PAA3905E1-Q: Optical Motion Tracking Chip

General Description

The PAA3905E1 is PixArt Imaging's optical motion tracking chip specifically designed for low light condition operation. The state of the art of architecture allows motion tracking under low-light conditions as low as 5 lux. The chip supports a wide working distance of 80 mm to infinity under challenging conditions to cater for different far-field application needs and ambient conditions. It is suitable for far-field tracking applications in providing motion tracking with a stable hovering function for Drone.

Key Features

- Wide working distance from 80mm to infinity
- No focal length calibration required
- Auto-detection of challenging conditions, e.g. checkerboards, stripes, glossy surface, and yawing
- 16-bit motion data output with motion detect pin output
- Internal oscillator
- Raw Data Output through register read
- Support synchronized multi-chip operation
- Automatic switching of Operation Mode

Operation Modes

Mode	Description	Lux (Typ)
0 @ 126 fps	Bright Mode for general motion tracking	60
1 @ 126 fps (Default)	Low Light Mode for low light motion tracking	30
2 @ 50 fps	Super Low Light Mode for super low light and low-speed motion tracking	5

Applications

- Drone or other devices that require far-field motion detection and hovering stability
- X-Y positioning in GPS denied environment

Key Parameters

Parameter	Value
Supply Voltage	VDD: 1.8 to 2.0 V VDDIO: 1.8 to 3.6 V
Working Range	80 mm to infinity
Interface	4-Wire SPI @ 2MHz
Speed	maximum 7.4 rad/s (Mode 0 & 1)
Effective Viewing angle	42°
Power Consumption	3.5 mA at RUN state
Package size (L x W x H)	12-pin LGA Package with L242-ZSZ1 Lens 4.0 x 5.0 x 3.0 mm^3

Ordering Information

Part Number	Description	Package Type	Packing Type	MOQ
PAA3905E1-Q	Optical Motion Tracking Chip	12-pin LGA	Tape & Reel	2500
L242-ZSZ1	Lens	Plastic lens	Tray	2880

RoHS compliant

For any additional inquiries, please contact us at http://www.pixart.com

Table of Content

List of Figures

List of Tables

1.0 Introduction

1.1 Overview

PAA3905E1-Q contains a Picture Element Acquisition System (PEAS), a hard-coded Digital Signal Processing System (DSPS), and a four-wire serial port interface. It is based on Optical Navigation Technology to measure changes in a position to acquire sequential picture elements for calculating the direction and magnitude of movement (delta X and delta Y).

Note: Throughout this document, the PAA3905E1-Q is referred to as the "chip".

1.2 Block Diagram

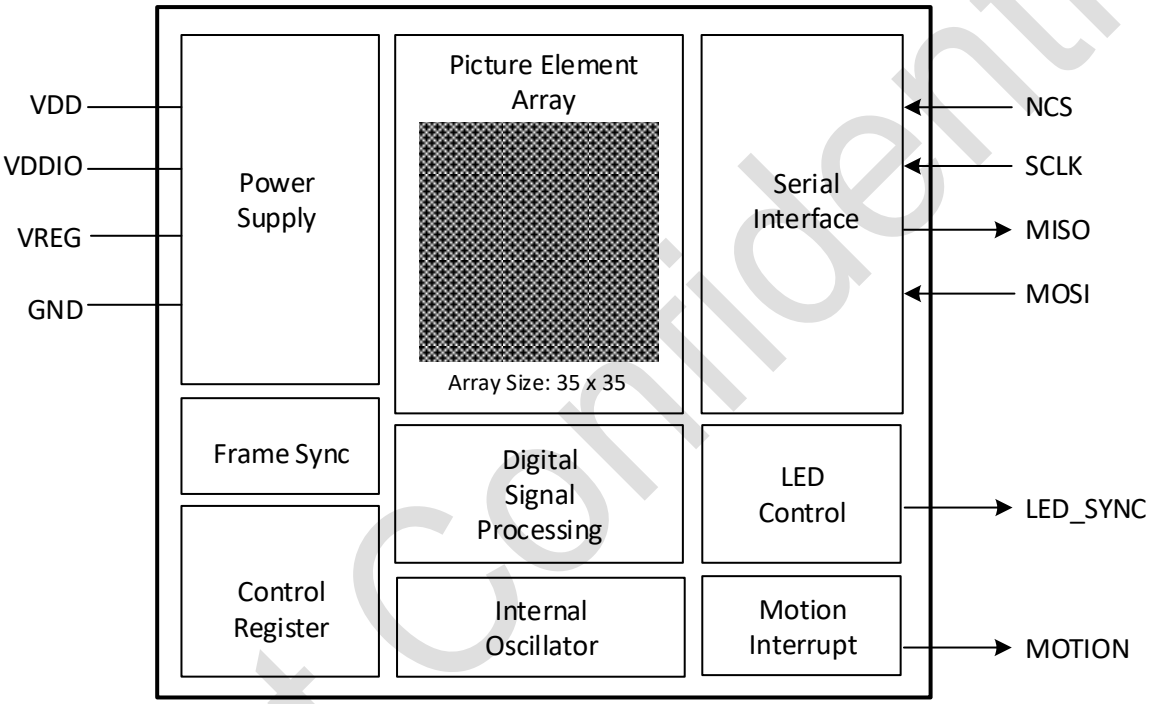


Figure 1. Block Diagram

1.3 Terminology

Term	Description
DSPS	Digital Signal Processing System
ESD	Electrostatic Discharge
LED	Light Emitting Diode
IC	Integrated Circuit
I/O	Input / Output
IR	Infrared
PCB	Printed Circuit Board
IMU	Inertial Measurement Unit
AMS	Auto Mode Switching

1.4 Pin Assignment and Signal Description

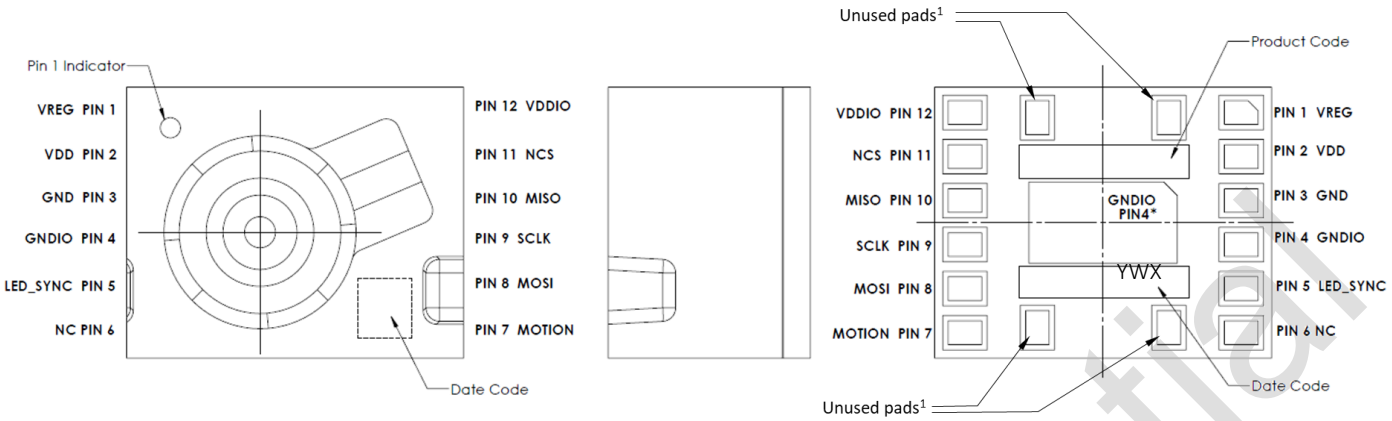


Figure 2. Pin Configuration

Table 1. Signal Pins Description

Function	Pin No.	Signal Name	Type	Description
Power Supplies	2	VDD	Power	Input power supply
	12	VDDIO	Power	I/O reference voltage
	1	VREG	Ground	Internal voltage output
	3	GND	Ground	Ground
	4	GNDIO	Ground	I/O ground
	4*	GNDIO	Ground	I/O ground pad
Control Interface	8	MOSI	Input	Serial data input
	9	SCLK	Input	Serial data clock
	10	MISO	Output	Serial data output
	11	NCS	Input	Chip select
Functional I/O	7	MOTION	Output	Motion interrupt (Active low)
	5	LED_SYNC	Output	External LED control pin (Active low).
Reserved	6	NC	NC	No Connection

Note: These unused pads must be left unconnected.

2.0 Operating Specifications

2.1 Absolute Maximum Ratings

Table 2. Absolute Maximum Ratings

Parameter	Symbol	Min.	Max.	Unit	Note
Storage Temperature	T_S	-40	85	°C	
Lead-Free Solder Temperature	T_{PEAK}		260	°C	
Supply Voltage	VDD	-0.5	2.0	V	
	VDDIO	-0.5	3.6	V	
I/O Pin Voltage		-0.5	VDDIO	V	All I/O pins
ESD	ESD_{HBM}		2	kV	All pins (Human Body Model)

Notes:

1. The above maximum ratings are the guaranteed parameters that the chip expects to be functional as per the intended design.
2. Exposure to these conditions or beyond those indicated may adversely affect device reliability and unexpected or premature functional failure.

2.2 Recommended Operating Conditions

Table 3. Recommended Operating Conditions

Parameter	Symbol	Min.	Typ.	Max.	Unit	Note
Ambient Temperature	T_A	0		60	°C	
Power Supply Voltage	VDD	1.8	1.9	2.0	V	Including supply noise
	VDDIO	1.8	1.9	3.6	V	VDDIO ≥ VDD
Power Supply Rise Time	t_R	0.15		20	ms	0 to VDD min
Supply Noise	V_{NA}			100	mV	10 kHz to 75 MHz, peak to peak
Serial Port Clock Frequency	f_{SCLK}	0.2		2	MHz	50% duty cycle, for burst read
	$f_{SCLK-RW}$	1			kHz	50% duty cycle, for individual read and write of register
Working Distance	Z	80			mm	from top of Chip to Tracking Surface[2]
Effective Viewing Angle	V_A		42		°	
Minimum Illuminance[3] (@ Crimson Carpet, Grey Vinyl & Light Grey Cement surfaces)	L_{XM0}		60		lux	Mode 0: Bright Mode
	L_{XM1}		30		lux	Mode 1: Low Light Mode (Default Mode)
	L_{XM2}		5		lux	Mode 2: Super Low Light Mode
Frame Rate	F_{RM0}		126		fps	Mode 0: Bright Mode
	F_{RM1}		126		fps	Mode 1: Low Light Mode (Default Mode)
	F_{RM2}		50		fps	Mode 2: Super Low Light Mode
Speed	S			7.4	rad/s	Mode 0 & 1

Notes:

1. PixArt does not guarantee the performance of the system beyond the recommended operating condition limits.
2. Working distance from top of chip to tracking surface, Z.

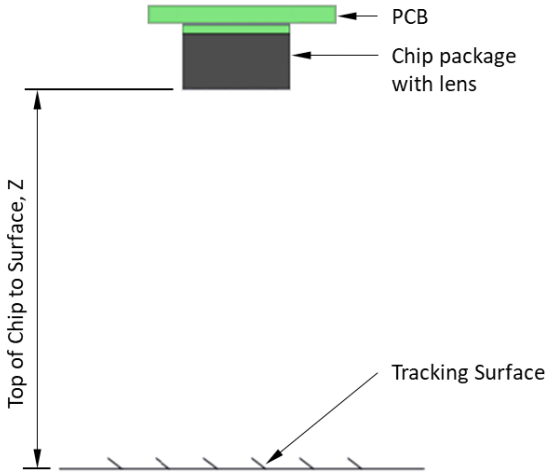


Figure 3. Cross-Sectional View of Working Distance, Z

3. In addition to visible light spectrum (lux), the chip is also sensitive to IR spectrum up to 940 nm of wavelength which aid in tracking under low light ambient condition.

2.3 DC Characteristics

Table 4. DC Electrical Specifications

Parameter	Symbol	Min.	Typ.	Max.	Unit	Note
Supply Current	I_{DD_RUN}		3.5		mA	Average current. No load on MISO, MOTION.
Power Down Current	I_{PD}		12		µA	
Input Low Voltage	V_{IL}			0.3 x VDDIO	V	SCLK, MOSI, NCS
Input High Voltage	V_{IH}	0.7 x VDDIO			V	SCLK, MOSI, NCS
Input Hysteresis	V_{I_HYS}		100		mV	SCLK, MOSI, NCS
Input Leakage Current	I_{LEAK}		± 1	± 10	µA	V_{IN}= VDDIO or 0V, SCLK, MOSI, NCS
Output Low Voltage	V_{OL}			0.45	V	I_{OUT}= 1mA, MISO, MOTION
Output High Voltage	V_{OH}	VDDIO - 0.45			V	I_{OUT}= -1mA, MISO, MOTION
Transient Supply Current	I_{DDT}			70	mA	Maximum supply current during the supply ramp from 0V to VDD with minimum 150 µs and maximum 20 ms rise time (does not include charging currents for bypass capacitors).
	I_{DDTIO}			70	mA	Maximum supply current during the supply ramp from 0V to VDDIO with minimum 150 µs and maximum 20 ms rise time (does not include charging currents for bypass capacitors).

Note: All the parameters are tested under operating conditions: VDD= 1.9V, VDDIO= 1.9V, T_A= 25°C.

2.4 AC Characteristics

Table 5. AC Electrical Specifications

Parameter	Symbol	Min.	Typ.	Max.	Unit	Note
Motion Delay After Reset	$t_{MOT-RST}$	50			ms	From reset to valid motion, assuming motion is present
Shutdown	t_{STDWN}			100	ms	From Shutdown state active to low current
Wake from Shutdown	t_{WAKEUP}	300			ms	From Shutdown state inactive to valid motion.

Note: All the parameters are tested under operating conditions: VDD= 1.9V, VDDIO= 1.9V, T_A= 25°C.

2.5 SPI Specifications

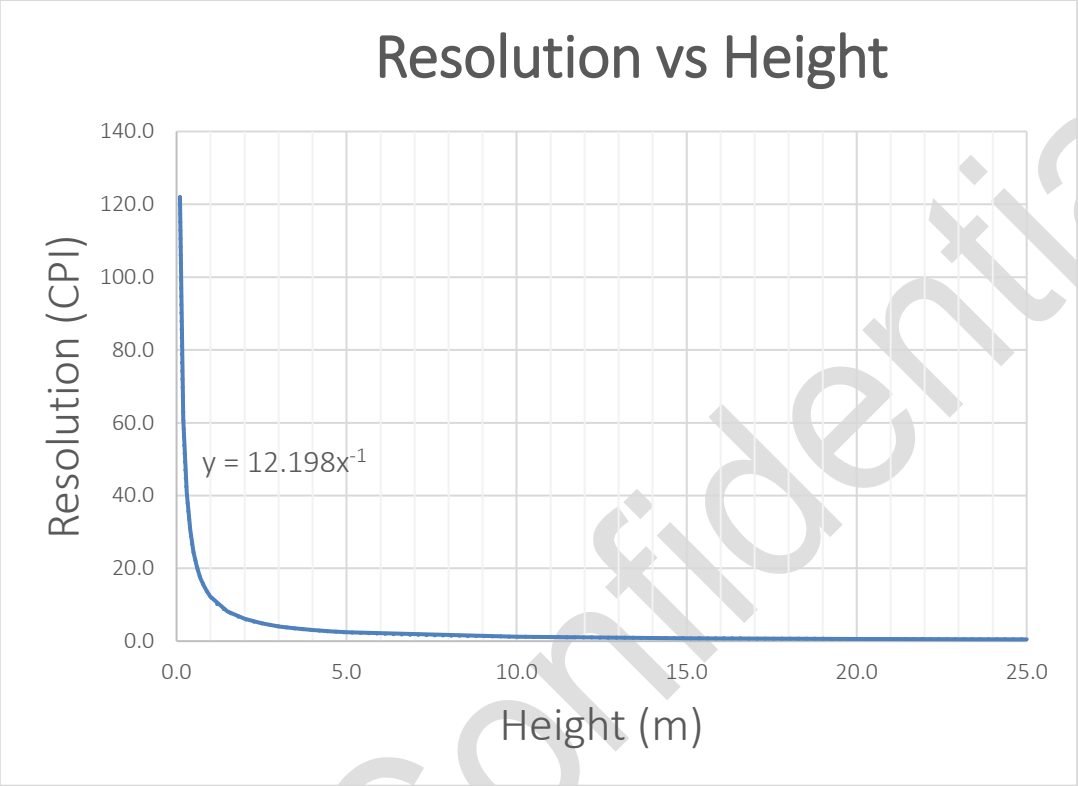
Table 6. SPI Timing Specifications

Parameter	Symbol	Min.	Typ.	Max.	Unit	Note
MISO Rise Time	$t_{r\text{-}MISO}$		50		ns	C_L = 100pF
MISO Fall Time	$t_{f\text{-}MISO}$		50		ns	C_L = 100pF
MISO Delay After SCLK	$t_{DLY\text{-}MISO}$			120	ns	From SCLK falling edge to MISO data valid, no load conditions
MISO Hold Time	$t_{hold\text{-}MISO}$	200			ns	Data held until the next falling SCLK edge
MOSI Hold Time	$t_{hold\text{-}MOSI}$	200			ns	The amount of time data is valid after SCLK rising edge
MOSI Setup Time	$t_{setup\text{-}MOSI}$	120			ns	From data valid to SCLK rising edge
SPI Time Between Write Commands	t_{SWW}	10.5			µs	From rising SCLK for the last bit of the first data byte to rising SCLK for the last bit of the second data byte.
SPI Time Between Write and Read Commands	t_{SWR}	6			µs	From rising SCLK for the last bit of the first data byte to rising SCLK for the last bit of the second address byte.
SPI Time Between Read and Subsequent Commands	t_{SRW} t_{SRR}	1.5			µs	From rising SCLK for the last bit of the first data byte to falling SCLK for the first bit of the address byte of the next command.
SPI Read Address-Data Delay	t_{SRAD}	2			µs	From rising SCLK for the last bit of the address byte to falling SCLK for the first bit of data being read.
NCS Inactive After Motion Burst	t_{BEXIT}	500			ns	Minimum NCS inactive time after motion burst before next SPI usage
NCS To SCLK Active	$t_{NCS\text{-}SCLK}$	120			ns	From last NCS falling edge to the first SCLK rising edge
SCLK To NCS Inactive (For Read Operation)	$t_{SCLK\text{-}NCS}$	120			ns	From the last SCLK rising edge to NCS rising edge, for valid MISO data transfer
SCLK To NCS Inactive (For Write Operation)	$t_{SCLK\text{-}NCS}$	2			µs	From the last SCLK rising edge to NCS rising edge, for valid MOSI data transfer
NCS To MISO High-Z	$t_{NCS\text{-}MISO}$			500	ns	From NCS rising edge to MISO high-Z state
MOTION Rise Time	$t_{r\text{-}MOTION}$		50		ns	C_L = 100pF
MOTION Fall Time	$t_{f\text{-}MOTION}$		50		ns	C_L = 100pF
Input Capacitance	C_{in}		50		pF	SCLK, MOSI, NCS
Load Capacitance	C_L			100	pF	MISO, MOTION

Note: All the parameters are tested under operating conditions: VDD= 1.9V, VDDIO= 1.9V, T_A= 25°C.

2.6 Resolution versus Height Chart

This chart serves as a reference for resolution count with its corresponding height based on the default setting. The resolution can be changed via register *RESOLUTION* (Section 8.2.2), with a maximum of 6x of the chart below.*



Note: Interpolation is applied to resolution count beyond 2 m.

Figure 4. Resolution versus Height Chart

* Chart is based on the *RESOLUTION* register (0x4E) written with the value of 0x2A after loading Initialization Register Setting in Section 7.1.2.

The complete formula, including *RESOLUTION* register, is as shown below:

$$CPI = \frac{12.198}{\text{Height (m)}} \text{x } (RESOLUTION + 1) \text{x} \frac{200}{8600}$$

At height= 1m,

Example 1:

With Resolution = 42 // value 0x2A in *RESOLUTION* register

CPI = 12.198/1 x (42 + 1) x (200/8600) // where (42 + 1) x (200/8600) = 1x value

= 12.198/1 x 1 // from the data point of the chart in Figure 4

≈ 12

Example 2:

With Resolution = 255 // With value 0xFF in *RESOLUTION* register

CPI = 12.198/1 x (255 + 1) x (200/8600) // where (255 + 1) x (200/8600) = 5.95 ≈ 6x value
= 12.198/1 x 5.95 // from the data point of the chart in Figure 4
≈ 72

To convert count to actual travel distance, divide value of 1 by the CPI

$$Travel\ distance\ per\ count = \frac{1}{CPI}\ (\text{inch})$$

For example,

At height= 1m, RESOLUTION register = 42 and CPI = 12.198:

Travel distance per count = 1/12.198 ≈ 0.082 inch
= 0.082 x 25.4
= 2.0828mm

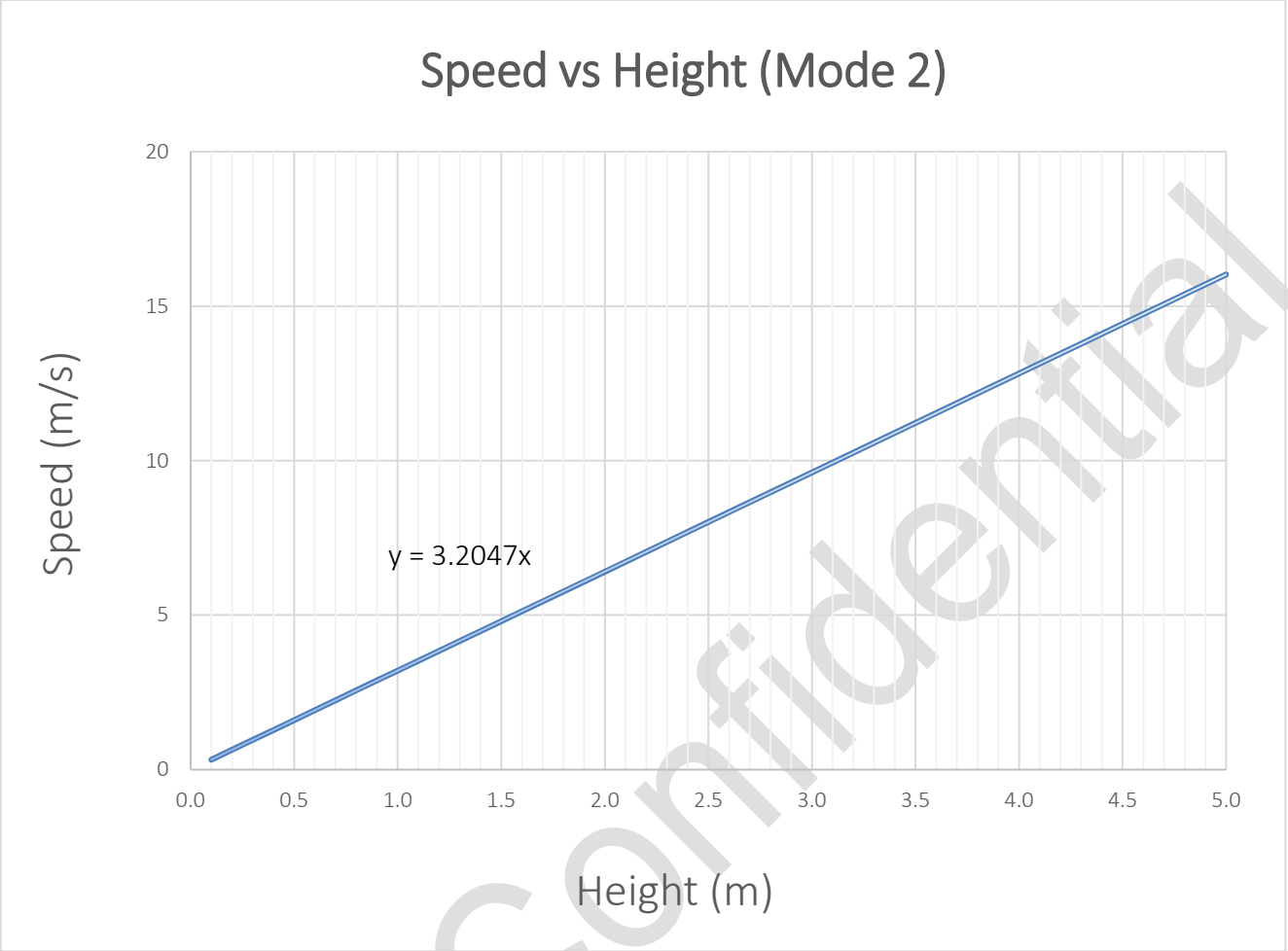
At height of 1m, 1 count of reported motion represent 2.0828mm of distance travelled.

2.7 Speed versus Height Chart

These charts serve as a theoretical reference of speed capability with its corresponding height for Mode 0 & 1 and Mode 2.

Note: Interpolation is applied beyond 0.5 m.

Figure 5. Speed versus Height Chart (Mode 0 & 1)

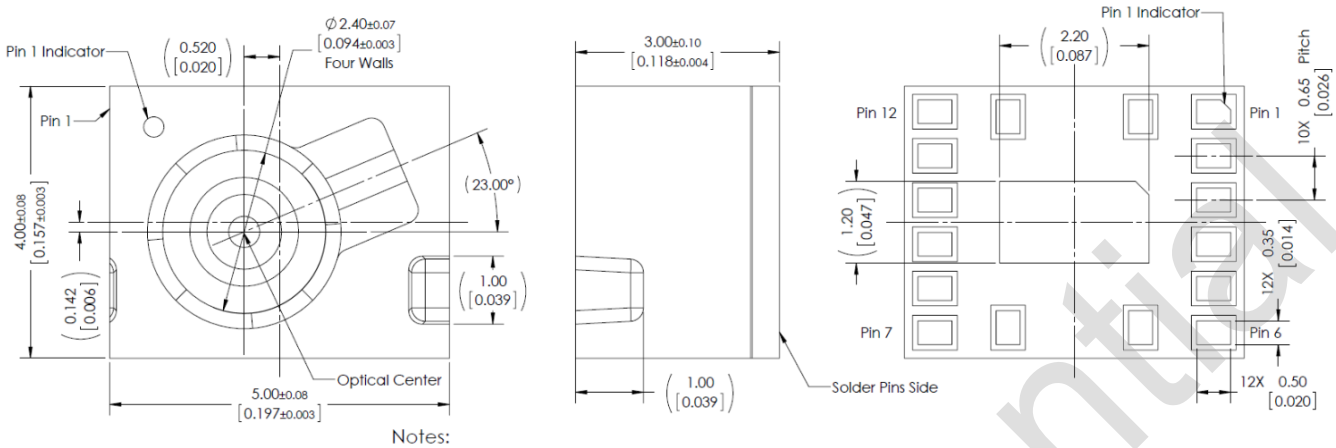


Note: Interpolation is applied beyond 0.5 m.

Figure 6. Speed versus Height Chart (Mode 2)

3.0 Mechanical Specifications

3.1 LGA Package Outline Drawing



Notes:
1. Dimensions in Milimeters [Inches]
2. Coplanarity of Pads: 0.08 mm
3. Non-cumulative pitch tolerance: ±0.10 mm
4. Maximum flash: 0.20 mm max
5. Dimensional tolerance: ±0.10 mm unless otherwise stated
6. Package referance: 12L-5X4-LGA_001

Figure 7. LGA Package Outline Drawing

3.2 Package Marking

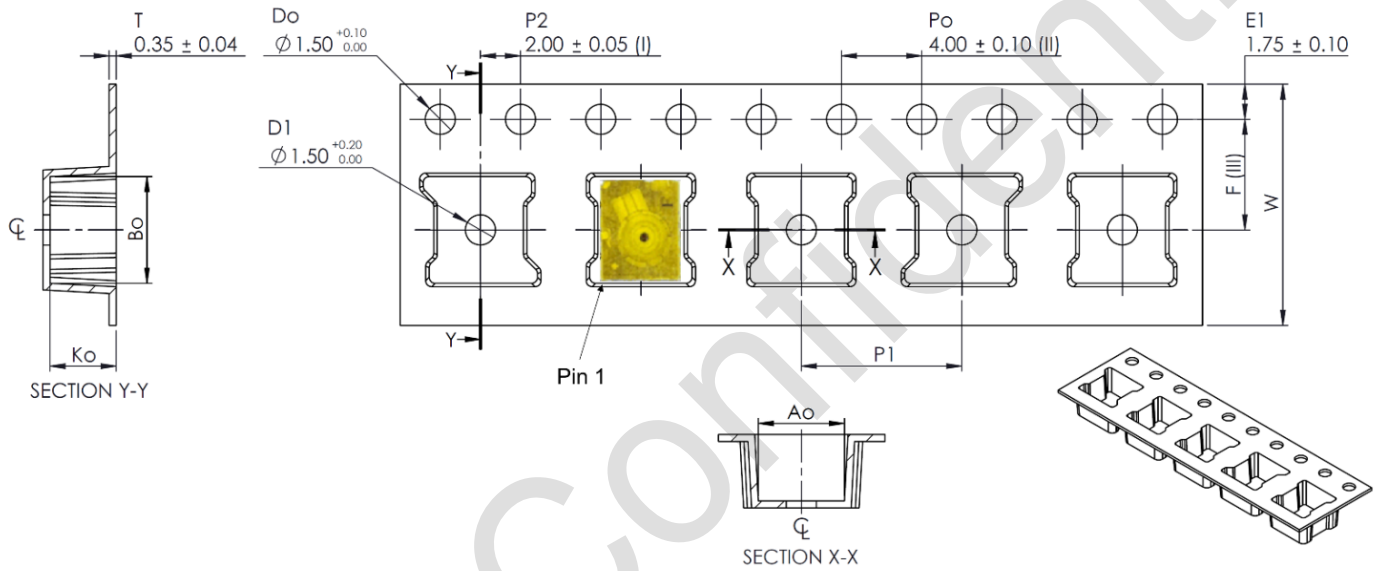
Refer to Figure 2. Pin Configuration for the code marking the location on the device package.

Table 7. Code Identification

Code	Marking	Description	Note
Product Number	P3905	Part number label	Marking on bottom of package
Date Code	YWXX	Y = Assembly house + Year W = Work Week XX = PixArt reference	Marking on bottom of package
Date Code	W	W = Week	Marking on top of package

3.3 Packing Information

Item	Description
Product number	PAA3905E1-Q
Package type	12-pin LGA
Quantity per reel	500 pcs
Inner box quantity	500 pcs [1 reel per inner box]
Shipping box quantity	2,500 pcs [5 inner box per shipping box]
Reel size	Ø178mm (7 inch)
Inner box size	270 x 270 x 30 mm^3
Shipping box size	295 x 290 x 190 mm^3



Ao	4.30	+/- 0.10
Bo	5.30	+/- 0.10
Ko	3.30	+/- 0.10
F	5.50	+/- 0.05
P1	8.00	+/- 0.10
W	12.00	+0.30 / -0.10

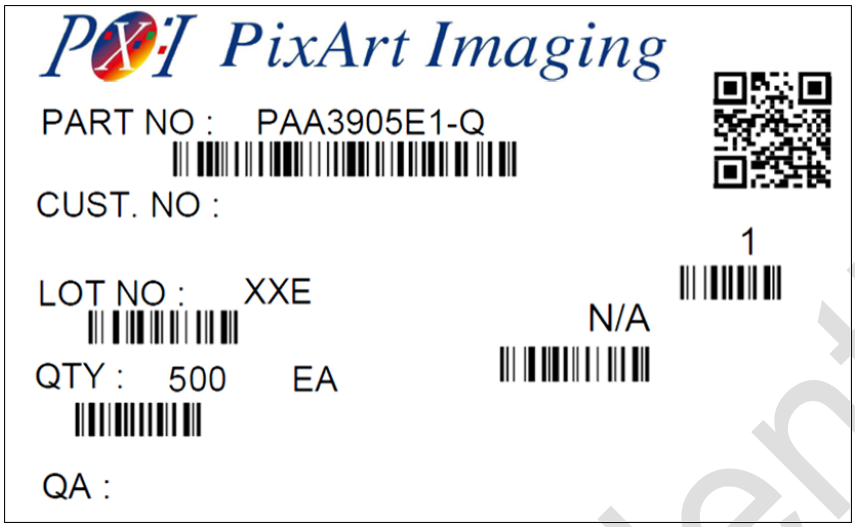
(I) Measured from centreline of sprocket hole to centreline of pocket.
(II) Cumulative tolerance of 10 sprocket holes is ± 0.20 .
(III) Measured from centreline of sprocket hole to centreline of pocket.
(IV) Other material available.
(V) Dimension with ( ) is used for design reference purposes, No measurement required.

ALL DIMENSIONS IN MILLIMETRES UNLESS OTHERWISE STATED.

Figure 8. Carrier Tape Dimension and Pin 1 Location

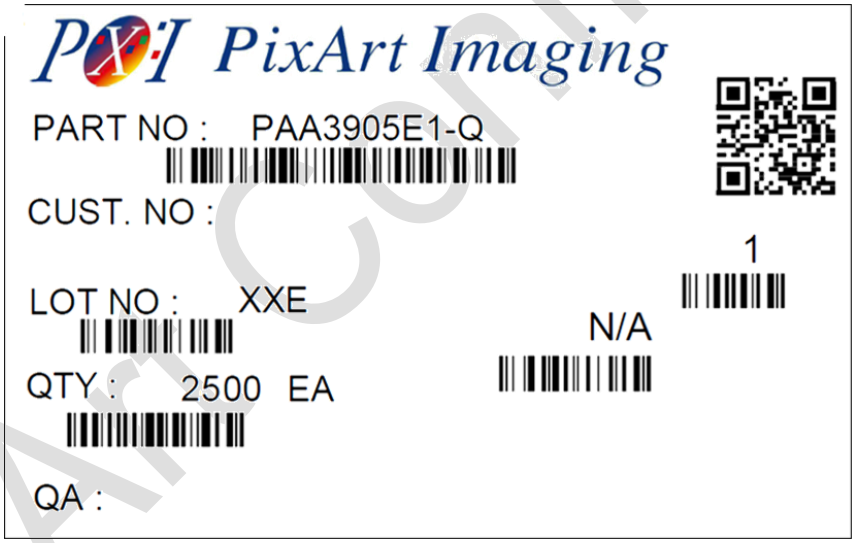
3.4 Package Handling Information

3.4.1 Sample of Inner Box Label



Note: This label is used on the inner box

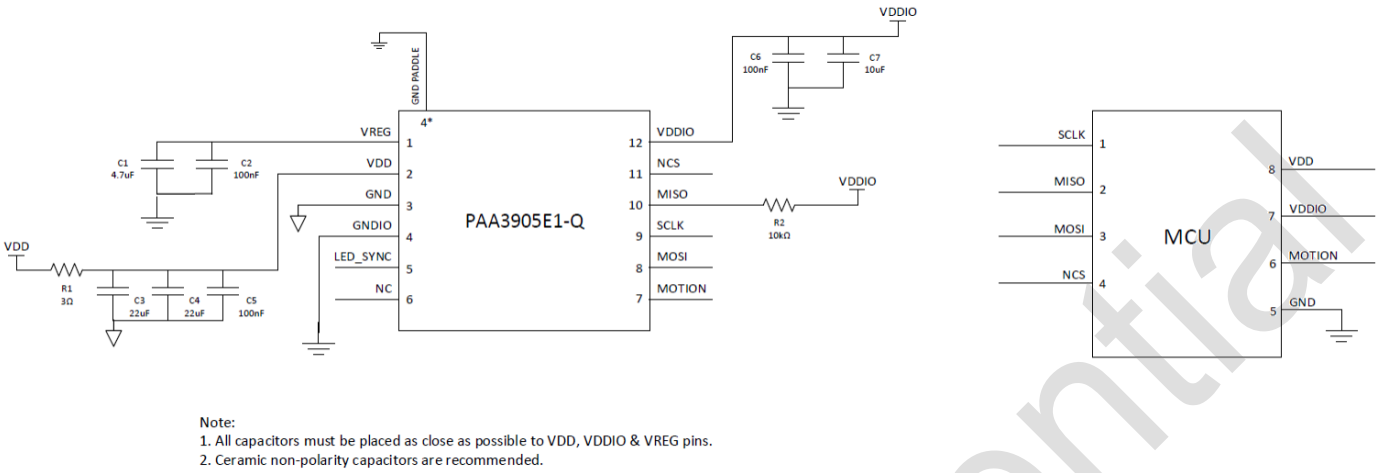
3.4.2 Sample of Shipping Box Label



Note: This label is used on the shipping box

4.0 Design Reference

4.1 General Reference Schematic

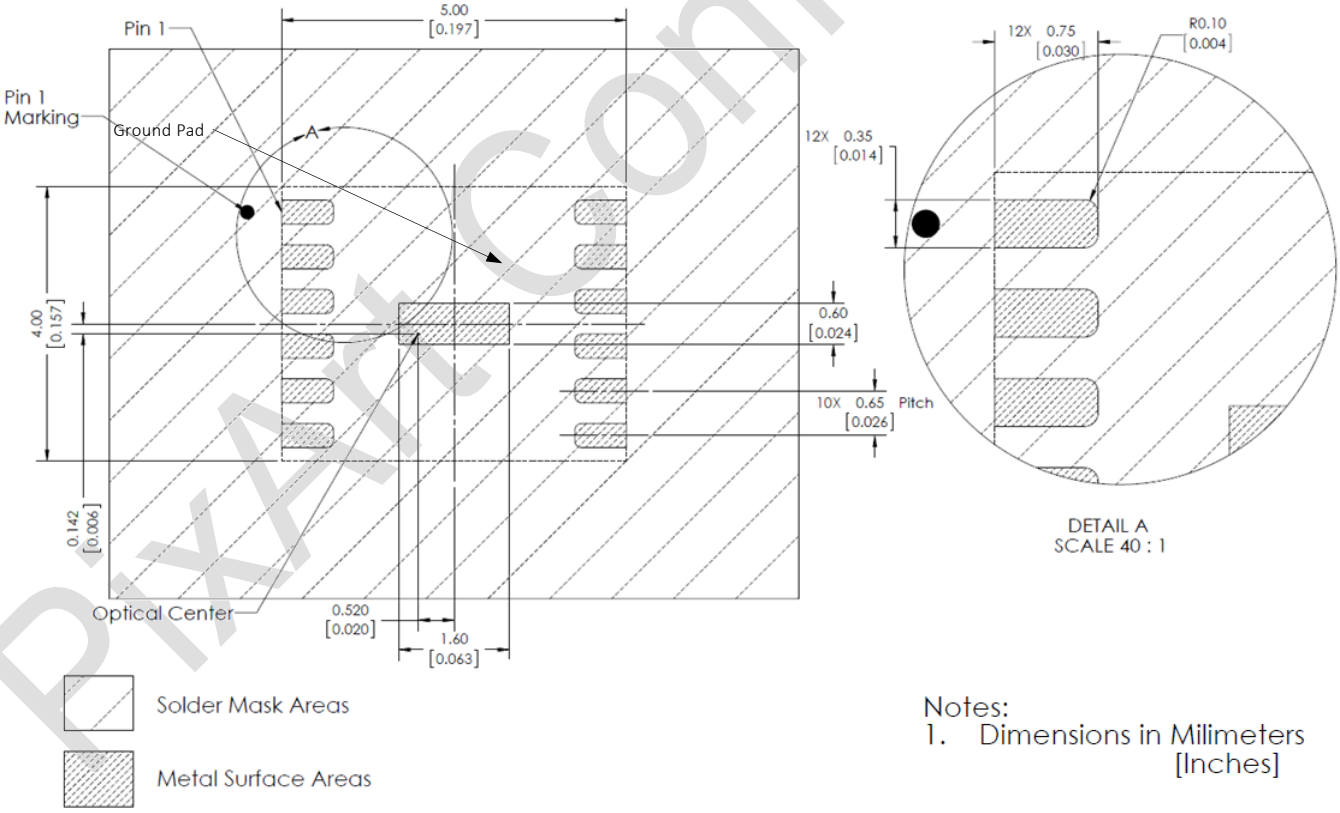


Note:
1. All capacitors must be placed as close as possible to VDD, VDDIO & VREG pins.
2. Ceramic non-polarity capacitors are recommended.

Figure 9. Reference Schematic

4.2 PCB Layout Design Guide

4.2.1 Recommended PCB Footprint



Note: The bottom ground pad of the LGA package must be connected to the circuit ground.

Figure 10. Recommended PCB Footprint

4.3 Assembly Guide

Proper grounding especially handling and assembly of the components in the PixArt manufacturing process is required to prevent the chip from being damaged prematurely or degradation of the chip performance.

4.3.1 IR Reflow Soldering Profile

- Surface mount the chip and all other electrical components onto PCB.
- Reflow the entire assembly in a no-wash solder process.

Note: It is recommended to generate a stencil profile for the reflow process.

- Remove the protective Kapton tape on top of the chip's package.

Note: Avoid contamination of the cavity surface.

- Recommend to hold the PCB assembly vertically when removing Kapton tape.

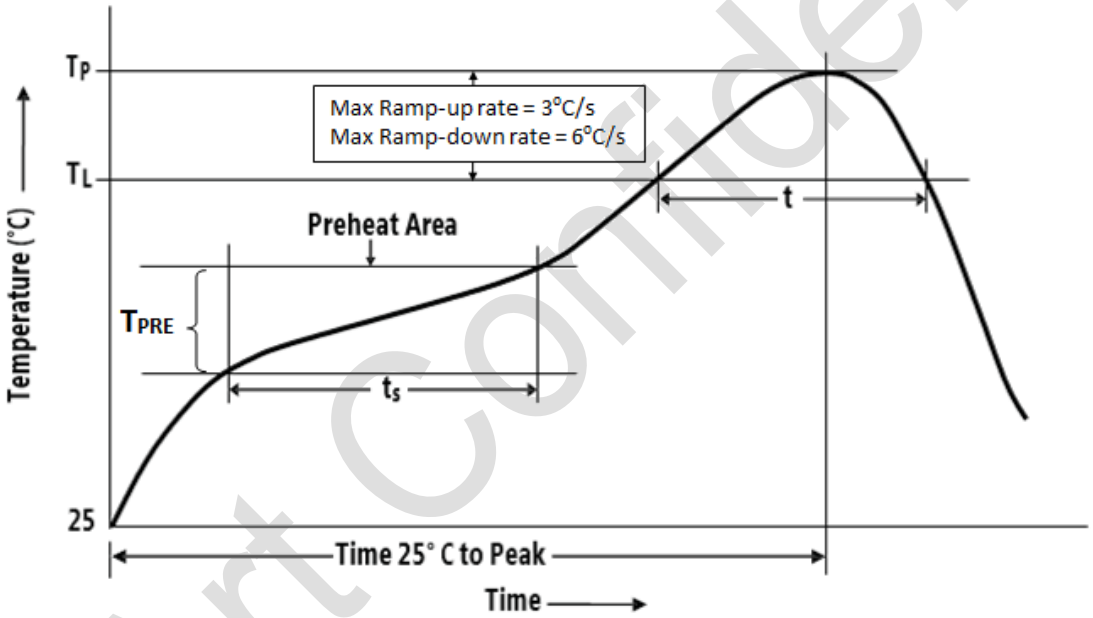


Figure 11. Solder Reflow Profile

Table 8. Reflow Profile

Profiling Information	Symbol	Min	Max	Unit	Note
Preheat Area Temperature	T_{PRE}	150	200	°C	
Preheat Area Duration	t_S	60	120	sec	From 150 to 200°C
Melting Duration	t	60	150	sec	T ≥ 217°C
Liquids Melting Temperature	T_L	217		°C	
Peak Temperature	T_P	230	260	°C	
Ramp-up rate	T_{RAMP_UP}		3	°C/sec	From T_L to T_P
Ramp-down rate	T_{RAMP_DOWN}		6	°C/sec	From T_P to T_L
Max. Time 25°C to Peak			5	min	

4.3.2 Assembly Recommendation

1. Place the PCB assembly horizontally with the package cavity facing up. Insert the lens set with plastic or soft-tip tweezers onto the optical aperture (the chip's cavity).
2. Insert the lens set onto the chip package cavity (the hole on the chip's package).
3. Use an appropriate flat tip jig to press the lens barrel vertically onto the upper cavity of the chip's package.
4. Insert the nozzle of the glue dispenser vertically inside the gluing slots and dispense glue appropriately.
5. Remove the nozzle of the glue dispenser and let the glue cure properly.

Note:

1. No lens calibration is required.
2. Refer to L242-ZSZ1 lens set datasheet for more information and detailed steps of the assembly process.

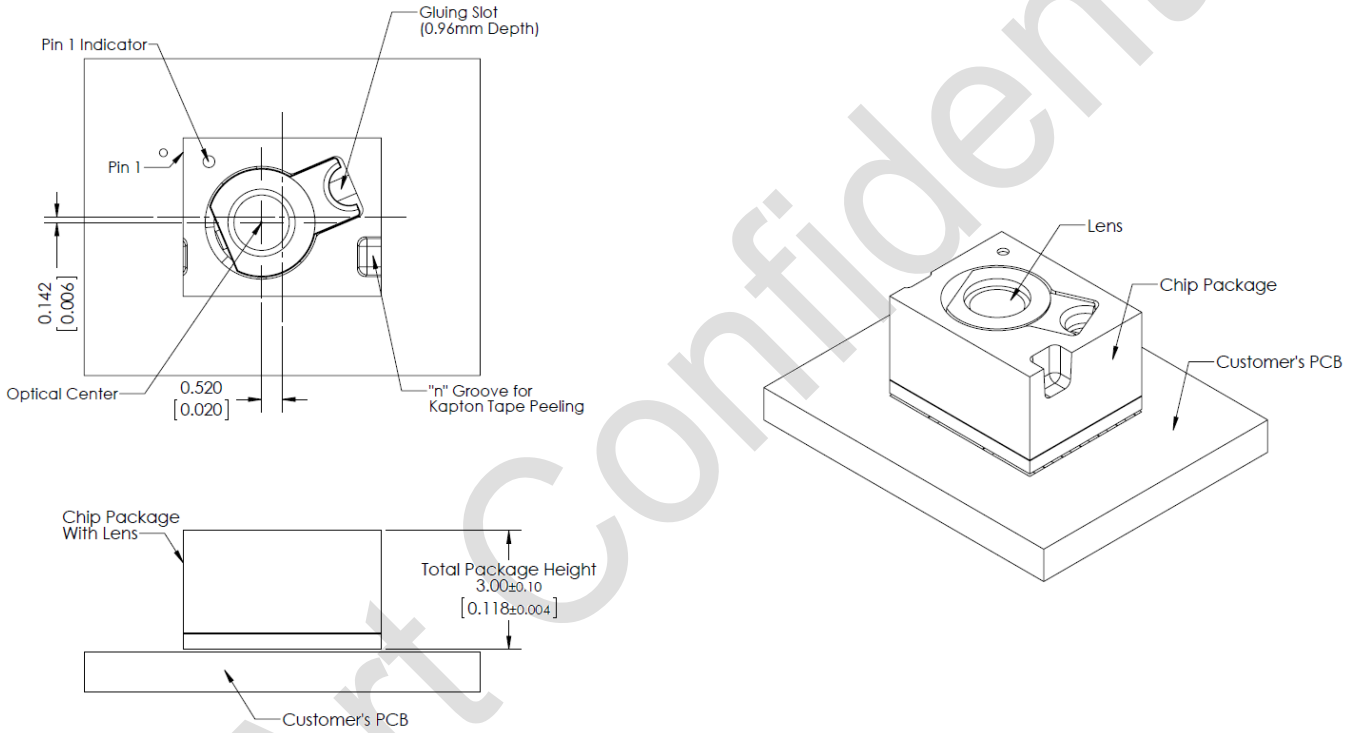


Figure 12. System Assembly View with L242-ZSZ1

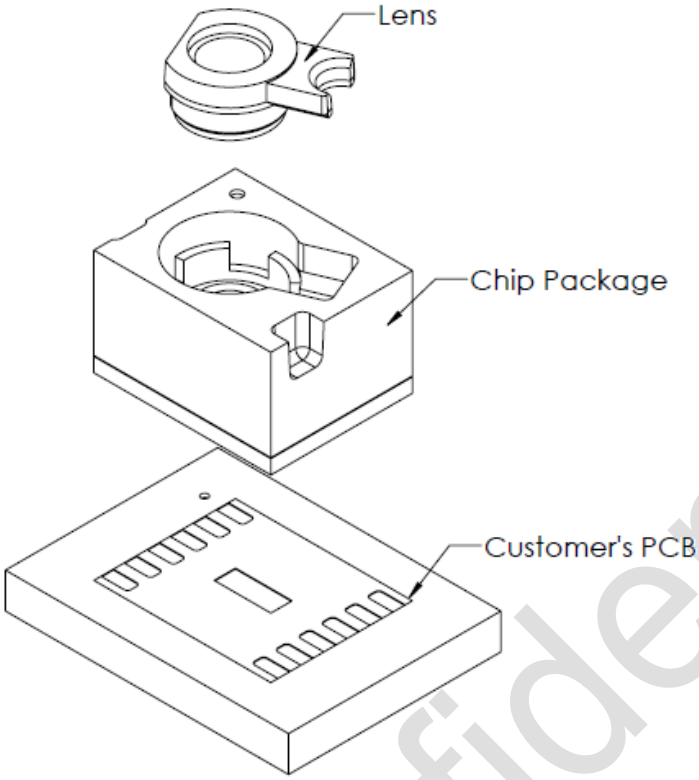


Figure 13. Exploded View of System Assembly

4.3.3 Manual Rework of Chip Assembly

This is an optional step.

For the de-solder chip package from PCB, do disassemble the lens set first as it melts under the soldering heat.

Do note below precautions for rework of the chip assembly.

1. Pry up the lens set from the chip package cavity from the gluing slot using a tweezer.

Note: It is important to remove the lens set as it may melt under the soldering heat.

2. Place the Kapton tape across the top of the chip package cavity to avoid contamination
3. Perform de-soldering and soldering activities as needed.
4. Remove Kapton tape and insert the lens set as outlined in the above section.

5.0 Power Management

5.1 Power Supply

The chip has two power supply inputs (VDD and VDDIO). VDD is the main power supply and VDDIO is the I/O reference voltage.

5.2 Power Sequence

5.2.1 Power on Sequence Requirement

VDDIO needs to be powered up earlier than VDD, the interval time between VDD and VDDIO must be less than 100ms.

Figure 14. Power-up Sequence Requirement

5.2.2 Power off Sequence Requirement

It is recommended to power off both VDD and VDDIO at the same time or the VDD first, followed by VDDIO.

5.3 Power State

5.3.1 States Description

Table 9. States Description

State	Description
OFF	No power supply and all the voltage rails and clocks are gated.
RUN	The chip enters RUN State after the regular power on and initialization.
Shutdown	The chip enters this state upon receiving the host command for power-saving purposes and all the parameter settings are still intact. The chip exits the shutdown state and transitions back to the RUN state upon receiving the host command. To exit the Shutdown state, only the power on command (write *Shutdown* register with value 0xC7) is accepted.

The table below shows the state of various pins during the shutdown.

Table 10. State of Signal Pins during Shutdown.

Pin	Status during Shutdown State
NCS	High [1]
MISO	Hi-Z [2]
SCLK	Ignore if NCS= 1 [3]
MOSI	Ignore if NCS= 1 [4]
MOTION	Output High

5.3.2 State Diagram

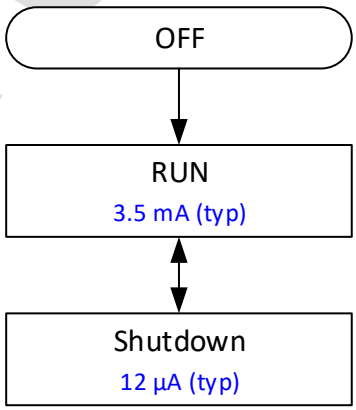


Figure 15. Power State Diagram

5.3.3 State Transition

State	Type	Description
OFF to RUN	Power-up	After the chip is powered on and initialization is completed, the chip enters the RUN state automatically.
RUN to Shutdown	Command	The host to send a command to enter the Shutdown state
Shutdown to RUN	Command	The hose to send command to exit the Shutdown state

- RUN to Shutdown

Write register 0x3B with value 0xB6 // Enter Shutdown state

- Shutdown to RUN

Step in sequence as follows:

1. Drive NCS to high state at least for 0.1ms to reset the SPI port[1]. // To reset the SPI port
2. Write register 0x3B with value 0xC7 // Exit Shutdown state command
3. Wait for at least 1ms.
4. Write register 0x3B with value 0x00
5. Wait for at least 1ms.
6. Read from registers 0x02, 0x03, 0x04, 0x05 and 0x06 // read once to clear the registers

Note:

1. If NCS is already at high state, then skip step 1.

5.4 Reset

The table below shows the state of the various pins during power-up and reset.

Table 11. State of Signal Pins during Power-Up and Reset

State of Signal Pins after VDD is Valid		
Pin	**During Reset**	**After Reset**
NCS	Ignored	Functional
MISO	Undefined	Depends on NCS
SCLK	Ignored	Depends on NCS
MOSI	Ignored	Depends on NCS
MOTION	Undefined	Functional

5.4.1 Power on Reset

During power-on, the chip does not need an external reset as there is an internal circuitry that performs the power-on reset function for the chip.

5.4.2 Software Reset

Write 0x5A to the *Power_Up_Reset* register to reset the chip. All register settings will be reverted to default values, and the chip is required for re-initialization.

5.5 Related Registers

Usage	Name	Bank	Address
Power Management	*Shutdown Register*	0	0x3B
	Power_Up_Reset	0	0x3A

- Shutdown Register

Register Name	Shutdown		
Bank	0	**Address**	0x3B
Access	W	**Default Value**	-
Description	Shutdown state control register. Refer to Section 5.3.3 for the details.		

- Reset Register

Register Name	Power_Up_Reset		
Bank	0	**Address**	0x3A
Access	W	**Default Value**	-
Description	Write 0x5A to this register to reset the chip. All settings will revert to default values. Reset is required to restore normal operation after Raw Data Output.		

6.0 Control Interface

The chip adopts Serial Port Interface Communication (SPI) interface. The interface is configured to mode 3 (CPHA= 1 and CPOL= 1) for slave device communication. The maximum SCLK speed is 2MHz.

6.1 Signal Description

The 4-wire synchronous serial port interface is used to write or read registers in the chip. The host is an SPI master device where it drives SCLK, MOSI, and NCS (active low) to initiate the communication. The interface supports single or multi-chip (slaves) with NCS control. For clock polarity and phase, the chip is operating in SPI mode 3.

Table 12. Signal Description

Pin	Reset Status	Description
SCLK	High	Clock input, generated by the host (master).
MOSI	High	Input data (Master Out / Slave In).
MISO	Hi-Z	Output data (Master In / Slave Out).
NCS	High	Chip select input (active low).

6.2 Chip Select Operation

The serial port is activated after NCS is asserted. If NCS is de-asserted during a transaction, the entire transaction is aborted and the serial port will be reset. After a transaction is aborted, the host needs to follow the normal address-to-data or transaction-to-transaction delay timing requirement before beginning the next transaction.

When the NCS pin is high, the SPI's inputs are ignored and the SPI's output is in tri-state. NCS can also be used to reset the serial port communication in case of an error occurred.

To improve communication reliability, all serial transactions should be framed by NCS assertion. The port should not be remained enabled when not in use to prevent misinterpretation of ESD and EFT/B events and put the chip into an unknown state.

6.3 Protocol

The transmission protocol is a 4-wire link, the half-duplex protocol between the host and the chip. All data changes at SCLK falling edge and latched at SCLK rising edge. The host controller always initiates communication and the chip never initiates data transfers.

The transmission protocol consists of two types of operations:

- Write Operation
- Read Operation

Both the two operation modes consist of two bytes. The first byte contains the registered address (7-bit) and bit[7] as MSB to indicate data direction. The second byte contains the data.

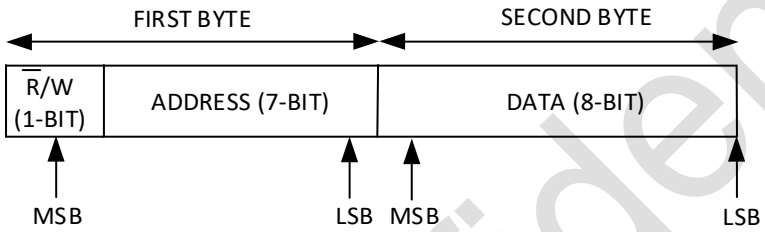


Figure 5. Transmission Protocol

6.4 Write Operation

The write operation consists of two bytes package. The first byte contains the 7-bit register address and a “1” in bit [7] indicates the data written directly from the host to the chip. The second byte contains the data to be written into the specified register. The chip latches MOSI data on rising edges of the SCLK.

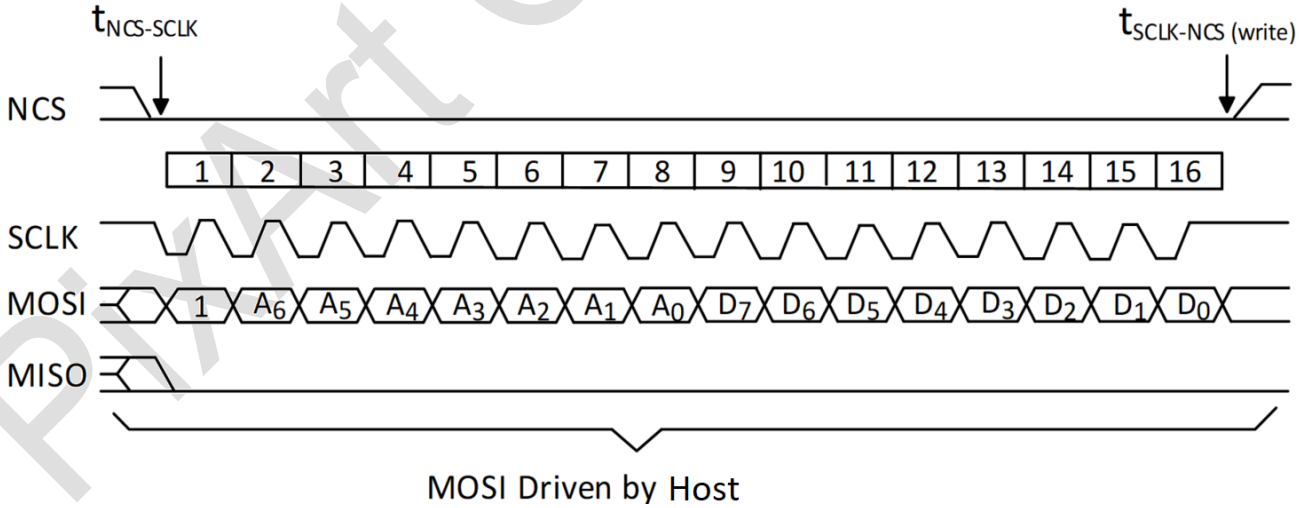


Figure 16. Write Operation

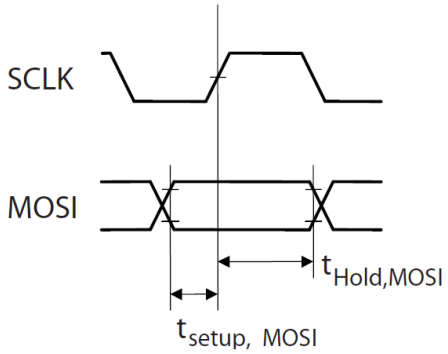


Figure 17. MOSI Set-up and Hold Time

6.5 Read Operation

The read operation consists of two bytes package. The first byte contains the 7-bit register address and a "0" in bit[7] indicates the data read directly from the chip to the host and the chip samples MOSI bits on every rising edge of SCLK. The second byte contains the data from the specified register and the chip outputs MISO bits on falling edges of SCLK.

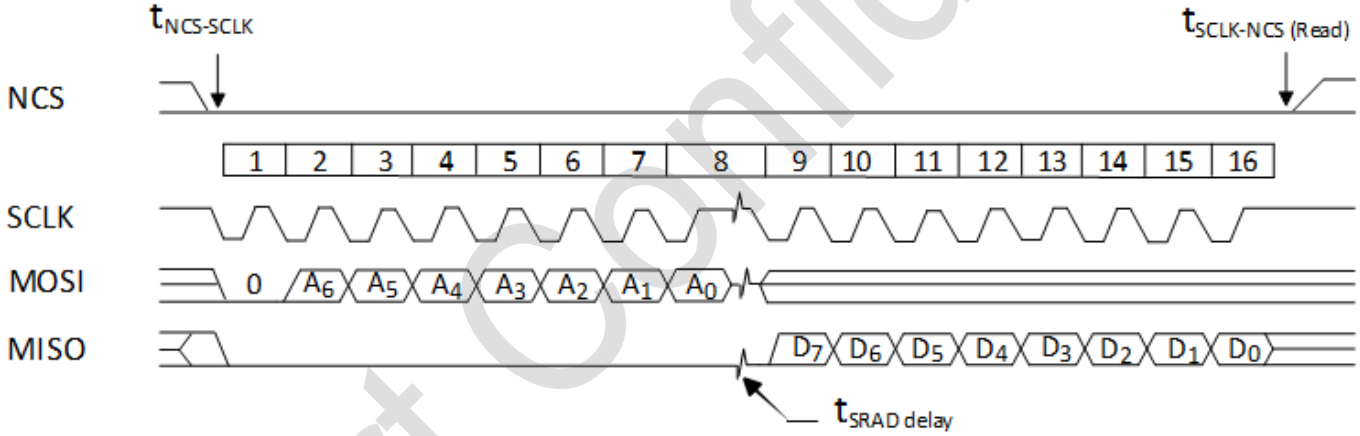


Figure 18. Read Operation

The minimum high state of SCLK is also the minimum MISO data hold time of the chip. Since the falling edge of SCLK is the start of the next read or write command, the chip will hold the state of data on MISO until the falling edge of SCLK.

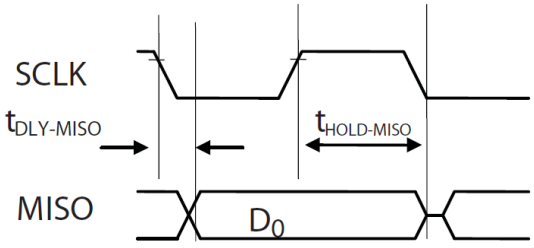


Figure 19. MISO Delay and Hold Time

6.6 Burst Read

Burst read is a special serial port interface operation mode that can be used to reduce the serial transaction time for Motion Read. The speed improvement is achieved by continuous data clocking to read multiple registers with a single dedicated address and not requiring the delay period between data bytes. Refer to Section 7.3.1 for the detail.

NCS must be de-asserted to terminate burst mode after each burst read operation transaction is completed. However, the host can terminate the Burst Read operation as needed by de-assert the NCS.

After sending the register address, the host must wait for t_{SRAD}, and then begin reading data. All data bits can be read with no delay between bytes by driving SCLK at the normal rate. The data are latched into the output buffer after the last address bit is received. After the burst transmission is completed, the host NCS is de-assert for at least t_{BEXIT} to terminate burst read operation. The serial port is not available for use until the assertion of NCS.

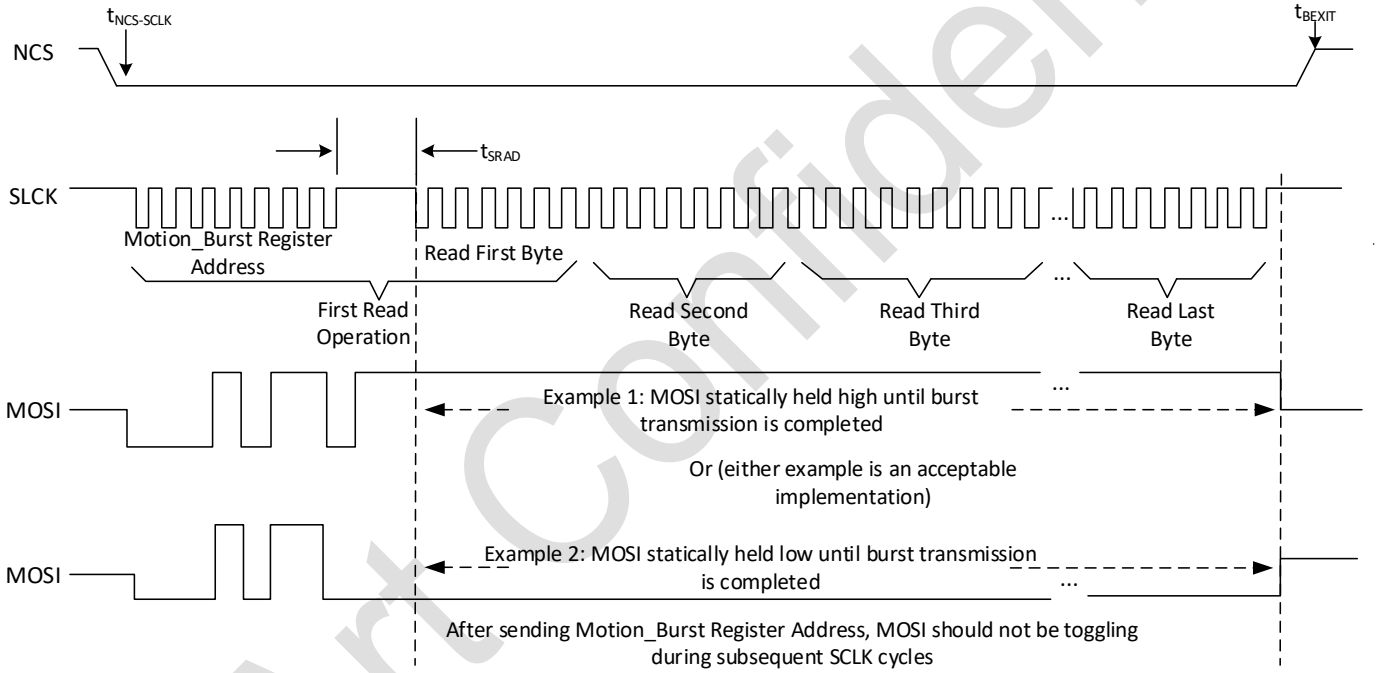


Figure 20. Burst Read Timing

6.7 Required Timing between Read and Write Commands (t_{SXX})

There are minimum timing requirements between read and write commands on the serial port.

If the rising edge of the SCLK for the last data bit of the second write command occurs before the t_{SWW} delay, then the first write command may not complete correctly.

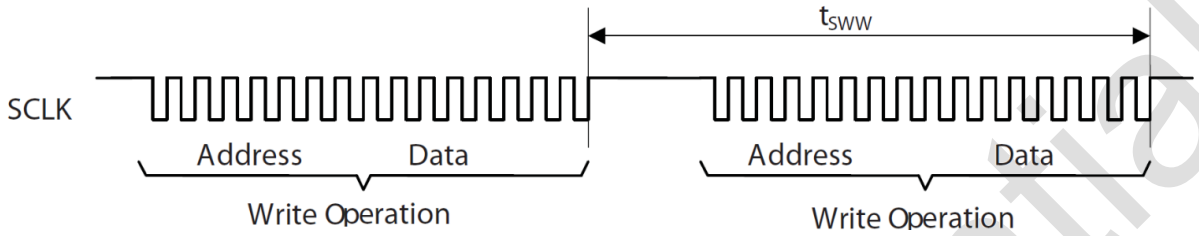


Figure 21. The timing between Two Write Commands

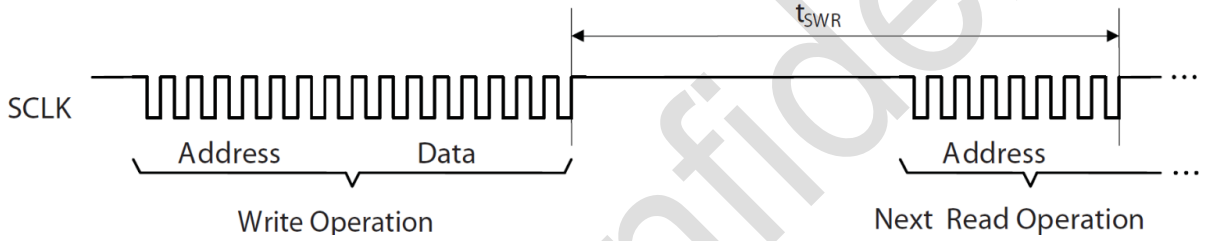


Figure 22. The Timing between Write and Read commands

If the rising edge of SCLK for the last address bit of the read command occurs before the t_{SWR} required delay, the write command may not complete correctly. During a read operation, SCLK should be delayed at least t_{SRAD} after the last address data bit to ensure that the chip has time to prepare the requested data.

The falling edge of SCLK for the first address bit of either the read or write command must be at least t_{SRR} or t_{SRW} after the last SCLK rising edge of the last data bit of the previous read operation. In addition, during a read operation, SCLK should be delayed after the last address data bit to ensure that the chip has time to prepare the requested data.

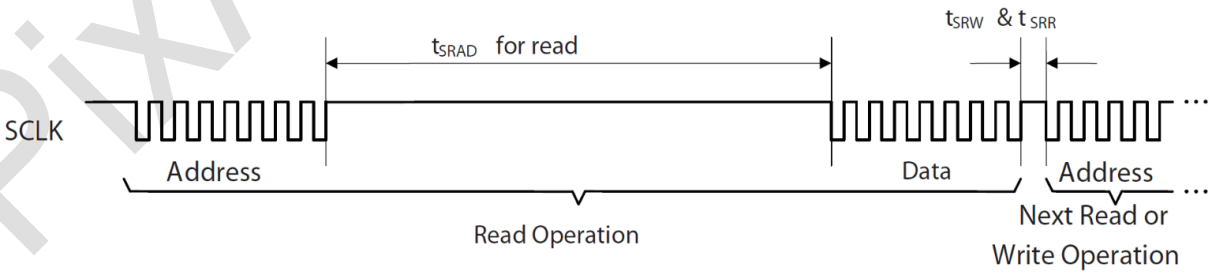


Figure 23. The timing between Read and either Write or subsequent Read commands

7.0 System Control

7.1 System Initialization

7.1.1 Initial Flow

Although the chip performs an internal self-power on reset, it is still recommended that the *Power_Up_Reset* register is written every time power is applied. The appropriate sequence is as follows:

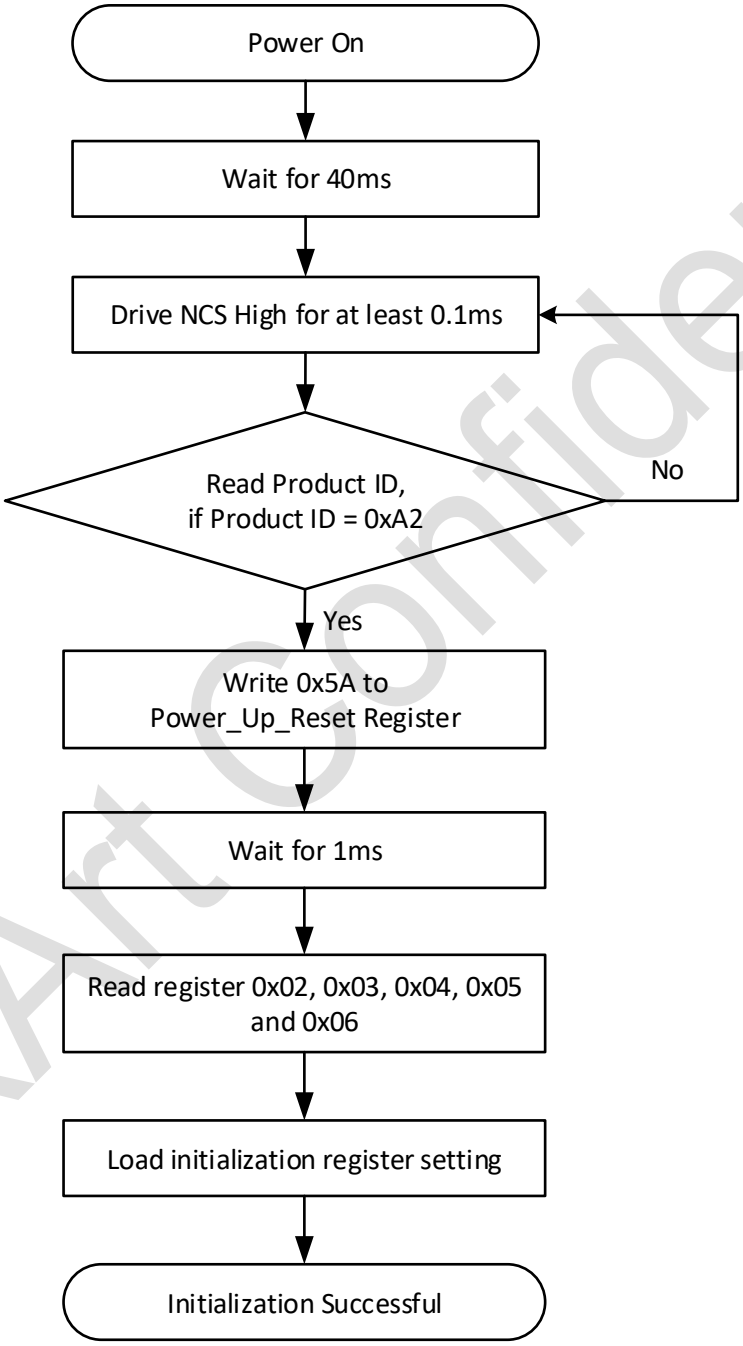


Figure 24. System Initialization Flow Chart

Note: If unable to read the correct Product ID, it is recommended to power off and power on the Module again. If the problem persists, do check the SPI interface connection and settings.

1. Apply power to VDDIO first and followed by VDD, with a delay of no more than 100ms in between each supply. Ensure all supplies are stable.
2. Wait for at least 40 ms.
3. Drive NCS to high state at least for 0.1ms to reset the SPI port[1]. The SPI port is always in a reset state whenever NCS is held high.
4. Read the *Product_ID* register to verify the PID before any other register read or write.
5. Write 0x5A to the *Power_Up_Reset* register.
6. Wait for at least 1 ms.
7. Read from registers 0x02, 0x03, 0x04, 0x05 and 0x06 one time regardless of the motion pin state to clear the motion data.
8. Refer to Section 7.1.2 to configure the needed registers to achieve optimum performance of the chip.

Note: If NCS is already at high state, then skip step 3.

7.1.2 Initialization Register Setting

Upon power-up of the chip, there are several registers to be configured to initialize the chip. There are two types of Detection Mode Setting, Standard Detection Mode Setting, and Enhanced Detection Mode Setting.

- A standard Detection Setting is recommended for general tracking operations. In this mode, the chip can detect when it is operating over a striped, checkerboard, and glossy tile surfaces where tracking performance is compromised.
- Enhance Detection Setting has better detection sensitivity to challenging conditions. However, this setting reduces tracking performance, compared to the Standard Detection Setting, as it is more sensitive to trigger motion cut-off. Thus, it is recommended to use this setting where yaw motion detection is required, and more sensitive challenging condition detection is required. The recommended operating height must be greater than 15 cm to avoid false detection in challenging conditions due to an increase in sensitivity.

- Standard Detection Setting

Recommend to perform software reset before loading below mode switching.

1. Write register 0x7F with value 0x00
2. Write register 0x51 with value 0xFF
3. Write register 0x4E with value 0x2A
4. Write register 0x66 with value 0x3E
5. Write register 0x7F with value 0x14
6. Write register 0x7E with value 0x71
7. Write register 0x55 with value 0x00
8. Write register 0x59 with value 0x00
9. Write register 0x6F with value 0x2C
10. Write register 0x7F with value 0x05
11. Write register 0x4D with value 0xAC
12. Write register 0x4E with value 0x32
13. Write register 0x7F with value 0x09
14. Write register 0x5C with value 0xAF
15. Write register 0x5F with value 0xAF
16. Write register 0x70 with value 0x08
17. Write register 0x71 with value 0x04
18. Write register 0x72 with value 0x06
19. Write register 0x74 with value 0x3C
20. Write register 0x75 with value 0x28
21. Write register 0x76 with value 0x20
22. Write register 0x4E with value 0xBF
23. Write register 0x7F with value 0x03
24. Write register 0x64 with value 0x14
25. Write register 0x65 with value 0x0A
26. Write register 0x66 with value 0x10
27. Write register 0x55 with value 0x3C
28. Write register 0x56 with value 0x28
29. Write register 0x57 with value 0x20
30. Write register 0x4A with value 0x2D
31. Write register 0x4B with value 0x2D
32. Write register 0x4E with value 0x4B
33. Write register 0x69 with value 0xFA
34. Write register 0x7F with value 0x05
35. Write register 0x69 with value 0x1F
36. Write register 0x47 with value 0x1F
37. Write register 0x48 with value 0x0C
38. Write register 0x5A with value 0x20
39. Write register 0x75 with value 0x0F
40. Write register 0x4A with value 0x0F
41. Write register 0x42 with value 0x02
42. Write register 0x45 with value 0x03
43. Write register 0x65 with value 0x00
44. Write register 0x67 with value 0x76
45. Write register 0x68 with value 0x76
46. Write register 0x6A with value 0xC5
47. Write register 0x43 with value 0x00
48. Write register 0x7F with value 0x06
49. Write register 0x4A with value 0x18
50. Write register 0x4B with value 0x0C
51. Write register 0x4C with value 0x0C
52. Write register 0x4D with value 0x0C
53. Write register 0x46 with value 0x0A
54. Write register 0x59 with value 0xCD
55. Write register 0x7F with value 0x0A
56. Write register 0x4A with value 0x2A
57. Write register 0x48 with value 0x96
58. Write register 0x52 with value 0xB4
59. Write register 0x7F with value 0x00
60. Write register 0x5B with value 0xA0

- Enhanced Detection Mode

Recommend to perform software reset before loading below settings.

1. Write register 0x7F with value 0x00
2. Write register 0x51 with value 0xFF
3. Write register 0x4E with value 0x2A
4. Write register 0x66 with value 0x26
5. Write register 0x7F with value 0x14
6. Write register 0x7E with value 0x71
7. Write register 0x55 with value 0x00
8. Write register 0x59 with value 0x00
9. Write register 0x6F with value 0x2C
10. Write register 0x7F with value 0x05
11. Write register 0x4D with value 0xAC
12. Write register 0x4E with value 0x65
13. Write register 0x7F with value 0x09
14. Write register 0x5C with value 0xAF
15. Write register 0x5F with value 0xAF
16. Write register 0x70 with value 0x00
17. Write register 0x71 with value 0x00
18. Write register 0x72 with value 0x00
19. Write register 0x74 with value 0x14
20. Write register 0x75 with value 0x14
21. Write register 0x76 with value 0x06
22. Write register 0x4E with value 0x8F
23. Write register 0x7F with value 0x03
24. Write register 0x64 with value 0x00
25. Write register 0x65 with value 0x00
26. Write register 0x66 with value 0x00
27. Write register 0x55 with value 0x14
28. Write register 0x56 with value 0x14
29. Write register 0x57 with value 0x06
30. Write register 0x4A with value 0x20
31. Write register 0x4B with value 0x20
32. Write register 0x4E with value 0x32
33. Write register 0x69 with value 0xFE
34. Write register 0x7F with value 0x05
35. Write register 0x69 with value 0x14
36. Write register 0x47 with value 0x14
37. Write register 0x48 with value 0x1C
38. Write register 0x5A with value 0x20
39. Write register 0x75 with value 0xE5
40. Write register 0x4A with value 0x05
41. Write register 0x42 with value 0x04
42. Write register 0x45 with value 0x03
43. Write register 0x65 with value 0x00
44. Write register 0x67 with value 0x50
45. Write register 0x68 with value 0x50
46. Write register 0x6A with value 0xC5
47. Write register 0x43 with value 0x00
48. Write register 0x7F with value 0x06
49. Write register 0x4A with value 0x1E
50. Write register 0x4B with value 0x1E
51. Write register 0x4C with value 0x34
52. Write register 0x4D with value 0x34
53. Write register 0x46 with value 0x32
54. Write register 0x59 with value 0x0D
55. Write register 0x7F with value 0x0A
56. Write register 0x4A with value 0x2A
57. Write register 0x48 with value 0x96
58. Write register 0x52 with value 0xB4
59. Write register 0x7F with value 0x00
60. Write register 0x5B with value 0xA0

7.2 Operation Modes

Upon the chip power on as per the recommended power-on sequence in Section 7.1.1, the chip automatically switches among three operation modes when operating in different brightness conditions. The three operation modes are to cater for different far-field application needs and ambient conditions. The modes are described below.

Table 13. Three Operation Modes

Mode	Frame Rate (Typ)	Description	Lux (Typ)
0	126 fps	**Bright Mode** for general motion tracking	60
1	126 fps	**Low Light Mode** for low light motion tracking	30
2	50 fps	**Super Low Light Mode** for super low light and low-speed motion tracking	5

7.2.1 Automatic Switching of Operation Modes

The chip automatically switches between three operation modes when operating under different brightness conditions. To enable the automatic switching of operation modes, perform the below register writes in sequence. Do note that the chip enables the automatic switching between Mode 0, 1, and 2 by default after loading the initialization register setting. In the event of the sensor is switched to manual switching mode (Section 7.2.2), the automatic switching between modes can be re-enabled by performing below register writes.

- Automatic switching between Mode 0, 1, and 2:

1. Write register 0x7F with value 0x08
2. Write register 0x68 with value 0x02
3. Write register 0x7F with value 0x00

- Automatic switching between Mode 0 and 1 only:

1. Write register 0x7F with value 0x08
2. Write register 0x68 with value 0x01
3. Write register 0x7F with value 0x00

7.2.2 Manual Switching of Operation Modes

The operation mode can be manually switched depending on the user's specific application. Perform below register writes to set to the desired operation mode.

- Mode 0

1. Write register 0x7F with value 0x08
2. Write register 0x68 with value 0x00
3. Write register 0x7F with value 0x00

- Mode 1

1. Write register 0x7F with value 0x08
2. Write register 0x68 with value 0x11
3. Write register 0x7F with value 0x00

- Mode 2

1. Write register 0x7F with value 0x08
2. Write register 0x68 with value 0x22
3. Write register 0x7F with value 0x00

Note: This section aims to provide flexibility for the user to select the operation mode. The user owns the responsibility to select the appropriate operation mode to meet the desired end application.

7.3 Registers Access

The host can access the register through the SPI interface. Refer to Section 6.0 for detail.

7.3.1 Register Address Mapping

The chip has two type of register assignment. They are register bank and register address mapping. Each register bank consists of 7 bits width address mapping (0x00 to 0x7F).

7.3.2 Burst Read Operation

Burst read operation is a special serial port interface operation mode which use to reduce the serial transaction time for motion read. The speed improvement is achieved by continuous data clocking from the host for multiple registers read without further specifying the register address. In the burst read operation, NCS must be remained asserted until the burst mode transaction is fully completed.

Note: A single read of any motion-related registers (0x02 to 0x06) should be avoided during burst mode.

Read the *Motion_Burst* register activates burst mode. The chip will respond with the following motion burst report in order.

Motion burst report:

BYTE[00] = Motion
BYTE[01] = Observation
BYTE[02] = Delta_X_L
BYTE[03] = Delta_X_H
BYTE[04] = Delta_Y_L
BYTE[05] = Delta_Y_H
BYTE[06] = Challenging_Conditions
BYTE[07] = SQUAL
BYTE[08] = RawData_Sum
BYTE[09] = Maximum_RawData
BYTE[10] = Minimum_RawData
BYTE[11] = Shutter_Upper
BYTE[12] = Shutter_Middle
BYTE[13] = Shutter_Lower

Procedure to start motion burst:

1. Lower NCS signal, and wait for $t_{NCS\text{-}SCLK}$ delay.
2. Send *Motion_Burst* address (0x16). After sending this address, MOSI should be held either high or low until the burst transmission is complete (MOSI should not be toggling during subsequent SCLK cycles).
3. Wait for t_{SRAD}.
4. Start reading SPI Data continuously up to 14 bytes.
5. The NCS is de-assert after the burst transmission is complete. Alternatively, a motion burst can be terminated by pulling NCS high as needed. However, the NCS de-assert interval needs to be at least t_{BEXIT} before the next operation.
6. Access SQUAL & Shutter values to check for validity of the motion data. Refer to Section 7.4.3 for detail.
7. To read new motion burst data, repeats step 1.

7.3.3 Related Registers

Usage	Name	Bank	Address
Burst Read	*Motion_Burst*	0	0x10

Register Name	Motion_Burst		
Bank	0	**Address**	0x16
Access	R	**Default Value**	0x00
Description	The *Motion_Burst* register is used for high-speed access of up to 12 register bytes. Refer to Section 7.3.1 for use details.		

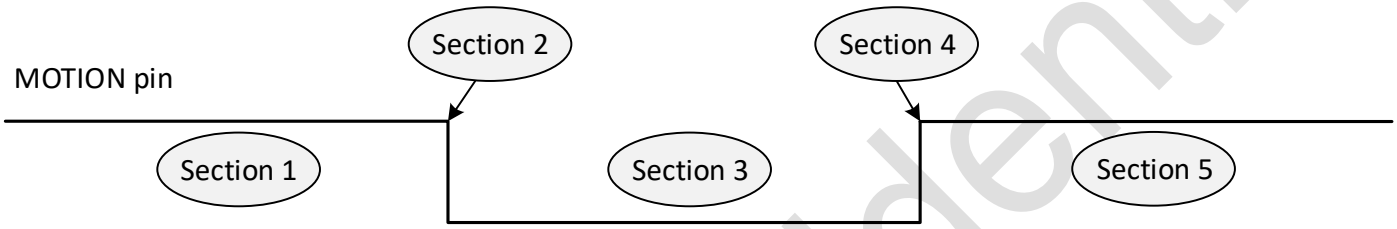
7.4 Output

7.4.1 Motion Bit and Motion Pin Interrupt

The motion bit in the *Motion* register (address 0x02) is an indicator flag for motion detection. Before reading motion data registers, the host must check the motion bit in the *Motion* register (address0x02). Motion bit will be updated as 1 if the motion data in the motion registers are available.

The motion pin is an interrupt (active low) pin that triggers the host when motion has occurred. The MOTION pin goes low whenever the motion bit in the *Motion* register (address 0x02) is "1".

Note: No pull-up resistor is required at the MOTION pin or at the host side to reduce power when the motion pin is in a low state.



Section 1 : No motion detected, motion data are invalid. Motion bit= 0 and motion pin de-asserted.
Section 2 : When motion detected, motion pin is asserted, motion bit= 1 and motion data accumulation begins.
Section 3 : Motion data accumulation in progress and wait for host to access.
Section 4 : Upon motion data register is accessed, motion bit= 0, interrupt pin is de-asserted and motion data registers are automatic cleared.
Section 5 : No motion detected, motion data are invalid. Motion bit= 0 and motion pin de-asserted.

Figure 25. Motion Interrupt Pin Function

7.4.2 Output Access

The accumulated motion data are stored in *Delta_X_L, Delta_X_H, Delta_Y_L,* and *Delta_Y_H* registers. The data can be accessed through single or burst read operation as needed.

Single read access the motion data by using normal read operation when the data in register *Delta_X_L, Delta_X_H, Delta_Y_L,* and *Delta_Y_H* are available. Burst mode operation is a special serial port operation mode which use to reduce the serial transaction time for motion read, refer to Section 7.3.1 for the detail.

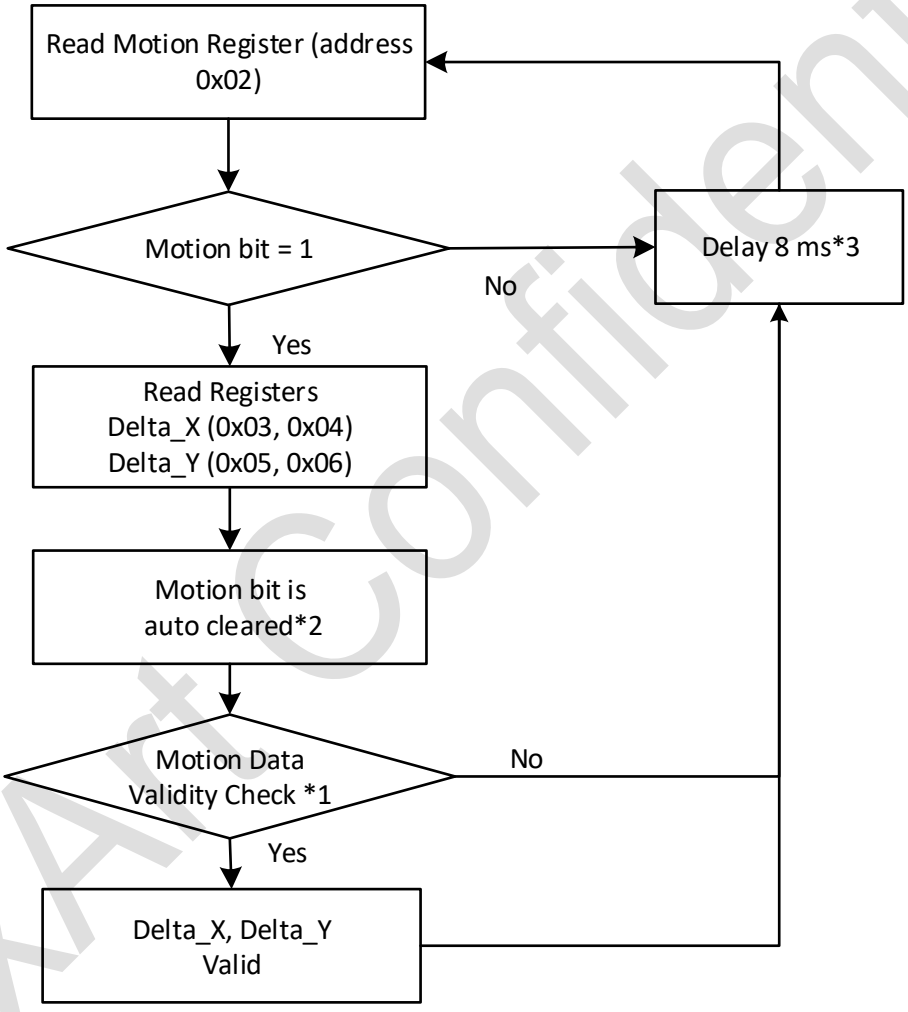
There are two methods to read out the motion data:

- Polling method
- Interrupt method

- Polling Method

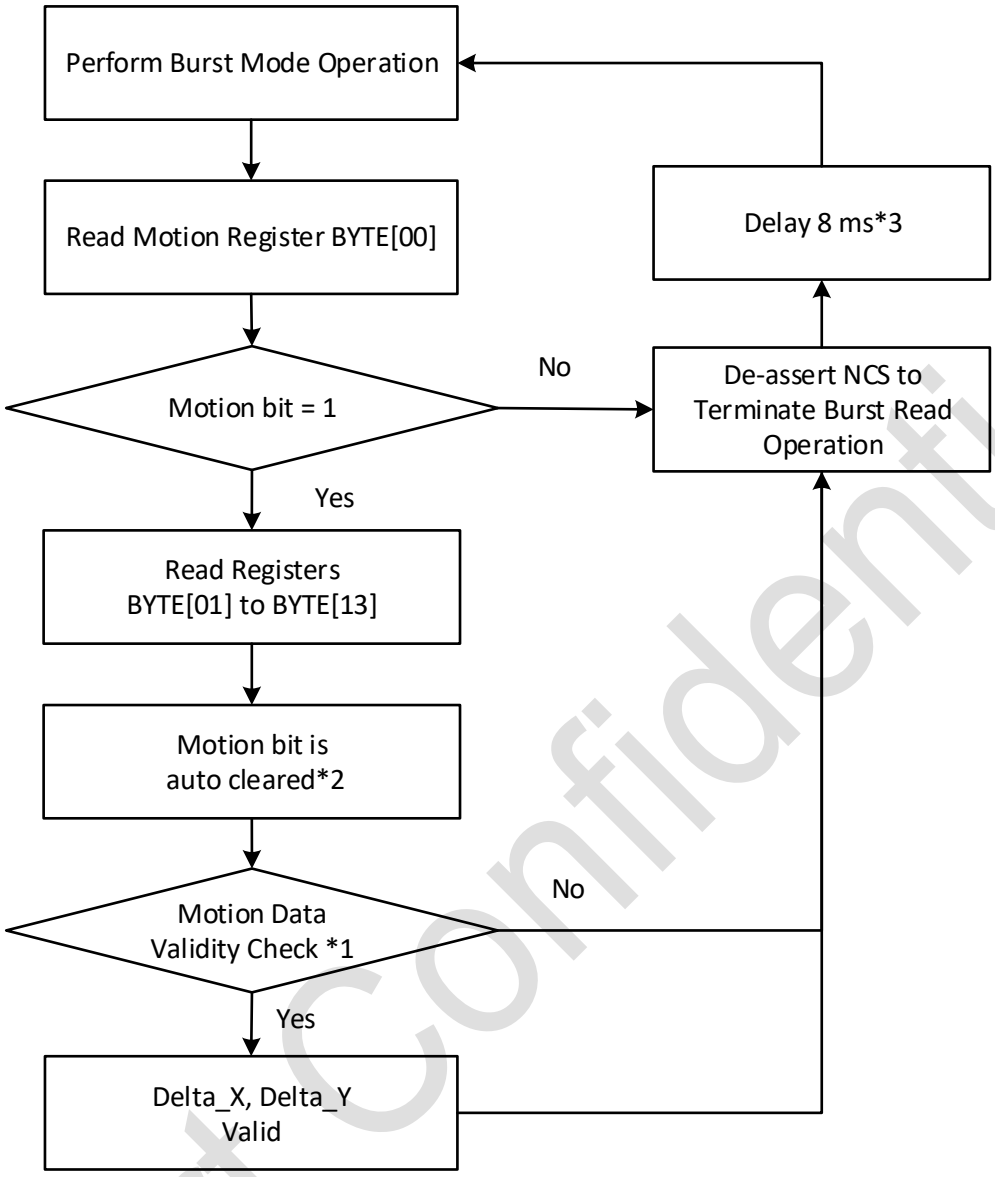
The motion bit in the *Motion* register (address 0x02) is an output that signals the host when motion has occurred. The motion bit turns to “1" whenever the motion is detected; in other words, whenever there is non-zero data in the *Delta_X_L, Delta_X_H, Delta_Y_L,* or *Delta_Y_H* registers. Motion bit is cleared to “0” by reading or writing to the *MOTION* register.

By reading and checking the *Motion* register periodically, the host can get the motion data simply through the SPI interface. This register allows the user to determine if a motion has occurred since the last time it was read. If the Motion register bit[7]= 1, the host can read register 0x03, 0x04, 0x05 and 0x06 to get the accumulated motion data.



Note:
*1: Refer Section 7.4.3 for detail
*2: Motion bit is auto cleared after reading *MOTION* register
*3: The polling interval is configured by the host

Figure 26. Polling Method – Single Read



Note:
*1: Refer Section 7.4.3 for the detail
*2: Motion bit is auto cleared after reading *MOTION* register
*3:The polling interval is configured by the host

Figure 27. Polling Method – Burst Read

- Interrupt Method

When the MOTION pin is low, the register is pending host to read. Otherwise, the MOTION pin goes high when the register is cleared. When the chip detects the occurrence of motion, the MOTION pin is asserted and triggers the host to read data.

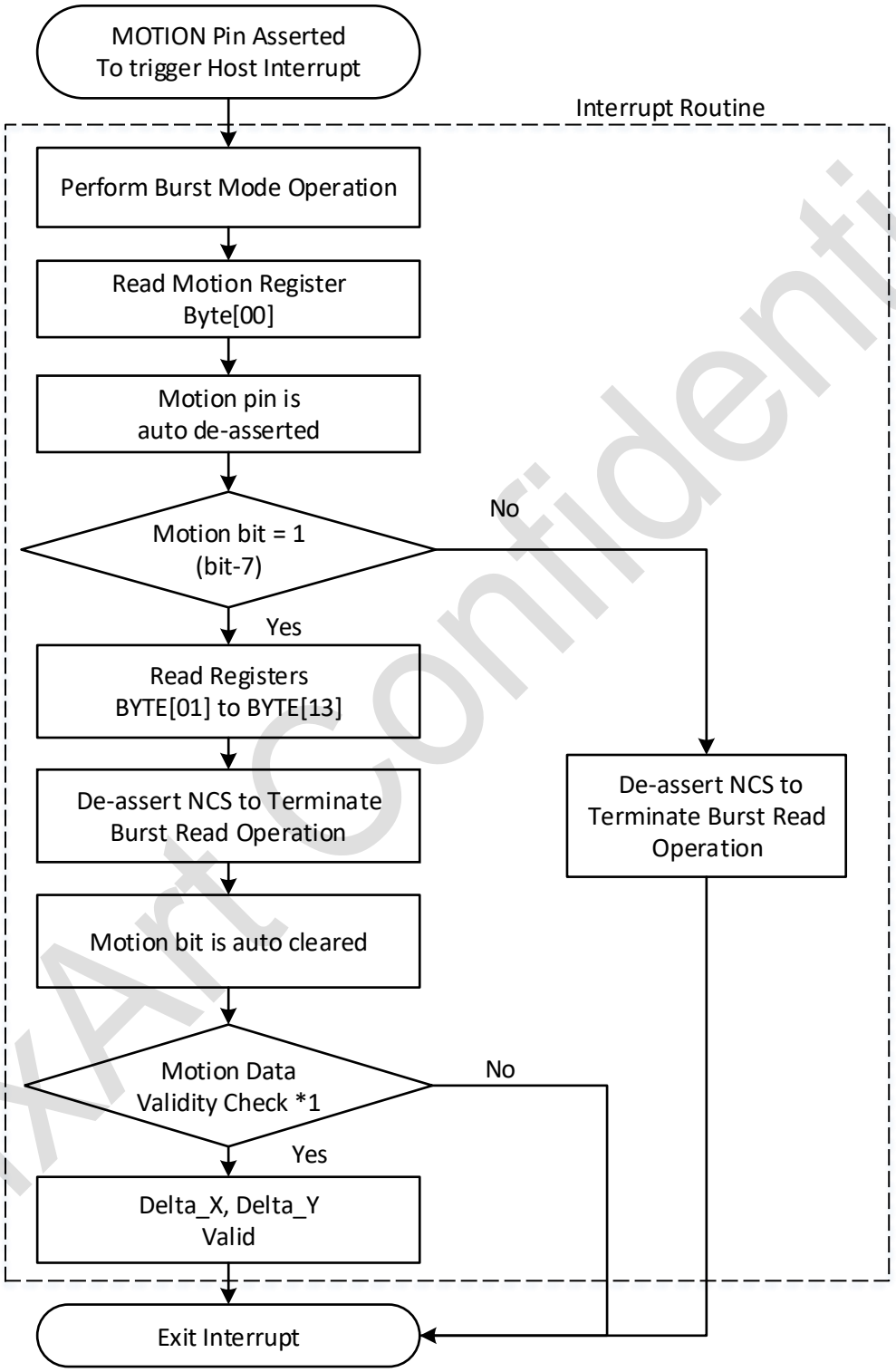


Figure 28. Motion Data Access - Interrupt Method

7.4.3 Invalid Motion Data Condition

There is additional verification that is required to check for the validity of the motion data. The step to as follow:

1. Read *Observation* register bit[6] and bit[7] to check the current operation mode.
2. Access both *SQUAL* and *Shutter* register.
3. At the respective operation mode, to suppress false motion reports, discard Delta X and Delta Y values if the SQUAL and Shutter values meet the condition outlined in the below table.

Mode	SQUAL	Shutter
0: Bright Mode	< 0x19	≥ 0x00FF80
1: Low Light Mode	< 0x46	≥ 0x00FF80
2: Super Low Light Mode	< 0x55	≥ 0x025998

7.4.4 Data Lost and Corruption

Motion data register storage limit is -32767 to 32767 for X-axis and Y-axis respectively. When motion is detected, motion data will be generated, accumulated, and stored in registers. If the host did not read out motion data before data reach the limit, data corruption may occur.

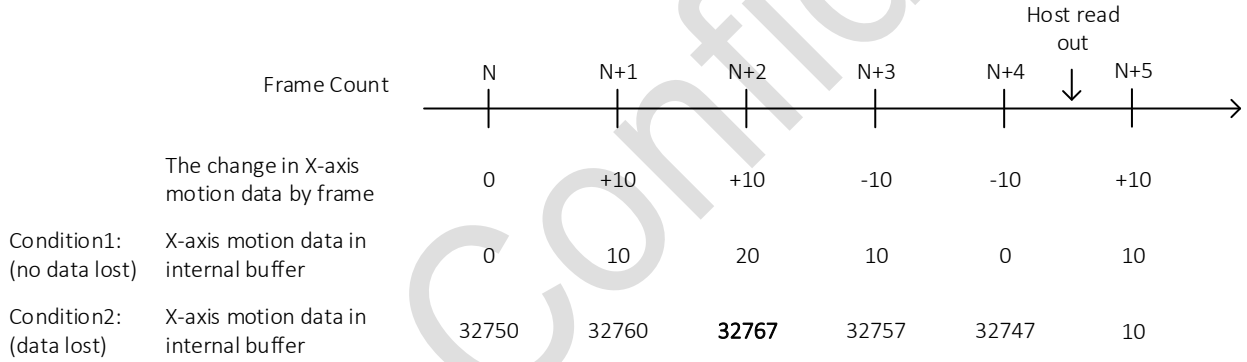


Figure 29. Example of Data Lost When Internal Buffer Reach limit

From frame N to frame N+4, the total change of motion data is 0.

- Condition1

  If the motion data register value is 0 in frame N, the host may read motion data value 0 in between frames N+4 and N+5.

- Condition2

  If the motion data register value is 32750 in frame N, the host may read motion data value 32747 in between frames N+4 and N+5.

The above two conditions' movement behavior are identical. However, condition 2 register hitting limit of 32767, host read value 32747 instead of 32750 where some motion data are lost.

To prevent this

1. The host is required to respond to interrupt triggering as soon as possible for reading the motion data register.
2. The host needs to shorten the polling interval.
3. The host is required to monitor and avoid hitting the limit of motion data.
4. The chip resolution can be optimized.

7.4.5 Challenging Conditions Detection

The chip can detect several types of challenging conditions, where the tracking performance is compromised. Upon detecting such conditions, the chip stops random and erratic motion output, in an effort to mitigate the adverse impact on the intended operations of the host.

When the chip detects such challenging condition(s), bit[0] of the *Motion* register (address 0x02) is set. In addition to this, the *Challenging_Conditions* register (address 0x19) indicates the type of challenging condition(s) detected. This register is also available in the motion burst report; refer to Section 7.3.1 for the location of this register among the motion burst's bytes.

With the default setting (Section 7.1.2), motion output will not be reported once the challenging condition is detected, where bit[0] of the Motion register (address 0x02) is set. The motion output cut-off can be disabled via register writes.

- To disable motion cut-off (which enables motion to be reported when a challenging condition is detected)

1. Write register 0x7F with value 0x05
2. Write register 0x6A with value 0xFF
3. Write register 0x6B with value 0x3F
4. Write register 0x7F with value 0x00

- To enable motion cut-off (default setting)

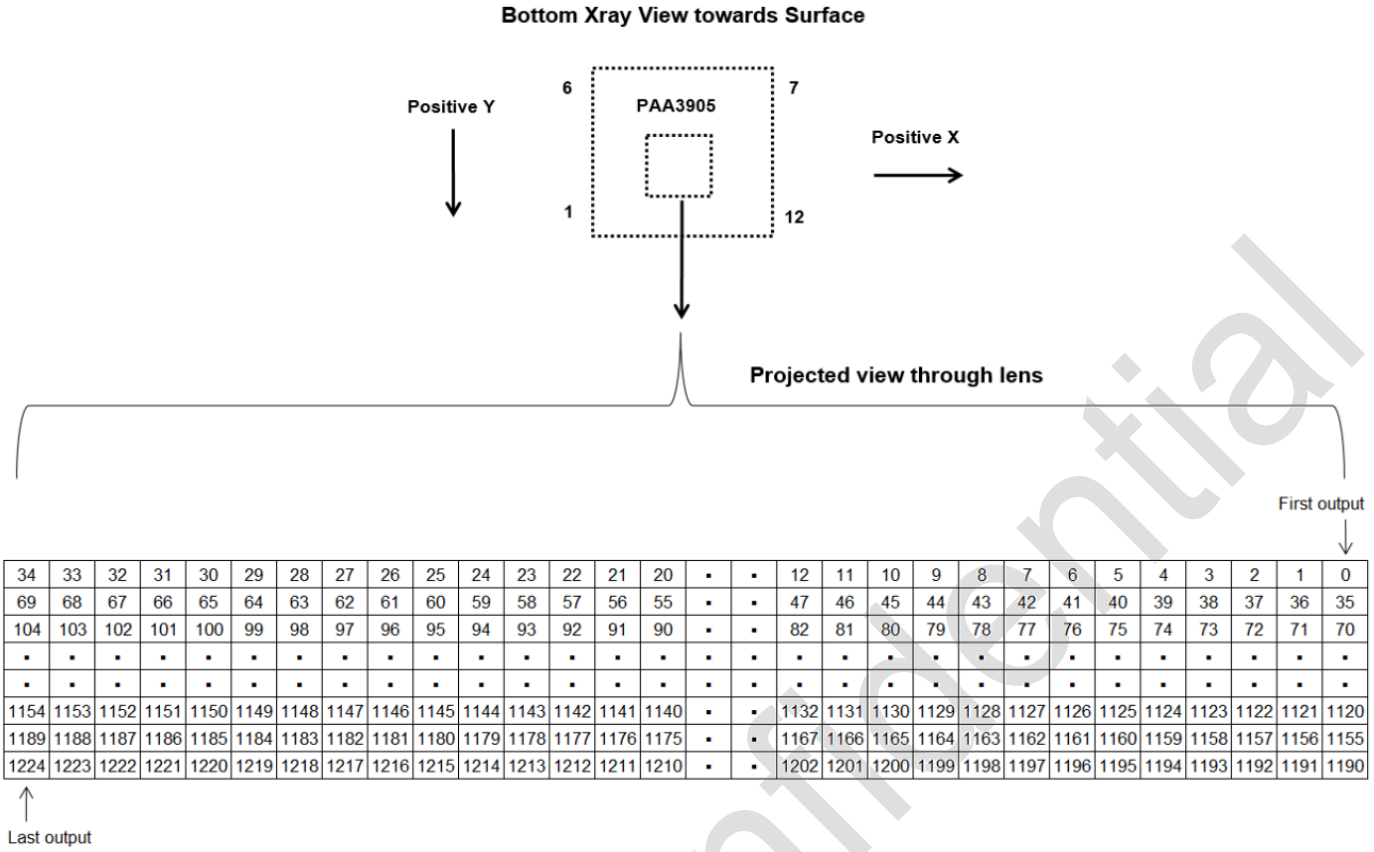
1. Write register 0x7F with value 0x05
2. Write register 0x6A with value 0xC5
3. Write register 0x6B with value 0x15
4. Write register 0x7F with value 0x00

7.4.6 Raw Data Output

Raw Data Output is a special mode to download the full array of raw data using a register read operation. This mode disables navigation and no other SPI activity is allowed during this period. To restore navigation for normal operation, a software reset is required.

System Initialization sequence (Section 7.1) should have been completed before performing Raw Data Output. The Raw Data Output procedure is outlined below:

1. To enter Raw Data Output mode, perform the below register writes in sequence:
    a. Write register 0x7F with value 0x00
    b. Write register 0x67 with value 0x25
    c. Wait for 25ms.
    d. Write register 0x55 with value 0x20
    e. Write register 0x7F with value 0x13
    f. Write register 0x42 with value 0x01
    g. Wait for 25ms.
    h. Write register 0x7F with value 0x00
    i. Write register 0x0F with value 0x20
    j. Check *RawData_Grab_Status* register (address 0x10) to make sure bit 1 is clear, expected value is 0x00.
    k. Write register 0x0F with value 0x11
    l. Write register 0x0F with value 0x13
    m. Write register 0x0F with value 0x11
2. Poll *RawData_Grab_Status* register (address 0x10) until getting bit 0 equal to 1, which means pixel data is ready to read.
3. Write register 0x13 with any value.
4. Read register 0x13 for 1225 times to form a complete picture element array information.
   After the read process is done, bit 1 of the *RawData_Grab_Status* register (address 0x10) will be asserted to 1 to indicate all pixel data is being read successfully.
5. Disable Raw Data output Mode, perform below register writes in sequence.
    a. Write register 0x7F with value 0x00
    b. Write register 0x0F with value 0x00
    c. Write register 0x55 with value 0x00
    d. Write register 0x7F with value 0x13
    e. Write register 0x42 with value 0x00
    f. Write register 0x7F with value 0x00
    g. Write register 0x67 with value 0xA5
6. To continue to download the raw data for the next frame, repeat steps 1 to step 5.
7. To resume the navigation, issue a software reset command to the chip and initialize the chip as per section 5.4.2.



34	33	32	31	30	29	28	27	26	25	24	23	22	21	20	▪	▪	12	11	10	9	8	7	6	5	4	3	2	1	0
69	68	67	66	65	64	63	62	61	60	59	58	57	56	55	▪	▪	47	46	45	44	43	42	41	40	39	38	37	36	35
104	103	102	101	100	99	98	97	96	95	94	93	92	91	90	▪	▪	82	81	80	79	78	77	76	75	74	73	72	71	70
▪	▪	▪	▪	▪	▪	▪	▪	▪	▪	▪	▪	▪	▪	▪	▪	▪	▪	▪	▪	▪	▪	▪	▪	▪	▪	▪	▪	▪	▪
▪	▪	▪	▪	▪	▪	▪	▪	▪	▪	▪	▪	▪	▪	▪	▪	▪	▪	▪	▪	▪	▪	▪	▪	▪	▪	▪	▪	▪	▪
1154	1153	1152	1151	1150	1149	1148	1147	1146	1145	1144	1143	1142	1141	1140	▪	▪	1132	1131	1130	1129	1128	1127	1126	1125	1124	1123	1122	1121	1120
1189	1188	1187	1186	1185	1184	1183	1182	1181	1180	1179	1178	1177	1176	1175	▪	▪	1167	1166	1165	1164	1163	1162	1161	1160	1159	1158	1157	1156	1155
1224	1223	1222	1221	1220	1219	1218	1217	1216	1215	1214	1213	1212	1211	1210	▪	▪	1202	1201	1200	1199	1198	1197	1196	1195	1194	1193	1192	1191	1190

Note: The direction of the XY motion is based on sensor moving above the surface, and *Orientation* register (address 0x5B) is set to 0xA0 after loading Initialization Register Setting in Section 7.1.2.

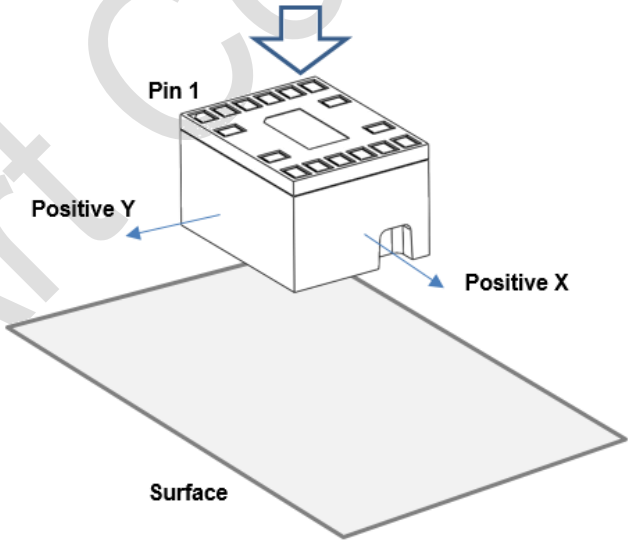


Figure 30. Raw Data Map

7.4.7 Related Register

Usage	Name	Bank	Address
Output Access	*Motion*	0	0x02
	Delta_X_L	0	0x03
	Delta_X_H	0	0x04
	Delta_Y_L	0	0x05
	Delta_Y_H	0	0x06
Challenging Condition Detection	*Challenging_Conditions*	0	0x19
Raw Data Output	*RawData_Grab_Status*	0	0x10
	RawData_Grab	0	0x13

Register Name	Motion		
Bank	0	**Address**	0x02
Access	R	**Default Value**	0x00

Description

This register allows the user to determine if a motion has occurred since the last time it was read. The procedure to read the motion registers (*Delta_X_L, Delta_X_H, Delta_Y_L,* and *Delta_Y_H*) are as follows:

1. Read the *Motion* register. This will freeze *the Delta_X_L, Delta_X_H, Delta_Y_L,* and *Delta_Y_H* register values.
2. If the MOT bit is set, *Delta_X_L, Delta_X_H, Delta_Y_L,* and *Delta_Y_H* registers should be read in the given sequence to get the accumulated motion. Note that if *Delta_X_L, Delta_X_H, Delta_Y_L,* and *Delta_Y_H* registers are not read before the motion register is read for the second time, the data in *Delta_X_L, Delta_X_H, Delta_Y_L,* and *Delta_Y_H* registers will be lost.
3. To read a new set of motion data (*Delta_X_L, Delta_X_H, Delta_Y_L,* and *Delta_Y_H* registers), repeat from step (1).

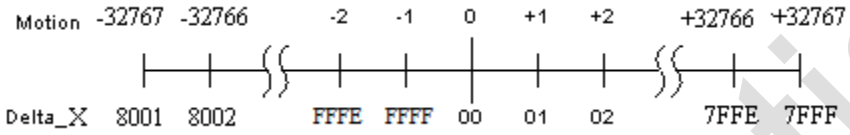
Note:

1. To suppress false motion reports, discard Delta X and Delta Y values if the ***SQUAL*** and ***Shutter*** values meet the condition outlined in the below table. Read *Observation* register bit[6] and bit[7] to check the current operation mode.

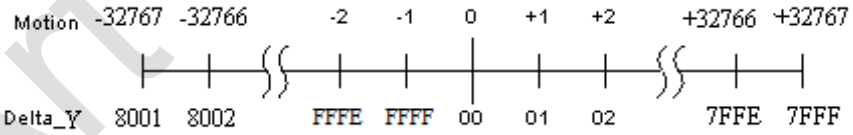
Mode	SQUAL	Shutter
0: Bright Mode	< 0x19	≥ 0x00FF80
1: Low Light Mode	< 0x46	≥ 0x00FF80
2: Super Low Light Mode	< 0x55	≥ 0x025998

2. Write any value to this register to clear the *Delta_X_L, Delta_X_H, Delta_Y_L*, and *Delta_Y_H* registers.

Bit Field	Name	Default Value	Description
7	MOT[7]	0	Motion since the last report 0: No motion 1: Motion occurred, data ready for reading in *Delta_X_L, Delta_X_H, Delta_Y_L,* and *Delta_Y_H* registers.
0	MOT[0]	0	0: Challenging condition is not detected 1: Challenging condition is detected

Register Name	Delta_X_L		
Bank	0	**Address**	0x03
Access	R	**Default Value**	0x00
Description	16 bits 2's complement number. Lower 8 bits of *Delta_X*. X movement is in counts since the last report. Absolute value is determined by resolution. 		

Register Name	Delta_X_H		
Bank	0	**Address**	0x04
Access	R	**Default Value**	0x00
Description	16 bits 2's complement number. Upper 8 bits of *Delta_X*. *Delta_X_H* must be read after *Delta_X_L* to have the full motion data. **Note:** It is recommended that registers 0x02, 0x03, 0x04, 0x05 and 0x06 be read sequentially.		

Register Name	Delta_Y_L		
Bank	0	**Address**	0x05
Access	R	**Default Value**	0x00
Description	16-bit 2's complement number. Lower 8 bits of *Delta_Y*. Y movement is in the count since the last report. Absolute value is determined by resolution. 		

Register Name	Delta_Y_H		
Bank	0	**Address**	0x06
Access	R	**Default Value**	0x00
Description	16-bit 2's complement number. Upper 8 bits of *Delta_Y*. *Delta_Y_H* must be read after *Delta_Y_L* to have the full motion data. **Note:** It is recommended that registers 0x02, 0x03, 0x04, 0x05 and 0x06 to be read sequentially.		

Register Name	Challenging_Conditions		
Bank	0	**Address**	0x19
Access	R	**Default Value**	0x00
Description	This register provides types of challenging surfaces and conditions detected when bit 0 of the Motion register is set.		

Bit Field	Name	Default Value	Description
7	CSD[7]	0	0: Not detected. 1: Compromised tracking performance i.e. Stripe surface is detected.
6	CSD[6]	0	0: Not detected. 1: Compromised tracking performance i.e. Checkerboard is detected.
5	CSD[5]	0	0: Not detected. 1: Compromised tracking performance e.g. Yaw motion or glossy surface is detected.
3	CSD[3]	0	0: Not detected. 1: Compromised tracking performance e.g. Yaw motion or glossy surface is detected.

Register Name	RawData_Grab_Status		
Bank	0	**Address**	0x10
Access	R	**Default Value**	0x00
Description	This register provides the status of the raw data grab process.		

Bit Field	Name	Default Value	Description
1	RDG[1]	0	0: Default Read counter is cleared 1: Frame Grab is Done. All pixel data is read
0	RDG[0]	0	0: Raw data is not ready. 1: Raw data is ready.

Register Name	RawData_Grab		
Bank	0	**Address**	0x13
Access	R/W	**Default Value**	0x00
Description	This register is used to read out the full array of raw data values, the chip needs to be held stationary for the duration of grabbing raw data until the full array is completely read out, as the information is read out one data at a time. This process is initialized by a single write of any value to this register (only needs to be written once per frame). Reading this register will unload 8-bit raw data at a time.		

7.5 LED Control

The chip support external LED control to extend the appropriate illumination base on applications.

This section provides information and reference schematics to support external LED circuitry. Synchronizing the LED_SYNC pulse control is a significant power-saving feature for battery-powered applications.

NOTE: While this section aims to guide in utilizing the LED_SYNC pin, the user owns the responsibility to select the appropriate LED and design its circuitry to meet the desired end application.

For power-saving purposes, LED control (*LED_SYNC*) is disabled by default. Below is the step to enable the LED Control (*LED_SYNC*).

1. Power up the chip and initialize register settings as outlined in Section 7.1.1 Initial Flow.

2. Perform below register writes in sequence:
    a. Write register 0x7F with value 0x14
    b. Write register 0x6F with value 0x0C
    c. Write register 0x7F with value 0x00

By probing the LED_SYNC pin, pulse waveforms can be observed as shown below.

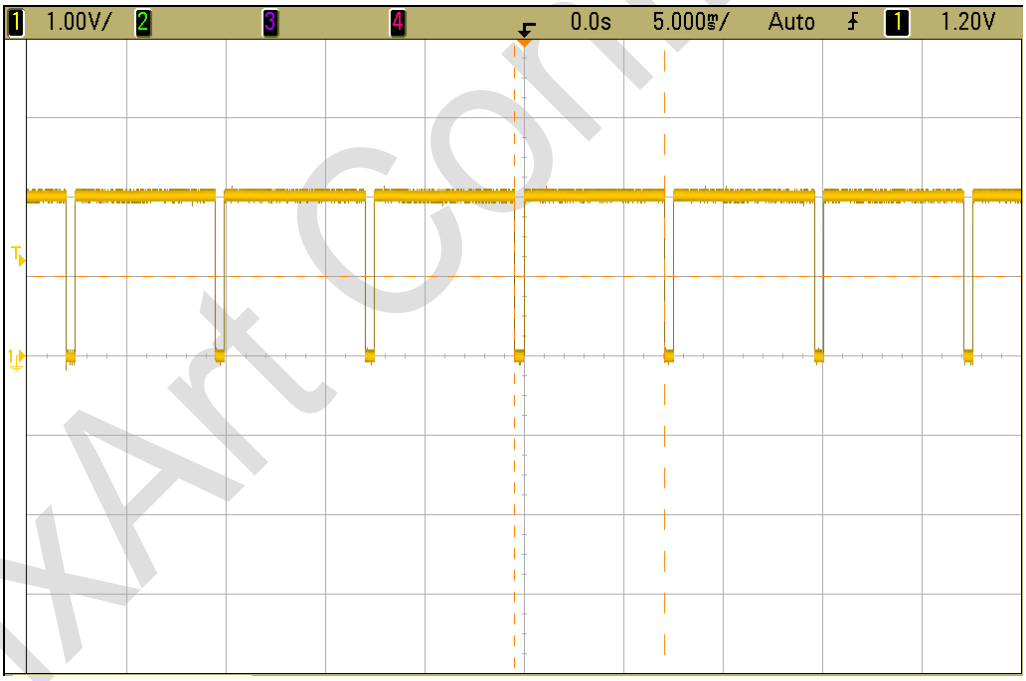


Figure 31. LED_SYNC pulsing

An example of synchronizing the LED_SYNC pulse to control an external LED circuitry using an IR LED is shown below:

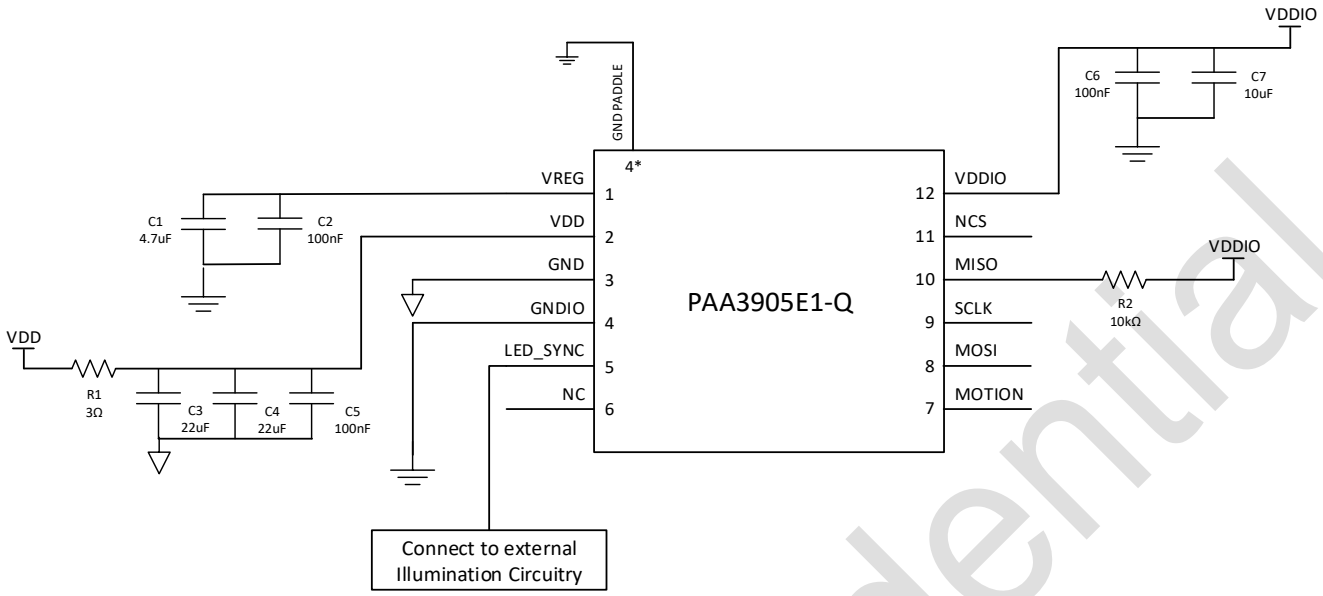
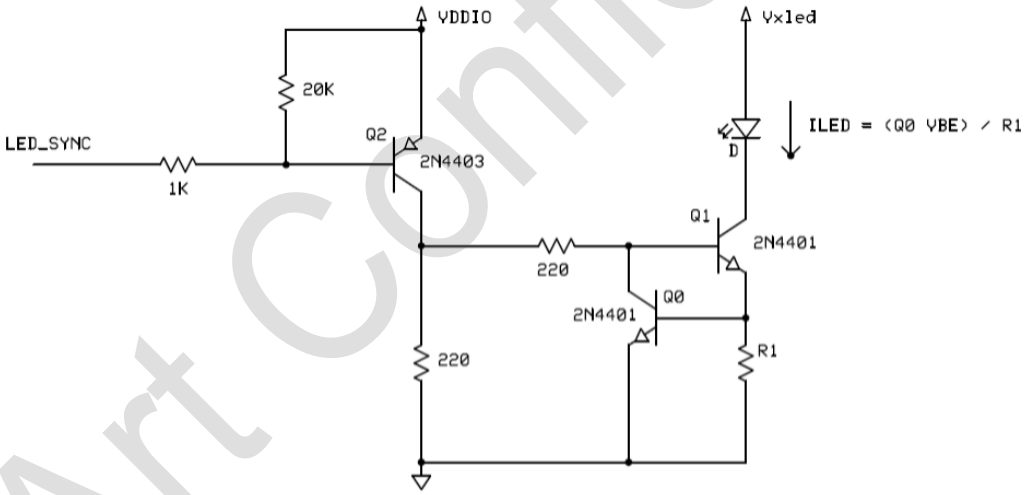


Figure 32. Reference schematics to drive external LED circuitry



Note:
1. The maximum continuous current for 2N4401 is 600mA.
2. For driving more than one LED, connect the extra LED(s) in parallel.
Adjust R1 accordingly to cater to the total current consumption of all LEDs.

Figure 33. Reference Schematic of External LED Circuitry

To disable the LED control and back to the default state, perform the below register writes in sequence:

a. Write register 0x7F with value 0x14
b. Write register 0x6F with value 0x2C
c. Write register 0x7F with value 0x00

7.6 Frame Synchronization (FS)

This chip supports a frame synchronization function, a single or multiple chip can be synchronized to other devices at the desired frame rates through a series of register writes.

For example, in a drone application system, the output from the chip can be matched to the output rate of the IMU sensor to ensure data collection from these devices is from the same time stamp. In the case where multiple chips are employed in a system, synchronized data output from these multiple chips is crucial to provide accurate results in a specific usage model.

7.6.1 Hardware Requirement

There is no additional hardware connection needed for Frame Synchronization (FS).

7.6.2 Single Chip Synchronization

The basis of this operation is to provide flexibility to the host to start the frame of the chip at any given time. The procedure to synchronize the chip to the desired frame rate is as below:

1. Power-on the chip as per Section 7.1.

2. Perform below register writes in sequence. This issues a command to the chip to stop operation upon the completion of the current frame.
    a. Write register 0x7F with value 0x07
    b. Write register 0x41 with value 0x82

3. Delay for 100 ms to allow the chip sufficient time to complete the current frame's activities.

4. Read Register 0x15. Check Bit [7:6] for AMS mode and store it into a variable.

Bit [7:6]	AMS mode
0x0	Mode 0
0x1	Mode 1
0x2	Mode 2

5. Issue start operation command to the chip with the following register write.
   Write register 0x1F with value 0x00

6. Start timer period before issuing start operation command to the chip. Use the timer period corresponding with the chip's AMS mode as shown below

Mode 0 & 1		Mode 2	
Timer (ms)	Frame Rate (fps)	Timer (ms)	Frame Rate (fps)
8.8	114	22.0	45

Note: Additional time (10% of actual AMS mode frame period) is added to buffer for clock variation between the host and the chip.

7. Poll motion data through single read or burst read.

   Note: Check *Motion* register for bit[6] to be zero (bit[6]= 0). If bit[6] is set (bit[6]= 1) for 3 consecutive frames. Exit the routine, power cycle the chip, and repeat from Step 1.

8. Upon expiration of the timer period, repeat Steps 4 to 7 to continue the synchronization routine and normal operation of the chip. If these steps are not repeated, there will be no data output from the chip.

9. To exit the synchronization routine, issue a soft reset command to the chip and configure the registers as per Section 7.1

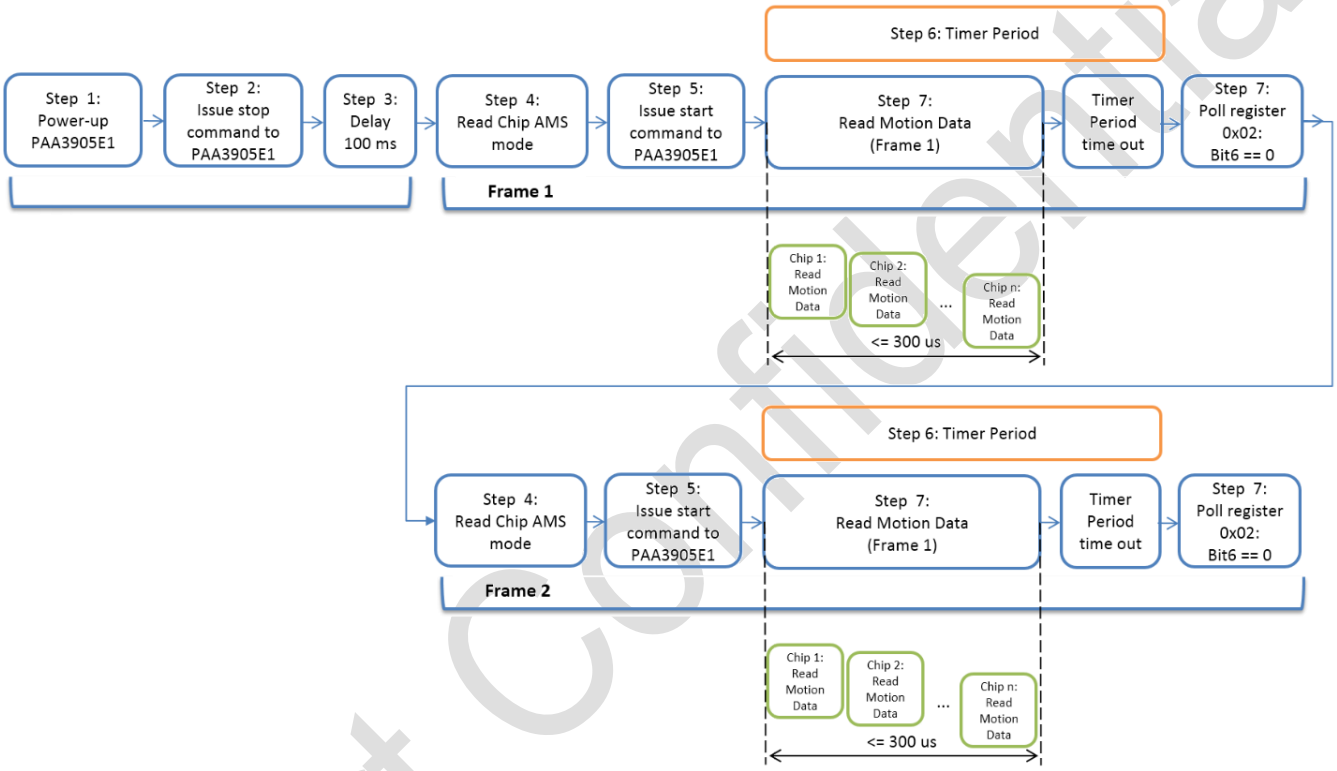


Figure 34. Single Chip Synchronization

7.6.3 Multiple Chips Synchronization

To synchronize multiple chips, additional steps are added to the steps in Section 7.6.2 Single Chip Synchronization. One chip will be selected to be a master chip with the rest being slave chips. The master chip is the only chip with AMS enabled. Slave chips will be following the same AMS mode as the master sensor through host force AMS mode register write. All the chips will be synchronized by receiving the start frame register write at the same time. To reiterate, every chip's NCS pin should be active low during the start frame register write. This ensures all the chips start the frame at the same time and as close as possible.

All motion data should be read immediately from one chip to another, in a sequential manner. The maximum period to read motion data for all the chips should not exceed 300 µs (regardless of the timer period used). This helps to ensure the motion data read from all the chips are synchronized. If motion data for the chips are read outside these time frames, the motion data obtained (for some of the chips) might not be from the same frame.

The procedure to synchronize multiple chips to the desired frame rate is as below:

1. Power up the chip as per Section 7.1.

2. Perform below register writes in sequence to all master and slave chips at the same time. This issues a command to the chip to stop operation upon the completion of the current frame."
    a. Write register 0x7F with value 0x07
    b. Write register 0x41 with value 0x82

3. Delay for 100 ms to allow the chip sufficient time to complete the current frame's activities.

4. Read Register 0x15 from the master chip. Check Bit[7:6] for AMS mode.

Bit [7:6]	AMS mode	Force_AMS_Mode
0x0	Mode 0	0x00
0x1	Mode 1	0x11
0x2	Mode 2	0x22

   Perform below register writes in sequence to force AMS mode to all slave chips.
    a. Write register 0x7F with value 0x08
    b. Write register 0x68 with value in Force_AMS_Mode

5. Issue start operation command to the chip with the following register write to all master and slave chips at the same time.

   Write register 0x1F with 0x00

6. Start timer period before issuing start operation command to the chip. Use the timer period corresponding with the chip's AMS mode as shown below:

Mode 0 & 1		Mode 2	
Timer (ms)	Frame Rate (fps)	Timer (ms)	Frame Rate (fps)
8.8	114	22.0	45

Note:

Additional time (10% of actual AMS mode frame period) is added to buffer for clock variation between the host and the chip.

7. Poll motion data through single read or burst read from one chip to another, starting with the master chip followed by slave chips in a sequential manner

Note:

a. Check the *Motion* register for bit[6] to be zero (bit[6] = 0) for all chips. If bit[6] is set (bit[6]= 1) for 3 consecutive frames for any of the chips, do exit the routine, power cycle the chip, and repeat from Step 1.

b. The maximum period to read motion data for all the chips should not exceed 300 µs (regardless of the timer period used)

8. Upon expiration of the timer period, repeat Steps 4 to 7 to continue the synchronization routine and normal operation of the chip. If these steps are not repeated, there will be no data output from the chip.

9. To exit the synchronization routine, issue a soft reset command to the chip and configure the registers as per Section 7.1.

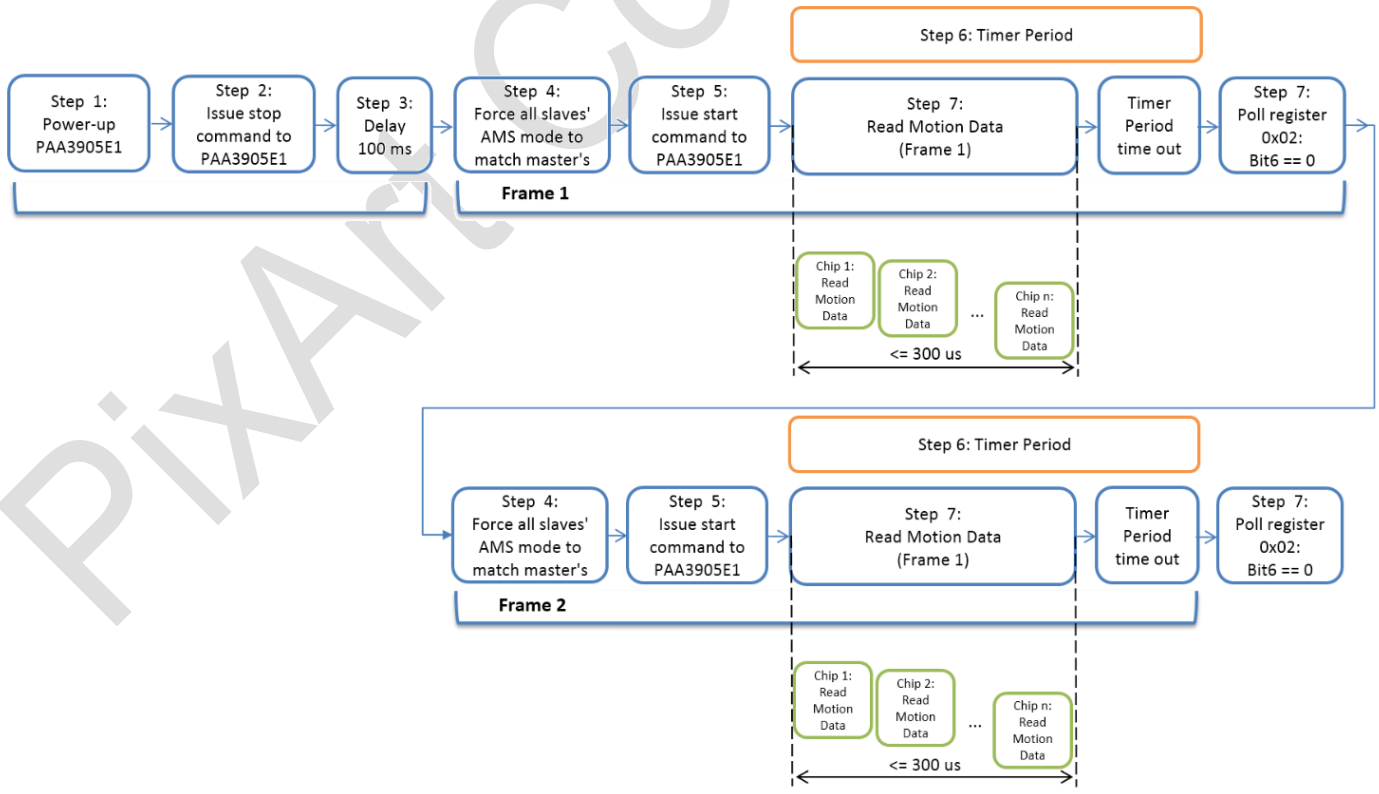


Figure 35. Multiple Chip Synchronization

8.0 Register

8.1 Registers Map

Table 14. Register List

Address	Bit Field	Name	Access	Bit Field Default Value	Register Default Value
0x00	-	*Product_ID*	R	-	0xA2
0x01	-	*Revision_ID*	R	-	0x00
0x02	-	*Motion*	R/W	-	0x00
0x03	-	*Delta_X_L*	R	-	0x00
0x04	-	*Delta_X_H*	R	-	0x00
0x05	-	*Delta_Y_L*	R	-	0x00
0x06	-	*Delta_Y_H*	R	-	0x00
0x07	-	*SQUAL*	R	-	0x00
0x08	-	*RawData_Sum*	R	-	0x00
0x09	-	*Maximum_RawData*	R	-	0x00
0x0A	-	*Minimum_RawData*	R	-	0xFF
0x0B	-	*Shutter_Lower*	R	-	0x00
0x0C	-	*Shutter_Middle*	R	-	0x00
0x0D	-	*Shutter_Upper*	R	-	0x00
0x10	-	*RawData_Grab_Status*	R	-	0x00
0x13	-	*RawData_Grab*	R/W	-	0x00
0x15	-	*Observation*	R/W	-	0x00
0x16	-	*Motion_Burst*	R	-	0x00
0x3A	-	*Power_Up_Reset*	W	-	N/A
0x3B	-	*Shutdown*	W	-	N/A
0x4E	-	*Resolution*	R/W	-	0x2A
0x5B	-	*Orientation*	R/W	-	0xE0
0x5F	-	*Inverse_Product_ID*	R	-	0x5D

8.2 Register Description

8.2.1 Product ID

Register Name	Product_ID		
Bank	0	**Address**	0x00
Access	R	**Default Value**	0xA2
Description	This value is a unique identification assigned to this model only. The value in this register does not change; it can be used to verify that the serial communications link is functional.		

Register Name	Revision_ID		
Bank	0	**Address**	0x01
Access	R	**Default Value**	0x00
Description	This register contains the current IC revision. It is subject to change when new IC versions are released.		

Register Name	Inverse_Product_ID		
Bank	0	**Address**	0x5F
Access	R	**Default Value**	0x5D
Description	This value is the inverse of the *Product_ID*. It is used to test the SPI port hardware.		

8.2.2 Operational Control

Register Name	Resolution		
Bank	0	**Address**	0x4E
Access	R/W	**Default Value**	0x2A
Description	This register sets the X and Y resolution of the chip. To calculate the approximate resolution value of each register setting, use the formula below: Approximate Resolution = (Register Value + 1) x (200 ÷ 8600) where every increment / decrement of the register value generates approximately 2.3% value change, referenced in Figure 4. Resolution versus Height Chart. The maximum register value is 0xFF (6x of default). The minimum register value is 0. Example at height = 1m, 2.3% value change of 12 CPI ≈ 0.28 count. Delta between register value 0x2A and 0xFF = 255 - 42= 213 register values. Therefore, 213 x 0.28 count ≈ 60 counts. As such, the CPI ≈ 12 + 60 = 72		

Register Name	Orientation		
Bank	0	**Address**	0x5B
Access	R	**Default Value**	0xE0
Description	This register sets the orientation of the reported X and Y positions. This includes swapping X / Y and flipping (inverting) the direction of the X and Y axis. If both are selected, swapping is done before flipping.		

Bit Field	Name	Default Value	Description
7	ORT7	1	Swap X and Y 0: No Swap 1: Swap
6	ORT6	1	Invert Y direction 0: Not inverted 1: Inverted
5	ORT5	1	Invert X direction 0: Not inverted 1: Inverted

8.2.3 Operational Check

Register Name	SQUAL		
Bank	0	**Address**	0x07
Access	R	**Default Value**	0x00
Description	The *SQUAL* (Surface quality) register is a measure of the number of valid features visible by the chip in the current frame. Use the following formula to find the total number of valid features: Number of Features = SQUAL Register Value x 4 The maximum *SQUAL* register value is 0xFF. Since small changes in the current frame can result in changes in *SQUAL*, variations in *SQUAL* when looking at a surface are expected. *SQUAL* values are only valid in run mode.		

Register Name	RawData_Sum		
Bank	0	**Address**	0x08
Access	R	**Default Value**	0x00
Description	This register is used to find the average raw data value. To find the average raw data value, use the formula below: Average Raw Data = (Register Value x 8192) ÷ 1225 The maximum register value is 0x98. The minimum register value is 0. The *RawData_Sum* value can change every frame.		

Register Name	Maximum_RawData		
Bank	0	Address	0x09
Access	R	Default Value	0x00
Description	*Maximum raw data* value in the current frame. Minimum value = 0, maximum value = 255. The *maximum raw data* value can change every frame.		

Register Name	Minimum_RawData		
Bank	0	Address	0x0A
Access	R	Default Value	0xFF
Description	*Minimum raw data* value in the current frame. Minimum value= 0, maximum value= 255. The *minimum raw data* value can change every frame.		

Register Name	Shutter_Lower		
Bank	0	Address	0x0B
Access	R	Default Value	0x00
Register Name	Shutter_Middle		
Bank	0	Address	0x0C
Access	R	Default Value	0x00
Register Name	Shutter_Upper		
Bank	0	Address	0x0D
Access	R	Default Value	0x00

Bit Field	Name	Default Value	Description
6:0	S[22:16]	0	Upper 7-bit of the 23-bit *Shutter* register.
Description	Unit is clock cycles of the internal oscillator/ 4. Read *Shutter_Upper* first, followed by *Shutter_Middle* and then *Shutter_Lower.* They should be read consecutively. The shutter is adjusted to keep the average raw data values within the normal operating range. The shutter value is checked and automatically adjusted to a new value if needed on every frame when operating in default mode.		

8.2.4 Troubleshooting

Register Name	Observation		
Bank	0	**Address**	0x15
Access	R/W	**Default Value**	0x00
Description	The user must clear the register by writing 0x00, wait for 30 ms, and read the register. The active processes bit[5:0] will have set their corresponding bits. The read-back value for bit[5:0] should be 0x3F. Bit[7:6] indicates the current operation mode of the chip.		

Bit Field	Name	Default Value	Description
7:6	OB[7:6]	0	0: Bright Mode 1: Low Light Mode 2: Super Low Light Mode 3: Reserved
5:0	OB[5:0]	0	0x3F: chip is working correctly Other value: recommend issuing a software reset command to the chip and configuring the register as per section 7.1 in this datasheet.

Revision History

Revision Number	Date	Description
0.8	04 Sept 2020	Initial Release
0.81	29 Nov 2021	Section 4.2.1: add in a note “Bottom ground pad of LGA package must be connected to circuit ground” under Figure 8. Section 4.3.2: update the exploded view of the assembly drawing in Figure 11.
1.0	5 July 2022	Section 2.6: Updated Figure 4. Resolution versus Height Chart. Section 0: Updated Figure 5. Speed versus Height Chart (Mode 0 & 1) Added Figure 6. Speed versus Height Chart (Mode 2) Section 7.1.2: Modified the description for Enhance Detection Setting Section 7.3.1: Changed BYTE[06] from Reserved to Challenging_Conditions Section 7.4.5: New section added for Challenging Conditions Detection Section 7.5: Added registers setting to disable LED control
1.1	15 Nov 2023	Change the packing type from tube to tape and reel Ordering Information: Update packing type for PAA3905 and MOQ for PAA3905 to 2500 and Lens L242 to 2880 Section 2.6: Added full formula and examples for Figure 4 Section 3.3: Updated Packing Information Figure 8: Change to carrier tape dimension Figure 30: Updated Raw Data Map Section 8.2.2: 1. Revised description and added example for *Resolution* register (address 0x4E) 2. Corrected the bits’ definition for *Orientation* register (address 0x5B) Section 5.3.3 and 7.1.1: Revised the statement for NCS to reset SPI port
1.2	05 Jul 2024	Section 1.4: Labelled 4 pads as “Unused pads” in Figure 2 and added a note for those pads must be left unconnected. Table 3: 1. Added min. SPI clock frequency for burst read and individual register read and write 2. Revised working distance to be from top of package to surface. Added Figure 3 for illustration of working distance. Section 2.6: Added example of converting motion counts to travel distance Table 7: Added new row of date code that is on top of the package Figure 8: Updated the drawing, dimension, and tolerance of carrier tape Section 4.3.1: Updated the graph in Figure 11 and also respective legends’ description in Table 8. Section 6.1: Added SPI mode 3 for SPI clock polarity and phase Section 7.4.4 & 7.4.7: Corrected the motion data register limit to -32767 to 32767 Figure 33: Added reference of external LED circuitry schematic Section 7.5: Remove the statement to connect pull up 1kΩ resistor to LED_SYNC, as it is not needed