PAW3399DM-T4QU: Optical Gaming Navigation Chip

General Description

PAW3399DM-T4QU is PixArt Imaging's new low power high end gaming navigation chip with illumination source in a 16-pin molded lead-frame DIP package. It provides best in class gaming experience with the enhanced features of high speed, high resolution, high accuracy and selectable lift detection height to fulfill professional gamers' need. It is designed to be used with LM19-LSI or L0AE-LSI1 to achieve optimum performance.

Key Features

- Low power consumption of typical 2.4mA in run mode (HP Mode)
- 16-pin molded lead-frame DIP package with 850nm illumination source
- Enhanced programmability
  - Gaming Mode
    - High Performance Mode (HP Mode)
    - Low Power Mode (LP Mode)
    - Corded Gaming Mode
  - Lift detection options
    - 1mm, 2mm and 3mm setting
    - Manual lift cut off calibration
  - Selectable resolutions up to 20000cpi with 50cpi step size
  - Angle snapping
  - Angle tunability
- Resolution error of 0.4% (typical) at 5000cpi on QCK up to 200ips
- High speed motion detection 650ips* (typical) and acceleration 50g* (typical)
- Self–adjusting variable frame rate for optimum performance
- Internal oscillator — no clock input needed
- 4-wire serial port interface (SPI)
- Motion interrupt output
- Applications
  - Corded and cordless optical gaming mice
  - Integrated input devices

Key Parameters

Parameter	Value
Power supply Range (V)	VDD: 1.80 – 2.05 VDDIO: 1.80 – 3.30
Lens Magnification	1:1
Interface	4-wire Serial Port Interface
Typical Operating Current @ VDD = 1.9V Note: includes LED current	Run: 2.4 mA (HP Mode) Run: 1.7 mA (LP Mode) Rest1: 680 µA Rest2: 15 µA Rest3: 6 µA Power Down: 4 µA
Resolution (cpi)	Up to 20000
Tracking Speed (ips)	650* (typical)
Acceleration (g)	50* (typical)
Package Type	16 pin molded lead frame DIP package assemble with lens 10.90 x 16.20 x 9.81 mm

Note: * - HP Mode

Ordering Information:

Part Number	Package Type
PAW3399DM-T4QU	16-pin DIP
LM19-LSI	Lens
L0AE-LSI1	Trim Lens

RoHS compliant

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

Table of Contents

List of Figures

List of Tables

1.0 Introduction

1.1 Chip Overview

PAW3399DM is an optical navigation chip targeted for high-end cordless and corded gaming mouse. It contains a picture element array as Image Acquisition System (IAS), a Digital Signal Processor (DSP), a 4-wire serial port, a power control circuit and built-in LED driver integrated with IR LED in a package as shown in the block diagram. The chip measures changes in position by optically acquiring sequential surface images (frames) and mathematically determining the direction and magnitude of movement. The IAS acquires microscopic surface images via the lens and illumination system. These images are processed by the DSP to determine the direction and distance of motion. The DSP calculates the Δx and Δy relative displacement values. An external microcontroller reads the Δx and Δy information from the chip serial port. The microcontroller then translates the data into USB, or RF signals before sending them to the host PC or game console.

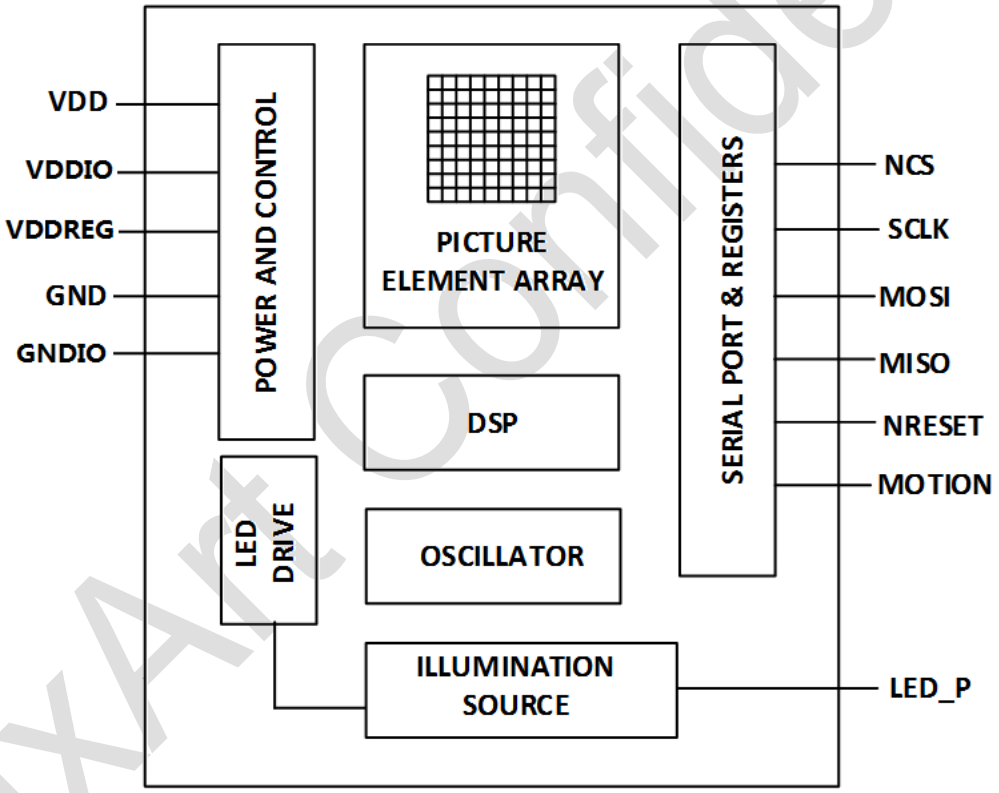


Figure 1. Block Diagram

1.2 Pin Configuration

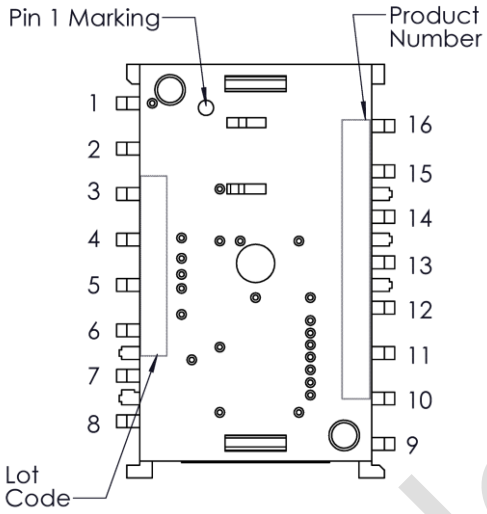


Figure 2. Device Pinout

Table 1. Pin Definition

Pin No.	Function	Symbol	Type	Description
1	Reserved	NC	NC	No connection
2	Reserved	NC	NC	No connection
3	Supply Ground	GND	GND	Ground
4	Supply Voltage	VDD	Power	Input power supply
5	LDO Output	VDDREG	Power	LDO output for digital core (only for internal usage)
6	Reserved	NC	NC	No connection
7	I/O Voltage	VDDIO	Power	I/O power supply
8	I/O Ground	GNDIO	GND	I/O Ground
9	Motion Output	MOTION	Output	Motion detect
10	4-wire SPI	SCLK	Input	Serial data clock
11		MOSI	Input	Serial data input
12		MISO	Output	Serial data output
13		NCS	Input	Chip select (Active Low)
14	Reset Control	NRESET	Input	Chip reset (Active Low)
15	LED	LED_P	Input	LED Anode
16	Reserved	NC	NC	No connection

2.0 Mechanical Specifications

This section covers PAW3399's guidelines and recommendations in term of chip, lens & PCB assemblies.

2.1 Chip Package Dimension

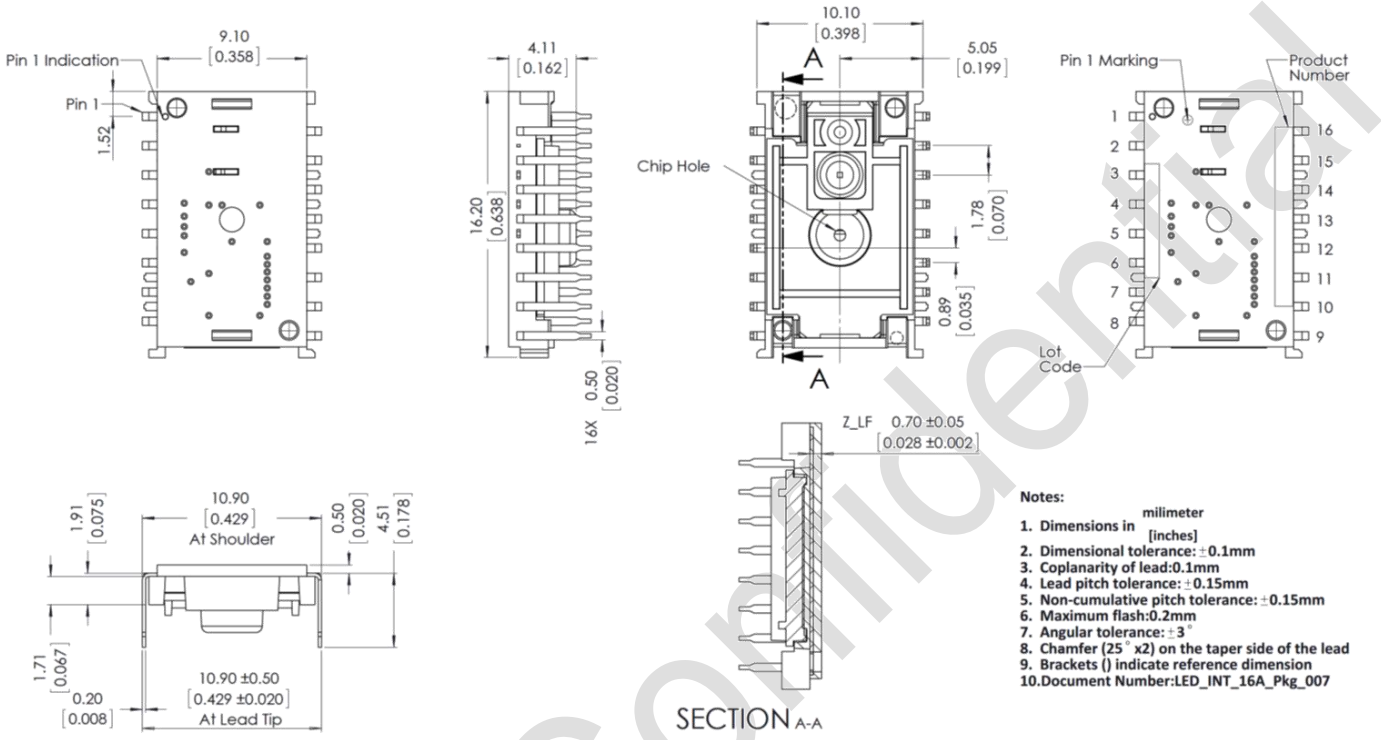


Figure 3. Packages Outline Drawing

CAUTION: It is advised that normal static discharge precautions be taken in handling and assembling of this component to prevent damage and/or degradation which may be induced by ESD.

2.2 Package Marking

Table 2. Package Marking Description

Items	Marking	Remark
Product Number	PAW3399DM-T4QU	
Lot Code	AYWWXXXXX	A : Assembly house Y: Year WW: Week XXXXX: PixArt reference

2.3 Chip Assembly Drawings

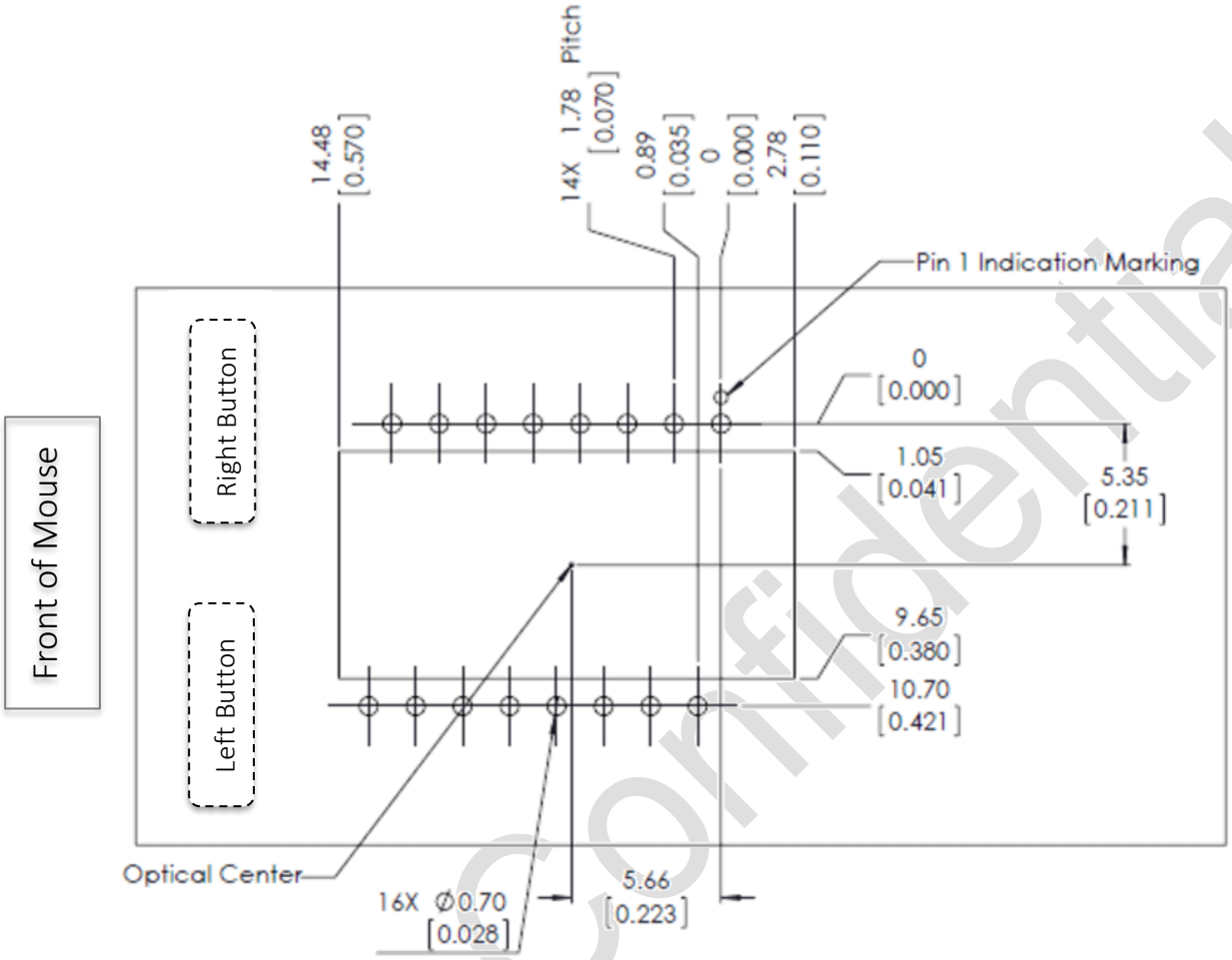


Figure 4. Recommended Chip Orientation, Mechanical Cutouts and Spacing (Top View)

Note: It is highly recommended to follow the chip orientation in Figure 4 to achieve optimum tracking performance.

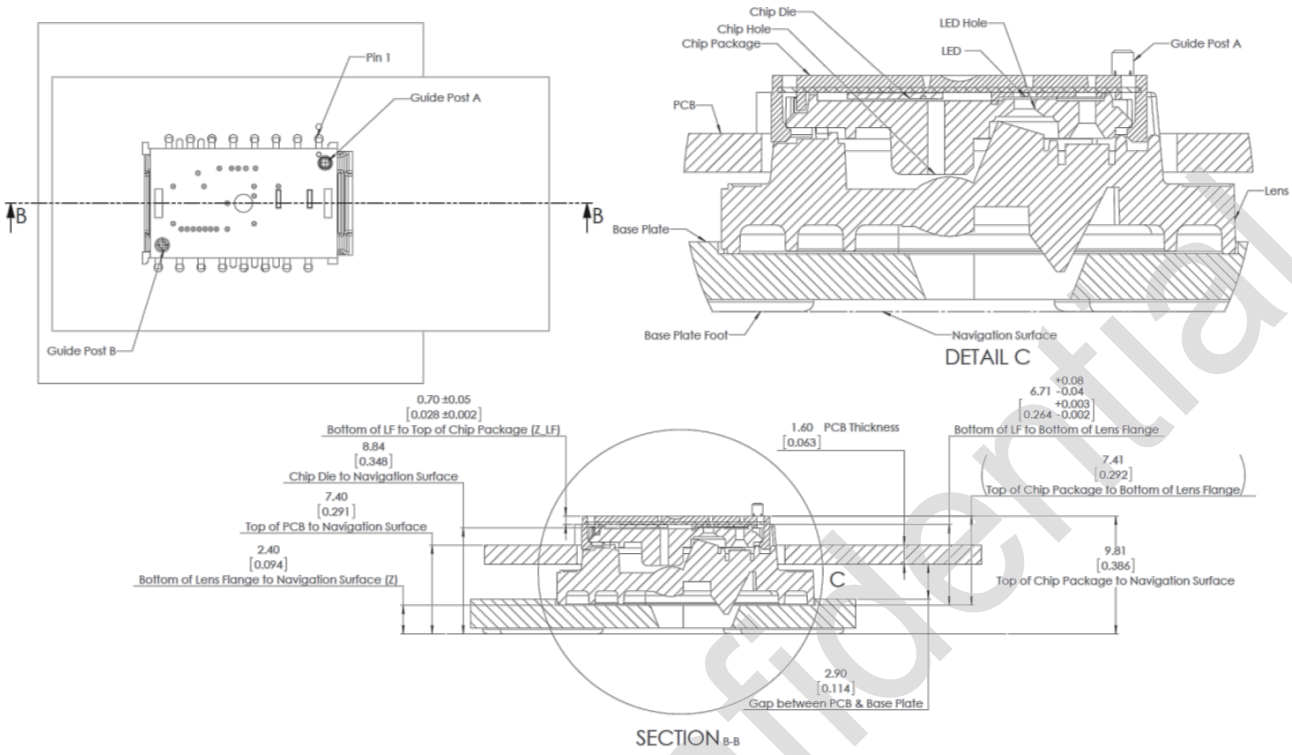


Figure 5. Assembly Drawing of PAW3399DM-T4QU and Distance from Lens Reference Plane To Tracking Surface (Z)

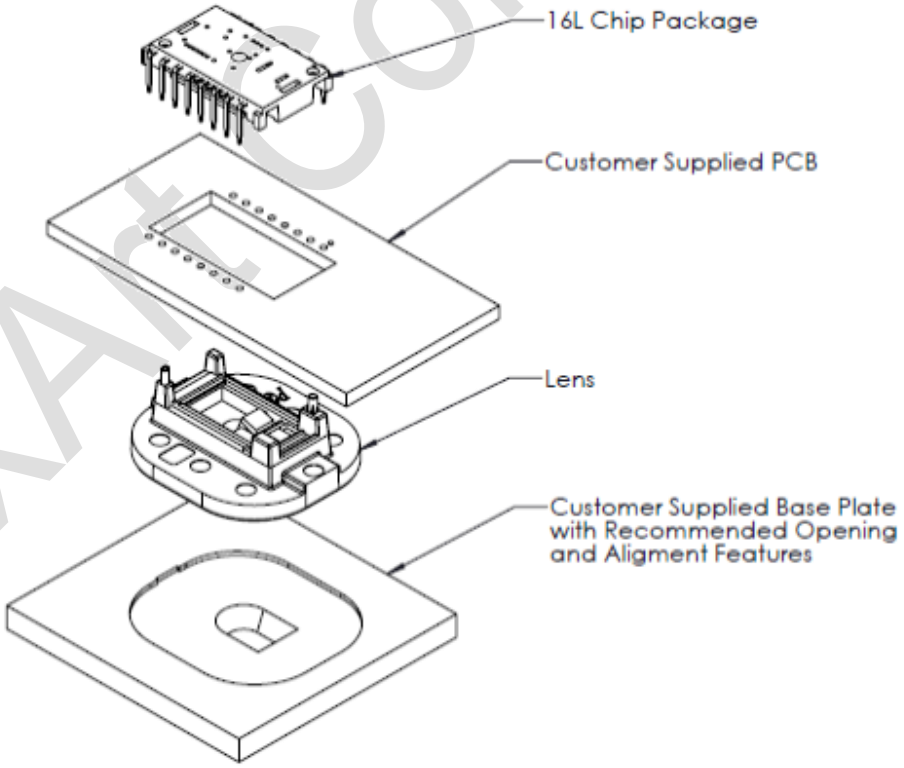


Figure 6. Exploded View of Assembly

2.4 Lens Dimensions

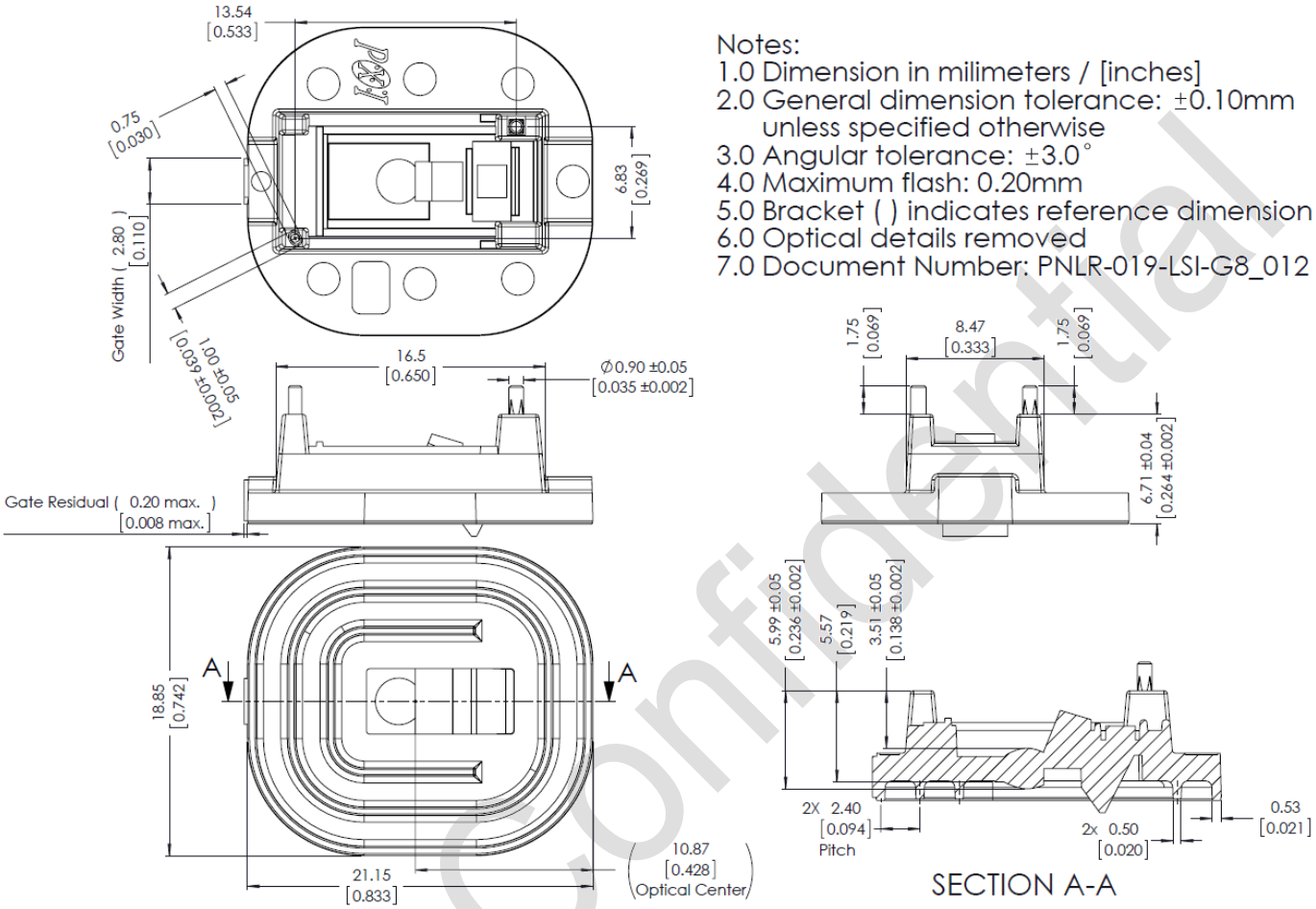


Figure 7. LM19-LSI Lens Outline Drawing

2.5 Lens Assembly Drawings

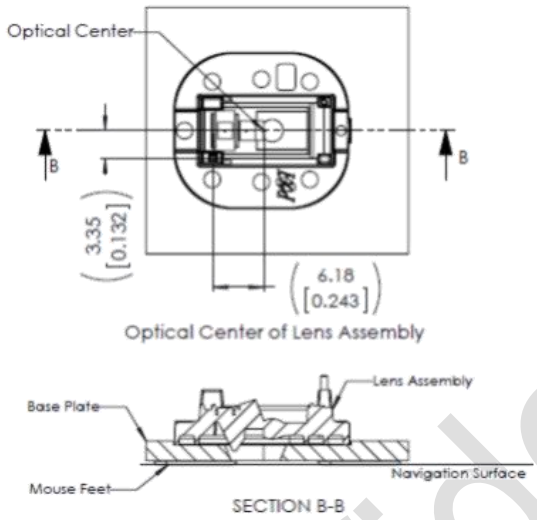


Figure 8. Cross Section View of LM19-LSI Lens Assembly

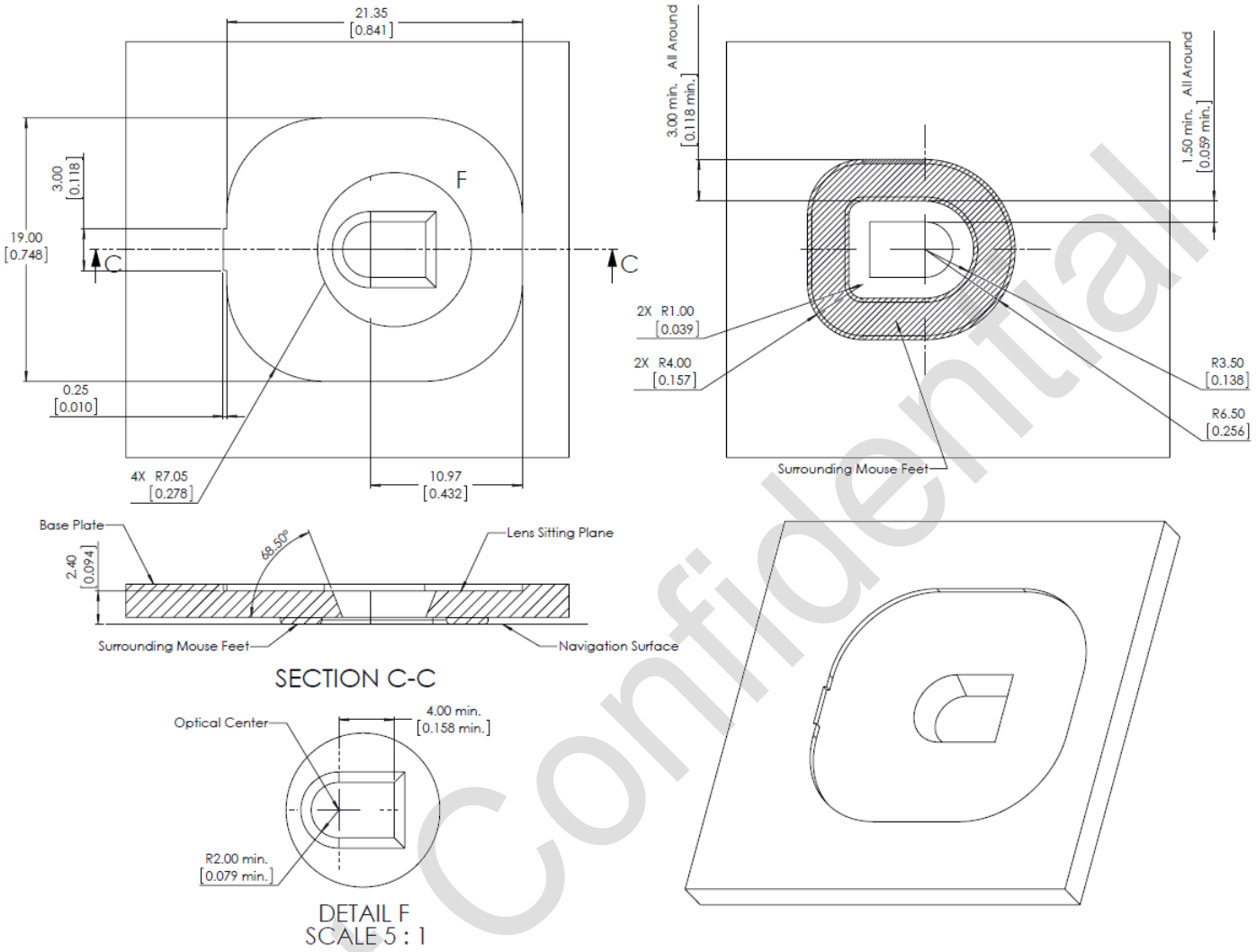


Figure 9. Recommended Base Plate Opening with LM19-LSI

Note: Surrounding mouse feet should be placed close to the optical opening to stabilize mouse tracking on the surface within the FOV of the chip.

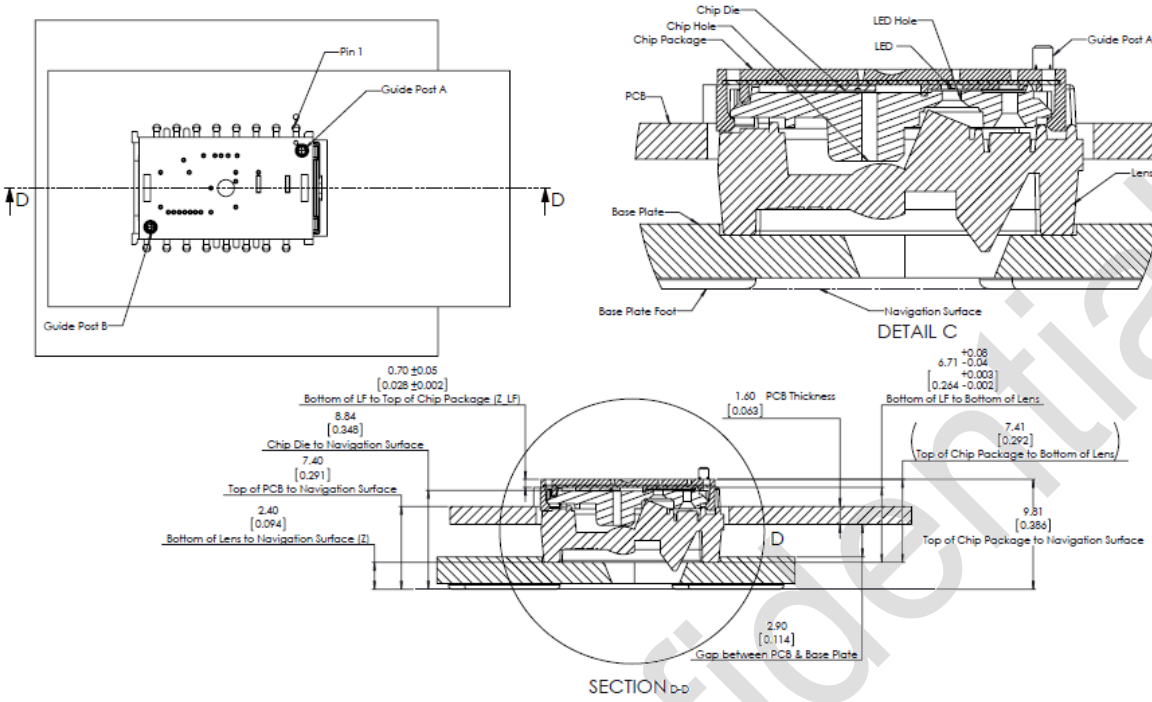


Figure 10. Assembly drawing of PAW3399DM-T4QU and distance from L0AE-LSI1 lens reference plane to tracking surface (Z)

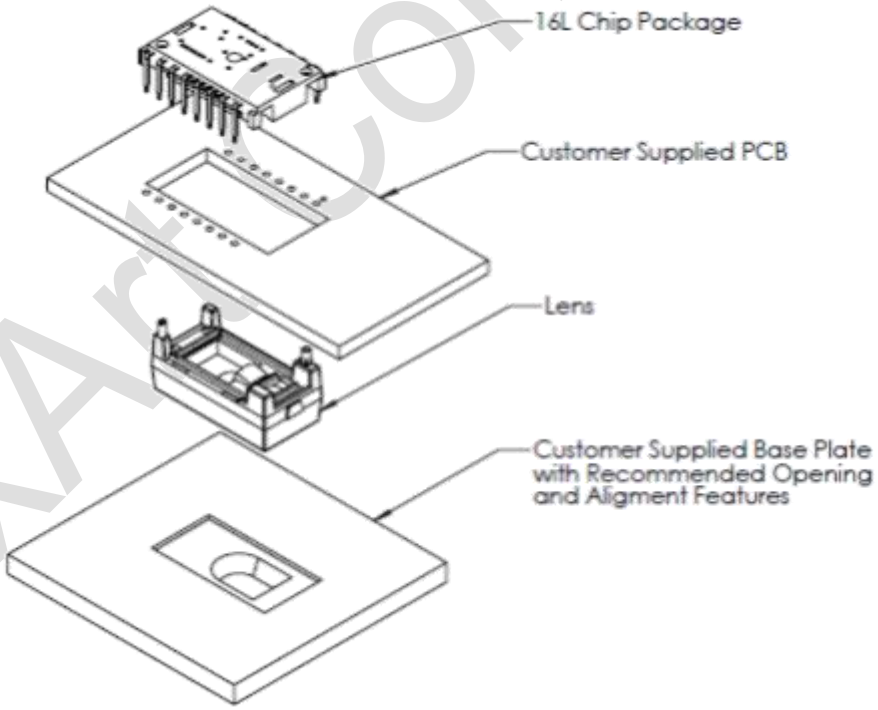


Figure 11. Exploded View of System Assembly with L0AE-LSI1 Lens

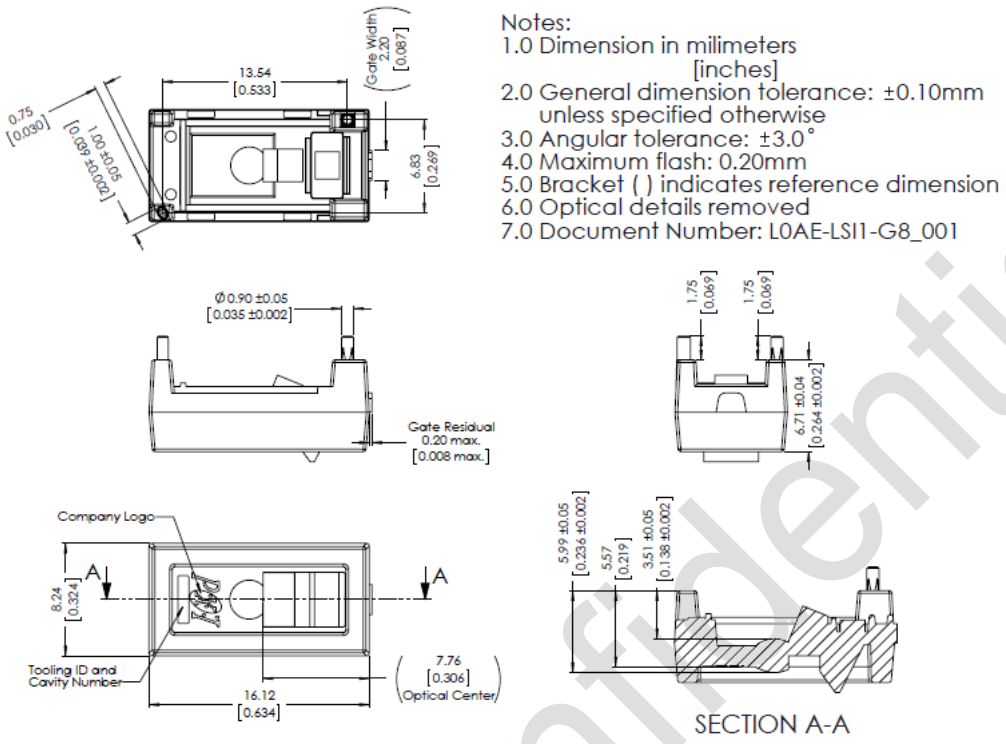


Figure 12. L0AE-LSI1 trim lens outline drawing and detail

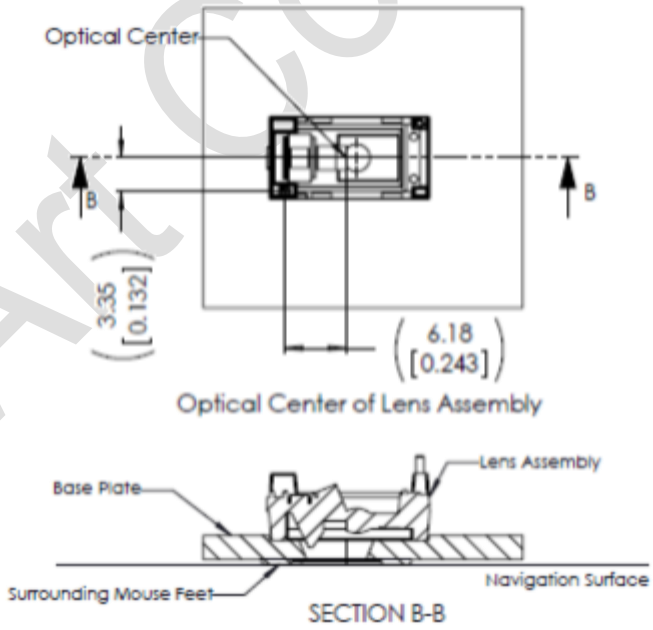


Figure 13. Cross Section View of L0AE-LSI1 lens Assembly

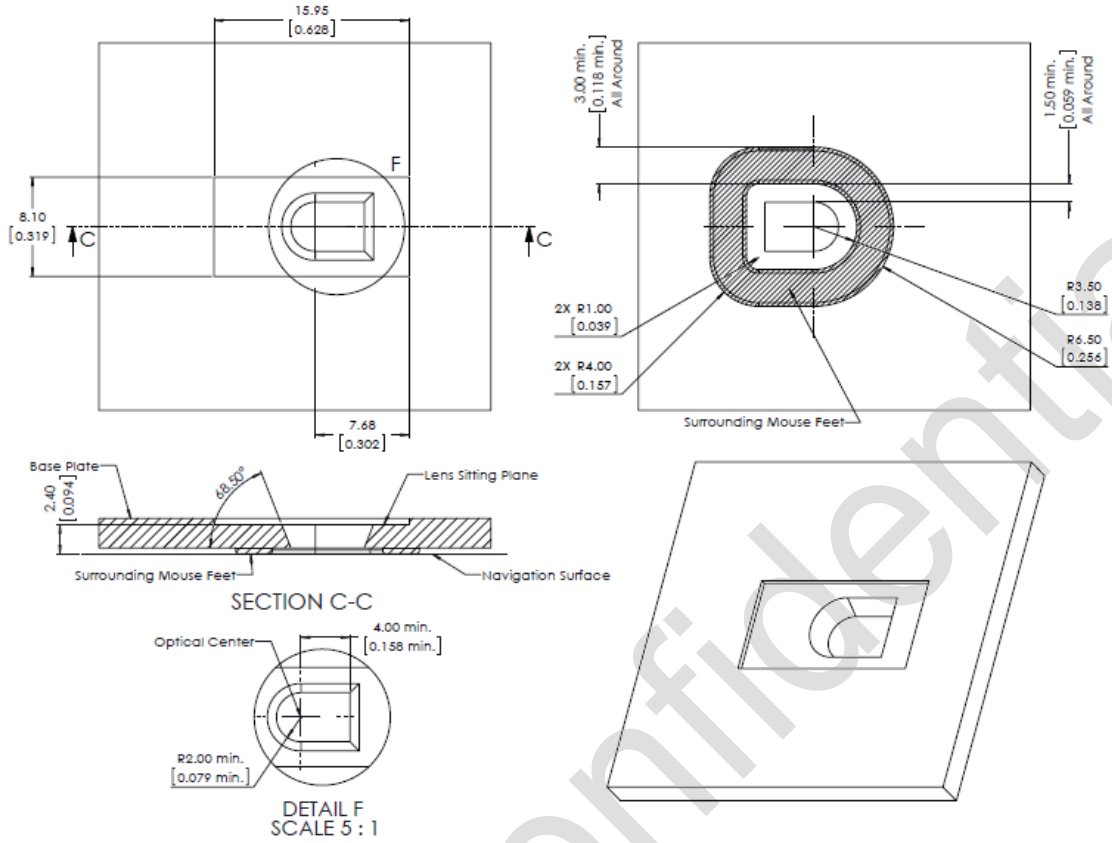


Figure 14. Recommended Base Plate Design with L0AE-LSI1 Lens

2.6 PCB Assembly Recommendations

1. Insert the integrated chip and all other electrical components into PCB.
2. Wave-solder the entire assembly in a no-wash solder process utilizing solder-fixture. A solder-fixture is required to protect the chip from flux spray and wave solder paste.
3. Avoid getting any solder flux onto the chip body as there is potential for flux to seep into the chip package, the solder fixture should be designed to expose only the chip leads to flux spray & molten solder while shielding the chip body and optical apertures. The fixture should also set the chip at the correct position and height on the PCB.
4. Place the lens onto the base plate. Care must be taken to avoid contamination on the optical surfaces.
5. Remove the protective Kapton tapes from optical apertures of the chip. Care must be taken to prevent contaminants from entering the apertures. Do not place the PCB with the chip facing up during the entire mouse assembly process. Hold the PCB vertically when removing Kapton tape.
6. Insert PCB assembly over the lens onto the base plate aligning post to retain PCB assembly. The chip package will self-align to the lens via the guide posts. The optical position reference for the PCB is set by the base plate and lens. Note that the PCB motion due to button presses must be minimized to maintain optical alignment.
7. Recommendation: The lens can be permanently secured to the chip package by melting the lens' guide posts over the chip with heat staking process. Please refer to Application Note titled *"LM19-LSI Lens: PCB Assembly & Lens Heat Staking Recommendations"* for details and recommendation on the lens heat staking process.
8. Install mouse top case. There must be a feature in the top case to press down onto the PCB assembly to ensure all components are stacked or interlocked to the correct vertical height.
9. It is recommended to place mouse feet around the base plate opening to stabilize mouse tracking on the surface.

3.0 Reference Schematics

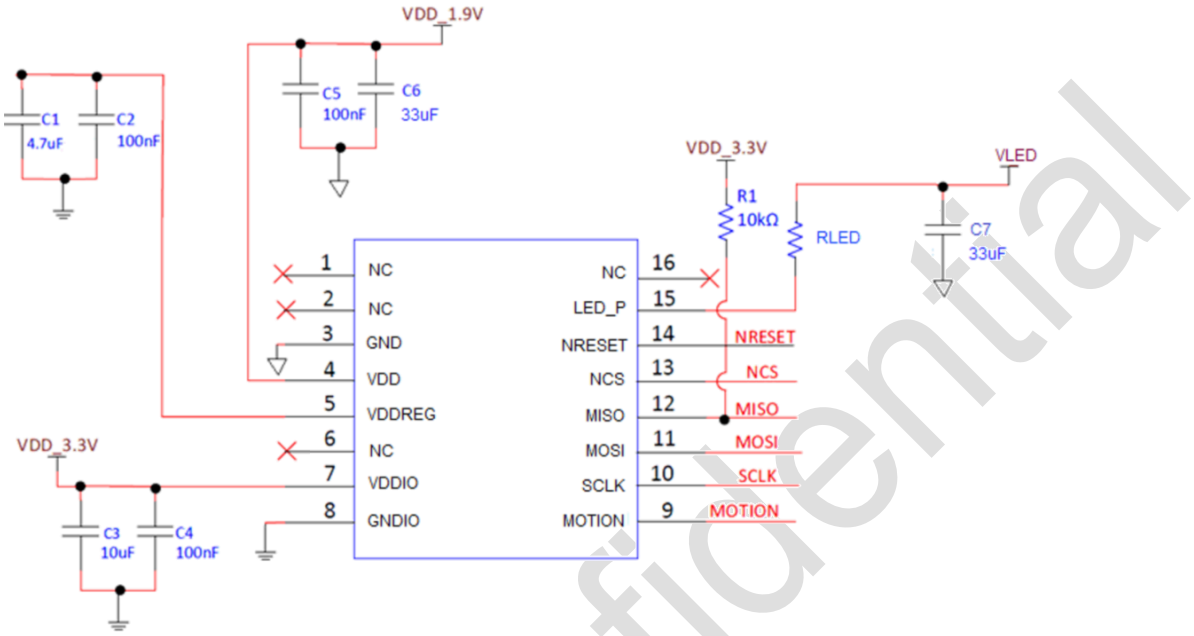


Figure 15. Reference Schematic diagram for PAW3399DM-T4QU

Note: It is not recommended to leave the NRESET pin floating, it should be constantly driven by an output pin from the microcontroller to establish its state.

Table 3 shows the recommended value of R_{LED} and V_{LED} to obtain 29mA current for LED. Recommend to use R_{LED} with 1% tolerance.

Table 3. Recommended R_{LED}

V_{LED} (V)	Recommended R_{LED} (Ω)
1.9V	13
2.0V	16

4.0 Electrical Specifications

4.1 Regulatory Requirements

- Passes FCC "Part15, Subpart B, Class B", "ICES-003:2016 Issue 6, Class B" and "ANSI C63.4:2014" when assembled into a mouse with shielded USB cable using ferrite bead and following PixArt's recommendations.
- Passes IEC 62471: 2006 Photo biological safety of lamps and lamp systems.

4.2 Absolute Maximum Ratings

Table 4. Absolute Maximum Ratings

Parameter	Symbol	Min	Max	Units	Notes
Storage Temperature	T_S	-40	85	°C	
Lead Solder Temperature	T_{SOLDER}		260	°C	For 7 seconds, 1.6mm below seating plane
Supply Voltage	V_{DD}	-0.5	2.05	V	
	V_{DDIO}	-0.5	3.30	V	
ESD	ESD_{HBM}		2	kV	Human Body Model on all pins
Input Voltage	V_{IN}	-0.5	3.30	V	All I/O pins

4.3 Recommended Operating Conditions

Table 5. Recommended Operating Condition

Parameter	Symbol	Min	Typ.	Max	Units	Notes
Operating Temperature	T_A	0		40	°C	
Power Supply Voltage	V_{DD}	1.80	1.90	2.05	V	Excluding supply noise
	V_{DDIO}	1.80	1.90	3.30	V	Excluding supply noise. (VDDIO must be the same or greater than VDD)
Power Supply Rise Time	t_{RT}	0.15		20	ms	0 to VDD min
Supply Noise	V_{NA}			100	mVp-p	10 kHz —75 MHz
Serial Port Clock Frequency	f_{SCLK}			10	MHz	50% duty cycle
Distance from Lens Reference Plane to Tracking Surface	Z	2.2	2.4	2.6	mm	
Speed High Performance Mode Low Power Mode Corded Gaming Mode Office Mode	S	 650 480 650 200			ips	In run mode at 45 degree
Acceleration High Performance Mode Low Power Mode Corded Gaming Mode Office Mode	A	 50 40 50 10			g	
Resolution Error High Performance Mode Low Power Mode Corded Gaming Mode	Res_{Err}		 0.4 0.4 0.4		%	Up to 200ips on QCK at 5000cpi
Lift Cutoff 1mm setting	$Lift_{1mm}$		1		mm	PixArt standard gaming surface
Lift Cutoff 2mm setting	$Lift_{2mm}$		2		mm	PixArt standard gaming surface
Lift Cutoff 3mm setting	$Lift_{3mm}$		3		mm	PixArt standard gaming surface

4.4 AC Electrical Specifications

Table 6. AC Electrical Specifications

Chip electrical characteristics over recommended operating conditions. Typical values at 25°C, VDD = 1.9V, VDDIO=1.9V

Parameter	Symbol	Min	Typical	Max	Units	Notes
Motion Delay After Reset	$t_{MOT-RST}$	50			ms	From reset to valid motion, assuming motion is present
Shutdown	t_{STDWN}			500	ms	From Shutdown mode active to low current
Wake From Shutdown	t_{WAKEUP}	50			ms	From Shutdown mode inactive to valid motion. Notes: A RESET must be asserted after a shutdown. Refer to section "Notes on Shutdown"
MISO Rise Time	t_{r-MISO}		6		ns	C_L = 20pF
MISO Fall Time	t_{f-MISO}		6		ns	C_L = 20pF
MISO Delay After SCLK	$t_{DLY-MISO}$			35	ns	From SCLK falling edge to MISO data valid C_L = 20pF
MISO Hold Time	$t_{hold-MISO}$	25			ns	Data held until next falling SCLK edge
MOSI Hold Time	$t_{hold-MOSI}$	25			ns	Amount of time data is valid after SCLK rising edge
MOSI Setup Time	$t_{setup-MOSI}$	25			ns	From data valid to SCLK rising edge
SPI Time Between Write Commands	t_{SWW}	5			µs	From rising SCLK for last bit of the first data byte, to rising SCLK for last bit of the second data byte
SPI Time Between Write And Read Commands	t_{SWR}	5			µs	From rising SCLK for last bit of the first data byte, to rising SCLK for last bit of the second address byte
SPI Time Between Read And Subsequent Commands	t_{SRW} t_{SRR}	2			µs	From rising SCLK for 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 last bit of the address byte, to falling SCLK for 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

Parameter	Symbol	Min	Typical	Max	Units	Notes
NCS To SCLK Active	$t_{NCS-SCLK}$	120			ns	From last NCS falling edge to first SCLK rising edge
SCLK To NCS Inactive (For Read Operation)	$t_{SCLK-NCS}$	120			ns	From last SCLK rising edge to NCS rising edge, for valid MISO data transfer
SCLK To NCS Inactive (For Write Operation)	$t_{SCLK-NCS}$	1			µs	From last SCLK rising edge to NCS rising edge, for valid MOSI data transfer
NCS To MISO High-Z	$t_{NCS-MISO}$			500	ns	From NCS rising edge to MISO high-Z state
MOTION Rise Time	$t_{r-MOTION}$		300		ns	C_L = 20pF
MOTION Fall Time	$t_{f-MOTION}$		300		ns	C_L = 20pF
Input Capacitance	C_{in}		10		pF	SCLK, MOSI, NCS
Load Capacitance	CL			20	pF	MISO, MOTION
Transient Supply Current	I_{DDT}			70	mA	Max supply current during the supply ramp from 0V to V_{DD} with min 150 us and max 20ms rise time. (Does not include charging currents for bypass capacitors)
	I_{DDTIO}			60	mA	Max supply current during the supply ramp from 0V to V_{DDIO} with min 150 us and max 20ms rise time. (Does not include charging currents for bypass capacitors)

4.5 DC Electrical Specifications

Table 7. DC Electrical Specifications

Chip electrical characteristics over recommended operating conditions. Typical values at 25°C, VDD = 1.9V, VDDIO = 1.9V, and with LED current at 29mA.

Parameter	Symbol	Min	Typ.	Max	Units	Notes
DC Supply Current (High Performance Mode)	IDD_{RUN} IDD_{REST1} IDD_{REST2} IDD_{REST3}		2.4 680 15 6		mA µA µA µA	IDD_{RUN} : Average current consumption, including LED current with 1ms polling IDD_{REST} : Average current consumption, including LED current
DC Supply Current (Low Power Mode)	IDD_{RUN} IDD_{REST1} IDD_{REST2} IDD_{REST3}		1.7 680 15 6		mA µA µA µA	
DC Supply Current (Corded Gaming Mode)	IDD_{RUN}		13		mA	IDD_{RUN} average up to 200ips IDD_{RUN} : Average current consumption, including LED current with 0.125ms polling
DC Supply Current (Office Mode)	IDD_{RUN} IDD_{REST1} IDD_{REST2} IDD_{REST3}		0.5 90 15 6		mA µA µA µA	IDD_{RUN} average up to 30ips IDD_{RUN} : Average current consumption, including LED current with 8ms polling IDD_{REST} : Average current consumption, including LED current
Shutdown Current	I_{PD}		4		µA	
Input Low Voltage	V_{IL}			0.3*VDDIO	V	SCLK, MOSI, NCS
Input High Voltage	V_{IH}	0.7*V_{DDIO}			V	SCLK, MOSI, NCS
Input Hysteresis	V_{I_HYS}		100		mV	SCLK, MOSI, NCS
Input Leakage Current	I_{leak}		±1	±10	µA	Vin=VDDIO or 0V, SCLK, MOSI, NCS
Output Low Voltage	V_{OL}			0.45	V	Iout = 1mA for MISO Iout = 0.1mA for MOTION
Output High Voltage	V_{OH}	V_{DDIO} - 0.45			V	Iout = -1mA for MISO Iout = -0.1mA for MOTION

5.0 Serial Peripheral Interface (SPI)

5.1 Signal Description

The synchronous serial port is used to write and read registers in the chip.

The port is a 4-wire port. The host microcontroller always initiates communication. The chip never initiates any data transfers. SCLK, MOSI and NCS may be driven directly by a microcontroller. The port pins may be shared with other SPI slave devices. When the NCS pin is driven high, the input signals are ignored and the output is tri-stated.

Table 8. SPI Port Signals Description

Signal Name	Functional Description
SCLK	Clock input, generated by the master (microcontroller).
MOSI	Input data. (Master Out/Slave In)
MISO	Output data. (Master In/Slave Out)
NCS	Chip select input (active low). NCS needs to be low to activate the serial port; otherwise, MISO will be high Z, and MOSI & SCLK will be ignored. NCS can also be used to reset the serial port in case of an error.

5.2 Motion Pin Timing

The motion pin is an active low output that signals the micro-controller when motion has occurred. The motion pin is lowered whenever the motion bit is set; 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. Clearing the motion bit (by reading Delta_X_L, Delta_X_H, Delta_Y_L or Delta_Y_H registers) will put the motion pin high.

5.3 Chip Select Operation

The serial port is activated after NCS goes low. If NCS is raised during a transaction, the entire transaction is aborted and the serial port will be reset. After a transaction is aborted, the normal address-to-data or transaction-to-transaction delay is required before beginning the next transaction. In order to improve communication reliability, all serial transactions should be framed by NCS. In other words, the port should not remain enabled during periods of non-use because any ESD and EFT/B event could be interpreted as serial communication and put the chip into an unknown state. In addition, NCS must be raised after each burst-mode transaction is complete or to terminate burst-mode operation. The port is not available for further use until burst-mode is terminated.

5.4 Write Operation

Write operation, defined as data going from the micro-controller to chip, is always initiated by the micro-controller and consists of two bytes. The first byte contains the address (seven bits) and has a "1" as its MSB to indicate data direction. The second byte contains the data. The chip reads MOSI on rising edges of SCLK.

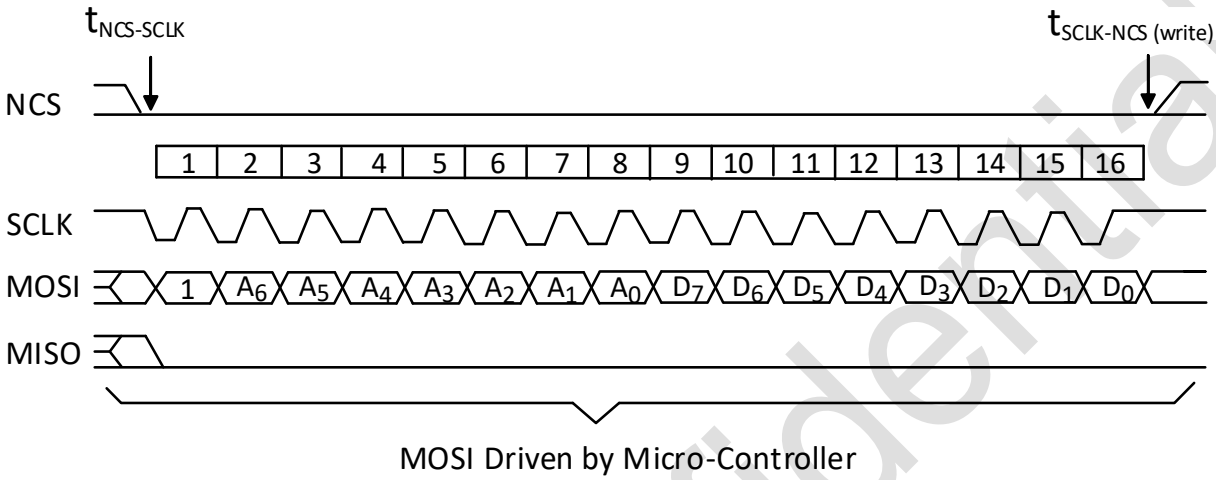


Figure 16. Write Operation

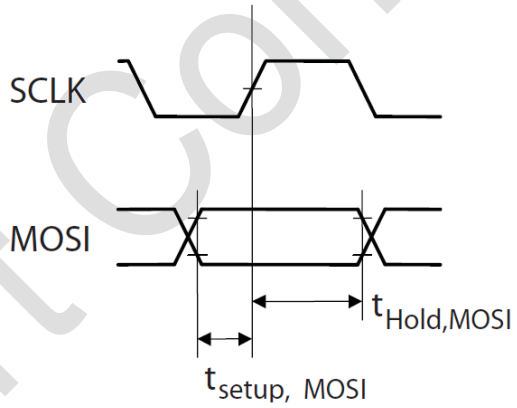


Figure 17. MOSI Setup and Hold Time

5.5 Read Operation

A read operation, defined as data going from chip to the micro-controller, is always initiated by the micro-controller and consists of two bytes. The first byte contains the address, is sent by the micro-controller over MOSI, and has a "0" as its MSB to indicate data direction. The second byte contains the data and is driven by PAW3399DM-T4QU chip over MISO. The chip outputs MISO bits on falling edges of SCLK and samples MOSI bits on every rising edge of SCLK.

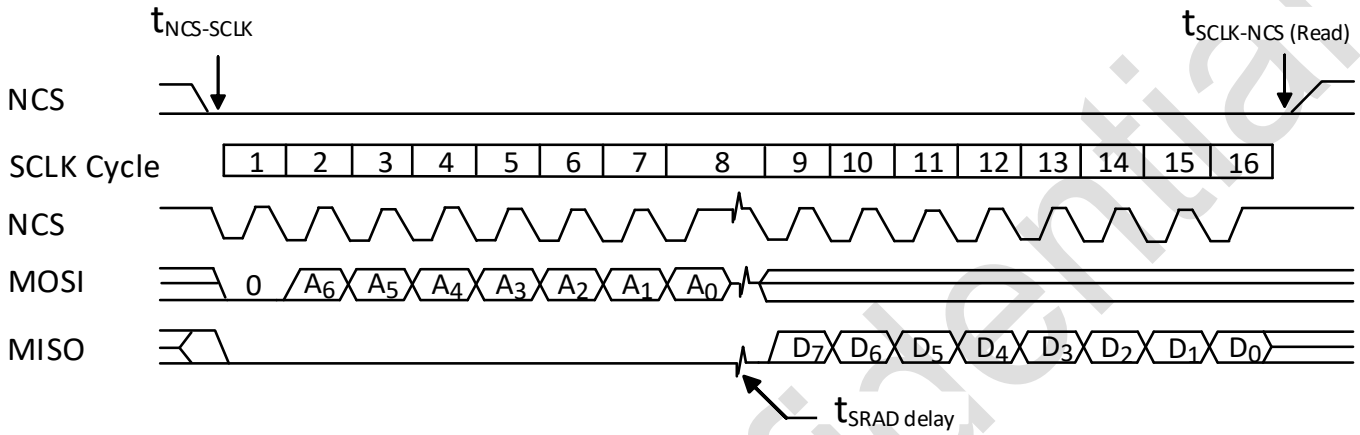


Figure 18. Read operation

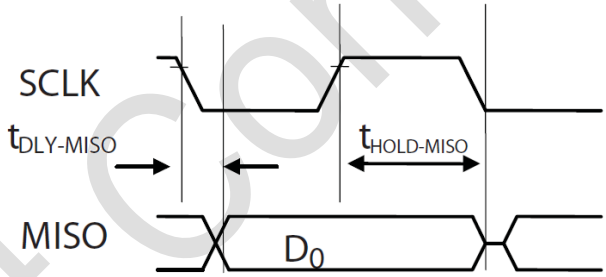


Figure 19. MISO Delay and hold time

Note: The minimum high state of SCLK is also the minimum MISO data hold time of PAW3399DM-T4QU chip. Since the falling edge of SCLK is actually 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.

5.6 Required timing between Read and Write Commands (tsxx)

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.

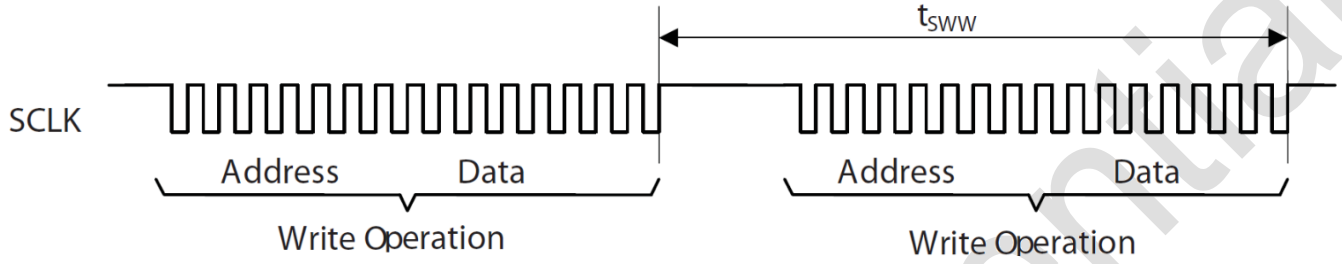


Figure 20. Timing between Two Write Commands

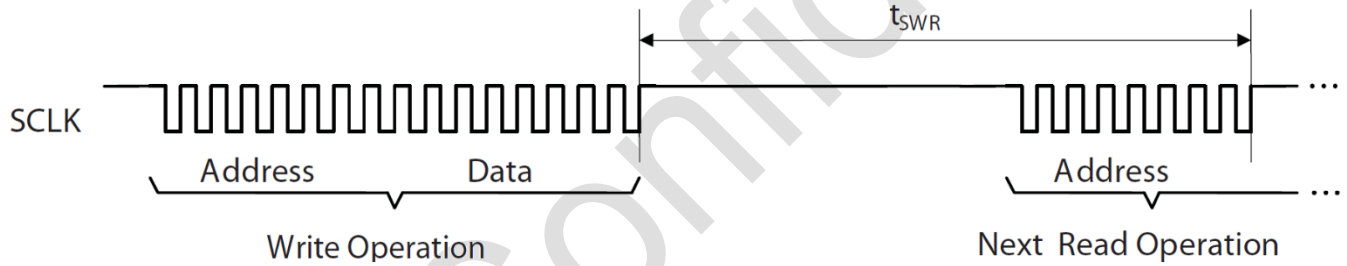


Figure 21. Timing between Write and Either Write or Subsequent Read Commands

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.

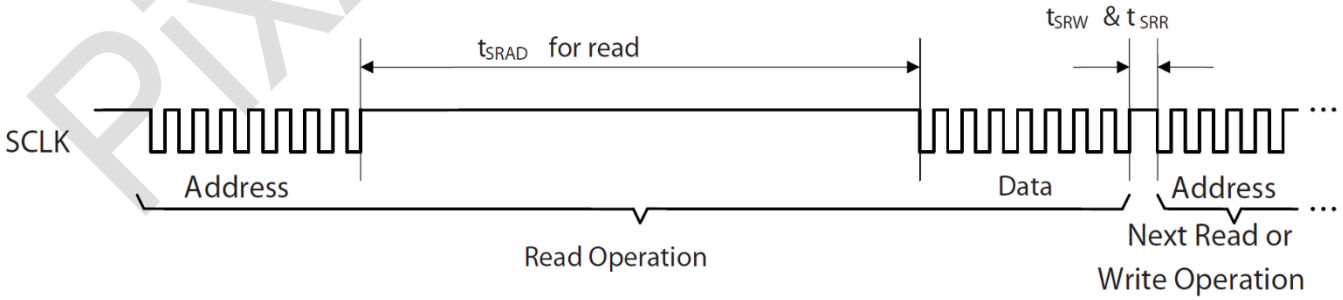


Figure 22. Timing between Read and Either Write or Subsequent Read Commands

5.7 Burst Mode Operation

Burst mode is a special serial port operation mode which is used to reduce the serial transaction time for predefined registers. The speed improvement is achieved by continuous data clocking to or from multiple registers without the need to specify the register address and by not requiring the normal delay period between data bytes.

5.7.1 Motion Read

Reading the Motion_Burst register activates the Motion Read mode. The chip will respond with the following motion burst report in this order.
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] = SQUAL
BYTE[07] = RawData_Sum
BYTE[08] = Maximum_RawData
BYTE[09] = Minimum_Rawdata
BYTE[10] = Shutter_Upper
BYTE[11] = Shutter_Lower

After sending the Motion_Burst register address, the microcontroller must wait for t_{SRAD}, and then begins reading data. All data bits can be read with no delay between bytes by driving SCLK at the normal rate. The data is latched into the output buffer after the last address bit is received. After the burst transmission is complete, the microcontroller must raise the NCS line for at least t_{BEXIT} to terminate burst mode. The serial port is not available for use until it is reset with NCS, even for a second burst transmission.

5.7.2 Procedure to Start Motion Burst

1. Lower NCS.
2. Wait for $t_{NCS\text{-}SCLK}$
3. Send Motion_Burst address (0x16). After sending this address, MOSI should be held static (either high or low) until the burst transmission is complete.
4. Wait for t_{SRAD}
5. Start reading SPI data continuously up to 12 bytes. Motion burst may be terminated by pulling NCS high for at least t_{BEXIT}.
6. To read new motion burst data, repeat from step 1.

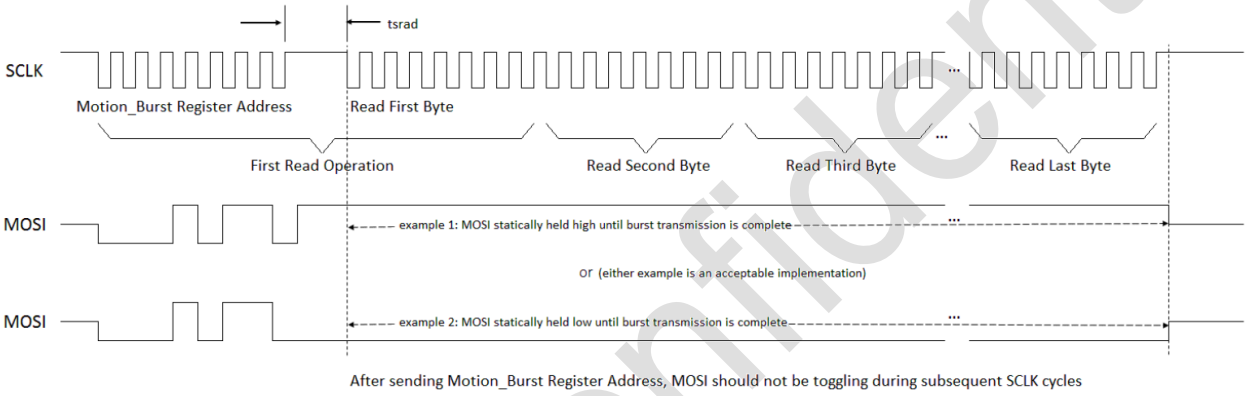


Figure 23. Motion Read Sequence

Note: Motion burst data can be read from the Burst_Motion_Read register even in run or rest mode.

6.0 Power-Up Sequences

6.1 Power-Up

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

1. Apply power to V_{DD} and V_{DDIO} in any order, with a delay of no more than 100ms in between each supply. Ensure all supplies are stable.
2. Wait for at least 50 ms.
3. Drive NCS high, and then low to reset the SPI port.
4. Write 0x5A to Power_Up_Reset register (or alternatively toggle the NRESET pin).
5. Wait for at least 5ms.
6. Load Power-up initialization register setting.
7. Read registers 0x02, 0x03, 0x04, 0x05 and 0x06 one time regardless of the motion bit state.

6.2 Power-Up Initialization Register Setting

1. Write register 0x40 with value 0x80
2. Write register 0x7F with value 0x0E
3. Write register 0x55 with value 0x0D
4. Write register 0x56 with value 0x1B
5. Write register 0x57 with value 0xE8
6. Write register 0x58 with value 0xD5
7. Write register 0x7F with value 0x14
8. Write register 0x42 with value 0xBC
9. Write register 0x43 with value 0x74
10. Write register 0x4B with value 0x20
11. Write register 0x4D with value 0x00
12. Write register 0x53 with value 0x0D
13. Write register 0x7F with value 0x05
14. Write register 0x51 with value 0x40
15. Write register 0x53 with value 0x40
16. Write register 0x55 with value 0xCA
17. Write register 0x61 with value 0x31
18. Write register 0x62 with value 0x64
19. Write register 0x6D with value 0xB8
20. Write register 0x6E with value 0x0F
21. Write register 0x70 with value 0x02
22. Write register 0x4A with value 0x2A
23. Write register 0x60 with value 0x26
24. Write register 0x7F with value 0x06

25. Write register 0x6D with value 0x70
26. Write register 0x6E with value 0x60
27. Write register 0x6F with value 0x04
28. Write register 0x53 with value 0x02
29. Write register 0x55 with value 0x11
30. Write register 0x7D with value 0x51
31. Write register 0x7F with value 0x08
32. Write register 0x71 with value 0x4F
33. Write register 0x7F with value 0x09
34. Write register 0x62 with value 0x1F
35. Write register 0x63 with value 0x1F
36. Write register 0x65 with value 0x03
37. Write register 0x66 with value 0x03
38. Write register 0x67 with value 0x1F
39. Write register 0x68 with value 0x1F
40. Write register 0x69 with value 0x03
41. Write register 0x6A with value 0x03
42. Write register 0x6C with value 0x1F
43. Write register 0x6D with value 0x1F
44. Write register 0x51 with value 0x04
45. Write register 0x53 with value 0x20
46. Write register 0x54 with value 0x20
47. Write register 0x71 with value 0x0F
48. Write register 0x72 with value 0x0A
49. Write register 0x7F with value 0x0A
50. Write register 0x4A with value 0x14
51. Write register 0x4C with value 0x14
52. Write register 0x55 with value 0x19
53. Write register 0x7F with value 0x14
54. Write register 0x63 with value 0x16
55. Write register 0x7F with value 0x0C
56. Write register 0x41 with value 0x30
57. Write register 0x55 with value 0x14
58. Write register 0x49 with value 0x0A
59. Write register 0x42 with value 0x00
60. Write register 0x44 with value 0x0D
61. Write register 0x4A with value 0x12
62. Write register 0x4B with value 0x09
63. Write register 0x4C with value 0x30

64. Write register 0x5A with value 0x0D
65. Write register 0x5F with value 0x1E
66. Write register 0x5B with value 0x05
67. Write register 0x5E with value 0x0F
68. Write register 0x7F with value 0x0D
69. Write register 0x48 with value 0xDD
70. Write register 0x4F with value 0x03
71. Write register 0x5A with value 0x29
72. Write register 0x5B with value 0x47
73. Write register 0x5C with value 0x81
74. Write register 0x5D with value 0x40
75. Write register 0x71 with value 0xDC
76. Write register 0x70 with value 0x07
77. Write register 0x73 with value 0x00
78. Write register 0x72 with value 0x08
79. Write register 0x75 with value 0xDC
80. Write register 0x74 with value 0x07
81. Write register 0x77 with value 0x00
82. Write register 0x76 with value 0x08
83. Write register 0x7F with value 0x10
84. Write register 0x4C with value 0xD0
85. Write register 0x7F with value 0x00
86. Write register 0x4F with value 0x63
87. Write register 0x4E with value 0x00
88. Write register 0x52 with value 0x63
89. Write register 0x51 with value 0x00
90. Write register 0x5A with value 0x10
91. Write register 0x77 with value 0x4F
92. Write register 0x47 with value 0x01
93. Write register 0x5B with value 0x40
94. Write register 0x66 with value 0x13
95. Write register 0x67 with value 0x0F
96. Write register 0x78 with value 0x01
97. Write register 0x79 with value 0x9C
98. Write register 0x55 with value 0x02
99. Write register 0x23 with value 0x70
100. Write register 0x22 with value 0x01
101. Wait for 1ms

102. Read register 0x6C at 1ms interval until value 0x80 is obtained or read up to 60 times, this register read interval must be carried out at 1ms interval with timing tolerance of +/-1%
103. If value of 0x80 is not obtained from register0x6C after 60 times:
   a. Write register 0x7F with value 0x14
   b. Write register 0x6C with value 0x00
   c. Write register 0x7F with value 0x00
104. Write register 0x22 with value 0x00
105. Write register 0x55 with value 0x00
106. Write register 0x7F with value 0x00
107. Write register 0x40 with value 0x00

During power-up there will be a period of time after the power supply is high but before normal operation. The table below shows the state of the various pins during power-up and reset.

Table 9. State of Signal Pins After VDD is Valid

Pin	During Reset	After Reset
NRESET	Functional	Functional
NCS	Ignored	Functional
MISO	Undefined	Depends on NCS
SCLK	Ignored	Depends on NCS
MOSI	Ignored	Depends on NCS
MOTION	Undefined	Functional

6.3 NRESET

The NRESET pin is used to perform the chip full chip reset. When asserted, it performs the same reset function as the Power_Up_Reset_Register. The NRESET pin needs to be asserted (held to logic 0) for at least 100 ns duration for the chip to reset.

Note: NRESET pin has a built in weak pull up circuit. During active low reset phase, the NRESET pin can draw a static current of up to 600µA.

7.0 Operation Guides

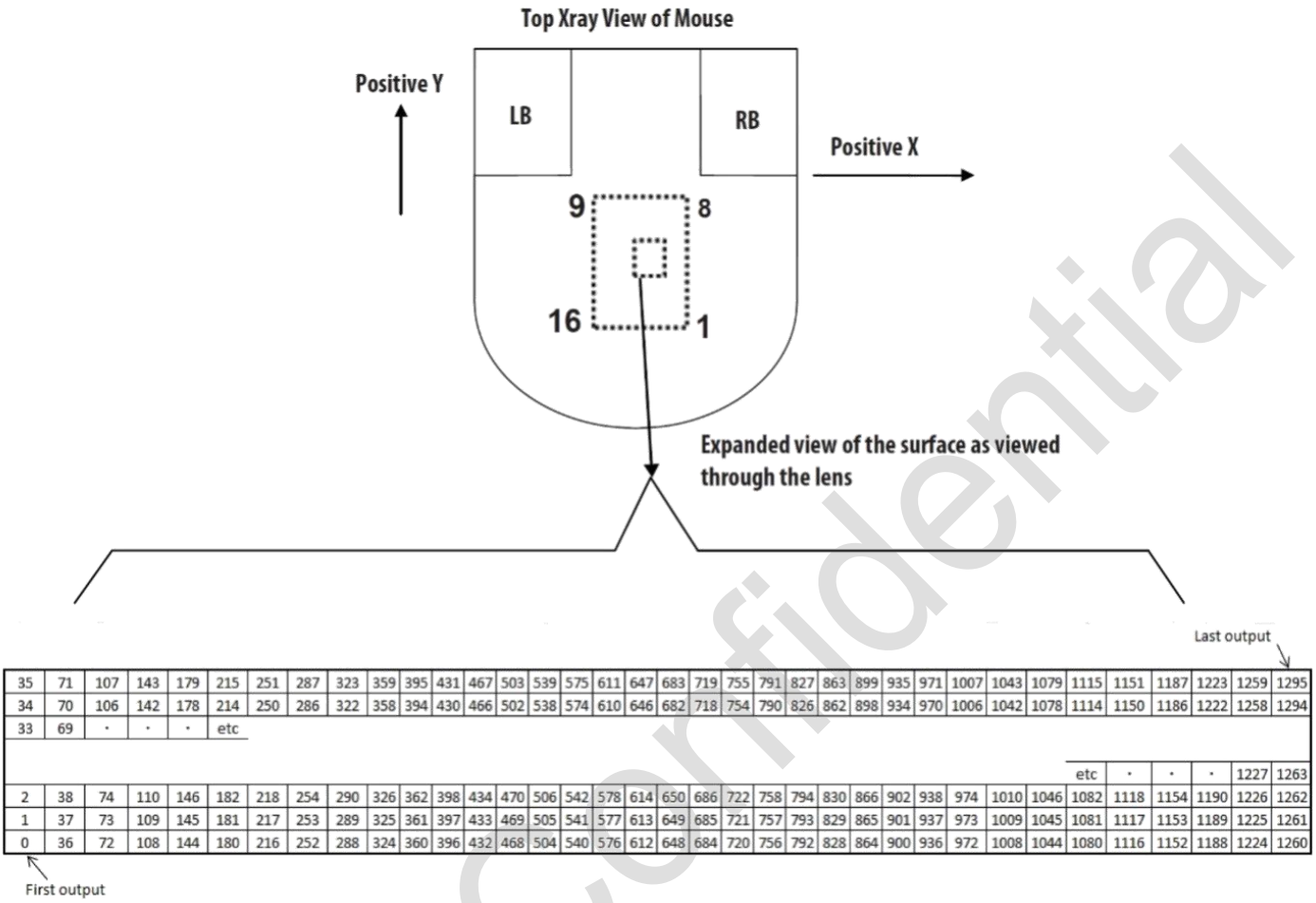
7.1 RawData Output

This section describes the method to download a full array of RawData values.

In order to trigger the RawData Output, write to the RawData_Grab register. The one element of rawdata is retrieved by reading the RAWDATA_GRAB register using register read method after RAWDATA_GRAB_STATUS register reports PG_VALID to be TRUE. During the RawData Output process, it is a MUST to place the mouse at stationary position.

RawData Output procedure:

1. The chip should be powered up and reset correctly.
2. Write register 0x7F with value 0x00
3. Write register 0x55 with value 0x04
4. Write register 0x50 with value 0x01
5. Write register 0x40 with value 0x80
6. Continuously read register 0x02 (Motion) until getting both OP_Mode_1 and OP_Mode_0 equal to 0.
7. Write register 0x58 with value 0xFF
8. Continuously read register 0x59 until getting both PG_FIRST and PG_VALID as "1"
9. Read the first rawdata from register 0x58
10. Continuously read register 0x59 until getting PG_VALID is "1".
11. Read register 0x58 for 7 bits ADC data (RAWDATA 6-0). Repeat (10) and (11) for 1295 times to form a complete picture element array information.
12. Write register 0x40 with value 0x00
13. Write register 0x50 with value 0x00
14. Write register 0x55 with value 0x00



35	71	107	143	179	215	251	287	323	359	395	431	467	503	539	575	611	647	683	719	755	791	827	863	899	935	971	1007	1043	1079	1115	1151	1187	1223	1259	1295
34	70	106	142	178	214	250	286	322	358	394	430	466	502	538	574	610	646	682	718	754	790	826	862	898	934	970	1006	1042	1078	1114	1150	1186	1222	1258	1294
33	69	·	·	·	etc																														
																														etc	·	·	·	1227	1263
2	38	74	110	146	182	218	254	290	326	362	398	434	470	506	542	578	614	650	686	722	758	794	830	866	902	938	974	1010	1046	1082	1118	1154	1190	1226	1262
1	37	73	109	145	181	217	253	289	325	361	397	433	469	505	541	577	613	649	685	721	757	793	829	865	901	937	973	1009	1045	1081	1117	1153	1189	1225	1261
0	36	72	108	144	180	216	252	288	324	360	396	432	468	504	540	576	612	648	684	720	756	792	828	864	900	936	972	1008	1044	1080	1116	1152	1188	1224	1260

Figure 24. RawData Map (Surface Referenced)

7.2 Shutdown

The chip can be set in Shutdown mode by writing to the Shutdown register 0x3B with value 0xB6. The SPI port should not be accessed when Shutdown mode is asserted except the power-up command (writing 0x5a to register 0x3a). Other ICs on the same SPI bus can be accessed so long as the chip's NCS pin is not asserted.

To de-assert Shutdown mode, please perform Power-Up sequence from step 2.

Table 10. Pin Status in Shutdown Mode

Pin	Status
NRESET	High
NCS	High*1
MISO	Hi-Z*2
SCLK	Ignore if NCS = 1*3
MOSI	Ignore if NCS = 1*4
MOTION	Output High

Notes:

*1. NCS pin must be held to 1 (high) if SPI bus is shared with other devices. It is recommended to hold to 1 (high) during Shutdown
unless powering up the Chip. It must be held to 0 (low) if the chip is to be re-powered up from shutdown (writing 0x5a to register 0x3a).

*2. MISO should be pulled up during shutdown in order to meet the low power consumption specification in the datasheet.

*3. SCLK is ignored if NCS is 1 (high). It is functional if NCS is 0 (low).

*4. MOSI is ignored if NCS is 1 (high). If NCS is 0 (low), any command present on the MOSI pin will be ignored except power-up command (writing 0x5a to register 0x3a).

CAUTION: *There is long wakeup time from shutdown. Shutdown should not be used for power management during normal mouse motion.*

7.3 Gaming and Office Mode Setting

PAW3399DM can be programmed to different gaming and office modes per the register settings in the table below. Please note that upon chip start-up per the recommended Power-Up Sequence, the chip is set to High Performance Mode as default.

High Performance Mode (Default)	Low Power Mode	Office Mode
- write register 0x7F with value 0x05 - write register 0x51 with value 0x40 - write register 0x53 with value 0x40 - write register 0x61 with value 0x31 - write register 0x6E with value 0x0F - write register 0x7F with value 0x07 - write register 0x42 with value 0x32 - write register 0x43 with value 0x00 - write register 0x7F with value 0x0D - write register 0x51 with value 0x00 - write register 0x52 with value 0x49 - write register 0x53 with value 0x00 - write register 0x54 with value 0x5B - write register 0x55 with value 0x00 - write register 0x56 with value 0x64 - write register 0x57 with value 0x02 - write register 0x58 with value 0xa5 - write register 0x7F with value 0x14 - write register 0x63 with value 0x16 - write register 0x7F with value 0x00 - write register 0x54 with value 0x54 - write register 0x78 with value 0x01 - write register 0x79 with value 0x9C - write register 0x40 bit[1:0] with value 0x0	- write register 0x7F with value 0x05 - write register 0x51 with value 0x40 - write register 0x53 with value 0x40 - write register 0x61 with value 0x3B - write register 0x6E with value 0x1F - write register 0x7F with value 0x07 - write register 0x42 with value 0x32 - write register 0x43 with value 0x00 - write register 0x7F with value 0x0D - write register 0x51 with value 0x00 - write register 0x52 with value 0x49 - write register 0x53 with value 0x00 - write register 0x54 with value 0x5B - write register 0x55 with value 0x00 - write register 0x56 with value 0x64 - write register 0x57 with value 0x02 - write register 0x58 with value 0xa5 - write register 0x7F with value 0x14 - write register 0x63 with value 0x16 - write register 0x7F with value 0x00 - write register 0x54 with value 0x54 - write register 0x78 with value 0x01 - write register 0x79 with value 0x9C - write register 0x40 bit[1:0] with value 0x1	- write register 0x7F with value 0x05 - write register 0x51 with value 0x28 - write register 0x53 with value 0x30 - write register 0x61 with value 0x3B - write register 0x6E with value 0x1F - write register 0x7F with value 0x07 - write register 0x42 with value 0x32 - write register 0x43 with value 0x00 - write register 0x7F with value 0x0D - write register 0x51 with value 0x00 - write register 0x52 with value 0x49 - write register 0x53 with value 0x00 - write register 0x54 with value 0x5B - write register 0x55 with value 0x00 - write register 0x56 with value 0x64 - write register 0x57 with value 0x02 - write register 0x58 with value 0xa5 - write register 0x7F with value 0x14 - write register 0x63 with value 0x16 - write register 0x7F with value 0x00 - write register 0x54 with value 0x52 - write register 0x78 with value 0x0A - write register 0x79 with value 0x0F - write register 0x40 bit[1:0] with value 0x02

Note:

Special precaution needs to be taken for register 0x40 to avoid overwrite other bits in the register. When writing the bit[1:0] to configure to different modes, one need to read and store its current value first, then apply bit masking and write back the new value into the register. Refer to section 8.3 for the detail.

Corded Gaming Mode
- write register 0x7F with value 0x05 - write register 0x51 with value 0x40 - write register 0x53 with value 0x40 - write register 0x61 with value 0x31 - write register 0x6E with value 0x0F - write register 0x7F with value 0x07 - write register 0x42 with value 0x2F - write register 0x43 with value 0x00 - write register 0x7F with value 0x0D - write register 0x51 with value 0x12 - write register 0x52 with value 0xDB - write register 0x53 with value 0x12 - write register 0x54 with value 0xDC - write register 0x55 with value 0x12 - write register 0x56 with value 0xE4 - write register 0x57 with value 0x15 - write register 0x58 with value 0x2D - write register 0x7F with value 0x14 - write register 0x63 with value 0x1E - write register 0x7F with value 0x00 - write register 0x54 with value 0x55 - write register 0x40 with value 0x83

7.4 Universal Lift Cut Off

PAW3399DM chip provides 1mm, 2mm and 3mm universal lift cut off setting and the setting applies to all mats, refer to *LIFT_CONFIG* register for the detail of lift cut off setting configuration. Upon ship start-up per the recommended Power-Up sequence in the datasheet, the chip is set to 1mm lift cut off setting as default.

7.5 Manual Lift Cut Off Calibration

PAW3399DM chip has the capability to optimize its lift performance by tuning parameters on a specific gaming mat or tracking surface, this feature involves end user interaction.

7.5.1 Lift Cut off Calibration Procedures

1. Ensured that the chip is powered up according to the Power Up Sequence in section 6.1.

2. Prompt the user that the manual lift cut off calibration is about to begin and ensure that the mouse is placed nominally on the surface (mouse is not lifted).

3. Start the calibration procedure by loading the following register values in sequence.
   - Read register 0x40 and store its value into Var_Mode
   - Write register 0x7F with value 0x00
   - Write register 0x40 with value 0x80
   - Write register 0x7F with value 0x05
   - Write register 0x43 with value 0xE7
   - Write register 0x7F with value 0x04
   - Write register 0x40 with value 0xC0
   - Write register 0x41 with value 0x10
   - Write register 0x44 with value 0x0F
   - Write register 0x45 with value 0x0F
   - Write register 0x46 with value 0x0F
   - Write register 0x47with value 0x0F
   - Write register 0x48 with value 0x0F
   - Write register 0x49 with value 0x0F
   - Write register 0x4A with value 0x0F
   - Write register 0x4B with value 0x0F
   - Write register 0x40 with value 0xC1

4. The calibration procedure can be started by a SW prompt to the user or user-initiated through a mouse-click event. Recommend to move the mouse over a distance of >20inch to cover most area of the mat.

5. Write register 0x40 with value 0x40 to stop the calibration process.

6. Continuously read register 0x4C bit[3:0] to check the status of the calibration process.
   - If returned value equals to 0x5 indicates the calibration is successful. Calibration can proceed to the next step or continue until user initiates a mouse-click event.
   - Else, the calibration is failed, load the following register values to return back to Universal 1mm setting and the calibration process need to be restarted from step 2.
     - Write register 0x4E with value 0x08
     - Write register 0x7F with value 0x05
     - Write register 0x43 with value 0xE4
     - Write register 0x7F with value 0x00
     - Write register 0x40 with Var_Mode

7. Write the following set of register values in sequence if the calibration is successful,
   - Read register 0x4D and store its value into VarA
   - Write register 0x7F with value 0x0C
   - If VarA >= 0x32, store value 0x05 into VarB. Else, store value 0x03 into VarB
   - Write register 0x4E with value 0x08
   - Write register 0x7F with value 0x05
   - Write register 0x43 with value 0xE4
   - Write register 0x7F with value 0x00
   - Write register 0x40 with Var_Mode

7.5.2 Enable Lift Cut off Calibration Register Setting

Write the following set of register values to enable the lift cut off calibration register setting on a specific gaming mat. VarA and VarB obtained from the section 7.5.1 would be used in this section.

1. Write register 0x7F with value 0x0C
2. Write register 0x41 with VarA
3. Write register 0x43 with value 0x30
4. Write register 0x44 with VarB
5. Write register 0x4E with value 0x08
6. Write register 0x5A with value 0x0D
7. Write register 0x5B with value 0x05
8. Write register 0x7F with value 0x05
9. Write register 0x6E with value 0x0F
10. Write register 0x7F with value 0x09
11. Write register 0x71 with value 0x0F
12. Write register 0x7F with value 0x00

7.5.3 Disable Lift Cut off Calibration Register Setting

Write the following set of register values to disable lift cut off calibration register setting in Section 7.5.2 and revert to default universal 1 mm lift cut off setting.

1. Write register 0x7F with value 0x0C
2. Write register 0x41 with value 0x30
3. Write register 0x43 with value 0x20
4. Write register 0x44 with value 0x0D
5. Write register 0x4A with value 0x12
6. Write register 0x4B with value 0x09
7. Write register 0x4C with value 0x30
8. Write register 0x4E with value 0x08
9. Write register 0x53 with value 0x16
10. Write register 0x55 with value 0x14
11. Write register 0x5A with value 0x0D
12. Write register 0x5B with value 0x05
13. Write register 0x5F with value 0x1E
14. Write register 0x66 with value 0x30
15. Write register 0x7F with value 0x05
16. Write register 0x6E with value 0x0F
17. Write register 0x7F with value 0x09

18. Write register 0x71 with value 0x0F
19. Write register 0x72 with value 0x0A
20. Write register 0x7F with value 0x00

8.0 Registers

8.1 Registers Summary Table

PAW3399DM-T4QU registers are accessible via the serial port. The registers are used to read motion data and status as well as to set the device configuration.

Table 11. Register List

Address	Register	Access	Default Value	Address	Register	Access	Default Value
0x00	Product_ID	R	0x4F	0x4A	Resolution _Y_Low	RW	0x63
0x01	Revision_ID	R	0x00	0x4B	Resolution _Y_High	RW	0x00
0x02	Motion	RW	0x00	0x56	Angle Snap	RW	0x0d
0x03	Delta_X_L	R	0x00	0x58	RawData output	R	0x00
0x04	Delta_X_H	R	0x00	0x59	RawData status	R	0x00
0x05	Delta_Y_L	R	0x00	0x5A	Ripple_Control	RW	0x00
0x06	Delta_Y_H	R	0x00	0x5B	Axis_Control	RW	0x60
0x07	SQUAL	R	0x00	0x5C	Motion_Ctrl	RW	0x02
0x08	RawData_Sum	R	0x00	0x5F	Inv_Product_ID	R	0XB0
0x09	Maximum_RawData	R	0x00	0x77	Run_Downshift	RW	0x14
0x0A	Minimum_RawData	R	0x00	0x78	Rest1_Rate	RW	0x01
0x0B	Shutter_Lower	R	0x00	0x79	Rest1_Downshift	RW	0x90
0x0C	Shutter_Upper	R	0x01	0x7A	Rest2_Rate	RW	0x19
0x15	Observation	RW	0x80	0x7B	Rest2_Downshift	RW	0x5E
0x16	Motion_Burst	RW	0x00	0x7C	Rest3_Rate	RW	0x3F
0X3A	Power_Up_Reset	W	N/A	0x7D	Run_Downshift_Mult	RW	0x07
0x3B	Shutdown	W	N/A	0x7E	Rest_Downshift_Mult	RW	0x55
0x40	Performance	RW	0x00	0x0577*	Angle_Tune1	RW	0x00
0x47	Set_Resolution	W	0x00	0x0578*	Angle_Tune2	RW	0x00
0x48	Resolution _X_Low	RW	0x63	0x0C4E*	Lift_Config	RW	0x08
0x49	Resolution _X_High	RW	0x00				

Note:

1. R = Read, W = Write, Read/Write= RW
2. * - In order to access the register:
   d. Write register 0x7F with the value of MSB(byte) in the address.
   e. Read/Write the register value with the lower byte address.
   f. Write register 0x7F with the value 0x00.

Example 1: To write register 0x**0D4F** (Dynamic Cursor Control) with value 0x01

*Write register 0x7F with value 0x***0D***,*

*Write register 0x***4F** *with value 0x01, //set to Dynamic Cursor Control Level 2*

Write register 0x7F with value 0x00

Example2: To read register 0x**0D4F** (Dynamic Cursor Control)

*Write register 0x7F with value 0x***0D***,*

*Read register 0x***4F***,*

Write register 0x7F with value 0x00

8.2 Registers Description

Register Name	PRODUCT_ID							
Address	0x00							
Access	Read			Reset Value		0x4F		
Bit	7	6	5	4	3	2	1	0
Field	PID_{7-0}							
Description	This register contains a unique identification assigned to the PAW3399DM-T4QU. 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							
Address	0x01							
Access	Read			Reset Value		0x00		
Bit	7	6	5	4	3	2	1	0
Field	PID_{7-0}							
Description	This register contains the current IC revision.							

Register Name	MOTION							
Address	0x02							
Access	Read/Write			Reset Value		0x00		
Bit	7	6	5	4	3	2	1	0
Field	MOT	Reserved	Reserved	Reserved	Lift_Stat	Reserved	OP_Mode_1	OP_Mode_0

Description

This register allows the user to determine if 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) is as follows:

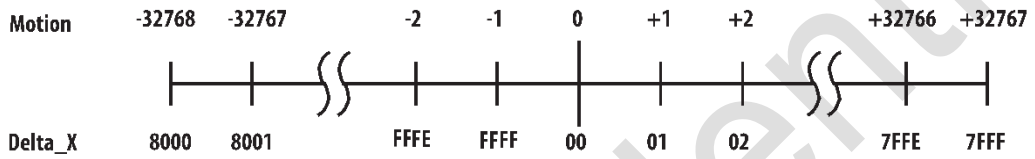
1. Read the Motion register. This will freeze the Delta_X_L, Delta_X_H, Delta_Y_L and Delta_Y_H register values.

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: 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 will be lost.

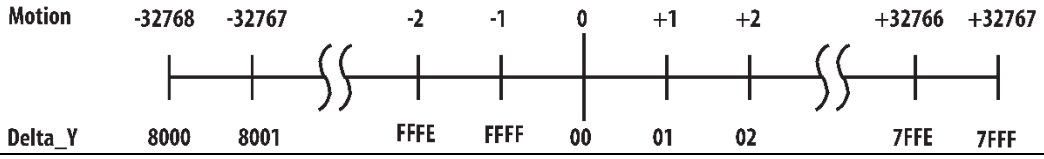
3. To read a new set of motion data (Delta_X_L, Delta_X_H, Delta_Y_L and Delta_Y_H), repeat from Step 1.

Field Name	Description
MOT	Motion since 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
Lift_Stat	Indicate the lift status of chip 0 – Chip on surface 1 – Chip lifted
$OP\_Mode_{1:0}$	00 – Run Mode 01 - Rest 1 10 - Rest 2 11 - Rest 3

Write any value to this register will clear all motion data.

Register Name	DELTA_X_L							
Address	0x03							
Access	Read			Reset Value		0x00		
Bit	7	6	5	4	3	2	1	0
Field	X_7	X_6	X_5	X_4	X_3	X_2	X_1	X_0
Description	16 bits 2's complement number. Lower 8 bits of Delta_X. X movement is counts since last report. Absolute value is determined by resolution. 							

Register Name	DELTA_X_H							
Address	0x04							
Access	Read			Reset Value		0x00		
Bit	7	6	5	4	3	2	1	0
Field	X_{15}	X_{14}	X_{13}	X_{12}	X_{111}	X_{10}	X_9	X_8
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 register 0x02, 0x03, 0x04, 0x05 and 0x06 to be read sequentially.*							

Register Name	DELTA_Y_L							
Address	0x05							
Access	Read			Reset Value		0x00		
Bit	7	6	5	4	3	2	1	0
Field	Y_7	Y_6	Y_5	Y_4	Y_3	Y_2	Y_1	Y_0
Description	16 bits 2's complement number. Lower 8 bits of Delta_Y. Y movement is counts since last report. Absolute value is determined by resolution. 							

Register Name	DELTA_Y_H							
Address	0x06							
Access	Read			Reset Value		0x00		
Bit	7	6	5	4	3	2	1	0
Field	Y_{15}	Y_{14}	Y_{13}	Y_{12}	Y_{11}	Y_{10}	Y_9	Y_8
Description	16 bits 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 register 0x02, 0x03, 0x04, 0x05 and 0x06 to be read sequentially.*							

Register Name	SQUAL							
Address	0x07							
Access	Read			Reset Value		0x00		
Bit	7	6	5	4	3	2	1	0
Field	SQ_7	SQ_6	SQ_5	SQ_4	SQ_3	SQ_2	SQ_1	SQ_0
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 * 4 The maximum SQUAL register value is 0xB6. Since small changes in the current frame can result in changes in SQUAL, variations in SQUAL when looking at a surface is expected. SQUAL values are only valid when chip is in run mode. Disable Rest mode before measuring SQUAL.							

Register Name	RAWDATA_SUM							
Address	0x08							
Access	Read			Reset Value		0x00		
Bit	7	6	5	4	3	2	1	0
Field	RDS_7	RDS_6	RDS_5	RDS_4	RDS_3	RDS_2	RDS_1	RDS_0
Description	This register is used to find the chip average rawdata value. It reports the upper byte of an 18-bit counter which sums all 1296 rawdata in the current frame. To find the average rawdata value follows the formula below: Average pixel value = PIX_ACCUM*1024/1296 The maximum register value is 0xA0 (hex) or 160 (dec) and the minimum register value is 0. The data sum value can change every frame. Disable rest mode before reading RawData sum value.							

Register Name	MAXIMUM_RAWDATA							
Address	0x09							
Access	Read			Reset Value		0x00		
Bit	7	6	5	4	3	2	1	0
Field	$MaxRD_7$	$MaxRD_6$	$MaxRD_5$	$MaxRD_4$	$MaxRD_3$	$MaxRD_2$	$MaxRD_1$	$MaxRD_0$
Description	Maximum RawData value in current frame. Minimum value = 0, maximum value = 127. The maximum rawdata value can change every frame.							

Register Name	MINIMUM_RAWDATA							
Address	0x0A							
Access	Read			Reset Value		0x7F		
Bit	7	6	5	4	3	2	1	0
Field	$MinRD_7$	$MinRD_6$	$MinRD_5$	$MinRD_4$	$MinRD_3$	$MinRD_2$	$MinRD_1$	$MinRD_0$
Description	Minimum RawData value in current frame. Minimum value = 0, maximum value = 127. The minimum rawdata value can change every frame.							

Register Name	SHUTTER_LOWER							
Address	0x0B							
Access	Read			Reset Value		0x00		
Bit	7	6	5	4	3	2	1	0
Field	S_7	S_6	S_5	S_4	S_3	S_2	S_1	S_0
Description	Lower byte of the 12-bit Shutter register.							

Register Name	SHUTTER_UPPER							
Address	0x0C							
Access	Read			Reset Value		0x01		
Bit	7	6	5	4	3	2	1	0
Field	Reserved	Reserved	Reserved	Reserved	S_{11}	S_{10}	S_9	S_8
Description	Upper 4-bit of the 12-bit Shutter register. Units are clock cycles of the internal oscillator(nominal 68MHz). Read Shutter_Upper first, then Shutter_Lower. They should be read consecutively. The shutter is adjusted to keep the average rawdata values within normal operating ranges. The shutter value is checked and automatically adjusted to a new value if needed on every frame when operating in default mode.							

Register Name	CHIP_OBSERVATION							
Address	0x15							
Access	Read/Write			Reset Value		0x80		
Bit	7	6	5	4	3	2	1	0
Field	CO_7	CO_6	CO_5	CO_4	CO_3	CO_2	CO_1	CO_0
Description	The user must clear the register by writing 0x00, wait for a minimum T_{dly_obs} ms & read the register. The value of CO_{7-0} should be 0xB7 or 0xBF if the chip is working correctly. The register may be used as part of recovery scheme to detect a problem caused by EFT/B or ESD event. T_{dly_obs} is defined as the longest frame period + 10% variation. The longest frame period is when chip is in Rest3 mode. Clock frequency tolerance value need to be considered. For example if the default Rest3 period of 500ms is used, then T_{dly_obs} = 500ms + 50ms							

Register Name	BURST_MOTION_READ							
Address	0x16							
Access	Read			Reset Value		0x00		
Bit	7	6	5	4	3	2	1	0
Field	MB_7	MB_6	MB_5	MB_4	MB_3	MB_2	MB_1	MB_0
Description	The Burst_Motion_Read register is used for high-speed access to the Motion, Observation, Delta_X_L, Delta_X_H, Delta_Y_L, Delta_Y_H, SQUAL, RawData_Sum, Maximum_RawData, Minimum_RawData, Shutter_Upper and Shutter_Lower registers. See Burst Mode-Motion Read section 5.8 for use details.							

Register Name	POWER_UP_RESET							
Address	0x3A							
Access	Write			Reset Value		N/A		
Bit	7	6	5	4	3	2	1	0
Field	$PRST_7$	$PRST_6$	$PRST_5$	$PRST_4$	$PRST_3$	$PRST_2$	$PRST_1$	$PRST_0$
Description	Write 0x5A to this register to reset the chip and all settings will revert to default values. Reset is required after recovering from Shutdown mode.							

Register Name	SHUTDOWN							
Address	0x3B							
Access	Write			Reset Value		N/A		
Bit	7	6	5	4	3	2	1	0
Field	SD_7	SD_6	SD_5	SD_4	SD_3	SD_2	SD_1	SD_0
Description	Write 0xB6 to set the chip to Shutdown mode. Refer to the Shutdown section for more details on recovery procedure.							

Register Name	PERFORMANCE							
Address	0x40							
Access	Read/Write			Reset Value		0x00		
Bit	7	6	5	4	3	2	1	0
Field	AWAKE	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved
Description	This register configures the operating mode of the chip.							

Field Name	Description
AWAKE	0: Enable Rest Mode 1: Disable Rest Mode

Register Name	SET_RESOLUTION							
Address	0x47							
Access	Write			Reset Value		0x00		
Bit	7	6	5	4	3	2	1	0
Field	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	SET_RES
Description	After update the resolution setting, either in RESOLUTION_X or/and RESOLUTION_Y, write value 0x01 into SET_RESOLUTION for the chip to use the new resolution setting.							

Field Name	Description
SET_RES	1: update resolution setting

Register Name	RESOLUTION_X_LOW							
Address	0x48							
Access	Read/Write			Reset Value		0x63		
Bit	7	6	5	4	3	2	1	0
Field	$RESX_7$	$RESX_6$	$RESX_5$	$RESX_4$	$RESX_3$	$RESX_2$	$RESX_1$	$RESX_0$
Description	Lower byte of RESOLUTION_X register. Write to RESOLUTION_X_LOW first, then RESOLUTION_X_HIGH. They should be written consecutively.							

Register Name	RESOLUTION_X_HIGH							
Address	0x49							
Access	Read/Write			Reset Value		0x00		
Bit	7	6	5	4	3	2	1	0
Field	$RESX_{15}$	$RESX_{14}$	$RESX_{13}$	$RESX_{12}$	$RESX_{11}$	$RESX_{10}$	$RESX_{9}$	$RESX_{8}$

Description

This register allows to change the resolution of X-axis of the chip. Write to RESOLUTION_X_LOW first, then RESOLUTION_X_HIGH. They should be written consecutively.

After update the resolution setting, either in RESOLUTION_X or/and RESOLUTION_Y, write value 0x01 into SET_RESOLUTION for the chip to use the new resolution setting.

Note: It is recommended to set bit-7 in RIPPLE_CONTROL register to enable the ripple control when select 9000 CPI and above.

Field Name	Description
RESX [15:0]	Set X-axis Resolution: 0x000: 50CPI 0x001: 100CPI 0x002: 150CPI : 0x063: 5000CPI (Default) : 0x18F: 20000CPI (max)

Register Name	RESOLUTION_Y_LOW							
Address	0x4A							
Access	Read/Write			Reset Value		0x63		
Bit	7	6	5	4	3	2	1	0
Field	$RESY_{7}$	$RESY_{6}$	$RESY_{5}$	$RESY_{4}$	$RESY_{3}$	$RESY_{2}$	$RESY_{1}$	$RESY_{0}$

Description

Lower byte of RESOLUTION_Y register. Write to RESOLUTION_Y_LOW first, then RESOLUTION_Y_HIGH. They should be written consecutively.

Register Name	RESOLUTION_Y_HIGH							
Address	0x4B							
Access	Read/Write			Reset Value		0x00		
Bit	7	6	5	4	3	2	1	0
Field	$RESY_{15}$	$RESY_{14}$	$RESY_{13}$	$RESY_{12}$	$RESY_{11}$	$RESY_{10}$	$RESY_{9}$	$RESY_{8}$

Description

This register allows to change the resolution of Y-axis of the chip. Write to RESOLUTION_Y_LOW first, then RESOLUTION_Y_HIGH. They should be writer consecutively.

After update the resolution setting, either in RESOLUTION_X or/and RESOLUTION_Y, write value 0x01 into SET_RESOLUTION for the chip to use the new resolution setting.

Note: It is recommended to set bit-7 in RIPPLE_CONTROL register to enable the ripple control when select 9000 CPI and above.

Field Name	Description
RESY [15:0]	Set Y-axis Resolution: 0x000: 50CPI 0x001: 100CPI 0x002: 150CPI : 0x063: 5000CPI (Default) : 0x18F: 20000CPI (max)

Register Name	ANGLE_SNAP							
Address	0x56							
Access	Read/Write			Reset Value		0x0D		
Bit	7	6	5	4	3	2	1	0
Field	EN	0	0	0	1	1	0	1

Description

Write to this register to enable angle snap feature.

Field Name	Description
EN	0: Angle snap disable 1: Angle snap enable

Register Name	RAWDATA_GRAB							
Address	0x58							
Access	Read/Write			Reset Value		0x00		
Bit	7	6	5	4	3	2	1	0
Field	$RAWDATA_7$	$RAWDATA_6$	$RAWDATA_5$	$RAWDATA_4$	$RAWDATA_3$	$RAWDATA_2$	$RAWDATA_1$	$RAWDATA_0$
Description	This register contains the rawdata levels when the RawData Grab process is enabled. For details of the RawData Grab process please refer to section 7.1.							

Register Name	RAWDATA_GRAB_STATUS							
Address	0x59							
Access	Read			Reset Value		0x00		
Bit	7	6	5	4	3	2	1	0
Field	PG_VALID	PG_FIRST	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved

Description

This register provides additional information for user to monitor the chip navigation status.

Field Name	Description
PG_VALID	1: RawData Grab valid
PG_FIRST	1: RawData Grab first

Register Name	RIPPLE_CONTROL							
Address	0x5A							
Access	Read/Write			Reset Value		0x00		
Bit	7	6	5	4	3	2	1	0
Field	$CTRL_8$	Reserved	Reserved	1	Reserved	Reserved	Reserved	Reserved

Description

Write to this register to enable or disable Ripple Control feature. Upon chip start-up per the recommended Power-Up sequence in the section 6.1, Ripple Control is disabled as default.

Field Name	Description
$CTRL_8$	0: Ripple Control disable 1: Ripple Control enable

Register Name	AXIS_CONTROL							
Address	0x5B							
Access	Read/Write			Reset Value		0x60		
Bit	7	6	5	4	3	2	1	0
Field	Swap_XY	INV_Y	INV_X	Reserved	Reserved	Reserved	Reserved	Reserved

Description

The register set the axis direction of the chip reporting.

Field Name	Description
Swap_XY	1: Swap XY directions
INV_Y	1: Invert Y direction
INV_X	1: Invert X direction

Register Name	MOTION_CTRL							
Address	0x5C							
Access	Read/Write			Reset Value		0x02		
Bit	7	6	5	4	3	2	1	0
Field	MOT_Set	Reserved	Reserved	Reserved	Reserved	Reserved	RES_MOD	Reserved

Description

Configures the motion pin setting and select the X-axis and Y-axis resolution mode.

Field Name	Description
MOT_Set	0: motion active low (default) 1: motion active high
RES_Mod	0: Both X-axis and Y-axis resolution are defined by RESOLUTION_X_LOW and RESOLUTION_X_HIGH 1: X-axis resolution is defined by RESOLUTION_X_LOW and RESOLUTION_X_HIGH and Y-axis resolution is defined by RESOLUION_Y_LOW and RESOLUION_Y_HIGH (default)

Register Name	INV_PROD_ID							
Address	0x5F							
Access	Read			Reset Value		0xB0		
Bit	7	6	5	4	3	2	1	0
Field	$IPID_{7-0}$							
Description	This register value is the inverse of the Product_ID register value. It is used to test the SPI port hardware.							

Register Name	RUN_DOWNSHIFT							
Address	0x77							
Access	Read/Write			Reset Value		0x14		
Bit	7	6	5	4	3	2	1	0
Field	RD_7	RD_6	RD_5	RD_4	RD_3	RD_2	RD_1	RD_0
Description	This register set the Run to Rest1 downshift time. Use the formula below for calculation. Run Downshift time (ms) = RD[7:0] x RUN_DOWNSHIFT_MULT (default 256) x 50us Default Run Downshift = 20 x 256 x 50us = 256ms Max Downshift time is 256x 256(default) x 50us = 3276ms Min value is 0x01. A value of 0x00 will be internally clipped to 0x01. All the above values are expected to have +/- 10% tolerance.							

Register Name	REST1_PERIOD							
Address	0x78							
Access	Read/Write			Reset Value		0x01		
Bit	7	6	5	4	3	2	1	0
Field	$R1R_7$	$R1R_6$	$R1R_5$	$R1R_4$	$R1R_3$	$R1R_2$	$R1R_1$	$R1R_0$
Description	This register set the Rest1 period Rest1 period = R1P[7:0] x 1ms Default Rest1 period = 1 x 1ms = 1ms Min value is 0x01. A value of 0x00 is invalid. All the above values are expected to have +/- 10% tolerance.							

Register Name	REST1_DOWNSHIFT							
Address	0x79							
Access	Read/Write			Reset Value		0x90		
Bit	7	6	5	4	3	2	1	0
Field	$R1D_7$	$R1D_6$	$R1D_5$	$R1D_4$	$R1D_3$	$R1D_2$	$R1D_1$	$R1D_0$
Description	This register set the Rest1 to Rest2 downshift time. Use the formula below for calculation. Rest1 Downshift time (ms) = R1D[7:0] x REST1_DOWNSHIFT_MULT (default 64) x REST1 period (default 1ms) Default = 144 x 64 x 1ms = 9216ms = 9.2s Min value is 0x01. A value of 0x00 will be internally clipped to 0x01. All the above values are expected to have +/- 10% tolerance.							

Register Name	REST2_PERIOD							
Address	0x7A							
Access	Read/Write			Reset Value		0x19		
Bit	7	6	5	4	3	2	1	0
Field	$R2P_7$	$R2P_6$	$R2P_5$	$R2P_4$	$R2P_3$	$R2P_2$	$R2P_1$	$R2P_0$
Description	This register set the Rest2 period Rest2 period = R2P[7:0] x slow clock (1ms) x 4 Default Rest2 period = 25 x 1ms x 4= 100ms Min value is 0x01. A value of 0x00 is invalid. All the above values are expected to have +/- 10% tolerance.							

Register Name	REST2_DOWNSHIFT							
Address	0x7B							
Access	Read/Write			Reset Value		0x5E		
Bit	7	6	5	4	3	2	1	0
Field	$R2D_7$	$R2D_6$	$R2D_5$	$R2D_4$	$R2D_3$	$R2D_2$	$R2D_1$	$R2D_0$
Description	This register set the Rest2 to Rest3 downshift time. Use the formula below for calculation. Rest2 Downshift time (ms) = R2D[7:0] x REST2_DOWNSHIFT_MULT (default 64) x rest2_period (default 100ms) Default = 94 x 64 x 100ms = 601.6s = 10min Min value is 0x01. A value of 0x00 will be internally clipped to 0x01. All the above values are expected to have +/- 10% tolerance.							

Register Name	REST3_PERIOD							
Address	0x7C							
Access	Read/Write			Reset Value		0x3F		
Bit	7	6	5	4	3	2	1	0
Field	$R3P_7$	$R3P_6$	$R3P_5$	$R3P_4$	$R3P_3$	$R3P_2$	$R3P_1$	$R3P_0$
Description	This register set the Rest3 period Rest3 period = R3P[7:0] x slow clock (1ms) x 8 Default Rest3 period = 63 x 1ms x 8 = 504ms Min value is 0x01. A value of 0x00 is invalid. All the above values are expected to have +/- 10% tolerance.							

Register Name	RUN_DOWNSHIFT_MULT							
Address	0x7D							
Access	Read/Write			Reset Value		0x07		
Bit	7	6	5	4	3	2	1	0
Field	Reserved	Reserved	Reserved	Reserved	RUN_M_3	RUN_M_2	RUN_M_1	RUN_M_0

Description

This register set the Run Downshift Multiplier. (Refer to the formula in Register RUN_DOWNSHIFT)

Field Name	Description	
$RUN\_M_{0:3}$	Hex	RUN_DOWNSHIFT_MULT
	0x0	2
	0x1	4
	0x2	8
	0x3	16
	0x4	32
	0x5	64
	0x6	128
	0x7	256 (default)
	0x8	512
	0x9	1024
	0xA	2048

Register Name	REST_DOWNSHIFT_MULT							
Address	0x7E							
Access	Read/Write			Reset Value		0x55		
Bit	7	6	5	4	3	2	1	0
Field	Reserved	REST_M$_6$	REST_M$_5$	REST_M$_4$	Reserved	REST_M$_2$	REST_M$_1$	REST_M$_0$

Description

This register set the REST Downshift Multiplier.
(Refer to the formula in Register REST1_DOWNSHIFT and REST2_DOWNSHIFT)

Field Name	Description	
REST_M$_{0:2}$	Hex	REST1_DOWNSHIFT_MULT
	0x0	2
	0x1	4
	0x2	8
	0x3	16
	0x4	32
	0x5	64 (default)
	0x6	128
	0x7	256
REST_M$_{4:6}$	Hex	REST2_DOWNSHIFT_MULT
	0x0	2
	0x1	4
	0x2	8
	0x3	16
	0x4	32
	0x5	64 (default)
	0x6	128
	0x7	256

Register Name	ANGLE_TUNE1							
Address	0x0577							
Access	Read/Write			Reset Value		0x00		
Bit	7	6	5	4	3	2	1	0
Field	$ANGLE_7$	$ANGLE_6$	$ANGLE_5$	$ANGLE_4$	$ANGLE_3$	$ANGLE_2$	$ANGLE_1$	$ANGLE_0$

Description

Field Name	Description
ANGLE[7:0]	0xE2 -30 degree 0xF6 -10 degree **0x00 0 degree (default)** 0x0F +15 degree 0x1E +30 degree

Register Name	ANGLE_TUNE2							
Address	0x0578							
Access	Read/Write			Reset Value		0x00		
Bit	7	6	5	4	3	2	1	0
Field	EN	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved

Description

Write to this register to enable angle tune feature.

Field Name	Description
EN	**0: Angle tune disable (default)** 1: Angle tune enable

Register Name	LIFT_CONFIG							
Address	0x0C4E							
Access	Read/Write			Reset Value		0x08		
Bit	7	6	5	4	3	2	1	0
Field	Reserved	Reserved	Reserved	Reserved	1	Reserved	$LIFT_1$	$LIFT_0$

Description

This register configures the lift setting.

Field Name	Description	
LIFT[1:0]	**Hex**	**Lift Setting**
	0x0	1mm (default)
	0x1	2mm
	0x2	3mm

8.3 Bit Masks for Register Write

Special precaution needs to be taken for some of the registers have “Reserved” bit. In order to overwrite specific bits in the register, one need to read and store its current value first, then apply bit masking and write back the new value into the register. This is accomplished by using bitwise operators such as AND(&), OR(|), or INVERSE(~).

Example:

To disable the Rest Mode in Register 0x40 (set bit-7 to 1)

Read register 0x40 and store in VarA

VarA |= 0x80

Write register 0x40 with value VarA

To enable the Rest Mode in Register 0x40 (set bit-7 to 0)

Read register 0x40 and store in VarA

VarA &= ~ 0x80

Write register 0x40 with value VarA

Document Revision History

Revision Number	Date	Description
0.80	28 Sept 2020	Initial Creation
1.00	7 Dec 2020	Added Corded Gaming Mode