AN4159
Application note

Hardware and software guidelines for use of the LPS331AP

Introduction

The LPS331AP is a digital MEMs pressure sensor with small package footprint and enhanced digital features.

The official reference specification remains the datasheet.

Several demonstration systems supporting LPS331AP are readily available.

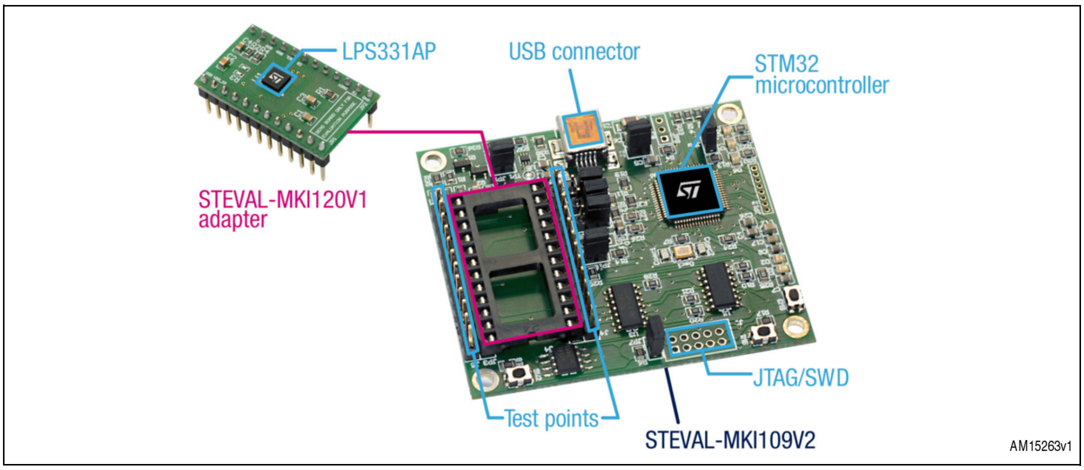
Contents

List of figures

1 Pressure sensor demonstration boards

UNICO/eMotion demo system: [STEVAL-MKI109V2]+[STEVAL-MKI120V1].

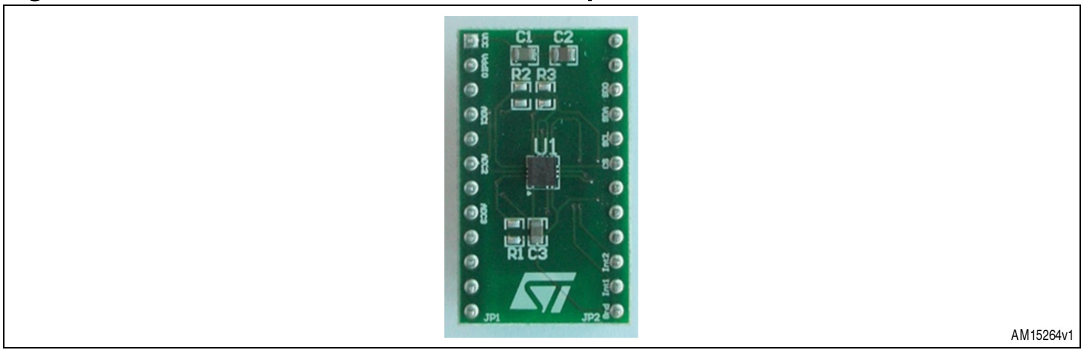
Figure 1. STEVAL-MKI120V1 LPS331AP adapter, STEVAL-MKI109V2 MEMS motherboard



The STEVAL-MKI109V2 is a motherboard designed to provide users with a complete, ready-to-use platform for the demonstration of STMicroelectronics' MEMS products. The board features a DIL24 socket to mount all available adapters for both digital and analog output MEMS devices.

The motherboard includes a high-performance 32-bit microcontroller, which functions as a bridge between the sensor and a PC, on which it is possible to use the downloadable graphical user interface (GUI), or dedicated software routines for customized applications.

Figure 2. STEVAL-MKI120V1 LPS331AP adapter



The STEVAL-MKI120V1 adapter board is designed to facilitate the demonstration of the LPS331AP product. The board offers an effective solution for fast system prototyping and device evaluation directly within the user's own application.

The STEVAL-MKI120V1 can be plugged into a standard DIL 24 socket. The adapter provides the complete LPS331AP pinout and comes ready-to-use with the required decoupling capacitors on the VDD power supply line.

The pinout of the adapter is fully compatible with all other available adapter boards, making it possible to easily switch from one sensor to another during device evaluation without the need for board redesign.

2 Hardware (designing PCB schematics and layout)

2.1 LPS331AP device package, interconnect and polarization

2.1.1 Typical application circuit

Figure 3. Pin mapping

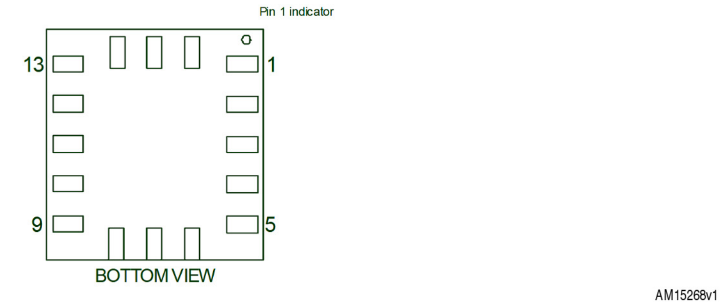


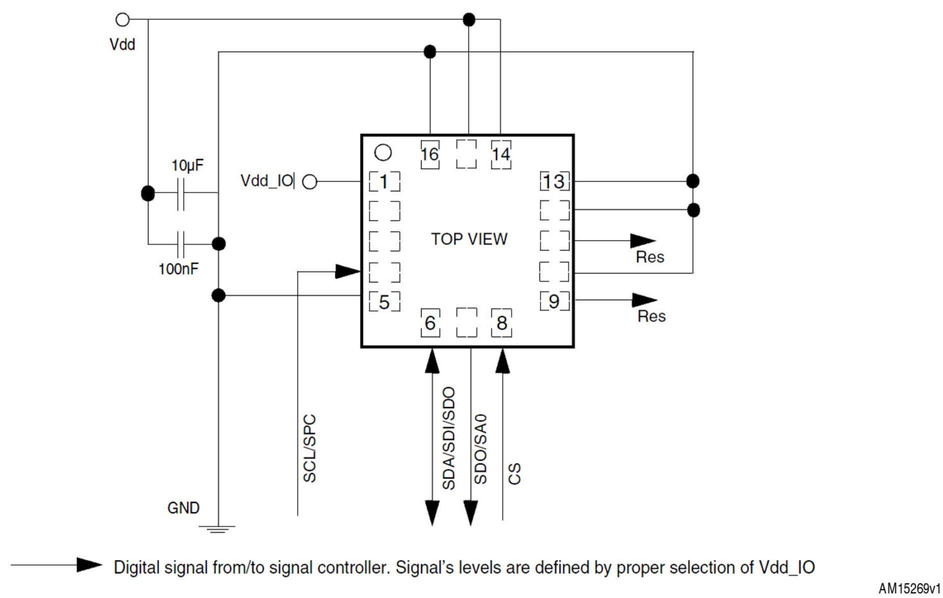
Figure 4. HCLGA-16L 3x3x1 mm

3 Pin mapping

Table 1. LPS331AP pin mapping details

Pin #	Name	Class	What to do
1	VDD_IO	IO supply	Power supply for I/Os (1.8 ~ 3.3 V)
2	NC		Don't connect
3	NC		Don't connect
4	SCL (I2C) SPC (SPI)	In (open-drain) In (open-drain)	External pull-up needed (2.2 kΩ ~ 10 kΩ to VDD_IO) ---
5	GND		Connect to PCB ground
6	SDA (I2C) SDI (SPI 4W) SDIO (SPI 3W)	I2C (open-drain) SPI 4-wire data input SPI 3-wire data bi-dir	External pull-up needed (2.2 kΩ ~ 10 kΩ to VDD_IO)
7	SA0 (I2C) SDO (SPI 4W) NC (SPI 3W)	I2C slave address select SPI 4-wire data output Not connect SPI 3-wire	High (VDD_IO): 0xBA/BB I2C slave address (best) Low (GND_IO): 0xB8/B9 I2C slave address
8	CS	SPI slave select and I2C/SPI mode selection	1 = VDD_IO: I2C mode 0 = GND: SPI mode
9	INT1	Interrupt 1 / data ready	Leave unconnected if unused
10	GND		Connect to PCB ground
11	INT2	Interrupt 2 / data ready	Leave unconnected if unused
12	GND_IO		Connect to PCB ground
13	GND		Connect to PCB ground
14	VDD	Core supply voltage	1.8 ~ 3.3 V very clean supply (put 10 µF and 100 nF decoupling caps near device)
15	VDD		1.8 ~ 3.3 V
16	GND		Connect to PCB ground

Figure 5. LPS331AP electrical connection



Key notes:

- SDA and SCL pull-up resistors should be connected to VDD_IO
- VDD_IO should be same or lower than VDD (use level shifters otherwise)
- If VDD_IO is higher than VDD, high non-destructive current may occur
- If there is choice and interface is I2C, use SA0 = VDDIO slave adress by default.

3.0.1 Pressure sensor PCB layout and solder recommendations

The LPS331AP has an opening on top of the package, sensor performance can be compromised by:

- Mechanical stress coming from PCB board
  - The whole package surface + air should have minimum temperature gradient
  - Avoid placement in long and narrow PCB area, warp free area
- Temperature gradients (non-uniform/rapidly changing temperature around sensor)
- Strong electrical field / light source
- Localized air pressure stability (unwanted fast air pressure variation, fans)
- Dust and water exposure/condensation (gore-tex protection, etc.)

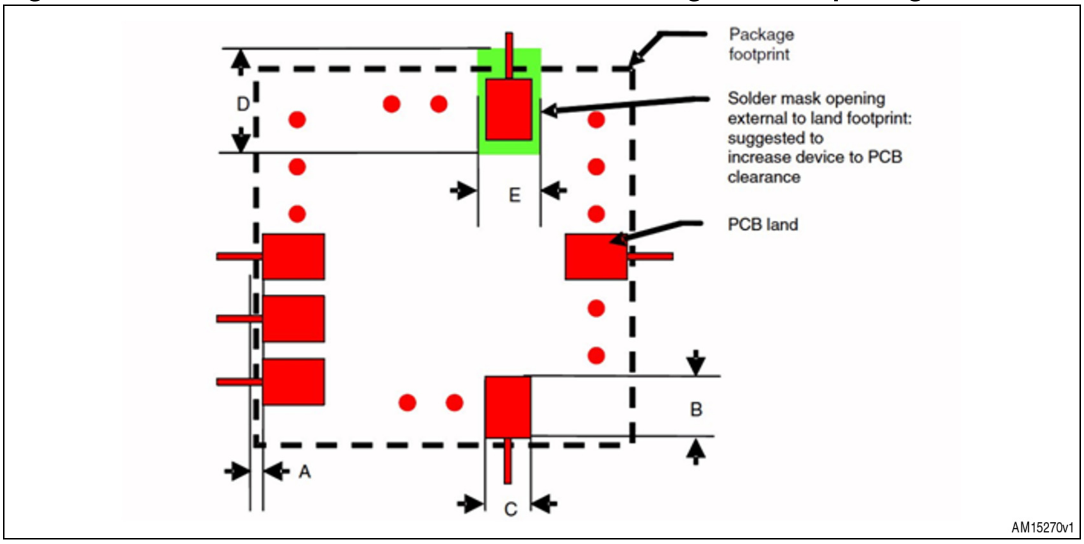
4 PCB design rules

The pressure sensor senses mechanical stress coming from the PCB board, hence it should be kept minimal.

PCB land and solder masking general recommendations are shown below. Refer to the LPS331AP datasheet for pad count, size and pitch.

- It is recommended to open a solder mask external to PCB land
- The area below the sensor (on the same side of the board) must be defined as keepout area. It is strongly recommended not to place any structure in top metal layer underneath the sensor.
- Traces connected to pads should be as much symmetric as possible. Symmetry and balance for pad connection will help component self alignment and will lead to a better control of solder paste reduction after reflow
- For better performances over temperature, it is strongly recommended not to place large insertion components like buttons or shielding boxes at distances less than 2 mm from the sensor
- Pin #1 indicator must be left unconnected to ensure proper device operation

Figure 6. Recommended land and solder mask design for LGA packages



- A = Clearance from PCB land edge to solder mask opening ≥ 0.25 mm to ensure that solder mask is opened externally to device area
- B = PCB land length = LGA solder pad length + 0.1 mm
- C = land width = LGA solder pad width + 0.1 mm
- D = Solder mask opening length = PCB land length + 0.3 mm: design 0.05 mm inside and 0.25 mm outside
- E = Solder mask opening width = PCB land width + 0.1 mm

Stencil design and solder paste application

The soldering paste thickness and pattern are important for the proper pressure sensor mounting process.

- Stainless steel stencils are recommended
- Stencil thickness of 90 - 150 µm (3.5 - 6 mils) is recommended for screen printing
- The final soldering paste thickness should allow proper cleaning of flux residuals and clearance between sensor package and PCB
- Stencil aperture should have a rectangular shape with dimensions up to 25 µm (1 mil) smaller than PCB land
- The openings of the stencil for the signal pads should be between 70 - 80 % of the PCB pad area
- Optionally, for better solder paste release, the aperture walls should be trapezoidal and the corners rounded
- The fine IC leads pitch requires accurate alignement of the stencil to the PCB. The stencil and printed circuit assembly should be aligned to withing 25 µm (1 mil) prior to application of the solder paste

Process consideration

In case of use of no self-cleaning solder paste, proper board washing after soldering must be done to remove any possible source of leakage between pads due to flux residues.

The PCB soldering profile depends on the number, size and placement of components on the board. The soldering profile should be defined by experience more than the pressure sensor soldering profile only.

No solder material reflow on the side of the package is allowed since LGA packages show metal trace out of package side.

Solder heat resistance and environmental specification:

The second level interconnect category on ST ECOPACK® lead-free package is marked on the inner box label, in compliance with JEDEC Standard JESD97. Soldering conditions maximum ratings are also marked on the same label.

LGA packages for pressure sensor are qualified for soldering heat resistance according to JEDEC J-STD-020, in MSL3 condition.

4.1 Power supply and sequencing, fail-safe I/Os

There are 2 voltage supplies, VDD_IO and VDD in LPS331AP.

VDD is the core voltage supply to the internal circuits, power on reset, and sensor.

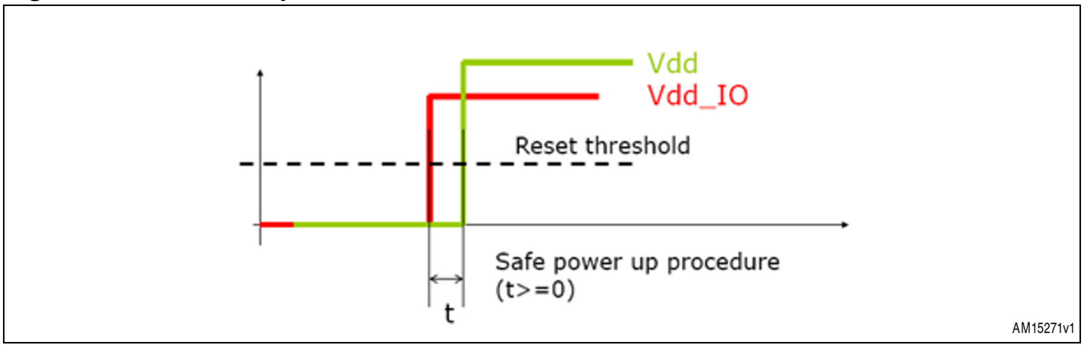
VDD_IO is the supply for the I2C blocks and interface signals.

The operating voltage for both VDD_IO and VDD is from 1.71 V to 3.6 V.

In order to prevent possible leakage in the operational condition, it is necessary to ensure VDD_IO <= VDD.

Power sequence - case #1 (recommended)

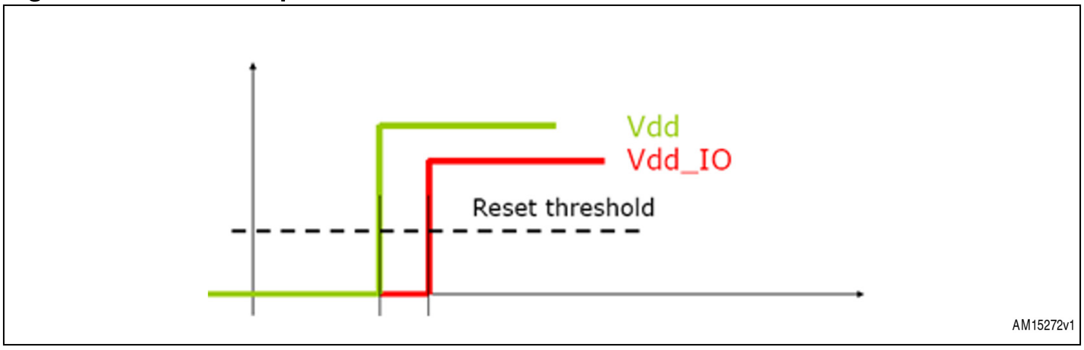
Figure 7. Power sequence - case #1



Power sequence is similar to ST accelerometers. Vdd_IO is always recommended to rise before (or at same time) with Vdd.

Power sequence - case #2

Figure 8. Power sequence - case #2



It is advised that Vdd_IO rise no later than 1 msec after Vdd.

5 Using the device step-by-step, from basic to advanced

5.1 First time bring-up

1. Start by using the same supply for VDD_IO and VDD to check the device functionality.
2. Make sure of the supply impedances and check all the pins voltages in static condition.
3. The device is a simple slave with 1 byte sub-address which bit 7 should always be '1' (bit 7 = 1 enables I2C sub-address multi-byte autoincrement = use it always!)
   Use an oscilloscope to check that the device acknowledge its slave address.
   Use a normal I2C bus speed (<400 kHz)
   Make sure there are no other slaves with same address on the I2C bus lines.
   The maximum I2C speed is driven by the slowest slave on the same I2C bus.
4. Read the Chip ID (sub address 0x8F [=0x0F]) byte: 0xBB should be read

5.1.1 One shot mode measurement sequence

5. Power down the device (clean start)
   – LPS331AP_WriteByte(LPS331AP_CTRL_REG1_ADDR,0x00); // @0x20=0x00
6. Set the pressure sensor to higher-precision
   – LPS331AP_WriteByte(LPS331AP_RES_ADDR,0x7A);// @0x10=0x7A
   – Please note that 0x7A is forbidden in auto-mode (25Hz,25Hz): use 0x79 instead.
7. Turn on the pressure sensor analog front end in single shot mode
   – LPS331AP_WriteByte(LPS331AP_CTRL_REG1_ADDR,0x84); // @0x20=0x84
8. Run one shot measurement (Temperature and Pressure), self clearing bit when done.
   – LPS331AP_WriteByte(LPS331AP_CTRL_REG2_ADDR,0x01); // @0x21=0x01
9. Wait until the measurement is completed: Wait that reading (@0x21=0x00)
10. Read the Temperature measurement (2 bytes to read)
   – LPS331AP_Read((u8*)pu8, LPS331AP_TEMP_OUT_ADDR, 2); // @0x2B~2C
   – Temp_Reg_s16 = ((u16) pu8[1]<<8) | pu8[0]; // make a SIGNED 16 bit variable
   – Temperature_DegC = 42.5 + Temp_Ref_s16 / (120*4); // scale and offset

- Read the Temperature-compensated Pressure measurement
   – LPS331AP_Read((u8*) pu8, LPS331AP_PRESS_OUT_ADDR, 3); // reading autoincremented @0x28/29/2A
   – LPS331AP_pressure_hw_reg_u32 = ((u32)pu8[2]<<16)|((u32)pu8[1]<<8)|pu8[0]; // make a unsigned 32 bit variable
   – LPS331AP_pressure_mb = LPS331AP_pressure_hw_reg_u32 / 4096; // scale

11. Check the temperature and pressure values make sense
   – Reading fixed 760 mb, means the sensing element is damaged.

Example of register measurements and conversion:

P = 0x3FF58D means 1023.347 mb

T = 0xE07C means 42.5 + (-8068/120) = -25.7 °C

100 °C = 0x6BD0, 0 °C = 0xB050

5.1.2 Power optimization and estimation

- Pavg = bit [3..0] of I2C register @0x10 (LPS331AP_RES_ADDR)
- Tavg = bit [6..4] of I2C register @0x10 (LPS331AP_RES_ADDR)

Table 2. Resolution mode

T_RES=bit[6..4]@0x10	Tavg	P_RES=bit[3:0]@0x10	Pavg
0	1	0	1
1	2	1	2
2	4	2	4
3	8	3	8
4	16	4	16
5	32	5	32
6	64	6	64
7	128	7	128
		8	256
		9	384
		A	512

Stand-by: The minimum power consumption is when the device is in PD=0 AND the interrupt pins are not configured to drain current.

Operating: The power consumption depends on HW averaging and AutoRefresh frequency. In one shot mode, the device automatically goes stand by when measurement is completed. The right compromise between consumption, update speed and software filtering has to be tuned application by application.

- Icc = [1 µA/Hz + 48 nA/Hz*Pavg] * ODRP + 32 nA/Hz * Tavg * ODRT
  ODRP= ODR Pressure (Hz)
  ODRT= ODR Temperature (Hz)
  Pavg=nPAve= Pressure Average (1,2,4…512)
  Tavg=nTAve= Temperature Average (1,2,4..128)
- Examples 1:
  ODRT=ODR Temperature=25 Hz
  ODRP=ODR Pressure=25 Hz
  Pavg=Pressure average=512
  Tavg=Temperature Average=64
  Icc=690 µA (Vdd independent at first approximation)

- Examples 2:
  ODRT=ODR Temperature=1 Hz
  ODRP=ODR Pressure=1 Hz
  Pavg=Pressure average=512
  Tavg=Temperature Average=64
  Icc=27.6 µA
- When using one shot mode, the current consumption is equivalent to the current consumption at 1 sample/Hz, during the conversion time. (see conversion time)

6 Software

6.1 Device register list

Table 3. Registers address map

Name	Type	Register address		Default	Comment
		Hex	Binary		
Reserved (do not modify)		00-07 0D - 0E			Reserved
REF_P_XL	R/W	8	1000	0	
REF_P_L	R/W	9	1001	0	
REF_P_H	R/W	0 A	1010	0	
WHO_AM_I	R	0 F	1111	10111011	Dummy register
RES_CONF	R/W	10	10000	11111010	
Reserved (do not modify)		11-1F			Reserved
CTRL_REG1	R/W	20	010 0000	0	
CTRL_REG2	R/W	21	010 0001	0	
CTRL_REG3	R/W	22	010 0010	0	
INT_CFG_REG	R/W	23	100011	0	
INT_SOURCE_REG	R	24	100100	0	
THS_P_LOW_REG	R/W	25	100101	0	
THS_P_HIGH_REG	R/W	26	100110	0	
STATUS_REG	R	27	010 0111	0	
PRESS_POUT_XL_REH	R	28	010 1000	output	
PRESS_OUT_L	R	29	010 1001	output	
PRESS_OUT_H	R	2A	010 1010	output	
TEMP_OUT_L	R	2B	010 1011	output	
TEMP_OUT_H	R	2C	010 1100	output	
Reserved (do not modify)		2D-2F			Reserved

6.2 Hints

- (AVGT,AVGP) = (128,512) (RES=0x7A) is not supported with 25/25 Hz ODR.
  - We recommend to use 0x79 as default value to try in a new system
  - Symptom of wrong configuration can be pressure stuck to 760 mb
- If the I2C line have glitches and slaves are holding SDA low, a simple I2C error recovery would be to send 9 stop bits (which will flush the bus), this is usually done by S/W using the I2C GPIOs resources of the host device. A good practice would be to check that SDA is high prior to generating a START bit.

- The serial bus power consumption can be reduced by reducing the data traffic:
  - Using built-in H/W averaging instead of S/W averaging
  - Reduce the polling rate when waiting for the one shot measurement completion (2.1.1 #9)
  - The INT pin can be used as event to minimize serial bus polling
  - Always set the PD bit to zero before changing the register settings

6.2.1 One shot mode conversion time estimation

- Typical conversion time ≈ 62.5*(Pavg+Tavg) + 1545 µs
  - ex: Tavg = 128; Pavg = 512; Typ. conversation time ≈ 41545 µs (or 24 Hz => this configuration is not compatible with 25 Hz)
  - ex: Tavg = 128; Pavg = 384; Typ. conversation time ≈ 33545 µs
  - The formula is accurate within +/- 3% at room ambiant

6.2.2 Reference S/W to get started with LPS331AP

There is a list of C source files that provide the baseline in initializing, configuring and performing measurements with the pressure sensor.

6.2.3 Pressure to altitude conversion

The simplest and widely used barometer (altitude) formula used in most watches come from the US Standard Atmosphere, for example the 1976 edition.

Example of the source code is shown below:

```
void From_Pressure_mb_To_Altitude_US_Std_Atmosphere_1976_ft(double*
Pressure_mb, double* Altitude_ft) {
   //=(1-(A18/1013.25)^0.190284)*145366.45
   *Altitude_ft = (1-pow(*Pressure_mb/1013.25,0.190284))*145366.45;
}

void From_ft_To_m(double* ft, double* m) {

   //=D18/3.280839895
   *m = *ft/3.280839895;
}
```

Terminology is very delicate for altitude. There is not one altitude, there are many.

For example, the GPS altitude is different than the pressure altitude or density altitude.

Aeroplanes flying between airports are using the ISA altitude which is based on yearly/earth wide sea level pressure average value (as shown above) to avoid plane collision: it is the altimeter reference.

When aeroplanes gets near the airport, local equivalent sea level and airfield temperature are wirelessly shared from airport to plane to compute local instant airfield altitude. This is known as QNH, using METAR data.

6.2.4 Auto mode

Some applications have an enable/measure/disable scheme. In such cases, it can be useful to use the auto mode (CTRL1_REG ODN[1:0]). This way, the device runs measurements at fixed time interval, and the application can get the latest measurement without latency.

6.2.5 SW filtering

If higher precision is required, or if the air pressure/flow is unstable, a S/W filter could be implemented on the sensor measurements.

While a classical moving average can be implemented as a simple solution it has some drawback on the RAM usage and response time.

For more advanced filtering, pressure variation limiter can be implemented: using a fast measuring rate (12 or 25 Hz rate), decide what is the maximum pressure variation between 2 successive measurement. Use this variation limit to dampen the high-frequency noise.

For indoor navigation when sub meter detection is required, a special recursive filter could be used. Tailoring the S/W filter for the sensor characteristic and the application requirement is usually a good move. Sensor fusion is the most advanced filtering which uses all available information and uses them to reduce the positioning noise.

6.2.6 BOOT/SWRESET bit

The boot bit can be used when the device is enabled to reinitialize the volatile registers from internal non-volatile trimming memory. Follow this sequence:

1. Turn on BOOT bit
2. Wait BOOT bit self-clear (with S/W timeout)
3. Wait an additional 5 msec
4. Configure the registers
5. Power up the device for normal use

The SWRESET bit resets all the other registers to their reset values and stops any internal state machines.

6.2.7 Absolute accuracy

The devices are trimmed at final test with a typical absolute accuracy of +/-2mb with no external S/W pressure temperature compensation over the temperature range 0°C - 65°C, after quadratic compensation

Reflow soldering may cause an additional spread of the device population, the spread is PCB construction, assembly and layout specific.

Beware that normal sockets and board warpage can cause absolute accuracy errors.

If very good absolute accuracy is needed, a 1 point calibration in the production line could be implemented. Use a high precision barometer, and store in the application a S/W compensation offset, which could also take into account ageing test results.

LPS331AP devices may have higher absolute accuracy by implementing an extra S/W pressure temperature compensation algorithm, which is called "quadratic pressure compensation in temperature", and is described in this chapter.

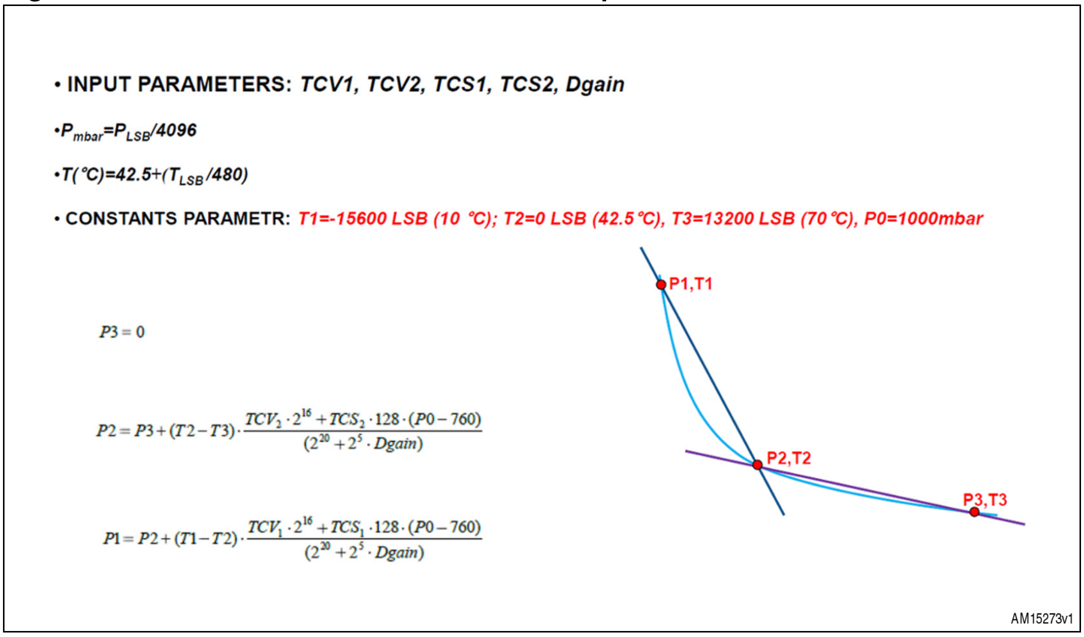
General presentation of the concept

A linear pressure temperature compensation is implemented in the ASIC, using 3 embedded sensor calibration point information.

A second order of accuracy can be achieved (less pressure variation when temperature changes) by implementing a quadratic compensation (second order polynomial approximation) on top of the piecewise linear (PWL) one, by S/W, using the same calibration data built-in the sensor registers.

Here we describe the basic quadratic concept, assuming the T0 = 10 °C:

Figure 9. Overview first to second order compensation



- Input parameters: TCV1, TCV2, TCS1, TCS2, Dgain
- Constants parametr: T1=480*(10-42.5)LSB; T2=0 LSB, T3=480*(70-42.5) LSB; P0=1000mbar.

Step 1: pressure calibration points calculation

Equation 1

$$P3 = 0$$

$$P2 = P3 + (T2 - T3) \cdot \frac{TCV_2 \cdot 2^{16} + TCS_2 \cdot 128 \cdot (P0 - 760)}{(2^{20} + 2^5 \cdot Dgain)}$$

$$P1 = P2 + (T1 - T2) \cdot \frac{TCV_1 \cdot 2^{16} + TCS_1 \cdot 128 \cdot (P0 - 760)}{(2^{20} + 2^5 \cdot Dgain)}$$

Step 2: linear system resolution for Quadratic law determination

Equation 2

$$\begin{bmatrix} a \\ b \\ c \end{bmatrix} = \begin{bmatrix} T1^2 & T1 & 1 \\ T2^2 & T2 & 1 \\ T3^2 & T3 & 1 \end{bmatrix}^{-1} \otimes \begin{bmatrix} P1 \\ P2 \\ P3 \end{bmatrix}$$

Coefficient Parabola Equation: $y=a*T^2+b*T+c$

Step 3: linear system resolution for pwl compensation

Equation 3

$$\begin{bmatrix} e_1 \\ d_1 \end{bmatrix} = \begin{bmatrix} T1 & 1 \\ T2 & 1 \end{bmatrix}^{-1} \otimes \begin{bmatrix} P1 \\ P2 \end{bmatrix} \qquad \begin{bmatrix} e_2 \\ d_2 \end{bmatrix} = \begin{bmatrix} T2 & 1 \\ T3 & 1 \end{bmatrix}^{-1} \otimes \begin{bmatrix} P2 \\ P3 \end{bmatrix}$$

Coefficient straight line equation:

Equation 4

y1=e1*T+d1; y2=e2*T+d2

Since d1=d2=c, the linear system resolution became:

Equation 5

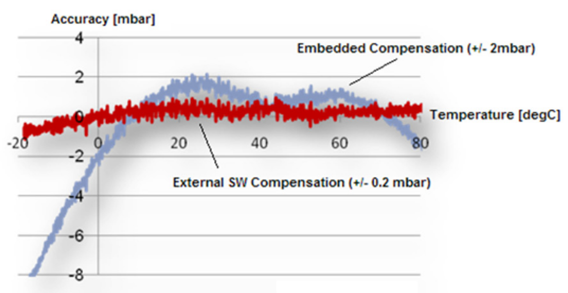
$$e_1 = \frac{P1-c}{T1} \qquad e_2 = \frac{P3-c}{T3}$$

Step 4: output pressure correction

Equation 6

$$Pcomp_{LSB} = \begin{cases} Pout_{LSB} + a \cdot Tout_{LSB}^{\ 2} + (b-e1) \cdot Tout_{LSB} & Tout_{LSB} < 0 \\ Pout_{LSB} + a \cdot Tout_{LSB}^{\ 2} + (b-e2) \cdot Tout_{LSB} & Tout_{LSB} > 0 \end{cases}$$

Figure 10. Comparison first and second order compensation at fixed pressure



All the needed coefficients to compute the quadratic pressure in the microcontroller are available within the sensor registers. A spreadsheet and reference C source code (using floating point or non-floating point arithmetic) are available.

6.2.8 Self test

Devices can be self tested to a certain degree:

- If the I2C slave address is not acknowledged, the power supply, I2C lines and pull-ups are missing or the wrong slave address has been selected
- If the device respond and the pressure is always fixed and out of range (eg 760 mb), check that if u are using the auto reporting of the ODR, make sure that Tave and Pave is in line with the specs and if u are in one shot mode, waiting for the data ready bit flag before to do a read pressure, if all configurations are in line and the pressure is still fixed at 760mb, the sensing element (PZT membrane/bonding wires between sensing element and ASIC) is damaged. Do a registers dump for us to do a diagnostic. This feature is available on FROG. LGA socket to unico adapter is available, especially for failure analysis preliminary report.
- A very big or "negative" pressure could come from bonding wire being cut by mishandling of the device (bonding between sensing element and ASIC)
- An example of C type self-test function is available

6.2.9 Questions and answers

Case 1:

"We are testing the barometer board through 4 wire SPI interface.

"After setting CTRL_REG1 (0x20) to 0xF0, we are polling STATUS_REG (0x27) and PRESS_OUT_XL/L/H, TEMP_OUT_L/H every second.

"The value of STATUS_REG is 0x11 for every read.

"The temperature values are updating correctly; it's about 22.5C after the conversion, and goes up if I put my hand on the board.

"However the pressure value is always constant at 0x2f8000 (PRESS_OUT_H & PRESS_OUT_L & PRESS_OUT_XL), it does not seem to work properly.

- Are there any additional configurations required?
  - RES register was set to @0x10 = 0x7 A while ODR was set to (25 Hz, 25 Hz). Solved by changing RES to 0x79 or ODR to (12.5 Hz, 12.5 Hz).

Case 2:

"The measured pressure altitude by the sensor is different than my portable GPS Barometer"

- Absolute pressure sensor measures the air pressure at the sensing point. The base rule of thumb is that the pressure drops by 1mbar every 8.3 meters.
- The "sea level pressure average" is 1013.25 mb (standard altitude measurement done in most of digital barometer wrist watches)
- The "instant sea level pressure" is weather and location dependent and can vary by more than 5 mbar (compare SFO and SIN airports on the web)
- Also beware that pressure altitude may be different than GPS altitude, which may also be different than density altitude….

Case 3:

The PCB design rule on page 4 says:

The area below the sensor (on the same side of the board) must be defined as keepout area. It is strongly recommended to not place any structure in top metal layer underneath the sensor.

- Does this mean I can't even place ground plane immediately under LPS331AP?
- Can I place ground plane in the PCB middle layer that under LPS331AP?"
  - You can place ground plane in the PCB middle layer under LPS331AP, but not in the plane immediately under the LPS331AP.

Case 4:

We are going to mount the LPS331AP and the LIS3DH on an accessory.

The accessory is powered by battery coins and Vdd fluctuations are possible.

For this device/accessory, we plan to use lithium coin battery (CR2032), and directly supply the voltage to the sensors.

However, depending on the load conditions, the voltage supplied by the coin battery greatly fluctuates (as much as 0.4 V).

Assuming that the voltage supplied to the sensor is within the "normal operating voltage conditions" (even after this 0.4 V drop),

- Do you foresee any issues with the performance and/or characteristics of LIS3DH and LPS331AP?"
  - The max peak current from pressure sensor is 1 mA. A CR2032 is usually 220 mAH, 3.3 V with around 33 Ω intrinsic resistance
  - As the battery ages, the supply may drop down to 2.6 V.
  - As long as the recommended decoupling capacitors are placed near the sensor, our tests with batteries and switched resistive load shows that it should be fine even with fast slew rates.

Case 5:

- Does LPS331AP have internal pull-up resistor on the I2C lines?
  - No
- Absolute accuracy pressure +/1 mbar?
  - When the LGA are calibrated in final test, yes, it is with absolute accurary with +/-1 mb "registers DELTA_PRESS_xx are not mentioned in the mapping table, and are not well defined, it is not clear how user has to manage this register (R/W?)"
  - From application point of view, we suggest not to touch these registers, as a S/W subtract can be done at the host place, no matter where the S/W offset data is saved
  - These are R/W registers, which perform a H/W subtraction with the registers DELTA_PRESS_xx when enabled by bit 1 of CTRL1_REG, before placing the final result in P_OUT register.

- Do we have possible to store the values in a permanent way (after switch off/on)? Any recommendation on the way to realize calibration?"
  - The S/W offset changes with the weather: The sea level pressure is changing regularly and in practice is a data coming from the web, so that at system point of view, keeping this information in
  - The system non-volatile memory maybe the best option.
  - The pressure sensor has some free extra rewritable non-volatile memory bytes. This NVM write option is only for ST-chosen users. It also requires that the VDD is 2.8 V or higher. A sample reference source code can be provided under LUA.

Case 6:

"what is a recommended way to perform a manual soldering of the LGA package"

Apply solder paste (eg. ALPHA® OM-338-T) on the PCB solder pads using syringe (air bubble free), and position the LGA accordingly, then, press it, make sure it is flat on PCB, then heat up in the over using 260C JDEC profile.

First, remember that Hand solder is out of spec. Otherwise, coat your solder pad with tin-lead, then use solder wick to flatten the solder ball, then put solder iron on the lga land marks to deposit solder on LGA pads. Then, position it, and use hot air gun to melt both sides and get eutectic bonding (350C). This is not recommended. Hot air gun must be spread around the LGA dynamically: Keep moving the air gun around the LGA.

Case 7:

What is the consumption of the one shot read? For configuration at .04, 0.025 and .02mbar?

One acquisition per second:

- Accuracy = 0.04, current consumption = 25 µA
- Accuracy = 0.025, current consumption = 27 µA
- Accuracy = 0.025, current consumption = 30 µA

The power consumption is linear with the number of acquisition. It means that if they use 1/3 Hz the above numbers have to divide by 3:

One acquisition per 1/3 second:

  - Accuracy = 0.04, current consumption = 8.3 µA
  - Accuracy = 0.025, current consumption = 9 µA
  - Accuracy = 0.025, current consumption = 10 µA
- Wake-up time from power-down to normal mode?
  - 1 msec
- Wake-up time from switch off to normal mode?
  - 3.5 msec

Case 8:

- What is the offset (variance) of absolute accuracy @ 10 °C (one of ST calibration temperature) or 25 °C temperature?
  - LPS331AP absolute accuracy at 25 °C with one point calibration in ST fab is:
  - Max/Min=+/-1.8 mbar
  - This spread is bigger than what is possible to obtain at the end of the production line because of the non ideality of the final test (noise, temperature etc.).
  - If performed at PCB level after soldering the final accuracy is very high in the range of +/-1mbar totally.
- How ST define the soldering drift SPEC and What is the main factor to impact Soldering drift? And is there any guideline to reduce "Soldering drift"?
  - The soldering drift is a complex process and is not easy to identify the single root cause of the soldering stress.
  - We define the soldering shift as the difference between the accuracy of the pressure sensor before and after the soldering.
  - Soldering temperature profile is one of the major contributor to the soldering shift.
  - A well controlled temperature soldering profile that avoid peak temperature over the max JEDEC spec can reduce the accuracy shift.
  - The guideline are the same already provided in the approval sheet and keeping the temperature profile well controlled and do not exceed the max time at max temperature and if possible in the lower part of the JEDEC specification.
- What is the absolute accuracy SPEC (0 °C ~ 0 °C) after quadratic compensation by AP (application processor)?
  - We confirm a typical accuracy of +/- 1mbar (in this case, it is the relative accuracy, it means without the soldering shift)
- For "Long term drift", How many years ST can guarantee the absolute accuracy with "Long term drift SPEC (+-1mbar)"?
  - For 10 years.

Case 9:

- Will the device lifetime, calibration, accuracy or drift be negatively affected by temperature?
  - We performed life test at low and high temperature without any issue. Low temperature is not an issue for the pressure sensor.
- Could the device be damaged by continual use below -20 °C?
  - No.
- Will there be any power up issues below -20 °C?
  - No
- Is the method of offset calibration created robust enough?
  - Difficult to answer, but other customers successfully implemented this king of second order compensation to improve the accuracy under 0 °C. ?

Case 9:

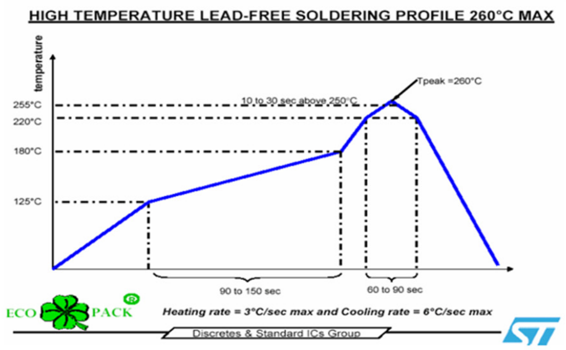
- Does the hole in the LPS331AP need to be covered for a post machine solder wash cycle? Does ST ship the parts with taped coverings?
  - No need, to use any tape to cover the hole on CAP of LPS331AP during reflow. The cleaning to PCB is not general (it's only cleaned when the PCB is contaminated during reflow). The LPS331AP pressure sensor is not covered.

Case 10:

- What are the soldering and processing guidelines for the LPS331AP pressure sensor chip? No wash?
- Ans: Please find reflow profile (Pb-free)IPC JEDEC standard here attached. There is no particular processing guidelines for the pressure sensor. Key thing:

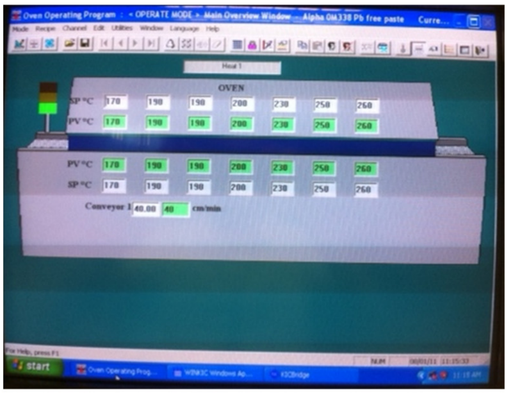
1. Place the pressure sensor at the edge of pcb where the pcb warpage is minimal.
2. There should be no cleaning process perform on the pressure sensors

Figure 11. High temperature lead-free soldering profile 260 °C max



AM15275v1

Figure 12. Example of solder machine profile



AM15276v1

7 Revision history

Table 4. Document revision history

Date	Revision	Changes
24-Sep-2012	1	Initial release.