AN4833
Application note

Measuring pressure data from ST's LPS22HB digital pressure sensor

Introduction

This application note describes the methods and techniques for measuring pressure data from the LPS22HB.

The LPS22HB is an ultra-compact piezo-resistive absolute pressure sensor which functions as a digital output barometer. The device comprises a sensing element and an IC interface which communicates through I^2C or SPI from the sensing element to the application. The sensing element, which detects absolute pressure, consists of a suspended membrane manufactured using a dedicated process developed by ST.

This document does not modify the content of the official datasheet. Please refer to the datasheet for parameter specifications.

Contents

List of tables

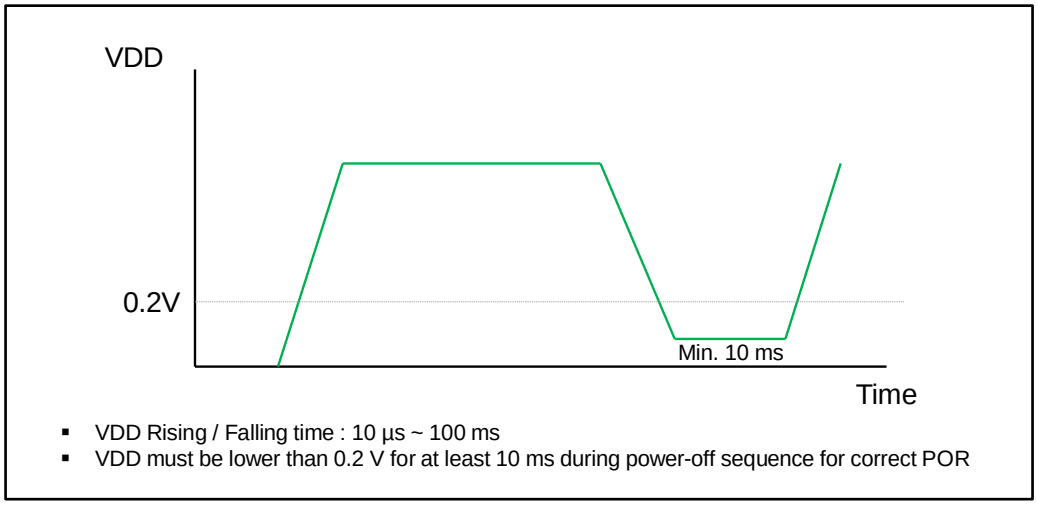
List of figures

1 Overview

The LPS22HB features three operating modes: power-down mode, one-shot mode and continuous mode.

The device offers a wide VDD voltage range from 1.7 V to 3.6 V and a VDDIO range from 1.7 V to VDD +0.1 V. In order to avoid potential conflicts, during the power-on sequence it is recommended to set the lines connected to the device IO pins to high-impedance state on the host side. Furthermore, to guarantee proper power-off of the device it is recommended to maintain the duration of the VDD line to GND (less than 0.2 V) for at least 10 ms as illustrated in the figure below.

Figure 1: LPS22HB power-on/off sequence



After the power supply is applied, the LPS22HB requires a boot procedure of 3 ms to load the trimming parameters and then the device is ready to communicate with the master for register configuration and pressure measurements.

1.1 Power-down mode

When the device is in power-down mode, almost all internal blocks of the device are switched off to minimize power consumption. I²C interface is active to allow communication with the device. The content of the configuration registers is preserved and the output data registers are not updated. Therefore the last sampled data are kept in memory once the device goes in power-down mode.

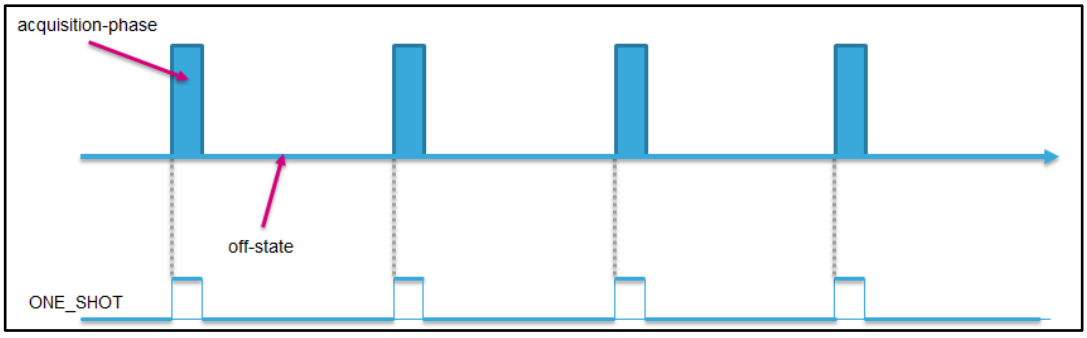
The device is in power-down mode when the ODR[2,0] bits in CTRL_REG1 (10h) are set to ‘000’.

1.2 One-shot mode

When the device is in one-shot mode, it acquires a new conversion when it is requested. After the acquisition has been completed, the device automatically is set to power-down mode.

One-shot mode can be enabled when the device is in power-down mode (ODR[2,0] bits in CTRL_REG1 (10h) set to '000') and when the ONE_SHOT bit in CTRL_REG2 (11h) is set to '1'.

Figure 2: One-shot mode



Once the acquisition is completed and the output registers updated, the device automatically enters power-down mode and the ONE_SHOT bit is self-cleared.

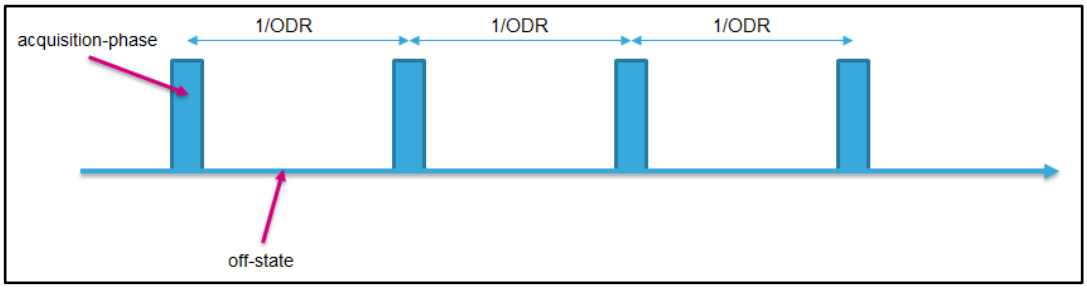
1.3 Continuous mode

When the ODR[2,0] bits in CTRL_REG1 (10h) register are set to a value different than '000', the device is in continuous mode and automatically acquires a set of data (pressure and temperature) at the frequency selected through the ODR[2,0] bits in CTRL_REG1 (10h) register.

Table 1: Sampling frequency selection

ODR2	ODR1	ODR0	Pressure ODR	Pressure ODR
0	0	0	Power-down / one shot mode enabled	
0	0	1	1 Hz	1 Hz
0	1	0	10 Hz	10 Hz
0	1	1	25 Hz	25 Hz
1	0	0	50 Hz	50 Hz
1	0	1	75 Hz	75 Hz

Figure 3: Continuous mode



1.4 Resolution configuration

The LPS22HB can be configured in two resolution modes that can be used in both one-shot mode and continuous mode.

The LC_EN bit in the RES_CONF (1Ah) register defines the resolution mode:

- LC_EN set to ‘0': normal mode – enabled by default
- LC_EN set to ‘1': low-current mode

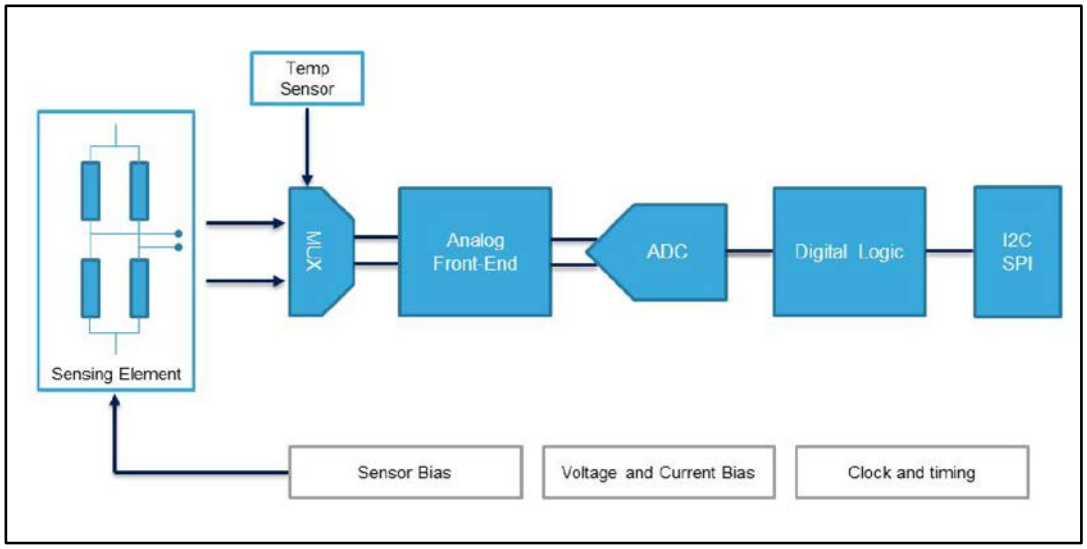
In normal mode, the device is optimized to reduce the noise, while in low-current mode the device minimizes current consumption.

For proper behavior of the pressure sensor, the LC_EN bit must be changed only when the device is in power-down.

2 Device architecture

The LPS22HB is a piezoresistive absolute pressure sensor which functions as a digital output barometer. The device comprises a sensing element and an IC interface which communicates through I²C or SPI from the sensing element to the application.

Figure 4: LPS22HB architecture block diagram



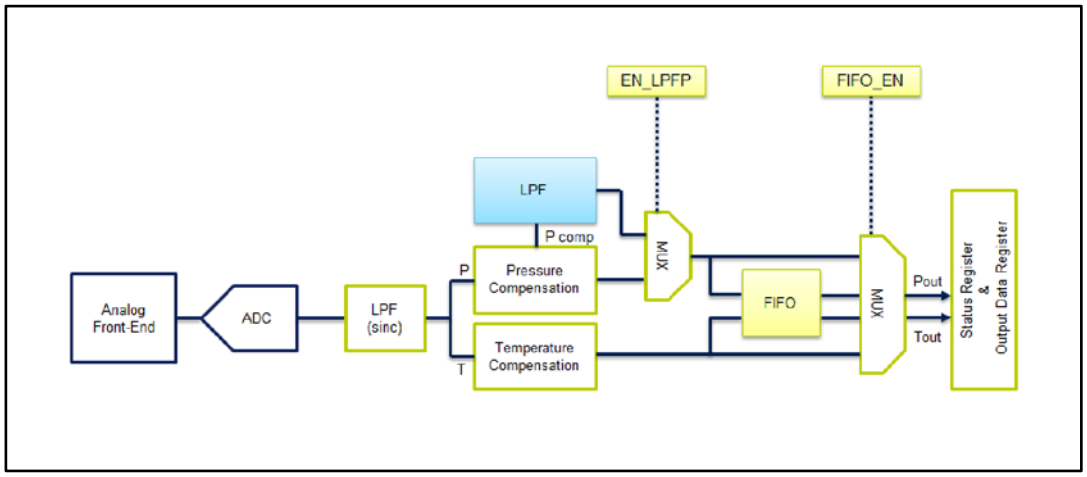
The sensing element, which detects absolute pressure, consists of a suspended membrane manufactured using a dedicated process developed by ST.

Information on how to interpret the pressure and the temperature readings can be found in the application note *"TN1229: How to interpret pressure and temperature readings in the LPS22HB pressure sensor", available on www.st.com*.

2.1 Digital low-pass filter

The LPS22HB embeds an additional low-pass filter that can be applied on the pressure readout path when the device is in continuous mode.

Figure 5: Device architecture



The optional digital filter can be enabled by setting the EN_LPFP bit in CTRL_REG1 (10h) and its bandwidth can be configured acting on the LPFP_CFG bit in CTRL_REG1 (10h) register.

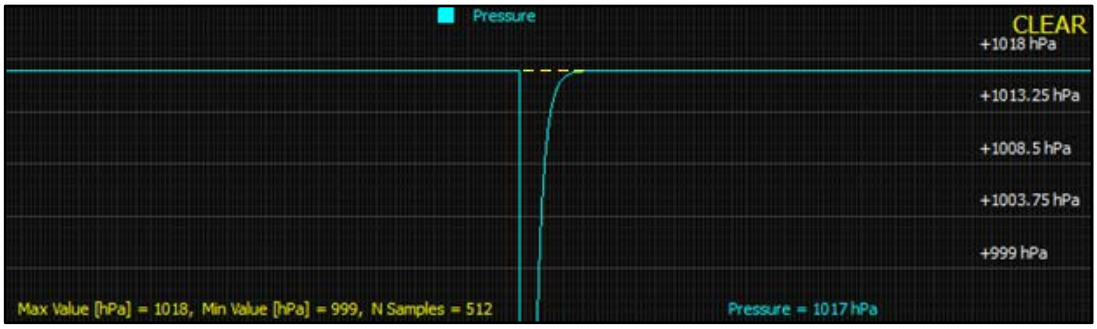
Table 2: Low-pass filter settings

EN_LPFP	LPF_CFG	Additional low-pass filter status	Device bandwidth
0	X	Disabled	ODR/2
1	0	Enabled	ODR/9
1	1	Enabled	ODR/20

2.1.1 Filter reset

If the LPFP is active, in order to avoid the transitory phase, the filter can be reset by reading this register before generating pressure measurements.

Figure 6: LPFP reset



Every time the LPFP is used, it is recommended to perform a reset of the filter immediately after the ODR has been set.

2.1.2 Examples of device LPF configurations

Table 3: ODR bits in CTRL_REG1 (10h) set to '100': ODR = 75 Hz

EN_LPFP	LPF_CFG	Additional low-pass filter status	Device bandwidth [Hz]
0	x	Disabled	37.5
1	0	Disabled	8.3
1	1	Enabled	3.75

Table 4: ODR bits in CTRL_REG1 (10h) set to '001': ODR = 1 Hz

EN_LPFP	LPF_CFG	Additional low-pass filter status	Device bandwidth [Hz]
0	x	Disabled	37.5
1	0	Disabled	8.3
1	1	Enabled	3.75

3 FIFO

The LPS22HB has embedded a 32-slot FIFO for pressure and temperature data in order to limit intervention by the host processor and facilitate post-processing data for event recognition.

FIFO use allows consistent power saving for the system, it can wake up only when needed and burst the significant data out from the FIFO.

The FIFO buffer can work according to seven different modes that guarantee a high-level of flexibility during application development:

1. Bypass mode
2. FIFO mode
3. Stream mode
4. Stream-to-FIFO mode
5. Bypass-to-Stream
6. Bypass-to-FIFO mode
7. Dynamic-Stream mode

Description of the FIFO is provided in *Section 4.1: "FIFO description"* , while examples of FIFO modes are given in *Section 8: "Appendix"*.

The FIFO can also be programmed to generate interrupt events on the INT_DRDY pin.

3.1 FIFO description

The FIFO buffer is able to store up to 32 pressure and temperature samples of 24-bit and 16-bit words respectively.

The data sample sets consist of 5 bytes (PRESS_OUT_XL, PRESS_OUT_L, PRESS_OUT_H, TEMP_OUT_L, TEMP_OUT_H) and they are released to the FIFO at the selected output data rate (ODR). The new sample set is placed in the first empty FIFO slot until the buffer is full, therefore, the oldest value is overwritten.

3.2 Retrieving data from FIFO

FIFO data is read from the PRESS_OUT registers (28h, 29h, 2Ah) and TEMP_OUT registers (2Bh, 2Ch).

A read operation from the PRESS_OUT registers provides the pressure data stored in the FIFO, while reading the TEMP_OUT registers provides temperature data. Every time a data set is read from the FIFO, the oldest entry is placed in the PRESS_OUT registers. Both single-read and multiple-read operations can be performed.

In case of multiple reads, the device automatically updates the reading address and it rolls back to 28h when register 2Ch is reached. To read all FIFO levels in a multiple-byte read, 160 bytes (5 output registers by 32 levels) must be read.

Information on interpreting the pressure and the temperature readings can be found in the application note TN1229: How to interpret pressure and temperature readings in the LPS22HB pressure sensor, available on *www.st.com*.

3.3 FIFO settings and control

Upon device power-up, FIFO is not enabled, and the pressure and temperature data are not stored in the FIFO, but stored in the output temperature and pressure registers.

The FIFO can be controlled using three registers:

- CTRL_REG2 for enabling the FIFO and watermark level definition
- FIFO_CTRL(14h) for setting the FIFO mode and watermark level
- FIFO_STATUS(26h) for reading the FIFO status during operation

To enable the FIFO buffer, the FIFO_EN bit in CTRL_REG2 (11h) has to be set to '1' and the FIFO mode of operation is defined by the F_MODE[2:0] bits in FIFO_CTRL (14h), as indicated in the table below.

Table 5: FIFO mode selection

F_MODE2	F_MODE1	F_MODE0	FIFO mode selection
0	0	0	Bypass mode
0	0	1	FIFO mode
0	1	0	Stream mode
0	1	1	Stream-to-FIFO mode
1	0	0	Bypass-to-Stream Mode
1	0	1	Reserved
1	1	0	Dynamic-Stream mode
1	1	1	Bypass-to-FIFO mode

The FIFO buffer can store up to 32 levels of data. The FIFO depth can be limited by setting the STOP_ON_FTH bit in CTRL2(11h) to ‘1’ and defining the required FIFO depth is performed by defining a watermark level with the WTM bits in FIFO_CTRL(14h). To convert the WTM bits into the number of levels stored in the FIFO, it is sufficient to convert from binary to decimal the value of the WTM bits and add 1. As an example, if the FIFO depth needs to be limited to 12 levels, the WTM bits have to be set to ‘01011’.

The FIFO_STATUS (26h) register provides information about the FIFO status:

- FTH_FIFO bit goes to ‘1’ if the number of unread samples is greater than or equal to the watermark level selected by WTM[4:0] in FIFO_CTRL (14h).
- OVR bit goes to ‘1’ if the FIFO buffer is full and at least one sample in the FIFO has been overwritten
- FSS[5:0] provides information on the data stored in the FIFO buffer.
  - FSS is equal to ‘000001' when 1 data set is stored in the FIFO
  - FSS is equal to ‘100000’ when 32 data sets are stored in the FIFO

3.4 FIFO modes

3.4.1 Bypass mode

In Bypass mode (FIFO_CTRL(F_MODE2:0) = ‘000’) the FIFO is not operational and the buffer remains empty. The pressure and temperature values are sent directly to the PRESS_OUT and TEMP_OUT registers.

3.4.2 FIFO mode

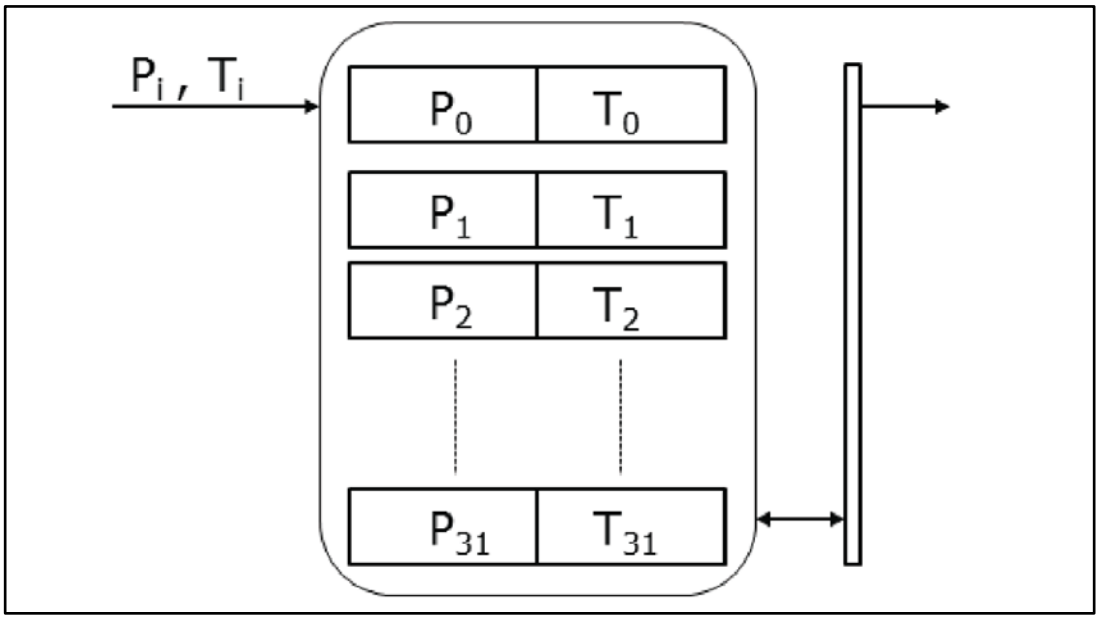
In FIFO mode (FIFO_CTRL(F_MODE2:0) = ‘001’) the pressure and temperature acquired are stored in the buffer: the content of the registers PRESS_OUT_XL (0x28), PRESS_OUT_L (0x29), PRESS_OUT_H (0x2A), TEMP_OUT_L (0x28) and TEMP_OUT_H (0x2A) are stored in the FIFO.

When the FIFO is full or the watermark is reached, the update in the FIFO is stopped until the buffer is read or reset.

It is mandatory to reset the FIFO if the FIFO is full and another sample is collected.

To reset FIFO content, the value '000' must be written in FIFO_CTRL(F_MODE2:0). After this reset command, it is possible to restart FIFO mode, writing the value '001' in FIFO_CTRL(F_MODE2:0).

Figure 7: FIFO mode

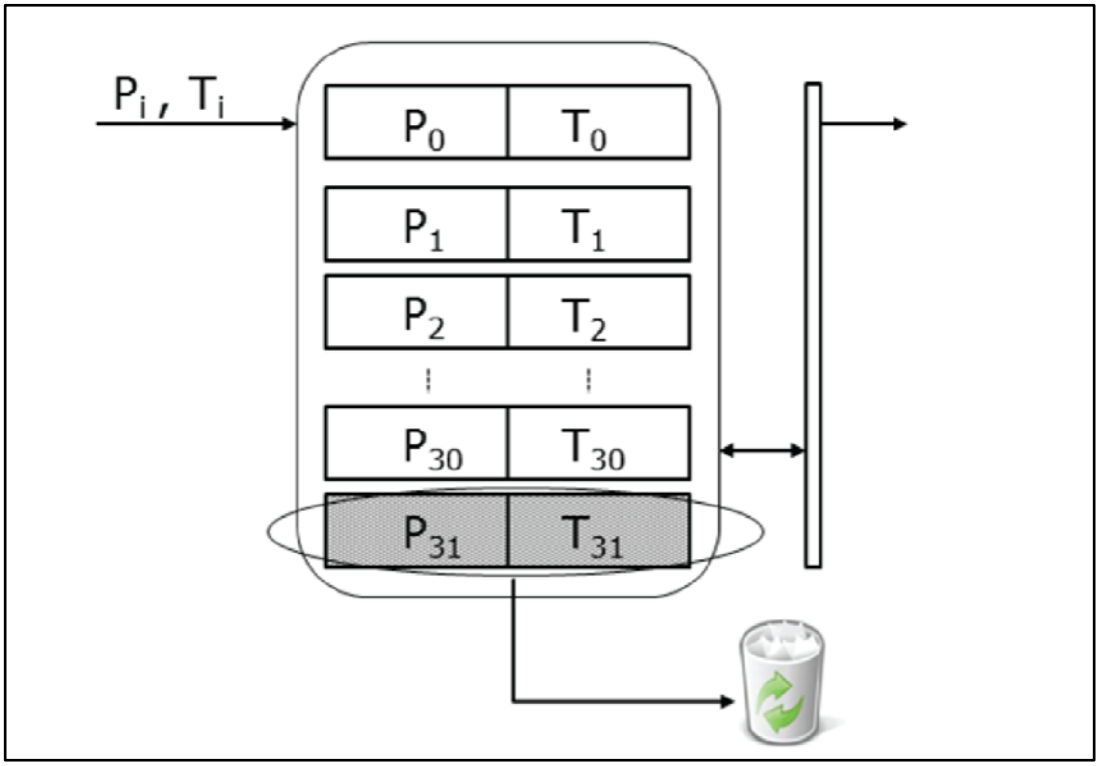


3.4.3 Stream mode

In Stream mode (FIFO_CTRL(F_MODE2:0) = ‘010’), the pressure and temperature acquired are stored in the buffer: the content of the registers PRESS_OUT_XL (0x28), PRESS_OUT_L (0x29), PRESS_OUT_H (0x2A), TEMP_OUT_L (0x28) and TEMP_OUT_H (0x2A) are stored in the FIFO. Once the FIFO is full or the watermark level is reached, the new data replace the older data stored in the buffer.

Once the entire FIFO has been read, the last data read remains in the FIFO. When a new sample is acquired, the FIFO_STATUS(FSS5:0) value rises from 0 to 2.

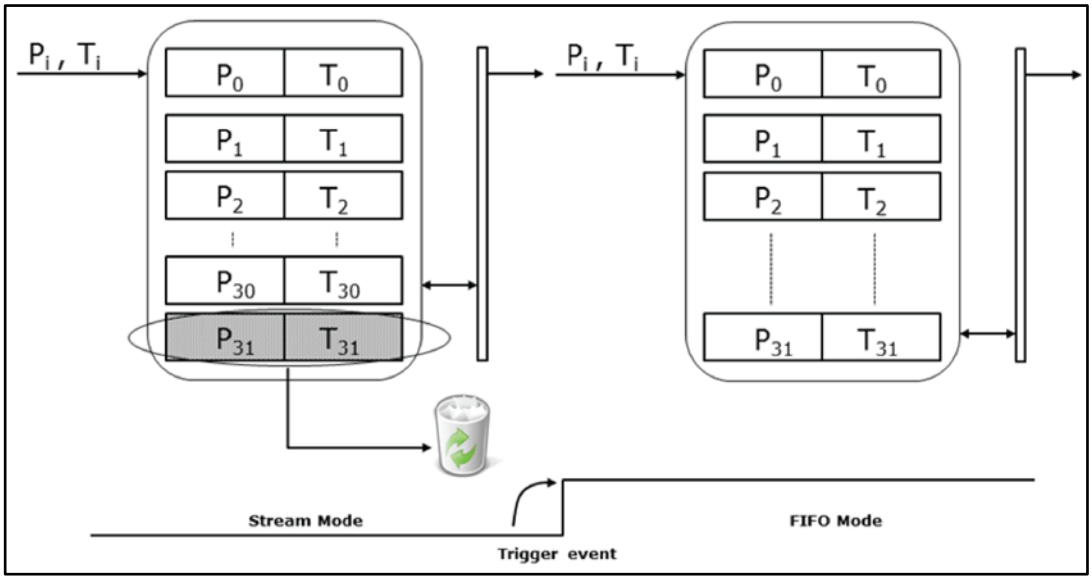
Figure 8: Stream mode



3.4.4 Stream-to-FIFO mode

In Stream-to-FIFO mode (FIFO_CTRL(F_MODE2:0) = ‘011’), the FIFO works in Stream mode until a trigger event is generated and then in FIFO mode. The trigger event can be set through INTERRUPT_CFG (0Bh). If the interrupt is triggered, the INT_SOURCE(IA) bit is equal to '1', and the FIFO switches from Stream to FIFO mode. When the interrupt is de-asserted, the INT_SOURCE(IA) bit is equal to '0', and the FIFO switches back to Stream mode.

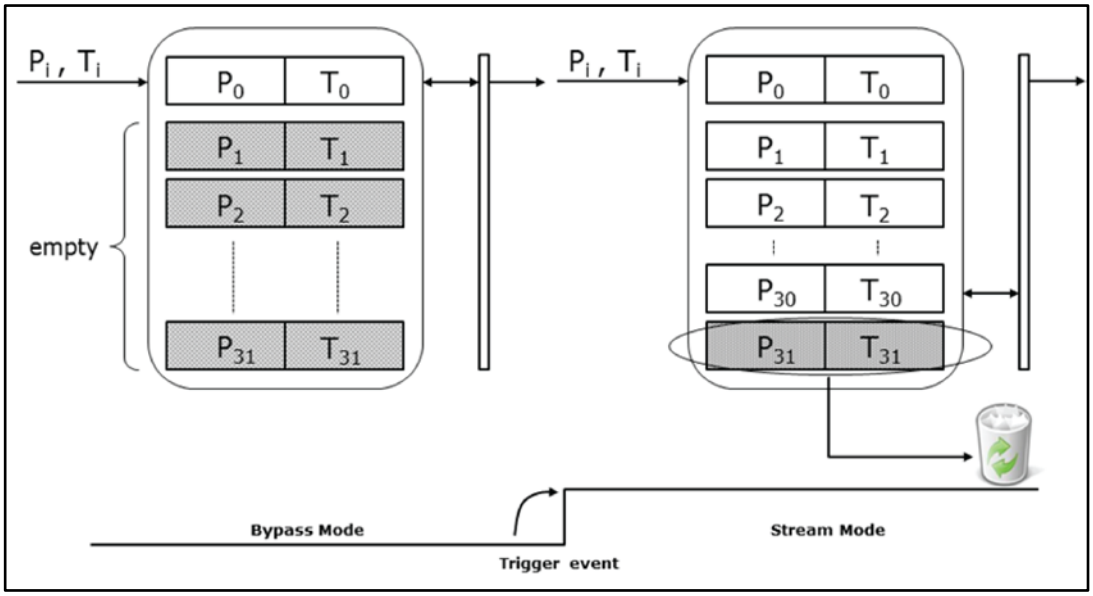
Figure 9: Stream-to-FIFO mode



3.4.5 Bypass-to-Stream mode

In Bypass-to-Stream mode (FIFO_CTRL(F_MODE2:0) = ‘100’), the FIFO works in Bypass mode until a trigger event is generated and then in Stream mode. The trigger event can be set through INTERRUPT_CFG (0Bh). If the interrupt is triggered, the INT_SOURCE(IA) bit is equal to '1', and the FIFO switches from Bypass to Stream mode. When the interrupt is de-asserted, the INT_SOURCE(IA) bit is equal to '0', and the FIFO switches back to Bypass mode.

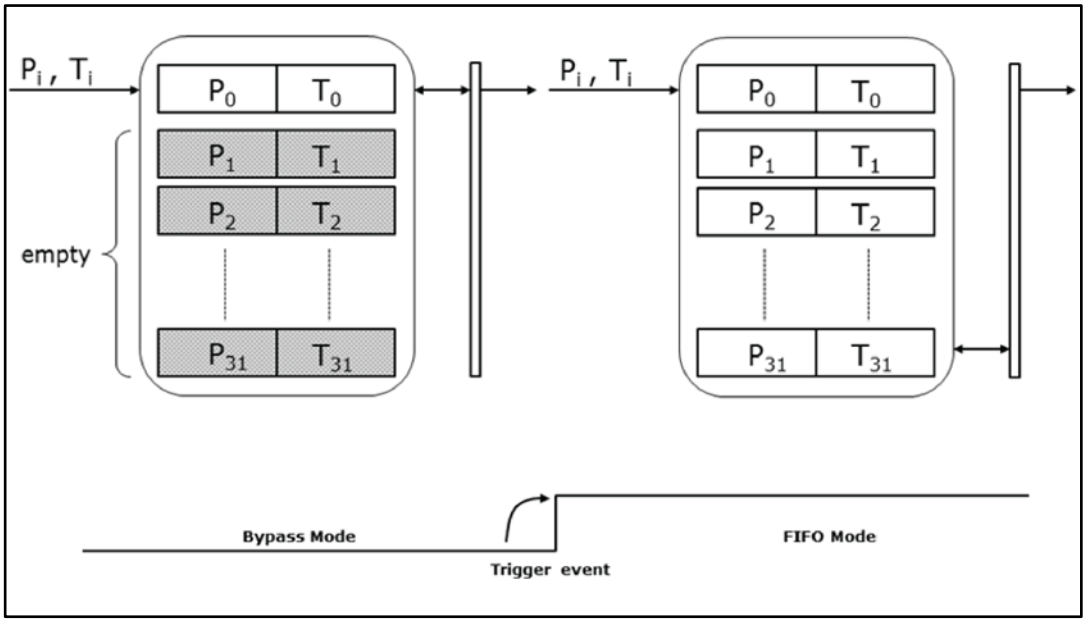
Figure 10: Bypass-to-Stream



3.4.6 Bypass-to-FIFO mode

In Bypass-to-FIFO mode (FIFO_CTRL(F_MODE2:0) = ‘111’), the FIFO works in Bypass mode until a trigger event is generated and then in FIFO mode. The trigger event can be set through INTERRUPT_CFG (0Bh). If the interrupt is triggered, the INT_SOURCE(IA) bit is equal to '1', and the FIFO switches from Bypass to FIFO mode. When the interrupt is de-asserted, INT_SOURCE(IA) bit is equal to '0', and the FIFO switches back to Bypass mode.

Figure 11: Bypass-to-FIFO

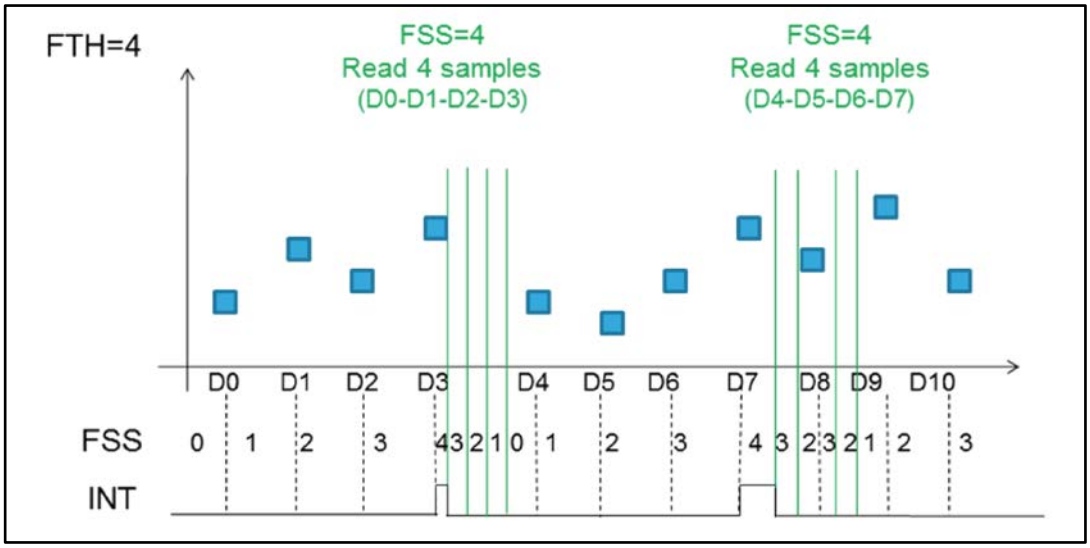


3.4.7 Dynamic Stream mode

In Dynamic Stream mode (FIFO_CTRL(F_MODE2:0) = 110) after emptying the FIFO, the first new sample that arrives becomes the first to be read in a subsequent read burst. In this way, the number of new data available in FIFO does not depend on the previous reading.

In Dynamic Stream mode FIFO_STATUS(FSS5:0) is the number of new pressure and temperature samples available in the FIFO buffer.

Figure 12: Dynamic Stream mode



4 Offset compensation (OPC)

If, after the soldering of the component, a residual offset is still present, it can be removed with a one-point calibration (OPC).

After soldering, the measured offset can be stored in the RPDS (18h, 19h) registers and automatically subtracted from the pressure output registers: the output pressure register PRESS_OUT (28h, 29h and 2Ah) is provided as the difference between the measured pressure and the content of the register 256*RPDS (18h, 19h) (DIFF_EN = ‘0’, AUTOZERO =‘0’, AUTORIFP= ‘0’).

5 Block data update (BDU)

The BDU (Block Data Update) bit is located in CTRL_REG1 (10h) and is used to inhibit the update of the output registers between the reading of upper, middle and lower register parts.

In default mode (BDU = ‘0’), the lower, middle and upper register parts are updated continuously.

When the BDU is activated (BDU = ‘1’), the content of the output registers is not updated until the PRESS_OUT_H register has been read in order to avoid output data corruption. To guarantee correct behavior of the BDU function, PRESS_OUT_H (2Ah) must be the last register to be read.

6 Interrupt mode settings

The LPS22HB can be configured to generate interrupt events related to pressure acquisition and FIFO status. A dedicated pad (INT_DRDY) can be set for selected interrupt events.

The interrupt modes related to pressure acquisition are the following:

- Data available
- Threshold-based

The interrupt modes related to the FIFO are the following:

- FIFO watermark
- FIFO full
- FIFO overrun

Interrupt examples can be found in *Section 8: "Appendix"*.

6.1 Interrupt events related to pressure acquisition

6.1.1 Data available

If enabled, it is possible to identify when a new pressure or temperature data is generated. Every time new pressure data is generated, the bit P_DA in STATUS (27h) register is set to ‘1’. This can be also be made available on the INT_DRDY pin.

Every time new temperature data is generated, the bit T_DA in STATUS (27h) register is set to ‘1’.

6.1.2 Threshold-based

With the LPS22HB pressure sensor, it is possible to generate an interrupt event based on a user-defined threshold. To enable this functionality, the DIFF_EN bit in INTERRUPT_CFG (0Bh) register must be set to ‘1’ and the threshold values stored in the THS_P registers (0Dh and 0Ch). The threshold value for pressure interrupt generation is a 15-bit unsigned right-justified value composed of THS_P_H (0Dh) and THS_P_L (0Ch). The value is expressed as:

Interrupt threshold (hPA) = ±THS_P / 16

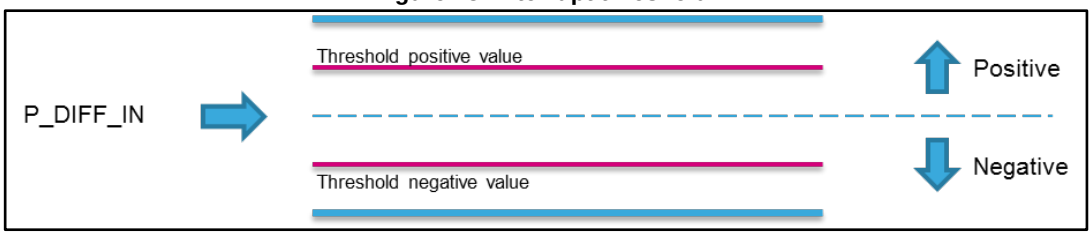
When DIFF_EN=1, the PHE bit or PLE bit (or both bits) in INTERRUPT_CFG (0Bh) have to be enabled. The PHE and PLE bits enable the interrupt generation on the positive or negative event, respectively.

When DIFF_EN is enabled and AUTOZERO or AUTORIFP is enabled, the defined pressure threshold values in THS_P (0Ch, 0Dh) are compared with:

P_DIFF_IN = measured pressure – pressure reference

The value of the pressure reference is assigned depending on the AUTOZERO and AUTORIFP modes given in *Section 7.1.2.1: "AUTOZERO mode"* and *Section 7.1.2.2: "AUTORIFP mode"*.

Figure 13: Interrupt threshold



6.1.2.1 AUTOZERO mode

When the AUTOZERO bit is set to ‘1’, the measured pressure is used as the pressure reference in the REF_P (15h, 16h and 17h) registers. From this point on, the output pressure registers PRESS_OUT (28h, 29h and 2Ah) are updated and the same value is used for the interrupt generation.

PRESS_OUT = measured pressure – REF_P

After the first conversion, the AUTOZERO bit is automatically set to ‘0’. To return back to normal mode, the RESET_AZ bit in INTERRUPT_CFG (0Bh) register has to be set to ‘1’.

6.1.2.2 AUTORIFP mode

When the AUTORIFP bit is set to ‘1’, the measured pressure becomes the pressure reference in the REF_P (15h, 16h and 17h) registers as in the case of the AUTOZERO mode, but the output pressure registers are not updated. Therefore, PRESS_OUT (28h, 29h and 2Ah) gives the difference between the measured pressure and the content of the RPDS registers (18h, 19h):

PRESS_OUT = measured pressure – RPDS*256.

After the first conversion, the AUTORIFP bit is automatically set to ‘0’. To return back to normal mode, the RESET_ARP bit in INTERRUPT_CFG (0Bh) register has to be set to ‘1’.

6.1.3 Interrupt events for FIFO triggers

The interrupt events related to pressure acquisition can be used to trigger FIFO dynamic mode transitions. For the FIFO mode Stream-to-FIFO, Bypass-to-Stream and Bypass-to-FIFO, the IA bit in INT_SOURCE register is used as a trigger event to drive the switch from one FIFO mode to the other one.

For example, considering the FIFO in Stream-to-FIFO mode, when the interrupt event is generated, the FIFO switches from Stream mode to FIFO mode.

6.2 Interrupt events related to FIFO status

6.2.1 FIFO interrupts triggered by FIFO status

With the LPS22HB pressure sensor, it is possible to generate interrupts based on the FIFO status. In particular it is possible to generate the following events by properly configuring the CTRL_REG3(12h) register:

- FIFO full condition: F_FSS5 set to ‘1’
- FIFO watermark level reached: F_FTH set to ‘1’
- FIFO overrun: F_OVR set to ‘1’

Once the interrupt events are generated, they are made available to the INT_DRDY pin based on the INT_S bits in CTRL_REG3(12h).

6.3 Interrupt events on the INT_DRDY pin

Interrupt events can be made available to the INT_DRDY pin, acting on the INT_S bits in CTRL_REG3(12h), as shown in the figure below.

Figure 14: Interrupt architecture

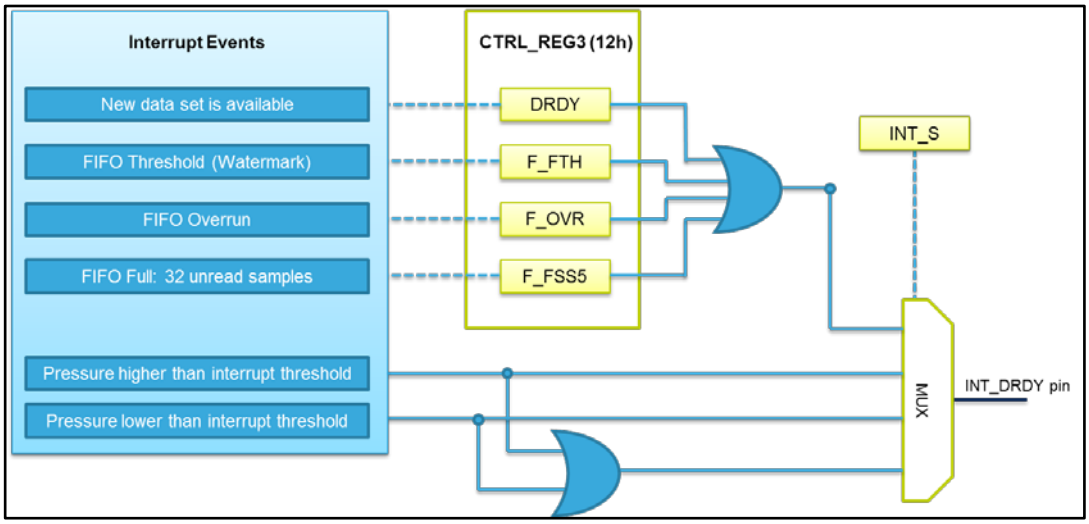


Table 6: INT_DRDY pin configuration

INT_S2	INT_S1	INT_DRDY Pin Configuration
0	0	Data signal. Refer to *Figure 14: "Interrupt architecture"*
0	1	Pressure high (P_high)
1	0	Pressure low (P_low)
1	1	Pressure low OR high

7 Appendix

7.1 FIFO Bypass mode example

Sensor configuration	**BASIC CONFIGURATION** CTRL_REG1 = ‘00111010’ binary = ‘3A’ Hex => ODR = 25 Hz (continuous mode), LPF active with ODR/9, BDU active CTRL_REG2 = ‘00010000’ binary = ‘10’ Hex => FIFO OFF and Multiple reading ON
Read operations	The device provides data in continuous mode without using the FIFO. Reads are performed by reading the following registers: PRESS_OUT_XL (0x28), PRESS_OUT_L (0x29), PRESS_OUT_H (0x2A), TEMP_OUT_L (0x2B) and TEMP_OUT_H (0x2C).
Notes	FIFO_CTRL is ‘00’ Hex per default The FIFO is fully bypassed.

7.2 FIFO mode example

Sensor configuration	**BASIC CONFIGURATION** CTRL_REG1 = ‘00111010’ binary = ‘3A’ Hex => ODR = 25 Hz (continuous mode), LPF active with ODR/9, BDU active CTRL_REG2 = ‘00010000’ binary = ‘10’ Hex => FIFO OFF and Multiple reading ON **FIFO CONFIGURATION** CTRL_REG2 = ‘01011000’ binary = ‘50’ Hex => FIFO is On, multiple reading active (IF_ADD_INC) FIFO_CTRL = ‘00000000’ binary = ‘00’ Hex => FIFO reset for flushing the FIFO FIFO_CTRL = ‘00100000’ binary = ‘20’ Hex => FIFO set to FIFO mode
FIFO reading and restart	FIFO Reading OUT using the registers PRESS_OUT_XL (0x28), PRESS_OUT_L (0x29), PRESS_OUT_H (0x2A), TEMP_OUT_L (0x2B) and TEMP_OUT_H (0x2C). **FIFO RESTART** FIFO_CTRL = ‘00000000’ binary = ‘00’ Hex => FIFO reset for flushing the FIFO FIFO_CTRL = ‘01000000’ binary = ‘40’ Hex => FIFO set to Stream mode CTRL_REG2 = ‘01011000’ binary = ‘50’ Hex => FIFO is On
Notes	FIFO is automatically stopped when full. After reading it starts again to collect pressure and temperature samples

7.3 Stream mode example

Sensor configuration	**BASIC CONFIGURATION** CTRL_REG1 = ‘00111010’ binary = ‘3A’ Hex => ODR = 25 Hz (continuous mode), LPF active with ODR/9, BDU active CTRL_REG2 = ‘00010000’ binary = ‘10’ Hex => FIFO OFF and Multiple reading ON **FIFO CONFIGURATION** CTRL_REG2 = ‘01011000’ binary = ‘50’ Hex => FIFO is On and multiple reading active (IF_ADD_INC) FIFO_CTRL = ‘00000000’ binary = ‘00’ Hex => FIFO reset for flushing out the FIFO FIFO_CTRL = ‘01000000’ binary = ‘40’ Hex => FIFO set to Stream mode
FIFO reading and restart	FIFO Reading OUT using the registers PRESS_OUT_XL (0x28), PRESS_OUT_L (0x29), PRESS_OUT_H (0x2A), TEMP_OUT_L (0x2B) and TEMP_OUT_H (0x2C).
Notes	FIFO doesn’t stop automatically. Data are continuously streamed from the device. The oldest data in the FIFO is discarded and replaced with the newest data.

7.4 Stream-to-FIFO mode example

Sensor configuration	**BASIC CONFIGURATION** CTRL_REG1 = ‘00111010’ binary = ‘3A’ Hex => ODR = 25 Hz (continuous mode), LPF active ODR/9, BDU active CTRL_REG2 = ‘00010000’ binary = ‘10’ Hex => FIFO OFF and Multiple reading ON **INTERRUPT CONFIGURATION** CTRL_REG3 = ‘00000001’ binary = ‘01’ Hex => INT_S[2 :1]=01 Pressure High INTERRUPT_CFG = ‘00001101’ binary = ‘0D’ Hex => DIFF_EN, LIR, PHE THS_P_L = ‘20’ Hex THS_P_H = ‘00’ Hex => Threshold set at 2 hPa INTERRUPT_CFG = ‘00101101’ Hex=‘2D’ => to activate the AUTOZERO **FIFO CONFIGURATION** CTRL_REG2 = ‘01011000’ binary = ‘50’ Hex => FIFO is On FIFO_CTRL = ‘00000000’ binary = ‘00’ Hex => FIFO reset to Bypass mode for flushing out the FIFO FIFO_CTRL = ‘01100000’ binary = ‘60’ Hex => FIFO set to Stream-to-FIFO mode
FIFO readings and restart	After that the interrupt occurs, FIFO reading OUT is performed using the registers PRESS_OUT_XL (0x28), PRESS_OUT_L (0x29), PRESS_OUT_H (0x2A), TEMP_OUT_L (0x2B) and TEMP_OUT_H (0x2C). **FIFO RESTART** INT SOURCE (29) read for resetting the interrupt FIFO_CTRL = ‘00000000’ binary = ‘00’ Hex => FIFO reset for flushing the FIFO FIFO_CTRL = ‘01100000’ binary = ‘E0’ Hex => FIFO set to Bypass-to-FIFO mode CTRL_REG2 = ‘01011000’ binary = ‘50’ Hex => FIFO is On FIFO_CTRL = ‘00000000’ binary = ‘00’ Hex => FIFO reset to Bypass mode for flushing the FIFO FIFO_CTRL = ‘01100000’ binary = ‘60’ Hex => FIFO set to Stream-to-FIFO mode

7.5 Bypass-to-Stream mode example

Sensor configuration	**BASIC CONFIGURATION** CTRL_REG1 = ‘00111010’ binary = ‘3A’ Hex => ODR = 25 Hz (continuous mode), LPF active ODR/9, BDU active CTRL_REG2 = ‘00010000’ binary = ‘10’ Hex => FIFO OFF and Multiple reading **INTERRUPT CONFIGURATION** CTRL_REG3 = ‘00000001’ binary = ‘01’ Hex => INT_S[2 :1]=01 Pressure High INTERRUPT_CFG = ‘00001101’ binary = ‘0D’ Hex => DIFF_EN, LIR, PHE THS_P_L = ‘20’ Hex THS_P_H = ‘00’ Hex => Threshold set at 2 hPa INTERRUPT_CFG = ‘00101101’ Hex=‘2D’ to activate the AUTOZERO **FIFO CONFIGURATION** CTRL_REG2 = ‘01011000’ binary = ‘50’ Hex => FIFO is On FIFO_CTRL = ‘00000000’ binary = ‘00’ Hex => FIFO reset to Bypass mode for flushing the FIFO FIFO_CTRL = ‘01100000’ binary = ‘80’ Hex => FIFO set to Bypass-to-Stream mode
FIFO reading and restart	After that the interrupt occurs, FIFO Reading OUT is performed using the registers PRESS_OUT_XL (0x28), PRESS_OUT_L (0x29), PRESS_OUT_H (0x2A), TEMP_OUT_L (0x2B) and TEMP_OUT_H (0x2C).. **FIFO RESTART** INT SOURCE (29) read for resetting the interrupt FIFO_CTRL = ‘00000000’ binary = ‘00’ Hex => FIFO reset for flushing the FIFO FIFO_CTRL = ‘01100000’ binary = ‘E0’ Hex => FIFO set to Bypass-to-FIFO mode CTRL_REG2 = ‘01011000’ binary = ‘50’ Hex => FIFO is On FIFO_CTRL = ‘00000000’ binary = ‘00’ Hex => FIFO reset to Bypass mode for flushing the FIFO FIFO_CTRL = ‘01100000’ binary = ‘E0’ Hex => FIFO set to Bypass-to-FIFO mode

7.6 Bypass-to-FIFO mode example

Sensor configuration	**BASIC CONFIGURATION** CTRL_REG1 = ‘00111010’ binary = ‘3A’ Hex => ODR = 25 Hz (continuous mode), LPF active ODR/9, BDU active CTRL_REG2 = ‘00010000’ binary = ‘10’ Hex => FIFO OFF and Multiple reading **INTERRUPT CONFIGURATION** CTRL_REG3 = ‘00000001’ binary = ‘01’ Hex => INT_S[2 :1]=01 Pressure High INTERRUPT_CFG = ‘00001101’ binary = ‘0D’ Hex => DIFF_EN, LIR, PHE THS_P_L = ‘20’ Hex THS_P_H = ‘00’ Hex => Threshold set at 2 hPa INTERRUPT_CFG = ‘00101101’ Hex=‘2D’ to activate the AUTOZERO **FIFO CONFIGURATION** CTRL_REG2 = ‘01011000’ binary = ‘50’ Hex => FIFO is On FIFO_CTRL = ‘00000000’ binary = ‘00’ Hex => FIFO reset to Bypass mode for flushing the FIFO FIFO_CTRL = ‘01100000’ binary = ‘E0’ Hex => FIFO set to Bypass-to-FIFO mode
FIFO reading and restart	After that the interrupt occurs, FIFO reading OUT is performed using the registers PRESS_OUT_XL (0x28), PRESS_OUT_L (0x29), PRESS_OUT_H (0x2A), TEMP_OUT_L (0x2B) and TEMP_OUT_H (0x2C). **FIFO RESTART** INT SOURCE (29) read for resetting the interrupt FIFO_CTRL = ‘00000000’ binary = ‘00’ Hex => FIFO reset for flushing the FIFO FIFO_CTRL = ‘01100000’ binary = ‘E0’ Hex => FIFO set to Bypass-to-FIFO mode CTRL_REG2 = ‘01011000’ binary = ‘50’ Hex => FIFO is On FIFO_CTRL = ‘00000000’ binary = ‘00’ Hex => FIFO reset to Bypass mode for flushing the FIFO FIFO_CTRL = ‘01100000’ binary = ‘E0’ Hex => FIFO set to Bypass-to-FIFO mode

7.7 Dynamic Stream mode example

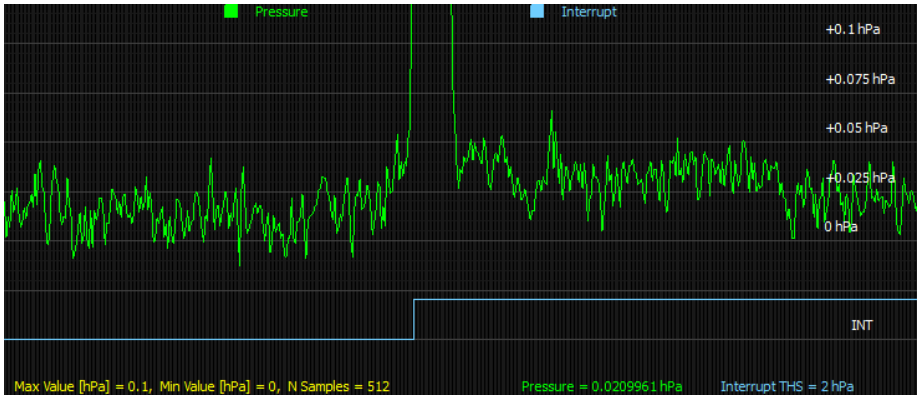
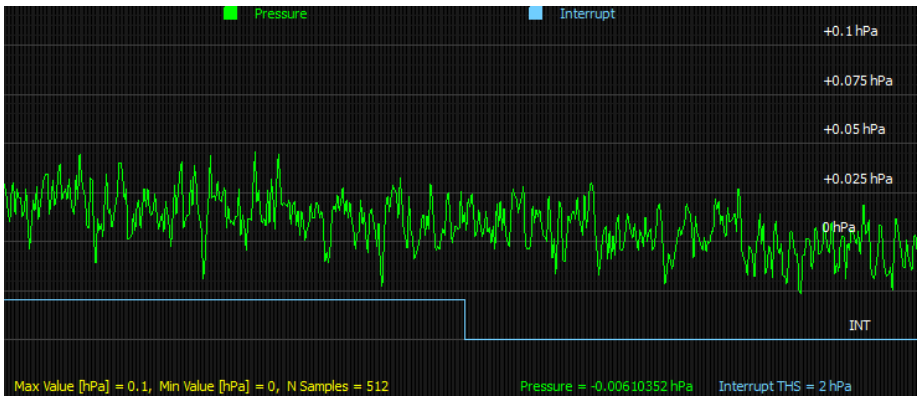
The following example shows how to set the FIFO in Dynamic Stream mode at ODR = 75 Hz and how to get pressure and temperature readings:

BASIC CONFIGURATION
CTRL_REG1 = ‘00111010’ binary = ‘3A’ Hex
=> ODR = 25Hz (continuous mode), LPF active ODR/9, BDU active
CTRL_REG2 = ‘00010000’ binary = ‘10’ Hex
=> FIFO OFF and Multiple reading
INTERRUPTS AND FIFO CONFIG
Write CTRL_REG3(12h) to 0xC8
Set INTERRUPT pin to OpenDrain/Active Low and FIFO OverRun flag
Write FIFO_CTRL(14h) to 0x00
Clear FIFO buffer
Write FIFO_CTRL(14h) to 0xC4
Set FIFO Dynamic Stream mode and Watermark (WTM) to 4
Write CTRL_REG2(11h) to 0x70
Enable FIFO depth to acquire up to WTM+1 samples
Write CTRL_REG1(10h) to 0x52
Set ODR to 75 Hz and Block Data Update active

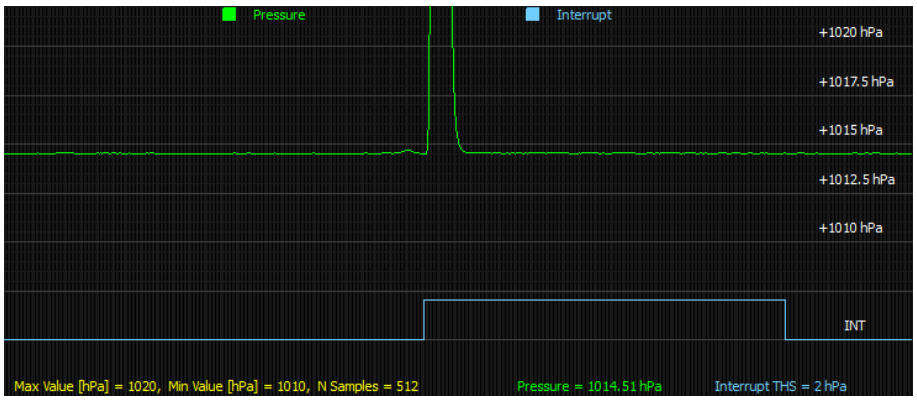
Device reading procedure

Trigger on INT_DRDY pin (pin7) event
For i = 1 to (WTM+1)
Read PressOut (28h-29h-2Ah) and TempOut (2Bh-2Ch)
Read the Output Data Registers 5 times (WTM+1)

7.8 Interrupt: Autozero mode example

Sensor configuration	**BASIC CONFIGURATION** CTRL_REG1 = '00111010' binary = '3A' Hex => ODR = 25 Hz (continuous mode), LPF active ODR/9, BDU active CTRL_REG2 = '00010000' binary = '10' Hex => FIFO OFF and Multiple reading **INTERRUPTS CONFIG** CTRL_REG3 = '00000001' binary = '01' Hex => INT_S[2 :1]=01 Pressure High INTERRUPT_CFG = '00001101' binary = '0D' Hex => DIFF_EN, LIR, PHE THS_P_L = '20' Hex THS_P_H = '00' Hex => Threshold set at 2 hPa INTERRUPT_CFG = '00101101' Hex='2D' to activate the AUTOZERO
Read operations	PRESS_OUT_XL (0x28), PRESS_OUT_L (0x29), PRESS_OUT_H (0x2A), TEMP_OUT_L (0x2B) and TEMP_OUT_H (0x2C).
Interrupt triggered by Differential Pressure > 2 hPa	
Interrupt line reset	INT SOURCE (29) read for resetting the interrupt
	

7.9 Interrupt: AUTORIFP mode example

Sensor configuration	**BASIC CONFIGURATION** CTRL_REG1 = ‘00111010’ binary = ‘3A’ Hex => ODR = 25 Hz (continuous mode), LPF active ODR/9, BDU active CTRL_REG2 = ‘00010000’ binary = ‘10’ Hex => FIFO OFF and Multiple reading **INTERRUPTS CONFIG** CTRL_REG3 = ‘00000001’ binary = ‘01’ Hex => INT_S[2 :1]=01 Pressure High INTERRUPT_CFG = ‘00001101’ binary = ‘0D’ Hex => DIFF_EN, LIR, PHE THS_P_L = ‘20’ Hex THS_P_H = ‘00’ Hex => Threshold set at 2 hPa INTERRUPT_CFG = ‘00001101’ Hex=‘8D’ to activate the AUTORIFP
Read operations	PRESS_OUT_XL (0x28), PRESS_OUT_L (0x29), PRESS_OUT_H (0x2A), TEMP_OUT_L (0x2B) and TEMP_OUT_H (0x2C).
Interrupt triggered by Differential Pressure > 2 hPa	

8 Technical references

Document type	Part number	Title
Datasheet/Data brief	LPS22HB	MEMS pressure sensor: 260-1260 hPa absolute digital output barometer
Evaluation board	STEVAL-MET001V1	LPS22HW adapter board for standard DIL24 socket
	UM0979	STEVAL-MKI109V1 and STEVAL-MKI109V2 - eMotion motherboards for MEMS adapter boards
Evaluation software	UM1049	Unico graphical user interface (GUI)
	UM1064	Software guide for Unico lite
Application note	AN4672	LPS22HB/LPS25HB digital pressure sensors: hardware guidelines for system integration
Technical note	TN1229	How to interpret pressure and temperature readings in the LPS22HB pressure sensor
MCU drivers and Linux/Android drivers for LPS22HB	STSW-MEMS039	Platform-independent device driver for LPS22HB

9 Revision history

Table 7: Document revision history

Date	Version	Changes
16-Mar-2016	1	Initial release
22-Sep-2017	2	Updated *Section 1: "Overview"* and added *Figure 1: "LPS22HB power-on/off sequence"* Minor textual changes

IMPORTANT NOTICE – PLEASE 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. 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 acknowledgement.

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 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.

ST and the ST logo are trademarks of ST. 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.