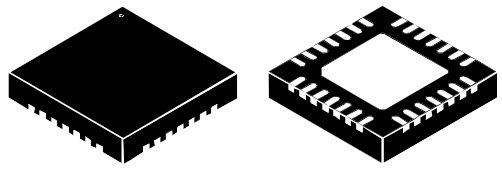
STAMP0

Microphone array processor with TDM output and recombination for wide dynamic range microphones

Datasheet - production data



VFQFPN 28L 5 x 5 x 1 mm

Features

- Up to 4 dual-channel PDM inputs supporting:
  - up to 8 single-membrane microphones
  - up to 4 dual-membrane microphones
- Embedded recombination for dual-membrane digital microphones
- I²C interface
- 3.3 V single supply operation
- Single serial slave TDM interface
- Supported PCM sample frequency 44.1 kHz and 48 kHz
- VFQFPN package
  - SMD-compliant
  - ECOPACK®, RoHS and “Green” compliant

Applications

- Microphone array applications
- Beam forming
- Audio zooming
- Speech recognition
- Sound source localization

Description

STAMP0 is a microphone processor designed to interface a plurality of digital microphones (PDM inputs) to a generic MCU or host controller through a serial TDM connection (PCM output).

STAMP0 has 4 PDM input lines to support connection of up to 8 single-membrane digital microphones or up to 4 dual-membrane microphones. Recombination of the dual-membrane microphone is embedded in the device.

An internal PLL provides stable clock references for the embedded processing as well as for the external microphone array.

An I²C interface allows setting the internal registers for the control of the device depending on the application requirements.

STAMP0 is housed in a small VFQFPN 28-lead 5 x 5 x 1 mm package.

Table 1: Device summary

Part number	Temperature range (°C)	Package	Packing
STAMP0	-40 to +125	VFQFPN 28L 5 x 5 x 1 mm	Tray
STAMP0TR	-40 to +125	VFQFPN 28L 5 x 5 x 1 mm	Tape and reel

Contents

List of tables

List of figures

1 Pin description

Figure 1: Pin connections (top view)

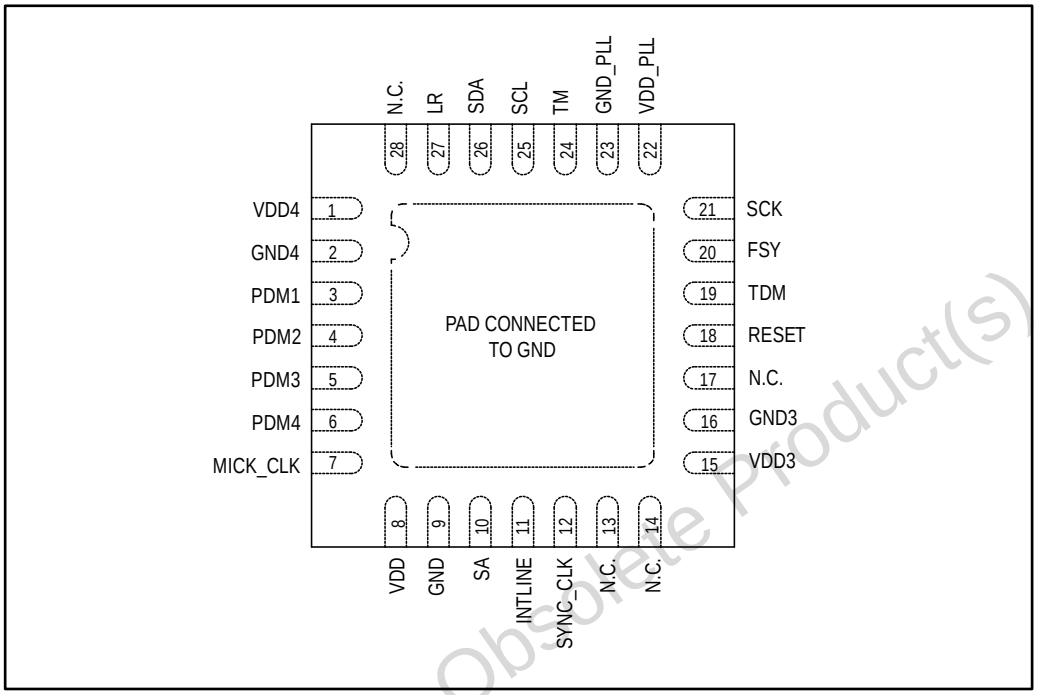


Table 2: Pin description

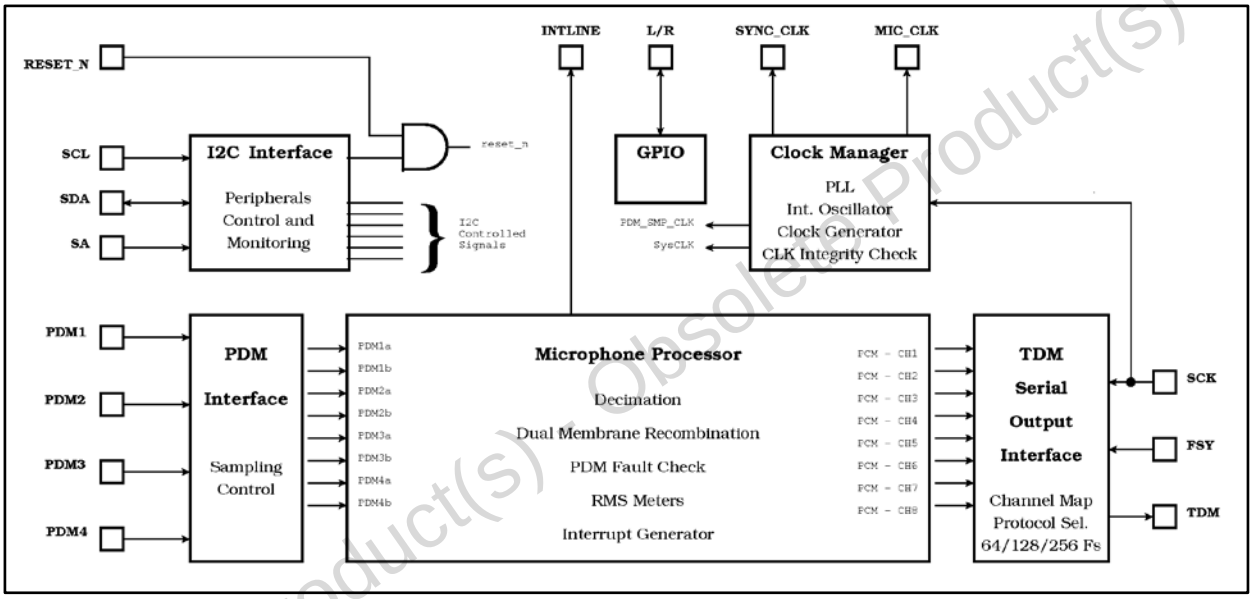
Pin #	Type	Pin name	Description
1	Supply	VDD4	Core and IOs power supply
2	Ground	GND4	Core and IOs ground
3	Digital input	PDM1	PDM input channels 1/2
4	Digital input	PDM2	PDM input channels 3/4
5	Digital input	PDM3	PDM input channels 5/6
6	Digital input	PDM4	PDM input channels 7/8
7	Digital output	MIC_CLK	Microphone clock
8	Supply	VDD	Core and IOs power supply
9	Ground	GND	Core and IOs ground
10	Digital input	SA	I²C device address select
11	Digital output	INTLINE	Interrupt line
12	Digital output	SYNC_CLK	Clock output
15	Supply	VDD3	Core and IOs power supply
16	Ground	GND3	Core and IOs ground
18	Digital input	RESET	Global asynchronous reset active-low
19	Digital output	TDM	TDM serial data out
20	Digital input	FSY	TDM frame sync
21	Digital input	SCK	TDM bit clock
22	Supply	VDD_PLL	PLL power supply

Pin #	Type	Pin name	Description
23	Ground	GND_PLL	PLL_GND
24	Digital input	TM	Test mode
25	Digital I/O	SCL	I²C serial clock
26	Digital I/O	SDA	I²C serial data
27	Digital I/O	LR	Left / Right mic control
13,14,17,28		N.C.	Not Connected

2 Description and block diagram

STAMP0 is a microphone processor designed to interface digital microphones with PDM interface, supporting a slave serial TDM output interface. STAMP0 has 4 PDM input lines to support connections of up to 8 single-membrane digital microphones or up to 4 dual-membrane microphones. Recombination of the dual-membrane microphone is embedded in the device. An internal PLL provides stable clock references for the embedded processing as well as for the external microphone arrays. An I²C interface allows setting the internal registers for the control of the device depending on the application requirements. STAMP0 is housed in a small VFQFPN 28-lead 5x5x1 mm package. The STAMP0 internal architecture is illustrated in *Figure 2: "Block diagram"*.

Figure 2: Block diagram



3 Electrical specifications

3.1 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 3: Absolute maximum ratings

Symbol	Parameter		Maximum value	Unit
VDD	Supply voltage		-0.3 to 4	V
VDD_PLL	PLL power supply		-0.3 to 4	V
T_{STG}	Storage temperature range		-40 to 150	°C
Tj	Junction temperature		-40 to 150	°C
ESD	Electrostatic discharge protection	HBM	4	kV
		CDM	1.5	kV
		MM	100	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.

3.2 Recommended operating conditions

Table 4: Recommended operating conditions

Symbol	Parameter	Min.	Typ.	Max.	Unit
VDD	Digital core and IOs power supply	3.0	3.3	3.6	V
VDD_PLL	PLL power supply	3.0	3.3	3.6	V
TA	Operating ambient temperature	-40	25	125	°C
Rth ja	Thermal resistance junction-to-ambient mounted on PCB		50		°C/W

Note: All ground connections must always be within 0.3 V of each other.

3.3 Electrical characteristics

The values listed in the table below are specified for VDD = 3.3 V, VDD_PLL = 3.3 V and T_{amb} = 25 °C, unless otherwise stated.

Table 5: Electrical specifications

Symbol	Parameter	Test condition	Min.	Typ. (1)	Max.	Unit
VDD	Supply voltage		3.0	3.3	3.6	V
VDD_PLL	PLL supply voltage		3.0	3.3	3.6	V
IDD	Operating current clk = mckl = 100 MHz	VDD = 3.6 V		70		mA
		VDD_PLL= 3.6 V		20		mA
IDD_STBY	Standby current	VDD = 3.6 V		2.5		mA
		VDD_PLL = 3.6 V		450		µA
Digital IO characteristics						
VIH	High-level input voltage		2			V
VIL	Low-level input voltage				0.8	V
VOH	High-level output voltage		VDD-0.15			V
VOL	Low-level output voltage				0.15	V
VHYST	Schmitt trigger hysteresis		400			mV
RDW	Pull-down resistance (@Vi = VDD)		32	50	120	kW
IIH/IIL	Input pad leakage		-2	0.02	2	µA
PLL characteristics						
CLK_IN	Input clock frequency range		2.048		49.152	MHz
D_IN	Input clock duty cycle		40		60	%
RT_IN	Clock in rise time				0.2	ns
FT_IN	Clock in fall time				0.2	ns
INF_IN	Phase comparator freq. range	FRAC CTRL=0	2.048		16.384	MHz
		FRAC CTRL=1	2.048		12.288	MHz
FVCO	PLL output frequency		65.536		98.304	MHz
	Free-running (CLCKIN=0)		10		32	MHz
PKVCO	Peal overshoot in VCO freq.				10	%
DVCO	FVCO duty cycle		35		65	%
LT	Lock time				200	µs
PLL_BW	PLL bandwidth	5 < NDIV < 31	2040/NDIV		4705/NDIV	kHz
		32 < NDIV < 48	77		149	kHz
JPK	Jitter peaking at BW				3	dB

Notes:

(1)Typical specifications are not guaranteed.

4 Architecture

In the following sections the STAMP0 internal functional blocks depicted in *Figure 2: "Block diagram"* are described in detail.

4.1 PDM sampling interface

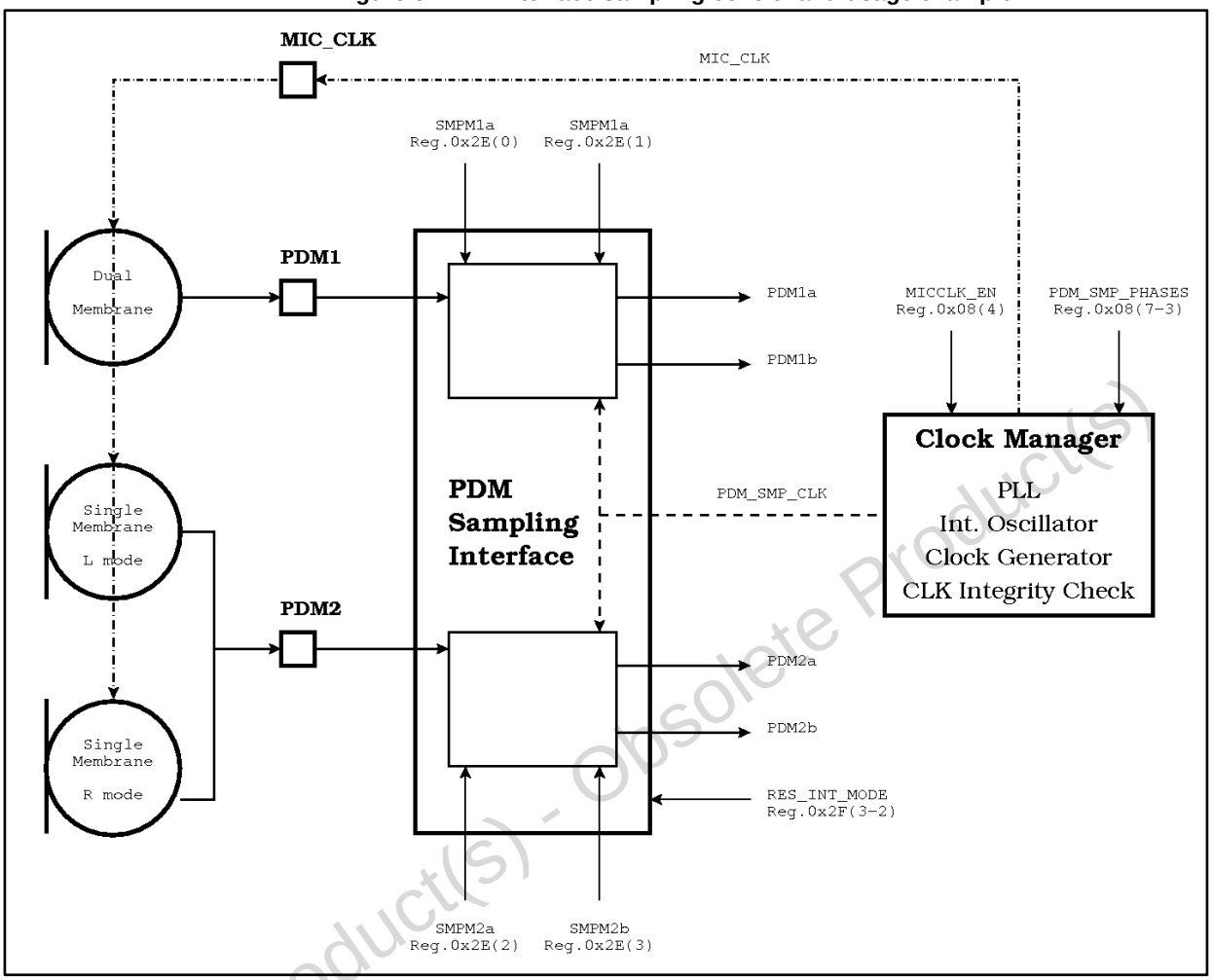
The STAMP0 has four PDM data interface lines to permit up to:

- 8 single-membrane digital microphones (with shared data lines)
- 4 dual-membrane digital microphones (such as MP34DTW01) with the internal microphone recombination features enabled
- A combination of single and dual-membrane microphones

The device will accept 64 Fs serial data in, where Fs is the audio bandwidth used in the processing section (for example 2.8224 MHz in case of 44.1 kHz or 3.072 MHz in case of 48 kHz).

The four PDM inputs can be sampled either at the rising or at the falling edge of the PDM sampling interface clock (PDM_SMP_CLK). As depicted in *Figure 3: "PDM interface sampling control and usage example"*, for each single PDM line, the desired edge can be selected using the SMPMxy I²C register (**Reg. 0x2E, bits 7:0**).

Figure 3: PDM interface sampling control and usage example



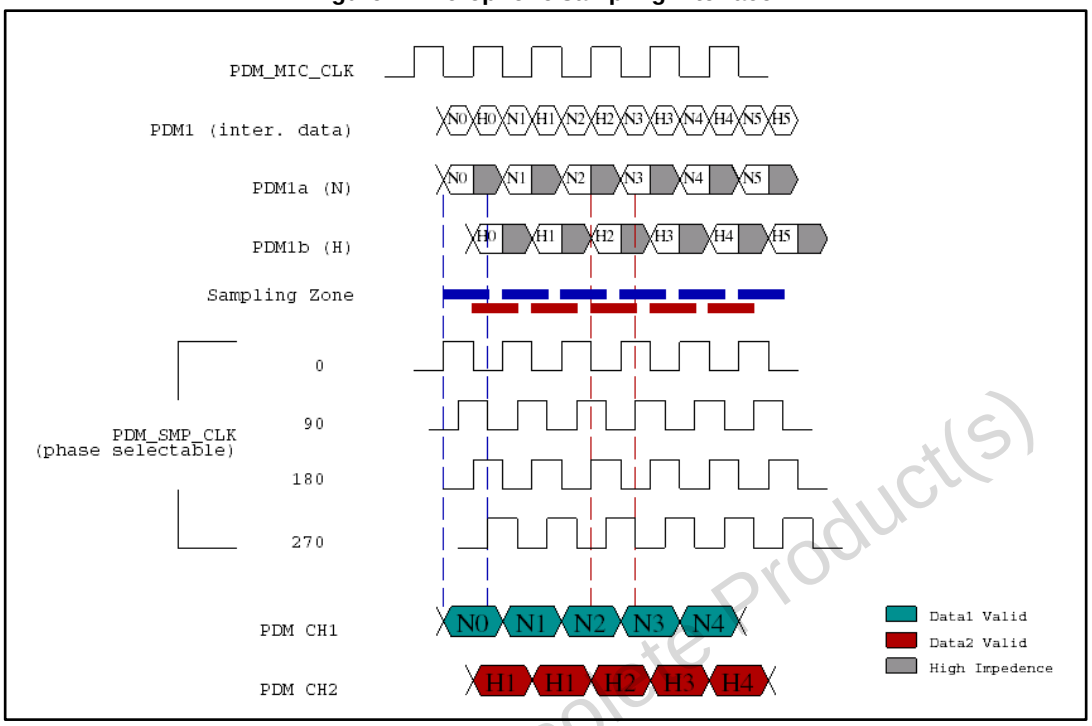
The PDM_SMP_CLK is generated, together with the microphone clock (PDM_MIC_CLK), inside the Clock Manager block (see *Section 5.8: "Clock manager"* for more details) and, they have the same frequency (64xFs). However the phase of the PDM_SMP_CLK is selectable employing the I²C register (**Reg. 0x08bits: B7, B6, B5),** and the supported phase shift values are given in *Table 6: "PDM_SMP_CLK phase shift values"*.

PDM-SMP_CLK = PDM_MIC_CLK + phase shift

Table 6: PDM_SMP_CLK phase shift values

Symbol	Parameter
Phase shift	000 : 0 degrees 001 : 45 degrees 010 : 90 degrees 011 : 135 degrees 100 : 180 degrees 101 : 225 degrees 110 : 270 degrees 111 : 315 degrees

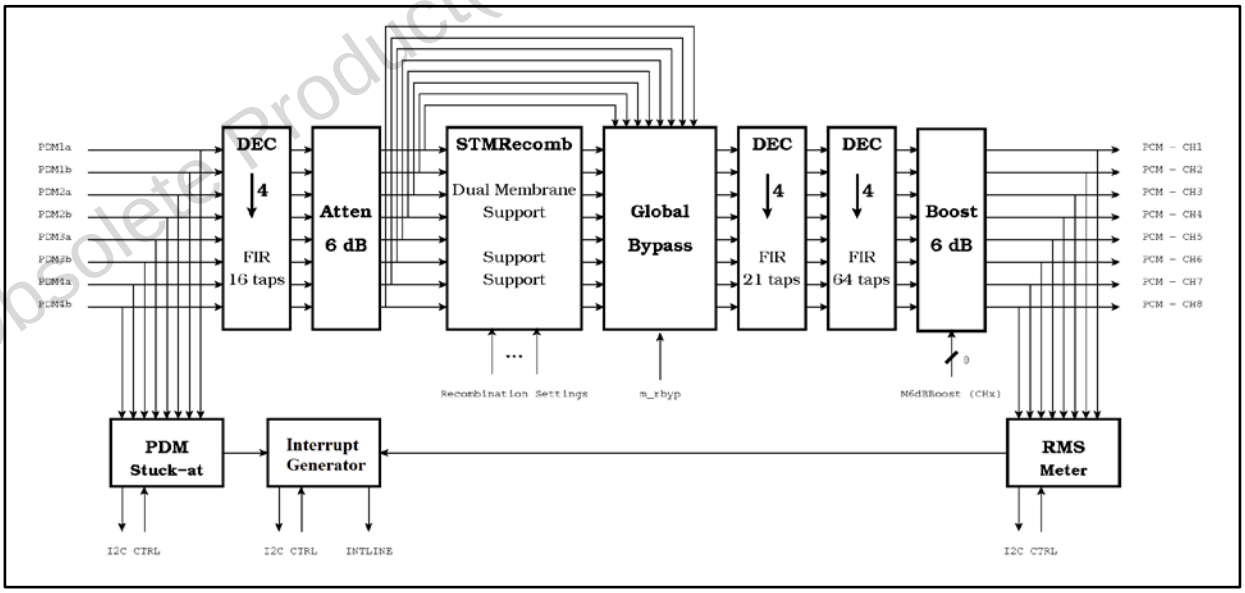
Figure 4: Microphone sampling interface



4.2 Microphone processor

The digital microphone processor is a fixed-point computational engine and it is used to implement all firmware functions.

Figure 5: Microphone processor internal blocks



The following sections describe the microphone processor internal blocks:

- Decimation filters
- Microphone recombination
- RMS level meter
- Interrupt generation (in case of PDM faults or when RMS level is over threshold)

4.2.1 Decimation filters

The STAMP0 provides a decimation section that is used to convert the sample rate from 64 x Fs to Fs.

The decimation is performed using three FIR filters (STAGE1, STAGE2 and STAGE3). Each filtering stage has a decimation factor of 4, consequently both the number of TAPS and coefficient values are different.

The total FIR frequency response (with respect to PDM frequency) is shown in *Figure 6: "Downsampling FIR frequency response (overall)"* and the characteristics are listed in *Table 7: "Downsampling section overall characteristics"*.

Figure 6: Downsampling FIR frequency response (overall)

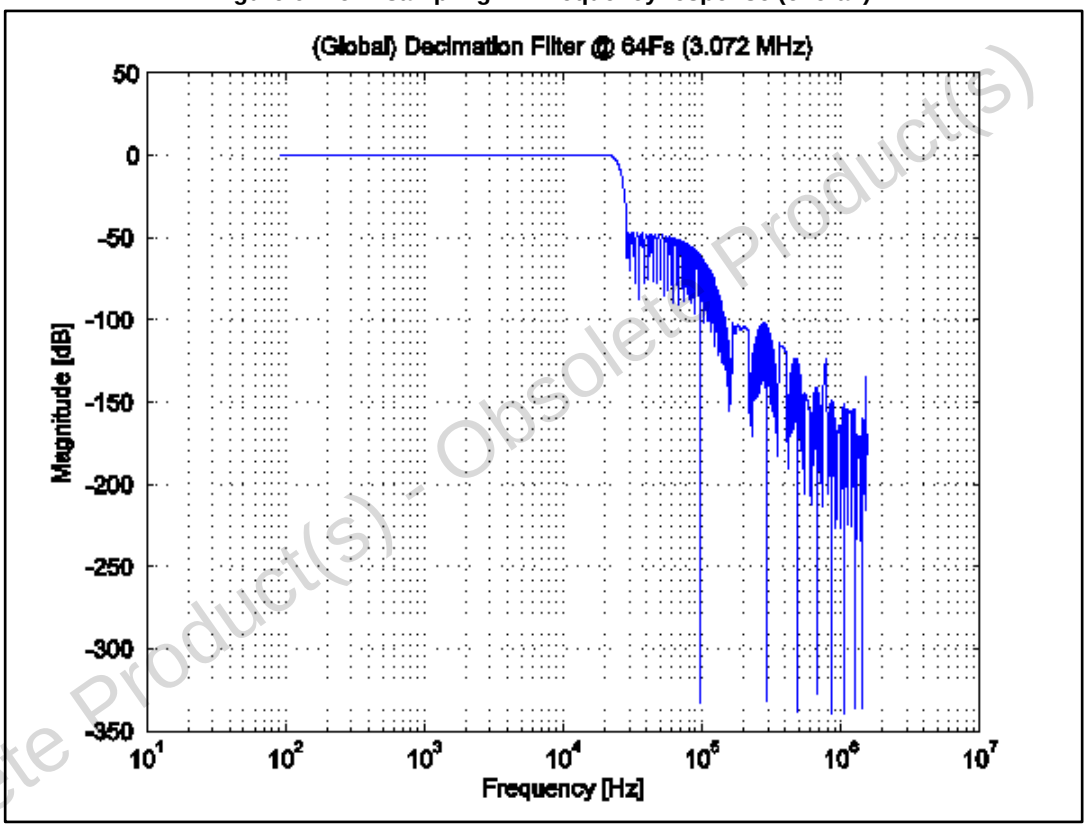


Table 7: Downsampling section overall characteristics

Symbol	Parameter	Min.	Typ.	Max.	Unit
PBW	Pass band relative to PDM sampling frequency		0.00703125		
SBW	Stop band relative to PDM sampling frequency		0.078125		
OOBA	Out of band attenuation	100			dB
IBR	In band ripple			0.1	dB
LT	Total latency			562	PDM samples

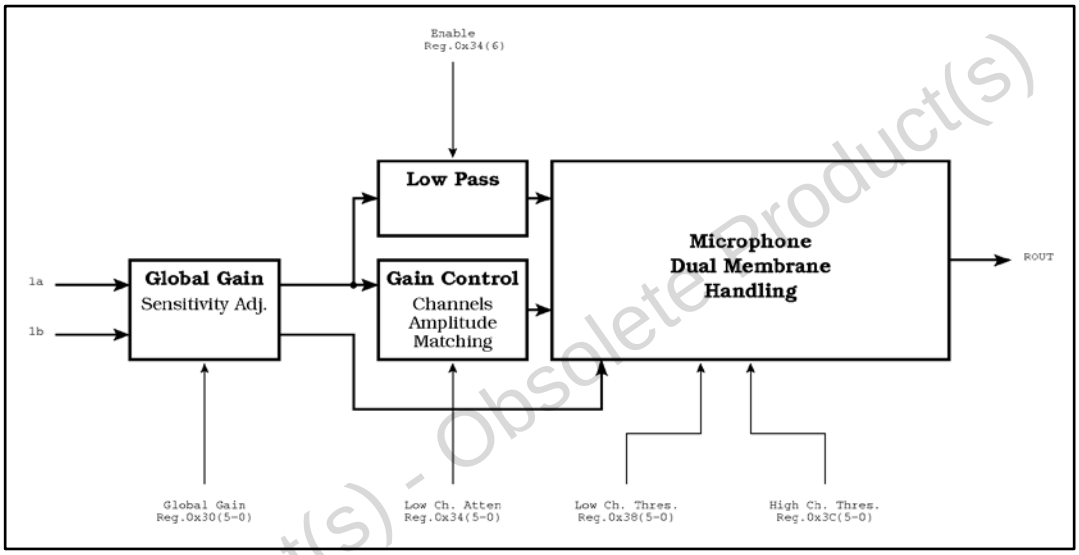
4.3 ST microphone recombination (STMRecomb)

The STMRecomb block provides support for dual-membrane microphones. It handles the two PDM bit-streams in order to obtain a single data channel with the best tradeoff between them; low distortion for High Sound Pressure Level Signals (HSPL) and high signal-to-noise ratio (SNR) for low SPL signals.

STAMP0 integrates four distinct and independent STMRecomb engines (one for every PDM input channel).

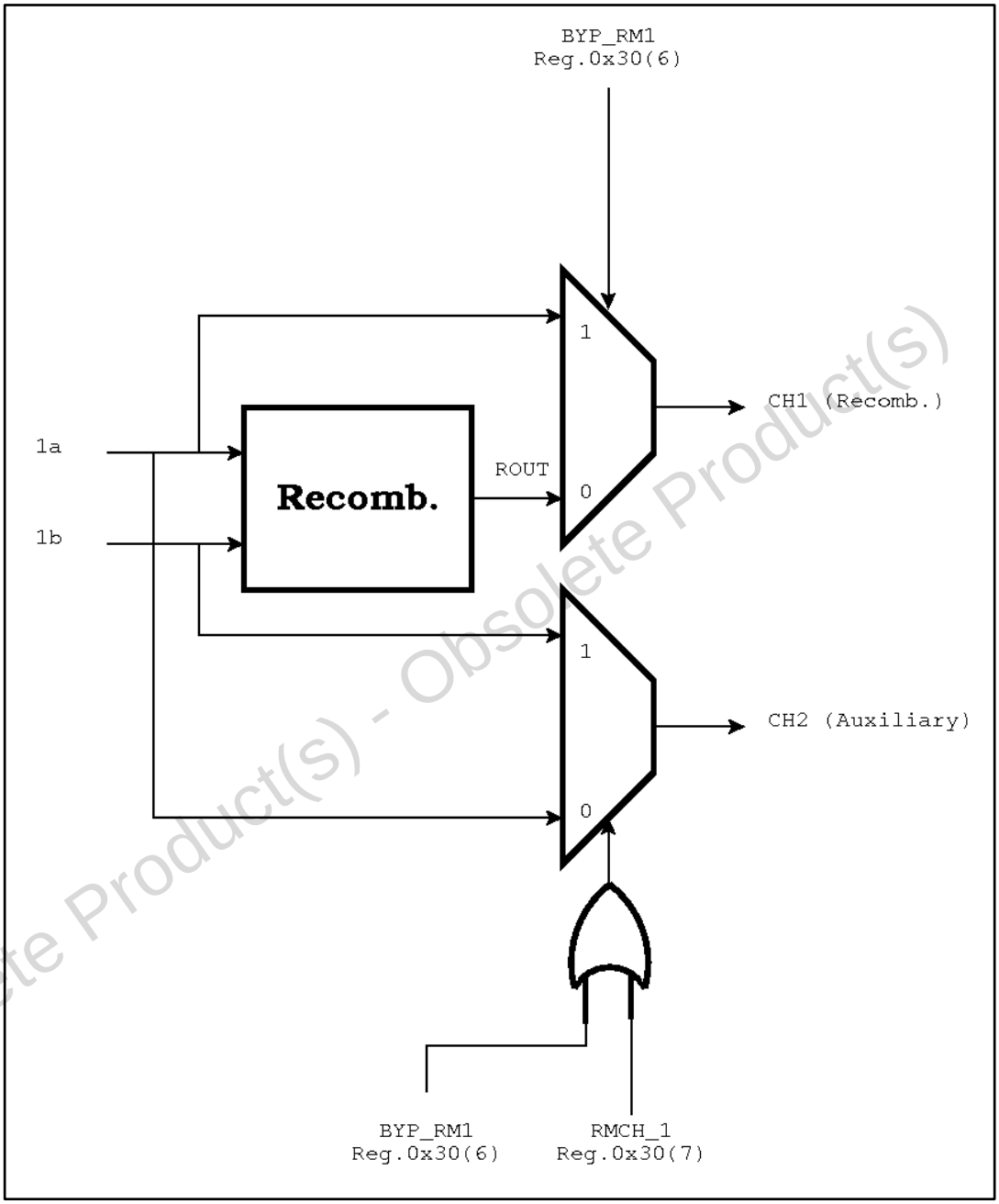
It can be configured in order to support microphones with several differences in sensitivity between two membranes.

Figure 7: STMRecomb block



4.3.1 Recombination bypass functionality

Figure 8: Recombination bypass



4.3.2 PDM interface channel mapping

The internal channels PDMxa, PDMxb must be mapped properly, otherwise the recombination algorithm will not work and the output will lead to unexpected results.

When the recombination function is not active, the two channels will be mapped to the serial output interface as they are.

When the recombination function is active:

- ODD channels (CH1, CH3, and CH5) will contain the recombination signal.
- EVEN channels (CH2, CH4 and CH6) will contain the signal coming from one membrane (user selectable)

4.3.3 Interrupts and masking

In the STAMP0 device four types of interrupt events are generated:

- PLL unlocked
- FSY and SCK integrity failure (FSY/SCK differ from 64, 128, 256)
- Upper limit (UL) reached at the output of each channel
- Stuck at PDM

Each interrupt event pulls the INTLINE pin low unless the interrupt event is masked.

The upper limit and stuck PDM events can be masked using dedicated bits in the I²C registers (Reg. 0x28, Reg. 0x29 and Reg. 0x2A). However the unlocked PLL and FSY/SCK events cannot be masked.

Whenever an interrupt event occurs, the respective interrupt flag (I²C bit in the register Reg. 0x24, Reg. 0x25, Reg. 0x26 and Reg. 0x27) is set to 1 and remains at 1 (even after the interrupt condition ends), until it is cleared (overwritten with 0) via I²C. If an interrupt flag is cleared while the interrupt condition persists, it remains at (or reverts to) 1, and the INTLINE remains low unless the respective interrupt event is masked.

Masking an interrupt event does not affect the interrupt flags register, but masking only affects the INTLINE pin.

4.3.4 Microphone failure detection

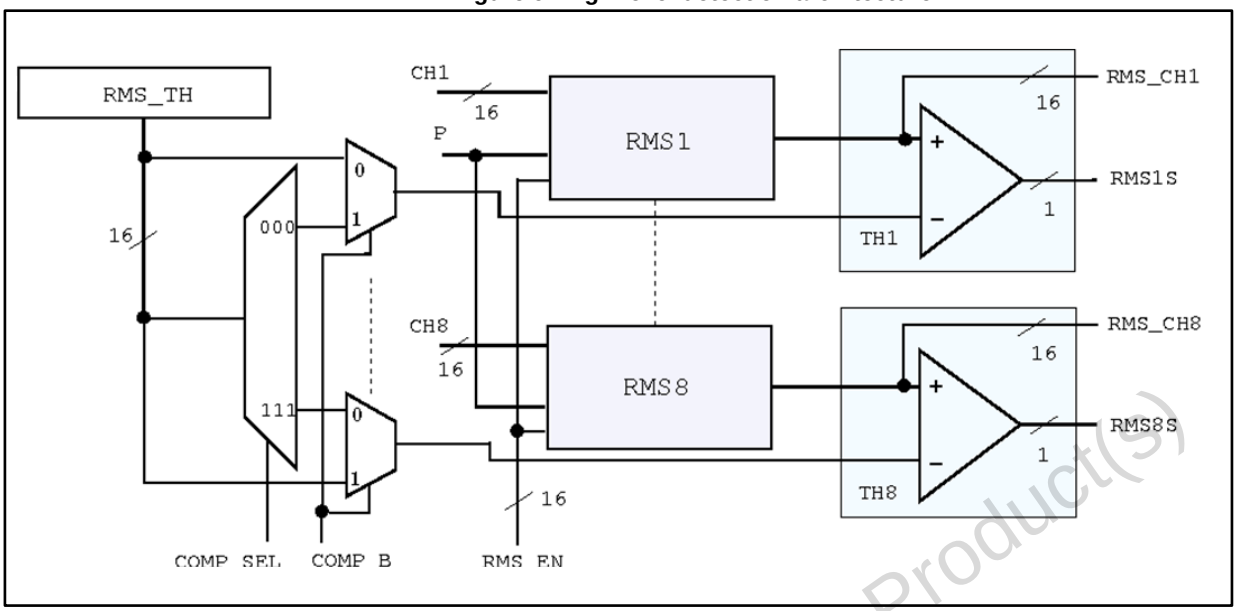
PDM stuck

In order to detect if a PDM is stuck at high/low, a fixed number of PDM samples defined as Stuck At Window (SAW) are monitored. The size of a SAW is fixed to Fs. If all the PDM values in N consecutive SAWs (where N= [1:15] can be configured through the I²C Reg. 0x23 bit B3, B2, B1 and B0) are equal to 1/0, a microphone failure is detected and an interrupt is generated.

Channel level detection (RMS)

In order to detect a high level limit at the output of each channel, the architecture depicted in *Figure 9: "High-level detection architecture"* has been implemented in the STAMP0 device.

Figure 9: High-level detection architecture



In the architecture illustrated in *Figure 9: "High-level detection architecture"* we have:

- 8 RMS meters detailed in *Section 5.3.5: "Averaging based RMS calculation"*
  - The RMS calculation can be enabled using the dedicated I²C register (Reg. 0x11 and Reg. 0x12).
- 8 digital comparators to detect a high level on the related channel. The threshold value for each channel can be programmed through the dedicated I²C register RMS_TH (Reg. 0x0F and 0x10). Moreover observing *Figure 9: "High-level detection architecture"* it is possible to see that two threshold programming modalities are supported:
  - **Single threshold programming**: using the COMP_SEL I²C register (Reg. 0x0E, bit B6, B5 and B4) it is possible to select and consequently to program the desired comparator threshold. In this case one single threshold can be programmed every time.
  - **Burst threshold programming**: using the COMP_B I²C register (Reg. 0x0E, bit B7) it is possible to program the same threshold at the input of each comparator.

When a channel level exceeds the high-level limit programmed through the I²C, the corresponding bit in the I²C register RMSxS (**Reg. 0x24**) is set to one and an interrupt is generated.

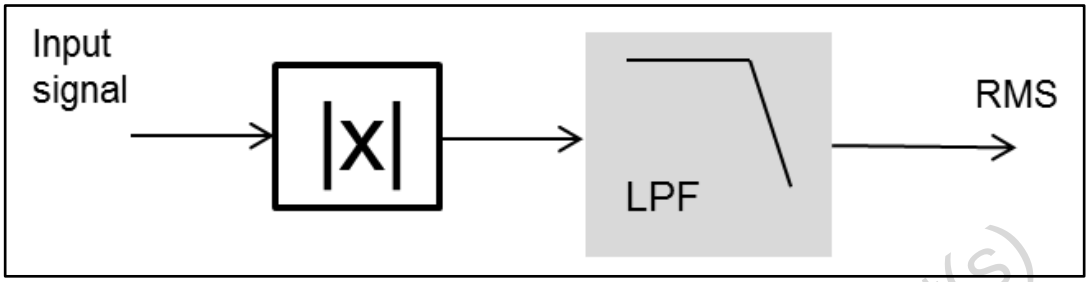
Furthermore the RMS value of each channel can be read back, reading the related I²C registers RMS_CHx (from **Reg. 0x13 to Reg. 0x22**), and its value in dB is given by the following equation:

$$\boldsymbol{RMS\_CHx_{dBFs}} = 20 \times \mathbf{log}_{10} \frac{\boldsymbol{RMS_{CHx}} \times 2^{10-p}}{2^{15} \times 0.635}$$

4.3.5 Averaging based RMS calculation

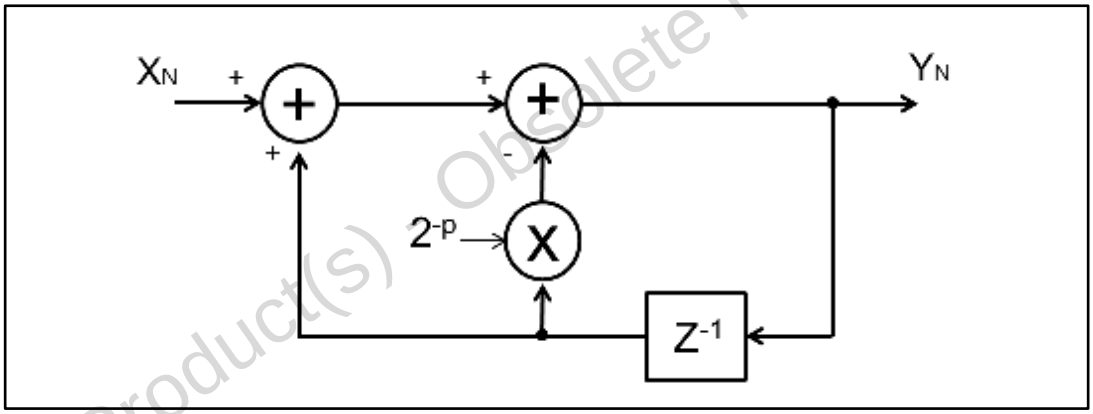
Figure 10: "RMS calculation data path" shows the architecture implementing the RMS block. In particular the absolute value of the two's complement input is taken first to reduce the number of bits involved, and finally the signal is low-pass filtered.

Figure 10: RMS calculation data path



The low-pass filtering solution is implemented with a first-order infinite impulse response filter (IIR filter). The architecture of this filter is depicted in *Figure 11: "IIR filter"*, the Z-transfer filter function is described in *Equation 2* below.

Figure 11: IIR filter



Equation 2

$$\frac{Y_N}{X_N} = \frac{1}{1-(1-2^{-p})Z^{-1}}$$

where 2^{-p} is the actual operand of the filter

In the STAMP0 device the P value can be programmed through the I²C register (Reg. 0x23 bits B7, B6, B5 and B4) and the default value is set to zero.

4.4 Reset

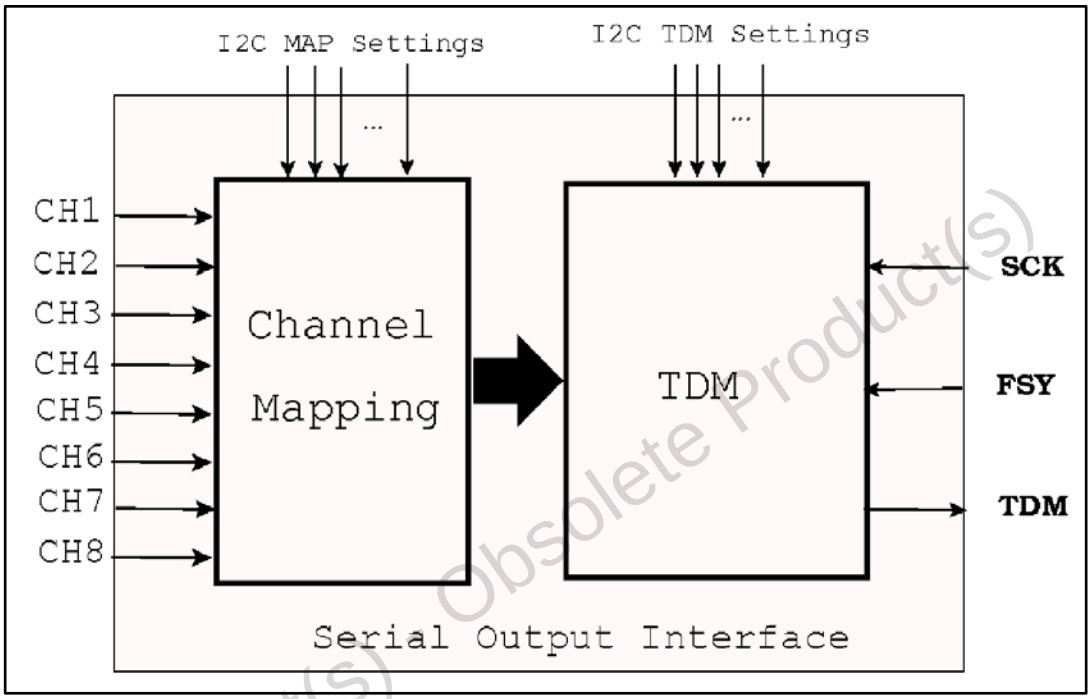
There are two types of resets in the STAMP0:

- **Hard reset**: the RESET_N pin is low
- **Soft reset**: the I²C register **Reg. 0x00** is written to any value

4.5 Serial output interface

The internal architecture of the serial output Interface is depicted in *Figure 12: "Serial output interface architecture"*, and it consists of a channel mapping and a time division multiplexing serial interface. In the following subsections the blocks mentioned before will be detailed.

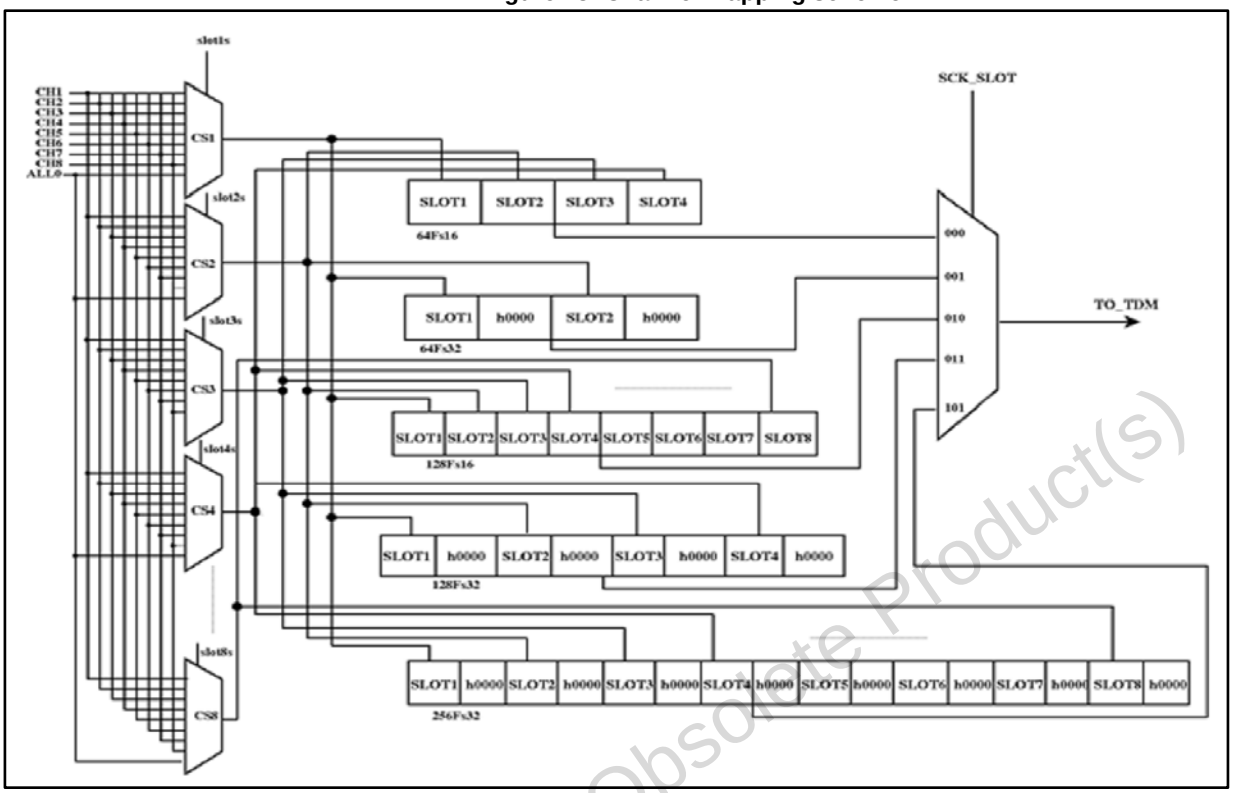
Figure 12: Serial output interface architecture



4.5.1 Channel mapping

After the downsampling block, the block depicted in *Figure 13: "Channel mapping scheme"* allows mapping any of the 8 processed channels to any of the output interface slots. Each slot consists of the audio data word followed by certain number of empty bits. For example, a 32-bit slot may consist of a 16-bit audio data word plus 16 empty bits. The SCK clock rate coming either from the AUTOPLL block (SCK_RATE signal when the PLL is automatically programmed) or directly from the I²C register (**Reg. 0x09 bit B4 and B3** when the PLL is programmed manually), and the slot size, coming from the I²C register (**Reg. 0x09 bit B5**), together define the number of slots available in a frame. In fact, as depicted in *Figure 13: "Channel mapping scheme"*, five possible frames are supported: 64Fs16, 64Fs32, 128Fs16, 128Fs32 and 256Fs32.

Figure 13: Channel mapping scheme



The mapping between the input streams and the frame output slots is flexible and can be configured using the Channel Selector (CSx in *Figure 13: "Channel mapping scheme"*) through the I²C registers **(**Reg. 0xA, Reg. 0xB, Reg. 0xC and Reg. 0xD**)**. Moreover each CSx has an extra input “ALL0” employed to set to zero the content of the slots that are not used (empty bits) in the frame out.

In each slot, audio data is arranged by the most significant bit (MSB) first, high byte first.

The frame output is selected using the SCK_SLOT signal whose value is determined by combining the SCK rate and the slot size.

However the default SCK_RATE is equal to 128xFs, the slot size is equal to 16 bits and the CH1 is mapped to the Slot1, the CH2 is mapped to the Slot2 and so on.

4.5.2 Time division multiplexing

Time-division multiplexing (TDM) is a method of putting multiple data streams in one data signal by separating the signal into many segments.

The TDM hardware interface has three signals, serial clock (SCK pin in *Figure 12: "Serial output interface architecture"*), frame sync (FS=FSY pin in *Figure 12: "Serial output interface architecture"*) and data (TDM pin in *Figure 12: "Serial output interface architecture"*).

The TDM and the FSY signals can change on the falling edges of the SCK and are valid on the rising edge of SCK (*Figure 14: "Data and FSY change on the falling edges of SCK and are valid on the rising edges of SCK. The frame starts on the rising edge of FSY (NO DELAY)"* and *Figure 16: "Data and FSY change on the falling edges of SCK and are valid on the rising edges of SCK.The frame starts on the falling edge of FSY (NO DELAY)"*) or

vice versa (*Figure 15: "Data and FSY change on the rising edges of SCK and are valid on the falling edges of SCK. The frame starts on the rising edge of FSY (NO DELAY)"* and *Figure 17: "Data and FSY change on the rising edges of SCK and are valid on the falling edges of SCK.The frame starts on the falling edge of FSY (NO DELAY)"*). This can be configured through the I²C register (Reg. 0x09 bit B2).

However, by default the TDM data and FSY signals change on the falling edges of the SCK and are valid on the rising edges.

The FSY signal is used to identify the beginning and the end of a frame. Two possible solutions are supported: the frame data start/stop on the rising edge of FSY (*Figure 14: "Data and FSY change on the falling edges of SCK and are valid on the rising edges of SCK. The frame starts on the rising edge of FSY (NO DELAY)"* and *Figure 15: "Data and FSY change on the rising edges of SCK and are valid on the falling edges of SCK. The frame starts on the rising edge of FSY (NO DELAY)"*); frame data start/stop on the falling edge of SCK (*Figure 16: "Data and FSY change on the falling edges of SCK and are valid on the rising edges of SCK.The frame starts on the falling edge of FSY (NO DELAY)"* and *Figure 17: "Data and FSY change on the rising edges of SCK and are valid on the falling edges of SCK.The frame starts on the falling edge of FSY (NO DELAY)"*). This can be configured through the I²C register (Reg. 0x09 bit B1), and by default the frame starts on the rising edge of FSY.

While the frame rate is usually the same as the audio sample rate such as 44.1 kHz and 48 kHz, the serial clock rate is a multiple of the frame rate. For example, SCK rate can be configured to be 64xFs, 128xFs and 256xFs through the I²C.

The width of the FSY signal pulse can vary from one serial clock to the highest supported SCK clock rate minus one. For example, on a system that supports 256xFs, the pulse width of the FSY signal can be from one SCK to 255 SCK clocks. So, the duty cycle of the FSY signal can vary and does not have to be 50%.

Figure 14: Data and FSY change on the falling edges of SCK and are valid on the rising edges of SCK. The frame starts on the rising edge of FSY (NO DELAY)

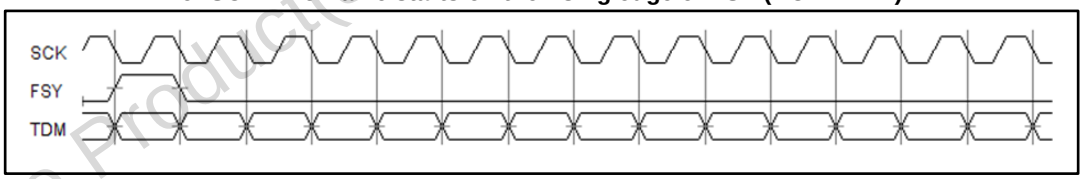


Figure 15: Data and FSY change on the rising edges of SCK and are valid on the falling edges of SCK. The frame starts on the rising edge of FSY (NO DELAY)

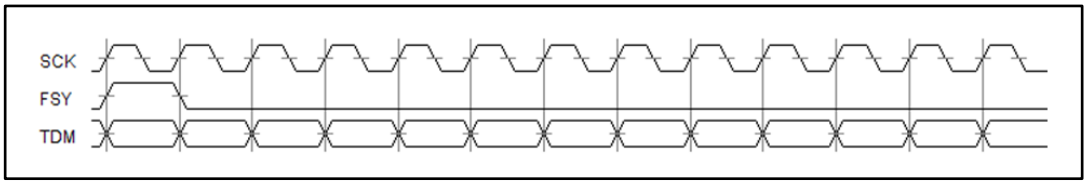


Figure 16: Data and FSY change on the falling edges of SCK and are valid on the rising edges of SCK.The frame starts on the falling edge of FSY (NO DELAY)

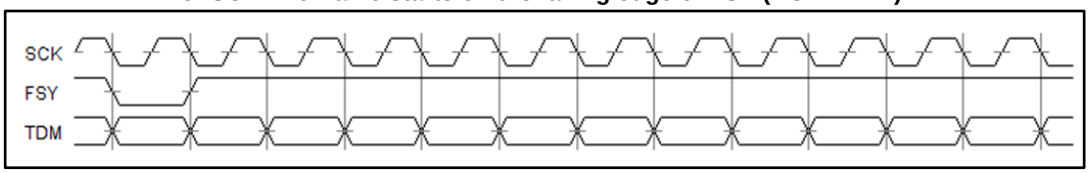
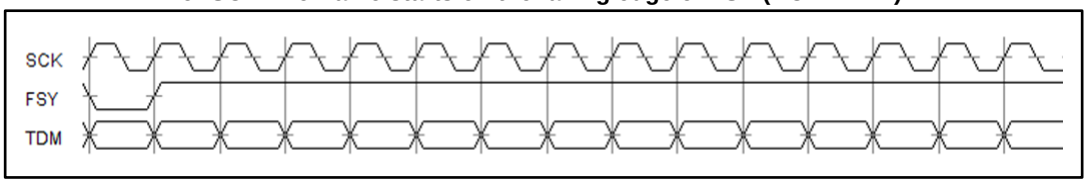


Figure 17: Data and FSY change on the rising edges of SCK and are valid on the falling edges of SCK.The frame starts on the falling edge of FSY (NO DELAY)

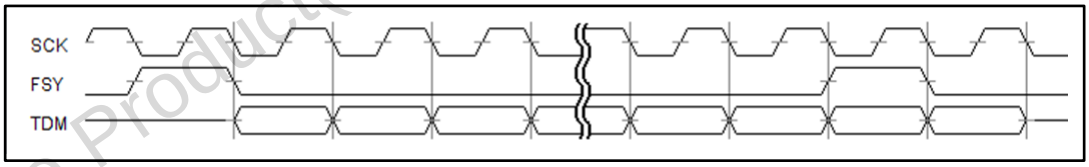


Within one frame, the DATA is divided into multiple slots as defined in *Section 5.5.1: "Channel mapping"*.

The TDM interface supports the non-delayed (*Figure 14: "Data and FSY change on the falling edges of SCK and are valid on the rising edges of SCK. The frame starts on the rising edge of FSY (NO DELAY)"*, *Figure 15: "Data and FSY change on the rising edges of SCK and are valid on the falling edges of SCK. The frame starts on the rising edge of FSY (NO DELAY)"*, *Figure 16: "Data and FSY change on the falling edges of SCK and are valid on the rising edges of SCK.The frame starts on the falling edge of FSY (NO DELAY)"* and *Figure 17: "Data and FSY change on the rising edges of SCK and are valid on the falling edges of SCK.The frame starts on the falling edge of FSY (NO DELAY)"*) sequential alignment and delayed sequential alignment data formats.

In the latter, as shown in *Figure 18: "Data and FSY change on the falling edges of SCK and are valid on the rising edges of SCK.The frame starts on the rising edge of FSY (DELAYED)"*, there is one SCK clock delay between the start/stop of the frame and the start/stop of the TDM data.

Figure 18: Data and FSY change on the falling edges of SCK and are valid on the rising edges of SCK.The frame starts on the rising edge of FSY (DELAYED)



The non-delayed and delayed alignments can be selected through the I²C register (Reg. 0x09 bit B6). The sequential (non-delayed) alignment is the default format.

4.6 I²C interface

In the STAMP0 each module contains several registers for configuration, status and testing, each one can be accessed using an I²C standard serial interface. A detailed description of the protocol is given in the following subsections.

4.6.1 I²C communication protocol

The STAMP0 supports the I²C protocol via the input ports SCL and SDA (master to slave communication). This protocol defines any device that sends data to the bus as a transmitter and any device that reads the data as a receiver. The device that controls the data transfer is known as the master and the other as the slave. The master always starts the transfer and provides the serial clock for synchronization.

4.6.2 Data transition or change

Data changes on the SDA line must only occur when the SCL clock is low. SDA transition while the clock is high is used to identify a START or STOP condition.

4.6.3 Start condition

START is identified by a high to low transition of the data bus SDA signal while the clock signal SCL is stable in the high state. A START condition must precede any command for data transfer.

4.6.4 Stop condition

STOP is identified by a low to high transition of the data bus SDA signal while the clock signal SCL is stable in the high state. A STOP condition terminates communication between the slave and the bus master.

4.6.5 Data input

During the data input the slave samples the SDA signal on the rising edge of clock SCL. For correct device operation the SDA signal must be stable during the rising edge of the clock and the data can change only when the SCL line is low.

4.6.6 Device addressing

To start communication between the master and the slave, the master must initiate with a start condition. Following this, the master sends 8 bits onto the SDA line (MSB first) corresponding to the device select address and read or write mode. The 7 most significant bits are the device address identifiers, corresponding to the I²C bus definition. The 8th bit (LSB) identifies read or write operation RW, this bit is set to 1 in read mode and 0 for write mode. In the STAMP0 device, employing the SA pin, two possible device addresses can be selected: 38 and 78. The former is selected by setting the SA pin equal to zero; the latter is selected by setting the SA input pin equal to one.

After a START condition the slave identifies on the bus the device address and if a match is found, it acknowledges the identification on the SDA bus during the 9th bit time. The byte following the device identification byte is the internal space address.

4.6.7 Write operation

Following the START condition the master sends a device select code with the RW bit set to 0. The slave acknowledges this and then waits for the byte of the internal address. After receiving the internal byte address, the slave again responds with an acknowledgement.

4.6.8 Byte write

In the byte write mode the master sends one data byte, this is acknowledged by the slave. The master then terminates the transfer by generating a STOP condition.

4.6.9 Multi-byte write

The multi-byte write modes can start from any internal address. The master generating a STOP condition terminates the transfer.

4.7 Read operation

4.7.1 Current address byte read

Following the START condition, the master sends a device select code with the RW bit set to 1. The slave acknowledges this and then responds by sending one byte of data. The master then terminates the transfer by generating a STOP condition.

4.7.2 Current address multi-byte read

The multi-byte read modes can start from any internal address. Sequential data bytes are read from sequential addresses within the slave. The master acknowledges each data byte read and then generates a STOP condition, terminating the transfer.

4.7.3 Random address byte read

Following the START condition, the master sends a device select code with the RW bit set to 0. The slave acknowledges this and then the master writes the internal address byte. After receiving the internal byte address, the slave again responds with an acknowledgement. The master then initiates another START condition and sends the device select code with the RW bit set to 1. The slave acknowledges this and then responds by sending one byte of data. The master then terminates the transfer by generating a STOP condition.

4.7.4 Random address multi-byte read

The multi-byte read mode can start from any internal address. Sequential data bytes are read from sequential addresses within the slave. The master acknowledges each data byte read and then generates a STOP condition, terminating the transfer.

Figure 19: Write mode sequence

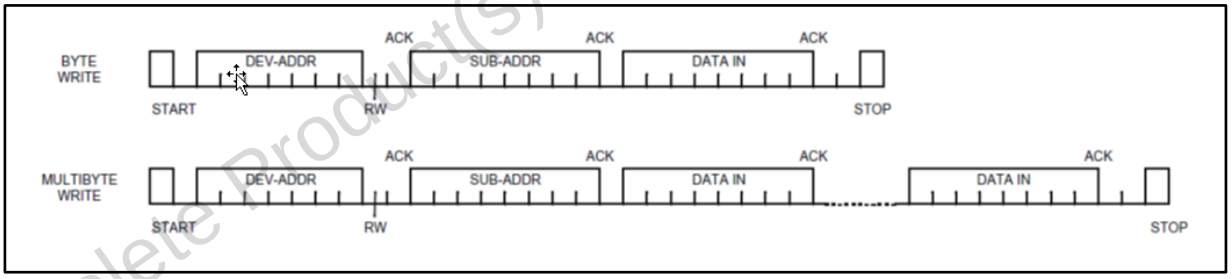
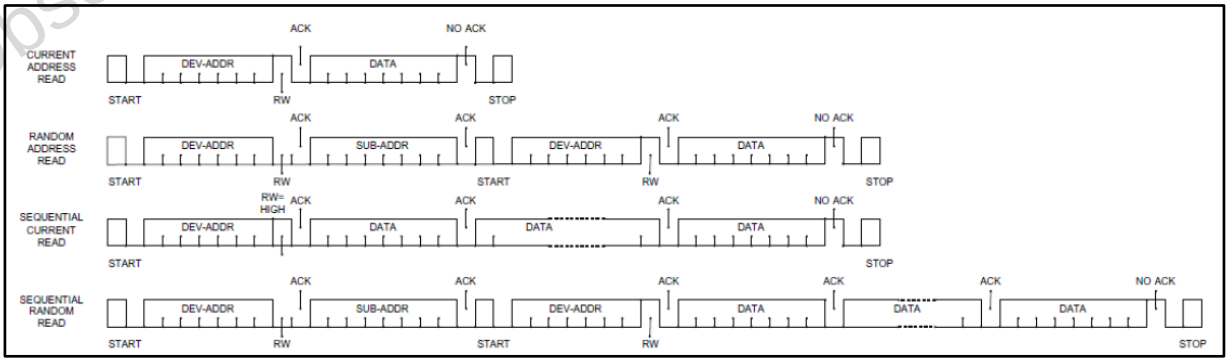


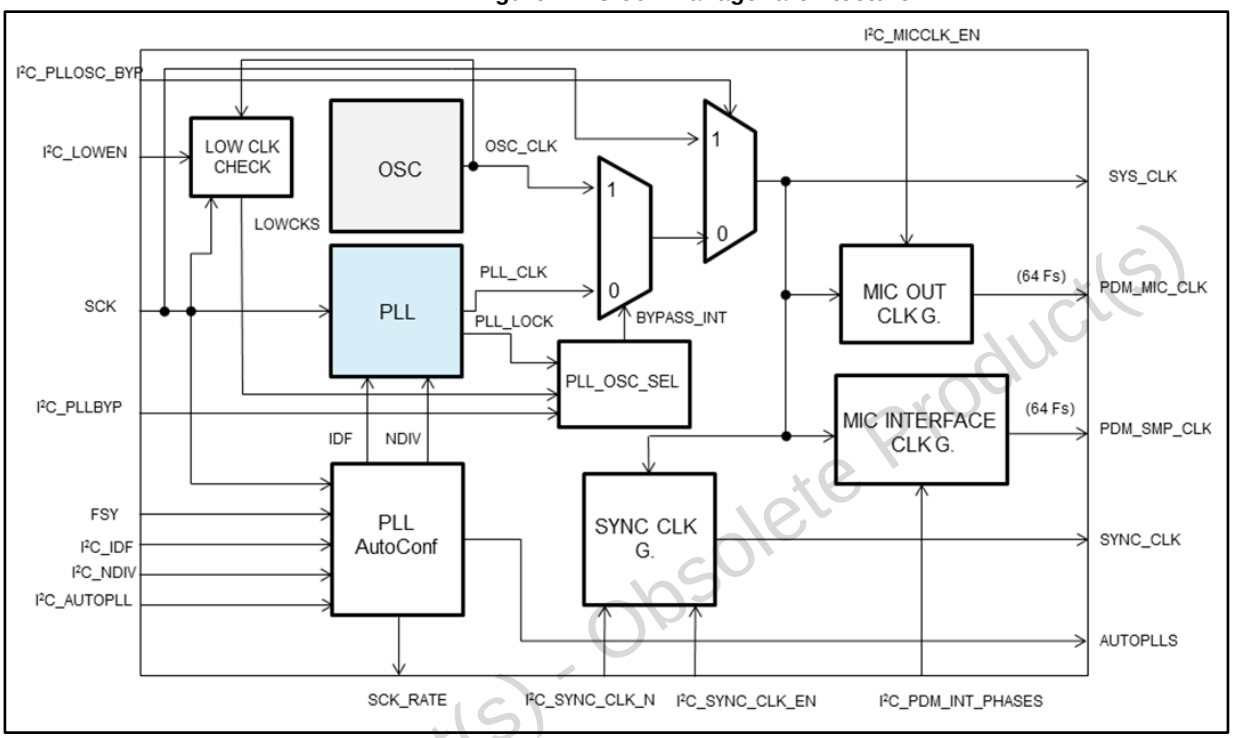
Figure 20: Read mode sequence



4.8 Clock manager

The clock manager handles all the internal and external clocks of the device. The internal architecture of the clock manager is depicted in *Figure 21: "Clock manager architecture"*. In the subsections below a detailed description of the clock manager internal blocks is given.

Figure 21: Clock manager architecture



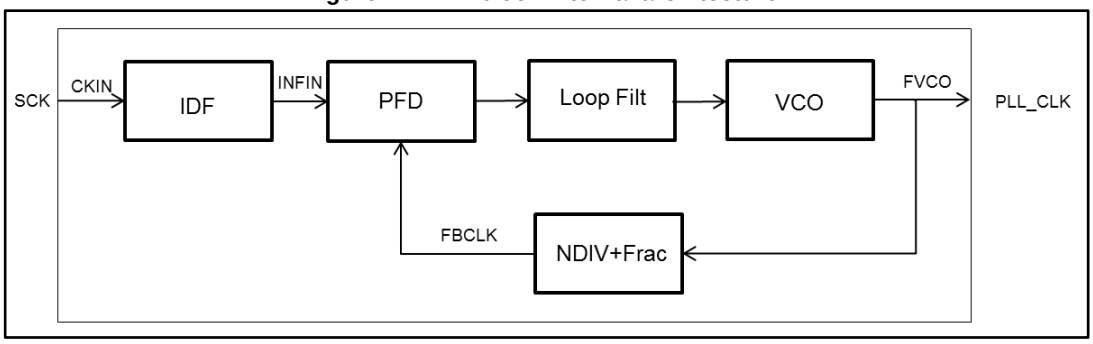
4.8.1 PLL

The PLL employed in the STAMP0 device belongs to the class of charge pump PLLs. Its internal architecture is depicted in *Figure 22: "PLL block internal architecture"*, and the internal key blocks are detailed below.

The input frequency divider divides the PLL Input frequency by a factor called the Input Division Factor (IDF). The divided frequency (INFIN) is the input to the Phase/Frequency Detector (PFD) of PLL.

The PFD block compares the phase difference between the corresponding rising edges of INFIN and clock output from the Loop Frequency Divider (FBCLK), and it generates voltage pulses with width proportional to the phase difference. The charge pump and loop filter block convert the voltage pulses from the PFD to current pulses which charge the PLL Loop Filter to generate the control voltage for the Voltage Controlled Oscillator (VCO).

Figure 22: PLL block internal architecture



The VCO inside the PLL produces a frequency output (FVCO) proportional to the Input Control Voltage. The loop frequency divider (NDIV+FRAC) is present within the PLL loop for dividing FVCO by a factor called the Loop Division Factor (NDIV). The output of this block is FBCLK.

The loop divider can be configured in fractional mode so that the overall PLL multiplication factor could be a rational number. The STAMP0 PLL provides a LOCKP signal so that the output of the Lock circuit is asserted high when the PLL enters the state of COARSE LOCK and low when the PLL is in the UNLOCK state. If the output frequency is within ±10% (approx.) of the desired frequency, the PLL is said to be in COARSE LOCK, otherwise the PLL is in the UNLOCK state. The LOCKP signal is refreshed after every 32 cycles of INFIN. This is generated based on the result of the comparison of the number of FBCLK cycles in a window of 14 INFIN cycles. The different cases generated after comparison are:

If LOCKP is already at "0", then in the next refresh cycle LOCKP goes to "1" if the number of FBCLK cycles in the 14 cycle INFIN window is 13, 14 or 15. Otherwise LOCKP stays at "0"

If LOCKP is already at "1", then in the next refresh cycle LOCKP goes to "0" if the number of FBCLK cycles in 14 cycle INFIN window is less than 11 or higher than 17, otherwise LOCKP stays at "1".

If LOCKP is already at "1" and CLKIN disappears, LOCKP will stay at "1" though the PLL will get unlocked.

The output PLL frequency can be expressed as:

Equation 3

$$F_{VCO} = F_{INFIN} \times \left(NDIV + \frac{FRAC}{2^{16}} + 2^{-17} \right)$$

where:

when DITHER_DISABLE[1] ='1', the factor 2^{-17} will not be present in the formula

when FRAC_CTRL='0' also the FRAC term will not be present in the formula.

The PLL receives as input the SCK external clock and it generates a PLL_CLK frequency equal to 2048xFs. Moreover the PLL is programmed using the IDF and NDIV parameters coming from the PLL AutoConf block. *Table 8: "PLL frequency values"* indicates the PLL_CLK frequencies generated as a function of the SCK and FSY frequencies.

Table 8: PLL frequency values

SCKxFs	SCK frequency (Fs = 44.1 kHz / 48.0 kHz)	IDF	NDIV	PLL_CLK
64 Fs	2.8224 MHz / 3.072 MHz	1	32	90.3168 MHz /98.304 MHz
128 Fs	5.6448 MHz / 6.144 MHz	1	16	90.3168 MHz /98.304 MHz
256 Fs	11.2896 MHz / 12.288 MHz	1	8	90.3168 MHz /98.304 MHz

4.8.2 Oscillator

The STAMPO internal oscillator guarantees the functionalities of all the internal blocks either if the PLL is unlocked or a low clock is detected. Its output frequency is equal to 20 MHz.

4.8.3 PLL autoconf

The PLL AutoConf main functionalities are:

- **PLL programming** either using the I²C registers Reg. 0x01 and 0x02 (PLL manual programming mode), or the SCK and FSY TDM external signals (automatic programming mode). The automatic and the manual PLL programming functionalities work only if both the SCK and FSY external clocks are given.
- **SCK and FSY frequencies integrity monitoring** (both in manual and in automatic programming mode)
- **SCK_RATE frequency computation** (only in manual mode)

This block receives as input the SCK, FSY, I²C_AUTOPLL, I²C_IDF and the I²C_NIDV signals and it generates the IDF and NDIV signals required to program the PLL, and the SCK_rate value to configure the channel mapping and the TDM interface.

This block generates an interrupt as soon as the FSY/SCK ratio differs from 64, 128 and 256.

4.8.4 Low-clock check

This block is a simply an SCK low frequency detection circuit. In particular it receives as input the SCK clock signal, the oscillator output frequency, and it generates the output LOWCKS signal equal to ‘1’ as soon as the SCK clock frequency falls below an internal threshold. The output of this block can be monitored using the I²C register (Reg. 0x06, bit B0**)**.

4.8.5 PLL_OSC_SEL

This block receives as input the LOWCKS, PLL_LOCK and the I²C_PLLBYP (**Reg. 0x05,** bit B2), and it generates as output the BYPASS_INT signal. The BYPASS_INT signal is equal to ‘1’ either if the PLL is unlocked or the LOWCKS signal is set to ‘1’. In all the other cases the BYPASS_INT signal is equal to ‘0’. Moreover the PLL can be bypassed through the I²C register REG 0x05 Bit 2 (I²C_PLLBYP signal in *Figure 21: "Clock manager architecture"*).

4.8.6 SYNC clock generator

This block simply receives as inputs the SYS_CLK, the I²C SYNC_CLK_N signal (Reg. 0x07 bits B5, B4, B3, B2, B1, B0), the I²C SYNC_CLK_EN signal (Reg. 0x07, bit B6) and it generates the SYNC_CLK frequency according to the value given in *Table 9: "Sync clock frequencies"*.

Table 9: Sync clock frequencies

Reg.0x07 (hex value)	SYNC_CLK_N (decimal value)	SYNC_CLK (Fs = 44.1 kHz) (MHz)	SYNC_CLK (Fs = 44.1 kHz) (MHz)
0x68	40	2.25792	2.4576
0x6A	42	2.1504	2.340571
0x6C	44	2.052655	2,234182
0x6E	46	1.963409	2.137043
0x70	48	1.881600	2.048000
0x72	50	1.806336	1.966080
0x74	52	1.736862	1.890462
0x76	54	1.672533	1.820444
0x78	56	1.6128	1.755429
0x7A	58	1.55718	1.694897

By default SYNC_CLK output is low when the chip powers up and stays low until the I²C SYNC_CLK_EN is not set to one. Moreover the SYNC_CLK output stays low when PLL is not locked. Finally when PLL changes from locked to unlocked, the SYNC_CLK output changes to low and an interrupt is generated.

4.8.7 MIC clock out generator

This block receives as input SYS_CLK and it generates a PDM_MIC_CLK clock signal whose frequency is equal to 64xFs.

By default the PDM_MIC_CLK output is active and can be disabled by setting the I²C register (Reg. 0x08 bit B4) to zero.

4.8.8 PLL and oscillator power-down

In the STAMP0 device it is possible to power off/on both the oscillator and the PLL using the I²C register (Reg. 0x2D bit B0).

5 I²C registers

5.1 Register summary

Table 10: Register summary

Addr.	Name	B7	B6	B5	B4	B3	B2	B1	B0
0x00	soft_rst	RST_SOFT							
Clock manager configuration registers									
0x01	PLL1	PDPDC	PLLFC	PLL STRB	PLL STRBB	IDF			
0x02	PLL2	DITHD		NDIV					
0x03	PLL3	PLLFI[15:8]							
0x04	PLL4	PLLFI[7:0]							
0x05	Clkmgr1	-	-	-	-	PLLOSC_ BYP	PLLBYP	LOWEN	AUTO PLL
0x06	Clkmgr2	-	-	-	AUTO PLLS	PLL BYPS	PLLPDS	OSCOK	LOW CKS
0x07	Clkmgr3	-	SYNC_ CLK_EN	SYNC_CLK_N					
0x08	Clkmgr4	PDM INT PHASES			MICCLK_ EN	-	-	-	-
Serial output interface									
0x09	TDM	-	DATA_ ALIGN	SLOT_ SIZE	SCK_ RATE		DATA_ FSY_ VALID	FRAME_ START	TDM_ EN
Channel mapping configuration registers									
0x0A	CH12_map	SLOT1s				SLOT2s			
0x0B	CH34_map	SLOT3s				SLOT4s			
0x0C	CH56_map	SLOT5s				SLOT6s			
0x0D	CH78_map	SLOT7s				SLOT8s			
RMS meter configuration registers									
0x0E	RMS1	COMP_B	COMP_SEL			-			
0x0F	RMS2	RMS_TH[15:8]							
0x10	RMS3	RMS_TH[7:0]							
0x11	RMS4	RMS_EN[15:8]							
0x12	RMS5	RMS_EN[7:0]							
0x13	RMS6	RMS_CH1[15:8]							
0x14	RMS7	RMS_CH1[7:0]							
0x15	RMS8	RMS_CH2[15:8]							
0x16	RMS9	RMS_CH2[7:0]							
0x17	RMS10	RMS_CH3[15:8]							

Addr.	Name	B7	B6	B5	B4	B3	B2	B1	B0
0x18	RMS11	RMS_CH3[7:0]							
0x19	RMS12	RMS_CH4[15:8]							
0x1A	RMS13	RMS_CH4[7:0]							
0x1B	RMS14	RMS_CH5[15:8]							
0x1C	RMS15	RMS_CH5[7:0]							
0x1D	RMS16	RMS_CH6[15:8]							
0x1E	RMS17	RMS_CH6[7:0]							
0x1F	RMS18	RMS_CH7[15:8]							
0x20	RMS19	RMS_CH7[7:0]							
0x21	RMS20	RMS_CH8[15:8]							
0x22	RMS21	RMS_CH8[7:0]							
Stuck at PDM configuration register									
0x23	PDM_STUCK	P				NSAW			
Interrupt and masking configuration registers									
0x24	Int1	RMS8s	RMS7s	RMS6s	RMS5s	RMS4s	RMS3	RMS2s	RMS1s
0x25	Int2	STUCK1_PDM8	STUCK0_PDM8	STUCK1_PDM7	STUCK0_PDM7	STUCK1_PDM6	STUCK0_PDM6	STUCK1_PDM5	STUCK0_PDM5
0x26	Int3	STUCK1_PDM4	STUCK0_PDM4	STUCK1_PDM3	STUCK0_PDM3	STUCK1_PDM2	STUCK0_PDM2	STUCK1_PDM1	STUCK0_PDM1
0x27	Int4	-	-	-	-	-	-	AUTO PLL	PLLBYP
0x28	Mask1	MRMS8s	MRMS7s	MRMS6s	MRMS5s	MRMS4s	MRMS3s	MRMS2s	MRMS1s
0x29	Mask2	MSTUCK1_PDM4	MSTUCK0_PDM4	MSTUCK1_PDM3	MSTUCK0_PDM3	MSTUCK1_PDM2	MSTUCK0_PDM2	MSTUCK1_PDM1	MSTUCK0_PDM1
0x2A	Mask3	MSTUCK1_PDM8	MSTUCK0_PDM8	MSTUCK1_PDM7	MSTUCK0_PDM7	MSTUCK1_PDM6	MSTUCK0_PDM6	MSTUCK1_PDM5	MSTUCK0_PDM5
GPIO configuration registers									
0x2B	GPIO1	-	-	-	-	-	-	LR_DIR	LR_OUT
0x2C	GPIO2	-	-	-	-	-	-	-	LR_IN
PLL and oscillator power-down configuration register									
0x2D	PPLOSC_PD	-	-	-	-	-	-	-	PLL_OSC_PD
Microphone processing: dual-membrane handling									
0x2E	SMPCTRL	smpM4b	smpM4a	smpM3b	smpM3a	smpM2b	smpM2a	smpM1b	smpM1a
0x2F	MIKE CTRL	-	HYST			res_int_mode		m_rbyp	m_en
0x30	MRSENS1	RMCH_1	BYP RM1	Microphone 1 Global Gain (sensitivity compensation)					
0x31	MRSENS2	RMCH_2	BYP RM2	Microphone 2 Global Gain (sensitivity compensation)					
0x32	MRSENS3	RMCH_3	BYP RM3	Microphone 3 Global Gain (sensitivity compensation)					

Addr.	Name	B7	B6	B5	B4	B3	B2	B1	B0
0x33	MRSENS4	RMCH_4	BYP RM4	Microphone 4 Global Gain (sensitivity compensation)					
0x34	MRATT1	-	LP1 EN	Microphone 1 Low Channel Attenuation					
0x35	MRATT2	-	LP2 EN	Microphone 2 Low Channel Attenuation					
0x36	MRATT3	-	LP3 EN	Microphone 3 Low Channel Attenuation					
0x37	MRATT4	-	LP4 EN	Microphone 4 Low Channel Attenuation					
0x38	MRTHN1	-		Microphone 1 Low Channel Threshold					
0x39	MRTHN2	-		Microphone 2 Low Channel Threshold					
0x3A	MRTHN3	-		Microphone 3 Low Channel Threshold					
0x3B	MRTHN4	-		Microphone 4 Low Channel Threshold					
0x3C	MRTHH1	-		Microphone 1 High Channel Threshold					
0x3D	MRTHH2	-		Microphone 2 High Channel Threshold					
0x3E	MRTHH3	-		Microphone 3 High Channel Threshold					
0x3F	MRTHH4	-		Microphone 4 High Channel Threshold					
0x40	M6dB BOOST	CH8	CH7	CH56	CH5	CH4	CH3	CH2	CH1

5.2 Register description

5.2.1 Soft-reset configuration (reg. 0x00)

B7	B6	B5	B4	B3	B2	B1	B0
RST_SOFT							
0	0	0	0	0	0	0	0

Bit	R/W	RST	Name	Description
0	R/W	0		Soft-reset active-low
1	R/W	0		
2	R/W	0		
3	R/W	0		
4	R/W	0		
5	R/W	0		
6	R/W	0		
7	R/W	0		

5.2.2 PLL input division factor (IDF) and others (reg. 0x01)

B7	B6	B5	B4	B3	B2	B1	B0
PDPDC	PLLFC	PLLSTRB	PLLSTRBB	IDF			
0	0	0	0	0	0	0	1

Bit	R/W	RST	Name	Description
7	R/W	0	PDPDC	**0**: if IDF and NDIV values are changed, the PLL power-OFF and power-ON sequences are performed **1**: if the IDF and NDIF values are changed, the PLL is definitively powered OFF
6	R/W	0	PLLFC	PLL fractional control
5	R/W	0	PLLSTRB	PLL strobe
4	R/W	0	PLLSTRBB	PLL strobe bypass
3	R/W	0	IDF	Input Division Factor
2	R/W	0		
1	R/W	0		
0	R/W	1		

For certain applications and to provide flexibility to the user, a manual PLL configuration can be used (setting PLLFC to 1). The output PLL frequency formula is:

Equation 4

$$F_{out} = \frac{F_{IN}}{IDF} \times \left(ND + \frac{FI}{2^{16}}\right) whenPLLFC = 1$$

Equation 5

$$F_{out} = \frac{F_{IN}}{IDF} \times (ND) whenPLLFC = 0$$

5.2.3 PLL multiplication factor (integral part) named as N division factor (NDIV) and dithering (reg. 0x02)

B7	B6	B5	B4	B3	B2	B1	B0
DITHD		NDIV					
0	0	1	0	0	0	0	0

Bit	RW	RST	Name	Description
7	R/W	0	DITHD1	PLL dithering: **00**: PLL clock dithering disabled **01**: PLL clock dithering enabled (triangular) **10**: PLL clock dithering enabled (rectangular) **11**: reserved
6	R/W	0	DITHD0	
5	R/W	1	NDIV5	N (loop) division factor values: 32, 16, 8
4	R/W	0	NDIV4	
3	R/W	0	NDIV3	
2	R/W	0	NDIV2	
1	R/W	0	NDIV1	
0	R/W	0	NDIV0	

5.2.4 PLL multiplication factor, fractional part MSB (reg. 0x03)

B7	B6	B5	B4	B3	B2	B1	B0
PLLFI[15:8]							
0	0	0	0	0	0	0	0

5.2.5 PLL multiplication factor, fractional part LSB (reg. 0x04)

B7	B6	B5	B4	B3	B2	B1	B0
PLLFI[7:0]							
0	0	0	0	0	0	0	0

5.2.6 Clock manager configuration register (reg.0x05)

B7	B6	B5	B4	B3	B2	B1	B0
-	-	-	-	PLLOSC_BYP	PLLBYP	LOWEN	AUTOPLL
-	-	-	-	0	0	0	1

Bit	R/W	RST	Name	Description
3	R/W	0	PLLOSC_BYP	PLLOSC_BYP: **1**: PLL and oscillator bypassed **0**: PLL and oscillator not bypassed
2	R/W	0	PLLBYP	PLL bypass: **0**: PLL not bypassed **1**: PLL bypassed
1	R/W	0	LOWEN	Low clock enable: **0**: if input clock is too slow, master clock will become the internal oscillator clock (20 MHz), PLL bypassed **1**: disabled
0	R/W	1	AUTOPLL	PLL programmed manually: **0**: PLL programmed using the SCK and FSY external signals **1**: PLL programmed using the I²C registers 0x1 and 0x2

5.2.7 Clock manager status register (reg. 0x06)

B7	B6	B5	B4	B3	B2	B1	B0
-	-	-	AUTOPLLS	PLLBYPS	PLLPDS	OSCOK	LOWCKS
-	-	-	0	0	0	0	0

Bit	R/W	RST	Name	Description
4	R	0	AUTOPLLS	**1**: the PLL is programmed correctly; **0**: FSY/SCK differ from 64,128, 256
3	R	0	PLLBYPS	PLL bypass status: **0**: normal **1**: bypassed
2	R	0	PLLPDS	PLL power-down status: **0**: normal **1**: standby
1	R	0	OSCOK	Oscillator clock OK: **0**: not ready 1: ready
0	R	0	LOWCKS	Low-clock status: **0**: normal **1**: input SCK clock too slow

5.2.8 Synchronization clock out configuration register (reg. 0x07)

B7	B6	B5	B4	B3	B2	B1	B0
-	-	SYNC_CLK_N					
-	-	1	0	1	1	1	0

Bit	R/W	RST	Name	Description
7	R/W	-	-	-
6	R/W	0	SYNC_CLK_EN	**0**: sync. clock set to zero **1**: sync. clock enabled
5	R/W	1	SYNC_CLK_N	See *Table 11: "Sync_CLK possible frequency values"*
4	R/W	0		
3	R/W	1		
2	R/W	1		
1	R/W	1		
0	R/W	0		

Table 11: Sync_CLK possible frequency values

Hex value (SYNC enabled)	SYNC_CLK_N (decimal value)	SYNC_CLK (Fs = 44.1 Hz) (Hz)	SYNC_CLK (Fs = 48.0 Hz) (Hz)
0x68	40	2.25792E+06	2.4576E+06
0x6A	42	2.1504E+06	2.340571E+06
0x6C	44	2.052655E+06	2.234182E+06
0x6E	46	1.963409E+06	2.137043E+06
0x70	48	1.881600E+06	2.048000E+06
0x72	50	1.806336E+06	1.966080E+06
0x74	52	1.736862E+06	1.890462E+06
0x76	54	1.672533E+06	1.820444E+06
0x78	56	1.6128E+06	1.755429E+06
0x7A	58	1.55718E+06	1.694897E+06

5.2.9 Clock manager configuration register (reg. 0x08)

B7	B6	B5	B4	B3	B2	B1	B0
PDM_INT_PHASES			MICCLK_EN	-	-	-	-
011			1	-	-	-	-

Bit	R/W	RST	Name	Description
7	R/W	1	PDM_INT_PHASES	**000**: 0 degrees phase shift **001**: 45 degrees phase shift **010**: 90 degrees phase shift **011**: 135 degrees phase shift **100**: 180 degrees phase shift **101**: 225 degrees phase shift **110**: 270 degrees phase shift **111**: 315 degrees phase shift
6	R/W	0		
5	R/W	0		
4	R/W	1	MICCLK_EN	Microphone clock enable: **0**: microphones clock OFF **1**: microphones clock ON

5.2.10 TDM configuration register (reg. 0x09)

B7	B6	B5	B4	B3	B2	B1	B0
-	DATA_ ALIGN	SLOT_ SIZE	SCK_RATE		DATA_FSY_ VALID	FRAME_ START	TDM_ EN
-	0	0	01		0	0	1

Bit	R/W	RST	Name	Description
6	R/W	0	DATA_ALIGN	TDM data alignment **0**: non delayed **1**: delayed
5	R/W	0	SLOT_SIZE	TDM slot size: **0**: 16 bits **1**: 32 bits
4	R/W	0	SCK_RATE	TDM SCK rate: **00**: 64 Fs **01**: 128 Fs **10**: 256 Fs **11**: reserved
3	R/W	1		
2	R/W	0	DATA_FSY_VALID	Data and frame sync valid: **0**: DATA and FSY valid on the rising edge of SCK **1**: DATA and FSY valid on the falling edge of SCK
1	R/W	0	FRAME_START	Frame start: **0**: on the rising edge of SCK **1**: on the falling edge of SCK
0	R/W	1	TDM_EN	TDM interface enable: **0**: disabled **1**: enabled

5.2.11 Channel 1 and 2 mapping configuration register (reg. 0x0A)

B7	B6	B5	B4	B3	B2	B1	B0
SLOT1S				SLOT2S			
0000 (See *Table 12: "TDM slots and channel mapping"*)				0001 (See *Table 12: "TDM slots and channel mapping"*)			

Table 12: TDM slots and channel mapping

R/W	B7/B3	B6/B2	B5/B1	B4/B0	Channel	TDM slots
R/W	0	0	0	0	CH1	**SLOTX**
R/W	0	0	0	1	CH2	
R/W	0	0	1	0	CH3	
R/W	0	0	1	1	CH4	
R/W	0	1	0	0	CH5	
R/W	0	1	0	1	CH6	
R/W	0	1	1	0	CH7	
R/W	0	1	1	1	CH8	
R/W	1	0	0	0	All zeros	

5.2.12 Channel 3 and 4 mapping configuration register (reg. 0x0B)

B7	B6	B5	B4	B3	B2	B1	B0
SLOT3S				SLOT4S			
0010 (See *Table 12: "TDM slots and channel mapping"*)				0011 (See *Table 12: "TDM slots and channel mapping"*)			

5.2.13 Channel 5 and 6 mapping configuration register (reg. 0x0C)

B7	B6	B5	B4	B3	B2	B1	B0
SLOT5S				SLOT6S			
0100 (See *Table 12: "TDM slots and channel mapping"*)				0101 (See *Table 12: "TDM slots and channel mapping"*)			

5.2.14 Channel 7 and 8 mapping configuration register (reg. 0x0D)

B7	B6	B5	B4	B3	B2	B1	B0
SLOT7S (See *Table 12: "TDM slots and channel mapping"*)				SLOT8S (See *Table 12: "TDM slots and channel mapping"*)			
0110				0111			

5.2.15 RMS meter threshold programming (reg. 0x0E)

B7	B6	B5	B4	B3	B2	B1	B0
COMP_B	COMP_SEL			-	-	-	-
0	000			-	-	-	-

Bit	R/W	RST	Name	Description
7	R/W	0	COMP_B	Burst threshold prog: **0**: single threshold prog. **1**: burst threshold prog.
6	R/W	0	COMP_SEL	Single threshold prog: **000**: TH1 is programmed **001**: TH2 is programmed **010**: TH3 is programmed **011**: TH4 is programmed **100**: TH5 is programmed **101**: TH6 is programmed **110**: TH7 is programmed **111**: TH8 is programmed
5	R/W	0		
4	R/W	0		

5.2.16 RMS meter threshold value MSB (reg. 0x0F)

B7	B6	B5	B4	B3	B2	B1	B0
RMS_TH[15:8]							
0	0	0	0	0	0	0	0

5.2.17 RMS meter threshold value LSB (reg. 0x10)

B7	B6	B5	B4	B3	B2	B1	B0
RMS_TH[7:0]							
0	0	0	1	1	1	1	1

5.2.18 RMS meter enable value MSB (reg. 0x11)

B7	B6	B5	B4	B3	B2	B1	B0
RMS_EN[15:8]							
0	0	0	0	0	0	0	0

5.2.19 RMS meter enable value LSB (reg.0x12)

B7	B6	B5	B4	B3	B2	B1	B0
RMS_EN[7:0]							
0	0	0	0	0	0	0	1

5.2.20 RMS channel1 value MSB (reg.0x13)

B7	B6	B5	B4	B3	B2	B1	B0
RMS_CH1[15:8]							
0	0	0	0	0	0	0	0

5.2.21 RMS channel1 value LSB (reg. 0x14)

B7	B6	B5	B4	B3	B2	B1	B0
RMS_CH1[7:0]							
0	0	0	0	0	0	0	0

5.2.22 RMS channel2 value MSB (reg. 0x15)

B7	B6	B5	B4	B3	B2	B1	B0
RMS_CH2[15:8]							
0	0	0	0	0	0	0	0

5.2.23 RMS channel2 value LSB (reg. 0x16)

B7	B6	B5	B4	B3	B2	B1	B0
RMS_CH2[7:0]							
0	0	0	0	0	0	0	0

5.2.24 RMS channel3 value MSB (reg. 0x17)

B7	B6	B5	B4	B3	B2	B1	B0
RMS_CH3[15:8]							
0	0	0	0	0	0	0	0

5.2.25 RMS channel3 value LSB (reg. 0x18)

B7	B6	B5	B4	B3	B2	B1	B0
RMS_CH3[7:0]							
0	0	0	0	0	0	0	0

5.2.26 RMS channel4 value MSB (reg. 0x19)

B7	B6	B5	B4	B3	B2	B1	B0
RMS_CH4[15:8]							
0	0	0	0	0	0	0	0

5.2.27 RMS channel4 value LSB (reg. 0x1A)

B7	B6	B5	B4	B3	B2	B1	B0
RMS_CH4[7:0]							
0	0	0	0	0	0	0	0

5.2.28 RMS channel5 value MSB (reg. 0x1B)

B7	B6	B5	B4	B3	B2	B1	B0
RMS_CH5[15:8]							
0	0	0	0	0	0	0	0

5.2.29 RMS channel5 value LSB (reg. 0x1C)

B7	B6	B5	B4	B3	B2	B1	B0
RMS_CH5[7:0]							
0	0	0	0	0	0	0	0

5.2.30 RMS channel6 value MSB (reg. 0x1D)

B7	B6	B5	B4	B3	B2	B1	B0
RMS_CH6[15:8]							
0	0	0	0	0	0	0	0

5.2.31 RMS channel6 value LSB (reg. 0x1E)

B7	B6	B5	B4	B3	B2	B1	B0
RMS_CH6[7:0]							
0	0	0	0	0	0	0	0

5.2.32 RMS channel7 value MSB (reg. 0x1F)

B7	B6	B5	B4	B3	B2	B1	B0
RMS_CH7[15:8]							
0	0	0	0	0	0	0	0

5.2.33 RMS channel7 value LSB (reg. 0x20)

B7	B6	B5	B4	B3	B2	B1	B0
RMS_CH7[7:0]							
0	0	0	0	0	0	0	0

5.2.34 RMS channel8 value MSB (reg. 0x21)

B7	B6	B5	B4	B3	B2	B1	B0
RMS_CH8[15:8]							
0	0	0	0	0	0	0	0

5.2.35 RMS channel8 value LSB (reg. 0x22)

B7	B6	B5	B4	B3	B2	B1	B0
RMS_CH8[7:0]							
0	0	0	0	0	0	0	0

5.2.36 PDM stuck-at configuration register (reg. 0x23)

B7	B6	B5	B4	B3	B2	B1	B0
P				NSAW			
0000				1000			

Bit	R/W	RST	Name	Description
7	R/W	0	P	RMS meter IIR low-pass filter P parameter. The possible values are in the range [0:15].
6	R/W	0		
5	R/W	0		
4	R/W	0		
3	R/W	1	NSAW	Number of consecutive stuck-at windows. The possible values are in the range [1:15].
2	R/W	0		
1	R/W	0		
0	R/W	0		

5.2.37 Interrupt 1 configuration register (reg. 0x24)

B7	B6	B5	B4	B3	B2	B1	B0
RMS8s	RMS7s	RMS6s	RMS5s	RMS4s	RMS3s	RMS2s	RMS1s
0	0	0	0	0	0	0	0

Bit	R/W	RST	Name	Description
7	R/W	0	RMS8s	RMSx interrupts: **0**: the RMS value of channel x is less than the upper limit **1**: the RMS value of channel x is greater than the upper limit
6	R/W	0	RMS7s	
5	R/W	0	RMS6s	
4	R/W	0	RMS5s	
3	R/W	0	RMS4s	
2	R/W	0	RMS3s	
1	R/W	0	RMS2s	
0	R/W	0	RMS1s	

5.2.38 Interrupt 2 configuration register (reg. 0x25)

B7	B6	B5	B4	B3	B2	B1	B0
STUCK1_ PDM8	STUCK0_ PDM8	STUCK1_ PDM7	STUCK0_ PDM7	STUCK1_ PDM6	STUCK0_ PDM6	STUCK1_ PDM5	STUCK0_ PDM5
0	0	0	0	0	0	0	0

Bit	R/W	RST	Name	Description
7	R/W	0	STUCK1_PDM8	**0:** PDM8 is not stuck at 1 **1:** PDM8 is stuck at 1
6	R/W	0	STUCK0_PDM8	**0:** PDM8 is not stuck at 0 **1:** PDM8 is stuck at 0
5	R/W	0	STUCK1_PDM7	**0:** PDM7 is not stuck at 1 **1:** PDM7 is stuck at 1
4	R/W	0	STUCK0_PDM7	**0:** PDM7 is not stuck at 0 **1:** PDM7 is stuck at 0
3	R/W	0	STUCK1_PDM6	**0:** PDM6 is not stuck at 1 **1:** PDM6 is stuck at 1
2	R/W	0	STUCK0_PDM6	**0:** PDM6 is not stuck at 0 **1:** PDM6 is stuck at 0
1	R/W	0	STUCK1_PDM5	**0:** PDM5 is not stuck at 1 **1:** PDM5 is stuck at 1
0	R/W	0	STUCK0_PDM5	**0:** PDM5 is not stuck at 0 **1:** PDM5 is stuck at 0

5.2.39 Interrupt 3 configuration register (reg. 0x26)

B7	B6	B5	B4	B3	B2	B1	B0
STUCK1_PDM4	STUCK0_PDM4	STUCK1_PDM3	STUCK0_PDM3	STUCK1_PDM2	STUCK0_PDM2	STUCK1_PDM1	STUCK0_PDM1
0	0	0	0	0	0	0	0

Bit	R/W	RST	Name	Description
7	R/W	0	STUCK1_PDM4	**0:** PDM4 is not stuck at 1 **1:** PDM4 is stuck at 1
6	R/W	0	STUCK0_PDM4	**0:** PDM4 is not stuck at 0 **1:** PDM4 is stuck at 0
5	R/W	0	STUCK1_PDM3	**0:** PDM3 is not stuck at 1 **1:** PDM3 is stuck at 1
4	R/W	0	STUCK0_PDM3	**0:** PDM3 is not stuck at 0 **1:** PDM3 is stuck at 0
3	R/W	0	STUCK1_PDM2	**0:** PDM2 is not stuck at 1 **1:** PDM2 is stuck at 1
2	R/W	0	STUCK0_PDM2	**0:** PDM2 is not stuck at 0 **1:** PDM2 is stuck at 0
1	R/W	0	STUCK1_PDM1	**0:** PDM1 is not stuck at 1 **1:** PDM1 is stuck at 1
0	R/W	0	STUCK0_PDM1	**0:** PDM1 is not stuck at 0 **1:** PDM1 is stuck at 0

5.2.40 Interrupt 4 configuration register (reg. 0x27)

B7	B6	B5	B4	B3	B2	B1	B0
-	-	-	-	-	-	AUTOPLL	PLLBYP
-	-	-	-	-	-	0	0

Bit	R/W	RST	Name	Description
1	R/W	0	AUTOPLL	**0**: PLL is programmed correctly **1**: PLL is not programmed correctly
0	R/W	0	PLLBYP	**0**: PLL locked **1**: PLL not locked

5.2.41 Interrupt masking 1 configuration register (reg. 0x28)

B7	B6	B5	B4	B3	B2	B1	B0
MRMS8s	MRMS7s	MRMS6s	MRMS5s	MRMS4s	MRMS3s	MRMS2s	MRMS1s
0	0	0	0	0	0	0	0

Bit	R/W	RST	Name	Description
7	R/W	0	MRMS8s	**0**: RMS upper limit interrupt on channel 8 is not masked **1**: RMS upper limit interrupt on channel 8 is masked
6	R/W	0	MRMS7s	**0**: RMS upper limit interrupt on channel 7 is not masked **1**: RMS upper limit interrupt on channel 7 is masked
5	R/W	0	MRMS6s	**0**: RMS upper limit interrupt on channel 6 is not masked **1**: RMS upper limit interrupt on channel 6 is masked
4	R/W	0	MRMS5s	**0**: RMS upper limit interrupt on channel 5 is not masked **1**: RMS upper limit interrupt on channel 5 is masked
3	R/W	0	MRMS4s	**0**: RMS upper limit interrupt on channel 4 is not masked **1**: RMS upper limit interrupt on channel 4 is masked
2	R/W	0	MRMS3s	**0**: RMS upper limit interrupt on channel 3 is not masked **1**: RMS upper limit interrupt on channel 3 is masked
1	R/W	0	MRMS2s	**0**: RMS upper limit interrupt on channel 2 is not masked **1**: RMS upper limit interrupt on channel 2 is masked
0	R/W	0	MRMS1s	**0:** RMS upper limit interrupt on channel 1 is not masked **1**: RMS upper limit interrupt on channel 1 is masked

5.2.42 Interrupt masking 2 configuration register (reg. 0x29)

B7	B6	B5	B4	B3	B2	B1	B0
MSTUCK 1_PDM4	MSTUCK 0_PDM4	MSTUCK 1_PDM3	MSTUCK 0_PDM3	MSTUCK 1_PDM2	MSTUCK 0_PDM2	MSTUCK 1_PDM1	MSTUCK 0_PDM1
0	0	0	0	0	0	0	0

Bit	R/W	RST	Name	Description
7	R/W	0	MSTUCK1_PDM4	**0**: PDM4 stuck at 1 event is not masked **1**: PDM4 stuck at 1 event is masked
6	R/W	0	MSTUCK0_PDM4	**0**: PDM4 stuck at 0 event is not masked **1**: PDM4 stuck at 0 event is masked
5	R/W	0	MSTUCK1_PDM3	**0**: PDM3 stuck at 1 event is not masked **1**: PDM3 stuck at 1 event is masked
4	R/W	0	MSTUCK0_PDM3	**0**: PDM3 stuck at 0 event is not masked **1**: PDM3 stuck at 0 event is masked
3	R/W	0	MSTUCK1_PDM2	**0**: PDM2 stuck at 1 event is not masked **1**: PDM2 stuck at 1 event is masked
2	R/W	0	MSTUCK0_PDM2	**0**: PDM2 stuck at 0 event is not masked **1**: PDM2 stuck at 0 event is masked
1	R/W	0	MSTUCK1_PDM1	**0**: PDM1 stuck at 1 event is not masked **1**: PDM1 stuck at 1 event is masked
0	R/W	0	MSTUCK0_PDM1	**0**: PDM1 stuck at 0 event is not masked **1**: PDM1 stuck at 0 event is masked

5.2.43 Interrupt masking 3 configuration register (reg. 0x2A)

B7	B6	B5	B4	B3	B2	B1	B0
MSTUCK 1_PDM8	MSTUCK 0_PDM8	MSTUCK 1_PDM7	MSTUCK 0_PDM7	MSTUCK 1_PDM6	MSTUCK 0_PDM6	MSTUCK 1_PDM5	MSTUCK 0_PDM5
0	0	0	0	0	0	0	0

Bit	R/W	RST	Name	Description
7	R/W	0	MSTUCK1_PDM8	**0**: PDM8 stuck at 1 event is not masked **1**: PDM8 stuck at 1 event is masked
6	R/W	0	MSTUCK0_PDM8	**0**: PDM8 stuck at 0 event is not masked **1**: PDM8 stuck at 0 event is masked
5	R/W	0	MSTUCK1_PDM7	**0**: PDM7 stuck at 1 event is not masked **1**: PDM7 stuck at 1 event is masked
4	R/W	0	MSTUCK0_PDM7	**0**: PDM7 stuck at 0 event is not masked **1**: PDM7 stuck at 0 event is masked
3	R/W	0	MSTUCK1_PDM6	**0**: PDM6 stuck at 1 event is not masked **1**: PDM6 stuck at 1 event is masked
2	R/W	0	MSTUCK0_PDM6	**0**: PDM6 stuck at 0 event is not masked **1**: PDM6 stuck at 0 event is masked
1	R/W	0	MSTUCK1_PDM5	**0**: PDM5 stuck at 1 event is not masked **1**: PDM5 stuck at 1 event is masked
0	R/W	0	MSTUCK0_PDM5	**0**: PDM5 stuck at 0 event is not masked **1**: PDM5 stuck at 0 event is masked

5.2.44 GPIO1 configuration register (reg. 0x2B)

B7	B6	B5	B4	B3	B2	B1	B0
-	-	-	-	-	-	LR_DIR	LR_OUT
-	-	-	-	-	-	1	0

Bit	R/W	RST	Name	Description
1	R/W	1	LR_DIR	**0**: LR pin used as output **1**: LR pin used as input
0	R/W	0	LR_OUT	Set/unset the LR value when the pin is used as an output

5.2.45 GPIO2 configuration register (reg. 0x2C)

B7	B6	B5	B4	B3	B2	B1	B0
-	-	-	-	-	-	-	LR_IN
-	-	-	-	-	-	-	0

Bit	R/W	RST	Name	Description
0	R	0	LR_IN	Read back the LR input value when the GPIO is used as an input

5.2.46 PLL and oscillator configuration register (reg. 0x2D)

B7	B6	B5	B4	B3	B2	B1	B0
-	-	-	-	-	-	-	PLL_OSC_PD
-	-	-	-	-	-	-	1

5.2.47 Sampling control register (reg.0x2E)

B7	B6	B5	B4	B3	B2	B1	B0
SMPM4B	SMPM4A	SMPM3B	SMPM3A	SMPM2B	SMPM2A	SMPM1B	SMPM1A
0	0	0	0	0	0	0	0

PDM sampling edge configuration in **advanced mode**(see reg. 0x2F).

Bit	R/W	RST	Name	Description
7	R/W	0	SMPM4B	**1:** CH4B sampled on rising **0:** CH4B sampled on falling
6	R/W	0	SMPM4A	**1:** CH4A sampled on rising **0:** CH4A sampled on falling
5	R/W	0	SMPM3B	**1:** CH3B sampled on rising **0:** CH3B sampled on falling
4	R/W	0	SMPM3A	**1:** CH3A sampled on rising **0:** CH3A sampled on falling
3	R/W	0	SMPM2B	**1:** CH2B sampled on rising **0:** CH2B sampled on falling
2	R/W	0	SMPM2A	**1:** CH2A sampled on rising **0:** CH2A sampled on falling
1	R/W	0	SMPM1B	**1:** CH1B sampled on rising **0:** CH1B sampled on falling
0	R/W	0	SMPM1A	**1:** CH1A sampled on rising **0:** CH1A sampled on falling

5.2.48 Sampling and STMRecomb control register (reg.0x2F)

B7	B6	B5	B4	B3	B2	B1	B0
-				RES_INT_MODE		M_RBYP	M_EN
-				00		1	1

Bit	R/W	RST	Name	Description
3-2	R/W	00	RES_INT_MODE	Resampling mode: **00:** Old **01:** Reserved **10:** Dual membrane **11:** Advanced (see 0x2E)
1	R/W	1	M_RBYP	**1:** STMRecomb bypassed (globally) **0:** STMRecomb NOT bypassed (globally)
0	R/W	1	M_EN	**1:** Microphone processor enabled **0:** Microphone processor disabled

5.2.49 STMRecomb. control register mike1 global gain (reg. 0x30)

B7	B6	B5	B4	B3	B2	B1	B0
RMCH_1	BYP RM1	Microphone 1 Global Gain (sensitivity compensation)					
1	0	100000					

Bit	R/W	RST	Name	Description
7	R/W	1	RMCH_1	Auxiliary channel selection: **0**: PDM1a **1**: PDM1b
6	R/W	0	BYP RM1	Mike1 recombination bypass: **1**: Recombination of Mike 1 is bypassed. Auxiliary ch. is forced to PDM1b. **0**: Recombination of Mike 1 is active
5-0	R/W	100000	Mike 1 Global Gain	See *Table 13: "Global gain"*

5.2.50 STMRecomb. control register mike2 global gain (reg. 0x31)

B7	B6	B5	B4	B3	B2	B1	B0
RMCH_2	BYB RM2	Microphone 2 Global Gain (sensitivity compensation)					
1	0	100000					

Bit	R/W	RST	Name	Description
7	R/W	1	RMCH_2	Auxiliary channel selection: **0**: PDM2a **1**: PDM2b
6	R/W	0	BYP RM2	Mike2 recombination bypass: **1**: Recombination of Mike 2 is bypassed. Auxiliary ch. is forced to PDM2b. **0**: Recombination of Mike 2 is active
5-0	R/W	100000	Mike 2 Global Gain	See *Table 13: "Global gain"*

5.2.51 STMRecomb. control register mike3 global gain (reg. 0x32)

B7	B6	B5	B4	B3	B2	B1	B0
RMCH_3	BYP RM3	Microphone 3 Global Gain (sensitivity compensation)					
1	0	100000					

Bit	R/W	RST	Name	Description
7	R/W	1	RMCH_3	Auxiliary channel selection: **0**: PDM3a **1**: PDM3b
6	R/W	0	BYP RM3	Mike3 recombination bypass: **1**: Recombination of Mike 3 is bypassed. Auxiliary ch. is forced to PDM3b. **0**: Recombination of Mike 3 is active
5-0	R/W	100000	Mike 3 Global Gain	See *Table 13: "Global gain"*

5.2.52 STMRecomb. control register mike4 global gain (reg. 0x33)

B7	B6	B5	B4	B3	B2	B1	B0
RMCH_4	Byp RM4	Microphone 4 Global Gain (sensitivity compensation)					
1	0	100000					

Bit	R/W	RST	Name	Description
7	R/W	1	RMCH_4	Auxiliary channel selection: **0**: PDM4a **1**: PDM4b
6	R/W	0	BYP RM4	Mike4 recombination bypass: **1**: Recombination of Mike 4 is bypassed. Auxiliary ch. is forced to PDM4b. **0**: Recombination of Mike 3 is active
5-0	R/W	100000	Mike 4 Global Gain	See *Table 13: "Global gain"*

Table 13: Global gain

Hex value	dB	Hex value	dB	Hex value	dB	Hex value	dB
0x00	-4.0	0x10	-2.0	0x20	0	0x30	2.0
0x01	-3.875	0x11	-1.875	0x21	0.125	0x31	2.125
0x02	-3.75	0x12	-1.75	0x22	0.250	0x32	2.250
0x03	-3.625	0x13	-1.625	0x23	0.375	0x33	2.375
0x04	-3.5	0x14	-1.5	0x24	0.5	0x34	2.5
0x05	-3.375	0x15	-1.375	0x25	0.625	0x35	2.625
0x06	-3.25	0x16	-1.25	0x26	0.750	0x36	2.750
0x07	-3.125	0x17	-1.125	0x27	0.875	0x37	2.875
0x08	-3.0	0x18	-1.0	0x28	1.0	0x38	3.0
0x09	-2.875	0x19	-1.875	0x29	1.125	0x39	3.125
0x0A	-2.75	0x1A	-1.75	0x2A	1.250	0x3A	3.250
0x0B	-2.625	0x1B	-1.625	0x2B	1.375	0x3B	3.375
0x0C	-2.5	0x1C	-1.5	0x2C	1.5	0x3C	3.5
0x0D	-2.375	0x1D	-1.375	0x2D	1.625	0x3D	3.625
0x0E	-2.25	0x1E	-1.25	0x2E	1.750	0x3E	3.750
0x0F	-2.125	0x1F	-1.125	0x2F	1.875	0x3F	3.875

5.2.53 STMRecomb. control register mike 1 attenuation (reg. 0x34)

B7	B6	B5	B4	B3	B2	B1	B0
-	LP1 EN	Microphone 1 Low Channel Attenuation					
-	1	100000					

Bit	R/W	RST	Name	Description
6	R/W	1	LP1 EN	Low-pass filter enable: **0**: L.P. filter is not enabled **1**: L.P. filter is enabled
5-0	R/W	100000	Microphone 1 Low Channel Attenuation	See *Table 14: "Channel attenuation"*

5.2.54 Recomb. control register mike 2 attenuation (reg. 0x35)

B7	B6	B5	B4	B3	B2	B1	B0
-	LP2 EN	Microphone 2 Low Channel Attenuation					
-	1	100000					

Bit	R/W	RST	Name	Description
6	R/W	1	LP2 EN	Low pass filter enable: **0**: L.P. filter is not enabled **1**: L.P. filter is enabled
5-0	R/W	100000	Microphone 2 Low Channel Attenuation	See *Table 14: "Channel attenuation"*

5.2.55 STMRecomb. control register mike 3 attenuation (reg. 0x36)

B7	B6	B5	B4	B3	B2	B1	B0
-	LP3 EN	Microphone 3 Low Channel Attenuation					
-	1	100000					

Bit	R/W	RST	Name	Description
6	R/W	1	LP3 EN	Low pass filter enable: **0**: L.P. filter is not enabled **1**: L.P. filter is enabled
5-0	R/W	100000	Microphone 3 Low Channel Attenuation	See *Table 14: "Channel attenuation"*

5.2.56 STMRecomb. control register mike 4 attenuation (reg. 0x37)

B7	B6	B5	B4	B3	B2	B1	B0
-	LP4 EN	Microphone 4 Low Channel Attenuation					
-	1	100000					

Bit	R/W	RST	Name	Description
6	R/W	1	LP4 EN	Low-pass filter enable: **0**: L.P. filter is not enabled **1**: L.P. filter is enabled
5-0	R/W	100000	Microphone 4 Low Channel Attenuation	See *Table 14: "Channel attenuation"*

Table 14: Channel attenuation

Hex value	dB	Hex value	dB	Hex value	dB	Hex value	dB
0x00	-4.0	0x10	-2.0	0x20	0	0x30	2.0
0x01	-3.875	0x11	-1.875	0x21	0.125	0x31	2.125
0x02	-3.75	0x12	-1.75	0x22	0.250	0x32	2.250
0x03	-3.625	0x13	-1.625	0x23	0.375	0x33	2.375
0x04	-3.5	0x14	-1.5	0x24	0.5	0x34	2.5
0x05	-3.375	0x15	-1.375	0x25	0.625	0x35	2.625
0x06	-3.25	0x16	-1.25	0x26	0.750	0x36	2.750
0x07	-3.125	0x17	-1.125	0x27	0.875	0x37	2.875
0x08	-3.0	0x18	-1.0	0x28	1.0	0x38	3.0
0x09	-2.875	0x19	-1.875	0x29	1.125	0x39	3.125
0x0A	-2.75	0x1A	-1.75	0x2A	1.250	0x3A	3.250
0x0B	-2.625	0x1B	-1.625	0x2B	1.375	0x3B	3.375
0x0C	-2.5	0x1C	-1.5	0x2C	1.5	0x3C	3.5
0x0D	-2.375	0x1D	-1.375	0x2D	1.625	0x3D	3.625
0x0E	-2.25	0x1E	-1.25	0x2E	1.750	0x3E	3.750
0x0F	-2.125	0x1F	-1.125	0x2F	1.875	0x3F	3.875

5.2.57 STMRecomb. control register mike 1 low channel threshold (reg. 0x38)

B7	B6	B5	B4	B3	B2	B1	B0
-	-	Microphone 1 Low Channel Threshold					
-	-	110011					

Bit	R/W	RST	Name	Description
5-0	R/W	110011	Microphone 1 Low Channel Threshold	See *Table 15: "Channel thresholds"*

5.2.58 STMRecomb. control register mike 2 low channel threshold (reg. 0x39)

B7	B6	B5	B4	B3	B2	B1	B0
-	-	Microphone 2 Low Channel Threshold					
-	-	110011					

Bit	R/W	RST	Name	Description
5-0	R/W	110011	Microphone 2 Low Channel Threshold	See *Table 15: "Channel thresholds"*

5.2.59 STMRecomb. control register mike 3 low channel threshold (reg. 0x3A)

B7	B6	B5	B4	B3	B2	B1	B0
-	-	Microphone 3 Low Channel Threshold					
-	-	110011					

Bit	R/W	RST	Name	Description
5-0	R/W	110011	Microphone 3 Low Channel Threshold	See *Table 15: "Channel thresholds"*

5.2.60 STMRecomb. control register mike 4 low channel threshold (reg. 0x3B)

B7	B6	B5	B4	B3	B2	B1	B0
-	-	Microphone 4 Low Channel Threshold					
-	-	110011					

Bit	R/W	RST	Name	Description
5-0	R/W	110011	Microphone 4 Low Channel Threshold	See *Table 15: "Channel thresholds"*

5.2.61 STMRecomb. control register mike 1 high channel threshold (reg.0x3C)

B7	B6	B5	B4	B3	B2	B1	B0
-	-	Microphone 1 High Channel Threshold					
-	-	011011					

Bit	R/W	RST	Name	Description
5-0	R/W	011011	Microphone 1 High Channel Threshold	See *Table 15: "Channel thresholds"*

5.2.62 STMRecomb. control register mike 2 high channel threshold (reg.0x3D)

B7	B6	B5	B4	B3	B2	B1	B0
-	-	Microphone 2 High Channel Threshold					
-	-	011011					

Bit	R/W	RST	Name	Description
5-0	R/W	011011	Microphone 2 High Channel Threshold	See *Table 15: "Channel thresholds"*

5.2.63 STMRecomb. control register mike 3 high channel threshold (reg.0x3E)

B7	B6	B5	B4	B3	B2	B1	B0
-	-	Microphone 3 High Channel Threshold					
-	-	011011					

Bit	R/W	RST	Name	Description
5-0	R/W	011011	Microphone 3 High Channel Threshold	See *Table 15: "Channel thresholds"*

5.2.64 STMRecomb. control register mike 4 high channel threshold (reg.0x3F)

B7	B6	B5	B4	B3	B2	B1	B0
-	-	Microphone 4 High Channel Threshold					
-	-	011011					

Bit	R/W	RST	Name	Description
5-0	R/W	011011	Microphone 4 High Channel Threshold	See *Table 15: "Channel thresholds"*

Table 15: Channel thresholds

Hex value	dB	Hex value	dB	Hex value	dB	Hex value	dB
0x00	0	0x10	-16	0x20	-32	0x30	-48
0x01	-1	0x11	-17	0x21	-33	0x31	-49
0x02	-2	0x12	-18	0x22	-34	0x32	-50
0x03	-3	0x13	-19	0x23	-35	0x33	-51
0x04	-4	0x14	-20	0x24	-36	0x34	-52
0x05	-5	0x15	-21	0x25	-37	0x35	-53
0x06	-6	0x16	-22	0x26	-38	0x36	-54
0x07	-7	0x17	-23	0x27	-39	0x37	-55
0x08	-8	0x18	-24	0x28	-40	0x38	-56
0x09	-9	0x19	-25	0x29	-41	0x39	-57
0x0A	-10	0x1A	-26	0x2A	-42	0x3A	-58
0x0B	-11	0x1B	-27	0x2B	-43	0x3B	-59
0x0C	-12	0x1C	-28	0x2C	-44	0x3C	-60
0x0D	-13	0x1D	-29	0x2D	-45	0x3D	-61
0x0E	-14	0x1E	-30	0x2E	-46	0x3E	-62
0x0F	-15	0x1F	-31	0x2F	-47	0x3F	-63

Note: The threshold values refer to the internal microphone recombination block, after the -6 dB attenuation. In order to get the threshold with respect to the input signal, add 6 dB.

5.2.65 Boost 6 dB control register (reg. 0x40)

B7	B6	B5	B4	B3	B2	B1	B0
CH8	CH7	CH6	CH5	CH4	CH3	CH2	CH1
0	0	0	0	0	0	0	0

Table 16: Enable boost 6 dB (x2) on selected channel

Bit	R/W	RST	Name	Description
7	R/W	0	CH8	**0**: - **1**: CH8 x 2
6	R/W	0	CH7	**0**: - **1**: CH7 x 2
5	R/W	0	CH6	**0**: - **1**: CH6 x 2
4	R/W	0	CH5	**0**: - **1**: CH5 x 2
3	R/W	0	CH4	**0**: - **1**: CH4 x 2
2	R/W	0	CH3	**0**: - **1**: CH3 x 2
1	R/W	0	CH2	**0**: - **1**: CH2 x 2
0	R/W	0	CH1	**0**: - **1**: CH1 x 2

6 Package information

In order 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.

6.1 VFQFPN 28L package information

Figure 23: VFQFPN (5 x 5 x 1.0 mm) 28L pitch 0.50 package outline

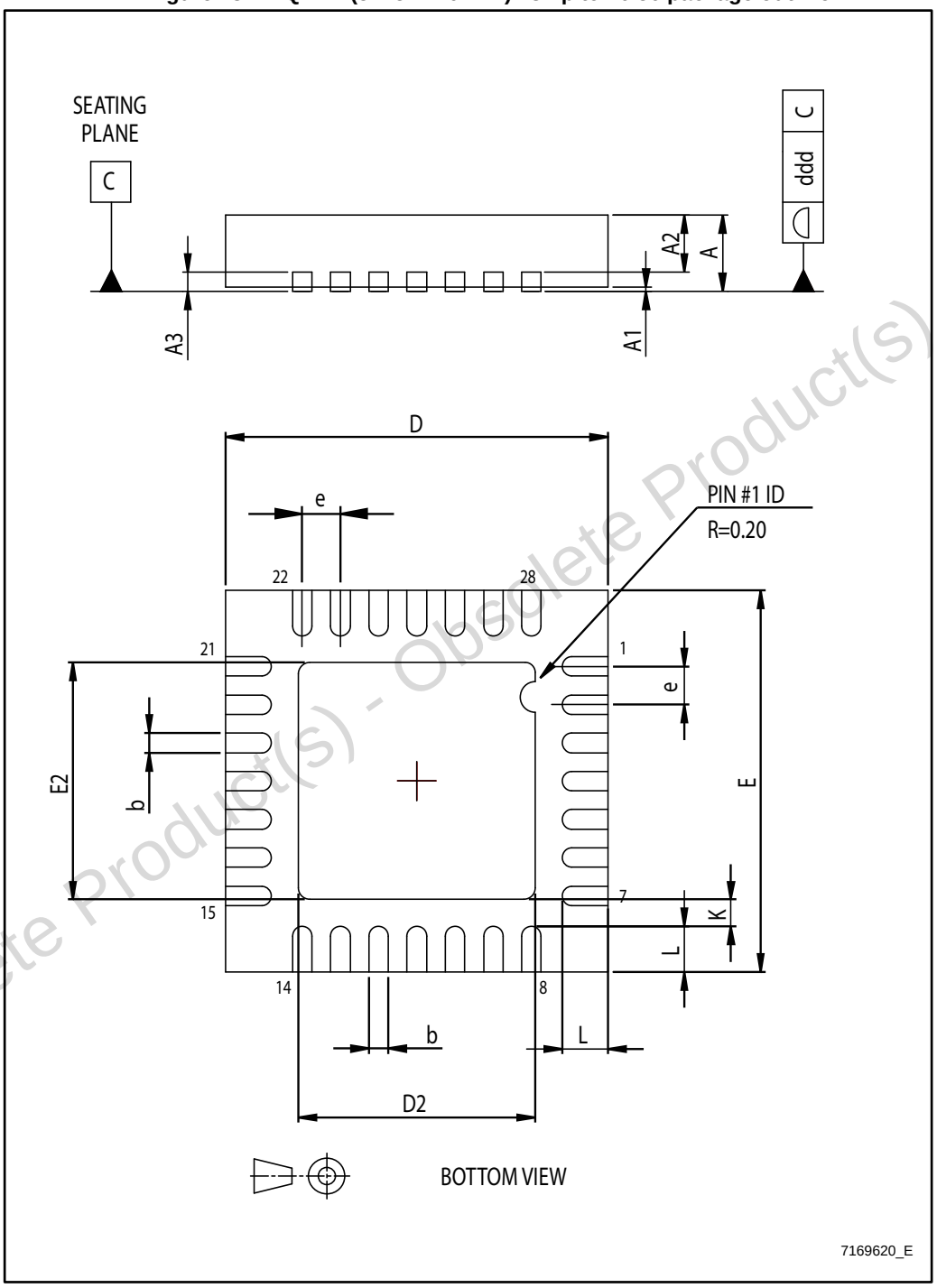


Table 17: VFQFPN (5 x 5 x 1.0 mm) 28L package mechanical data

Symbol	mm.		
	Min.	**Typ.**	**Max**
Symbol	**mm.**		
	Min.	**Typ.**	**Max.**
A	0.800	0.850	1.000
A1		0.010	0.050
A2		0.650	0.800
A3		0.200	
b	0.204	0.254	0.300
D	4.900	5.000	5.100
D2	SEE EXPOSED PAD VARIATIONS		
E	4.900	5.000	5.100
E2	SEE EXPOSED PAD VARIATIONS		
e	0.450	0.500	0.550
L	0.500	0.600	0.750
K	0.25		
ddd			0.050
VARIATION	EXPOSED PAD VARIATIONS		
	E2		
A	2.950	3.100	3.250
B	2.550	2.700	2.850

7 Revision history

Table 18: Document revision history

Date	Revision	Changes
28-Apr-2016	1	Initial release

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.