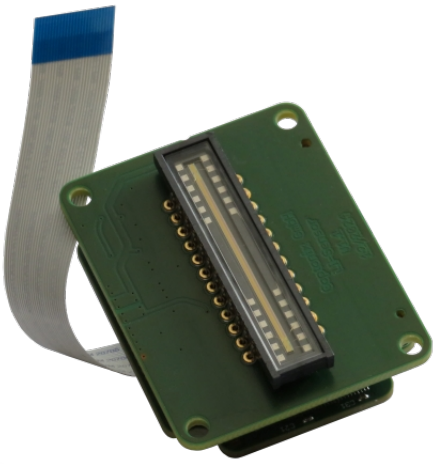
Linear Image Sensor Module
Parallel Interface



Coptonix GmbH
Falkentaler Steig 9
13467 Berlin, Germany
+49 30 61741248
support@coptonix.com

Contents

Key Features Overview

- 3.3V to 5V external power supply
- Supports various CMOS linear image sensors
- **Parallel Interface**
  - 3.3V logic level
  - 8-bit data bus
  - Multiplexed byte-wide output (8+8 format)
  - Operates in master mode
  - Synchronization signals
    - Data Clock with configurable frequency and polarity
    - Line Valid signal with configurable pixel count and polarity
    - Frame Valid signal with configurable line count and polarity
- **I2C Interface**
  - 3.3 V logic, 5 V tolerant
  - Operates as I2C Slave
  - Supports speeds up to 400 kHz (Fast Mode)
  - Programmable I2C address
  - Read/Write access to internal control and configure registers
- **High-Speed ADC**
  - 16-bit resolution
  - Programmable gain: 1x to 5.85x in 64 steps
  - Programmable offset: ±250 mV in 512 steps
  - Two selectable internal full scale ranges: 1.6 V and 2.0 V
  - Programmable reference voltage: 0 V to 2.0 V in 1024 steps
- **Operation Modes**
  - Free-running mode
  - Internal software trigger with 32 bit configurable period, adjustable in 1µs steps
  - External trigger mode
- **Trigger Input**
  - 3.3V logic level
  - Two inputs channels
  - Supported trigger modes:
    - Rising-edge trigger
      → Single-line acquisition per event
    - Rising-edge trigger
      → Multiple-lines acquisition / one complete frame per event
      → Configurable number of lines per frame per event
    - High-level sensitive trigger
      → Single or multiple line acquisition per event
      → Number of lines depends on trigger pulse width
    - 32 Bit quadrature encoder-based triggering
      → Trigger direction selectable: UP (CW) or DOWN (CCW) direction
      → Single-shot acquisition triggered by any UP or DOWN count events
      → Optional trigger after a predefined number of UP or DOWN events
- **Trigger Output**
  - 3.3V logic level
  - Single output channel with configurable polarity
  - 32 Bit configurable delay, adjustable in 1µs steps
  - 32 Bit configurable width, adjustable in 1µs steps
- **Connector**
  - 0.50 mm pitch FPC ZIF Connector, SMT Horizontal - Hinge Type
  - 26 pins
  - Würth Elektronik, part number 687126149022
  - Compatible FFC cable: Würth Elektronik 687626100002
- **Mechanical**
  - Ultra compact design 48 x 38 x 7.2 mm³ (two stacked PCBs, sensor not included)
  - 3D-Step file available

Overview

The LISM-PI26xx, developed by Coptonix GmbH, is a sophisticated board-level linear image sensor module designed to accommodate a wide range of CMOS linear image sensors, offering unparalleled versatility and precision in various industrial and scientific imaging applications.

Key Features and Functionalities

- **Broad Sensor Compatibility:** Capable of supporting an extensive array of CMOS linear image sensors with varying pixel counts, the LISM-PI26xx can be tailored to meet diverse imaging needs.
- **Advanced Programmability:** This module is easily programmable through a two-wire serial I2C Bus interface, allowing users to customize settings such as the number of pixels per line, number of lines per frame, line rate, exposure, gain, offset, and trigger configurations.
- **Precise Timing and Control:** The module autonomously generates all necessary internal timing and system control logic, including a pixel clock. This ensures synchronous streaming of pixel data, accurately marked by LINE_VALID and FRAME_VALID signals.
- **High-Resolution Data Output:** It outputs 16 bits of data per pixel across an 8-bit parallel interface in two segments of 8 bits each, ensuring detailed image data capture and processing.
- **Flexible Connectivity Options:** Equipped with a 26-pin FPC 0.5mm SMT ZIF Horizontal connector, it offers seamless integration with microcontrollers or FPGAs via an FFC cable.

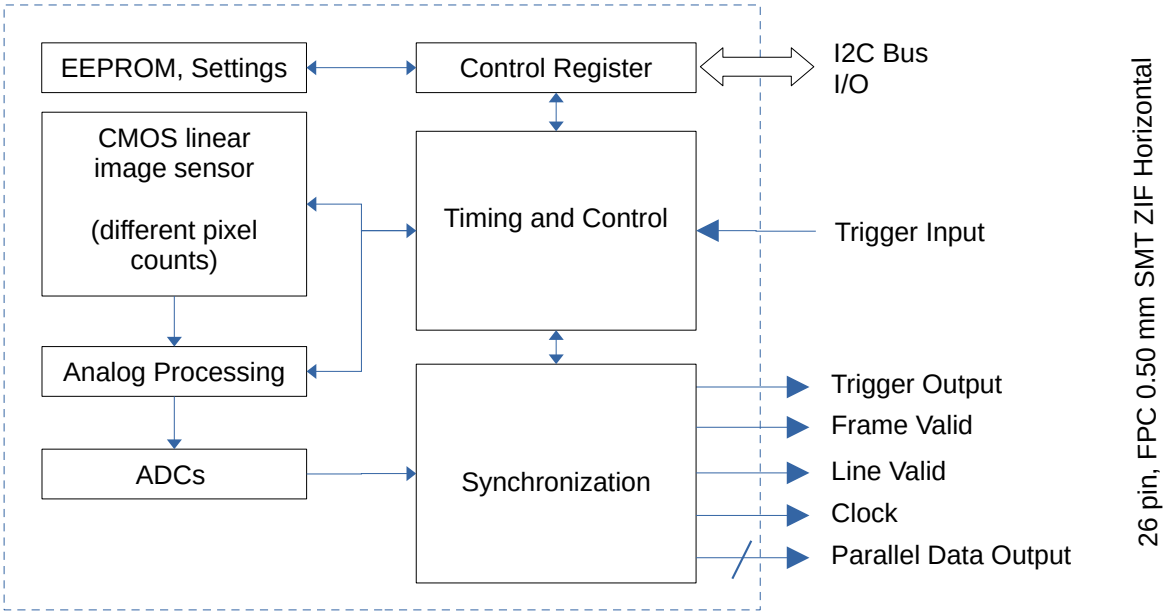
Modular and Adaptable Design

- **Dual Board Structure:** The module consists of a stack of two circuit boards - the processor board and the sensor board - with dimensions of 48mm x 38mm and a height of 7.2mm. This compact design facilitates easy integration into various setups without spatial constraints.
- **Fully Interchangeable Sensor Boards:** The sensor board includes pin receptacles that not only facilitate the quick swapping of pin-compatible sensors but also allow for the entire sensor board to be replaced. This capability supports the use of sensors that are not pin-compatible, providing flexibility to adapt to various specifications and new sensor technologies without requiring a complete system overhaul.

Ideal for Diverse Applications

- The module's high adaptability and precision make it an excellent choice for applications such as industrial automation, scientific research, and advanced imaging systems where high accuracy and customization are critical.

LISM-PI26xx block diagram



Characteristics

	Min.	Typ	Max.	Unit
Power-Supply				
Supply Voltage	3.3		5.0	V
Supply Current (without linear image sensor)			155	mA
Supply Current (with S11639-01 linear image sensor)			215	mA
I2C-Bus pins (5V tolerant I/O pins)				
V_{IH} High-level Input Voltage	2.3			V
V_{IL} Low-level Input Voltage			1.0	V
Output Voltage	0		3.3	V
External Trigger Inputs				
V_{IH} High-level Input Voltage	2.3			V
V_{IL} Low-level Input Voltage			1.6	V
Outputs				
Output Voltage	0		3.3	V
Limiting values				
I2C-Bus pins (5V tolerant I/O pins)				
Input Voltage	-0.5		5.0	V
External Trigger Inputs				
Input Voltage	0		3.3	V
Power-Supply				
Supply Voltage	3.3		5.0	V
Temperature				
Operating temperature	0		+70	°C

Timing Characteristics

Parameter	Symbol	Min.	Typ	Max.	Unit
Pixel Clock	f_{pclk}	0.2		10	MHz
Data Clock	f_{dclk}	0.4		20	MHz
Data Clock – Duty Cycle		45	50	55	%
Data Clock Falling to Frame Valid Rising	t_1		9		ns
Frame Valid Rising to Data Clock Rising	t_2		$(1/f_{dclk})/2 - t_1$		s
Data Clock Falling to Line Valid Rising	t_3		9		ns
Line Valid Rising to Data Clock Rising	t_4		$(1/f_{dclk})/2 - t_3$		s
Data Clock Falling to Data Output (Delay)	t_5		0	10	ns
Data output to Data Clock Rising (Setup)	t_6		$(1/f_{dclk})/2 - t_5$		s
Data Clock Falling to Line Valid Falling	t_7		8		ns
Data Clock Falling to Frame Valid Falling	t_8		8		ns
Frame Valid Rising to Line Valid Rising	t_9		$(1/f_{dclk})$ x 10		s
Line Valid Falling to Frame Valid Falling	t_{10}		$(1/f_{dclk})$ x 10		s
External Trigger Rising to Sensor Start Rising	t_{11}			220	ns
I2C frequency	f_{I2C}			400	kHz

Connector Pinout

Pin No.	Mnemonic	Type	Description
1	GND	P	Ground
2	EXT_TRIG_IN2	I	Quadrature Encoder Channel B
3	GND	P	Ground
4	EXT_TRIG_IN1	I	External Trigger Input / Quadrature Encoder Channel A
5	GND	P	Ground
6	TRIG_OUT	O	Trigger Output
7	LV	O	Line Valid Signal
8	FV	O	Frame Valid Signal
9	O1	O	Output. Not used
10	GND	P	Ground
11	DCLK	O	Data Clock
12	GND	P	Ground
13	D15 / D7	O	Data Bit 15 and Bit 7
14	D14 / D6	O	Data Bit 14 and Bit 6
15	D13 / D5	O	Data Bit 13 and Bit 5
16	D12 / D4	O	Data Bit 12 and Bit 4
17	D11 / D3	O	Data Bit 11 and Bit 3
18	D10 / D2	O	Data Bit 10 and Bit 2
19	D9 / D1	O	Data Bit 9 and Bit 1
20	D8 / D0	O	Data Bit 8 and Bit 0
21	GND	P	Ground
22	SDA	IO	I2C Serial Data Input/Output
23	SCL	I	I2C Serial Clock
24	GND	P	Ground
25	PWR	P	+3.3V to +5V Power Supply
26	PWR	P	+3.3V to +5V Power Supply

P = Power, I = Input, O = Output, IO = Input/Output

Pixel Data Output and Synchronization Signals

The LISM-PI26xx module outputs pixel data and synchronization signals with precise timing to ensure accurate and effective image capture and processing:

- **Pixel Data Output**
  - **Data Format:** Pixel data from the module is formatted in 16-bit values, which are divided and transmitted as two 8-bit segments: the high byte and the low byte.
  - **Output Sequence:** The high byte is output first, followed by the low byte.
  - **Clock Frequency and Polarity Configuration:** The polarity and frequency of the data clock are now configurable via the DATA_IF_CONFIG register:

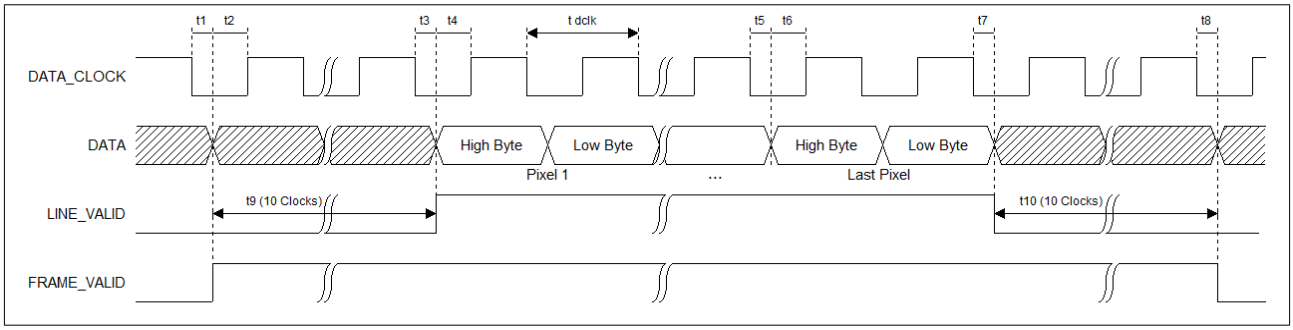
    Frequency Settings (Bits [2:0]):
    0 : Data Clock at 400 kHz, Pixel Clock at 200 kHz
    1 : Data Clock at 500 kHz, Pixel Clock at 250 kHz
    2 : Data Clock at 800 kHz, Pixel Clock at 400 kHz
    3 : Data Clock at 1 MHz, Pixel Clock at 500 kHz
    4 : Data Clock at 2 MHz, Pixel Clock at 1 MHz
    5 : Data Clock at 4 MHz, Pixel Clock at 2 MHz
    6 : Data Clock at 10 MHz, Pixel Clock at 5 MHz
    7 : Data Clock at 20 MHz, Pixel Clock at 10 MHz

    Polarity (Bit [3]):
    0 : The pixel data is output on the falling edge of the data clock (default setting).
    1 : The polarity of the data clock output is inverted, resulting in data output on the rising edge of the clock.

  - **Data Reading:** An external processor must adapt its reading edge based on the clock polarity:
    For Bit[3] = 0, then processor should read the data on the rising edge of the data clock.
    For Bit[3] = 1, then processor should read the data on the falling edge of the data clock.

- **Synchronization Signals**
  - **Signal Types:** Two key synchronization signals, FRAME VALID (FV) and LINE VALID (LV), are provided alongside the data clock to manage the timing and integrity of data transmission.
  - **LINE VALID (LV):** This signal is set HIGH during the period that pixel data is being output. It indicates when the data on the data line is valid and ready to be read.
  - **FRAME VALID (FV):** The FRAME VALID signal is activated HIGH ten clock cycles before the LINE VALID signal goes HIGH. It stays HIGH throughout the duration in which the required number of lines specified in the LINES_PER_FRAME register are captured. The FRAME VALID signal then goes LOW ten clock cycles after the LINE VALID signal has been set to LOW, marking the end of a frame.

  These synchronization features ensure that each frame and line of data are clearly delineated and correctly processed by any external systems connected to the LISM_PI26xx. The precise control over signal timing aids in the seamless integration of the module into a variety of imaging applications, where data integrity and accurate synchronization are critical. The polarity of both FRAME VALID and LINE VALID signals can be configured via the DATA_IF_CONFIG register.



I2C Bus

The LISM-PI26xx is equipped with an I2C interface supporting two primary operational modes:

Slave Receiver Mode (Write Mode):

In this mode, the LISM-PI26xx functions as a receiver while the I2C master acts as the transmitter. This setup enables the I2C master to configure settings on the LISM-PI26xx by transmitting zero or more bytes of data, which the LISM-PI26xx acknowledges upon receipt. Data can include 0, 1, 2, or 4 bytes, in addition to the register address, depending on the specific register's size.

Certain registers may not require data; upon receiving the register address, the LISM-PI26xx performs the associated function without additional data input.

S	Slave Address	W	A	Reg. [7:0]	A	P

STORE_SETTINGS = 0xF2

S	1111 111	0	A	1111 0010	A	P
	0xFE			0xF2		

1 byte Register:

S	Slave Address	W	A	Reg. [7:0]	A	d [7:0]	A	P

MODE_CONFIG = 0x03 , data = 0x34 (Quadrature encoder trigger, Up Counter, Counter enabled)

S	1111 111	0	A	0000 0011	A	0011 0100	A	P
	0xFE			0x03		0x34		

2 bytes Register, MSB first:

S	Slave Address	W	A	Reg. [7:0]	A	d [15:8]	A	d [7:0]	A	P

LINES_PER_FRAME = 0x06, data = 0x0400 (1024 lines per frame)

S	1111 111	0	A	0000 0110	A	0000 0100	A	0000 0000	A	P
	0xFE			0x06		0x04		0x00		

4 bytes Register, MSB first:

S	Slave Address	W	A	Reg. [7:0]	A	d [31:24]	A	d [23:16]	A	d [15:8]	A	d [7:0]	A	P

SOFT_TRIGGER_PERIOD = 0x08, data = 0x000003E8 (1000 µs)

S	1111 111	0	A	0000 1000	A	0000 0000	A	0000 0000	A	0000 0011	A	1110 1000	A	P
	0xFE			0x08		0x00		0x00		0x03		0xE8		

S - Start

P - Stop

W - Direction bit (WRITE)

R - Direction bit (READ)

A - Acknowlege (ACK)

$\overline{A}$ - Not Acknowlege (NACK)

Master to Slave

Slave to Master

Slave Transmitter Mode (Read Mode):

In this mode, the roles reverse with the LISM-PI26xx serving as the transmitter and the I2C master as the receiver. This allows the I2C master to access and read the register contents of the LISM-PI26xx. The master can retrieve varying amounts of data – 1, 2, or 4 bytes, based on the register size – acknowledging each byte received. To initiate a read operation, the I2C master first writes the address of the register from which data is to be read. Following this, the LISM-PI26xx begins transmitting the data from the specified register. It's important to note that if the register address is not specified prior to reading, the resulting data will be undefined.

I2C master writes the register address from which the needs to be read:

S	1111 111	W	A	cmd [7:0]	A	P

LISM-PI26xx transmits data that is located in the previously set register address:
1 byte register

S	1111 111	R	A	d [7:0]	$\overline{A}$	P

2 bytes register

S	1111 111	R	A	d [15:8]	A	d [7:0]	$\overline{A}$	P

4 bytes register

S	1111 111	R	A	d [31:24]	A	d [23:16]	A	d [15:8]	A	d [7:0]	$\overline{A}$	P

Example:
To read the SOFT_TRIGGER_PERIOD register, first write the register address 0x08:

S	1111 111	0	A	0000 1000	A	P
	0xFE			0x08		

LISM-PI26xx outputs 4 bytes of data, the contents of the previously set register:

S	1111 111	1	A	0000 0000	A	0000 0000	A	0000 0011	A	1110 1000	$\overline{A}$	P
	0xFF			0x00		0x00		0x03		0xE8		

S - Start
P - Stop
W - Direction bit (WRITE)
R - Direction bit (READ)
A - Acknowlege (ACK)
$\overline{A}$ - Not Acknowlege (NACK)

Master to Slave
Slave to Master

Note on Command Processing Time:

Some I2C commands require internal processing before a new command can be accepted. If the I2C master attempts to send a new read or write command during this time, the LISM-PI26xx will respond with a NAK (No Acknowledge). In such cases, the master should retry the operation after a short delay. This ensures reliable and sequential command execution.

I2C Commands

Name	Hex	Bytes	W/R	Description
RESUME_GENERATOR	0x00	1	W/R	Resume / Suspend clock and timing generator
UPDATE_PARAMS	0x01	0	W	Read and update parameters of clock and timing generator
START_ACQUISITION	0x02	1	W/R	Start / Stop Acquisition
MODE_CONFIG	0x03	1	W/R	Operation mode and configuration Bit[2:0] Operation mode: 0x00: Free Running 0x01: External rising edge trigger (Single-Line) 0x02: External high level trigger 0x03: Internal software trigger 0x04: Quadrature encoder trigger 0x05: External rising edge trigger (Multi-Line) Bit[3] Not used, reserved for future use Bit[5:4] Quadrature Encoder Config: Bit[4]: 0 = Up Counter (CW); 1 = Down Counter (CCW) Bit[5]: 0 = Counter disabled; 1 = Counter enabled Bit[7:6] Not used, reserved for future use
START_PULSE_HIGH	0x04	4	W/R	High period of the start pulse DATA_IF_CONFIG[2:0] = 0x00 → 5 µs steps DATA_IF_CONFIG[2:0] = 0x01 → 4 µs steps DATA_IF_CONFIG[2:0] = 0x02 → 2.5 µs steps DATA_IF_CONFIG[2:0] = 0x03 → 2 µs steps DATA_IF_CONFIG[2:0] = 0x04 → 1 µs steps DATA_IF_CONFIG[2:0] = 0x05 → 500 ns steps DATA_IF_CONFIG[2:0] = 0x06 → 200 ns steps DATA_IF_CONFIG[2:0] = 0x07 → 100 ns steps
START_PULSE_LOW	0x05	4	W/R	Low period of the start pulse see START_PULSE_HIGH
LINES_PER_FRAME	0x06	2	W/R	Number of lines to complete one frame
QUAD_COUNT	0x07	4	W/R	Number of quadrature encoder events to capture a single line
SOFT_TRIGGER_PERIOD	0x08	4	W/R	Internal trigger time period [1 µs steps]
TRIGGER_OUTPUT_DELAY	0x09	4	W/R	Trigger output pulse delay [1 µs steps]
TRIGGER_OUTPUT_WIDTH	0x0A	4	W/R	Trigger output pulse width [1 µs steps]
PIXEL_COUNT	0x0B	2	W/R	Number of pixels per line
EDGES_BEFORE_DATA	0x0C	1	W/R	Number of cycles / positive edges before data output
DATA_IF_CONFIG	0x0D	1	W/R	Parallel data interface configurations Bit[2:0]: 0x00 = Data Clock at 400 kHz; Pixel Clock at 200 kHz 0x01 = Data Clock at 500 kHz; Pixel Clock at 250 kHz 0x02 = Data Clock at 800 kHz; Pixel Clock at 400 kHz 0x03 = Data Clock at 1 MHz; Pixel Clock at 500 kHz 0x04 = Data Clock at 2 MHz; Pixel Clock at 1 MHz 0x05 = Data Clock at 4 MHz; Pixel Clock at 2 MHz 0x06 = Data Clock at 10 MHz; Pixel Clock at 5 MHz 0x07 = Data Clock at 20 MHz; Pixel Clock at 10 MHz

				Bit[3]: Data clock polarity 0 = Read on Positve Edge; 1 = Read on Negative Edge Bit[4]: Line Valid Signal 0 = High Active; 1 = Low Active Bit[5]: Frame Valid Signal 0 = High Active; 1 = Low Active Bit[6]: Trigger Output 0 = High Active; 1 = Low Active Bit[7]: Not used, reserved for future use
ADC_FULL_SCALE	0x20	1	W/R	Full scale of the ADC (1.6V or 2V)
ADC_GAIN	0x21	1	W/R	1~5.85x ADC gain, 64 increments
ADC_OFFSET	0x22	2	W/R	±250 mV ADC offset, 512 increments
ADC_EXT_REF	0x30	2	W/R	0V to 2V external reference voltage, 1024 increments
INIT_STATUS	0xEE	1	R	Initialization status of the clock and timing generator
COM_RESULT	0xEF	1	R	Result of last communication with clock and timing generator
TEMP_SLAVE_ADDRESS	0xF0	1	W/R	Set the temporary slave address to be saved later into eeprom
STORE_SLAVE_ADDRESS	0xF1	0	W	Store temporary slave address into eeprom
STORE_SETTINGS	0xF2	0	W	Store settings into eeprom
RELOAD_SETTINGS	0xF3	0	W	Read settings from eeprom and initialize hardware
RESET_SETTINGS	0xF4	0	W	Reset settings to default
SENSOR_BOARD_ID	0xFA	2	R	Sensor Board Identifier
SB_HW_VERSION	0xFB	2	R	Sensor Board Hardware Version
PB_HW_VERSION	0xFC	2	R	Processor Board Hardware Version
P1_FM_VERSION	0xFD	2	R	Firmware version of MCU
P2_FM_VERSION	0xFE	2	R	Firmware version of the clock and timing generator
HW_RESET	0xFF	0	W	Hardware reset of the clock and timing generator

RESUME_GENERATOR (0x00), Write/Read, 8-bit data

The RESUME_GENERATOR register controls the operation of the clock and timing generator in the LISM-PI26xx. To resume the generation of clock and control signals, write the value 0xAA to this register. Writing any other value will suspend the generator, placing it in a reset state where no signals are produced. The suspension does not impact the I2C interface; thus, all other registers remain accessible for both writing and reading during this period. This functionality allows the generator to be restarted later with its settings preserved, ensuring a return to a known operational state.

UPDATE_PARAMS (0x01), no data

The UPDATE_PARAMS command synchronizes the operational parameters of the clock and timing generator with those set in the control registers. Upon execution, this command refreshes the contents of all related registers to ensure that they reflect the current configurations accurately and are up-to-date for the I2C master to read. This ensures consistent performance and accuracy of the generator's settings.

START_ACQUISITION (0x02), Write/Read, 8-bit data

The START_ACQUISITION command is used to initiate the pixel data acquisition process. Writing a value of 0x01 to this register activates the timing and control generator, which in turn starts the pulse signal necessary for the linear image sensor to begin capturing data. To halt data acquisition, write a value of 0x00 to this register, which stops the generation of the start pulse signal and effectively ends the acquisition process.

MODE_CONFIG (0x03), Write/Read, 8 Bit data

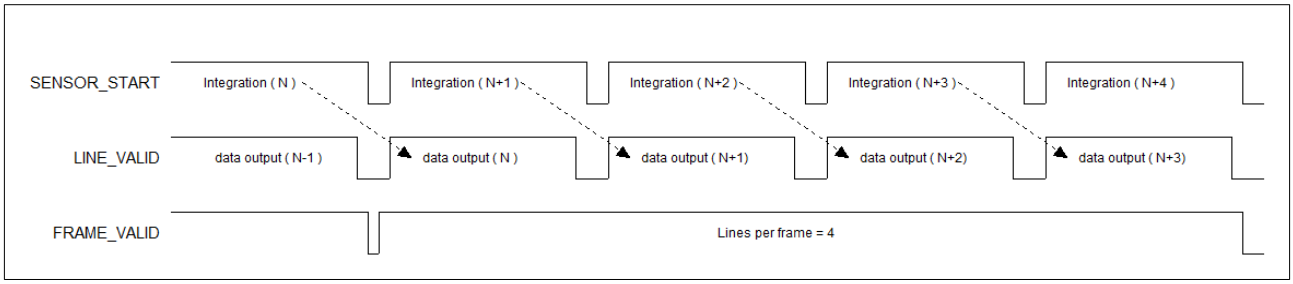
The MODE_CONFIG register in the LISM-PI26xx module is crucial for setting the operational mode and configuring the trigger input from the Quadrature Encoder and the data clock polarity. This register uses its 8 bits effectively to control different functionalities:

Bits	Value	Description
[2:0]	0x00	Free Running
	0x01	External rising edge trigger (Single-Line)
	0x02	External high level trigger
	0x03	Internal software trigger
	0x04	Quadrature encoder trigger
	0x05	External rising edge trigger (Multi-Line)
	0x06	Not used, default: Free Running
	0x07	Not used, default: Free Running
[3]	*x*	*Not used, reserved for future use*
[4]	0	Quadrature encoder configuration Up Counter (CW)
	1	Quadrature encoder configuration Down Counter /CCW)
[5]	0	Quadrature encoder configuration Counter disabled
	1	Quadrature encoder configuration Counter enabled
[7:6]	*x*	*Not used, reserved for future use*

Operating Mode Configuration:

Free Running

The module operates continuously without external triggers.

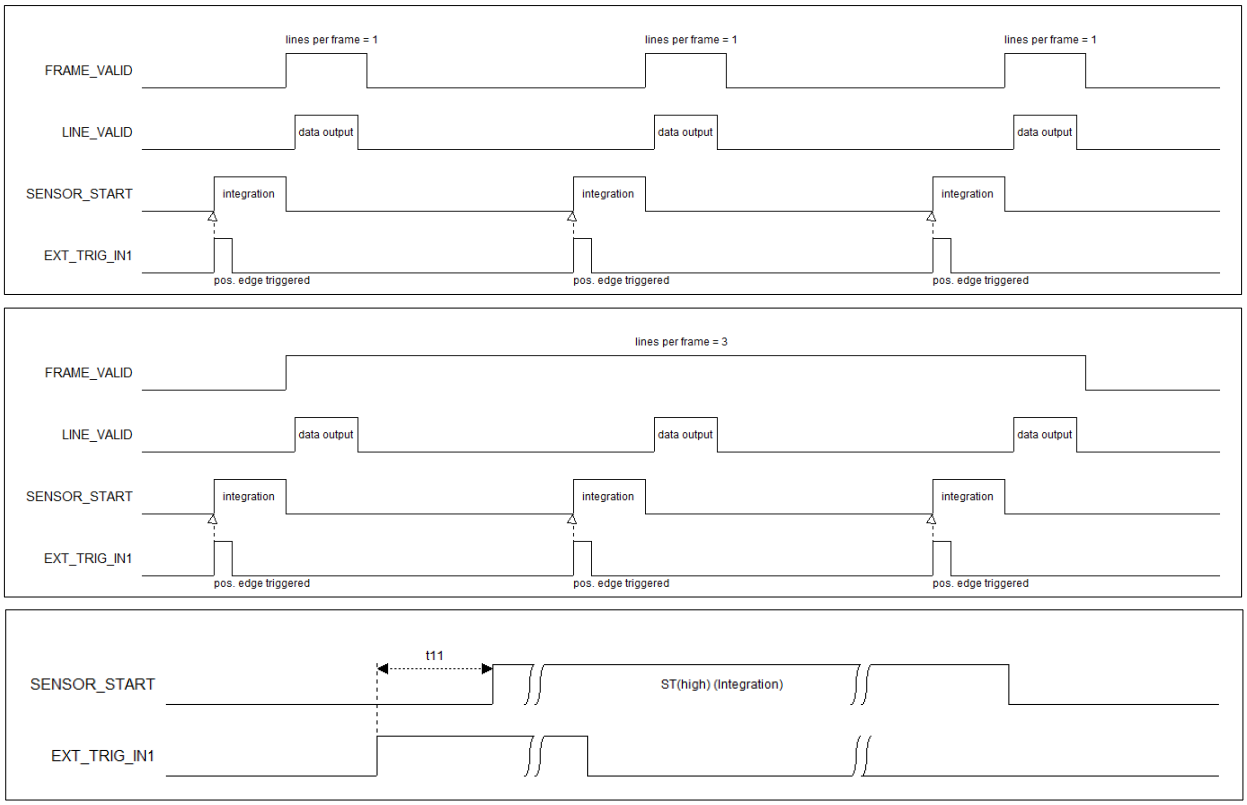


External rising edge trigger (Single-Line)

In the External Rising Edge Trigger (Single-Line) mode, the LISM-PI26xx module captures a single line of pixel data in response to each rising edge of an external trigger pulse:

- **Trigger Detection:** The module initiates the capture process upon detecting the rising edge of an external trigger pulse.
- **Integration Start:** Immediately following the trigger detection, the integration process begins, during which the sensor collects light information.
- **Data Output:** After the integration is complete, the synchronization signals are activated to HIGH. This change signals that the pixel data is now being outputted.

This mode is particularly useful for applications requiring precise control over the timing of each line capture, allowing synchronization with external events or other system components.

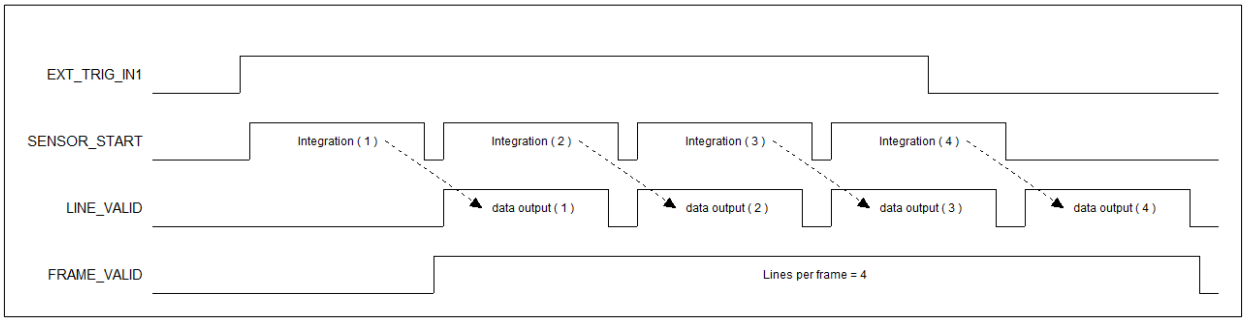


External High Level Trigger

In the External High Level Trigger mode, the LISM-PI26xx module operates in a manner that continuously captures multiple lines as long as the external trigger pulse remains high:

- **Continuous Capture**: While the external trigger pulse is sustained at a high level, the module remains in an active state, capturing line after line of pixel data. This process continues unabated throughout the duration of the high signal.
- **Pulse Width Dependent**: The total number of lines captured during this mode is directly dependent on the width of the trigger pulse. A longer pulse width results in more lines being captured, linking the duration of the pulse directly to the amount of data acquired.
- **Mode Characteristic**: This mode functions as a "gated" Free Running mode. It allows the system to freely run and capture data but only within the "gate" or period when the external trigger maintains a high state.

This mode is ideal for applications requiring continuous data capture that is gated by an external signal, providing flexibility in controlling the volume of data acquisition based on the length of the trigger pulse.

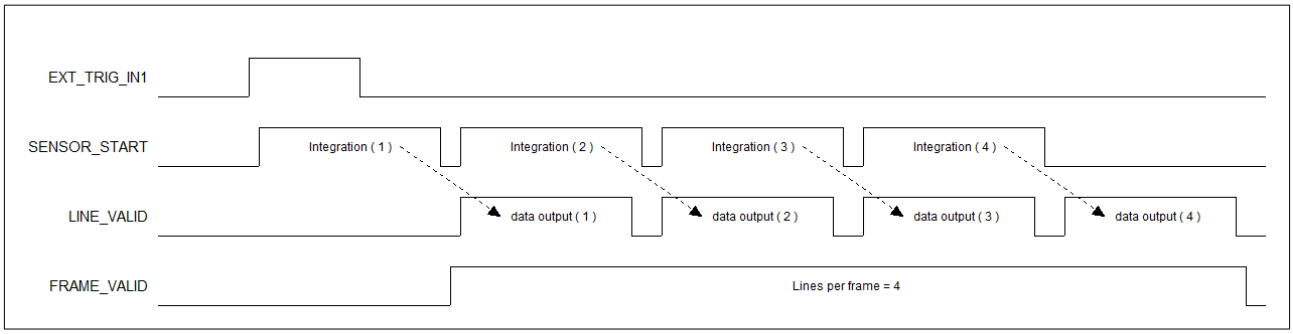


External Rising Edge Trigger (Multi-Line)

In the External Rising Edge Trigger (Multi-Line) mode, the LISM-PI26xx module is designed to capture multiple lines of pixel data upon each rising edge of an external trigger pulse:

- **Trigger Activation:** The capture process begins when a rising edge of an external trigger pulse is detected.
- **Sequential Processing:** Integration for each line occurs sequentially. As each line completes its integration, the module immediately outputs the corresponding pixel data for that line. This process is repeated line by line according to the number set in the LINES_PER_FRAME register.
- **Synchronization:** After the integration of each individual line, synchronization signals are set to HIGH, indicating the availability of valid pixel data from that line for output.

This operation mode ensures detailed and controlled acquisition of multiple lines in quick succession, making it ideal for scenarios requiring rapid yet precise data capture triggered by external events.

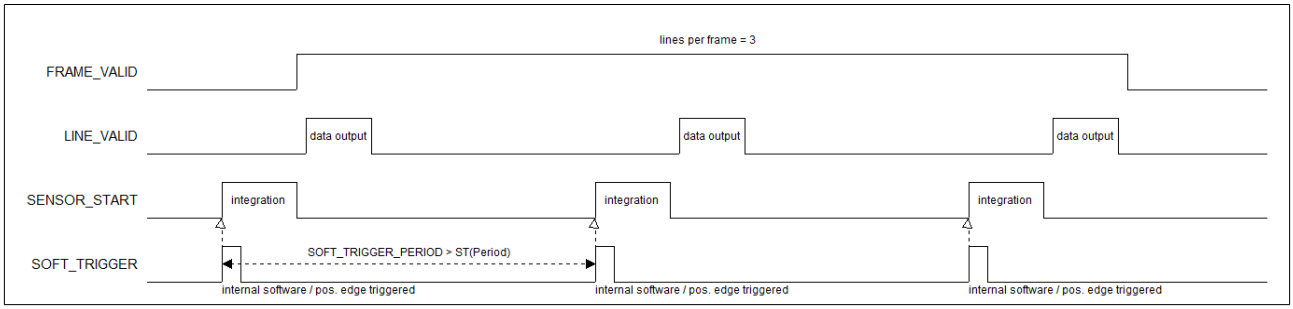


Internal Software Trigger Operation

In the Internal Software Trigger mode, the LISM-PI26xx module captures single lines of pixel data using a trigger signal that is generated internally by the Clock and Timing Generator:

- **Trigger Generation**: Unlike external trigger modes, the trigger in this mode does not come from an external device. Instead, it is generated internally, providing a high degree of control over the timing of data capture.
- **Configuration**: The frequency or period of the trigger signal is configurable through the SOFT_TRIGGER_PERIOD register. By setting this value, users can precisely manage when the internal trigger activates, allowing for consistent timing intervals between each captured line.
- **Operation Similarity**: This mode functions similarly to the External Rising Edge Trigger (Single Line) mode in that each trigger event results in the capture of a single line of pixel data. After the internal trigger signal is generated, the module begins the integration process for that line.
- **Data Output**: Once the integration is complete, synchronization signals are set to HIGH, and the pixel data for the captured line is output. This sequence ensures that each line is processed individually, with precise control over the capture timing based on the internal clock settings.

This mode is especially useful in environments where external triggering is impractical or where a highly stable and predictable triggering sequence is required.



Quadrature Encoder Trigger

The Quadrature Encoder Trigger mode in the LISM-PI26xx module allows pixel data acquisition to be controlled by signals from a quadrature encoder. This mode's functionality is configured using specific bits in the MODE_CONFIG register:

- **Directional Triggering:** The trigger behavior is determined by Bit[4] of the MODE_CONFIG register:
  - If **Bit[4] = 0**, trigger events are recognized only when the encoder rotates in the clockwise (CW) direction - corresponding to up-counting
  - If **Bit[4] = 1**, trigger events are recognized only when the encoder rotates in the counterclockwise (CCW) direction - corresponding to down-counting.

  In both cases, a single line of data is captured per trigger event.
- **Encoder Counter Control:** The operation further depends on the setting of Bit[5] in the MODE_CONFIG register:
  - If **Bit[5] = 0** (encoder counter DISABLED), integration and pixel data acquisition occur with each count detected by the encoder, facilitating continuous data capture synchronized to each encoder movement.
  - If **Bit[5] = 1** (encoder counter ENABLED), the module integrates and outputs pixel data only after the number of encoder events matches the value set in the QUAD_COUNT register. This setting allows for accumulation of encoder counts up to a pre-defined threshold before triggering data acquisition, offering precision in capturing data at specific intervals.

This trigger mode is particularly effective in applications requiring synchronization of image capture with mechanical movements, such as in conveyor belt monitoring or rotational scanning, where the movement direction and frequency of encoder signals directly influence imaging timing and sequence.

START_PULSE_HIGH (0x04), Write/Read, 32-bit data

The START_PULSE_HIGH register, designated as ST(High), specifies the duration of the high phase of the start pulse and primarily determines the integration time for the linear image sensor. This register is critical in setting the time during which the sensor actively integrates light information before readout. The unit of time for setting values in this register corresponds to the current setting of the pixel clock in the DATA_IF_CONFIG register:

- 200 kHz Pixel Clock: 5 µs per cycle
- 250 kHz Pixel Clock: 4 µs per cycle
- 400 kHz Pixel Clock: 2.5 µs per cycle
- 500 kHz Pixel Clock: 2 µs per cycle
- 1 MHz Pixel Clock: 1 µs per cycle
- 2 MHz Pixel Clock: 500 ns per cycle
- 5 MHz Pixel Clock: 200 ns per cycle
- 10 MHz Pixel Clock: 100 ns per cycle

START_PULSE_LOW (0x05), Write/Read, 32-bit data

The START_PULSE_LOW register, or ST(Low), configures the duration of the low phase of the start pulse. This setting does not directly affect the integration time but extends the readout period, impacting the total cycle time of sensor operation. The timing unit here also varies with the pixel clock setting as configured in the DATA_IF_CONFIG register.

Operational Dynamics of Start Pulse Registers

The frequency of the start pulse, which correlates directly to the line rate of the sensor (lines per second), is determined by the formula 1 / T, where T is the total period of the start pulse, calculated as ST(High) + ST(Low). These settings are adaptable depending on the specific requirements of the linear sensor in use and can be finely tuned to match the sensor's specifications for optimal performance.

Example Usage with S11639-1 Sensor

For a sensor configuration like the S11639-1, the start pulse timing can be set to optimize either the minimum integration time for higher line rates or the maximum allowable integration time based on operational needs. For instance, setting the ST(High) to the minimum required cycles and adjusting ST(Low) accordingly can achieve a high line rate, balancing the integration and readout times effectively. Conversely, maximizing ST(High) while keeping ST(Low) at the minimum effective level allows for the longest possible integration time, suitable for scenarios where high sensitivity is required.

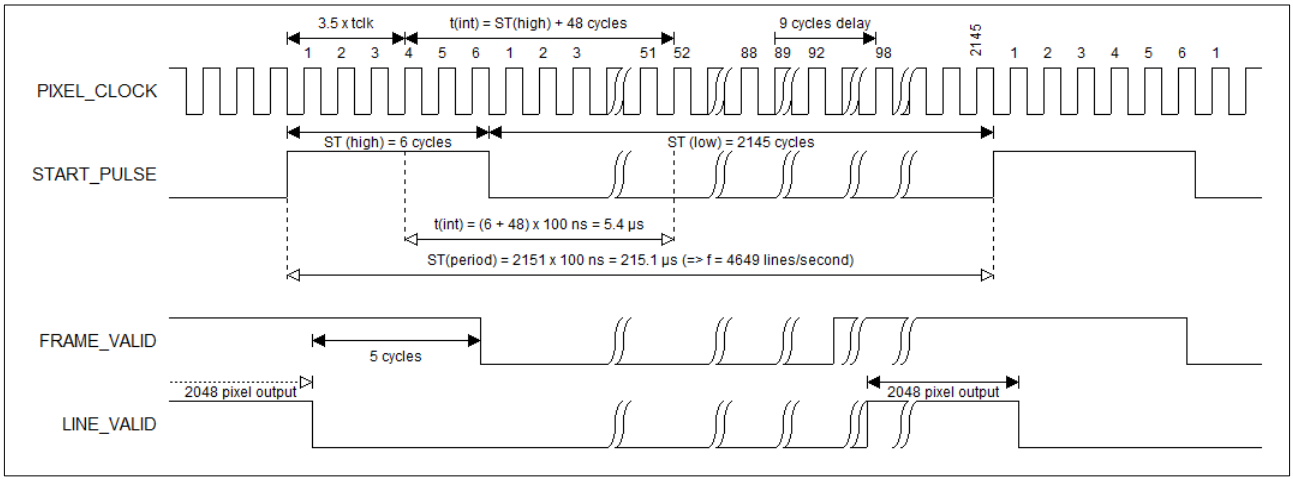
These registers offer a flexible approach to managing the timing requirements of various sensors, ensuring that the LISM-PI26xx can be adapted to a wide range of industrial and scientific imaging applications.

Using the S11639-1 sensor as an example, the functionality of both registers is detailed with the following specifications:

- *Pixel Clock: 10 MHz (DATA_IF_CONFIG = 0x07)*
- *Minimum Cycles for ST(Low)*: 100 cycles
- *Minimum Cycles for ST(High)*: 6 cycles
- *Integration Time*: Starts at the 4th edge following the positive edge of the start pulse and concludes at the 52nd edge following the negative edge.
- *Number of Active Pixels*: 2048
- *Valid Data Output:* Commences at the 89th edge (88 +1) following the negative edge of the start pulse.
- *ADC Output Delay*: 9 cycles
- *Frame Valid*: 5 trailing cycles

Example 1:
Minimum integration time and highest line rate possible:
ST(Period) = 2048 + (88 +1) + 9 + 5 ≥ 2151 cycles

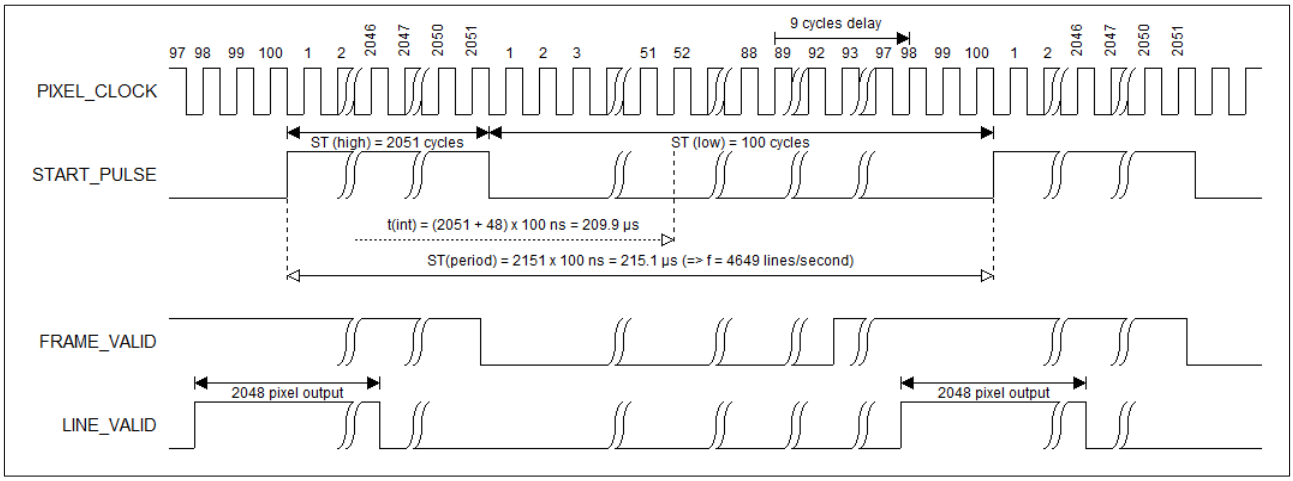
Time(Period) = 2151 cycles x 100 ns = 215.1 µs
=> Frequency = 1 / Time(Period) = 1 / 215.1 µs => 4649 lines per second
ST(High) = 6 cycles
=> Integration time = ST(High) – 4 + 52
= ST(High) + 48
= 6 + 48 = 54 cycles x 100 ns = 5.4 µs
ST(Low) = ST(Period) - ST(High) = 2151 - 6 = 2145 cycles



Example 2:
Maximum integration time and highest line rate possible:
ST(Period) = 2048 + (88 +1) + 9 + 5 ≥ 2151 cycles
Time(Period) = 2151 cycles x 100 ns = 215.1 µs
=> Frequency = 1 / Time(Period) = 1 / 215.1 µs => 4649 lines per second

ST(Low) = 89 + 9 = 98 cycles < ST(Min/Low) = 100 cycles
=> ST(Low) = 100 cycles

ST(High) = ST(Period) – ST(Low) = 2151 - 100 = 2051 cycles
=> Integration time = ST(High) – 4 + 52
= ST(High) + 48
= 2051 + 48 = 2099 cycles x 100 ns = 209.9 µs



LINES_PER_FRAME (0x06), Write/Read, 16-bit data

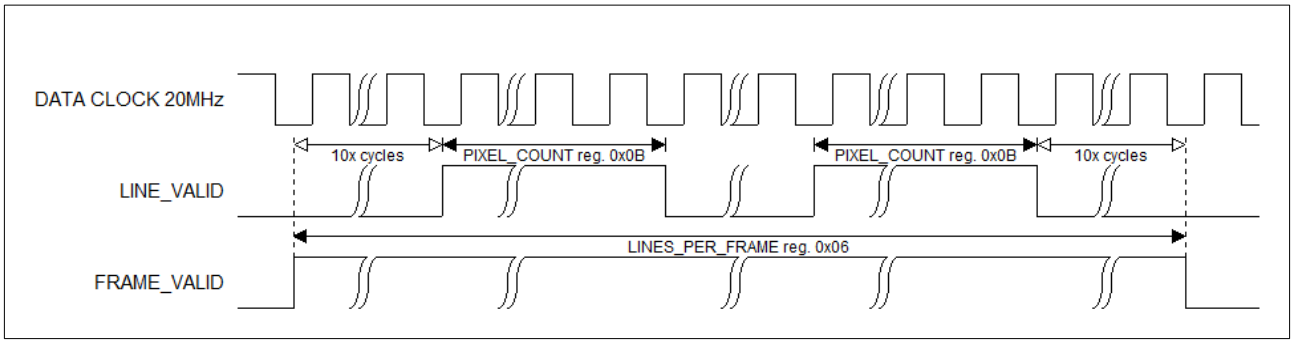
The LINES_PER_FRAME register within the LISM-PI26xx plays a crucial role in synchronizing the output of image data:

- The LINE_VALID signal remains in the HIGH state while valid pixel data for a single line is being output. This is contingent on the PIXEL_COUNT value in register (0x0B), which specifies the number of pixels per line.
- Concurrently, the LINES_PER_FRAME register specifies the quantity of lines that constitute a full image frame, which, in turn, controls the duration for which the FRAME_VALID signal maintains its HIGH state.

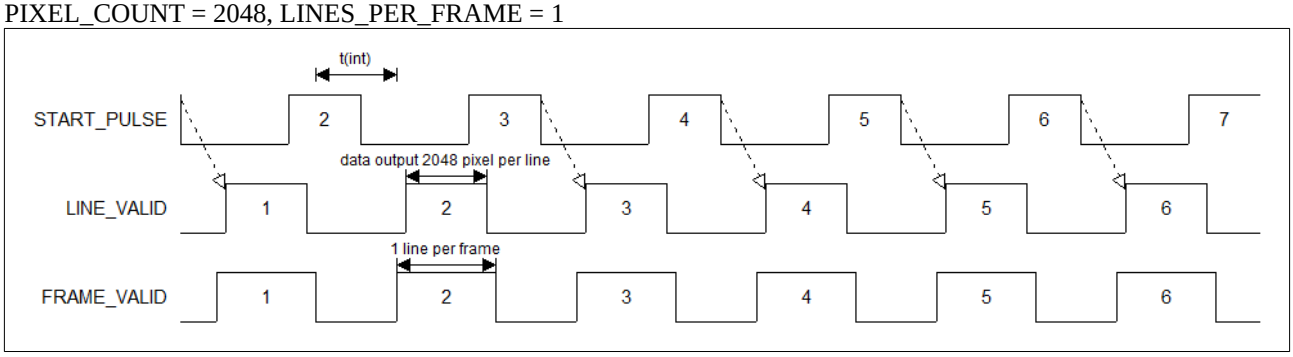
In the operational sequence, there is a designed delay of 10 data clock cycles:

- The LINE_VALID signal transitions to HIGH ten cycles after the rising edge of FRAME_VALID, signifying the start of a new line.
- Similarly, ten cycles after the falling edge of LINE_VALID, the FRAME_VALID signal transitions to its falling edge, indicating the end of the frame.

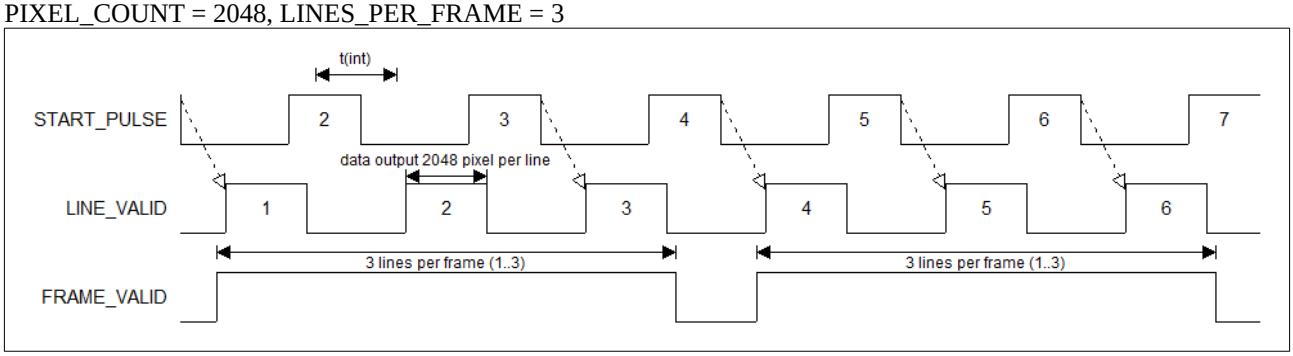
This synchronization ensures accurate alignment between the start of line data output and the delineation of frames, which is essential for the orderly processing and analysis of the image data captured by the LISM-PI26xx module.



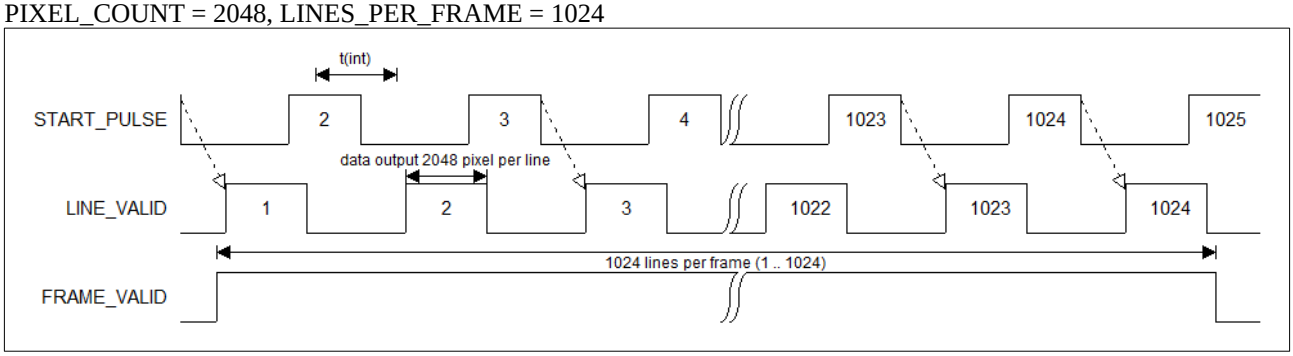
PIXEL_COUNT = 2048, LINES_PER_FRAME = 1



PIXEL_COUNT = 2048, LINES_PER_FRAME = 3



PIXEL_COUNT = 2048, LINES_PER_FRAME = 1024



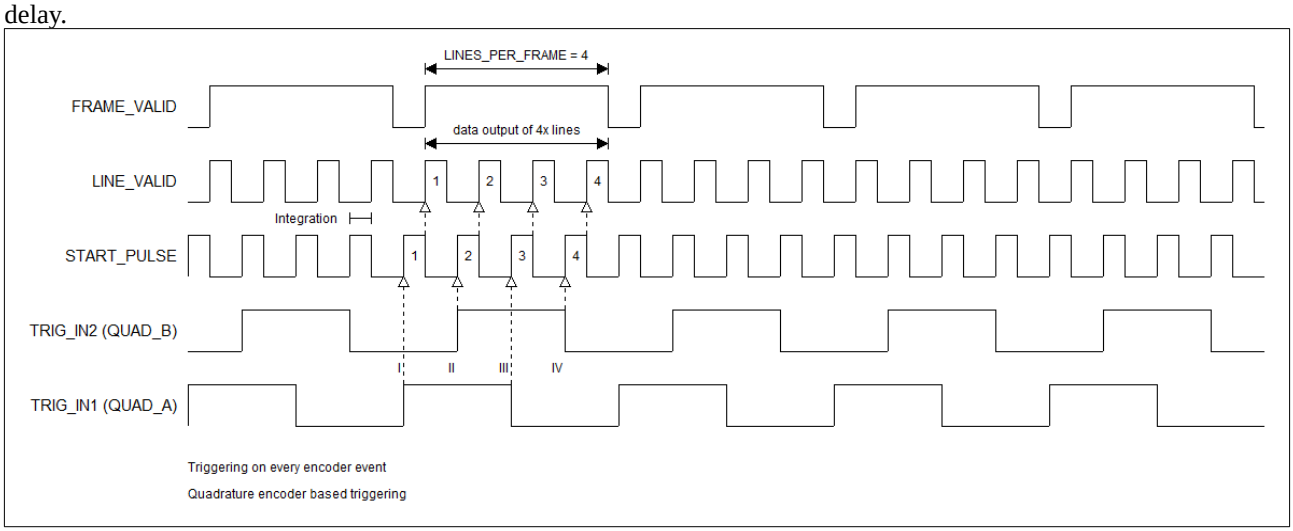
QUAD_COUNT (0x07), Write/Read, 32-bit data

The QUAD_COUNT register has a pivotal function in determining the trigger events based on the quadrature encoder's inputs:

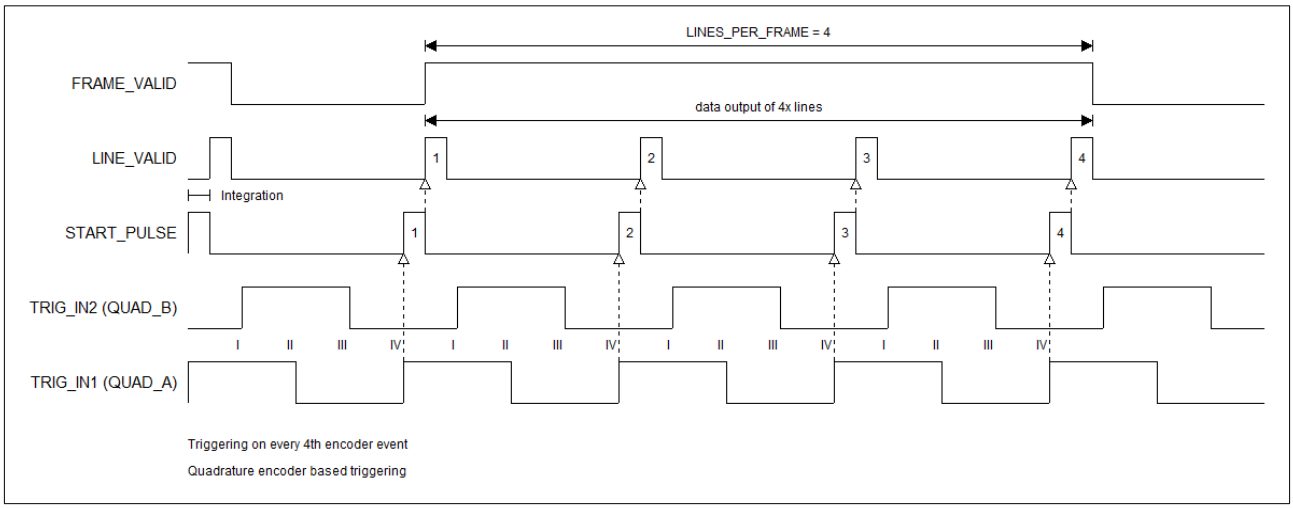
- It sets a specific count of quadrature encoder pulses required to initiate a single trigger event for the system.
- When the quadrature encoder counter feature is active, this register's value matches the number of encoder pulses after which the system starts the integration process and the acquisition of pixel data.

In operational scenarios:

Example 1:
With the quadrature encoder counter turned off, each encoder pulse directly triggers pixel data acquisition without delay.



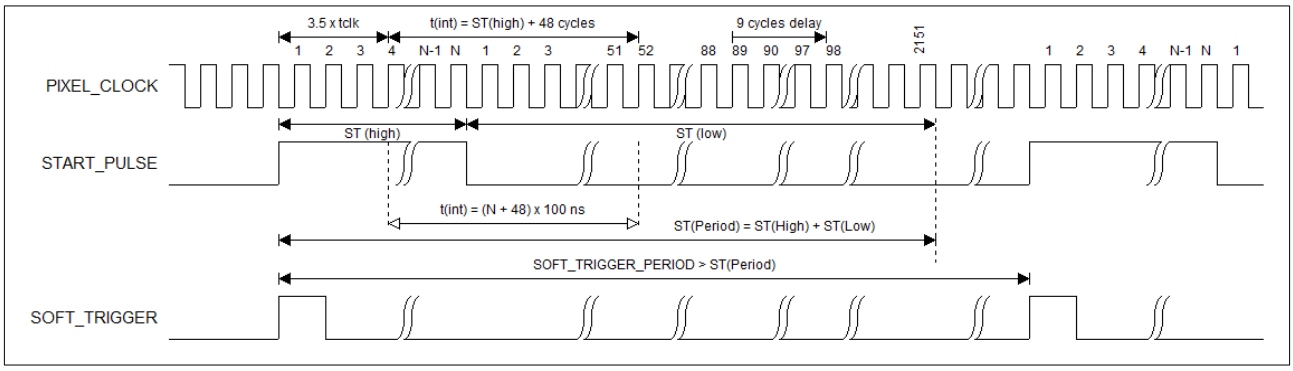
Example 2:
Conversely, with the quadrature encoder counter turned on and the QUAD_COUNT set to a value of 4, for example, pixel data acquisition will only be triggered after every fourth encoder pulse, allowing for more controlled and periodic data capture.



SOFT_TRIGGER_PERIOD (0x08), Write/Read, 32-bit data

The SOFT_TRIGGER_PERIOD register configures the interval for the module's internal software-triggered operations:

- The designated period within this register is measured in microseconds and determines the timing for the software trigger.
- The minimum allowable period, which corresponds to the maximum line rate, is constrained by the duration of the START_PULSE cycle, and thus, it is influenced by the specific sensor model in use.
- To ensure proper functioning, the value set for the SOFT_TRIGGER_PERIOD must exceed the total START_PULSE period. This total period is the sum of the ST(High) and ST(Low) durations as defined in their respective registers. T(SoftTrig) > ST(Period) = ST(High) + ST(Low).



TRIGGER_OUTPUT_DELAY (0x09), Write/Read, 32-bit data

The TRIGGER_OUTPUT_DELAY register on the LISM-PI26xx module allows for fine-tuning of the trigger output timing:

- This register adjusts the delay between the start of the integration process and the issuance of the trigger pulse. By setting this delay, users can synchronize external devices with the imaging process of the LISM-PI26xx module.
- The delay time is specified in microseconds, enabling precise control over the timing of the trigger relative to the integration event.
- Typically, the trigger pulse is emitted immediately after the start of integration. However, with the TRIGGER_OUTPUT_DELAY, the user can delay the trigger pulse to meet specific application requirements, ensuring that the trigger pulse occurs exactly when needed within the system's workflow.

TRIGGER_OUTPUT_WIDTH (0x0A), Write/Read, 32-bit data

The TRIGGER_OUTPUT_WIDTH register on the LISM-PI26xx module is responsible for defining the duration of the trigger signal:

- This register is utilized to set the width, or pulse duration, of the trigger signal. The width is determined based on the requirements of the system and is specified in microseconds.
- It's critical to ensure that the sum of the delay (set by the TRIGGER_OUTPUT_DELAY register) and the width (set by the TRIGGER_OUTPUT_WIDTH register) does not exceed the duration of the START_PULSE. If the total time of the delay and width surpasses the START_PULSE, the trigger signal is reset to low immediately before the commencement of the next integration cycle. This reset ensures that the trigger pulse can be accurately generated for the subsequent integration start.
- Adjusting the TRIGGER_OUTPUT_WIDTH allows for synchronization with peripheral devices or systems that require precise timing signals for operation in conjunction with the LISM-PI26xx module’s imaging processes.

PIXEL_COUNT (0x0B), Write/Read, 16-bit data

The PIXEL_COUNT register is a key component of the LISM-PI26xx module, providing adaptability to support line sensors with varying pixel specifications:

- This register allows users to specify the number of pixels to be read from the line sensor and transmitted via the parallel interface. It is synchronized with the LINE_VALID signal to ensure proper timing and data integrity.
- The LISM_PI26xx module is designed to accommodate different line sensors by enabling adjustment of the pixel count through the PIXEL_COUNT register, ensuring compatibility with sensors of varying pixel quantities.
- Users have the flexibility to read only a specific region of the line sensor—fewer pixels than the sensor's total count. This is achieved by delaying the start of the readout (configured by the EDGES_BEFORE_DATA register) and ending the readout earlier, which involves setting the PIXEL_COUNT to a lower number than the total pixels available on the sensor.
- It is crucial to ensure that the start pulse duration is adequate for the actual number of pixels to be clocked out from the sensor.

EDGES_BEFORE_DATA (0x0C), Write/Read, 8-bit data

The EDGES_BEFORE_DATA register is an integral part of the LISM-PI26xx module's timing configuration, tailored to the specifications of the connected line sensor:

- The value set in this register corresponds to the number of clock cycles that occur after the negative edge of the start pulse before the pixel data output begins. This sensor-specific value, detailed in the sensor's datasheet, is essential for aligning data output with the sensor's operational timing.
- As the LINE_VALID signal is activated concurrently with the data output, this register effectively dictates the timing of the LINE_VALID signal transition to high, marking the commencement of valid data output.
- The LINE_VALID signal remains high for the duration equal to the number of cycles specified in the PIXEL_COUNT register, ensuring that only the designated number of pixels are read out. Once this count is reached, the LINE_VALID signal transitions back to low, signaling the end of data transmission for the current line.
- Utilizing the EDGES_BEFORE_DATA register, users can fine-tune the start of data acquisition, allowing for a precise region of interest to be captured, particularly when reading a subset of the sensor's total pixel array.

DATA_IF_CONFIG (0x0D), Write/Read, 8-bit Data

The DATA_IF_CONFIG register allows for the configuration of the 8-bit parallel interface of the LISM_PI26xx module. This register provides users with the flexibility to adjust the frequency and polarity of the Data Clock to ensure optimal compatibility and performance for various applications.

Bit	Value	Description	
[2:0]		Pixel Clock Frequency	Data Clock Frequency
	0x00	200 kHz	400 kHz
	0x01	250 kHz	500 kHz
	0x02	400 kHz	800 kHz
	0x03	500 kHz	1 MHz
	0x04	1 MHz	2 MHz
	0x05	2 MHz	4 MHz
	0x06	5 MHz	10 MHz
	0x07	10 MHz	20 MHz

[3]	0	Data Clock Polarity Data is output on the falling edge of the Data Clock. This setting can be used when an external processor needs to capture data on the rising edge.
	1	The polarity of the Data Clock output is inverted, resulting in data output on the rising edge. This setting can be used when an external processor needs to capture data on the falling edge.
[4]	0	Line Valid Signal: High Active
	1	Line Valid Signal: Low Active
[5]	0	Frame Valid Signal: High Active
	1	Frame Valid Signal: Low Active
[6]	0	Trigger Output: High Active
	1	Trigger Output: Low Active
[7]	x	Not used, reserved for future use

ADC_FULL_SCALE (0x20), Write/Read, 8-bit data

The ADC_FULL_SCALE register within the LISM-PI26xx module is crucial for setting the measurement range of the Analog-to-Digital Converter (ADC):

- This register allows for the selection between two full-scale voltage ranges, affecting the resolution and gain of the ADC.
- Setting the register value to 0 configures the ADC to operate with a 1.6V full-scale range. This setting is typically used for applications requiring higher resolution within a smaller voltage span.
- Alternatively, inputting a value of 1 adjusts the ADC to a 2V full-scale range. This configuration is beneficial for applications that need to measure larger signal voltages or when a broader dynamic range is necessary.
- The choice of full-scale voltage directly influences the precision of digital conversion and is an important parameter to optimize based on the specific requirements of the sensor and the application it is used for.

ADC_GAIN (0x21), Write/Read, 8-bit data

The ADC_GAIN register is instrumental in adjusting the gain of the ADC in the LISM-PI26xx module:

- To configure the ADC gain, the upper three bits (D8, D7, and D6) of this 8-bit register must be set to zero. The remaining bits (D5 to D0) are used to define the gain value, allowing for fine adjustment between 1× and 5.85× gain across 64 discrete steps.
- The encoding for the ADC_GAIN register follows a straight binary format. A value of all zeros (000000) correlates to the minimum gain setting of 1×, whereas all ones (111111) corresponds to the maximum gain of 5.85×.

D8	D7	D6	D5	D4	D3	D2	D1	D0	Gain (V/V)	Gain (dB)
0	*0*	*MSB*						*LSB*		
0	0	0	0	0	0	0	0	0	1.0	0.0
0	0	0	0	0	0	0	0	1	1.013	0.11
				•					•	•
				•					•	•
				•					•	•
0	0	0	1	1	1	1	1	0	5.43	14.7
0	0	0	1	1	1	1	1	1	5.85	15.34

- The gain setting incrementally increases in a manner that is approximately linear on a decibel scale. The specific gain can be calculated using the formula:

$Gain = \frac{76}{76 - G}$ Here, 'G' represents the binary value set in the register, ranging from 0 to 63.

- By utilizing this register, users can precisely calibrate the ADC's gain to suit the dynamic range and sensitivity requirements of their specific applications, ensuring accurate and tailored signal amplification.

ADC_OFFSET (0x22), Write/Read, 16-bit data

The ADC_OFFSET register is crucial for calibrating the baseline of the ADC to accommodate the unique zero base line (dark signal offset) inherent to each sensor:

- The offset can be finely adjusted across a range from –250 mV to +250 mV, offering 512 discrete levels of calibration to ensure the sensor's measuring range is fully utilized from 0 to FULL_SCALE.
- Bits D8 through D0 are dedicated to setting the offset value. The register utilizes a Sign Magnitude representation where D8 is the sign bit. A '0' in D8 signifies a positive offset, while a '1' indicates a negative offset.
- The magnitude of the offset, defined by bits D7 to D0, allows users to correct for any dark signal offset that a sensor might exhibit, which is critical for precision measurements and optimizing the ADC's dynamic range.
- Through this register, the ADC can be fine-tuned to negate sensor-specific noise or offset, thereby enhancing the quality of data captured across various lighting conditions and ensuring accurate readings up to the full scale of the ADC.

D8	D7	D6	D5	D4	D3	D2	D1	D0	Offset (mV)
MSB								*LSB*	
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	1	+0.98
				•					•
				•					•
				•					•
0	1	1	1	1	1	1	1	1	+250
1	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	1	-0.98
				•					•
				•					•
				•					•
1	1	1	1	1	1	1	1	1	-250

ADC_EXT_REF (0x30), Write/Read, 16-bit data

The ADC_EXT_REF register is critical for setting the external reference voltage used by the Analog-to-Digital Converter (ADC) in the LISM-PI26xx module:

- This register enables the adjustment of the DAC (Digital-Analog Converter) output voltage, which serves as the external reference voltage for the ADC. The voltage set here is subtracted from the sensor's signal, effectively serving as a preliminary stage for fine-tuning the zero baseline, similar to the ADC_OFFSET register but on a broader scale.
- The external reference voltage can be adjusted between 0V and 2V, offering a high degree of precision with 1024 incremental steps (10 bits) available for setting the desired voltage level.
- Adjusting this voltage allows users to roughly set the baseline or zero point of the sensor's output signal, ensuring that the ADC can fully utilize its measuring range from 0 to its full scale. This is especially useful for compensating for sensor-specific variations and optimizing the initial conditions for further fine adjustments via the ADC_OFFSET.
- By manipulating this register, operators can enhance the accuracy and reliability of measurements by aligning the initial signal conditions to the optimal operating range of the ADC.

INIT_STATUS (0xEE), Read-Only, 8-bit data

The INIT_STATUS register serves as a crucial indicator of the initialization status of the LISM-PI26xx module following a power-on reset:

- This read-only register provides insight into whether all key components of the module - including the ADC, DAC, and the Clock and Timing Generator - have been successfully initialized using the settings stored in the EEPROM.
- Upon module reset, the INIT_STATUS register reflects the completion and current state of the system's initialization process, ensuring that all configurations and parameters are correctly set according to predefined settings.
- Monitoring the INIT_STATUS register is essential for diagnostic purposes and for verifying that the module is ready for operation post-reset. It assists in determining if the system has encountered any issues during the initialization phase, allowing for timely troubleshooting and corrective actions.
- This register is critical for system health checks and ensures that the device functions as intended, with all components properly configured from the onset of power-up.

COM_RESULT (0xEF), Read-Only, 8-bit data

The COM_RESULT register plays a vital role in ensuring the integrity of communications with the Clock and Timing Generator within the LISM-PI26xx module:

- This read-only register stores the outcome of the most recent data transmission to and from the Clock and Timing Generator, providing a direct indication of the communication success.
- After each operation involving the Clock and Timing Generator, whether sending or receiving data, the result is logged into the COM_RESULT register. This enables the detection of any communication errors that may have occurred during the transmission process.
- Regular monitoring of this register allows for the confirmation that all commands, configurations, and data exchanges with the Clock and Timing Generator have been executed correctly. It acts as a diagnostic tool to verify data integrity and proper system function.
- The ability to read the COM_RESULT register is crucial for maintaining system reliability and performance, ensuring that errors are quickly identified and addressed to prevent incorrect operation.

TEMP_SLAVE_ADDRESS (0xF0), Write/Read, 8-bit data

The TEMP_SLAVE_ADDRESS register is essential for managing the I2C slave address of the LISM-PI26xx module:

- This register allows for the temporary input and verification of a new I2C slave address before it is permanently stored in the EEPROM. This process ensures that any changes to the slave address are intentional and verified before becoming permanent.
- The new address should first be written to the TEMP_SLAVE_ADDRESS register. It can then be read back from this register to confirm correctness. Following verification, the new address can be committed to the EEPROM using the STORE_SLAVE_ADDRESS command.
- Addresses are to be written and stored in an 8-bit format. Importantly, the least significant bit (LSB) of the address must always be set to 0. This formatting rule ensures that the address conforms to standard I2C address specifications.
- Utilizing the TEMP_SLAVE_ADDRESS register allows for flexible and secure updates to the module's communication address, accommodating changes in network configurations or system expansions without disruption to the overall device operations.

STORE_SLAVE_ADDRESS (0xF1), no data

The STORE_SLAVE_ADDRESS register is specifically designed to commit the I2C slave address to the EEPROM of the LISM-PI26xx module:

- This write-only register does not accept data inputs but rather executes a critical command to permanently store the current value in the TEMP_SLAVE_ADDRESS register into the EEPROM.
- Upon execution, the I2C slave address stored in the EEPROM is updated, and this address will be used to initialize the module during the next startup. This ensures that the module can be correctly identified and communicated with in the I2C network after a reset or power cycle.
- The STORE_SLAVE_ADDRESS register is essential for finalizing changes to the module's communication settings, thereby providing a secure method to update and preserve the new configuration without direct data entry into the register itself.
- Utilizing this register allows system developers to effectively manage the module’s network settings, ensuring seamless integration and consistent operation within its designated I2C environment.

STORE_SETTINGS (0xF2), no data

The STORE_SETTINGS register is a crucial component of the LISM-PI26xx module for preserving the operational parameters and settings:

- This register does not accept data entries but is used exclusively to execute a command that saves all current module parameters and settings, except for the I2C slave address, into the EEPROM.
- The saved settings in the EEPROM are then utilized to initialize the module's components upon a system reset, ensuring that the module operates with the desired configurations that were set prior to the reset.
- The use of the STORE_SETTINGS register provides a secure and reliable way to maintain consistency in the module's performance and behavior, as it ensures all adjusted settings are preserved and reapplied after any reset or power interruption.
- This functionality is particularly important for system stability and operational readiness, allowing the module to recover swiftly and correctly without the need for reconfiguration after every reboot or power cycle.

RELOAD_SETTINGS (0xF3), no data

The RELOAD_SETTINGS register is designed to facilitate immediate reinitialization of the LISM-PI26xx module using the parameters stored in the EEPROM:

- This write-only register executes a command rather than storing data. When activated, it triggers the module to fetch and apply all configuration settings from the EEPROM.
- The functionality of the RELOAD_SETTINGS register is crucial for situations where system parameters need to be restored to their last saved state without undergoing a full system reset. This allows for a quick reconfiguration based on previously established settings, ensuring the module operates consistently according to the last configured parameters.
- Utilizing this register is especially useful in scenarios where modifications to settings need to be tested and possibly reverted quickly, enhancing operational flexibility and maintenance efficiency.
- The command ensures the module remains in alignment with the desired operational settings stored in the EEPROM, providing a robust method for managing system behavior dynamically and securely.

RESET_SETTINGS (0xF4), no data

The RESET_SETTINGS register is an essential tool for restoring the LISM-PI26xx module to its factory default settings:

- This write-only register does not take in data but executes a command to reset all parameters and settings stored in the EEPROM to their default state.
- Upon activation, the module immediately reinitializes itself using these default values, ensuring that any previous configurations are overwritten and the module returns to a known, baseline operational status.
- This functionality is particularly valuable for troubleshooting and maintenance, allowing system administrators to quickly revert the module to its original settings if errors arise or if there is a need to start configuration afresh.
- Utilizing the RESET_SETTINGS register can help prevent persistent issues that might be caused by improper configurations or corrupt settings, ensuring the module operates reliably within its designed specifications.

SENSOR_BOARD_ID (0xFA), Read-Only, 16-bit data

Sensor Board Identifier
The SENSOR_BOARD_ID register provides a unique identifier for the type of sensor board installed, based on pin count and sensor group classification. The 16-bit value is encoded as follows:

- **MSB**: Number of sensor pins (e.g., 24 for 24-pin sensors)
- **LSB**: Sensor group ID

Each group defines a set of CMOS sensors that are **electrically and pin-compatible**. Sensor boards from different groups (even with the same pin count) are not interchangeable, ensuring electrical safety and signal integrity.

Examples:
0x2401 → 24-pin sensor, Group 1
0x2402 → 24-pin sensor, Group 2 (not compatible with Group 1)
0x2201 → 22-pin sensor, Group 1

This register is essential for:
- Identifying sensor compatibility during initialization or system setup
- Supporting modularity with clear group-based sensor assignment

SB_HW_VERSION (0xFB), Read-Only, 16-bit data

Sensor Board Hardware Version
This register stores the hardware version of the sensor board connected to the LISM-PI26xx module. The 16-bit value follows the same structure:

- **MSB**: Major hardware version
- **LSB**: Minor hardware version

The version corresponds to the physical revision printed on the sensor board and indicates possible changes in layout, signal routing, or supported sensor groups.

This information is useful for:
- Automatically validating compatibility with certain sensor types
- Supporting service and traceability in modular systems
- Updating system behavior or calibration based on board revision

Accessing this register allows software to adapt dynamically to variations in sensor board hardware, improving robustness and reducing maintenance complexity.

PB_HW_VERSION (0xFC), Read-Only, 16-bit data

Processor Board Hardware Version
This register indicates the hardware version of the processor board in the LISM-PI26xx module. The value is encoded as a 16-bit word with the following format:

- **MSB**: Major hardware version
- **LSB**: Minor hardware version

The hardware version reflects the actual revision printed on the board and identifies electrical or mechanical design changes that may affect functionality or compatibility.

Typical uses of this register include:
- Verifying hardware compatibility with specific firmware versions or sensors
- Tracking hardware revisions in production or service environments
- Enabling software to adapt automatically to board-level differences

Knowing the processor board hardware version ensures correct system configuration and helps avoid issues due to mismatched hardware.

P1_FM_VERSION (0xFD), Read-Only, 16-bit data

The P1_FM_VERSION register is pivotal for identifying the firmware version of the microcontroller within the LISM-PI26xx module:

- This read-only register holds a 16-bit word that provides the firmware version installed on the microcontroller. The value stored is divided into two parts: the Most Significant Byte (MSB) represents the major version, and the Least Significant Byte (LSB) represents the minor version.
- The major version number indicates significant firmware releases that typically include new features or substantial changes, while the minor version number is used for smaller, incremental updates and bug fixes.
- Accessing this register allows users to verify the current firmware version, which is essential for troubleshooting, compatibility checks, and ensuring that the module operates with the latest enhancements and corrections.
- Knowing the firmware version is crucial for maintaining the reliability and functionality of the module, especially when diagnosing issues or performing system upgrades.

P2_FM_VERSION (0xFE), Read-Only, 16-bit data

The P2_FM_VERSION register provides essential information regarding the firmware version of the Clock and Timing Generator within the LISM-PI26xx module:

- This register is read-only and contains a 16-bit word that reflects the firmware version of the Clock and Timing Generator. This 16-bit format is split into two parts: the Most Significant Byte (MSB) details the major version, while the Least Significant Byte (LSB) outlines the minor version.
- The major version indicates significant updates that may include new features or major changes to the functionality, whereas the minor version encompasses minor updates and bug fixes, ensuring that the Clock and Timing Generator operates optimally.
- Accessing the P2_FM_VERSION register allows users to confirm the current firmware revision, facilitating effective management, compatibility assessments, and troubleshooting of the Clock and Timing Generator.
- Keeping track of the firmware version is crucial for maintaining operational integrity and compatibility, especially important in environments where precise timing and synchronization are required.

HW_RESET (0xFF), Write-Only, no data

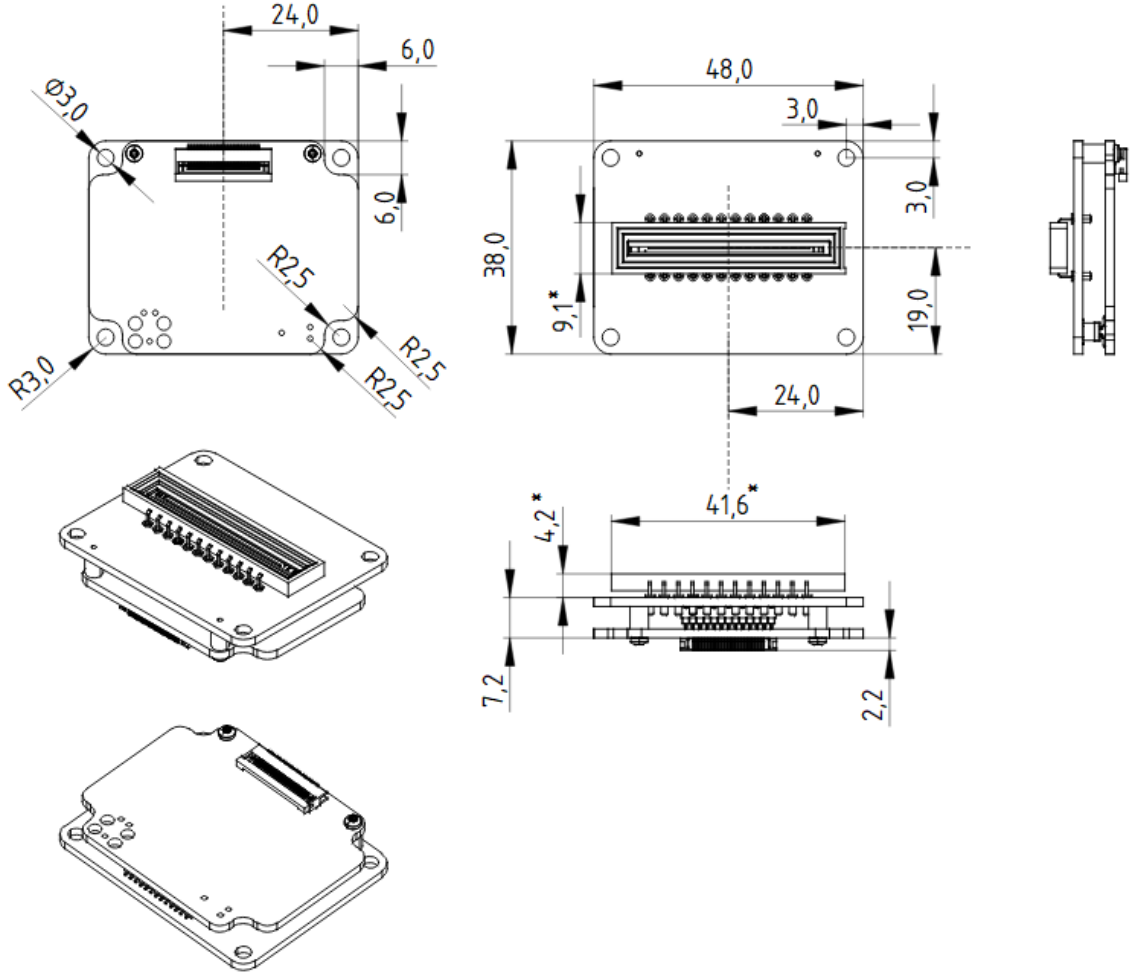
The HW_RESET register is a critical control mechanism within the LISM-PI26xx module, specifically designed for the Clock and Timing Generator:

- This write-only register does not require data input but rather executes a hardware reset command. Activating this register initiates a reset of the Timing and Clock Generator, effectively restarting the component.
- Upon activation, the HW_RESET command resets the Clock and Timing Generator to ensure it starts operating with the configuration settings stored in the EEPROM. This ensures that any temporary operational issues can be addressed by reverting to a known, stable configuration.

- The reset process facilitated by the HW_RESET register is essential for maintaining the reliability and precision of the Clock and Timing Generator, especially in scenarios where timing accuracy is critical.

- Utilizing this register allows for a straightforward method to ensure that the Clock and Timing Generator operates consistently and according to predefined settings, providing a quick recovery option from operational anomalies or after changes to the system configuration.

Dimensional outlines (unit mm)



*) depends on sensor installed

Revision history

Rev.	Date	Description
1.0	08 April 2025	LISM-PI26xx – Product data sheet
1.1	10 April 2025	Key Features Overview: Corrections made to Synchronization signals and Trigger Output descriptions. Data in the 'I2C Commands' table updated and corrected for the following registers: MODE_CONFIG START_PULSE_HIGH START_PULSE_LOW DATA_IF_CONFIG Data in the 'DATA_IF_CONFIG' table updated and corrected.
1.2	24 April 2025	Added new registers for board identification: SENSOR_BOARD_ID (0xFA) – Sensor board group and pin count identifier SB_HW_VERSION (0xFB) – Sensor board hardware version PB_HW_VERSION (0xFC) – Processor board hardware version Register descriptions added to the register map and detailed in the manual
1.3	07 May 2025	**Extended frequency options for Pixel Clock and Data Clock:** Pixel Clock now supports: 200 kHz, 250 kHz, 400 kHz, 500 kHz, 1 MHz, 2 MHz, 5 MHz, and 10 MHz Data Clock now supports: 400 kHz, 500 kHz, 800 kHz, 1 MHz, 2 MHz, 4 MHz, 10 MHz, and 20 MHz **Documentation updated accordingly:** Timing Characteristics table: Minimum and maximum values adjusted Section "Pixel Data Output and Synchronization Signals" updated I2C Commands table: Updated descriptions for START_PULSE_HIGH and DATA_IF_CONFIG Register descriptions: Updated entries for START_PULSE_HIGH and DATA_IF_CONFIG **I2C Bus section updated:** Added description of behavior when a new I2C command is received before the previous command is fully processed. Explained that the LISM-PI26xx responds with a NAK and that the master should retry after a short delay.
1.4	12 May 2025	**Timing Characteristics table updated:** Minimum and maximum values adjusted to reflect all available frequency options for Pixel Clock and Data Clock.