee a flat and alias-free frequency response
- Ultralow noise density: down to 35 µ*g*/√Hz
- Embedded ISPU (intelligent sensor processing unit): ultralow-power, high-performance programmable core to execute real-time signal processing and edge AI algorithms
- High stability of the sensitivity over temperature and against mechanical shocks
- Supports accurate external clock input in order to enable precise synchronization of an array of sensors and to improve ODR stability
- Extended temperature range from -40 to +125°C
- Multiple operating modes
  - Continuous mode (CM)
    - 40 / 80 kHz ODR @ >10 kHz bandwidth
    - 2.5 / 5 / 10 / 20 kHz ODR @ ODR/2 bandwidth
  - Burst mode (BM)
    - Configurable burst of acquisitions in CM mode triggered by the ISPU, FIFO, internal timer, or external conditions
- Low power: 1.9 mA with one axis active, 4.1 mA with all three axes active
- SPI serial interface / MIPI I3C® serial interface
- Embedded FIFO: 2048 slots with configurable batching for accelerometer, temperature sensor, and ISPU data
- Embedded temperature sensor
- Embedded self-test
- Supply voltage: 1.7 V to 3.6 V
- Compact package: LGA 4.5 x 4.5 x 1.5 mm 16-lead with wettable flanks
- ECOPACK and RoHS compliant

Applications

- Vibration monitoring
- Condition monitoring
- Predictive maintenance
- Test and measurements

Description

The IIS3DWB10IS is a system-in-package featuring a 3-axis digital vibration sensor with low noise over an ultrawide and flat frequency range. The wide bandwidth, low noise, very stable and repeatable sensitivity, along with the capability of operating over an extended temperature range (up to +125°C) and up to ±200 *g* acceleration full scale, make the device particularly suitable for vibration monitoring in any industrial application, including those in the harshest and most demanding environments.

The IIS3DWB10IS has a selectable full-scale acceleration range of ±50 / ±100 / ±200 *g* and is capable of measuring accelerations with a bandwidth above 10 kHz. A 2048-slot first-in, first-out (FIFO) buffer is integrated in the device to batch data from the integrated accelerometer, temperature sensor, or ISPU as well as to avoid any data loss and to limit intervention from the host processor.

The IIS3DWB10IS embeds a new ST category of processing, ISPU (intelligent sensor processing unit) to support real-time applications that rely on sensor data. The ISPU is an ultralow-power, high-performance programmable core that can execute real-time signal processing and edge AI algorithms. The ISPU can be programmed in C code and can rely on an ecosystem with libraries and tools/IDE from ST and 3rd party.

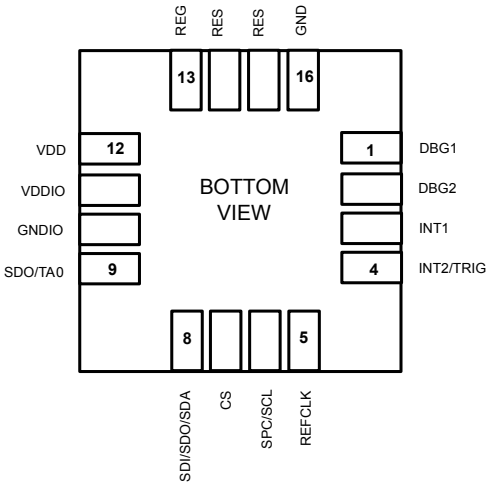
Its optimized ultralow-power hardware circuitry for real-time execution of the algorithms is a state-of-the-art feature for any wireless sensor node for Industry 5.0.

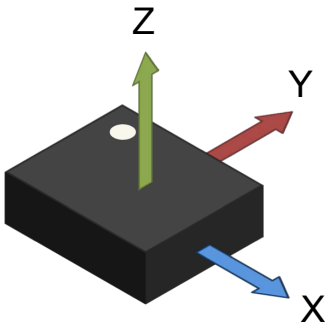
The MEMS sensor module family from ST leverages the robust and mature manufacturing processes already used for the production of micromachined accelerometers and gyroscopes to serve automotive, industrial, and consumer markets. The sensing elements are manufactured using ST's proprietary micromachining process, while the embedded IC interfaces are developed using CMOS technology.

The IIS3DWB10IS has a self-test capability, which allows checking the functioning of the sensor in the final application. The IIS3DWB10IS is available in a 16-lead plastic, land grid array (LGA) package and is guaranteed to operate over an extended temperature range from -40°C to +125°C.

1 Pin description

Figure 1. Pin connections





Direction of detectable acceleration (top view)

1.1 Pin connections

Table 1. Pin description

Pin#	Name	Function
1	DBG1[1]	During normal operation, this pin must be tied to GND. When the JTAG ISPU debug interface is enabled, it is the TCK pin of the JTAG interface.
2	DBG2[1]	During normal operation, this pin must be electrically unconnected but soldered to the PCB. When the JTAG ISPU debug interface is enabled, it is the TDO pin of the JTAG interface.
3	INT1	During normal operation, this pin is the programmable interrupt #1. When the JTAG[1] ISPU debug interface is enabled, it is the TDI pin of the JTAG interface
4	INT2 TRIG	During normal operation, this pin is the programmable interrupt #2 or external trigger input When the JTAG[1] ISPU debug interface is enabled, it is the TMS pin of the JTAG interface
5	REFCLK	Reference clock Connect to VDDIO or GNDIO if not used
6	SPC SCL	SPI serial port clock (SPC) I3C serial clock (SCL)
7	CS	SPI chip select
8	SDI SDO SDA	SPI serial data input (SDI) 3-wire interface serial data output (SDO) I3C serial data (SDA)
9	SDO TA0	SPI 4-wire interface serial data output (SDO) I3C least significant bit of the static address (TA0)
10	GNDIO	Ground for I/O pins
11	VDDIO	Power supply for I/O pins [2]
12	VDD	Power supply[2]
13	REG	Capacitance connection pin for internal regulator. A 100 nF ceramic capacitor to ground should be connected to this pin.
14	RES	Reserved This pin must soldered to the PCB. It is advisable, but not mandatory, to connect it either to GND/GNDIO or VDD/VDDIO.
15	RES	Reserved This pin must soldered to the PCB. It is advisable, but not mandatory, to connect it either to GND/GNDIO or VDD/VDDIO.
16	GND	Ground

1. *JTAG interface reserved for ST internal use only. Not supported for customer access.*
2. *Recommended 100 nF ceramic capacitor*

2 Module specifications

2.1 Mechanical characteristics

@VDD = 3.0 V, T = +25°C unless otherwise noted.

The product is factory calibrated at 3.0 V. The operational power supply range is from 1.7 V to 3.6 V.

Table 2. Mechanical characteristics

Symbol	Parameter	Test conditions	Min.(1)	Typ.(2)	Max.(1)	Unit
FS	Linear acceleration measurement range			±50		*g*
				±100		
				±200		
So	Linear acceleration sensitivity(3)	@FS = ±50 *g* (20-bit format)	-1%	0.095	+1%	mg/LSB
		@FS = ±100 *g* (20-bit format)		0.191		
		@FS = ±200 *g* (20-bit format)		0.381		
		@FS = ±50 *g* (16-bit format)		1.526		
		@FS = ±100 *g* (16-bit format)		3.052		
		@FS = ±200 *g* (16-bit format)		6.104		
SoDr	Linear acceleration sensitivity change vs. temperature(4)	from -40°C to +125°C	-0.016		+0.016	%/°C
TyOff	Linear acceleration zero-*g* level offset accuracy(5)	T = 25°C		1		*g*
TCOff	Linear acceleration zero-*g* level change vs. temperature(4)			36		mg/°C
Cx	Linear acceleration cross-axis sensitivity(3)	T = 25°C, X to Y axis		±0.2		%
		T = 25°C, X to Z axis		±1.8		
		T = 25°C, Y to Z axis		±0.6		
An	Acceleration noise density	X-axis, @FS = ±50 *g*		35		µg/√Hz
		Y-axis, @FS = ±50 *g*		35		
		Z-axis, @FS = ±50 *g*		50		
		X-axis, @FS = ±100 *g*		45		
		Y-axis, @FS = ±100 *g*		45		
		Z-axis, @FS = ±100 *g*		65		
		X-axis, @FS = ±200 *g*		77		
		Y-axis, @FS = ±200 *g*		77		
		Z-axis, @FS = ±200 *g*		86		
BW	Signal bandwidth	Continuous mode ODR = 40 / 80 kHz ±3 dB point				kHz
		X-axis	10	14.5		
		Y-axis	10	14.5		
		Z-axis	10	11.8		
ODR	Linear acceleration output data rate			2.5		kHz
				5		
				10		
				20		

Symbol	Parameter	Test conditions	Min.(1)	Typ.(2)	Max.(1)	Unit
				40 80		
ODR_ACC	ODR accuracy	@VDD 3.0 V, T = +25°C	-0.65		+0.65	%
ODR_TC	ODR change vs. temperature	@VDD 3.0 V, from -40°C to +125°C	-0.0065		+0.0065	%/°C
F0	Sensor resonant frequency	X-axis		23		kHz
		Y-axis		23		
		Z-axis		17		
Vst	Linear acceleration self-test output change(6)(7)(8)	FS = ±50 *g*, X-axis, Y-axis	4.0		9.0	*g*
		FS = ±50 *g*, Z-axis	1.0		7.0	
Top	Operating temperature range		-40		+125	°C

1. *Min/Max values are based on characterization results at 3σ on a limited number of samples, not tested in production and not guaranteed.*
2. *Typical specifications are not guaranteed.*
3. *Sensitivity values after factory calibration test and trimming.*
4. *Measurements are performed in a uniform temperature setup and they are based on characterization data in a limited number of samples. Not measured in production and not guaranteed.*
5. *Values after factory calibration test and trimming.*
6. *The sign of the linear acceleration self-test output change is defined by the ST_SIGN bit in ST_CTRL (17h - R/W) for all axes.*
7. *The linear acceleration self-test output change is defined with the device in stationary condition as the absolute value of: OUTPUT[LSB] (self-test enabled) - OUTPUT[LSB] (self-test disabled). For the LSB value expressed in mg, refer to Table 2.*
8. *Accelerometer self-test limits are full-scale independent.*

2.2 Electrical characteristics

@VDD = 3.0 V, T = 25°C unless otherwise noted.

Table 3. Electrical characteristics

Symbol	Parameter	Test conditions	Min.(1)	Typ. (2)	Max.(1)	Unit
VDD	Supply voltage		1.7		3.6	V
VDDIO	Power supply for I/O		1.62		VDD + 0.1	V
Idd	Accelerometer supply current	Continuous mode 3 axes enabled		4.1		mA
		Continuous mode 1 axis enabled		1.9		mA
IddPD	Accelerometer supply current during power-down			26		µA
Ton	Turn-on time(3)	ISPU not used		5		ms
V_{IH}(4)	Digital high-level input voltage		0.8*VDDIO			V
V_{IL}(4)	Digital low-level input voltage				0.2*VDDIO	V
V_{OH}(4)	High-level output voltage	I_{OH} = 4 mA	VDDIO - 0.2			V
V_{OL}(4)	Low-level output voltage	I_{OL} = 4 mA			0.2	V
F_{ext_clk}	External reference clock input frequency(5)		16		5120	kHz
Top	Operating temperature range		-40		+125	°C

1. *Min/Max values are based on characterization results at 3σ, not tested in production and not guaranteed.*
2. *Typical specifications are not guaranteed.*
3. *Time to obtain valid data switching from power-down to normal operation. The value specified is valid only if the ISPU is not used, otherwise the boot time of the ISPU should be added and it is firmware-dependent.*
4. *Guaranteed by design characterization and not tested in production.*
5. *The duty cycle of the external reference clock should be 50% ±5% tolerance. Rising and falling edges should be ≤20 ns.*

2.3 Temperature sensor characteristics

@VDD = 3.0 V, T = 25°C unless otherwise noted.

Table 4. Temperature sensor characteristics

Symbol	Parameter	Test condition	Min.(1)	Typ.(2)	Max.(1)	Unit
TODR	Temperature refresh rate			52		Hz
Toff	Temperature offset			1		°C
TSen	Temperature sensitivity			200		LSB/°C
T_acc	Temperature accuracy	from 50°C to 125°C	-3		+3	°C
T_ADC_res	Temperature ADC resolution			16		bit
Top	Operating temperature range		-40		+125	°C

1. *Min/Max values are based on characterization results at 3σ on a limited number of samples, not tested in production and not guaranteed.*
2. *Typical specifications are not guaranteed.*

2.4 Communication interface characteristics

2.4.1 SPI - serial peripheral interface

Subject to general operating conditions for VDD and Top. @ VDDIO = 3.0 V, T = 25°C unless otherwise noted.

Table 5. SPI target timing values

Symbol	Parameter	Value(1)			Unit
		Min	Typ	Max	
$f_{c(SPC)}$	SPI clock frequency			25	MHz
$t_{c(SPC)}$	SPI clock period	40			ns
$t_{high(SPC)}$	SPI clock high	45			
$t_{low(SPC)}$	SPI clock low	45			
$t_{su(CS)}$	CS setup time (mode 3)	5			
	CS setup time (mode 0)	20			
$t_{h(CS)}$	CS hold time (mode 3)	20			
	CS hold time (mode 0)	20			
$t_{su(SI)}$	SDI input setup time	5			
$t_{h(SI)}$	SDI input hold time	15			
$t_{v(SO)}$	SDO valid output time			25	
$t_{dis(SO)}$	SDO output disable time			50	
C_{load}	Bus capacitance			100	pF

1. *Values are evaluated at 25 MHz clock frequency for SPI with both 4 and 3 wires, based on characterization results, not tested in production*

Figure 2. SPI target timing in mode 0

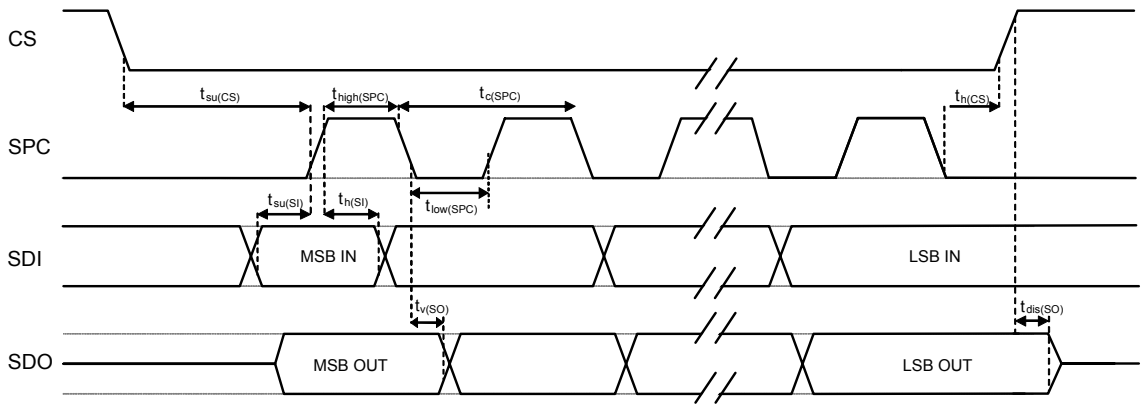
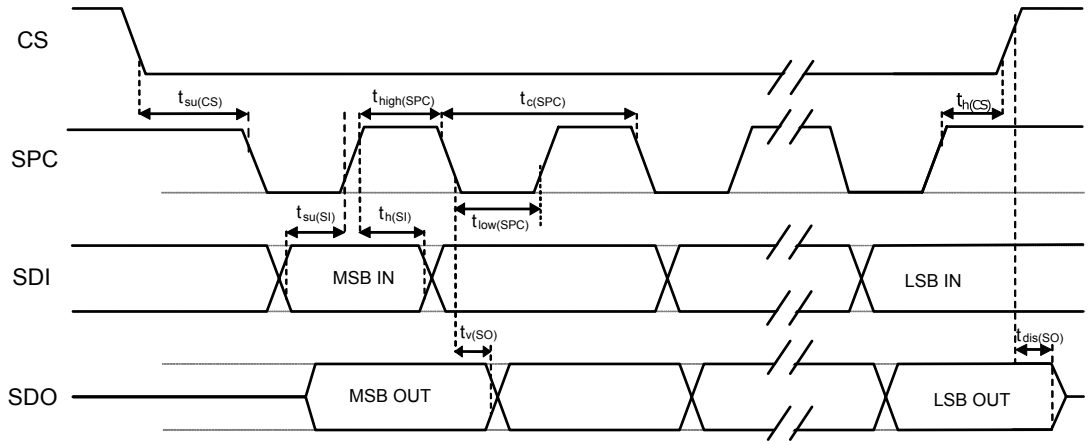


Figure 3. SPI target timing in mode 3



Note: Measurement points are done at 0.2·VDDIO and 0.8·VDDIO for both input and output ports.

2.5 Absolute maximum ratings

Stresses above those listed as “absolute maximum ratings” may cause permanent damage to the device. This is a stress rating only and functional operation of the device under these conditions is not implied. Exposure to maximum rating conditions for extended periods may affect device reliability.

Table 6. Absolute maximum ratings

Symbol	Ratings	Maximum value	Unit
VDD	Supply voltage	-0.3 to +4.8	V
T_{STG}	Storage temperature range	-40 to +150	°C
Sg	Acceleration *g* for 0.2 ms	10,000	*g*
ESD	Electrostatic discharge protection (HBM)	2	kV
Vin	Input voltage on any control pin (including CS, SCL/SPC, SDA/SDI/SDO, SDO/TA0)	-0.3 to VDDIO +0.3	V

Note: Supply voltage on any pin should never exceed 4.8 V.

This device is sensitive to mechanical shock, improper handling can cause permanent damage to the part.

This device is sensitive to electrostatic discharge (ESD), improper handling can cause permanent damage to the part.

3 Digital interfaces

3.1 SPI/I3C interface

The registers embedded inside the IIS3DWB10IS may be accessed through both the I3C and SPI serial interfaces. The latter may be software configured to operate either in 3-wire or 4-wire interface mode. The device is compatible with SPI modes 0 and 3.

The serial interfaces are mapped to the same pins.

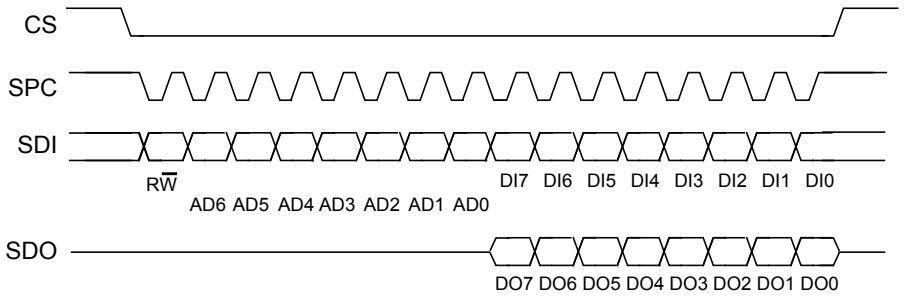
Table 7. Serial interface pin description

Pin name	Pin description
SPC	SPI serial port clock (SPC)
SCL	I3C serial clock (SCL)
CS	SPI chip select
SDI	SPI serial data input (SDI)
SDO	3-wire interface serial data output (SDO)
SDA	I3C serial data (SDA)
SDO	SPI 4-wire serial data output (SDO)
TA0	I3C less significant bit of the device address (TA0)

3.2 SPI bus interface

The SPI on the IIS3DWB10IS is a bus target that allows writing and reading the registers of the device.

Figure 4. Read and write protocol (in mode 3)



CS enables the serial port and it is controlled by the SPI controller. It goes low at the start of the transmission and goes back high at the end. **SPC** is the serial port clock and it is controlled by the SPI controller. It is stopped high when **CS** is high (no transmission). **SDI** and **SDO** are, respectively, the serial port data input and output. Those lines are driven at the falling edge of **SPC** and should be captured at the rising edge of **SPC**.

Both the read register and write register commands are completed in 16 clock pulses or in multiples of 8 in the case of multiple read/write bytes. Bit duration is the time between two falling edges of **SPC**. The first bit (bit 0) starts at the first falling edge of **SPC** after the falling edge of **CS** while the last bit (bit 15, bit 23, ...) starts at the last falling edge of SPC just before the rising edge of **CS**.

bit 0: $\overline{RW}$ bit. When 0, the data DI(7:0) is written into the device. When 1, the data DO(7:0) from the device is read. In the latter case, the chip drives **SDO** at the start of bit 8.

bit 1-7: address AD(6:0). This is the address field of the indexed register.

bit 8-15: data DI(7:0) (write mode). This is the data that is written into the device (MSb first).

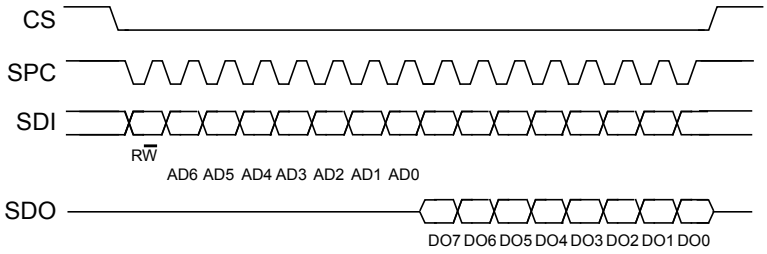
bit 8-15: data DO(7:0) (read mode). This is the data that is read from the device (MSb first).

In multiple read/write commands further blocks of 8 clock periods are added. When the CTRL3 (12h) (IF_INC) bit is 0, the address used to read/write data remains the same for every block. When the CTRL3 (12h) (IF_INC) bit is 1, the address used to read/write data is increased at every block.

The function and the behavior of **SDI** and **SDO** remain unchanged.

3.2.1 SPI read

Figure 5. **SPI read protocol (in mode 3)**



The SPI read command is performed with 16 clock pulses. A multiple byte read command is performed by adding blocks of 8 clock pulses to the previous one.

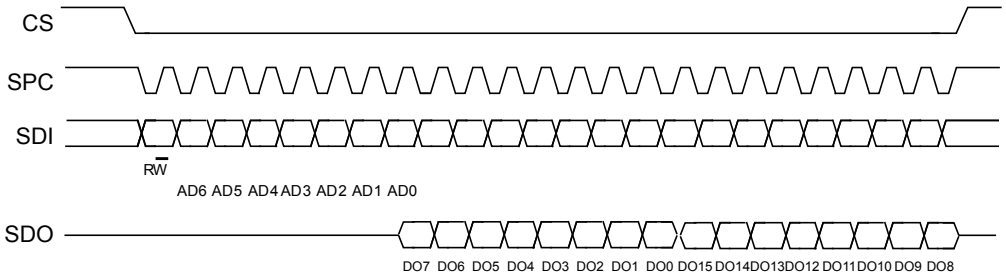
bit 0: READ bit. The value is 1.

bit 1-7: address AD(6:0). This is the address field of the indexed register.

bit 8-15: data DO(7:0) (read mode). This is the data that is read from the device (MSb first).

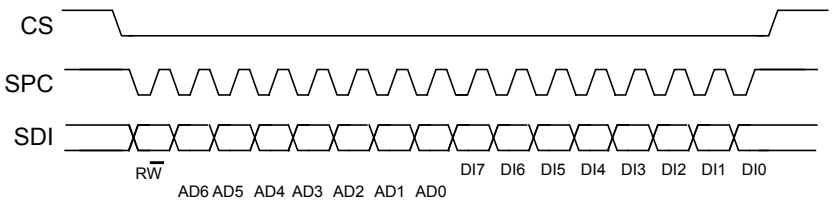
bit 16-...: data DO(...-8). Further data in multiple byte reads.

Figure 6. **Multiple byte SPI read protocol (2-byte example) (in mode 3)**



3.2.2 SPI write

Figure 7. **SPI write protocol (in mode 3)**



The SPI write command is performed with 16 clock pulses. A multiple byte write command is performed by adding blocks of 8 clock pulses to the previous one.

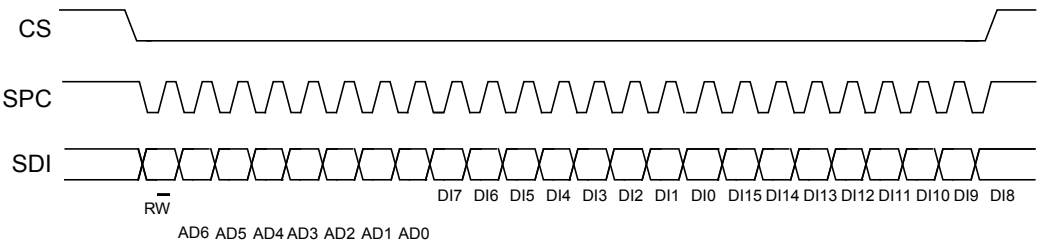
bit 0: WRITE bit. The value is 0.

bit 1-7: address AD(6:0). This is the address field of the indexed register.

bit 8-15: data DI(7:0) (write mode). This is the data that is written inside the device (MSb first).

bit 16-... : data DI(...-8). Further data in multiple byte writes.

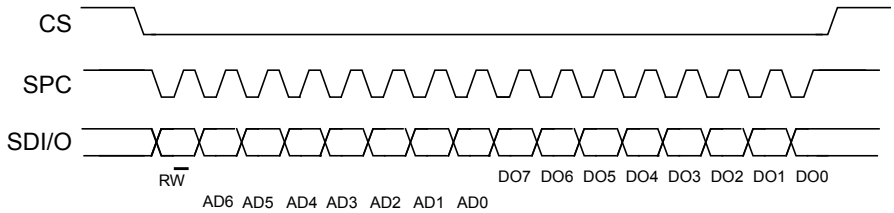
Figure 8. **Multiple byte SPI write protocol (2-byte example) (in mode 3)**



3.2.3 SPI read in 3-wire mode

Enter 3-wire mode by setting the SPI_CTRL (15h - R/W) (SIM) bit equal to 1 (SPI serial interface mode selection).

Figure 9. **SPI read protocol in 3-wire mode (in mode 3)**



The SPI read command is performed with 16 clock pulses:

bit 0: READ bit. The value is 1.

bit 1-7: address AD(6:0). This is the address field of the indexed register.

bit 8-15: data DO(7:0) (read mode). This is the data that is read from the device (MSb first).

A multiple read command is also available in 3-wire mode.

3.3 MIPI I3C® interface

3.3.1 MIPI I3C® target interface

The IIS3DWB10IS interface includes an MIPI I3C® SDR only target interface (compliant with release 1.1 of the specification) with MIPI I3C® SDR embedded features:

- CCC command
- Direct CCC communication (SET and GET)
- Broadcast CCC communication
- Private communications
- Private read and write for single byte
- Multiple read and write
- In-band interrupt request
- Target reset pattern
- Group address
- Asynchronous modes 0 and 1
- Error detection and recovery methods (S0-S6)
- Legacy I²C mode without 50 ns filter

In order to disable the MIPI I3C® block, I3C_disable = 1 must be written in IF_CFG (03h).

Table 8. I3C terminology

Term	Description
Transmitter	The device that sends data to the bus
Receiver	The device that receives data from the bus
Controller	The device that generates clock signals and terminates a transfer
Target	The device addressed by the controller
In-band interrupt	A method whereby a target device emits its address into the arbitrated address header on the I3C bus to notify the controller of an interrupt
Dynamic address	The unique address assigned by the controller for I3C devices during bus initialization
CCC command	A special type of communication that is used to control device behavior collectively or independently
Arbitration	If two devices start transmission at the same time, then arbitration is required to determine bus control.
The two signals associated with the I3C bus are SCL and SDA.	
SCL	I3C serial clock line
SDA	I3C serial data line

Note: The IIS3DWB10IS is natively compliant with the I3C interface and behaves as a pure I3C target.

3.3.2 MIPI I3C® CCC supported commands

The list of MIPI I3C® CCC commands supported by the device is detailed in the following table.

Table 9. MIPI I3C® CCC commands

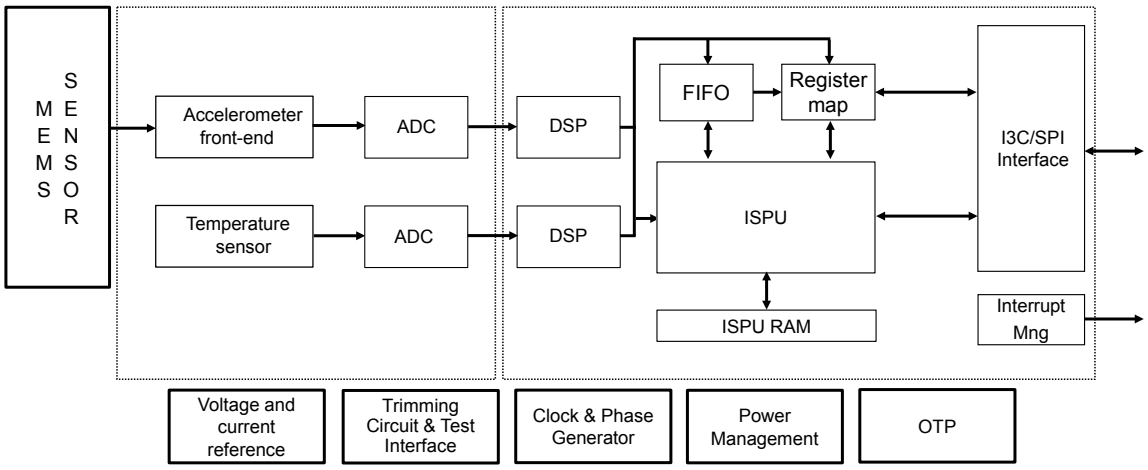
Command	Command code	Default	Description
ENTDAA	0x07		DAA procedure
SETDASA	0x87		Assign dynamic address using static address 0x1A/0x1B depending on SDO pin
ENEC	0x80 / 0x00		Target activity control (direct and broadcast)
DISEC	0x81/ 0x01		Target activity control (direct and broadcast)
ENTAS0	0x82 / 0x02		Enter activity state (direct and broadcast)
SETXTIME	0x98 / 0x28		Timing information exchange
GETXTIME	0x99	0x06 0x00 0x0A 0x92	Timing information exchange
RSTDAA	0x06		Reset the assigned dynamic address (broadcast only)
SETMWL	0x89 / 0x08		Define maximum write length during private write (direct and broadcast)
SETMRL	0x8A / 0x09		Define maximum read length during private read (direct and broadcast)
SETNEWDA	0x88		Change dynamic address
GETMWL	0x8B	0x00 0x08 (2 byte)	Get maximum write length during private write
GETMRL	0x8C	0x00 0x10 0x07 (3 byte)	Get maximum read length during private read
GETPID	0x8D	0x02 0x08 0x00 0x50 0x92 0x0B	SDO = 1
		0x02 0x08 0x00 0x50 0x12 0x0B	SDO = 0
GETBCR	0x8E	0x27 (1 byte)	Bus characteristics register
GETDCR	0x8F	0x41 default	MIPI I3C® device characteristics register
GETSTATUS	0x90	0x00 0x00 (2 byte)	Status register

Command	Command code	Default	Description
GETMXDS	0x94	0x08 0x60	Return max write and read speed
GETCAPS	0x95	0x01 0x91 0x19 0x00	Provide information about device capabilities and supported extended features
SETGRPA	0x9B		Group address assignment command
RSTGRPA	0x2C / 0x9C		Reset the group address
RSTACT	0x9A / 0x2A		Configure target reset action

4 Functionality

4.1 Block diagram

Figure 10. Block diagram

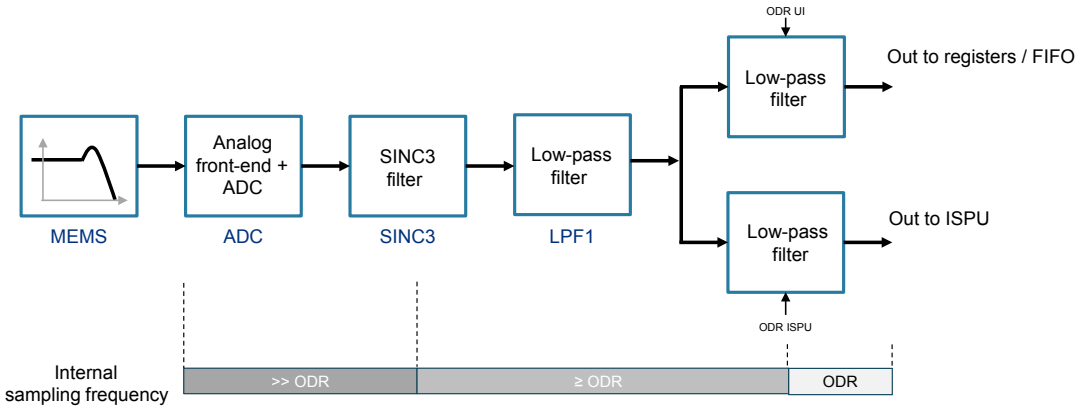


4.2 Accelerometer filtering and processing chain

The IIS3DWB10IS has been specifically designed to provide an accelerometer with an ultrawide bandwidth with a very flat and repeatable frequency response in the pass band and a very high attenuation in the stop band so as to virtually eliminate any frequency aliasing.

The following figure illustrates the embedded filtering and processing chain that make it possible.

Figure 11. Accelerometer filtering and processing chain



In the signal chain, the LPF1 can be configured by acting on the ST_CTRL (17h - R/W) register. Depending on the configuration of the LPF1 and the ODR of the sensor, the frequency response of the sensor can be shaped according to the needs. Examples of frequency response in different LPF1/ODR settings are indicated in the following Section 4.2.1.

4.2.1 Frequency response depending on LPF1 and ODR configuration

By changing the configuration of the low-pass filter LPF1 (register ST_CTRL (17h - R/W)) and of the ODR (register CTRL1 (10h - R/W)), it is possible to adapt, at any moment, the frequency response of the sensors to the application requirements.

The following table details the cutoff frequency (-3dB) of the sensor depending on the configuration of the low-pass filter LPF1 (register ST_CTRL (17h - R/W)) and of the ODR (register CTRL1 (10h - R/W)).

Table 10. Cutoff frequency (configuration of low-pass filter and ODR)

ODR (kHz)	ODR_XL_[3:0]	LPF1	LPF1_CFG_[3:0]	BW XY (kHz) - TYP(1)	BW Z (kHz) - TYP(1)
80	0111	0	0000	17.48	13.63
80	0111	1	0001	15.77	13.15
80	0111	2	0010	13.91	12.42
80	0111	3	0011	11.62	11.07
80	0111	4	0100	9.16	9.08
80	0111	5	0101	6.81	6.84
80	0111	6	0110	4.52	4.54
80	0111	7	0111	2.55	2.55
40	0110	0	0000	14.83	12.59
40	0110	1	0001	13.67	12.16
40	0110	2	0010	12.39	11.53
40	0110	3	0011	10.65	10.34
40	0110	4	0100	8.64	8.60
40	0110	5	0101	6.58	6.61
40	0110	6	0110	4.45	4.47
40	0110	7	0111	2.54	2.54
20	0101	1	0001	8.65	8.60
20	0101	2	0010	8.46	8.43
20	0101	3	0011	7.95	7.96
20	0101	4	0100	7.05	7.08
20	0101	5	0101	5.82	5.85
20	0101	6	0110	4.20	4.22
20	0101	7	0111	2.49	2.50
10	0100	4	0100	4.30	4.32
10	0100	5	0101	4.07	4.09
10	0100	6	0110	3.45	3.46
10	0100	7	0111	2.34	2.34
5.0	0011	6	0110	2.14	2.14
5.0	0011	7	0111	1.89	1.89
2.5	0010	7	0111	1.11	1.11

1. *Frequency response and cutoff frequency (-3 dB) determined by CAD simulation assuming TYP model of the device. Not measured on the device and not guaranteed.*

Frequency response at ODR 80 kHz

Figure 12. Frequency response - X,Y-axis: ODR = 80 kHz LPF1 = 0

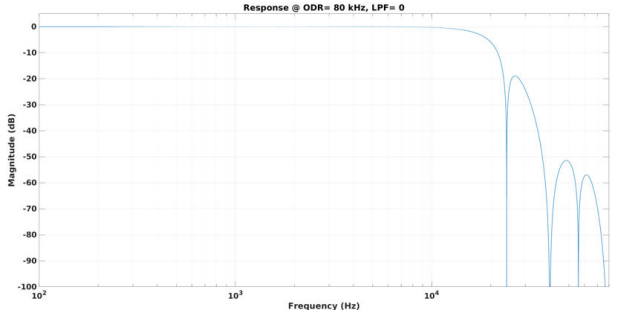


Figure 13. Frequency response - X,Y-axis: ODR = 80 kHz LPF1 = 1

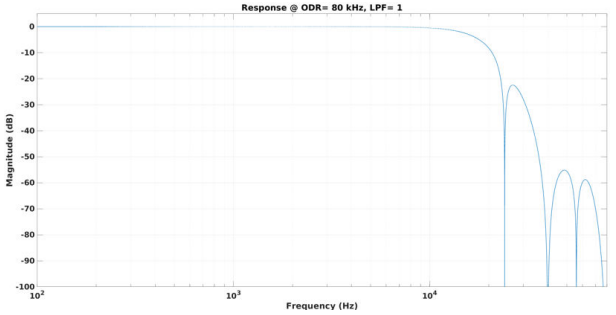


Figure 14. Frequency response - Z-axis: ODR = 80 kHz LPF1 = 0

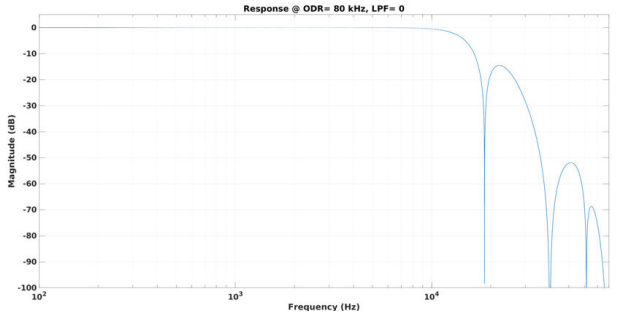


Figure 15. Frequency response - Z-axis: ODR = 80 kHz LPF1 = 1

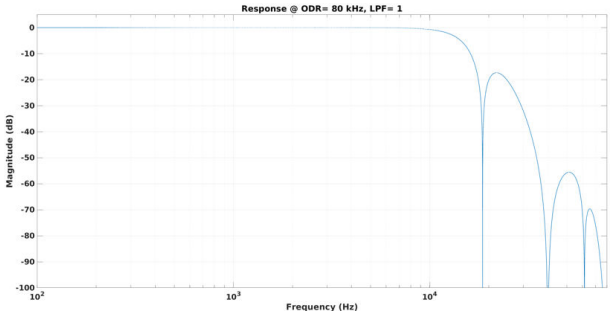


Figure 16. Frequency response - X,Y-axis: ODR = 80 kHz LPF1 = 3

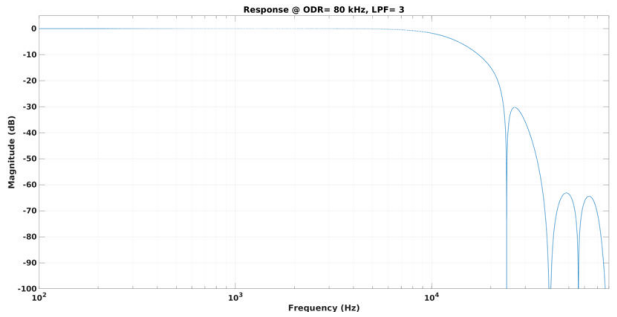


Figure 17. Frequency response - X,Y axis: ODR = 80 kHz LPF1 = 6

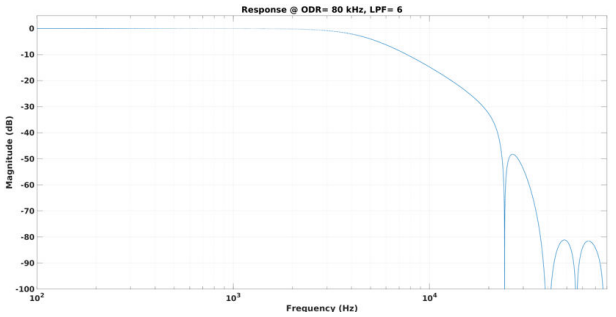


Figure 18. Frequency response - Z-axis: ODR = 80 kHz LPF1 = 3

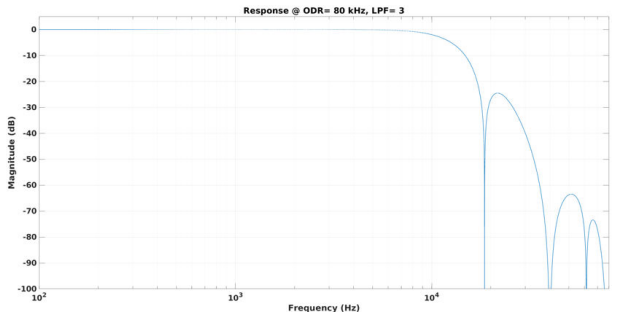
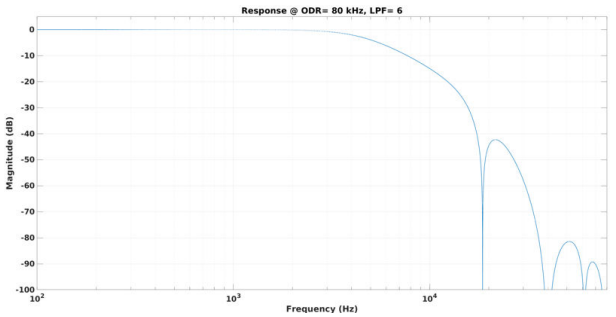


Figure 19. Frequency response - Z-axis: ODR = 80 kHz LPF1 = 6



Frequency response at ODR 40 kHz

Figure 20. Frequency response - X,Y axis: ODR = 40 kHz LPF1 = 0

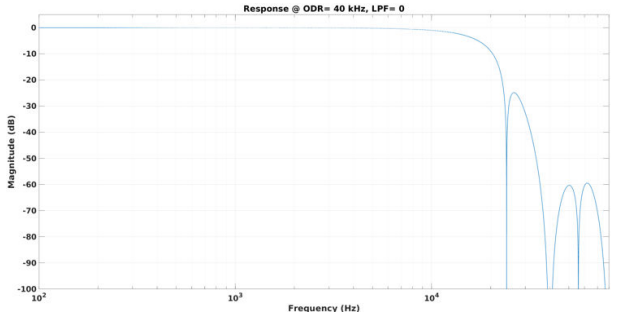


Figure 21. Frequency response - X,Y axis: ODR = 40 kHz LPF1 = 1

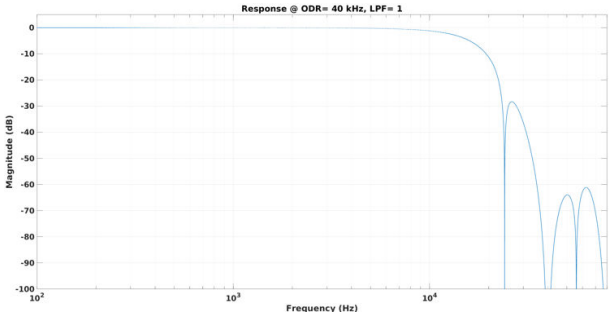


Figure 22. Frequency response - Z-axis: ODR = 40 kHz LPF1 = 0

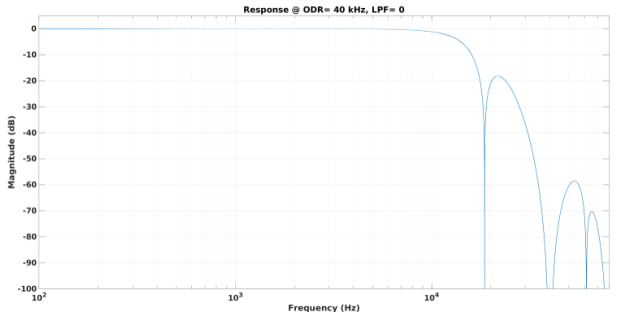


Figure 23. Frequency response - Z-axis: ODR = 40 kHz LPF1 = 1

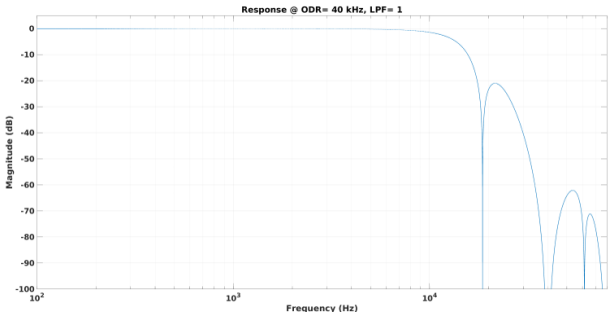


Figure 24. Frequency response - X,Y-axis: ODR = 40 kHz LPF1 = 3

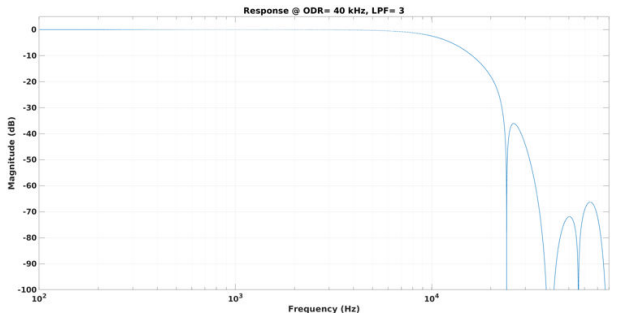


Figure 25. Frequency response - X,Y-axis: ODR = 40 kHz LPF1 = 5

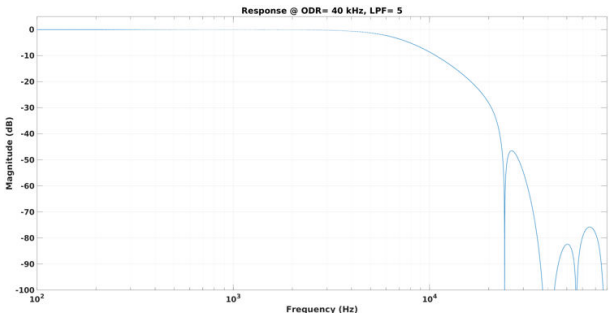


Figure 26. Frequency response - Z-axis: ODR = 40 kHz LPF1 = 3

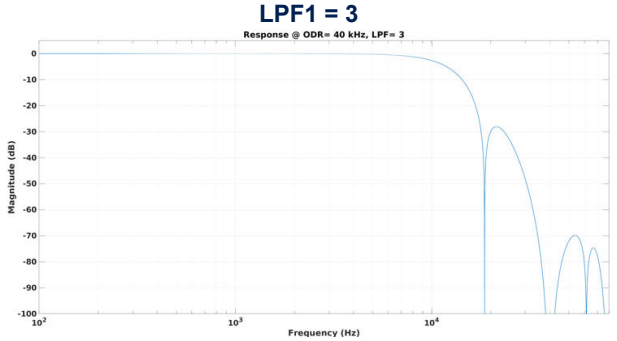
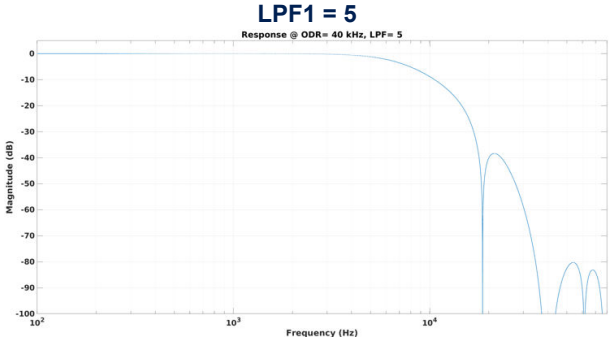


Figure 27. Frequency response - Z-axis: ODR = 40 kHz LPF1 = 5



Frequency response at ODR 20 kHz

Figure 28. Frequency response - X,Y-axis: ODR = 20 kHz LPF1 = 1

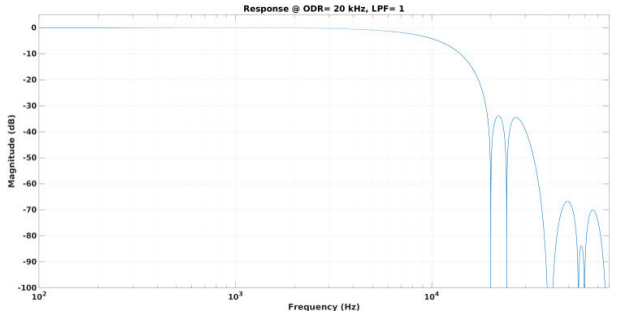


Figure 29. Frequency response - X,Y-axis: ODR = 20 kHz LPF1 = 4

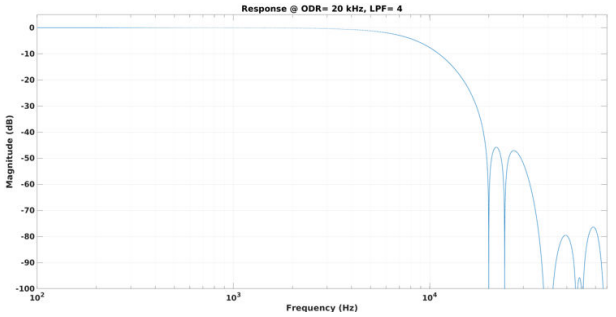


Figure 30. Frequency response - Z-axis: ODR = 20 kHz LPF1 = 1

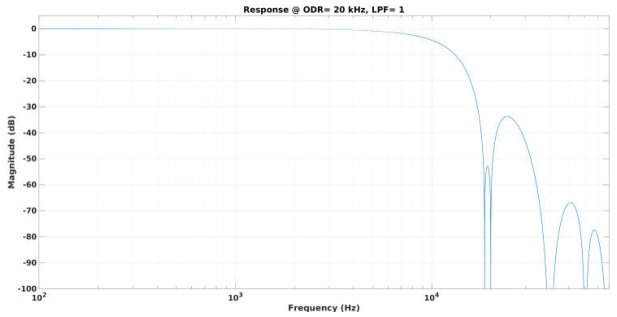


Figure 31. Frequency response - Z-axis: ODR = 20 kHz LPF1 = 4

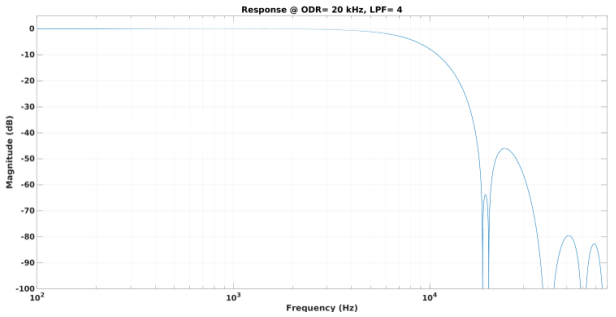


Figure 32. Frequency response - X,Y-axis: ODR = 20 kHz LPF1 = 5

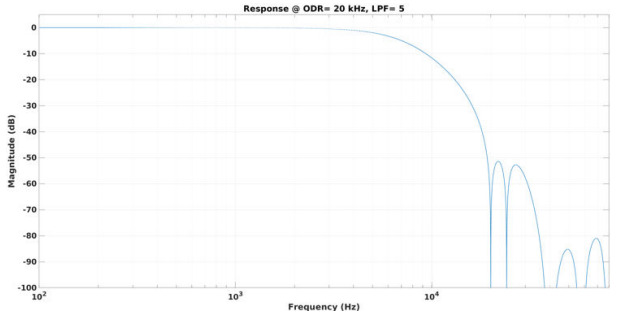


Figure 33. Frequency response - X,Y-axis: ODR = 20 kHz LPF1 = 7

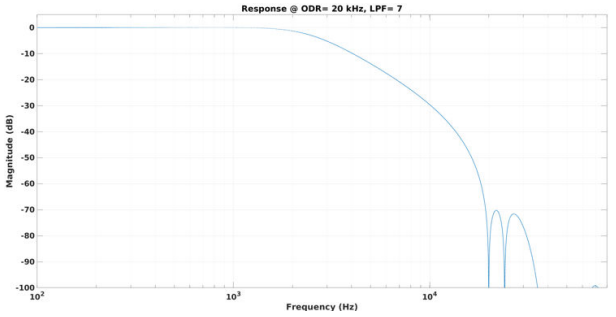


Figure 34. Frequency response - Z-axis: ODR = 20 kHz LPF1 = 5

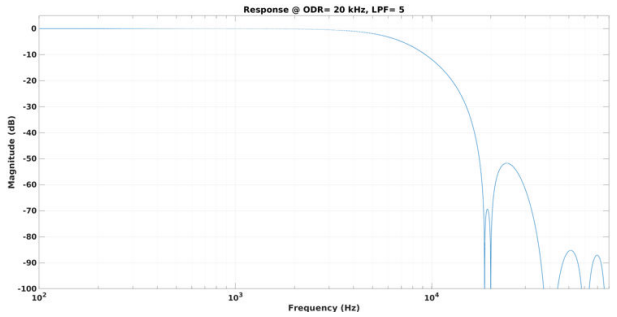
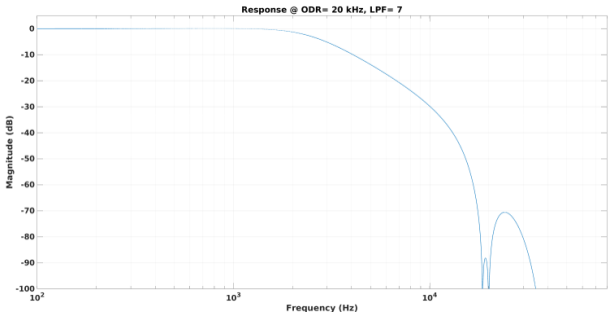


Figure 35. Frequency response - Z-axis: ODR = 20 kHz LPF1 = 7



Frequency response at ODR 10 kHz

Figure 36. Frequency response - X,Y-axis: ODR = 10 kHz LPF1 = 4

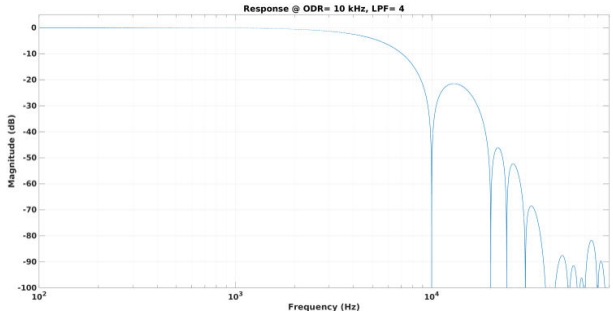


Figure 37. Frequency response - X,Y-axis: ODR = 10 kHz LPF1 = 7

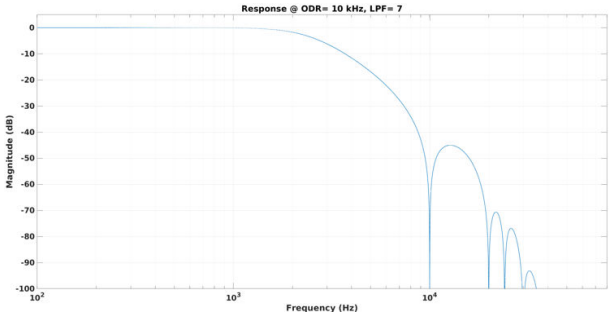


Figure 38. Frequency response - Z-axis: ODR = 10 kHz LPF1 = 4

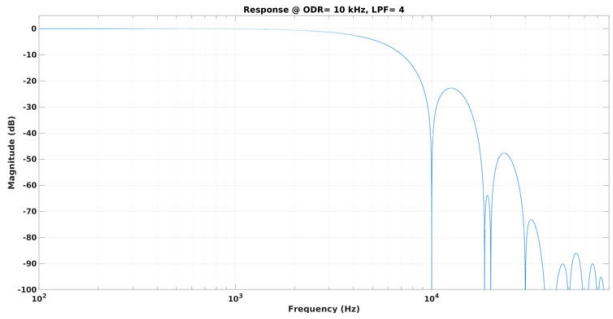
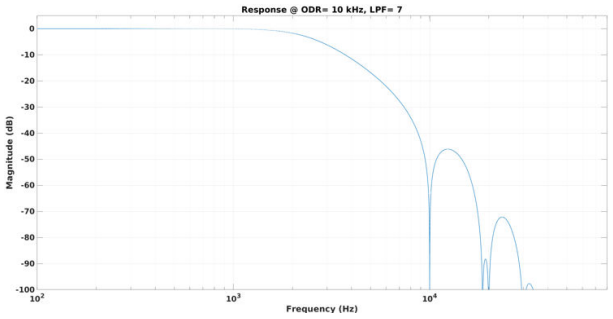


Figure 39. Frequency response - Z-axis: ODR = 10 kHz LPF1 = 7



Frequency response at ODR 5 kHz

Figure 40. Frequency response - X,Y-axis: ODR = 5 kHz LPF1 = 6

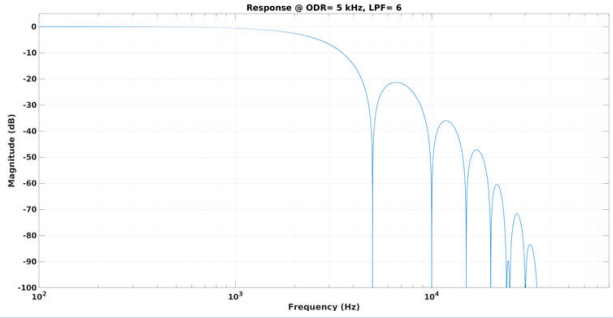


Figure 41. Frequency response - X,Y-axis: ODR = 5 kHz LPF1 = 7

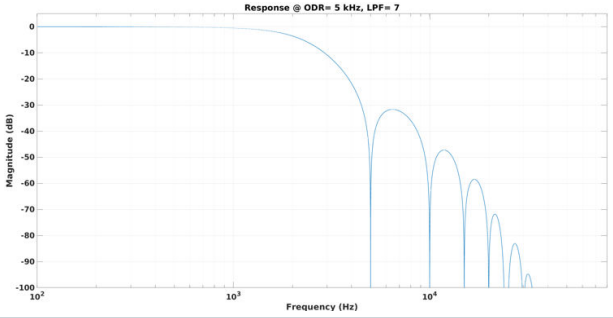


Figure 42. Frequency response - Z-axis: ODR = 5 kHz LPF1 = 6

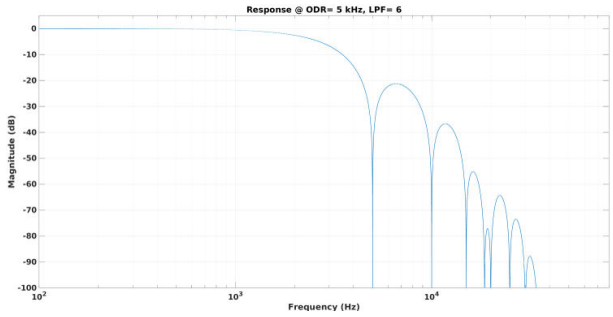
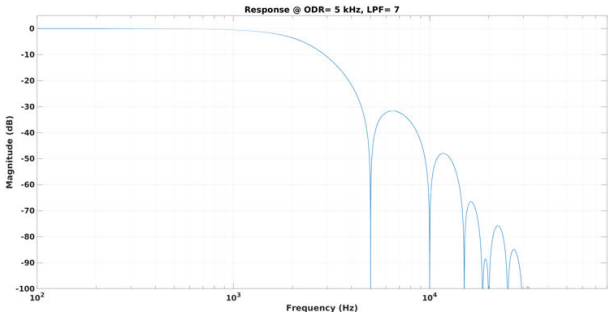


Figure 43. Frequency response - Z-axis: ODR = 5 kHz LPF1 = 7



Frequency response at ODR 2.5 kHz

Figure 44. Frequency response - X,Y-axis: ODR = 2.5 kHz LPF1 = 7

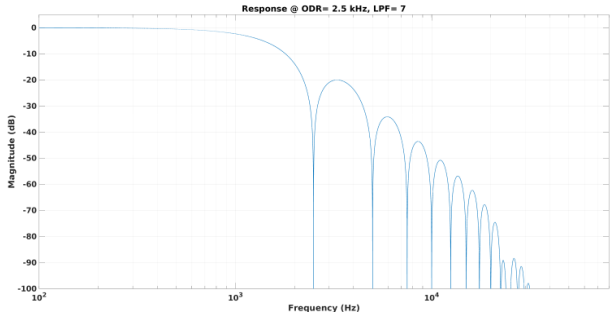
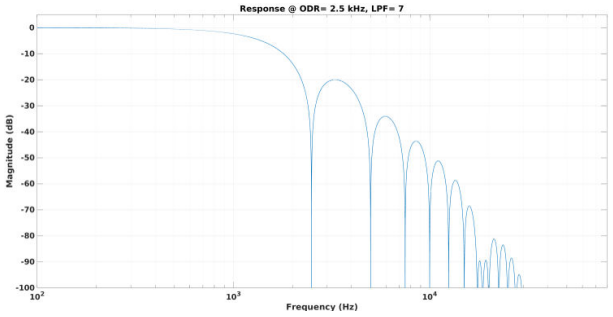


Figure 45. Frequency response - Z-axis: ODR = 2.5 kHz LPF1 = 7



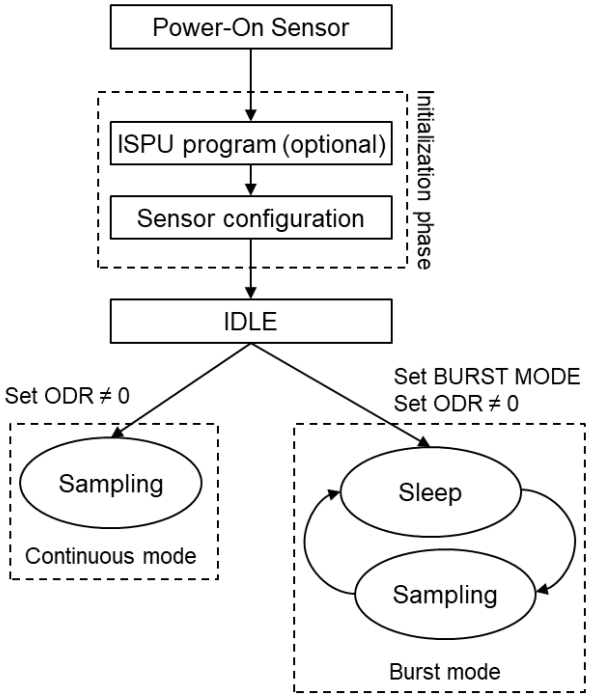
4.3 Operating modes and output data rate (ODR)

The IIS3DWB10IS has two different operating modes:

- Continuous mode (CM)
- Burst mode (BM)

A high-level overview of the operating modes is shown in the following figure.

Figure 46. Startup and operating modes

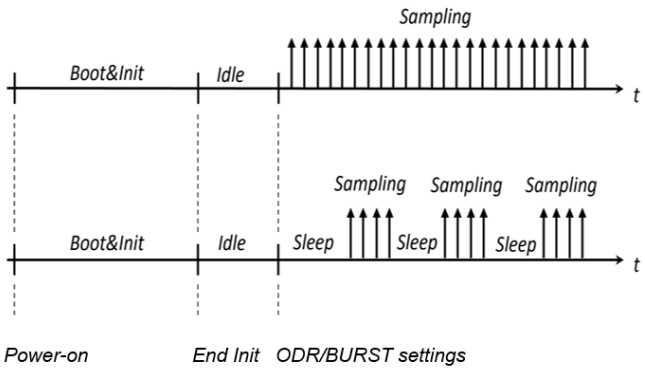


When the sensor powers on, a startup phase automatically sets the device.

Then, it is possible, but optional, to load the ISPU program memory and configure the sensor by accessing the register map over the serial interface.

Once the sensor boot is completed, it goes in idle mode and it is ready to operate according to one of the two possible operating modes: continuous mode or burst mode, depending on the user's settings.

Figure 47. Operating modes: continuous mode and burst mode



It should be noted that there is not a dedicated operating mode for the sampling of data from the temperature sensor, thus the data from this sensor is always available together with the sampling of the accelerometer sensor.

4.3.1 Continuous mode

In continuous mode, the accelerometer signal is continuously sampled (on all 3 axes) at the specified output data rate (ODR).

The available ODRs are listed in Table 11. ODR configuration and are configured through the ODR_XL_[3:0] bits of the CTRL1 (10h - R/W) register.

Table 11. ODR configuration

ODR_XL_3	ODR_XL_2	ODR_XL_1	ODR_XL_0	ODR
0	0	0	0	Idle mode
0	0	0	1	Reserved
0	0	1	0	2.5 kHz
0	0	1	1	5.0 kHz
0	1	0	0	10 kHz
0	1	0	1	20 kHz
0	1	1	0	40 kHz
0	1	1	1	80 kHz
1	x	x	x	Reserved

The ODR can be changed only in idle mode so the proper sequence should be:

1. Set in idle mode (ODR_XL= 0x0).
2. Wait a minimum of 0.45 ms, depending on the eventual end of the ISPU execution or serial interface activity.
3. Set the new ODR (ODR_XL=<your ODR>).
4. The sensor is now armed to sample again.

4.3.2 Burst mode

Burst mode consists of sampling the accelerometer signal in 'burst' fashion.

During a burst of samples, the real sampling ODR is kept high (from 2.5 kSa/s to 80 kSa/s as per the ODR_XL settings in the CTRL1 (10h - R/W) register), but the sensor automatically turns itself on and off, triggered by trigger-on and trigger-off conditions.

During sleep mode, in between the sampling burst, the sensor behaves like idle mode, but ODR_XL in CTRL1 (10h - R/W) is not zero, so it is armed to be fired by a trigger-on condition.

During sampling mode, the sensor behaves like continuous mode until a trigger-off condition puts the sensor back into sleep mode.

Burst mode can be set using the BURST_CFG bit in the CTRL1 (10h - R/W) register and must be done before setting the ODR using the ODR_XL bit in CTRL1 (10h - R/W). The complete list of burst mode capabilities is shown in the following table.

Table 12. Burst configuration

BURST_CFG_3	BURST_CFG_2	BURST_CFG_1	BURST_CFG_0	Mode	Trigger on	Trigger off
0	0	0	0	Continuous mode	-	-
0	0	0	1	Burst mode	User from interface	ISPU
0	0	1	0		User from interface	FIFO
0	1	0	0		Sleep counter	User from interface
0	1	0	1		Sleep counter	ISPU
0	1	1	0		Sleep counter	FIFO
1	0	0	0		External trigger	User from interface
1	0	0	1		External trigger	ISPU
1	0	1	0		External trigger	FIFO
Other values				Idle mode	-	-

The ODR is still defined with the ODR_XL bit in CTRL1 (10h - R/W), as in continuous mode. In fact, the sampling rate during the active phase is the same as continuous mode.

The trigger-on and trigger-off sources are described in the following table.

Table 13. Trigger-on and trigger-off sources

Burst trigger	Source	Comments
User from interface	BURST_FORCE_TRG bit in CTRL3 (12h - R/W)	Active when asserted. Clear before reuse.
ISPU	From firmware running on ISPU	
FIFO	FIFO watermark	It is triggered when the FIFO watermark is reached
Sleep counter	Sleep counter reaching the threshold	Generated when the sleep counter reaches the configured threshold. Refer to Section 4.6: Sleep counter.
External trigger	INT2 pin	The INT2 pin is automatically set as the trigger input (when in burst mode). Active on the rising edge.

The number of samples inside a burst is not defined with a specific parameter but it is determined by the trigger-off event.

For example, a single sample acquisition can be easily done by using the ISPU (the firmware sends a trigger-off after the first sample) or FIFO watermark (set 1 sample batch).

During sleep mode, all the clocks are switched off but no data is lost, nor in the registers, nor in the RAM. The FIFO is always accessible from the serial interface, also in sleep mode.

The RAM of the ISPU is also accessible from the serial interface.

4.4 Startup sequence

To operate the IIS3DWB10IS sensor, the following startup sequence must be executed.

1. Power on the device.
2. Wait 20 ms.
3. Turn on the FIFO or ISPU if needed by the application (refer to FIFO_EN/ISPU_EN in the CTRL3 (12h - R/W) register).
4. Wait 1.5 ms.
5. (optional) Execute the loading procedure of the ISPU memory and all the necessary steps to start the program execution, as described in the application note ANXXXX.
6. Set all sensor registers for configuration.
7. Set operative mode: BURST_CFG in CTRL1 (10h - R/W)
8. Finally, set ODR: ODR_XL in CTRL1 (10h - R/W). This starts continuous mode or waits for the trigger for burst mode.
9. The sensor is ready to sample in about 5 ms. If the ISPU has been programmed, this time is impacted. The sensor starts sampling once the ISPU boot has ended. This time depends on the firmware loaded in the ISPU.

Continuous mode: continuous sampling until the idle command from the interface ODR_XL= 0 in the CTRL1 (10h - R/W) register.

Burst modes: sampling until the trigger-off event, then returns to sleep mode automatically, waiting for the next trigger-on. Refer to Section 4.3: Operating modes and output data rate (ODR) for more details.

4.5 Synchronization management

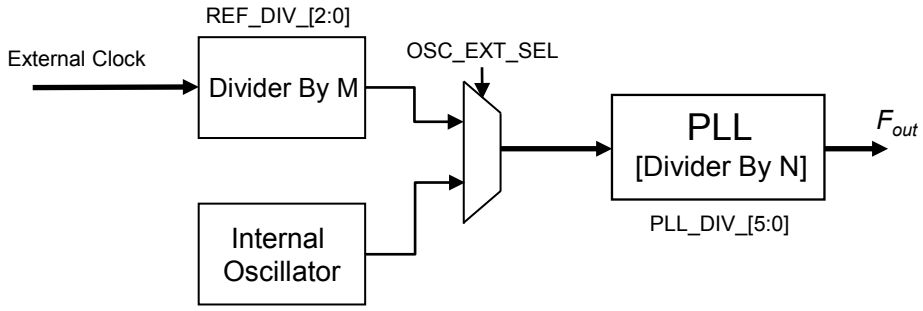
The IIS3DWB10IS contains an internal oscillator and PLL (phased locked loop) to be fully autonomous in the generation of all the internal synchronization signals and the ODR.

However, in order to provide both the capability of synchronizing the acquisition of an array of sensors and to provide a reference signal with the accuracy required by each application, it is possible to provide an external reference clock to the sensor.

The internal PLL synthesizes a 40.96 MHz frequency using a reference signal coming from an external clock or internal low-frequency oscillator. The external reference frequency range is from 16 kHz up to 5.12 MHz.

The reference is selected by means of the selection bit OSC_EXT_SEL in register PLL_CTRL_1 (0Ah - R/W). As shown in Figure 48. PLL block, the external reference signal is divided at the PLL input by a first programmable divider (M) and then in the feedback it is divided by a second one (N). In this way, despite the reference frequency, the output can be adapted to 40.96 MHz. For an exhaustive list of the configuration of the M and N dividers, refer to the application note ANXXXX.

Figure 48. PLL block



4.6 Sleep counter

A long-term timer is embedded in the IIS3DWB10IS to enable specific low-power modes based on timing activation, or also to serve the ISPU as a general-purpose timer.

The sleep counter works both in continuous mode and burst mode and even during the sleep conditions.

The sleep counter is clocked at 40 kHz and the range of counting can be set between 0.1 seconds and around 5 days.

Figure 49. Sleep counter overview (high level description)

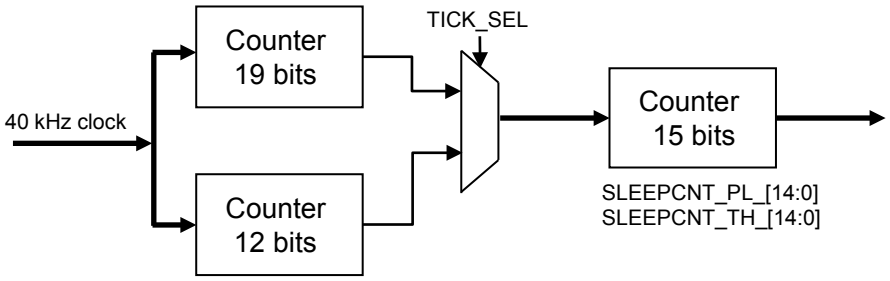


Table 14. Timescale for the sleep counter

Tick_sel	Tick time	Maximum time
0 = slow timescale	13.1072 s	~119.301 h
1 = fast timescale	0.1024 s	~55.922 m

4.7 ISPU (intelligent sensor processing unit)

The IIS3DWB10IS embeds a new ST category of processing: the ISPU (intelligent sensor processing unit).

The ISPU is an embedded programmable core, based on the proprietary ultralow-power STRED architecture that can run any type of real-time processing, from basic signal processing to artificial intelligence algorithms, on the data streamed by the sensor.

The ISPU is based on a 32-bit RISC Harvard architecture, features a floating-point unit (FPU) to accelerate single-precision floating-point operations (multiplication, addition, subtraction, maximum, minimum, accumulate and multiply, and square root) and implements additional hardware blocks (accelerators) to accelerate loops, circular buffers, and specific portions of the FFT computation.

The ISPU is supported by 32 KB of program memory + 56 KB of data memory with concurrent access. The whole memory can be however flexibly re-allocated as program or data memory as it is more convenient for the application.

The ISPU program can be written in standard C code, which allows a fast development cycle, high flexibility, and reuse of code written for other architectures.

The ISPU operates at a clock frequency of 40 MHz, ensuring fast processing capabilities with 40 MIPS and 40 MFLOPS and featuring a final Coremark evaluation of 1.95/MHz

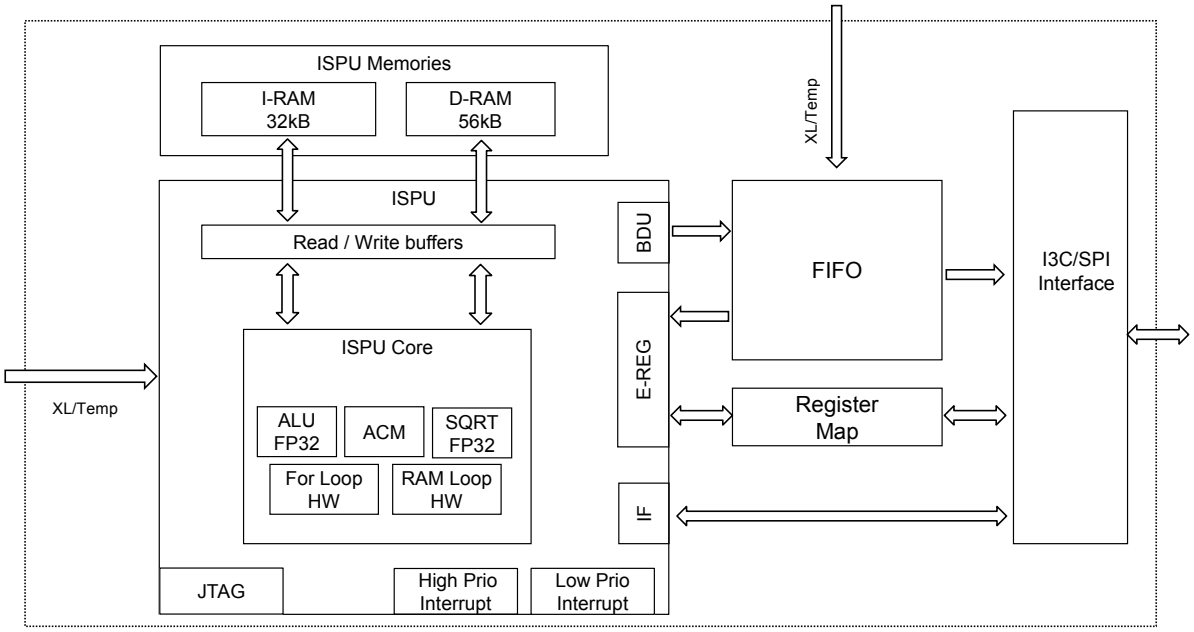
An ISPU toolchain is provided on the STMicroelectronics website, containing tools for the compilation of the code and the conversion of binary files to a format suitable to be loaded in the device.

The program of the ISPU, hosted in volatile memory, should be loaded at power-up of the device by an external host through the SPI/I3C interface.

The ISPU can acquire the data from the embedded accelerometer and temperature sensor. It can read the status of the device and change its configuration, and it can generate interrupts and exchange data with the host over the SPI/I3C interface by means of dedicated register access.

The detailed architecture of the ISPU is illustrated in the following figure.

Figure 50. ISPU block diagram



4.7.1 ISPU architecture highlights

4.7.1.1 Hardware accelerators

The ISPU includes four hardware accelerators designed to enhance algorithm performance by reducing execution time and power consumption. The list of these accelerators along with brief descriptions is as follows:

- Data addressing
  - This accelerator manages memory and bus addresses. When it identifies a specific address, it can remap it to a precalculated one. This feature supports addressing of data buffers, including circular buffers, and data involved in FFT computation.
- Instruction loop
  - This accelerator monitors memory and bus addresses and helps in efficiently managing repeated instruction sequences.
- ACM (accumulate and multiply)
  - This accelerator accumulates the results of previous multiplication operations and performs new multiplications, streamlining complex calculations.
- Square root FP32
  - This accelerator quickly computes the square root in IEEE-compliant FP32 format. It completes the calculation in four clock cycles without using iterative approximations, unlike software methods based on the fast inverse square root algorithm.

4.7.2 Onboard memories (RAM and FIFO) and register access

The IIS3DWB10IS device includes 88 KB of embedded RAM and 20 KB of FIFO. Both the RAM and FIFO can be accessed via the serial interface and the ISPU.

The ISPU, with its Harvard architecture, is equipped with separate instruction and data memories, allowing for true concurrent fetching of instructions and data. Despite minimal access time penalties, the ISPU can read and write flexibly across the entire 88 KB of RAM. From the user's perspective, code and data sections can thus be organized entirely in a flexible manner.

The ISPU can utilize the FIFO as a peripheral, enabling it to write an entire data batch to the FIFO or, conversely, read the FIFO's content. Read access to FIFO is mutually exclusive between the ISPU and the serial interface.

The ISPU can access a significant portion of the sensor's registers in both read and write modes. This capability allows the firmware to read or configure most sensor parameters and make real-time changes to the sensor's operating mode and configuration, adapting it to the context.

The details on how the ISPU can access the RAMs / FIFO / sensor registers and how the serial interface can access the RAMs are described in Section 10: ISPU core register access (registers accessible from the core of the ISPU).

The high-level view of the memory map of the ISPU is described in the table below.

Table 15. ISPU memory map

From	To	Size	Description	ISPU address space
0x00000	0x01FFF	8 KB	Reserved	
0x02000	0x0FFFF	56 KB	Data RAM	Memory
0x10000	0x3FFFF	192 KB	Reserved	
0x40000	0x47FFF	32 KB	Code RAM	Memory
0x48000	0x7FFFF	224 KB	Reserved	
0x80000	0x800FF	256 bytes	ISPU system registers	Registers
0x80100	0x807FF	1.75 KB	Reserved	
0x80800	0x808FF	256 bytes	ISPU registers	Registers
0x80900	0x809FF	256 bytes	External registers	Registers
0x80A00	0xBFFFF	253.5 KB	Reserved	
0xC0000	0xFFFFF	256 KB	Reserved	

4.7.3 Accelerometer datapath to ISPU

There are independent and dedicated datapaths from the accelerometer and temperature sensor to the ISPU and the serial interface (FIFO/output). The ODR of the two datapaths can be set independently.

4.7.4 Low-priority or high-priority interrupts

The ISPU features a high-priority interrupt system that allows it to wake up upon receiving new valid data from the sensors. It automatically initiates a dedicated firmware routine that executes with the highest priority. This high-priority routine can be fully customized by the programmer.

Additionally, the ISPU includes a low-priority interrupt mechanism. Similar to the high-priority system, the low-priority system pauses the execution of the main program when a low-priority interrupt is received. However, the high-priority mechanism takes precedence. If a high-priority interrupt occurs during a low-priority routine, the system switches to the high-priority routine and returns to the low-priority routine once the high-priority task is completed.

For a more detailed description of the interrupt mechanisms, refer to the application note ANXXXX.

4.7.5 ISPU output registers

The ISPU allocates 32-byte registers for general-purpose output data available to the serial interfaces.

These output registers (ISPU_DOUT) in the ISPU page can be read over the serial interface.

General-purpose output data means that the output data length can be arranged as desired in the ISPU program. The output can be, for example, 8, 16, or 32 bits in length.

4.7.6 External registers access

The ISPU has an external registers interface to address the 128 possible registers external to the ISPU and which interact with the sensor portion of the device.

The external registers interface unlocks many ISPU features and enables the cross-connections with the rest of the sensor functionalities.

The external registers interface data bus size is 32-bit, so firmware can access (both in read and write) the single/dual/quad registers in a single operation.

The register map of the external registers is listed in Section 10: ISPU core register access (registers accessible from the core of the ISPU).

4.7.7 Algorithm routines

The ISPU has built-in support to execute up to 16 algorithms within a configurable sequence order.

For a detailed description of the mechanism, refer to the application note ANXXXX.

4.8 FIFO

The presence of a FIFO buffer allows consistent power saving for the system since the host processor does not need to continuously poll data from the sensor, but it can wake up only when needed and burst the significant data out of the FIFO.

The IIS3DWB10IS embeds a large FIFO buffer capable of batching up to 2048 entries of 72 bit of data + 8 bits of tag.

Each entry can contain different kinds of sensor data:

- Main sensors (the physical sensors)
  - Accelerometer raw sensor data
  - Temperature sensor data
- Virtual sensors
  - Nonsensor data sources. Virtual sensors are generated by the ISPU.
- Auxiliary sensors (contain information of the status of the device)
  - Timestamp sensor

Each entry has its own TAG (8-bit) to specify the kind of entry.

The FIFO read is done by accessing a dedicated space in the register map.

The data stored in FIFO are accessible from dedicated registers and each FIFO word is composed of 10 bytes: one tag byte FIFO_DATA_OUT_TAG (40h - R), in order to identify the content of the entry, and 9 bytes of fixed data (from FIFO_DATA_OUT_D2 (43h R), FIFO_DATA_OUT_D1 (42h R), and FIFO_DATA_OUT_D0 (41h R) through FIFO_DATA_OUT_D8 (49h R), FIFO_DATA_OUT_D7 (48h R), and FIFO_DATA_OUT_D6 (47h R)).

4.8.1 FIFO TAG

When accessing the entry in the FIFO, the TAG associated with the data is stored in FIFO_DATA_OUT_TAG (40h - R) as:

- TAG sensor: identifies the sensor stored in the 9-byte data field according to the codification in Table 16. FIFO TAG names.

The timestamp is relative to the time slots, which are defined by the batching of the main sensors. The TAG supported are listed in Table 16. FIFO TAG names.

Table 16. FIFO TAG names

TAG name	TAG value	Data batched in FIFO
TAG_EMPTY	00000000	FIFO empty condition
TAG_XL	00010000	24-bit (20-bit sign-extended) X-axis + 24-bit (20-bit sign-extended) Y-axis + 24-bit (20-bit sign-extended) Z-axis
TAG_TEMP	00011000	16-bit temperature sensor data
TAG_TS	00100000	40-bit timestamp data
TAG_ISPU	8 bits defined by the ISPU	72-bit ISPU results

In order to save data in FIFO, it is mandatory to enable FIFO and the bit to batch the desired type of data, setting the corresponding bits in FIFO_CTRL1 (05h - R/W), FIFO_CTRL2 (06h - R/W), FIFO_CTRL3 (07h - R/W).

4.8.2 FIFO main sensors

The FIFO main sensors can trigger writes in FIFO. These sensors include:

- Accelerometer raw sensor data
- Temperature sensor

The description of FIFO data output for the main sensors depending on the TAG_SENSOR read is indicated in the following table.

Table 17. FIFO data output for main sensors

	TAG_XL	TAG_TEMP
FIFO_DATA_OUT_D0	XL OUTX_L	0x00
FIFO_DATA_OUT_D1	XL OUTX_M	OUT_TEMP_L
FIFO_DATA_OUT_D2	XL OUTX_H	OUT_TEMP_H
FIFO_DATA_OUT_D3	XL OUTY_L	x“00”
FIFO_DATA_OUT_D4	XL OUTY_M	x“00”
FIFO_DATA_OUT_D5	XL OUTY_H	x“00”
FIFO_DATA_OUT_D6	XL OUTZ_L	x“00”
FIFO_DATA_OUT_D7	XL OUTZ_M	x“00”
FIFO_DATA_OUT_D8	XL OUTZ_H	x“00”

4.8.3 FIFO auxiliary sensors

Auxiliary sensors are service sensors for the main sensors.

The timestamp sensor is an auxiliary sensor. It stores the timestamp of the FIFO time slot.

The batch rate of the timestamp sensor can be configured using DEC_TS_BATCH[1:0] in the FIFO_CTRL2 (06h - R/W) register.

It is possible to write the timestamp value in FIFO at the beginning or with a decimation of the data rate of the fastest main sensor divided by 128 or divided by 256.

When enabled, the timestamp is the first sensor written in FIFO at each main sensor BDR event.

The timestamp value is a 40-bit unsigned value and 1 LSB = 25 µs.

Table 18. Timestamp data in FIFO

Timestamp data	FIFO output
TIMESTAMP [7:0]	FIFO_DATA_OUT_D0
TIMESTAMP [15:8]	FIFO_DATA_OUT_D1
TIMESTAMP [23:16]	FIFO_DATA_OUT_D2
TIMESTAMP [31:24]	FIFO_DATA_OUT_D3
TIMESTAMP [40:32]	FIFO_DATA_OUT_D4

4.8.4 FIFO virtual sensors

It is possible to batch data from the ISPU in the FIFO. One reason to use this feature is, for example, to store some algorithms results from the ISPU in the FIFO, and then let the ISPU go to sleep mode, letting the user access the results over the serial interface in FIFO whenever required.

The protocol that the ISPU should adopt to store data in the FIFO is described in the application note ANXXXX.

4.8.5 FIFO modes

The FIFO buffer can work according to six different modes:

- Bypass mode
- FIFO mode
- Continuous mode
- Continuous-to-FIFO mode
- Bypass-to-continuous mode
- Bypass-to-FIFO mode

Each mode is selected by the FIFO_MODE[2:0] bits in the FIFO_CTRL3 (07h - R/W) register.

The programmable FIFO watermark is selected in the FIFO_CTRL1 (05h - R/W) and FIFO_CTRL2 (06h - R/W) registers.

The status of FIFO is available in the FIFO_STATUS_1 (4Ch - R) and FIFO_STATUS_2 (4Dh - R) registers and can be used to generate dedicated interrupts on the INT1 and INT2 pins.

The FIFO_WTM_IA bit in the FIFO_STATUS_2 (4Dh - R) register goes to 1 when the number of unread samples is greater than or equal to WTM_[11:0] in the FIFO_CTRL1 (05h - R/W) and FIFO_CTRL2 (06h - R/W) registers. If the WTM_[11:0] field is equal to 0, the FIFO_WTM_IA bit in FIFO_STATUS_2 (4Dh - R) always stays at 0.

FIFO_OVRN in the FIFO_STATUS_2 (4Dh - R) register is equal to 1 if a FIFO sample is overwritten. DIFF_[11:0] in the FIFO_STATUS_1 (4Ch - R) and FIFO_STATUS_2 (4Dh - R) registers contains the number of unread sensor data (TAG + 9 bytes) stored in FIFO

When DIFF_[11:0] is equal to 0x00, FIFO is empty. When DIFF_[11:0] is equal to 0x800, FIFO is full and the unread samples are 2048.

4.8.5.1 *Bypass mode*

In bypass mode (FIFO_MODE_[2:0] = 000), the FIFO is not operational, no data are collected in FIFO memory, which remains empty with the only actual sample available in the output registers.

Bypass mode is also used to reset the FIFO when in FIFO mode.

4.8.5.2 *FIFO mode*

In FIFO mode (FIFO_MODE_[2:0] = 001), data from the output channels are stored in the FIFO memory until it is full.

When 2048 unread samples are stored in FIFO, data collection is stopped.

To reset the FIFO content, bypass mode should be selected by writing FIFO_MODE_[2:0] = 000. After this reset command, it is possible to restart FIFO mode, writing FIFO_MODE_[2:0]=001.

4.8.5.3 *Continuous mode*

Continuous mode (FIFO_MODE_[2:0] = 110) provides a continuous FIFO update: when 2048 unread samples are stored in memory, and as new data arrives, the oldest data is discarded and overwritten by the newer.

4.8.5.4 *Continuous-to-FIFO mode*

In continuous-to-FIFO mode (FIFO_MODE_[2:0] = 011), FIFO operates in continuous mode and switches to FIFO mode upon an edge trigger event.

When the FIFO is full, data collection is stopped. The trigger event could arrive from the external INT2 pin or from the ISPU depending on the FIFO_TRIGGER_CFG bit in FIFO_CTRL2 (06h - R/W).

Figure 51. Continuous-to-FIFO mode

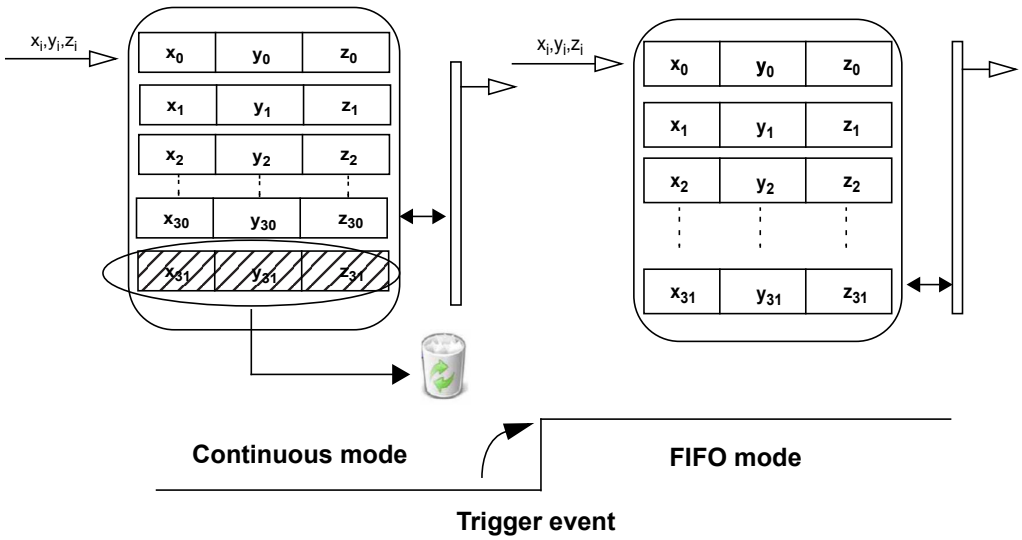
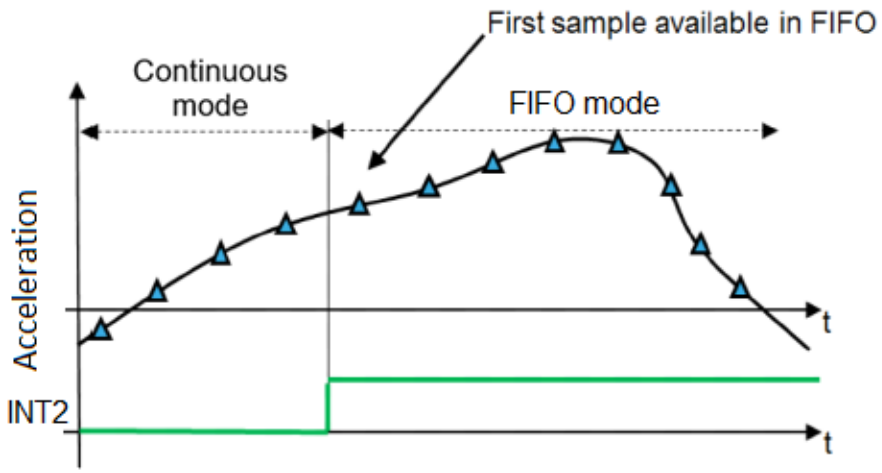


Figure 52. Internal asynchronous trigger to FIFO for continuous-to-FIFO mode



4.8.5.5 *Bypass-to-continuous mode*

In bypass-to-continuous mode (FIFO_MODE_[2:0] = 100), FIFO starts in bypass mode and, upon an edge trigger event, FIFO starts operating in continuous mode.

The trigger event could arrive from the external INT2 pin or from the ISPU depending on the FIFO_TRIGGER_CFG bit in FIFO_CTRL2 (06h - R/W).

Figure 53. Bypass-to-continuous mode

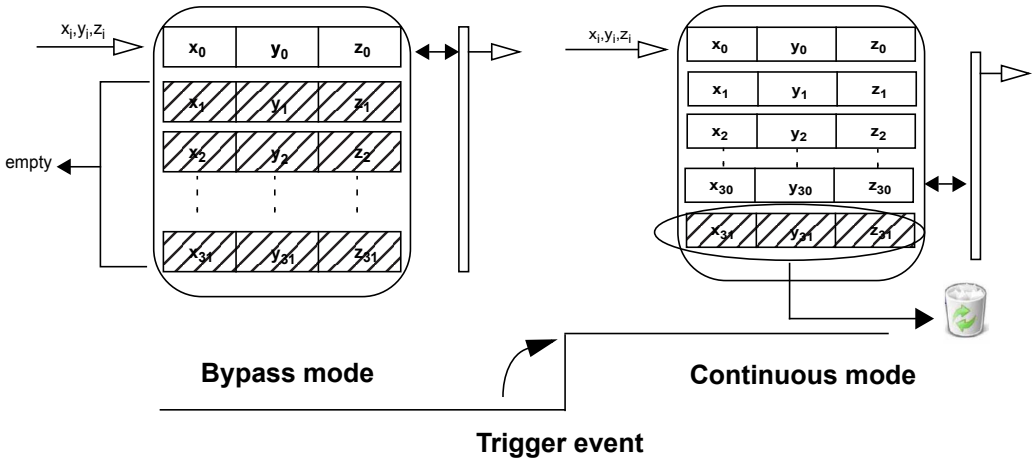
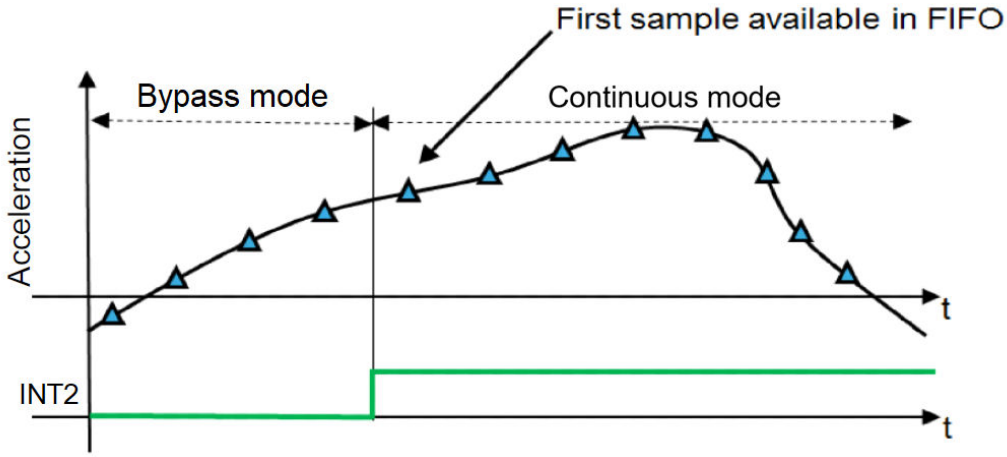


Figure 54. Internal asynchronous trigger to FIFO for bypass-to-continuous mode



4.8.5.6 *Bypass-to-FIFO mode*

In bypass-to-FIFO mode (FIFO_MODE_[2:0]=111), FIFO starts in bypass mode and, upon an edge trigger event, FIFO starts operating in FIFO mode.

The trigger event could arrive from the external INT2 pin or from the ISPU depending on the FIFO_TRIGGER_CFG bit in FIFO_CTRL2 (06h - R/W).

4.8.6 Sequence of reading output data from FIFO

When data from FIFO have to be read, in order to minimize interface transfer, an automatic address increment is provided.

To correctly manage the read pointer, every time data is read from FIFO, it should be read with multiple consecutive byte reads: ten byte reads from the address 40h to the address 49h.

FIFO empty condition

When FIFO is depleted, a special TAG is used to flag the empty condition and to avoid reading duplicated samples.

This TAG is the EMPTY_TAG[7:0] = 00000000.

Changing FIFO modes

When FIFO should be used, it should be kept turned on also during the bypass transitions.

The FIFO_EN bit in CTRL3 (12h - R/W) should be set.

To change FIFO mode, it is mandatory to switch to bypass mode before switching to the target mode.

5 Application hints

Figure 55. **IIS3DWB10IS electrical connections**

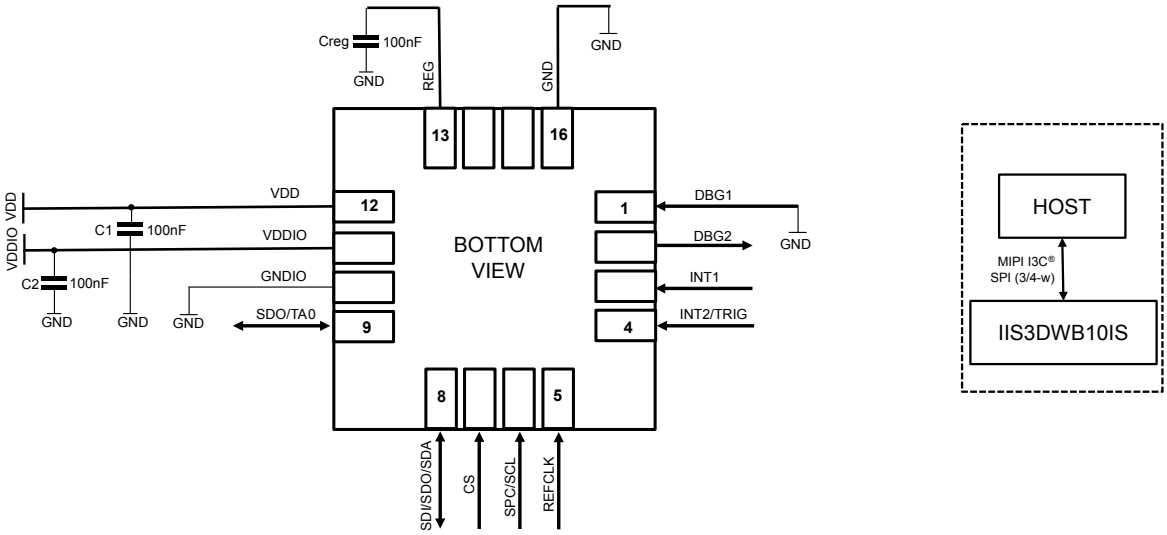


Table 19. Internal pin status

Pin#	Name	Function	Default status	Recommended connection
1	DBG1[1]	Debug interface pin	Input without pull-up	During normal operation, this pin must be tied to GND. When the JTAG ISPU debug interface is enabled, it is the TCK pin of the JTAG interface.
2	DBG2[1]	Debug interface pin	Output forced to GND	During normal operation, this pin must be electrically unconnected but soldered to the PCB. When the JTAG ISPU debug interface is enabled, it is the TDO pin of the JTAG interface.
3	INT1	Programmable interrupt #1	Input with pull-down	During normal operation, this pin is application specific. When the JTAG[1] ISPU debug interface is used and enabled, it is the TDI pin of the JTAG interface.
4	INT2 TRIG	Programmable interrupt #2 or External trigger input	Input with pull-down	During normal operation, this pin is application specific. When the JTAG[1] ISPU debug interface is used and enabled, it is the TMS pin of the JTAG interface.
5	REFCLK	Reference clock Connect to VDDIO or GNDIO if not used	Input without pull-up	Connect to VDDIO or GNDIO if not used
6	SPC SCL	SPI serial port clock (SPC) I3C serial clock (SCL)	Input without pull-up	Application specific
7	CS	SPI chip select	Input with pull-up	Application specific
8	SDI SDO SDA	SPI serial data input (SDI) 3-wire interface serial data output (SDO) I3C serial data (SDA)	Input without pull-up	Application specific
9	SDO TA0	SPI 4-wire interface serial data output (SDO) I3C least significant bit of the static address (TA0)	Input without pull-up	Application specific
10	GNDIO	Ground for IO pins	-	Connect to GND
11	VDDIO	Power supply for IO pins	-	Power supply decoupling is recommended (100 nF ceramic capacitor).
12	VDD	Power supply	-	Power supply decoupling is recommended (100 nF ceramic capacitor).
13	REG	Capacitance connection pin for internal regulator	-	A 100 nF ceramic capacitor to ground should be connected to this pin (100 nF value should be assured with a voltage close to 1.8 V).
14	RES	Reserved	-	This pin must be soldered to the PCB. It is advisable, but not mandatory, to connect it either to GND/GNDIO or VDD/VDDIO.
15	RES	Reserved	-	This pin must be soldered to the PCB. It is advisable, but not mandatory, to connect it either to GND/GNDIO or VDD/VDDIO.
16	GND	Ground	-	Connect to GND

1. JTAG interface reserved for ST internal use only. Not supported for customer access.

6 Register map (accessible from the host serial interface)

The table given below provides a list of the 8-bit registers embedded in the device and the corresponding addresses.

The register map is organized by pages. This means that different registers can be accessed through the same address by selecting the corresponding page. The page is determined by the value written in the following bit:

- PAGE_SEL in register RAM_ACCESS (01h - R/W), bit 7

The list and the descriptions of the registers in the sensor register map (PAGE_SEL = 0) are indicated in the following table.

Table 20. Sensor register map

Name	Type	Register address		Default
		Hex	Binary	
RAM_ACCESS	R/W	01	00000001	00000000
PIN_CTRL	R/W	02	00000010	10000011
IF_CFG	R/W	03	00000011	00001000
RESERVED	-	04		
FIFO_CTRL1	R/W	05	00000101	00000000
FIFO_CTRL2	R/W	06	00000110	00000000
FIFO_CTRL3	R/W	07	00000111	00000000
COUNTER_BDR_H	R/W	08	00001000	00000000
COUNTER_BDR_L	R/W	09	00001001	00000000
PLL_CTRL_1	R/W	0A	00001010	00000000
PLL_CTRL_2	R/W	0B	00001011	00001011
INT_CTRL1	R/W	0C	00001100	00000000
INT_CTRL2	R/W	0D	00001101	00000000
INT_CTRL3	R/W	0E	00001110	00000000
WHO_AM_I	R	0F	00001111	01010000
CTRL1	R/W	10	00010000	00000000
CTRL2	R/W	11	00010001	00000000
CTRL3	R/W	12	00010010	00000100
CTRL4	R/W	13	00010011	00011101
I3C_CTRL	R/W	14	00010100	00000000
SPI_CTRL	R/W	15	00010101	00000010
RESERVED	-	16		
ST_CTRL	R/W	17	00010111	00000000
ISPU_CTRL1	R/W	18	00011000	00000000
ISPU_CTRL2	R/W	19	00011001	00000000
ISPU_INT_STATUS1_MAINPAGE	R	1A	00011010	output
ISPU_INT_STATUS2_MAINPAGE	R	1B	00011011	output
RESERVED	-	1C - 1D		
STATUS_REG	R	1E	00011110	output
RESERVED	-	1F		
TPF	R	20	00100000	output

Name	Type	Register address		Default
		Hex	Binary	
DEVICE STATUS	R	21	00100001	output
OUT_TEMP_L	R	22	00100010	output
OUT_TEMP_H	R	23	00100011	output
OUTX_L	R	24	00100100	output
OUTX_M	R	25	00100101	output
OUTX_H	R	26	00100110	output
OUTX_HH	R	27	00100111	output
OUTY_L	R	28	00101000	output
OUTY_M	R	29	00101001	output
OUTY_H	R	2A	00101010	output
OUTY_HH	R	2B	00101011	output
OUTZ_L	R	2C	00101100	output
OUTZ_M	R	2D	00101101	output
OUTZ_H	R	2E	00101110	output
OUTZ_HH	R	2F	00101111	output
TIMESTAMP0	R	30	00110000	output
TIMESTAMP1	R	31	00110001	output
TIMESTAMP2	R	32	00110010	output
TIMESTAMP3	R	33	00110011	output
TIMESTAMP4	R	34	00110100	output
RESERVED	-	35-37		
SLEEP_CNT_CFG	R/W	38	00111000	00000000
LOPRIO_EN	R/W	39	00111001	00000000
RESERVED	-	3A – 3D		
SLEEPCNT_TH_L	R/W	3E	00111110	00000000
SLEEPCNT_TH_H	R/W	3F	00111111	00000000
FIFO_DATA_OUT_TAG	R	40	01000000	output
FIFO_DATA_OUT_D0	R	41	01000001	output
FIFO_DATA_OUT_D1	R	42	01000010	output
FIFO_DATA_OUT_D2	R	43	01000011	output
FIFO_DATA_OUT_D3	R	44	01000100	output
FIFO_DATA_OUT_D4	R	45	01000101	output
FIFO_DATA_OUT_D5	R	46	01000110	output
FIFO_DATA_OUT_D6	R	47	01000111	output
FIFO_DATA_OUT_D7	R	48	01001000	output
FIFO_DATA_OUT_D8	R	49	01001001	output
RESERVED	-	4A – 4B		
FIFO_STATUS1	R	4C	01001100	output
FIFO_STATUS2	R	4D	01001101	output
RESERVED	-	4E – 4F		

Name	Type	Register address		Default
		Hex	Binary	
INTERNAL_FREQ_FINE	R	50	01010000	output
RESERVED	-	51		
ISPU_DUMMYCFG1	R/W	52	01010010	00000000
ISPU_DUMMYCFG2	R/W	53	01010011	00000000
ISPU_DUMMYCFG3	R/W	54	01010100	00000000
ISPU_DUMMYCFG4	R/W	55	01010101	00000000
ISPU_DUMMYCFG5	R/W	56	01010110	00000000
ISPU_DUMMYCFG6	R/W	57	01010111	00000000
ISPU_DUMMYCFG7	R/W	58	01011000	00000000
ISPU_DUMMYCFG8	R/W	59	01011001	00000000
ISPU_DUMMYCFG9	R/W	5A	01011010	00000000
ISPU_DUMMYCFG10	R/W	5B	01011011	00000000
ISPU_DUMMYCFG11	R/W	5C	01011100	00000000
ISPU_DUMMYCFG12	R/W	5D	01011101	00000000
ISPU_DUMMYCFG13	R/W	5E	01011110	00000000
ISPU_DUMMYCFG14	R/W	5F	01011111	00000000
ISPU_DUMMYCFG15	R/W	60	01100000	00000000
ISPU_DUMMYCFG16	R/W	61	01100001	00000000
SLEEPCNT_TIME_L	R	62	01100010	output
SLEEPCNT_TIME_H	R	63	01100011	output
RESERVED	-	64 – 6D		
INT_CTRL4	R/W	6E	01101110	00000000
RESERVED	-	6F – 7F		

Reserved registers must not be changed. Writing to those registers may cause permanent damage to the device.

The content of the registers that are loaded at boot should not be changed. They contain the factory calibration values. Their content is automatically restored when the device is powered up.

7 Register description

The device contains a set of registers, which are used to control its behavior and to retrieve sensor data. The register addresses, made up of 7 bits, are used to identify them and to write the data through the serial interface.

7.1 RAM_ACCESS (01h - R/W)

Defines the accessible register map

Table 21. RAM_ACCESS register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
PAGE_SEL	ISPU_RAM_ ACCESS_IF	0(1)	0(1)	0(1)	0(1)	0(1)	SW_POR

1. *This bit must be set to the value indicated for the correct operation of the device.*

Table 22. RAM_ACCESS register description

PAGE_SEL	0x0	R/W	Register page accessible over the serial interface. Default value: 0 (0: sensor register map; 1: ISPU register map)
ISPU_RAM_ACCESS_IF	0x0	R/W	Enables the R/W access of the ISPU RAM over the serial interface. A latency of 0.45 ms (min) occurs before RAM access. Default value: 0 (0: disabled; 1: enabled)
SW_POR	0x0	R/W	Assert high to apply a complete reset of the sensor. It is equivalent to the IBI issued over the I3C interface. Once asserted, it take several ms to complete (1: invoke device reset)

7.2 PIN_CTRL (02h - R/W)

Pin control register

Table 23. PIN_CTRL register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
IBHR_ POR_EN	0(1)	0(1)	0(1)	PD_DIS_ INT1	PD_DIS_ INT0	IO_PIN_ STRENGTH_1	IO_PIN_ STRENGTH_0

1. *This bit must be set to the value indicated for the correct operation of the device.*

Table 24. PIN_CTRL register description

IBHR_POR_EN	0x1	R/W	Selects the I3C IBHR device reset pattern. Default value: 1 (0: configuration reset (software reset) + dynamic address reset; 1: global reset (POR reset))
PD_DIS_INT[1:0]	0x0	R/W	Disables the pull-down on the interrupt pins. Default value: 00 (00: pull-down on INT1 and INT2 enabled; 10: pull-down on INT1 enabled, INT2 disabled; 01: pull-down on INT1 disabled, INT2 enabled; 11: pull-down on INT1 disabled, INT2 disabled.)
IO_PIN_STRENGTH_[1:0]	0x1	R/W	Defines drive strength for the interrupt pins Default value: 01 (00: lower strength (recommended for VDDIO ≥3.0 V); 01: intermediate strength (recommended for VDDIO <3.0 V); 10: reserved (do not use); 11: highest strength)

7.3 IF_CFG (03h - R/W)

Interface control register

Table 25. IF_CFG register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
SDA_PU_EN	0(1)	0(1)	INT_ACTIVE_LEVEL	INT_PP_OD	0(1)	0(1)	I3C_disable

1. *This bit must be set to the value indicated for the correct operation of the device.*

Table 26. IF_CFG register description

SDA_PU_EN	0x0	R/W	Enables pull-up on SCL and SDA (I3C) pins. Default value: 0 (0: pull-up enabled; 1: pull-up disabled)
INT_ACTIVE_LEVEL	0x0	R/W	Defines the active level of the interrupt. Default value: 0 (0: active-high interrupts (default); 1: active-low interrupts)
INT_PP_OD	0x1	R/W	Defines the push-pull/open-drain mode for the INT pins. Default value: 1 (0: INT pins in push-pull mode; 1: INT pins in open-drain mode (default))
I3C_disable	0x0	R/W	Disables the I3C interface command. Default value: 0 (0: I3C interface command enabled (default); 1: I3C interface command disabled)

7.4 FIFO_CTRL1 (05h - R/W)

FIFO control register

Table 27. FIFO_CTRL1 register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
WTM_7	WTM_6	WTM_5	WTM_4	WTM_3	WTM_2	WTM_1	WTM_0

Table 28. FIFO_CTRL1 register description

WTM_[7:0]	0x00	R/W	FIFO watermark threshold 1 LSB = 1 sensor (9 bytes) + TAG (1 byte) written in FIFO Watermark flag rises when the number of bytes written in the FIFO is greater or equal to the threshold level.

7.5 FIFO_CTRL2 (06h - R/W)

FIFO control register

Table 29. FIFO_CTRL2 register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
FIFO_READ_FROM_ISPU	FIFO_TRIGGER_CFG	DEC_TS_BATCH_1	DEC_TS_BATCH_0	WTM_11	WTM_10	WTM_9	WTM_8

Table 30. FIFO_CTRL2 register description

FIFO_READ_FROM_ISPU	0x0	R/W	Defines whether the FIFO read is controlled from the ISPU. Default value: 0 (0: FIFO read is controlled by the interface (default); (1: FIFO read is controlled by the ISPU)
FIFO_TRIGGER_CFG	0x0	R/W	Defines whether the FIFO trigger is from the ISPU or INT2 pin. Default value: 0 (0: FIFO trigger is from the ISPU (default); 1: FIFO trigger is from the INT2 pin)
DEC_TS_BATCH_[1:0]	0x0	R/W	Selects the decimation for timestamp batching in FIFO. The batch data rate for the timestamp is the data rate of the fastest batched main sensor (accelerometer or temperature) divided by the decimation factor. (00: timestamp not batched in FIFO (default); 01: only the first timestamp sample is batched when FIFO is enabled. No periodic decimation is applied; 10: decimation 128; 11: decimation 256)
WTM_[11:8]	0x0	R/W	FIFO watermark threshold 1 LSB = 1 sensor (9 bytes) + TAG (1 byte) written in FIFO The watermark flag rises when the number of bytes written in the FIFO is greater or equal to the threshold level.

7.6 FIFO_CTRL3 (07h - R/W)

FIFO control register

Table 31. FIFO_CTRL3 register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
STOP_ON_WTM	0	ISPU_BATCH	T_BATCH	XL_BATCH	FIFO_MODE_2	FIFO_MODE_1	FIFO_MODE_0

Table 32. FIFO_CTRL3 register description

STOP_ON_WTM	0x0	R/W	The FIFO sensing chain stops storing samples when a threshold level is reached. Default value: 0 (0: FIFO storage of samples does not stop at the threshold (default); 1: FIFO storage of samples stops at the threshold level defined by the FIFO watermark threshold)
ISPU_BATCH	0x0	R/W	Enables batching ISPU data. Default value: 0 (0: ISPU batching off (default); 1: ISPU batching on)
T_BATCH	0x0	R/W	Enables batching the embedded temperature sensor data. Default value: 0 (0: temperature sensor batching off (default); 1: temperature sensor batching on)
XL_BATCH	0x0	R/W	Enables batching the accelerometer sensor data. Default value: 0 (0: accelerometer sensor batching off (default); 1: accelerometer sensor batching on)
FIFO_MODE_[2:0]	0x0	R/W	FIFO mode selection (000: bypass mode; 001: FIFO mode; 010: continuous mode; 011: continuous, FIFO on trigger mode (if trigger = 0, stays in continuous mode; if trigger = 1, stays in FIFO mode); 100: bypass, continuous on trigger mode (if trigger = 0, stays in bypass mode; if trigger = 1, stays in continuous mode); 101: reserved; 110: reserved; 111: bypass, FIFO on trigger mode (if trigger = 0, stays in bypass mode; if trigger = 1, stays in FIFO mode)

7.7 COUNTER_BDR_H (08h - R/W) and COUNTER_BDR_L (09h - R/W)

Table 33. COUNTER_BDR_H register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
RST_ COUNTER_BDR	0	TRIGGER_ COUNTER_BDR	RST_COUNTER_ BDR_RD_DIFF	CNT_BDR_TH1 1	CNT_BDR_TH1 0	CNT_BDR_TH9	CNT_BDR_TH8

Table 34. COUNTER_BDR_L register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
CNT_BDR_TH7	CNT_BDR_TH6	CNT_BDR_TH5	CNT_BDR_TH4	CNT_BDR_TH3	CNT_BDR_TH2	CNT_BDR_TH1	CNT_BDR_TH0

Table 35. COUNTER_BDR_H register description

RST_COUNTER_BDR	0x0	R/W	Resets the internal counter of batch events for a single sensor. Default value: 0 This bit is automatically reset to zero when it is set to 1. (1: apply reset)
TRIGGER_COUNTER_BDR	0x0	R/W	Defines whether the trigger for the BDR counter is the accelerometer write in FIFO or the ISPU write in FIFO. Default value: 0 (0: trigger for counter BDR is accelerometer write in FIFO (default); 1: trigger for counter BDR is ISPU write in FIFO)
RST_COUNTER_BDR_RD_DIFF	0x0	R/W	Enables the reset of the internal counter of batch events when the FIFO status registers are read (register at address: 4Ch and 4Dh). The two FIFO status registers must be read using a multiple read starting from address 4Ch. In this way, it is possible to reset the FIFO BDR counter when the FIFO is flushed even if no FIFO interrupt is generated.
CNT_BDR_TH[11:0]	0x0	R/W	Sets the threshold for the internal counter for event batching. When the counter reaches the threshold, the counter is reset and the COUNTER_BDR_IA flag in FIFO_STATUS2 is set to 1.

7.8 PLL_CTRL_1 (0Ah - R/W)

PLL control registers

Table 36. PLL_CTRL_1 register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
OSC_EXT_SEL	0(1)	0(1)	0(1)	0(1)	REF_DIV_2	REF_DIV_1	REF_DIV_0

1. *This bit must be set to the value indicated for the correct operation of the device.*

Table 37. PLL_CTRL_1 register description

OSC_EXT_SEL	0x0	R/W	Selection of the PLL reference signal. Default value: 0 (0: the PLL uses the internal clock as the reference (default); 1: the PLL uses the external clock as the reference.)
REF_DIV_[2:0]	0x0	R/W	Selects the division factor of the PLL input programmable divider. (000: no divider; 001: divide by 2; 010: divide by 4; 011: divide by 8; 100: divide by 16; 101: divide by 32; 110: divide by 64; 111: divide by 128)

7.9 PLL_CTRL_2 (0Bh - R/W)

PLL control registers

Table 38. PLL_CTRL_2 register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
0(1)	0(1)	PLL_DIV_5	PLL_DIV_4	PLL_DIV_3	PLL_DIV_2	PLL_DIV_1	PLL_DIV_0

1. *This bit must be set to the value indicated for the correct operation of the device.*

Table 39. PLL_CTRL_2 register description

PLL_DIV_[5:0]	0x0B	R/W	Selects the division factor of the PLL feedback programmable divider In order to set the proper divider factor (PLL_DIV) the following formula should be considered: Fpll = (Fin / REF_DIV) * (PLL_DIV + 32) * 32, for Fin > 20 kHz Fpll = (Fin / REF_DIV) * (PLL_DIV) * 64, for Fin < 20 kHz where REF_DIV can be set according to the description in register PLL_CTRL_1 (0Ah - R/W)

7.10 INT_CTRL1 (0Ch - R/W)

Interrupt control register 1

Table 40. INT_CTRL1 register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
PULSED_ DATAREADY	INT2_ ON_INT1	INT2_ SLEEP_ISPU	INT1_BOOT	0(1)	0(1)	INT1_ DRDY_TEMP	INT2_ DRDY_TEMP

1. *This bit must be set to the value indicated for the correct operation of the device.*

Table 41. INT_CTRL1 register description

PULSED_DATAREADY	0x0	R/W	Enables pulsed mode on data ready. Default value: 0 (0: latched mode data ready (default); 1: pulsed mode data ready)
INT2_ON_INT1	0x0	R/W	Moves INT2 interrupts to the INT1 pin (only those that do not have a dedicated configuration for INT1). Default value: 0
INT2_SLEEP_ISPU	0x0	R/W	Enables ISPU sleep signal on INT2. Default value: 0
INT1_BOOT	0x0	R/W	Enables boot status on the INT1 pin. Default value: 0
INT1_DRDY_TEMP	0x0	R/W	Enables temperature sensor data-ready interrupt on the INT1 pin.
INT2_DRDY_TEMP	0x0	R/W	Enables temperature sensor data-ready interrupt on the INT2 pin.

7.11 INT_CTRL2 (0Dh - R/W)

Interrupt control register 2

Table 42. **INT_CTRL2 register**

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
INT1_ COUNTER_BDR	INT1_ FIFO_FULL	INT1_ FIFO_OVR	INT1_ FIFO_TH	INT1_ EXT_TRIG	INT1_ SLEEPCNT	0(1)	INT1_ DRDY_XL

1. *This bit must be set to the value indicated for the correct operation of the device.*

Table 43. **INT_CTRL2 register description**

INT1_COUNTER_BDR	0x0	R/W	Enables BDR counter interrupt on the INT1 pin. Default value: 0.
INT1_FIFO_FULL	0x0	R/W	Enables FIFO full interrupt on the INT1 pin. Default value: 0
INT1_FIFO_OVR	0x0	R/W	Enables FIFO overrun interrupt on the INT1 pin. Default value: 0
INT1_FIFO_TH	0x0	R/W	Enables FIFO threshold interrupt on the INT1 pin. Default value: 0
INT1_EXT_TRIG	0x0	R/W	Enables the external trigger interrupt (from the INT2 pin) on INT1. Default value: 0
INT1_SLEEPCNT	0x0	R/W	Enables the sleep counter interrupt on the INT1 pin.
INT1_DRDY_XL	0x0	R/W	Enables the accelerometer data-ready interrupt on the INT1 pin. Default value: 0

7.12 INT_CTRL3 (0Eh - R/W)

Interrupt control register 3

Table 44. **INT_CTRL3 register**

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
INT2_ COUNTER_BDR	INT2_ FIFO_FULL	INT2_ FIFO_OVR	INT2_ FIFO_TH	0(1)	INT2_ SLEEPCNT	0(1)	INT2_ DRDY_XL

1. *This bit must be set to the value indicated for the correct operation of the device.*

Table 45. **INT_CTRL3 register description**

INT2_COUNTER_BDR	0x0	R/W	Enables BDR counter interrupt on the INT2 pin. Default value: 0.
INT2_FIFO_FULL	0x0	R/W	Enables FIFO full interrupt on the INT2 pin. Default value: 0
INT2_FIFO_OVR	0x0	R/W	Enables FIFO overrun interrupt on the INT2 pin. Default value: 0
INT2_FIFO_TH	0x0	R/W	Enables FIFO threshold interrupt on the INT2 pin. Default value: 0
INT2_SLEEPCNT	0x0	R/W	Enables the sleep counter interrupt on the INT2 pin.
INT2_DRDY_XL	0x0	R/W	Enables the accelerometer data-ready interrupt on the INT2 pin. Default value: 0

7.13 WHO_AM_I (0Fh - R)

Who am I register. Its value is fixed at 50h.

Table 46. **WHO_AM_I register**

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
0	1	0	1	0	0	0	0

7.14 CTRL1 (10h - R/W)

Device control register

Table 47. CTRL1 register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
BURST_CFG_3	BURST_CFG_2	BURST_CFG_1	BURST_CFG_0	ODR_XL_3	ODR_XL_2	ODR_XL_1	ODR_XL_0

Table 48. CTRL1 register description

BURST_CFG_[3:0]	0x0	R/W	Burst mode selection The default value is 0x0 and it sets the device in continuous mode. Any other configuration sets the device in burst mode, refer to Table 49. BURST_CFG must be programmed before the ODR setting ODR_XL. Otherwise, the BURST_CFG write transaction is ignored. It is mandatory to change BURST_CFG only when ODR_XL = 0x0.
ODR_XL_[3:0]	0x0	R/W	Accelerometer output data rate (ODR) selection The default value is 0x0 and it sets the device in idle mode, refer to Table 50. Any other configuration enables the sensor operations with an output data rate (ODR) according to Table 50.

Table 49. Burst configuration

BURST_CFG_3	BURST_CFG_2	BURST_CFG_1	BURST_CFG_0	Mode	Trigger on	Trigger off
0	0	0	0	Continous mode	-	-
0	0	0	1	Burst mode	User from interface	ISPU
0	0	1	0		User from interface	FIFO
0	1	0	0		Sleep counter	User from interface
0	1	0	1		Sleep counter	ISPU
0	1	1	0		Sleep counter	FIFO
1	0	0	0		External trigger	User from interface
1	0	0	1		External trigger	ISPU
1	0	1	0		External trigger	FIFO
Other values				Idle mode	-	-

Table 50. ODR configuration

ODR_XL_3	ODR_XL_2	ODR_XL_1	ODR_XL_0	ODR
0	0	0	0	Idle mode
0	0	0	1	Reserved
0	0	1	0	2.5 kHz
0	0	1	1	5.0 kHz
0	1	0	0	10 kHz
0	1	0	1	20 kHz
0	1	1	0	40 kHz
0	1	1	1	80 kHz
1	x	x	x	Reserved

7.15 CTRL2 (11h - R/W)

Device control register

Table 51. CTRL2 register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
0	FS_XL_1	FS_XL_0	0(1)	0(1)	0(1)	0(1)	0(1)

1. *This bit must be set to the value indicated for the correct operation of the device.*

Table 52. CTRL2 register description

FS_XL_[1:0]	0x0	R/W	Vibration sensor full-scale selection. Default 0x00 (00: ±50 *g*; 01: ±100 *g*; 10: ±200 *g*; 11: reserved)

7.16 CTRL3 (12h - R/W)

Device control register

Table 53. CTRL3 register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
BOOT	BDU	BURST_ FORCE_TRG	FIFO_EN	ISPU_EN	IF_INC	0(1)	SW_RESET

1. *This bit must be set to the value indicated for the correct operation of the device.*

Table 54. CTRL3 register description

BOOT	0x0	R/W	Reloads the OTP content. This bit is automatically cleared. (0: normal mode; 1: reloads memory content)
BDU	0x0	R/W	Enables the block data update feature, which allows reading output data asynchronously, ensuring that all the bytes read are from the same sample. Supported outputs are: • OUTX,Y,Z • OUT_TEMP • TIMESTAMP • ISPU_DOUT_00,02,04,06,08,10,12,14 (working on data of 4 bytes), only if ISPU_BDU is set to 1.
BURST_FORCE_TRG	0x0	R/W	Burst mode trigger command from the user. This is needed to trigger on or trigger off the sensor in burst mode, refer to BURST_CFG_[3:0] and burst mode.
FIFO_EN	0x0	R/W	Powers up the FIFO. Applicable only in idle mode. This bit must be asserted if any FIFO operation is needed. After asserting the FIFO_EN bit, wait at least 1.5 ms before enabling the device (setting ODR_XL to a value other than 0).
ISPU_EN	0x0	R/W	Powers up the ISPU. Applicable only in idle mode. This bit must be asserted if any operation involving the ISPU is needed. After asserting the ISPU_EN bit, wait at least 1.5 ms before enabling the device (setting ODR_XL to a value other than 0). It is mandatory to assert also the FIFO_EN bit.
IF_INC	0x1	R/W	Enables the register address to be automatically incremented during a multiple-byte access with a serial interface
SW_RESET	0x0	R/W	Software reset, which resets registers to their default value The following registers are NOT restored to their default value: • PIN_CTRL • IF_CFG • I3C_CTRL • SPI_CTRL

7.17 CTRL4 (13h - R/W)

Device control register

Table 55. CTRL4 register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
ROUNDING_1	ROUNDING_0	0	Z_AXIS_ ENABLE	Y_AXIS_ ENABLE	X_AXIS_ ENABLE	0(1)	INT2_IN_LH

1. *This bit must be set to the value indicated for the correct operation of the device.*

Table 56. CTRL4 register description

ROUNDING_[1:0]	0x0	R/W	(00: wraparound disabled in output data registers; 01: wraparound enabled from OUTZ_HH_A to OUTX_L_A; 10: wraparound enabled on accelerometer data in 16b mode: from OUTX_M to OUTY_L, from OUTY_M to OUTZ_L, from OUTZ_M to OUTX_L. Not compatible with XL stored in FIFO; 11: reserved)
Z_AXIS_ENABLE	0x1	R/W	When asserted, Z-axis sensing is enabled. Default: 0x1
Y_AXIS_ENABLE	0x1	R/W	When asserted, Y-axis sensing is enabled. Default: 0x1
X_AXIS_ENABLE	0x1	R/W	When asserted, X-axis sensing is enabled. Default: 0x1
INT2_IN_LH	0x0	R/W	If set to 1, the INT2 digital input pin is inverted before being used, enabling the falling-edge detection on the external triggers (for burst modes or the ISPU).

7.18 I3C_CTRL (14h - R/W)

I3C control register

Table 57. I3C_CTRL register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
0(1)	0(1)	0(1)	0(1)	0(1)	BUS_ACT_ SEL_1	BUS_ACT_ SEL_0	INT_EN_I3C

1. *This bit must be set to the value indicated for the correct operation of the device.*

Table 58. I3C_CTRL register description

BUS_ACT_ SEL_[1:0]	0x00	R/W	This bit is used to select the bus available time when I3C IBI is used. Default value: 0x00 (00: 2 µs (default); 01: 12.5 µs; 10: 25 µs; 11: 50 µs)
INT_EN_I3C	0x0	R/W	Enables the INT1/INT2 pins when I3C is enabled

7.19 SPI_CTRL (15h - R/W)

SPI control register

Table 59. SPI_CTRL register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
0(1)	0(1)	0(1)	0(1)	0(1)	0(1)	1(1)	SIM

1. *This bit must be set to the value indicated for the correct operation of the device.*

Table 60. SPI_CTRL register description

SIM	0x0	R/W	Enables SPI 3-wire mode

7.20 ST_CTRL (17h - R/W)

Self-test control register

Table 61. ST_CTRL register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
LPF1_CFG_3	LPF1_CFG_2	LPF1_CFG_1	LPF1_CFG_0	0(1)	ST_EN_1	ST_EN_0	ST_SIGN

1. *This bit must be set to the value indicated for the correct operation of the device.*

Table 62. ST_CTRL register description

LPF1_CFG_[3:0]	0x0	R/W	Configuration of the low-pass filter LPF1. Refer to Table 63. LPF1 bandwidth selection.
ST_EN_[1:0]	0x0	R/W	(00: self-test disabled; 01: reserved; 10: reserved; 11: self-test enabled)
ST_SIGN	0x0	R/W	Defines self-test sign (0: positive; 1: negative)

Table 63. LPF1 bandwidth selection

LPF1_CFG_[3:0]	LPF1 bandwidth (kHz)	ODR [Hz]
0000	20.0	40 kHz & 80 kHz
0001	17.2	20 kHz
0010	14.7	-
0011	12.0	-
0100	9.32	10 kHz
0101	6.89	-
0110	4.55	5 kHz
0111	2.56	2.5 kHz
1000	Autoswitch with ODR (follow ODR column)	-
1001-1111	Reserved	-

7.21 ISPU_CTRL1 (18h - R/W)

ISPU control register

Table 64. ISPU_CTRL1 register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
ISPU_RATE_3	ISPU_RATE_2	ISPU_RATE_1	ISPU_RATE_0	REG_ACCESS_CONFIRM	LOPRIO_USER_TRG	0	ISPU_BDU

Table 65. ISPU_CTRL1 register description

ISPU_RATE_[3:0]	0x0	R/W	ODR setting for a dedicated signal path for the ISPU. Refer to Table 50. ODR configuration. Note that this ODR setting is independent from the ODR sensor. However, this register does not act on sensor power-on, which is still driven by the CTRL1 register only. Refer to CTRL1 (10h - R/W).
REG_ACCESS_CONFIRM	0x0	R/W	Can be used to confirm to the ISPU that it can take write access to the sensor register map.
LOPRIO_USER_TRG	0x0	R/W	ISPU low-priority interrupt trigger from the user interface. If asserted, the low-priority firmware routine is executed.
ISPU_BDU	0x0	R/W	Enables the BDU for ISPU. When set to 1, the ISPU_DOUT registers can be safely read using the block data update feature. Note that CTRL3[6] = 1 must be asserted and the firmware must write words based on 32 bits in the ISPU_DOUT registers.

7.22 ISPU_CTRL2 (19h - R/W)

ISPU control register

Table 66. ISPU_CTRL2 register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
0(1)	0(1)	TIMESTAMP_EN	SW_RESET_ISPU	0(1)	0(1)	0(1)	0(1)

1. *This bit must be set to the value indicated for the correct operation of the device.*

Table 67. ISPU_CTRL2 register

TIMESTAMP_EN	0x0	R/W	Enables timestamp
SW_RESET_ISPU	0x0	R/W	If asserted, the ISPU is reset. All internal logic is reset to the POR (power-on reset) values. This register must be de-asserted by the user

7.23 ISPU_INT_STATUS1_MAINPAGE (1Ah - R)

Interrupt status register

Table 68. ISPU_INT_STATUS1_MAINPAGE register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
ISPU_IA_7	ISPU_IA_6	ISPU_IA_5	ISPU_IA_4	ISPU_IA_3	ISPU_IA_2	ISPU_IA_1	ISPU_IA_0

Table 69. ISPU_INT_STATUS1_MAINPAGE register

ISPU_IA_[7:0]	0x00	R	Interrupt status from ISPU Algorithms running on the ISPU can generate interrupts according to their needs in latched or pulsed mode. Interrupt events are stored inside this register.

7.24 ISPU_INT_STATUS2_MAINPAGE (1Bh - R)

Interrupt status register

Table 70. ISPU_INT_STATUS2_MAINPAGE register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
ISPU_IA_15	ISPU_IA_14	ISPU_IA_13	ISPU_IA_12	ISPU_IA_11	ISPU_IA_10	ISPU_IA_9	ISPU_IA_8

Table 71. ISPU_INT_STATUS2_MAINPAGE register description

ISPU_IA_[15:8]	0x00	R	Interrupt status from the ISPU Algorithms running on the ISPU can generate interrupts according to their needs in latched or pulsed mode. Interrupt events are stored inside this register.

7.25 STATUS_REG (1Eh - R)

Status register

Table 72. STATUS_REG register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
TIMESTAMP_ ENDCOUNT	ISPU_IA	0	EXT_TRIG_IA	SLEEPCNT_IA	TDA	0	XLDA

Table 73. STATUS_REG register description

TIMESTAMP_ENDCOUNT	0x0	R	Timestamp end count reached
ISPU_IA	0x0	R	ISPU global interrupt generated
EXT_TRIG_IA	0x0	R	External trigger interrupt from the INT2 pin
SLEEPCNT_IA	0x0	R	Sleep counter interrupt
TDA	0x0	R	Temperature sensor new data available
XLDA	0x0	R	Accelerometer new data available

7.26 TPF (20h – R)

Testing related flags

Table 74. TPF register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
0	0	0	0	TPF_3	TPF_2	TPF_1	TPF_0

Table 75. TPF register description

TPF_[3:0]	-	R	Testing related flags

7.27 DEVICE_STATUS (21h - R)

Table 76. DEVICE_STATUS register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
ISPU_CORE_SLEEP	ISPU_CTRL_ACCESS	0	0	0	0	0	0

Table 77. DEVICE_STATUS register description

ISPU_CORE_SLEEP	0x0	R	If high, the ISPU core is in sleep.
ISPU_CTRL_ACCESS	0x0	R	If high, the ISPU firmware has control of the register map access in write mode. In this case, the interface cannot write to the register map.

7.28 OUT_TEMP_H (23h - R) and OUT_TEMP_L (22h - R)

Temperature sensor output data

Table 78. OUT_TEMP_H register

bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
OUT_TEMP_H_7	OUT_TEMP_H_6	OUT_TEMP_H_5	OUT_TEMP_H_4	OUT_TEMP_H_3	OUT_TEMP_H_2	OUT_TEMP_H_1	OUT_TEMP_H_0

Table 79. OUT_TEMP_L register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
OUT_TEMP_L_7	OUT_TEMP_L_6	OUT_TEMP_L_5	OUT_TEMP_L_4	OUT_TEMP_L_3	OUT_TEMP_L_2	OUT_TEMP_L_1	OUT_TEMP_L_0

Table 80. OUT_TEMP register description

OUT_TEMP_[15:0]	0x0000	R	Temperature sensor data. Output expressed in two's complement.

7.29 OUTX_HH (27h - R), OUTX_H (26h - R), OUTX_M (25h - R), and OUTX_L (24h - R)

Accelerometer sensor output data (X-axis)

Table 81. OUTX_HH register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D31	D30	D29	D28	D27	D26	D25	D24

Table 82. OUTX_H register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D23	D22	D21	D20	D19	D18	D17	D16

Table 83. OUTX_M register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D15	D14	D13	D12	D11	D10	D9	D8

Table 84. OUTX_L register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D7	D6	D5	D4	D3	D2	D1	D0

Table 85. OUTX register description

D[31:0]	0x00000000	R	Accelerometer sensor output data (X-axis) Two's complement, sign-extended 20-bit format or 16-bit if ROUNDING_[1:0] = 10. The digital word (20 or 16 bits) is right-justified.

7.30 OUTY_HH (2Bh - R), OUTY_H (2Ah - R), OUTY_M (29h - R), and OUTY_L (28h - R)

Accelerometer sensor output data (Y-axis)

Table 86. OUTY_HH register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D31	D30	D29	D28	D27	D26	D25	D24

Table 87. OUTY_H register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D23	D22	D21	D20	D19	D18	D17	D16

Table 88. OUTY_M register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D15	D14	D13	D12	D11	D10	D9	D8

Table 89. OUTY_L register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D7	D6	D5	D4	D3	D2	D1	D0

Table 90. OUTY register description

D[31:0]	0x00000000	R	Accelerometer sensor output data (Y-axis) Two's complement, sign-extended 20-bit format or 16-bit if ROUNDING_[1:0] = 10. The digital word (20 or 16 bits) is right-justified.

7.31 OUTZ_HH (2Fh - R), OUTZ_H (2Eh - R), OUTZ_M (2Dh - R), and OUTZ_L (2Ch - R)

Accelerometer sensor output data (Z-axis)

Table 91. OUTZ_HH register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D31	D30	D29	D28	D27	D26	D25	D24

Table 92. OUTZ_H register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D23	D22	D21	D20	D19	D18	D17	D16

Table 93. OUTZ_M register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D15	D14	D13	D12	D11	D10	D9	D8

Table 94. OUTZ_L register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D7	D6	D5	D4	D3	D2	D1	D0

Table 95. OUTZ register description

D[31:0]	0x00000000	R	Accelerometer sensor output data (Z-axis) Two's complement, sign-extended 20-bit format or 16-bit if ROUNDING_[1:0] = 10. The digital word (20 or 16 bits) is right-justified.

7.32 TIMESTAMP4 (34h - R), TIMESTAMP3 (33h - R), TIMESTAMP2 (32h - R), TIMESTAMP1 (31h - R), TIMESTAMP0 (30h - R)

Timestamp register

Table 96. TIMESTAMP4 output registers

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
TS39	TS38	TS37	TS36	TS35	TS34	TS33	TS32

Table 97. TIMESTAMP3 output registers

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
TS31	TS30	TS29	TS28	TS27	TS26	TS25	TS24

Table 98. TIMESTAMP2 output registers

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
TS23	TS22	TS21	TS20	TS19	TS18	TS17	TS16

Table 99. TIMESTAMP1 output registers

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
TS15	TS14	TS13	TS12	TS11	TS10	TS9	TS8

Table 100. TIMESTAMP0 output registers

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
TS7	TS6	TS5	TS4	TS3	TS2	TS1	TS0

Table 101. TIMESTAMP register description

TS[39:0]	0x00000000	R	Timestamp register. 1 LSB = 25 µs

7.33 SLEEPCNT_CFG (38h - R/W)

Sleep counter configuration register

Table 102. SLEEPCNT_CFG register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
0(1)	0(1)	0(1)	0(1)	RST_SLEEP CNT	0(1)	ENABLE_ SLEEPCNT	TICK_SEL

1. *This bit must be set to the value indicated for the correct operation of the device.*

Table 103. SLEEPCNT_CFG register

RST_SLEEPCNT	0x0	R/W	If asserted, the sleep counter is reset
ENABLE_SLEEPCNT	0x0	R/W	(0: sleep counter is frozen; 1: sleep counter is counting) It is possible to enable and use the sleep counter in idle mode, without any other sensor setting.
TICK_SEL	0x0	R/W	Timescale for the sleep counter (0: slow timescale; 1: fast timescale)

7.34 ISPU_LOPRIO_EN (39h - R/W)

Low-priority interrupts configuration register

Table 104. ISPU_LOPRIO_EN register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
0(1)	0(1)	LOPRIO_ TEMP_EN	LOPRIO_ USER_TRG_EN	LOPRIO_ EXT_TRG_EN	0	LOPRIO_ SLEEPCNT_EN	LOPRIO_ FIFO_EN

1. *This bit must be set to the value indicated for the correct operation of the device.*

Table 105. ISPU_LOPRIO_EN register description

LOPRIO_TEMP_EN	0x0	R/W	Enables the temperature data-ready to trigger the low-priority routine on the ISPU.
LOPRIO_USER_TRG_EN	0x0	R/W	Enables the user to trigger the low-priority routine on the ISPU.
LOPRIO_EXT_TRG_EN	0x0	R/W	Enables the external signal on the INT2 pin to trigger the low-priority routine on the ISPU.
LOPRIO_SLEEPCNT_EN	0x0	R/W	Enables the sleep counter to trigger the low-priority routine on the ISPU.
LOPRIO_FIFO_EN	0x0	R/W	Enables the FIFO to trigger the low-priority routine on the ISPU.

7.35 SLEEPCNT_TH_H (3Fh R/W) and SLEEPCNT_TH_L (3Eh R/W)

Sleep counter threshold value

Table 106. SLEEPCNT_TH_H register

bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
0	SLEEPCNT_TH_H_6	SLEEPCNT_TH_H_5	SLEEPCNT_TH_H_4	SLEEPCNT_TH_H_3	SLEEPCNT_TH_H_2	SLEEPCNT_TH_H_1	SLEEPCNT_TH_H_0

Table 107. SLEEPCNT_TH_L register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
SLEEPCNT_TH_L_7	SLEEPCNT_TH_L_6	SLEEPCNT_TH_L_5	SLEEPCNT_TH_L_4	SLEEPCNT_TH_L_3	SLEEPCNT_TH_L_2	SLEEPCNT_TH_L_1	SLEEPCNT_TH_L_0

Table 108. SLEEPCNT_TH register description

SLEEPCNT_TH_[14:0]	0x0000	R	Sleep counter threshold When the sleep counter reaches the value stored in this register, an interrupt or a trigger event can be generated. For the timescale, refer to Table 14. Timescale for the sleep counter.

7.36 FIFO_DATA_OUT_TAG (40h - R)

Table 109. FIFO_DATA_OUT_TAG register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
TAG_SENSOR_7	TAG_SENSOR_6	TAG_SENSOR_5	TAG_SENSOR_4	TAG_SENSOR_3	TAG_SENSOR_2	TAG_SENSOR_1	TAG_SENSOR_0

Table 110. FIFO_DATA_OUT_TAG register description

TAG_SENSOR_[7:0]	0x00	R	Identifies data type. Refer to Table 111.

Table 111. FIFO TAG names

TAG name	TAG value	Data batched in FIFO
TAG_EMPTY	00000000	FIFO empty condition
TAG_XL	00010000	24-bit (20-bit sign-extended) X-axis + 2-bit (20-bit sign-extended) Y-axis + 24-bit (20-bit sign-extended) Z-axis
TAG_TEMP	00011000	16-bit temperature sensor data
TAG_TS	00100000	40-bit timestamp data
TAG_ISPU	8 bits defined by the ISPU	72-bit ISPU results

7.37 FIFO_DATA_OUT_D2 (43h R), FIFO_DATA_OUT_D1 (42h R), and FIFO_DATA_OUT_D0 (41h R)

FIFO data output X

Table 112. FIFO_DATA_OUT_D2 registers

bit7	bit6	bit6	bit6	bit6	bit6	bit6	bit6
D23	D22	D21	D20	D19	D18	D17	D16

Table 113. FIFO_DATA_OUT_D1 registers

bit7	bit6	bit6	bit6	bit6	bit6	bit6	bit6
D15	D14	D13	D12	D11	D10	D9	D8

Table 114. FIFO_DATA_OUT_D0 registers

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D7	D6	D5	D4	D3	D2	D1	D0

Table 115. FIFO_DATA_OUT register description

D[23:0]	0x000000000000	R	FIFO data out The FIFO DATA OUT registers must be read in a multiple-byte read from FIFO_DATA_OUT_TAG to FIFO_DATA_OUT_D8.

7.38 FIFO_DATA_OUT_D5 (46h R), FIFO_DATA_OUT_D4 (45h R), and FIFO_DATA_OUT_D3 (44h R)

FIFO data output Y

Table 116. FIFO_DATA_OUT_D5 registers

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D23	D22	D21	D20	D19	D18	D17	D16

Table 117. FIFO_DATA_OUT_D4 registers

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D15	D14	D13	D12	D11	D10	D9	D8

Table 118. FIFO_DATA_OUT_D3 registers

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D7	D6	D5	D4	D3	D2	D1	D0

Table 119. FIFO_DATA_OUT register description

D[23:0]	0x000000000000	R	FIFO data out The FIFO DATA OUT registers must be read in a multiple-byte read from FIFO_DATA_OUT_TAG to FIFO_DATA_OUT_D8.

7.39 FIFO_DATA_OUT_D8 (49h R), FIFO_DATA_OUT_D7 (48h R), and FIFO_DATA_OUT_D6 (47h R)

FIFO data output Z

Table 120. FIFO_DATA_OUT_8 register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D23	D22	D21	D20	D19	D18	D17	D16

Table 121. FIFO_DATA_OUT_7 register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D15	D14	D13	D12	D11	D10	D9	D8

Table 122. FIFO_DATA_OUT_6 register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D7	D6	D5	D4	D3	D2	D1	D0

Table 123. FIFO_DATA_OUT register description

D[23:0]	0x000000000000	R	FIFO data out The FIFO DATA OUT registers must be read in a multiple-byte read from FIFO_DATA_OUT_TAG to FIFO_DATA_OUT_D8.

7.40 FIFO_STATUS_1 (4Ch - R)

FIFO status register

Table 124. FIFO_STATUS_1 register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
DIFF_FIFO_7	DIFF_FIFO_6	DIFF_FIFO_5	DIFF_FIFO_4	DIFF_FIFO_3	DIFF_FIFO_2	DIFF_FIFO_1	DIFF_FIFO_0

Table 125. FIFO_STATUS_1 register description

DIFF_FIFO_[7:0]	0x00	R	Number of unread sensor data (TAG + 9 bytes) stored in FIFO

7.41 FIFO_STATUS_2 (4Dh - R)

FIFO status register

Table 126. FIFO_STATUS_2 register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
FIFO_ WTM_IA	FIFO_ OVR_IA	COUNTER_ BDR_IA	FIFO_ FULL_IA	DIFF_FIFO_11	DIFF_FIFO_10	DIFF_FIFO_9	DIFF_FIFO_8

Table 127. FIFO_STATUS_2 register description

FIFO_WTM_IA	0x0	R	FIFO watermark status (0: FIFO filling is lower than the watermark; 1: FIFO filling is equal or higher than the watermark)
FIFO_OVR_IA	0x0	R	FIFO overrun status If set, this bit indicates that FIFO has overwritten data.
COUNTER_BDR_IA	0x0	R	When the counter BDR reaches the CNT_BDR_TH[11:0] threshold. This bit is reset when this register is read.
FIFO_FULL_IA	0x0	R	FIFO full status If set, this bit indicates that FIFO will be full at the next ODR.
DIFF_FIFO_[11:8]	0x00	R	Number of unread sensor data (TAG + 9 bytes) stored in FIFO

7.42 INTERNAL_FREQ_FINE (50h - R)

ODR fine-tuning register

Table 128. INTERNAL_FREQ_FINE register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
INTERNAL_ FREQ_FINE_7	INTERNAL_ FREQ_FINE_6	INTERNAL_ FREQ_FINE_5	INTERNAL_ FREQ_FINE_4	INTERNAL_ FREQ_FINE_3	INTERNAL_ FREQ_FINE_2	INTERNAL_ FREQ_FINE_1	INTERNAL_ FREQ_FINE_0

Table 129. INTERNAL_FREQ_FINE register description

INTERNAL_FREQ_FINE_[7:0]	0x00	R	Difference (in percentage) of the real ODR (and timestamp rate) with respect to the ideal one. The content of this register is accurate and can be used only if the bits [3:0] of TPF (20h – R) are not all 0. Step: 1 LSB = 1%/128 = 0.0078125% ODR accuracy, in two's complement

7.43 ISPU_DUMMY_CFG_1 (52h - R/W) through ISPU_DUMMY_CFG_16 (61h - R/W)

General-purpose registers

Table 130. ISPU_DUMMY_CFG_1 register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
ISPU_DUMMY_CFG_7	ISPU_DUMMY_CFG_6	ISPU_DUMMY_CFG_5	ISPU_DUMMY_CFG_4	ISPU_DUMMY_CFG_3	ISPU_DUMMY_CFG_2	ISPU_DUMMY_CFG_1	ISPU_DUMMY_CFG_0

Table 131. ISPU_DUMMY_CFG_1 register description

ISPU_DUMMY_CFG_[7:0]	0x00	R	General-purpose registers readable by the ISPU. Registers available for user-defined usage of the ISPU.

Table 132. ISPU_DUMMY_CFG_2 register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
ISPU_DUMMY_CFG_7	ISPU_DUMMY_CFG_6	ISPU_DUMMY_CFG_5	ISPU_DUMMY_CFG_4	ISPU_DUMMY_CFG_3	ISPU_DUMMY_CFG_2	ISPU_DUMMY_CFG_1	ISPU_DUMMY_CFG_0

Table 133. ISPU_DUMMY_CFG_2 register description

ISPU_DUMMY_CFG_[7:0]	0x00	R	General-purpose registers readable by the ISPU. Registers available for user-defined usage of the ISPU.

Table 134. ISPU_DUMMY_CFG_3 register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
ISPU_DUMMY_CFG_7	ISPU_DUMMY_CFG_6	ISPU_DUMMY_CFG_5	ISPU_DUMMY_CFG_4	ISPU_DUMMY_CFG_3	ISPU_DUMMY_CFG_2	ISPU_DUMMY_CFG_1	ISPU_DUMMY_CFG_0

Table 135. ISPU_DUMMY_CFG_3 register description

ISPU_DUMMY_CFG_[7:0]	0x00	R	General-purpose registers readable by the ISPU. Registers available for user-defined usage of the ISPU.

Table 136. ISPU_DUMMY_CFG_4 register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
ISPU_DUMMY_CFG_7	ISPU_DUMMY_CFG_6	ISPU_DUMMY_CFG_5	ISPU_DUMMY_CFG_4	ISPU_DUMMY_CFG_3	ISPU_DUMMY_CFG_2	ISPU_DUMMY_CFG_1	ISPU_DUMMY_CFG_0

Table 137. ISPU_DUMMY_CFG_4 register description

ISPU_DUMMY_CFG_[7:0]	0x00	R	General-purpose registers readable by the ISPU. Registers available for user-defined usage of the ISPU.

Table 138. ISPU_DUMMY_CFG_5 register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
ISPU_DUMMY_CFG_7	ISPU_DUMMY_CFG_6	ISPU_DUMMY_CFG_5	ISPU_DUMMY_CFG_4	ISPU_DUMMY_CFG_3	ISPU_DUMMY_CFG_2	ISPU_DUMMY_CFG_1	ISPU_DUMMY_CFG_0

Table 139. ISPU_DUMMY_CFG_5 register description

ISPU_DUMMY_CFG_[7:0]	0x00	R	General-purpose registers readable by the ISPU. Registers available for user-defined usage of the ISPU.

Table 140. **ISPU_DUMMY_CFG_6 register**

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
ISPU_DUMMY_CFG_7	ISPU_DUMMY_CFG_6	ISPU_DUMMY_CFG_5	ISPU_DUMMY_CFG_4	ISPU_DUMMY_CFG_3	ISPU_DUMMY_CFG_2	ISPU_DUMMY_CFG_1	ISPU_DUMMY_CFG_0

Table 141. **ISPU_DUMMY_CFG_6 register description**

ISPU_DUMMY_CFG_[7:0]	0x00	R	General-purpose registers readable by the ISPU. Registers available for user-defined usage of the ISPU.

Table 142. **ISPU_DUMMY_CFG_7 register**

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
ISPU_DUMMY_CFG_7	ISPU_DUMMY_CFG_6	ISPU_DUMMY_CFG_5	ISPU_DUMMY_CFG_4	ISPU_DUMMY_CFG_3	ISPU_DUMMY_CFG_2	ISPU_DUMMY_CFG_1	ISPU_DUMMY_CFG_0

Table 143. **ISPU_DUMMY_CFG_7 register description**

ISPU_DUMMY_CFG_[7:0]	0x00	R	General-purpose registers readable by the ISPU. Registers available for user-defined usage of the ISPU.

Table 144. **ISPU_DUMMY_CFG_8 register**

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
ISPU_DUMMY_CFG_7	ISPU_DUMMY_CFG_6	ISPU_DUMMY_CFG_5	ISPU_DUMMY_CFG_4	ISPU_DUMMY_CFG_3	ISPU_DUMMY_CFG_2	ISPU_DUMMY_CFG_1	ISPU_DUMMY_CFG_0

Table 145. **ISPU_DUMMY_CFG_8 register description**

ISPU_DUMMY_CFG_[7:0]	0x00	R	General-purpose registers readable by the ISPU. Registers available for user-defined usage of the ISPU.

Table 146. **ISPU_DUMMY_CFG_9 register**

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
ISPU_DUMMY_CFG_7	ISPU_DUMMY_CFG_6	ISPU_DUMMY_CFG_5	ISPU_DUMMY_CFG_4	ISPU_DUMMY_CFG_3	ISPU_DUMMY_CFG_2	ISPU_DUMMY_CFG_1	ISPU_DUMMY_CFG_0

Table 147. **ISPU_DUMMY_CFG_9 register description**

ISPU_DUMMY_CFG_[7:0]	0x00	R	General-purpose registers readable by the ISPU. Registers available for user-defined usage of the ISPU.

Table 148. **ISPU_DUMMY_CFG_10 register**

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
ISPU_DUMMY_CFG_7	ISPU_DUMMY_CFG_6	ISPU_DUMMY_CFG_5	ISPU_DUMMY_CFG_4	ISPU_DUMMY_CFG_3	ISPU_DUMMY_CFG_2	ISPU_DUMMY_CFG_1	ISPU_DUMMY_CFG_0

Table 149. **ISPU_DUMMY_CFG_10 register description**

ISPU_DUMMY_CFG_[7:0]	0x00	R	General-purpose registers readable by the ISPU. Registers available for user-defined usage of the ISPU.

Table 150. **ISPU_DUMMY_CFG_11 register**

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
ISPU_DUMMY_CFG_7	ISPU_DUMMY_CFG_6	ISPU_DUMMY_CFG_5	ISPU_DUMMY_CFG_4	ISPU_DUMMY_CFG_3	ISPU_DUMMY_CFG_2	ISPU_DUMMY_CFG_1	ISPU_DUMMY_CFG_0

Table 151. **ISPU_DUMMY_CFG_11 register description**

ISPU_DUMMY_CFG_[7:0]	0x00	R	General-purpose registers readable by the ISPU. Registers available for user-defined usage of the ISPU.

Table 152. **ISPU_DUMMY_CFG_12 register**

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
ISPU_DUMMY_CFG_7	ISPU_DUMMY_CFG_6	ISPU_DUMMY_CFG_5	ISPU_DUMMY_CFG_4	ISPU_DUMMY_CFG_3	ISPU_DUMMY_CFG_2	ISPU_DUMMY_CFG_1	ISPU_DUMMY_CFG_0

Table 153. **ISPU_DUMMY_CFG_12 register description**

ISPU_DUMMY_CFG_[7:0]	0x00	R	General-purpose registers readable by the ISPU. Registers available for user-defined usage of the ISPU.

Table 154. **ISPU_DUMMY_CFG_13 register**

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
ISPU_DUMMY_CFG_7	ISPU_DUMMY_CFG_6	ISPU_DUMMY_CFG_5	ISPU_DUMMY_CFG_4	ISPU_DUMMY_CFG_3	ISPU_DUMMY_CFG_2	ISPU_DUMMY_CFG_1	ISPU_DUMMY_CFG_0

Table 155. **ISPU_DUMMY_CFG_13 register description**

ISPU_DUMMY_CFG_[7:0]	0x00	R	General-purpose registers readable by the ISPU. Registers available for user-defined usage of the ISPU.

Table 156. **ISPU_DUMMY_CFG_14 register**

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
ISPU_DUMMY_CFG_7	ISPU_DUMMY_CFG_6	ISPU_DUMMY_CFG_5	ISPU_DUMMY_CFG_4	ISPU_DUMMY_CFG_3	ISPU_DUMMY_CFG_2	ISPU_DUMMY_CFG_1	ISPU_DUMMY_CFG_0

Table 157. **ISPU_DUMMY_CFG_14 register description**

ISPU_DUMMY_CFG_[7:0]	0x00	R	General-purpose registers readable by the ISPU. Registers available for user-defined usage of the ISPU.

Table 158. **ISPU_DUMMY_CFG_15 register**

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
ISPU_DUMMY_CFG_7	ISPU_DUMMY_CFG_6	ISPU_DUMMY_CFG_5	ISPU_DUMMY_CFG_4	ISPU_DUMMY_CFG_3	ISPU_DUMMY_CFG_2	ISPU_DUMMY_CFG_1	ISPU_DUMMY_CFG_0

Table 159. **ISPU_DUMMY_CFG_15 register description**

ISPU_DUMMY_CFG_[7:0]	0x00	R	General-purpose registers readable by the ISPU. Registers available for user-defined usage of the ISPU.

Table 160. **ISPU_DUMMY_CFG_16 register**

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
ISPU_DUMMY_CFG_7	ISPU_DUMMY_CFG_6	ISPU_DUMMY_CFG_5	ISPU_DUMMY_CFG_4	ISPU_DUMMY_CFG_3	ISPU_DUMMY_CFG_2	ISPU_DUMMY_CFG_1	ISPU_DUMMY_CFG_0

Table 161. **ISPU_DUMMY_CFG_16 register description**

ISPU_DUMMY_CFG_[7:0]	0x00	R	General-purpose registers readable by the ISPU. Registers available for user-defined usage of the ISPU.

7.44 SLEEPCNT_TIME_H (63h R) and SLEEPCNT_TIME_L (62h R)

Table 162. **SLEEPCNT_TIME_H register**

bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
0	SLEEPCNT_TIME_H_6	SLEEPCNT_TIME_H_5	SLEEPCNT_TIME_H_4	SLEEPCNT_TIME_H_3	SLEEPCNT_TIME_H_2	SLEEPCNT_TIME_H_1	SLEEPCNT_TIME_H_0

Table 163. **SLEEPCNT_TIME_L register**

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
SLEEPCNT_TIME_L_7	SLEEPCNT_TIME_L_6	SLEEPCNT_TIME_L_5	SLEEPCNT_TIME_L_4	SLEEPCNT_TIME_L_3	SLEEPCNT_TIME_L_2	SLEEPCNT_TIME_L_1	SLEEPCNT_TIME_L_0

Table 164. **SLEEPCNT_TIME register**

SLEEPCNT_TIME_[14:0]	0x0000	R	Sleep counter current value

7.45 INT_CTRL4 (6Eh - R/W)

Interrupt control register 3

Table 165. **INT_CTRL4 register**

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
0(1)	0(1)	0(1)	0(1)	INT1_TIMESTAMP	INT2_TIMESTAMP	INT1_ISPU	INT2_ISPU

1. *This bit must be set to the value indicated for the correct operation of the device.*

Table 166. **INT_CTRL4 register description**

INT1_TIMESTAMP	0x0	R/W	Enables TIMESTAMP_IA to be routed to the INT1 pin
INT2_TIMESTAMP	0x0	R/W	Enables TIMESTAMP_IA to be routed to the INT2 pin
INT1_ISPU	0x0	R/W	Enables mask for INT1 ISPU interrupt
INT2_ISPU	0x0	R/W	Enables mask for INT2 ISPU interrupt

8 ISPU register map (accessible from the host serial interface)

The table given below provides a list of the 8-bit registers embedded in the device and the corresponding addresses.

The register map is organized by pages. This means that different registers can be accessed through the same address by selecting the corresponding page. The page is determined by the value written in the following bits:

- PAGE_SEL in register RAM_ACCESS (01h - R/W), bit 7

The list and the descriptions of the registers in the ISPU register map (PAGE_SEL = 1) are indicated in the following table.

Table 167. ISPU register map

Name	Type	Register address		Default
		Hex	Binary	
ISPU_CONFIG	R/W	02	00000010	00000010
ISPU_STATUS	R	04	00000100	output
ISPU_MEM_SEL	R	07	00000111	output
ISPU_MEMADDR_2	R/W	08	00001000	00000000
ISPU_MEMADDR_1	R/W	09	00001001	00000000
ISPU_MEMDATA_1	R/W	0A	00001010	00000000
ISPU_MEMDATA_2	R	0B	00001011	output
ISPU_IF2S_FLAG_L	R	0C	00001100	output
ISPU_IF2S_FLAG_H	R	0D	00001101	output
ISPU_S2IF_FLAG_L	R	0E	00001110	output
ISPU_S2IF_FLAG_H	R	0F	00001111	output
ISPU_DOUT_00_L	R	10	00010000	output
ISPU_DOUT_00_H	R	11	00010001	output
ISPU_DOUT_01_L	R	12	00010010	output
ISPU_DOUT_01_H	R	13	00010011	output
ISPU_DOUT_02_L	R	14	00010100	output
ISPU_DOUT_02_H	R	15	00010101	output
ISPU_DOUT_03_L	R	16	00010110	output
ISPU_DOUT_03_H	R	17	00010111	output
ISPU_DOUT_04_L	R	18	00011000	output
ISPU_DOUT_04_H	R	19	00011001	output
ISPU_DOUT_05_L	R	1A	00011010	output
ISPU_DOUT_05_H	R	1B	00011011	output
ISPU_DOUT_06_L	R	1C	00011100	output
ISPU_DOUT_06_H	R	1D	00011101	output
ISPU_DOUT_07_L	R	1E	00011110	output
ISPU_DOUT_07_H	R	1F	00011111	output
ISPU_DOUT_08_L	R	20	00100000	output
ISPU_DOUT_08_H	R	21	00100001	output
ISPU_DOUT_09_L	R	22	00100010	output
ISPU_DOUT_09_H	R	23	00100011	output

Name	Type	Register address		Default
		Hex	Binary	
ISPU_DOUT_10_L	R	24	00100100	output
ISPU_DOUT_10_H	R	25	00100101	output
ISPU_DOUT_11_L	R	26	00100110	output
ISPU_DOUT_11_H	R	27	00100111	output
ISPU_DOUT_12_L	R	28	00101000	output
ISPU_DOUT_12_H	R	29	00101001	output
ISPU_DOUT_13_L	R	2A	00101010	output
ISPU_DOUT_13_H	R	2B	00101011	output
ISPU_DOUT_14_L	R	2C	00101100	output
ISPU_DOUT_14_H	R	2D	00101101	output
ISPU_DOUT_15_L	R	2E	00101110	output
ISPU_DOUT_15_H	R	2F	00101111	output
ISPU_INT1_CTRL1	R/W	50	01010000	00000000
ISPU_INT1_CTRL2	R/W	51	01010001	00000000
ISPU_INT2_CTRL1	R/W	54	01010100	00000000
ISPU_INT2_CTRL2	R/W	55	01010101	00000000
ISPU_INT_STATUS1	R	58	01011000	output
ISPU_INT_STATUS2	R	59	01011001	output
ISPU_INT_PIN	R	5C	01011100	output
ISPU_ALGO1	R/W	70	01110000	00000000
ISPU_ALGO2	R/W	71	01110001	00000000
RESERVED	-	72-77		

Reserved registers must not be changed. Writing to those registers may cause permanent damage to the device.

The content of the registers that are loaded at boot should not be changed. They contain the factory calibration values. Their content is automatically restored when the device is powered up.

9 ISPU registers description

The device contains a set of registers that are used to control its behavior and to interact with the ISPU. The register addresses, made up of 7 bits, are used to identify them and to write the data over the serial interface.

9.1 ISPU_CONFIG (02h - R/W)

Table 168. ISPU_CONFIG register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
0	DBG_RST_STS	0	LATCHED_1	LATCHED_0	0	CLK_DIS	RST_N

Table 169. ISPU_CONFIG register description

DBG_RST_STS	0x0	R	(0: the last reset was a power-on reset; 1: the last reset was a JTAG-induced reset)
LATCHED_[1:0]	0x0	R/W	Configures the interrupt generation. (00: interrupt status and pin pulsed; 01: interrupt status latched and interrupt pin pulsed; 1x: interrupt status and pin latched)
CLK_DIS	0x1	R/W	Writing to 0 enables the internal ISPU clock. It is mandatory to start firmware execution.
RST_N	0x0	R/W	Writing to 1 removes the internal ISPU reset and allows firmware execution.

9.2 ISPU_STATUS (04h - R)

Table 170. ISPU_STATUS register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
0	BOOT_END	0	0	0	0	0	ORUN

Table 171. ISPU_STATUS register description

BOOT_END	0x0	R	If high, the ISPU firmware boot has concluded.
ORUN	0x0	R	If high, an overrun event has occurred.

9.3 ISPU_MEM_SEL (07h - RW)

Table 172. ISPU_MEM_SEL register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
0	READ_RAM	0	0	0	ROM_SEL	0	0

Table 173. ISPU_MEM_SEL register description

READ_RAM	0x0	R	Select write to RAM (= 0) or read from RAM (= 1).
ROM_SEL	0x0	R	If 1, access to program RAM. If 0, access to data RAM.

9.4 ISPU_MEM_ADDR_2 (08h R/W), ISPU_MEM_ADDR_1 (09h R/W)

Table 174. ISPU_MEM_ADDR_2 register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
A15	A14	A13	A12	A11	A10	A9	A8

Table 175. ISPU_MEM_ADDR_1 register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
A7	A6	A5	A4	A3	A2	A1	A0

Table 176. ISPU_MEM_ADDR register description

A[15:0]	0x0000	R/W	RAM address to be written/read

9.5 ISPU_MEM_DATA_2 (0Bh R/W), ISPU_MEM_DATA_1 (0Ah R)

Table 177. ISPU_MEM_DATA_2 register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D15	D14	D13	D12	D11	D10	D9	D8

Table 178. ISPU_MEM_DATA_1 register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D7	D6	D5	D4	D3	D2	D1	D0

Table 179. ISPU_MEM_DATA register description

D[15:0]	0x0000	R/W	Bytes to write in RAM if in write mode, data read from RAM in read mode.

9.6 ISPU_IF2S_FLAG_H (0Dh R/W), ISPU_IF2S_FLAG_L (0Ch R/W)

Table 180. ISPU_IF2S_FLAG_H register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D15	D14	D13	D12	D11	D10	D9	D8

Table 181. ISPU_IF2S_FLAG_L register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D7	D6	D5	D4	D3	D2	D1	D0

Table 182. ISPU_IF2S_FLAG register description

D[15:0]	0x0000	R/W	Handshake mechanism. The interface can set these bits, the ISPU can read and clear these bits

9.7 ISPU_S2IF_FLAG_H (0Fh) and ISPU_S2IF_FLAG_L (0Eh)

Table 183. ISPU_S2IF_FLAG_H register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D15	D14	D13	D12	D11	D10	D9	D8

Table 184. ISPU_S2IF_FLAG_L register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D7	D6	D5	D4	D3	D2	D1	D0

Table 185. ISPU_S2IF_FLAG register description

D[15:0]	-	R	Handshake mechanism. The ISPU can set these bits, the serial interface can read and clear these bits.

9.8 ISPU_DOUT_00_H (11h R), ISPU_DOUT_00_L (10h R)

Table 186. ISPU_DOUT_00_H register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D15	D14	D13	D12	D11	D10	D9	D8

Table 187. ISPU_DOUT_00_L register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D7	D6	D5	D4	D3	D2	D1	D0

Table 188. ISPU_DOUT_00 register description

D[15:0]	-	R	General-purpose output driven by the ISPU firmware. They can be combined as desired and used, for example, as 8-bit output, 16-bit output, or 32-bit output data.

9.9 ISPU_DOUT_01_H (13h R), ISPU_DOUT_01_L (12h R)

Table 189. ISPU_DOUT_01_H register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D15	D14	D13	D12	D11	D10	D9	D8

Table 190. ISPU_DOUT_01_L register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D7	D6	D5	D4	D3	D2	D1	D0

Table 191. ISPU_DOUT_01 register description

D[15:0]	-	R	General-purpose output driven by the ISPU firmware. They can be combined as desired and used, for example, as 8-bit output, 16-bit output, or 32-bit output data.

9.10 ISPU_DOUT_02_H (15h R), ISPU_DOUT_02_L (14h R)

Table 192. ISPU_DOUT_02_H register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D15	D14	D13	D12	D11	D10	D9	D8

Table 193. ISPU_DOUT_02_L register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D7	D6	D5	D4	D3	D2	D1	D0

Table 194. ISPU_DOUT_02 register description

D[15:0]	-	R	General-purpose output driven by the ISPU firmware. They can be combined as desired and used, for example, as 8-bit output, 16-bit output, or 32-bit output data.

9.11 ISPU_DOUT_03_H (17h R), ISPU_DOUT_03_L (16h R)

Table 195. ISPU_DOUT_03_H register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D15	D14	D13	D12	D11	D10	D9	D8

Table 196. ISPU_DOUT_03_L register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D7	D6	D5	D4	D3	D2	D1	D0

Table 197. ISPU_DOUT_03 register description

D[15:0]	-	R	General-purpose output driven by the ISPU firmware. They can be combined as desired and used, for example, as 8-bit output, 16-bit output, or 32-bit output data.

9.12 ISPU_DOUT_04_H (19h R), ISPU_DOUT_04_L (18h R)

Table 198. ISPU_DOUT_04_H register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D15	D14	D13	D12	D11	D10	D9	D8

Table 199. ISPU_DOUT_04_L register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D7	D6	D5	D4	D3	D2	D1	D0

Table 200. ISPU_DOUT_04 register description

D[15:0]	-	R	General-purpose output driven by the ISPU firmware. They can be combined as desired and used, for example, as 8-bit output, 16-bit output, or 32-bit output data.

9.13 ISPU_DOUT_05_H (1Bh R), ISPU_DOUT_05_L (1Ah R)

Table 201. ISPU_DOUT_05_H register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D15	D14	D13	D12	D11	D10	D9	D8

Table 202. ISPU_DOUT_05_L register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D7	D6	D5	D4	D3	D2	D1	D0

Table 203. ISPU_DOUT_05 register description

D[15:0]	-	R	General-purpose output driven by the ISPU firmware. They can be combined as desired and used, for example, as 8-bit output, 16-bit output, or 32-bit output data.

9.14 ISPU_DOUT_06_H (1Dh R), ISPU_DOUT_06_L (1Ch R)

Table 204. ISPU_DOUT_06_H register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D15	D14	D13	D12	D11	D10	D9	D8

Table 205. ISPU_DOUT_06_L register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D7	D6	D5	D4	D3	D2	D1	D0

Table 206. ISPU_DOUT_06 register description

D[15:0]	-	R	General-purpose output driven by the ISPU firmware. They can be combined as desired and used, for example, as 8-bit output, 16-bit output, or 32-bit output data.

9.15 ISPU_DOUT_07_H (1Fh R), ISPU_DOUT_07_L (1Eh R)

Table 207. ISPU_DOUT_07_H register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D15	D14	D13	D12	D11	D10	D9	D8

Table 208. ISPU_DOUT_07_L register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D7	D6	D5	D4	D3	D2	D1	D0

Table 209. ISPU_DOUT_07 register description

D[15:0]	-	R	General-purpose output driven by the ISPU firmware. They can be combined as desired and used, for example, as 8-bit output, 16-bit output, or 32-bit output data.

9.16 ISPU_DOUT_08_H (21h R), ISPU_DOUT_08_L (20h R)

Table 210. ISPU_DOUT_08_H register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D15	D14	D13	D12	D11	D10	D9	D8

Table 211. ISPU_DOUT_08_L register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D7	D6	D5	D4	D3	D2	D1	D0

Table 212. ISPU_DOUT_08 register description

D[15:0]	-	R	General-purpose output driven by the ISPU firmware. They can be combined as desired and used, for example, as 8-bit output, 16-bit output, or 32-bit output data.

9.17 ISPU_DOUT_09_H (23h R), ISPU_DOUT_09_L (22h R)

Table 213. ISPU_DOUT_09_H register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D15	D14	D13	D12	D11	D10	D9	D8

Table 214. ISPU_DOUT_09_L register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D7	D6	D5	D4	D3	D2	D1	D0

Table 215. ISPU_DOUT_09 register description

D[15:0]	-	R	General-purpose output driven by the ISPU firmware. They can be combined as desired and used, for example, as 8-bit output, 16-bit output, or 32-bit output data.

9.18 ISPU_DOUT_10_H (25h R), ISPU_DOUT_10_L (24h R)

Table 216. ISPU_DOUT_10_H register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D15	D14	D13	D12	D11	D10	D9	D8

Table 217. ISPU_DOUT_10_L register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D7	D6	D5	D4	D3	D2	D1	D0

Table 218. ISPU_DOUT_10 register description

D[15:0]	-	R	General-purpose output driven by the ISPU firmware. They can be combined as desired and used, for example, as 8-bit output, 16-bit output, or 32-bit output data.

9.19 ISPU_DOUT_11_H (27h R), ISPU_DOUT_11_L (26h R)

Table 219. ISPU_DOUT_11_H register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D15	D14	D13	D12	D11	D10	D9	D8

Table 220. ISPU_DOUT_11_L register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D7	D6	D5	D4	D3	D2	D1	D0

Table 221. ISPU_DOUT_11 register description

D[15:0]	-	R	General-purpose output driven by the ISPU firmware. They can be combined as desired and used, for example, as 8-bit output, 16-bit output, or 32-bit output data.

9.20 ISPU_DOUT_12_H (29h R), ISPU_DOUT_12_L (28h R)

Table 222. ISPU_DOUT_12_H register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D15	D14	D13	D12	D11	D10	D9	D8

Table 223. ISPU_DOUT_12_L register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D7	D6	D5	D4	D3	D2	D1	D0

Table 224. ISPU_DOUT_12 register description

D[15:0]	-	R	General-purpose output driven by the ISPU firmware. They can be combined as desired and used, for example, as 8-bit output, 16-bit output, or 32-bit output data.

9.21 ISPU_DOUT_13_H (2Bh R), ISPU_DOUT_13_L (2Ah R)

Table 225. ISPU_DOUT_13_H register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D15	D14	D13	D12	D11	D10	D9	D8

Table 226. ISPU_DOUT_13_L register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D7	D6	D5	D4	D3	D2	D1	D0

Table 227. ISPU_DOUT_13 register description

D[15:0]	-	R	General-purpose output driven by the ISPU firmware. They can be combined as desired and used, for example, as 8-bit output, 16-bit output, or 32-bit output data.

9.22 ISPU_DOUT_14_H (2Dh R), ISPU_DOUT_14_L (2Ch R)

Table 228. ISPU_DOUT_14_H register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D15	D14	D13	D12	D11	D10	D9	D8

Table 229. ISPU_DOUT_14_L register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D7	D6	D5	D4	D3	D2	D1	D0

Table 230. ISPU_DOUT_14 register description

D[15:0]	-	R	General-purpose output driven by the ISPU firmware. They can be combined as desired and used, for example, as 8-bit output, 16-bit output, or 32-bit output data.

9.23 ISPU_DOUT_15_H (2Fh R), ISPU_DOUT_15_L (2Eh R)

Table 231. ISPU_DOUT_15_H register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D15	D14	D13	D12	D11	D10	D9	D8

Table 232. ISPU_DOUT_15_L register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D7	D6	D5	D4	D3	D2	D1	D0

Table 233. ISPU_DOUT_15 register description

D[15:0]	-	R	General-purpose output driven by the ISPU firmware. They can be combined as desired and used, for example, as 8-bit output, 16-bit output, or 32-bit output data.

9.24 ISPU_INT1_CTRL2 (51h R/W), ISPU_INT1_CTRL1 (50h R/W)

Table 234. ISPU_INT1_CTRL2 register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D15	D14	D13	D12	D11	D10	D9	D8

Table 235. ISPU_INT1_CTRL1 register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D7	D6	D5	D4	D3	D2	D1	D0

Table 236. ISPU_INT1_CTRL register description

D[15:0]	0x0000	R/W	Routes the 16-bit interrupt flags to the INT1 pin

9.25 ISPU_INT2_CTRL2 (55h R/W), ISPU_INT2_CTRL1 (54h R/W)

Table 237. ISPU_INT2_CTRL2 register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D15	D14	D13	D12	D11	D10	D9	D8

Table 238. ISPU_INT2_CTRL1 register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D7	D6	D5	D4	D3	D2	D1	D0

Table 239. ISPU_INT2_CTRL register description

D[15:0]	0x0000	R/W	Routes the 16-bit interrupt flags to the INT2 pin

9.26 ISPU_INT_STATUS2 (59h R), ISPU_INT_STATUS1 (58h R)

Table 240. ISPU_INT_STATUS2 register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D15	D14	D13	D12	D11	D10	D9	D8

Table 241. ISPU_INT_STATUS1 register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D7	D6	D5	D4	D3	D2	D1	D0

Table 242. ISPU_INT_STATUS register description

D[15:0]	-	R	16-bit output interrupt flags from ISPU firmware

9.27 ISPU_INT_PIN (5Ch R)

Table 243. ISPU_INT_PIN register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
0	0	0	0	0	0	INT2	INT1

Table 244. ISPU_INT_PIN register description

INT2	-	R	Flag for interrupt generation on the INT2 pin
INT1	-	R	Flag for interrupt generation on the INT1 pin

9.28 ISPU_ALGO2 (71h R/W) and ISPU_ALGO1 (70h R/W)

Table 245. ISPU_ALGO2 register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
ISPU_ALGO_EN15	ISPU_ALGO_EN14	ISPU_ALGO_EN13	ISPU_ALGO_EN12	ISPU_ALGO_EN11	ISPU_ALGO_EN10	ISPU_ALGO_EN9	ISPU_ALGO_EN8

Table 246. ISPU_ALGO1 register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
ISPU_ALGO_EN7	ISPU_ALGO_EN6	ISPU_ALGO_EN5	ISPU_ALGO_EN4	ISPU_ALGO_EN3	ISPU_ALGO_EN2	ISPU_ALGO_EN1	ISPU_ALGO_EN0

Table 247. ISPU_ALGO register description

ISPU_ALGO_EN[15:0]	0x0000	R/W	Enable configurations in order to run up to 16 independent algorithms

10 ISPU core register access (registers accessible from the core of the ISPU)

The ISPU core can access a significant portion of the sensor's registers in both read and write modes. This capability allows the firmware to read or configure most sensor parameters and make real-time changes to the sensor's operating mode and configuration, adapting it to the context.

The high-level view of the register access memory map of the ISPU core is described in the following table.

From	To	Size	Description	ISPU address space
0x80000	0x800FF	256 Bytes	ISPU system registers	Registers
0x80800	0x808FF	256 Bytes	ISPU registers	Registers
0x80900	0x809FF	256 Bytes	External registers	Registers

11 ISPU core register map

The ISPU contains a set of registers that are used to control its behavior and to interact with the sensor portion of the device. The register addresses indicated in the tables is referred to as the ISPU core address space.

The full maps of the registers accessible from the ISPU core are listed in Table 248. ISPU core register map. Where indicated, some registers are accessible also from the host serial interface, either in the main register page or in the ISPU register page. In this case, the address of the corresponding register through the host serial interface is indicated as well.

For the register description and details of those registers, which are accessible also with the host serial interface, refer to the description in Section 9: ISPU registers description.

The registers only accessible by the ISPU core are detailed in Section 12: ISPU core registers description.

Table 248. ISPU core register map

Register name	ISPU core address (hex)	ISPU core access type	Host IF address (hex)	Host IF access type	Register description
ISPU system registers					
ISPU_LOOP_START	0x80080	R/W	-	-	ISPU core register map - ISPU_LOOP_START (0x80080 - R/W)
ISPU_LOOP_END	0x80084	R/W	-	-	ISPU core register map - ISPU_LOOP_END (0x80084 - R/W)
ISPU_LOOP_CNT	0x80088	R/W	-	-	ISPU core register map - ISPU_LOOP_CNT (0x80088 - R/W)
ISPU_FFT_MATCH_TW	0x8008C	R/W	-	-	ISPU core register map - ISPU_FFT_MATCH_TW (0x8008C - R/W)
ISPU_FFT_START_TW	0x80090	R/W	-	-	ISPU core register map - ISPU_FFT_START_TW (0x80090 - R/W)
ISPU_FFT_END_TW	0x80094	R/W	-	-	ISPU core register map - ISPU_FFT_END_TW (0x80094 - R/W)
ISPU_FFT_CURR_TW1	0x80098	R/W	-	-	ISPU core register map - ISPU_FFT_CURR_TW1 (0x80098 - R/W)
ISPU_FFT_CURR_TW2	0x8009C	R/W	-	-	ISPU core register map - ISPU_FFT_CURR_TW2 (0x8009C - R/W)
ISPU_FFT_STRIDE_TW	0x800A0	R/W	-	-	ISPU core register map - ISPU_FFT_STRIDE_TW (0x800A0 - R/W)
ISPU_FFT_STATE_TW	0x800A4	R/W	-	-	ISPU core register map - ISPU_FFT_STATE_TW (0x800A4 - R/W)
ISPU_FFT_MATCH_D	0x800C0	R/W	-	-	ISPU core register map - ISPU_FFT_MATCH_D (0x800C0 - R/W)
ISPU_FFT_CURR_D1	0x800C4	R/W	-	-	ISPU core register map - ISPU_FFT_CURR_D1 (0x800C4 - R/W)
ISPU_FFT_CURR_D2	0x800C8	R/W	-	-	ISPU core register map - ISPU_FFT_CURR_D2 (0x800C8 - R/W)
ISPU_FFT_TWIDDLE	0x800CC	R/W	-	-	ISPU core register map - ISPU_FFT_TWIDDLE (0x800CC - R/W)
ISPU_FFT_STATE_D	0x800D0	R/W	-	-	ISPU core register map - ISPU_FFT_STATE_D (0x800D0 - R/W)
ISPU_FFT_STATE_BRA	0x800D4	R/W	-	-	ISPU core register map - ISPU_FFT_STATE_BRA (0x800D4 - R/W)
ISPU_FFT_MATCH_BRA	0x800D8	R/W	-	-	ISPU core register map - ISPU_FFT_MATCH_BRA (0x800D8 - R/W)

Register name	ISPU core address (hex)	ISPU core access type	Host IF address (hex)	Host IF access type	Register description
ISPU_FFT_CURR_BRA	0x800DC	R/W	-	-	ISPU core register map - ISPU_FFT_CURR_BRA (0x800DC - R/W)
ISPU_FFT_N_ELEM_BRA	0x800E0	R/W	-	-	ISPU core register map - ISPU_FFT_N_ELEM_BRA (0x800E0 - R/W)
ISPU_CIRCULAR_MATCH	0x800E4	R/W	-	-	ISPU core register map - ISPU_CIRCULAR_MATCH (0x800E4 - R/W)
ISPU_CIRCULAR_START	0x800E8	R/W	-	-	ISPU core register map - ISPU_CIRCULAR_START (0x800E8 - R/W)
ISPU_CIRCULAR_END	0x800EC	R/W	-	-	ISPU core register map - ISPU_CIRCULAR_END (0x800EC - R/W)
ISPU_CIRCULAR_CURR	0x800F0	R/W	-	-	ISPU core register map - ISPU_CIRCULAR_CURR (0x800F0 - R/W)
ISPU_CIRCULAR_STRIDE	0x800F4	R/W	-	-	ISPU core register map - ISPU_CIRCULAR_STRIDE (0x800F4 - R/W)
ISPU_SQRT_I	0x800F8	R/W	-	-	ISPU core register map - ISPU_SQRT_I (0x800F8 - R/W)
ISPU_SQRT_O	0x800FC	R	-	-	ISPU core register map - ISPU_SQRT_O (0x800FC - R)
ISPU registers					
ISPU_GLB_CALL_EN	0x80800	R/W	-	-	ISPU core register map - ISPU_GLB_CALL_EN (0x80800 - R/W)
ISPU_CONFIG	0x80802	R/W	0x02	R/W	ISPU register map (PAGE_SEL 1) - ISPU_CONFIG (02h - R/W)
ISPU_STATUS	0x80804	R/W	0x04		ISPU register map (PAGE_SEL 1) - ISPU_STATUS (04h - R)
ISPU_IF2S_FLAG_L	0x8080C	Clear/W	0x0C	R/Set only	ISPU register map (PAGE_SEL 1) - ISPU_IF2S_FLAG_H (0Dh R/W), ISPU_IF2S_FLAG_L (0Ch R/W)
ISPU_IF2S_FLAG_H	0x8080D	Clear/W	0x0D	R/Set only	ISPU register map (PAGE_SEL 1) - ISPU_IF2S_FLAG_H (0Dh R/W), ISPU_IF2S_FLAG_L (0Ch R/W)
ISPU_S2IF_FLAG_L	0x8080E	Set/W	0x0E	R/Clear only	ISPU register map (PAGE_SEL 1) - ISPU_S2IF_FLAG_H (0Fh) and ISPU_S2IF_FLAG_L (0Eh)
ISPU_S2IF_FLAG_H	0x8080F	Set/W	0x0F	R/Clear only	ISPU register map (PAGE_SEL 1) - ISPU_S2IF_FLAG_H (0Fh) and ISPU_S2IF_FLAG_L (0Eh)
ISPU_DOUT_00_L	0x80810	R/W	0x10	R	ISPU register map (PAGE_SEL 1) - ISPU_DOUT_00_H (11h R), ISPU_DOUT_00_L (10h R)
ISPU_DOUT_00_H	0x80811	R/W	0x11	R	ISPU register map (PAGE_SEL 1) - ISPU_DOUT_00_H (11h R), ISPU_DOUT_00_L (10h R)
ISPU_DOUT_01_L	0x80812	R/W	0x12	R	ISPU register map (PAGE_SEL 1) - ISPU_DOUT_01_H (13h R), ISPU_DOUT_01_L (12h R)
ISPU_DOUT_01_H	0x80813	R/W	0x13	R	ISPU register map (PAGE_SEL 1) - ISPU_DOUT_01_H (13h R), ISPU_DOUT_01_L (12h R)
ISPU_DOUT_02_L	0x80814	R/W	0x14	R	ISPU register map (PAGE_SEL 1) - ISPU_DOUT_02_H (15h R), ISPU_DOUT_02_L (14h R)
ISPU_DOUT_02_H	0x80815	R/W	0x15	R	ISPU register map (PAGE_SEL 1) - ISPU_DOUT_02_H (15h R), ISPU_DOUT_02_L (14h R)
ISPU_DOUT_03_L	0x80816	R/W	0x16	R	ISPU register map (PAGE_SEL 1) - ISPU_DOUT_03_H (17h R), ISPU_DOUT_03_L (16h R)
ISPU_DOUT_03_H	0x80817	R/W	0x17	R	ISPU register map (PAGE_SEL 1) - ISPU_DOUT_03_H (17h R), ISPU_DOUT_03_L (16h R)
ISPU_DOUT_04_L	0x80818	R/W	0x18	R	ISPU register map (PAGE_SEL 1) - ISPU_DOUT_04_H (19h R), ISPU_DOUT_04_L (18h R)

Register name	ISPU core address (hex)	ISPU core access type	Host IF address (hex)	Host IF access type	Register description
ISPU_DOUT_04_H	0x80819	R/W	0x19	R	ISPU register map (PAGE_SEL 1) - ISPU_DOUT_04_H (19h R), ISPU_DOUT_04_L (18h R)
ISPU_DOUT_05_L	0x8081A	R/W	0x1A	R	ISPU register map (PAGE_SEL 1) - ISPU_DOUT_05_H (1Bh R), ISPU_DOUT_05_L (1Ah R)
ISPU_DOUT_05_H	0x8081B	R/W	0x1B	R	ISPU register map (PAGE_SEL 1) - ISPU_DOUT_05_H (1Bh R), ISPU_DOUT_05_L (1Ah R)
ISPU_DOUT_06_L	0x8081C	R/W	0x1C	R	ISPU register map (PAGE_SEL 1) - ISPU_DOUT_06_H (1Dh R), ISPU_DOUT_06_L (1Ch R)
ISPU_DOUT_06_H	0x8081D	R/W	0x1D	R	ISPU register map (PAGE_SEL 1) - ISPU_DOUT_06_H (1Dh R), ISPU_DOUT_06_L (1Ch R)
ISPU_DOUT_07_L	0x8081E	R/W	0x1E	R	ISPU register map (PAGE_SEL 1) - ISPU_DOUT_07_H (1Fh R), ISPU_DOUT_07_L (1Eh R)
ISPU_DOUT_07_H	0x8081F	R/W	0x1F	R	ISPU register map (PAGE_SEL 1) - ISPU_DOUT_07_H (1Fh R), ISPU_DOUT_07_L (1Eh R)
ISPU_DOUT_08_L	0x80820	R/W	0x20	R	ISPU register mapping (PAGE_SEL 1) - ISPU_DOUT_08_H (21h R), ISPU_DOUT_08_L (20h R)
ISPU_DOUT_08_H	0x80821	R/W	0x21	R	ISPU register mapping (PAGE_SEL 1) - ISPU_DOUT_08_H (21h R), ISPU_DOUT_08_L (20h R)
ISPU_DOUT_09_L	0x80822	R/W	0x22	R	ISPU register mapping (PAGE_SEL 1) - ISPU_DOUT_09_H (23h R), ISPU_DOUT_09_L (22h R)
ISPU_DOUT_09_H	0x80823	R/W	0x23	R	ISPU register mapping (PAGE_SEL 1) - ISPU_DOUT_09_H (23h R), ISPU_DOUT_09_L (22h R)
ISPU_DOUT_10_L	0x80824	R/W	0x24	R	ISPU register mapping (PAGE_SEL 1) - ISPU_DOUT_10_H (25h R), ISPU_DOUT_10_L (24h R)
ISPU_DOUT_10_H	0x80825	R/W	0x25	R	ISPU register mapping (PAGE_SEL 1) - ISPU_DOUT_10_H (25h R), ISPU_DOUT_10_L (24h R)
ISPU_DOUT_11_L	0x80826	R/W	0x26	R	ISPU register mapping (PAGE_SEL 1) - ISPU_DOUT_11_H (27h R), ISPU_DOUT_11_L (26h R)
ISPU_DOUT_11_H	0x80827	R/W	0x27	R	ISPU register mapping (PAGE_SEL 1) - ISPU_DOUT_11_H (27h R), ISPU_DOUT_11_L (26h R)
ISPU_DOUT_12_L	0x80828	R/W	0x28	R	ISPU register mapping (PAGE_SEL 1) - ISPU_DOUT_12_H (29h R), ISPU_DOUT_12_L (28h R)
ISPU_DOUT_12_H	0x80829	R/W	0x29	R	ISPU register mapping (PAGE_SEL 1) - ISPU_DOUT_12_H (29h R), ISPU_DOUT_12_L (28h R)
ISPU_DOUT_13_L	0x8082A	R/W	0x2A	R	ISPU register mapping (PAGE_SEL 1) - ISPU_DOUT_13_H (2Bh R), ISPU_DOUT_13_L (2Ah R)
ISPU_DOUT_13_H	0x8082B	R/W	0x2B	R	ISPU register mapping (PAGE_SEL 1) - ISPU_DOUT_13_H (2Bh R), ISPU_DOUT_13_L (2Ah R)
ISPU_DOUT_14_L	0x8082C	R/W	0x2C	R	ISPU register mapping (PAGE_SEL 1) - ISPU_DOUT_14_H (2Dh R), ISPU_DOUT_14_L (2Ch R)
ISPU_DOUT_14_H	0x8082D	R/W	0x2D	R	ISPU register mapping (PAGE_SEL 1) - ISPU_DOUT_14_H (2Dh R), ISPU_DOUT_14_L (2Ch R)
ISPU_DOUT_15_L	0x8082E	R/W	0x2E	R	ISPU register mapping (PAGE_SEL 1) - ISPU_DOUT_15_H (2Fh R), ISPU_DOUT_15_L (2Eh R)
ISPU_DOUT_15_H	0x8082F	R/W	0x2F	R	ISPU register mapping (PAGE_SEL 1) - ISPU_DOUT_15_H (2Fh R), ISPU_DOUT_15_L (2Eh R)
ISPU_INT1_CTRL1	0x80850	R	0x50	R/W	ISPU register mapping (PAGE_SEL 1) - ISPU_INT1_CTRL2 (51h R/W), ISPU_INT1_CTRL1 (50h R/W)

Register name	ISPU core address (hex)	ISPU core access type	Host IF address (hex)	Host IF access type	Register description
ISPU_INT1_CTRL2	0x80851	R	0x51	R/W	ISPU register mapping (PAGE_SEL 1) - ISPU_INT1_CTRL2 (51h R/W), ISPU_INT1_CTRL1 (50h R/W)
ISPU_INT2_CTRL1	0x80854	R	0x54	R/W	ISPU register mapping (PAGE_SEL 1) - ISPU_INT2_CTRL2 (55h R/W), ISPU_INT2_CTRL1 (54h R/W)
ISPU_INT2_CTRL2	0x80855	R	0x55	R/W	ISPU register mapping (PAGE_SEL 1) - ISPU_INT2_CTRL2 (55h R/W), ISPU_INT2_CTRL1 (54h R/W)
ISPU_INT_STATUS1	0x80858	R/W	0x58	R	ISPU register mapping (PAGE_SEL 1) - ISPU_INT_STATUS2 (59h R), ISPU_INT_STATUS1 (58h R)
ISPU_INT_STATUS2	0x80859	R/W	0x59	R	ISPU register mapping (PAGE_SEL 1) - ISPU_INT_STATUS2 (59h R), ISPU_INT_STATUS1 (58h R)
ISPU_INT_PIN	0x8085C	R/W	0x5C	R	ISPU register mapping (PAGE_SEL 1) - ISPU_INT_PIN (5Ch R)
ISPU_ALGO1	0x80870	R/clear only	0x70	R/W	ISPU register mapping (PAGE_SEL 1) - ISPU_ALGO2 (71h R/W) and ISPU_ALGO1 (70h R/W)
ISPU_ALGO2	0x80871	R/clear only	0x71	R/W	ISPU register mapping (PAGE_SEL 1) - ISPU_ALGO2 (71h R/W) and ISPU_ALGO1 (70h R/W)
ISPU_ARAW_X_L	0x80880	R	-	-	ISPU core register map - ISPU_ARAW_X_HH (0x80883 - R), ISPU_ARAW_X_H (0x80882 - R), ISPU_ARAW_X_M (0x80881 - R), and ISPU_ARAW_X_L (0x80880 - R)
ISPU_ARAW_X_M	0x80881	R	-	-	ISPU core register map - ISPU_ARAW_X_HH (0x80883 - R), ISPU_ARAW_X_H (0x80882 - R), ISPU_ARAW_X_M (0x80881 - R), and ISPU_ARAW_X_L (0x80880 - R)
ISPU_ARAW_X_H	0x80882	R	-	-	ISPU core register map - ISPU_ARAW_X_HH (0x80883 - R), ISPU_ARAW_X_H (0x80882 - R), ISPU_ARAW_X_M (0x80881 - R), and ISPU_ARAW_X_L (0x80880 - R)
ISPU_ARAW_X_HH	0x80883	R	-	-	ISPU core register map - ISPU_ARAW_X_HH (0x80883 - R), ISPU_ARAW_X_H (0x80882 - R), ISPU_ARAW_X_M (0x80881 - R), and ISPU_ARAW_X_L (0x80880 - R)
ISPU_ARAW_Y_L	0x80884	R	-	-	ISPU core register map - ISPU_ARAW_Y_HH (0x80887 - R), ISPU_ARAW_Y_H (0x80886 - R), ISPU_ARAW_Y_M (0x80885 - R), and ISPU_ARAW_Y_L (0x80884 - R)
ISPU_ARAW_Y_M	0x80885	R	-	-	ISPU core register map - ISPU_ARAW_Y_HH (0x80887 - R), ISPU_ARAW_Y_H (0x80886 - R), ISPU_ARAW_Y_M (0x80885 - R), and ISPU_ARAW_Y_L (0x80884 - R)
ISPU_ARAW_Y_H	0x80886	R	-	-	ISPU core register map - ISPU_ARAW_Y_HH (0x80887 - R), ISPU_ARAW_Y_H (0x80886 - R), ISPU_ARAW_Y_M (0x80885 - R), and ISPU_ARAW_Y_L (0x80884 - R)
ISPU_ARAW_Y_HH	0x80887	R	-	-	ISPU core register map - ISPU_ARAW_Y_HH (0x80887 - R), ISPU_ARAW_Y_H (0x80886 - R), ISPU_ARAW_Y_M (0x80885 - R), and ISPU_ARAW_Y_L (0x80884 - R)
ISPU_ARAW_Z_L	0x80888	R	-	-	ISPU core register map - ISPU_ARAW_Z_HH (0x8088B - R), ISPU_ARAW_Z_H (0x8088A - R), ISPU_ARAW_Z_M (0x80889 - R), and ISPU_ARAW_Z_L (0x80888 - R)
ISPU_ARAW_Z_M	0x80889	R	-	-	ISPU core register map - ISPU_ARAW_Z_HH (0x8088B - R), ISPU_ARAW_Z_H (0x8088A - R), ISPU_ARAW_Z_M (0x80889 - R), and ISPU_ARAW_Z_L (0x80888 - R)
ISPU_ARAW_Z_H	0x8088A	R	-	-	ISPU core register map - ISPU_ARAW_Z_HH (0x8088B - R), ISPU_ARAW_Z_H (0x8088A - R), ISPU_ARAW_Z_M (0x80889 - R), and ISPU_ARAW_Z_L (0x80888 - R)

Register name	ISPU core address (hex)	ISPU core access type	Host IF address (hex)	Host IF access type	Register description
ISPU_ARAW_Z_HH	0x8088B	R	-	-	ISPU core register map - ISPU_ARAW_Z_HH (0x8088B - R), ISPU_ARAW_Z_H (0x8088A - R), ISPU_ARAW_Z_M (0x80889 - R), and ISPU_ARAW_Z_L (0x80888 - R)
ISPU_TEMP_L	0x808A4	R	-	-	ISPU core register map - ISPU_TEMP_H (0x808A5 - R) and ISPU_TEMP_L (0x808A4 - R)
ISPU_TEMP_H	0x808A5	R	-	-	ISPU core register map - ISPU_TEMP_H (0x808A5 - R) and ISPU_TEMP_L (0x808A4 - R)
ISPU_CALL_EN1	0x808B8	R/W	-	-	ISPU core register map - ISPU_CALL_EN4 (0x808BB - R/W), ISPU_CALL_EN3 (0x808BA - R/W), ISPU_CALL_EN2 (0x808B9 - R/W), and ISPU_CALL_EN1 (0x808B8 - R/W)
ISPU_CALL_EN2	0x808B9	R/W	-	-	ISPU core register map - ISPU_CALL_EN4 (0x808BB - R/W), ISPU_CALL_EN3 (0x808BA - R/W), ISPU_CALL_EN2 (0x808B9 - R/W), and ISPU_CALL_EN1 (0x808B8 - R/W)
ISPU_CALL_EN3	0x808BA	R/W	-	-	ISPU core register map - ISPU_CALL_EN4 (0x808BB - R/W), ISPU_CALL_EN3 (0x808BA - R/W), ISPU_CALL_EN2 (0x808B9 - R/W), and ISPU_CALL_EN1 (0x808B8 - R/W)
ISPU_CALL_EN4	0x808BB	R/W	-	-	ISPU core register map - ISPU_CALL_EN4 (0x808BB - R/W), ISPU_CALL_EN3 (0x808BA - R/W), ISPU_CALL_EN2 (0x808B9 - R/W), and ISPU_CALL_EN1 (0x808B8 - R/W)
External registers					
EXT_CTRL	0x80900	R/W	-	-	ISPU core register map - EXT_CTRL (0x80900 - R/W)
CTRL1	0x80904	R	10	R/W	IF register map (PAGE_SEL 0) - CTRL1 (10h - R/W)
CTRL2	0x80905	R/W	11	R/W	IF register map (PAGE_SEL 0) - CTRL2 (11h - R/W)
CTRL3	0x80906	R	12	R/W	IF register map (PAGE_SEL 0) - CTRL3 (12h - R/W)
CTRL4	0x80907	R/W[0]	13	R/W	IF register map (PAGE_SEL 0) - CTRL4 (13h - R/W)
FIFO_CTRL1	0x80908	R	05	R/W	IF register map (PAGE_SEL 0) - FIFO_CTRL1 (05h - R/W)
FIFO_CTRL2	0x80909	R	06	R/W	IF register map (PAGE_SEL 0) - FIFO_CTRL2 (06h - R/W)
FIFO_CTRL3	0x8090A	R	07	R/W	IF register map (PAGE_SEL 0) - FIFO_CTRL3 (07h - R/W)
IF_CFG	0x8090F	R/W[2]	03	R/W	IF register map (PAGE_SEL 0) - IF_CFG (03h - R/W)
FIFO_CTRL1_ISPU	0x80910	R/W	-	-	ISPU core register map - FIFO_CTRL1_ISPU (0x80910 – R/W)
FIFO_CTRL2_ISPU	0x80911	R/W	-	-	ISPU core register map - FIFO_CTRL2_ISPU (0x80911 – R/W)
FIFO_CTRL3_ISPU	0x80912	R/W	-	-	ISPU core register map - FIFO_CTRL3_ISPU (0x80912 – R/W)
SLEEPCNT_CFG	0x80917	R/W	38	R/W	IF register map (PAGE_SEL 0) - SLEEPCNT_CFG (38h - R/W)
ISPU_CTRL1	0x80918	R/W[7:4]	18	R/W	IF register map (PAGE_SEL 0) - ISPU_CTRL1 (18h - R/W)
ISPU_CTRL2	0x80919	R	19	R/W	IF register map (PAGE_SEL 0) - ISPU_CTRL2 (19h - R/W)
TPF	0x80920	R	20	R	IF register map (PAGE_SEL 0) - TPF (20h – R)
FIFO_STATUS1	0x8091C	R	4C	R	IF register map (PAGE_SEL 0) - FIFO_STATUS_1 (4Ch - R)
FIFO_STATUS2	0x8091D	R	4D	R	IF register map (PAGE_SEL 0) - FIFO_STATUS_2 (4Dh - R)
WHO_AM_I	0x80921	R	0F	R	IF register map (PAGE_SEL 0) - WHO_AM_I (0Fh - R)

Register name	ISPU core address (hex)	ISPU core access type	Host IF address (hex)	Host IF access type	Register description
TIMESTAMP0	0x80930	R	30	R	IF register map (PAGE_SEL 0) - TIMESTAMP4 (34h - R), TIMESTAMP3 (33h - R), TIMESTAMP2 (32h - R), TIMESTAMP1 (31h - R), TIMESTAMP0 (30h - R)
TIMESTAMP1	0x80931	R	31	R	IF register map (PAGE_SEL 0) - TIMESTAMP4 (34h - R), TIMESTAMP3 (33h - R), TIMESTAMP2 (32h - R), TIMESTAMP1 (31h - R), TIMESTAMP0 (30h - R)
TIMESTAMP2	0x80932	R	32	R	IF register map (PAGE_SEL 0) - TIMESTAMP4 (34h - R), TIMESTAMP3 (33h - R), TIMESTAMP2 (32h - R), TIMESTAMP1 (31h - R), TIMESTAMP0 (30h - R)
TIMESTAMP3	0x80933	R	33	R	IF register map (PAGE_SEL 0) - TIMESTAMP4 (34h - R), TIMESTAMP3 (33h - R), TIMESTAMP2 (32h - R), TIMESTAMP1 (31h - R), TIMESTAMP0 (30h - R)
TIMESTAMP4	0x80934	R	34	R	IF register map (PAGE_SEL 0) - TIMESTAMP4 (34h - R), TIMESTAMP3 (33h - R), TIMESTAMP2 (32h - R), TIMESTAMP1 (31h - R), TIMESTAMP0 (30h - R)
SLEEPCNT_TH_L	0x8093A	R/W	3E	R/W	IF register map (PAGE_SEL 0) - SLEEPCNT_TH_H (3Fh R/W) and SLEEPCNT_TH_L (3Eh R/W)
SLEEPCNT_TH_H	0x8093B	R/W	3F	R/W	IF register map (PAGE_SEL 0) - SLEEPCNT_TH_H (3Fh R/W) and SLEEPCNT_TH_L (3Eh R/W)
INTERNAL_FREQ_FINE	0x80947	R	50	R	IF register map (PAGE_SEL 0) - INTERNAL_FREQ_FINE (50h - R)
SLEEPCNT_TIME_L	0x80948	R	62	R	IF register map (PAGE_SEL 0) - SLEEPCNT_TIME_H (63h R) and SLEEPCNT_TIME_L (62h R)
SLEEPCNT_TIME_H	0x80949	R	63	R	IF egister map (PAGE_SEL 0) - SLEEPCNT_TIME_H (63h R) and SLEEPCNT_TIME_L (62h R)
ISPU_DUMMY_CFG_1	0x80954	R	52	R/W	IF register map (PAGE_SEL 0) - ISPU_DUMMY_CFG_1 (52h - R/W) through ISPU_DUMMY_CFG_16 (61h - R/W)
ISPU_DUMMY_CFG_2	0x80955	R	53	R/W	IF register map (PAGE_SEL 0) - ISPU_DUMMY_CFG_1 (52h - R/W) through ISPU_DUMMY_CFG_16 (61h - R/W)
ISPU_DUMMY_CFG_3	0x80956	R	54	R/W	IF register map (PAGE_SEL 0) - ISPU_DUMMY_CFG_1 (52h - R/W) through ISPU_DUMMY_CFG_16 (61h - R/W)
ISPU_DUMMY_CFG_4	0x80957	R	55	R/W	IF register map (PAGE_SEL 0) - ISPU_DUMMY_CFG_1 (52h - R/W) through ISPU_DUMMY_CFG_16 (61h - R/W)
ISPU_DUMMY_CFG_5	0x80958	R	56	R/W	IF register map (PAGE_SEL 0) - ISPU_DUMMY_CFG_1 (52h - R/W) through ISPU_DUMMY_CFG_16 (61h - R/W)
ISPU_DUMMY_CFG_6	0x80959	R	57	R/W	IF register map (PAGE_SEL 0) - ISPU_DUMMY_CFG_1 (52h - R/W) through ISPU_DUMMY_CFG_16 (61h - R/W)
ISPU_DUMMY_CFG_7	0x8095A	R	58	R/W	IF register map (PAGE_SEL 0) - ISPU_DUMMY_CFG_1 (52h - R/W) through ISPU_DUMMY_CFG_16 (61h - R/W)
ISPU_DUMMY_CFG_8	0x8095B	R	59	R/W	IF register mapping (PAGE_SEL 0) - ISPU_DUMMY_CFG_1 (52h - R/W) through ISPU_DUMMY_CFG_16 (61h - R/W)
ISPU_DUMMY_CFG_9	0x8095C	R	5A	R/W	IF register map (PAGE_SEL 0) - ISPU_DUMMY_CFG_1 (52h - R/W) through ISPU_DUMMY_CFG_16 (61h - R/W)
ISPU_DUMMY_CFG_10	0x8095D	R	5B	R/W	IF register map (PAGE_SEL 0) - ISPU_DUMMY_CFG_1 (52h - R/W) through ISPU_DUMMY_CFG_16 (61h - R/W)
ISPU_DUMMY_CFG_11	0x8095E	R	5C	R/W	IF register map (PAGE_SEL 0) - ISPU_DUMMY_CFG_1 (52h - R/W) through ISPU_DUMMY_CFG_16 (61h - R/W)
ISPU_DUMMY_CFG_12	0x8095F	R	5D	R/W	IF register map (PAGE_SEL 0) - ISPU_DUMMY_CFG_1 (52h - R/W) through ISPU_DUMMY_CFG_16 (61h - R/W)

Register name	ISPU core address (hex)	ISPU core access type	Host IF address (hex)	Host IF access type	Register description
ISPU_DUMMY_CFG_13	0x80960	R	5E	R/W	IF register map (PAGE_SEL 0) - ISPU_DUMMY_CFG_1 (52h - R/W) through ISPU_DUMMY_CFG_16 (61h - R/W)
ISPU_DUMMY_CFG_14	0x80961	R	5F	R/W	IF register map (PAGE_SEL 0) - ISPU_DUMMY_CFG_1 (52h - R/W) through ISPU_DUMMY_CFG_16 (61h - R/W)
ISPU_DUMMY_CFG_15	0x80962	R	60	R/W	IF register map (PAGE_SEL 0) - ISPU_DUMMY_CFG_1 (52h - R/W) through ISPU_DUMMY_CFG_16 (61h - R/W)
ISPU_DUMMY_CFG_16	0x80963	R	61	R/W	IF register map (PAGE_SEL 0) - ISPU_DUMMY_CFG_1 (52h - R/W) through ISPU_DUMMY_CFG_16 (61h - R/W)
ISPU_LOPRIO_INT	0x80978	R/clear only	-	-	ISPU core register map - ISPU_LOPRIO_INT (0x80978 – R/W)
ISPU2FIFO_FLAG	0x8097C	R	-	-	ISPU core register map - ISPU2FIFO_FLAG (0x8097C – R/W)
FIFO2ISPU_TAG	0x8097D	R	-	-	ISPU core register map - FIFO2ISPU_TAG (0x8097D – R/W)
FIFO2ISPU_CTRL	0x8097E	R/W	-	-	ISPU core register map - FIFO2ISPU_CTRL (0x8097E – R/W)
ISPU2FIFO_INT	0x8097F	R/W	-	-	ISPU core register map - ISPU2FIFO_INT (0x8097F – R/W)

12 ISPU core registers description

12.1 ISPU_LOOP_START (0x80080 - R/W)

Table 249. ISPU_LOOP_START register

bit31	bit30	bit29	bit28	bit27	bit26	bit25	bit24
0	0	0	0	0	0	0	0
bit23	**bit22**	**bit21**	**bit20**	**bit19**	**bit18**	**bit17**	**bit16**
0	0	0	0	A19	A18	A17	A16
bit15	**bit14**	**bit13**	**bit12**	**bit11**	**bit10**	**bit9**	**bit8**
A15	A14	A13	A12	A11	A10	A9	A8
bit7	**bit6**	**bit5**	**bit4**	**bit3**	**bit2**	**bit1**	**bit0**
A7	A6	A5	A4	A3	A2	A1	A0(1)

1. *This bit is not writable and its value is fixed at 0.*

Table 250. ISPU_LOOP_START register description

A[19:0]	0x000000	R/W	Loop start address. The least significant bit is always 0 since the address must be aligned to 2 bytes.

12.2 ISPU_LOOP_END (0x80084 - R/W)

Table 251. ISPU_LOOP_END register

bit31	bit30	bit29	bit28	bit27	bit26	bit25	bit24
0	0	0	0	0	0	0	0
bit23	**bit22**	**bit21**	**bit20**	**bit19**	**bit18**	**bit17**	**bit16**
0	0	0	0	A19	A18	A17	A16
bit15	**bit14**	**bit13**	**bit12**	**bit11**	**bit10**	**bit9**	**bit8**
A15	A14	A13	A12	A11	A10	A9	A8
bit7	**bit6**	**bit5**	**bit4**	**bit3**	**bit2**	**bit1**	**bit0**
A7	A6	A5	A4	A3	A2	A1	A0(1)

1. *This bit is not writable and its value is fixed at 0.*

Table 252. ISPU_LOOP_END register description

A[19:0]	0x000000	R/W	Loop end address. The least significant bit is always 0 since the address must be aligned to 2 bytes.

12.3 ISPU_LOOP_CNT (0x80088 - R/W)

Table 253. ISPU_LOOP_CNT register

bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
0	0	0	0	0	LOOP_EN	LOOP_CNT9	LOOP_CNT8
bit7	**bit6**	**bit5**	**bit4**	**bit3**	**bit2**	**bit1**	**bit0**
LOOP_CNT7	LOOP_CNT6	LOOP_CNT5	LOOP_CNT4	LOOP_CNT3	LOOP_CNT2	LOOP_CNT1	LOOP_CNT0

Table 254. ISPU_LOOP_CNT register description

LOOP_EN	0x0	R/W	Set to 1 to enable the loop.
LOOP_CNT[9:0]	0x000	R/W	Number of iterations minus 1.

12.4 ISPU_FFT_MATCH_TW (0x8008C - R/W)

Table 255. ISPU_FFT_MATCH_TW register

bit31	bit30	bit29	bit28	bit27	bit26	bit25	bit24
D31	D30	D29	D28	D27	D26	D25	D24
bit23	**bit22**	**bit21**	**bit20**	**bit19**	**bit18**	**bit17**	**bit16**
D23	D22	D21	D20	D19	D18	D17	D16
bit15	**bit14**	**bit13**	**bit12**	**bit11**	**bit10**	**bit9**	**bit8**
D15	D14	D13	D12	D11	D10	D9	D8
bit7	**bit6**	**bit5**	**bit4**	**bit3**	**bit2**	**bit1**	**bit0**
D7	D6	D5	D4	D3	D2	D1	D0

Table 256. ISPU_FFT_MATCH_TW register description

D[31:0]	-	R/W	Currently addressed FFT twiddle factors.

12.5 ISPU_FFT_START_TW (0x80090 - R/W)

Table 257. ISPU_FFT_START_TW register

bit31	bit30	bit29	bit28	bit27	bit26	bit25	bit24
0	0	0	0	0	0	0	0
bit23	**bit22**	**bit21**	**bit20**	**bit19**	**bit18**	**bit17**	**bit16**
0	0	0	0	A19	A18	A17	A16
bit15	**bit14**	**bit13**	**bit12**	**bit11**	**bit10**	**bit9**	**bit8**
A15	A14	A13	A12	A11	A10	A9	A8
bit7	**bit6**	**bit5**	**bit4**	**bit3**	**bit2**	**bit1**	**bit0**
A7(1)	A6(1)	A5(1)	A4(1)	A3(1)	A2(1)	A1(1)	A0(1)

1. *This bit is not writable and its value is fixed at 0.*

Table 258. ISPU_FFT_START_TW register description

A[19:0]	0x000000	R/W	Start address for FFT twiddle factors. The 8 least significant bits are always 0 since the address must be aligned to at least 256 bytes (up to 4096 bytes depending on the number of points of the FFT).

12.6 ISPU_FFT_END_TW (0x80094 - R/W)

Table 259. ISPU_FFT_END_TW register

bit31	bit30	bit29	bit28	bit27	bit26	bit25	bit24
0	0	0	0	0	0	0	0
bit23	**bit22**	**bit21**	**bit20**	**bit19**	**bit18**	**bit17**	**bit16**
0	0	0	0	0	0	0	0
bit15	**bit14**	**bit13**	**bit12**	**bit11**	**bit10**	**bit9**	**bit8**
0	0	0	0	A11	A10	A9	A8
bit7	**bit6**	**bit5**	**bit4**	**bit3**	**bit2**	**bit1**	**bit0**
A7	A6	A5	A4	A3	A2	A1(1)	A0(1)

1. *This bit is not writable and its value is fixed at 0.*

Table 260. ISPU_FFT_END_TW register description

A[11:0]	0x000	R/W	End address for FFT twiddle factors. The 2 least significant bits are always 0 since the address must be aligned to 4 bytes. The most significant bits (bits 19 to 12) of the address are taken from ISPU_FFT_START_TW.

12.7 ISPU_FFT_CURR_TW1 (0x80098 - R/W)

Table 261. ISPU_FFT_CURR_TW1 register

bit31	bit30	bit29	bit28	bit27	bit26	bit25	bit24
0	0	0	0	0	0	0	0
bit23	**bit22**	**bit21**	**bit20**	**bit19**	**bit18**	**bit17**	**bit16**
0	0	0	0	0	0	0	0
bit15	**bit14**	**bit13**	**bit12**	**bit11**	**bit10**	**bit9**	**bit8**
0	0	0	0	A11	A10	A9	A8
bit7	**bit6**	**bit5**	**bit4**	**bit3**	**bit2**	**bit1**	**bit0**
A7	A6	A5	A4	A3	A2	A1(1)	A0(1)

1. *This bit is not writable and its value is fixed at 0.*

Table 262. ISPU_FFT_CURR_TW1 register description

A[11:0]	0x000	R/W	Current address 1 for FFT twiddle factors. The 2 least significant bits are always 0 since the address must be aligned to 4 bytes. The most significant bits (bits 19 to 12) of the address are taken from ISPU_FFT_START_TW.

12.8 ISPU_FFT_CURR_TW2 (0x8009C - R/W)

Table 263. ISPU_FFT_CURR_TW2 register

bit31	bit30	bit29	bit28	bit27	bit26	bit25	bit24
0	0	0	0	0	0	0	0
bit23	**bit22**	**bit21**	**bit20**	**bit19**	**bit18**	**bit17**	**bit16**
0	0	0	0	0	0	0	0
bit15	**bit14**	**bit13**	**bit12**	**bit11**	**bit10**	**bit9**	**bit8**
0	0	0	0	A11	A10	A9	A8
bit7	**bit6**	**bit5**	**bit4**	**bit3**	**bit2**	**bit1**	**bit0**
A7	A6	A5	A4	A3	A2	A1(1)	A0(1)

1. *This bit is not writable and its value is fixed at 0.*

Table 264. ISPU_FFT_CURR_TW2 register description

A[11:0]	0x000	R/W	Current address 2 for FFT twiddle factors. The 2 least significant bits are always 0 since the address must be aligned to 4 bytes. The most significant bits (bits 19 to 12) of the address are taken from ISPU_FFT_START_TW.

12.9 ISPU_FFT_STRIDE_TW (0x800A0 - R/W)

Table 265. ISPU_FFT_STRIDE_TW register

bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
0	0	0	0	0	0	FFT_SPLIT	FFT_STRIDE_TW8
bit7	**bit6**	**bit5**	**bit4**	**bit3**	**bit2**	**bit1**	**bit0**
FFT_STRIDE_TW7	FFT_STRIDE_TW6	FFT_STRIDE_TW5	FFT_STRIDE_TW4	FFT_STRIDE_TW3	FFT_STRIDE_TW2	FFT_STRIDE_TW1	FFT_STRIDE_TW0

Table 266. ISPU_FFT_STRIDE_TW register description

FFT_SPLIT	0x0	R/W	Set to 1 for split FFT.
FFT_STRIDE_TW[8:0]	0x000	R/W	Stride for CFFT (max 128).

12.10 ISPU_FFT_STATE_TW (0x800A4 - R/W)

Table 267. ISPU_FFT_STATE_TW register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
0	0	0	0	0	0	FFT_STATE_TW1	FFT_STATE_TW0

Table 268. ISPU_FFT_STATE_TW register description

FFT_STATE_TW[1:0]	0x0	R/W	FFT twiddle state.

12.11 ISPU_FFT_MATCH_D (0x800C0 - R/W)

Table 269. ISPU_FFT_MATCH_D register

bit31	bit30	bit29	bit28	bit27	bit26	bit25	bit24
D31	D30	D29	D28	D27	D26	D25	D24
bit23	**bit22**	**bit21**	**bit20**	**bit19**	**bit18**	**bit17**	**bit16**
D23	D22	D21	D20	D19	D18	D17	D16
bit15	**bit14**	**bit13**	**bit12**	**bit11**	**bit10**	**bit9**	**bit8**
D15	D14	D13	D12	D11	D10	D9	D8
bit7	**bit6**	**bit5**	**bit4**	**bit3**	**bit2**	**bit1**	**bit0**
D7	D6	D5	D4	D3	D2	D1	D0

Table 270. ISPU_FFT_MATCH_D register description

D[31:0]	-	R/W	Currently addressed FFT internal data.

12.12 ISPU_FFT_CURR_D1 (0x800C4 - R/W)

Table 271. ISPU_FFT_CURR_D1 register

bit31	bit30	bit29	bit28	bit27	bit26	bit25	bit24
0	0	0	0	0	0	0	0
bit23	**bit22**	**bit21**	**bit20**	**bit19**	**bit18**	**bit17**	**bit16**
0	0	0	0	A19	A18	A17	A16
bit15	**bit14**	**bit13**	**bit12**	**bit11**	**bit10**	**bit9**	**bit8**
A15	A14	A13	A12	A11	A10	A9	A8
bit7	**bit6**	**bit5**	**bit4**	**bit3**	**bit2**	**bit1**	**bit0**
A7	A6	A5	A4	A3	A2	A1(1)	A0(1)

1. *This bit is not writable and its value is fixed at 0.*

Table 272. ISPU_FFT_CURR_D1 register description

A[19:0]	0x000000	R/W	Current address 1 for FFT internal data. The 2 least significant bits are always 0 since the address must be aligned to 4 bytes.

12.13 ISPU_FFT_CURR_D2 (0x800C8 - R/W)

Table 273. ISPU_FFT_CURR_D2 register

bit31	bit30	bit29	bit28	bit27	bit26	bit25	bit24
0	0	0	0	0	0	0	0
bit23	**bit22**	**bit21**	**bit20**	**bit19**	**bit18**	**bit17**	**bit16**
0	0	0	0	0	0	0	0
bit15	**bit14**	**bit13**	**bit12**	**bit11**	**bit10**	**bit9**	**bit8**
0	0	0	A12	A11	A10	A9	A8
bit7	**bit6**	**bit5**	**bit4**	**bit3**	**bit2**	**bit1**	**bit0**
A7	A6	A5	A4	A3	A2	A1(1)	A0(1)

1. *This bit is not writable and its value is fixed at 0.*

Table 274. ISPU_FFT_CURR_D2 register description

A[12:0]	0x0000	R/W	Current address 2 for FFT internal data. The 2 least significant bits are always 0 since the address must be aligned to 4 bytes. The most significant bits (bits 19 to 13) of the address are taken from ISPU_FFT_CURR_D1 (0x800C4 - R/W).

12.14 ISPU_FFT_TWIDDLE (0x800CC - R/W)

Table 275. ISPU_FFT_TWIDDLE register

bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
0	0	0	0	0	0	FFT_TWIDDLE9	FFT_TWIDDLE8
bit7	**bit6**	**bit5**	**bit4**	**bit3**	**bit2**	**bit1**	**bit0**
FFT_TWIDDLE7	FFT_TWIDDLE6	FFT_TWIDDLE5	FFT_TWIDDLE4	FFT_TWIDDLE2	FFT_TWIDDLE1	FFT_TWIDDLE1	FFT_TWIDDLE0

Table 276. ISPU_FFT_TWIDDLE register description

FFT_TWIDDLE[9:0]	0x000	R/W	Number of twiddle factors (max 512).

12.15 ISPU_FFT_STATE_D (0x800D0 - R/W)

Table 277. ISPU_FFT_STATE_D register

bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
0	0	0	FFT_CNT_D9	FFT_CNT_D8	FFT_CNT_D7	FFT_CNT_D6	FFT_CNT_D5
bit7	**bit6**	**bit5**	**bit4**	**bit3**	**bit2**	**bit1**	**bit0**
FFT_CNT_D4	FFT_CNT_D3	FFT_CNT_D2	FFT_CNT_D1	FFT_CNT_D0	FFT_STATE_D2	FFT_STATE_D1	FFT_STATE_D0

Table 278. ISPU_FFT_STATE_D register description

FFT_CNT_D[9:0]	0x000	R/W	FFT internal data counter.
FFT_STATE_D[0:2]	0x0	R/W	FFT internal data state.

12.16 ISPU_FFT_STATE_BRA (0x800D4 - R/W)

Table 279. ISPU_FFT_STATE_BRA register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
0	0	0	0	0	FFT_STATE_BRA2	FFT_STATE_BRA1	FFT_STATE_BRA0

Table 280. ISPU_FFT_STATE_BRA register description

FFT_STATE_BRA[2:0]	0x0	R/W	FFT bit reverse addressing state.

12.17 ISPU_FFT_MATCH_BRA (0x800D8 - R/W)

Table 281. ISPU_FFT_MATCH_BRA register

bit31	bit30	bit29	bit28	bit27	bit26	bit25	bit24
D31	D30	D29	D28	D27	D26	D25	D24
bit23	**bit22**	**bit21**	**bit20**	**bit19**	**bit18**	**bit17**	**bit16**
D23	D22	D21	D20	D19	D18	D17	D16
bit15	**bit14**	**bit13**	**bit12**	**bit11**	**bit10**	**bit9**	**bit8**
D15	D14	D13	D12	D11	D10	D9	D8
bit7	**bit6**	**bit5**	**bit4**	**bit3**	**bit2**	**bit1**	**bit0**
D7	D6	D5	D4	D3	D2	D1	D0

Table 282. ISPU_FFT_MATCH_BRA register description

D[31:0]	-	R/W	Currently addressed FFT data for bit reverse addressing.

12.18 ISPU_FFT_CURR_BRA (0x800DC - R/W)

Table 283. ISPU_FFT_CURR_BRA register

bit31	bit30	bit29	bit28	bit27	bit26	bit25	bit24
0	0	0	0	0	0	0	0
bit23	**bit22**	**bit21**	**bit20**	**bit19**	**bit18**	**bit17**	**bit16**
0	0	0	0	A19	A18	A17	A16
bit15	**bit14**	**bit13**	**bit12**	**bit11**	**bit10**	**bit9**	**bit8**
A15	A14	A13	A12	A11	A10	A9	A8
bit7	**bit6**	**bit5**	**bit4**	**bit3**	**bit2**	**bit1**	**bit0**
A7	A6	A5	A4	A3	A2	A1(1)	A0(1)

1. *This bit is not writable and its value is fixed at 0.*

Table 284. ISPU_FFT_CURR_BRA register description

A[19:0]	0x000000	R/W	Current address for FFT bit reverse addressing. The 2 least significant bits are always 0 since the address must be aligned to 4 bytes.

12.19 ISPU_FFT_N_ELEM_BRA (0x800E0 - R/W)

Table 285. ISPU_FFT_N_ELEM_BRA register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
0	0	0	0	0	FFT_N_BRA2	FFT_N_BRA1	FFT_N_BRA0

Table 286. ISPU_FFT_N_ELEM_BRA register description

FFT_N_BRA[2:0]	0x0	R/W	Number of points for FFT (000 =16 points, 001 = 32 points, 010 = 64 points, 011 = 128 points, 100 = 256 points, 101 = 512 points, 110 = 1024 points).

12.20 ISPU_CIRCULAR_MATCH (0x800E4 - R/W)

Table 287. ISPU_CIRCULAR_MATCH register

bit31	bit30	bit29	bit28	bit27	bit26	bit25	bit24
D31	D30	D29	D28	D27	D26	D25	D24
bit23	**bit22**	**bit21**	**bit20**	**bit19**	**bit18**	**bit17**	**bit16**
D23	D22	D21	D20	D19	D18	D17	D16
bit15	**bit14**	**bit13**	**bit12**	**bit11**	**bit10**	**bit9**	**bit8**
D15	D14	D13	D12	D11	D10	D9	D8
bit7	**bit6**	**bit5**	**bit4**	**bit3**	**bit2**	**bit1**	**bit0**
D7	D6	D5	D4	D3	D2	D1	D0

Table 288. ISPU_CIRCULAR_MATCH register description

D[31:0]	-	R/W	Currently addressed element of the circular buffer.

12.21 ISPU_CIRCULAR_START (0x800E8 - R/W)

Table 289. ISPU_CIRCULAR_START register

bit31	bit30	bit29	bit28	bit27	bit26	bit25	bit24
0	0	0	0	0	0	0	0
bit23	**bit22**	**bit21**	**bit20**	**bit19**	**bit18**	**bit17**	**bit16**
0	0	0	0	A19	A18	A17	A16
bit15	**bit14**	**bit13**	**bit12**	**bit11**	**bit10**	**bit9**	**bit8**
A15	A14	A13	A12	A11	A10	A9	A8
bit7	**bit6**	**bit5**	**bit4**	**bit3**	**bit2**	**bit1**	**bit0**
A7(1)	A6(1)	A5(1)	A4(1)	A3(1)	A2(1)	A1(1)	A0(1)

1. *This bit is not writable and its value is fixed at 0.*

Table 290. ISPU_CIRCULAR_START register description

A[19:0]	0x000000	R/W	Start address of the circular buffer. The 8 least significant bits are always 0 since the address must be aligned to at least 256 bytes (4096 bytes if wrapping from the end to the start of the buffer).

12.22 ISPU_CIRCULAR_END (0x800EC - R/W)

Table 291. ISPU_CIRCULAR_END register

bit31	bit30	bit29	bit28	bit27	bit26	bit25	bit24
0	0	0	0	0	0	0	0
bit23	**bit22**	**bit21**	**bit20**	**bit19**	**bit18**	**bit17**	**bit16**
0	0	0	0	0	0	0	0
bit15	**bit14**	**bit13**	**bit12**	**bit11**	**bit10**	**bit9**	**bit8**
0	0	0	0	A11	A10	A9	A8
bit7	**bit6**	**bit5**	**bit4**	**bit3**	**bit2**	**bit1**	**bit0**
A7	A6	A5	A4	A3	A2	A1(1)	A0(1)

1. *This bit is not writable and its value is fixed at 0.*

Table 292. ISPU_CIRCULAR_END register description

A[11:0]	0x000	R/W	End address of the circular buffer. The 2 least significant bits are always 0 since the address must be aligned to 4 bytes. The most significant bits (bits 19 to 12) of the address are taken from ISPU_CIRCULAR_START.

12.23 ISPU_CIRCULAR_CURR (0x800F0 - R/W)

Table 293. ISPU_CIRCULAR_CURR register

bit31	bit30	bit29	bit28	bit27	bit26	bit25	bit24
0	0	0	0	0	0	0	0
bit23	**bit22**	**bit21**	**bit20**	**bit19**	**bit18**	**bit17**	**bit16**
0	0	0	0	0	0	0	0
bit15	**bit14**	**bit13**	**bit12**	**bit11**	**bit10**	**bit9**	**bit8**
0	0	0	0	A11	A10	A9	A8
bit7	**bit6**	**bit5**	**bit4**	**bit3**	**bit2**	**bit1**	**bit0**
A7	A6	A5	A4	A3	A2	A1(1)	A0(1)

1. *This bit is not writable and its value is fixed at 0.*

Table 294. ISPU_CIRCULAR_CURR register description

A[11:0]	0x000	R/W	Current address for the circular buffer. The 2 least significant bits are always 0 since the address must be aligned to 4 bytes. The most significant bits (bits 19 to 12) of the address are taken from ISPU_CIRCULAR_START.

12.24 ISPU_CIRCULAR_STRIDE (0x800F4 - R/W)

Table 295. ISPU_CIRCULAR_STRIDE register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
0	0	0	0	0	CIRCULAR_STRIDE2	CIRCULAR_STRIDE1	CIRCULAR_STRIDE0

Table 296. ISPU_CIRCULAR_STRIDE register description

CIRCULAR_STRIDE[2:0]	0x0	R/W	Stride for the circular buffer.

12.25 ISPU_SQRT_I (0x800F8 - R/W)

Table 297. ISPU_SQRT_I register

bit31	bit30	bit29	bit28	bit27	bit26	bit25	bit24
D31	D30	D29	D28	D27	D26	D25	D24
bit23	**bit22**	**bit21**	**bit20**	**bit19**	**bit18**	**bit17**	**bit16**
D23	D22	D21	D20	D19	D18	D17	D16
bit15	**bit14**	**bit13**	**bit12**	**bit11**	**bit10**	**bit9**	**bit8**
D15	D14	D13	D12	D11	D10	D9	D8
bit7	**bit6**	**bit5**	**bit4**	**bit3**	**bit2**	**bit1**	**bit0**
D7	D6	D5	D4	D3	D2	D1	D0

Table 298. ISPU_SQRT_I register description

D[31:0]	0x00000000	R/W	Square root input value.

12.26 ISPU_SQRT_O (0x800FC - R)

Table 299. ISPU_SQRT_O register

bit31	bit30	bit29	bit28	bit27	bit26	bit25	bit24
D31	D30	D29	D28	D27	D26	D25	D24
bit23	**bit22**	**bit21**	**bit20**	**bit19**	**bit18**	**bit17**	**bit16**
D23	D22	D21	D20	D19	D18	D17	D16
bit15	**bit14**	**bit13**	**bit12**	**bit11**	**bit10**	**bit9**	**bit8**
D15	D14	D13	D12	D11	D10	D9	D8
bit7	**bit6**	**bit5**	**bit4**	**bit3**	**bit2**	**bit1**	**bit0**
D7	D6	D5	D4	D3	D2	D1	D0

Table 300. ISPU_SQRT_O register description

D[31:0]	0x00000000	R	Square root output value.

12.27 ISPU_GLB_CALL_EN (0x80800 - R/W)

Table 301. ISPU_GLB_CALL_EN register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
0	0	0	0	EN_DVAL_INT	EN_LOPRI_INT	EN_ORUN_INT	EN_ALGO_INT

Table 302. ISPU_GLB_CALL_EN register description

EN_DVAL_INT	0x0	R/W	If asserted by the ISPU, high-priority interrupt routine is enabled and executed when a high interrupt is received by the ISPU.
EN_LOPRI_INT	0x0	R/W	If asserted by the ISPU, low-priority interrupt routine is enabled and executed when a low interrupt is received by the ISPU.
EN_ORUN_INT	0x0	R/W	If asserted by the ISPU, the overrun interrupt routine is enabled and executed when an overrun is detected.
EN_ALGO_INT	0x0	R/W	If asserted by the ISPU, the algorithms scheduler will be executed (common to every algorithm).

12.28 ISPU_ARAW_X_HH (0x80883 - R), ISPU_ARAW_X_H (0x80882 - R), ISPU_ARAW_X_M (0x80881 - R), and ISPU_ARAW_X_L (0x80880 - R)

Accelerometer sensor data (X-axis)

Table 303. ISPU_ARAW_X_HH register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D31	D30	D29	D28	D27	D26	D25	D24

Table 304. ISPU_ARAW_X_H register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D23	D22	D21	D20	D19	D18	D17	D16

Table 305. ISPU_ARAW_X_M register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D15	D14	D13	D12	D11	D10	D9	D8

Table 306. ISPU_ARAW_X_L register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D7	D6	D5	D4	D3	D2	D1	D0

Table 307. ISPU_ARAW_X register description

D[31:0]	0x00000000	R	Accelerometer sensor data (X-axis) Two's complement, sign-extended 20-bit format

12.29 ISPU_ARAW_Y_HH (0x80887 - R), ISPU_ARAW_Y_H (0x80886 - R), ISPU_ARAW_Y_M (0x80885 - R), and ISPU_ARAW_Y_L (0x80884 - R)

Accelerometer sensor data (Y-axis)

Table 308. ISPU_ARAW_Y_HH register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D31	D30	D29	D28	D27	D26	D25	D24

Table 309. ISPU_ARAW_Y_H register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D23	D22	D21	D20	D19	D18	D17	D16

Table 310. ISPU_ARAW_Y_M register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D15	D14	D13	D12	D11	D10	D9	D8

Table 311. ISPU_ARAW_Y_L register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D7	D6	D5	D4	D3	D2	D1	D0

Table 312. ISPU_ARAW_Y register description

D[31:0]	0x00000000	R	Accelerometer sensor data (Y-axis). Two's complement, sign-extended 20-bit format.

12.30 ISPU_ARAW_Z_HH (0x8088B - R), ISPU_ARAW_Z_H (0x8088A - R), ISPU_ARAW_Z_M (0x80889 - R), and ISPU_ARAW_Z_L (0x80888 - R)

Accelerometer sensor data (Z-axis)

Table 313. ISPU_ARAW_Z_HH register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D31	D30	D29	D28	D27	D26	D25	D24

Table 314. ISPU_ARAW_Z_H register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D23	D22	D21	D20	D19	D18	D17	D16

Table 315. ISPU_ARAW_Z_M register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D15	D14	D13	D12	D11	D10	D9	D8

Table 316. ISPU_ARAW_Z_L register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D7	D6	D5	D4	D3	D2	D1	D0

Table 317. ISPU_ARAW_Z register description

D[31:0]	0x00000000	R	Accelerometer sensor data (Z-axis) Two's complement, sign-extended 20-bit format

12.31 ISPU_TEMP_H (0x808A5 - R) and ISPU_TEMP_L (0x808A4 - R)

Temperature sensor data

Table 318. ISPU_TEMP_H register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D15	D14	D13	D12	D11	D10	D9	D8

Table 319. ISPU_TEMP_L register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
D7	D6	D5	D4	D3	D2	D1	D0

Table 320. ISPU_TEMP register description

D[15:0]	0x0000	R	Temperature sensor data Output expressed in two's complement

12.32 ISPU_CALL_EN4 (0x808BB - R/W), ISPU_CALL_EN3 (0x808BA - R/W), ISPU_CALL_EN2 (0x808B9 - R/W), and ISPU_CALL_EN1 (0x808B8 - R/W)

Algorithm enabled by the ISPU

Table 321. ISPU_CALL_EN4 register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
0(1)	0(1)	0(1)	0(1)	0(1)	0(1)	0(1)	0(1)

Table 322. ISPU_CALL_EN3 register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
0(1)	0(1)	0(1)	0(1)	ISPU_CALL_EN_ALGO15	ISPU_CALL_EN_ALGO14	ISPU_CALL_EN_ALGO13	ISPU_CALL_EN_ALGO12

Table 323. ISPU_CALL_EN2 register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
ISPU_CALL_EN_ALGO11	ISPU_CALL_EN_ALGO10	ISPU_CALL_EN_ALGO9	ISPU_CALL_EN_ALGO8	ISPU_CALL_EN_ALGO7	ISPU_CALL_EN_ALGO6	ISPU_CALL_EN_ALGO5	ISPU_CALL_EN_ALGO4

Table 324. ISPU_CALL_EN1 register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
ISPU_CALL_EN_ALGO3	ISPU_CALL_EN_ALGO2	ISPU_CALL_EN_ALGO1	ISPU_CALL_EN_ALGO0	0(1)	0(1)	0(1)	0(1)

1. *This bit must be set to 0 for the correct operation of the device.*

Table 325. ISPU_CALL_EN register description

ISPU_CALL_EN_ALGO[15:0]	0x0000	R/W	Each bit fires an interrupt that runs the associated algorithm.

12.33 EXT_CTRL (0x80900 - R/W)

The ISPU core can access many registers in the sensor. The access to the sensor registers in write mode is mutually exclusive between the ISPU core and the device serial host interface. The switch is controlled by the ISPU core through this EXT_CTRL register.

As write access to the sensor registers is mutually exclusive between the ISPU core and the device serial host interface, it is recommended to implement a software handshake when switching control.

Table 326. EXT_CTRL register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
0(1)	0(1)	0(1)	0(1)	0(1)	0(1)	0(1)	EN_WRITE_EXT_REG

1. *This bit must be set to 0 for the correct operation of the device.*

Table 327. EXT_CTRL register description

EN_WRITE_EXT_REG	0x0	R/W	When set to 1, the ISPU core has register map access in R/W-mode to the allowed registers specified in the external registers map table. Note: Write access is mutually exclusive between the ISPU core and the device serial host interface.

12.34 FIFO_CTRL1_ISPU (0x80910 – R/W)

FIFO control register when FIFO is controlled by the ISPU core

Table 328. FIFO_CTRL1_ISPU register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
WTM_7	WTM_6	WTM_5	WTM_4	WTM_3	WTM_2	WTM_1	WTM_0

Table 329. FIFO_CTRL1_ISPU register description

WTM_[7:0]	0x00	R/W	FIFO watermark threshold 1 LSB = 1 sensor (9 bytes) + TAG (1 byte) written in FIFO The watermark flag rises when the number of bytes written in the FIFO is greater or equal to the threshold level.

12.35 FIFO_CTRL2_ISPU (0x80911 – R/W)

FIFO control register when FIFO is controlled by the ISPU core

Table 330. FIFO_CTRL2_ISPU register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
0	0	DEC_TS_ BATCH_1	DEC_TS_ BATCH_0	WTM_11	WTM_10	WTM_9	WTM_8

Table 331. FIFO_CTRL2_ISPU register description

DEC_TS_BATCH_[1:0]	0x0	R/W	Selects the decimation for timestamp batching in FIFO. The batch data rate for the timestamp is the data rate of the fastest batched main sensor (accelerometer, temperature) divided by the decimation factor. (00: timestamp not batched in FIFO (default); 01: only the first timestamp sample is batched when FIFO is enabled. No periodic decimation is applied; 10: decimation 128; 11: decimation 256)
WTM_[11:8]	0x0	R/W	FIFO watermark threshold 1 LSB = 1 sensor (9 bytes) + TAG (1 byte) written in FIFO The watermark flag rises when the number of bytes written in the FIFO is greater or equal to the threshold level.

12.36 FIFO_CTRL3_ISPU (0x80912 – R/W)

FIFO control register when FIFO is controlled by the ISPU core

Table 332. FIFO_CTRL3_ISPU register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
STOP_ON_FTH	0	0	T_BATCH	XL_BATCH	FIFO_MODE_2	FIFO_MODE_1	FIFO_MODE_0

Table 333. FIFO_CTRL3_ISPU register description

STOP_ON_FTH	0x0	R/W	Sensing chain FIFO stops storing samples once the threshold level is reached Default value: 0 (0: FIFO storage of samples does not stop at the threshold (default); 1: FIFO storage of samples stops at the threshold level defined by the FIFO watermark threshold)
T_BATCH	0x0	RW	Enables batching from the embedded temperature sensor. Default value: 0 (0: temperature sensor batching off (default); 1: temperature sensor batching on)
XL_BATCH	0x0	R/W	Enables batching from the vibration sensor. Default value: 0 (0: vibration sensor batching off (default); 1: vibration sensor batching on)
FIFO_MODE[2:0]	0x0	R/W	FIFO mode selection (000: bypass mode; 001: FIFO mode; 010: continuous mode; 011: continuous, FIFO on trigger mode (if trigger = 0 stays in continuous mode; if trigger = 1 stays in FIFO mode) 100: bypass, continuous on trigger mode (if trigger = 0, stays in bypass mode; if trigger = 1, stays in continuous mode) 101: reserved; 110: reserved; 111: bypass, FIFO on trigger mode (if trigger = 0, stays in bypass mode; if trigger = 1, stays in FIFO mode))

12.37 ISPU_LOPRIO_INT (0x80978 – R/W)

Low-priority interrupt control registers

Table 334. ISPU_LOPRIO_INT register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
0	0	ISPU_DATA_VALID_TEMP	ISPU_LOPRIO_USER_TRG	ISPU_LOPRIO_EXT_TRG	0	ISPU_LOPRIO_SLEEP_CNT	ISPU_LOPRIO_FIFO

Table 335. ISPU_LOPRIO_INT register description

ISPU_DATA_VALID_TEMP	0x0	R/clear only	Temperature sensor data valid for the low-priority interrupt is latched here.
ISPU_LOPRIO_USER_TRG	0x0	R/clear only	User trigger for the low-priority interrupt is latched here.
ISPU_LOPRIO_EXT_TRG	0x0	R/clear only	External trigger for the low-priority interrupt is latched here.
ISPU_LOPRIO_SLEEP_CNT	0x0	R/clear only	Sleep counter for the low-priority interrupt is latched here.
ISPU_LOPRIO_FIFO	0x0	R/clear only	FIFO watermark/full for the low-priority interrupt is latched here.

12.38 ISPU2FIFO_FLAG (0x8097C – R/W)

Exchange bit between FIFO (clear only) and the ISPU (set only)

Table 336. ISPU2FIFO_FLAG register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
0	0	0	0	0	0	0	ISPU2FIFO_FLAG

Table 337. ISPU2FIFO_FLAG register description

ISPU2FIFO_FLAG	0x0	R	Set this bit to 1 to request the write to FIFO and wait for it to go to 0 for the request completion.

12.39 FIFO2ISPU_TAG (0x8097D – R/W)

TAG 8-bit value when reading the FIFO from the ISPU

Table 338. FIFO2ISPU_TAG register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
FIFO2ISPU_TAG_7	FIFO2ISPU_TAG_6	FIFO2ISPU_TAG_5	FIFO2ISPU_TAG_4	FIFO2ISPU_TAG_3	FIFO2ISPU_TAG_2	FIFO2ISPU_TAG_1	FIFO2ISPU_TAG_0

Table 339. FIFO2ISPU register description

FIFO2ISPU_TAG_[7:0]	0x00	R	Identifies data type when reading the FIFO from the ISPU. Refer to Table 111. FIFO TAG names.

12.40 FIFO2ISPU_CTRL (0x8097E – R/W)

Handshake signals for reading the FIFO from the ISPU

Table 340. FIFO2ISPU_CTRL register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
0	0	0	0	0	FIFO2ISPU_READ_EN	FIFO2ISPU_READ_REQ	FIFO2ISPU_READ_VALID

Table 341. FIFO2ISPU_CTRL register description

FIFO2ISPU_READ_EN	0x0	R/W	Enables reading data from FIFO
FIFO2ISPU_READ_REQ	0x0	R/W	Requests the read of one FIFO entry
FIFO2ISPU_READ_VALID	0x0	R/W	If this bit is 1, the read request has been executed and the data is available in the ISPU_ARAW registers.

12.41 ISPU2FIFO_INT (0x8097F – R/W)

FIFO batching trigger from the ISPU

Table 342. ISPU2FIFO_CTRL register

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
0	0	0	0	0	0	0	ISPU2FIFO_INTERRUPT

Table 343. ISPU2FIFO_CTRL register description

ISPU2FIFO_INTERRUPT	0x00	R/W	If FIFO_TRIGGER_CFG is set to 0 (FIFO trigger from the ISPU), when ISPU writes 1 in this field, FIFO changes batching mode according to the FIFO_MODE setting.

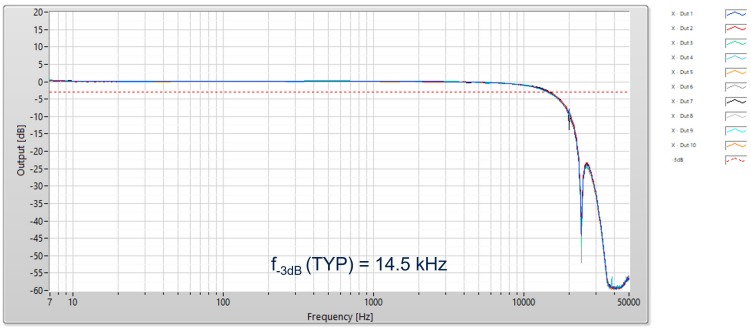
13 Typical performance characteristics

13.1 Frequency response measurements

The frequency response of the IIS3DWB10IS, measured on a mechanical shaker, is indicated in the following figures.

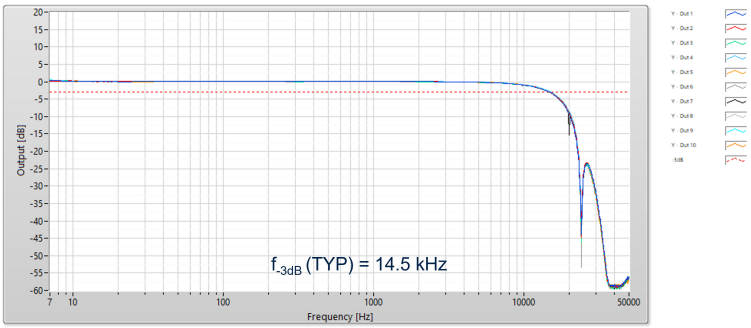
Measurements have been performed with the IIS3DWB10IS configured in continuous mode (CM) with 40 kHz ODR, FS = 50 *g*, VDD = 3 V and Tambient = 25°C.

Figure 56. Frequency response X-axis



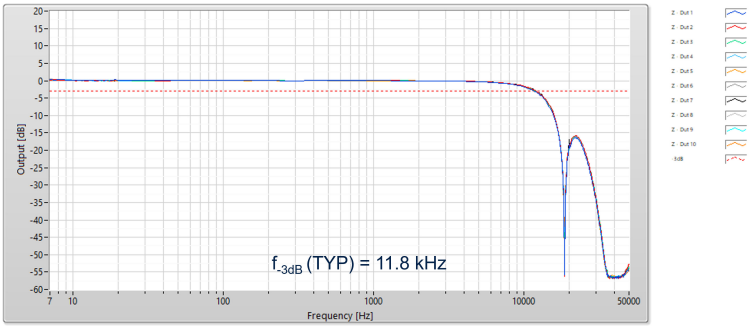
Note: Characterization data on 10 parts. Not measured in production and not guaranteed.

Figure 57. Frequency response Y-axis



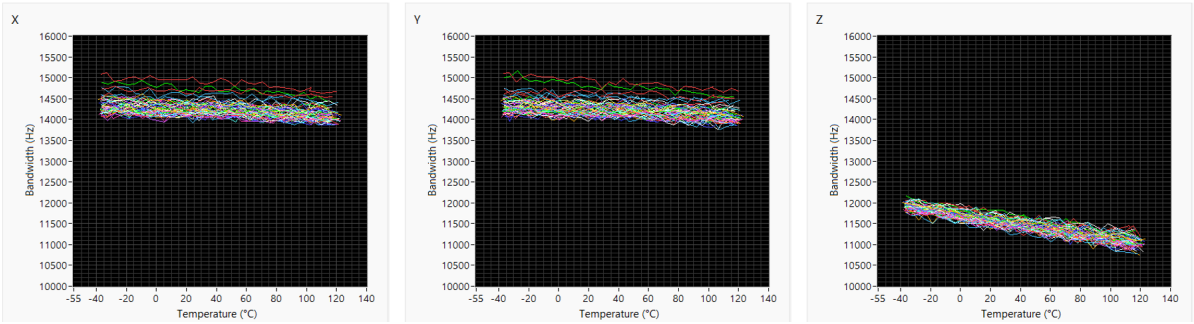
Note: Characterization data on 10 parts. Not measured in production and not guaranteed.

Figure 58. Frequency response Z-axis



Note: Characterization data on 10 parts. Not measured in production and not guaranteed.

Figure 59. f_{-3dB} change versus temperature

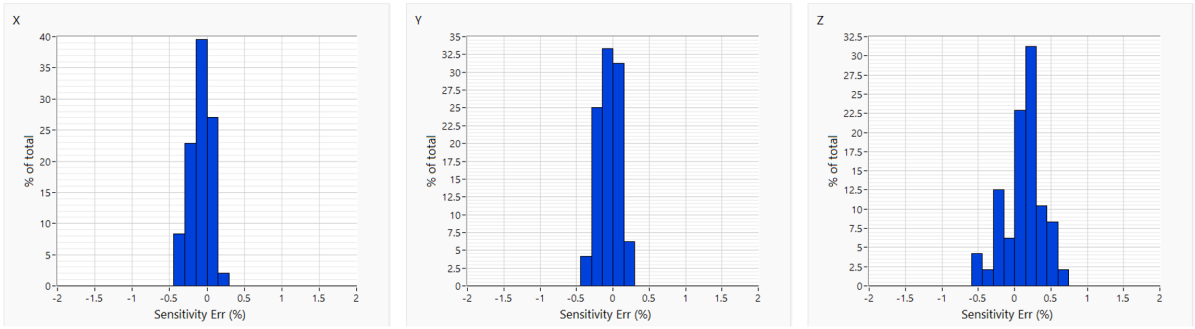


Note: Characterization data. Not measured in production and not guaranteed.

13.2 Sensitivity change versus temperature

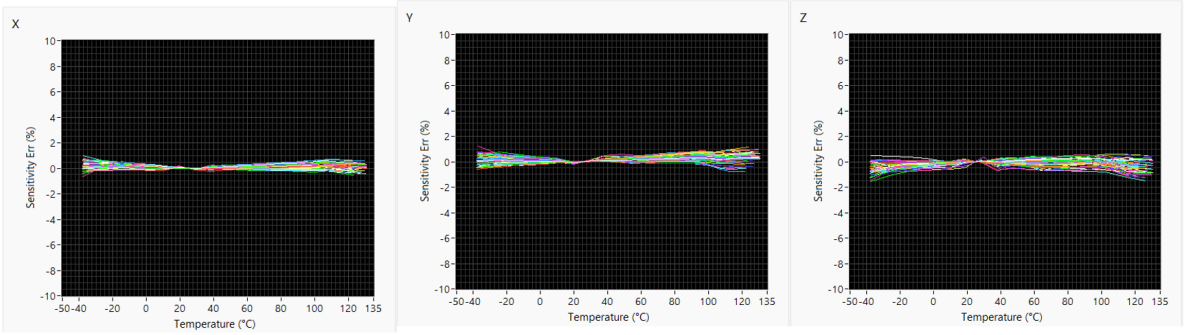
Measurements have been performed with the IIS3DWB10IS soldered on a PCB, configured in continuous mode (CM) with 40 kHz ODR, FS = 50 *g* and VDD = 3 V.

Figure 60. Sensitivity



Note: Characterization data. Not measured in production and not guaranteed.

Figure 61. Sensitivity change versus temperature

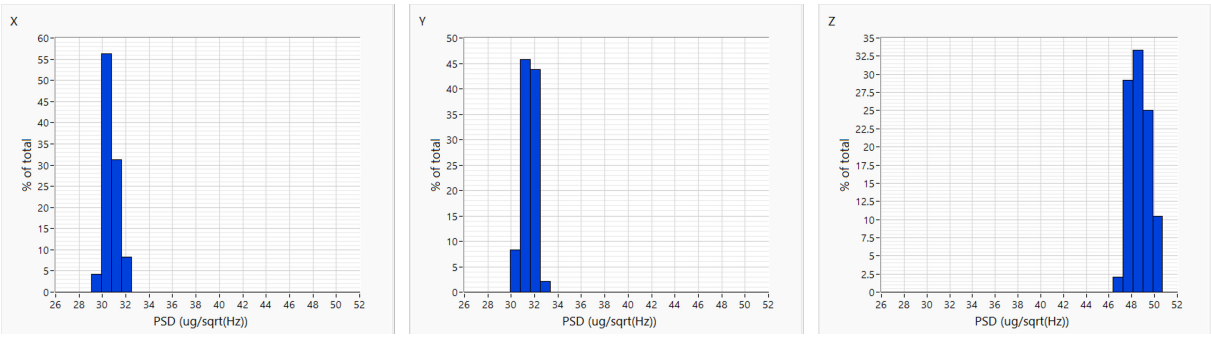


Note: Characterization data. Not measured in production and not guaranteed.

13.3 Noise density

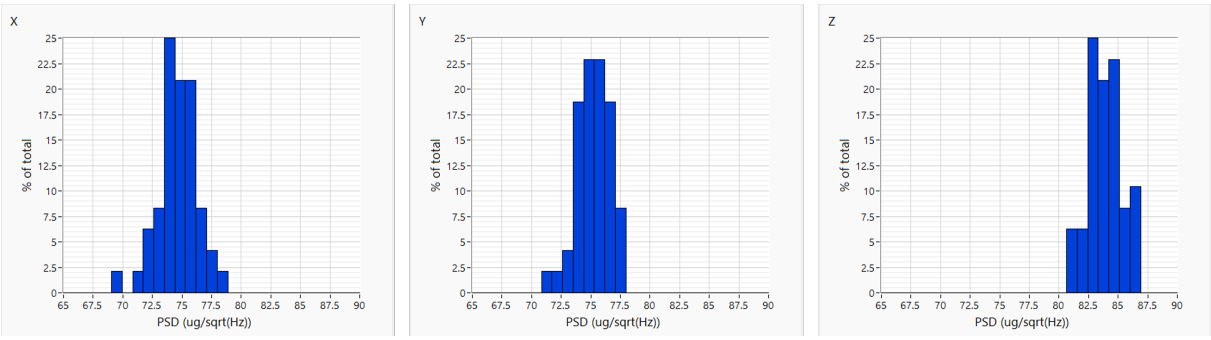
Measurements have been performed with the IIS3DWB10IS soldered on a PCB, configured in continuous mode (CM) with 40 kHz ODR, VDD = 3 V and at two different FS: 50 *g* and 200 *g*.

Figure 62. Noise density @ FS 50 *g*



Note: Characterization data. Not measured in production and not guaranteed.

Figure 63. Noise density @ FS 200 *g*

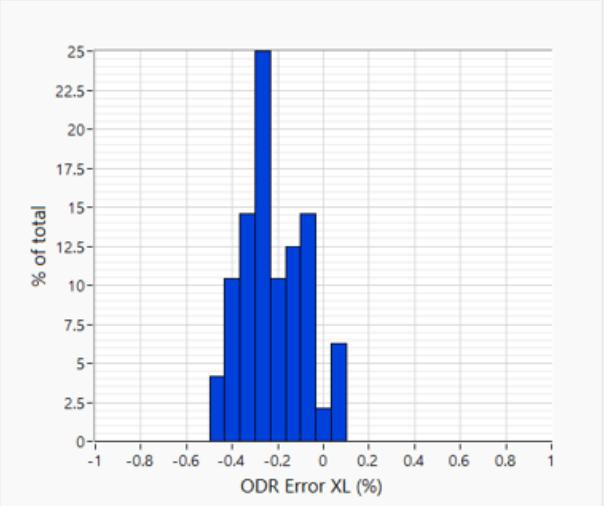


Note: Characterization data. Not measured in production and not guaranteed.

13.4 ODR accuracy

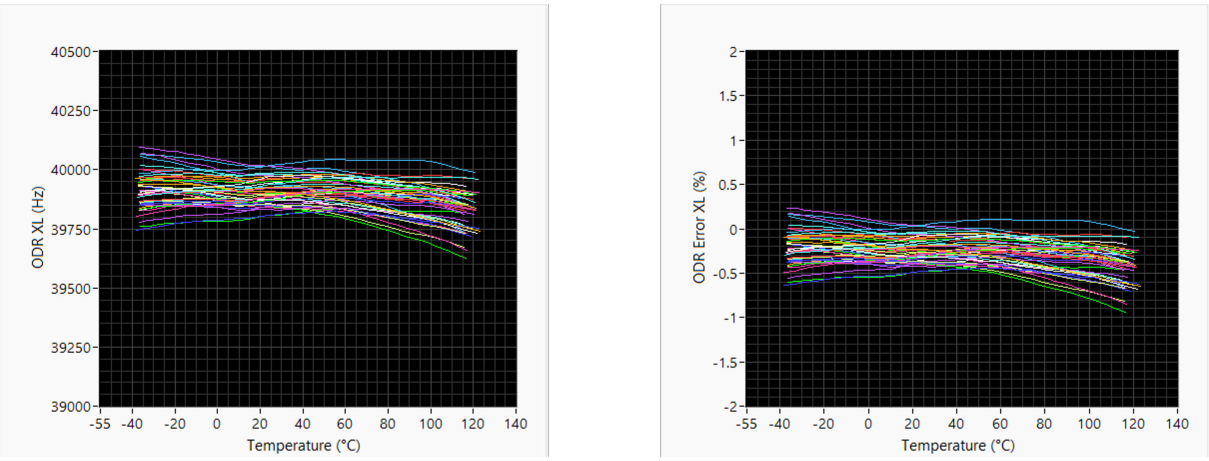
Measurements have been performed with the IIS3DWB10IS soldered on a PCB, configured in continuous mode (CM) with VDD = 3 V and 40 kHz ODR.

Figure 64. ODR accuracy



Note: Characterization data. Not measured in production and not guaranteed.

Figure 65. ODR change versus temperature



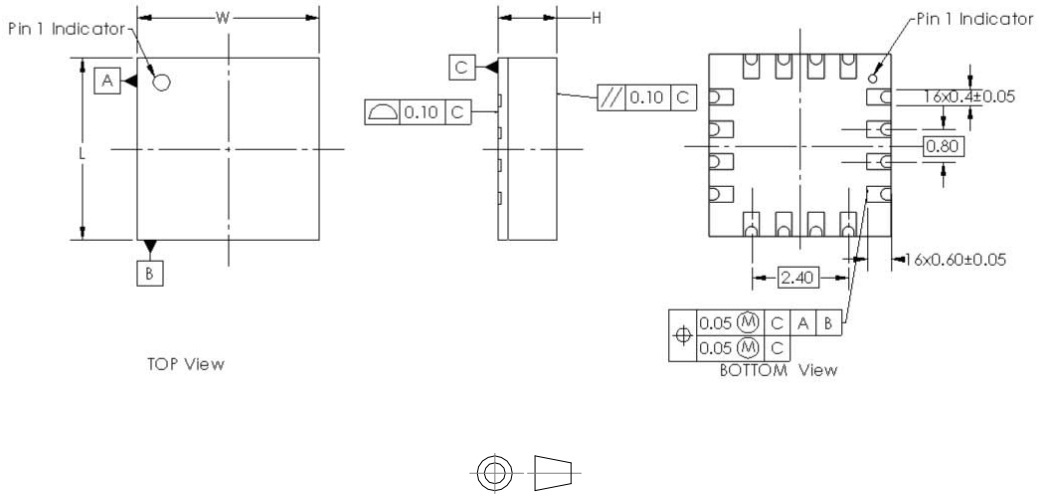
Note: Characterization data. Not measured in production and not guaranteed.

14 Package information

To meet environmental requirements, ST offers these devices in different grades of ECOPACK packages, depending on their level of environmental compliance. ECOPACK specifications, grade definitions, and product status are available at: www.st.com. ECOPACK is an ST trademark.

14.1 LGA-16L package information

Figure 66. LGA-16L 4.5 x 4.5 x 1.5 mm package outline and mechanical data



Dimensions are in millimeter unless otherwise specified
General tolerance is +/-0.1mm unless otherwise specified

OUTER DIMENSIONS

ITEM	DIMENSION [mm]	TOLERANCE [mm]
Length [L]	4.50	±0.05
Width [W]	4.50	±0.05
Height [H]	1.5	MAX

DM00953946_2

Revision history

Table 344. **Document revision history**

Date	Version	Changes
26-May-2026	1	Initial release

Contents

List of tables

List of figures

IMPORTANT NOTICE – READ CAREFULLY

STMicroelectronics NV and its subsidiaries (“ST”) reserve the right to make changes, corrections, enhancements, modifications, and improvements to ST products and/or to this document at any time without notice.

In the event of any conflict between the provisions of this document and the provisions of any contractual arrangement in force between the purchasers and ST, the provisions of such contractual arrangement shall prevail.

The purchasers should obtain the latest relevant information on ST products before placing orders. ST products are sold pursuant to ST’s terms and conditions of sale in place at the time of order acknowledgment.

The purchasers are solely responsible for the choice, selection, and use of ST products and ST assumes no liability for application assistance or the design of the purchasers’ products.

No license, express or implied, to any intellectual property right is granted by ST herein.

Resale of ST products with provisions different from the information set forth herein shall void any warranty granted by ST for such product.

If the purchasers identify an ST product that meets their functional and performance requirements but that is not designated for the purchasers' market segment, the purchasers shall contact ST for more information.

ST and the ST logo are trademarks of ST. For additional information about ST trademarks, refer to www.st.com/trademarks. All other product or service names are the property of their respective owners.

Information in this document supersedes and replaces information previously supplied in any prior versions of this document.