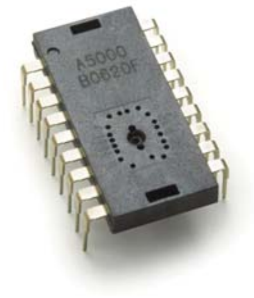
ADNS-5000
Optical Mouse Sensor

AVAGO TECHNOLOGIES

Data Sheet

Lead (Pb) Free RoHS 6 fully compliant



Description

The ADNS-5000 is a one-chip USB optical mouse sensor for implementing a non-mechanical tracking engine for computer mice.

It is based on optical navigation technology that measures changes in position by optically acquiring sequential surface images (frames) and mathematically determining the direction and magnitude of movement.

The sensor is in a 18-pin optical package that is designed to be used with the ADNS-5100 Round Lens or ADNS-5100-001 Trim Lens, the ADNS-5200 Clip, and the HLMP-ED80-XX000 LED. These parts provide a complete and compact mouse sensor. There are no moving parts, and precision optical alignment is not required, facilitating high volume assembly.

The output format is USB. This device meets USB revision 1.1 specifications and is compatible with USB Revision 2.0 specification.

Default resolution is specified as 500 counts per inch, with rates of motion up to 16 inches per second and 2g acceleration. Resolution can also be programmed to 1000 cpi. Frame rate is varied internally by the sensor to achieve tracking and speed performance, eliminating the need for the use of many registers.

A complete mouse can be built with the addition of a PC board, switches, mechanical Z-wheel, plastic case and cable. A 1% pull up resistor is needed for the USB port to signify a low speed HID device.

Features

- Optical navigation technology
- No mechanical moving parts
- High reliability
- Complete 2-D motion sensor
- High speed motion detection
- Accurate navigation over a wide variety of surfaces
- No precision optical alignment needed
- Wave Solderable
- IEC 60825-1 eye safe under single fault conditions
- Single 5.0 volt power supply
- Meets USB Revision 1.1 Specification and compatible with USB Revision 2.0 specification
- Meets HID Revision 1.1
- On Chip LED Drive with regulated current

Applications

- Mice for desktop PC's, Workstations, and portable PC's
- Trackballs
- Integrated input devices

Theory of Operation

The ADNS-5000 is based on Optical Navigation Technology. It contains an Image Acquisition System (IAS), a Digital Signal Processor (DSP) and USB stream output.

The IAS acquires microscopic surface images via the lens and illumination system provided by the ADNS-5100 Round Lens or ADNS-5100-001 Trim Lens, ADNS-5200, and HLMP-ED80-XX000. These images are processed by the DSP to determine the direction and distance of motion. The DSP generates the Δx and Δy relative displacement values which are converted to USB motion data.

Pinout

Pin	Pin	Description
1	D +	USB D+ line
2	D -	USB D- line
3	ZA	Scroll wheel quadrature input
4	ZB	Scroll wheel quadrature input
5	LGND	LED ground
6	XYLED	XYLED Input
7	VDD5	5 Volt Power (USB VBUS)
8	GND	System ground
9	REG0	3 Volt Power
10	VDD3	3 Volt Power
11	OPT 0	Descriptor Select 1 or B4
12	OPT 1	Descriptor Select 2 or B5
13	GND	System ground
14	OSC_IN	Ceramic resonator input
15	OSC_OUT	Ceramic resonator output
16	B3	Button 3 input (switch to ground)
17	B2	Button 2 input (switch to ground)
18	B1	Button 1 input (switch to ground)

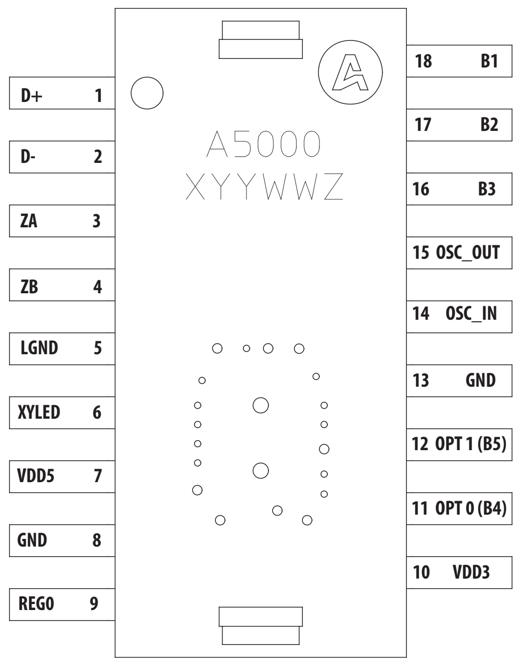
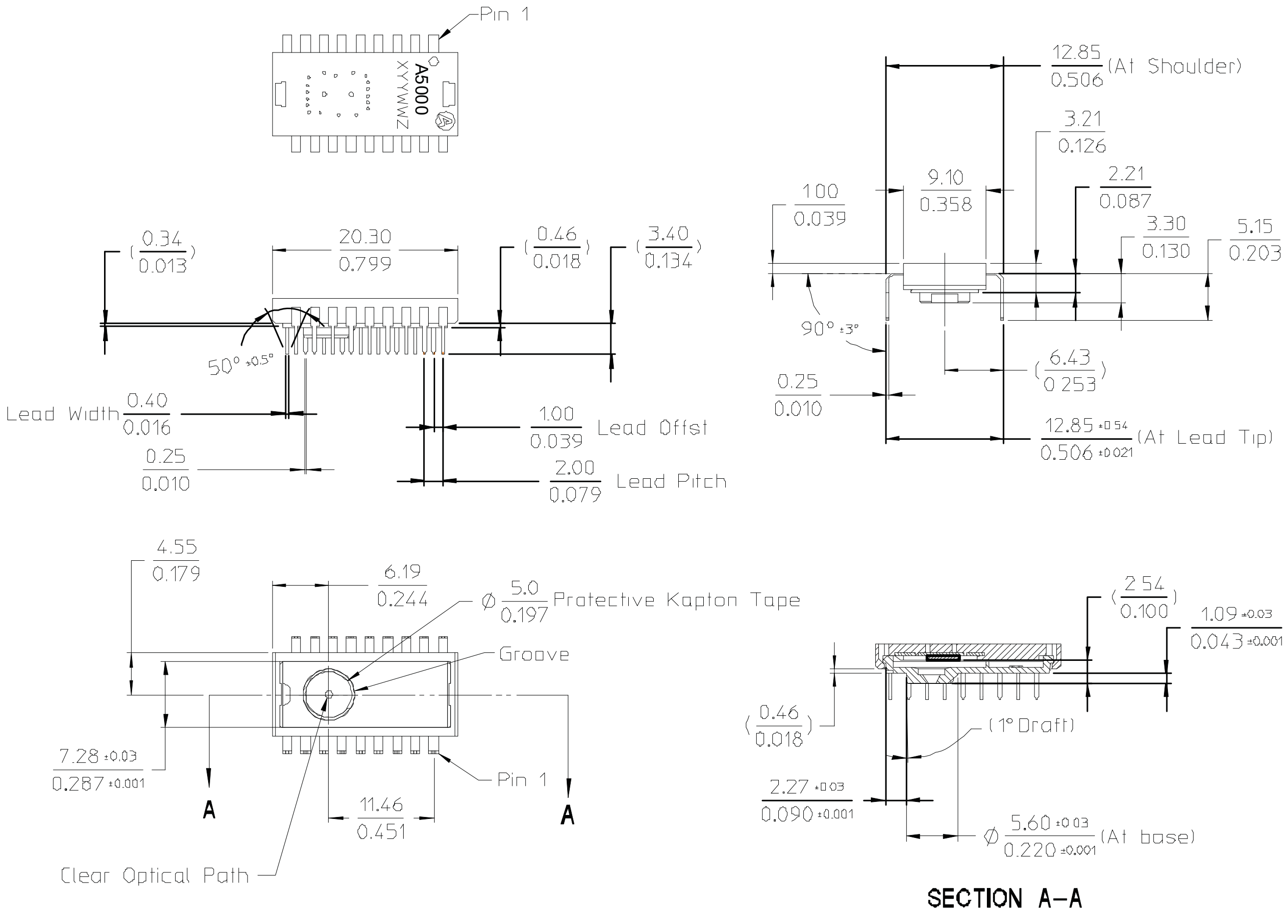


Figure 1. Package outline drawing (top view)



Notes.
1. Dimensions in milimeters / inches.
2. Dimensional tolerance. ±0.1mm.
3. Coplanarity of Leads: ±0.1mm.
4. Lead pitch tolerance: ±0.15mm.
5. Non-cumulative pitch tolerance: ±0.15mm.
6. Angular tolerance. ±3.0°.
7. Maximum flash. +0.2mm.
8. Chamfer (25° X 2) on the taper side of lead.
9. () Bracket dimensions are for reference only and should not be used to mechanically reference the sensor.

Figure 2. Package outline drawing

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

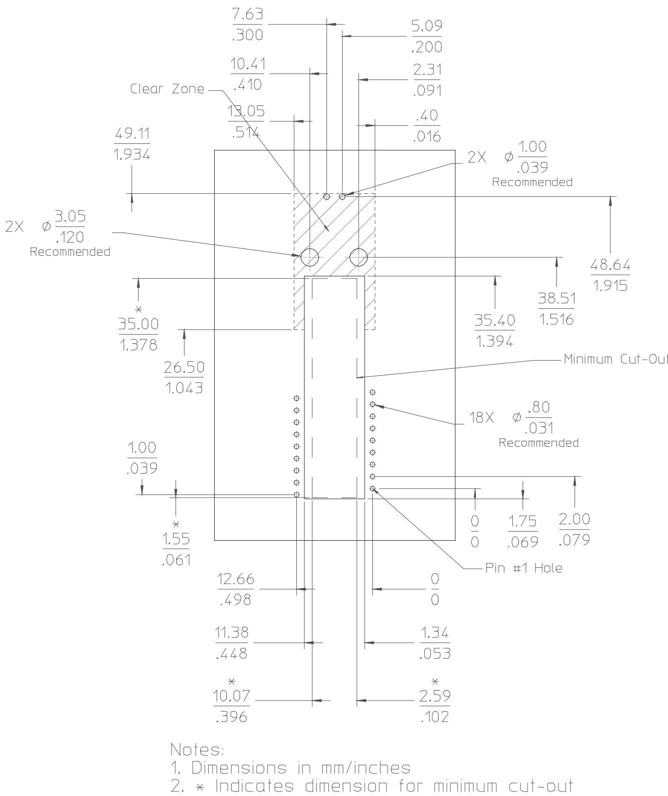


Figure 3. Recommended PCB mechanical cutouts and spacing (Top view)

Note: The recommended pin hole dimension of the sensor is 0.7 mm.

Shown with ADNS-5100, ADNS-5200 and HLMP-ED80-XX000

Avago Technologies provides an IGES file drawing describing the base plate molding features for lens and PCB alignment.

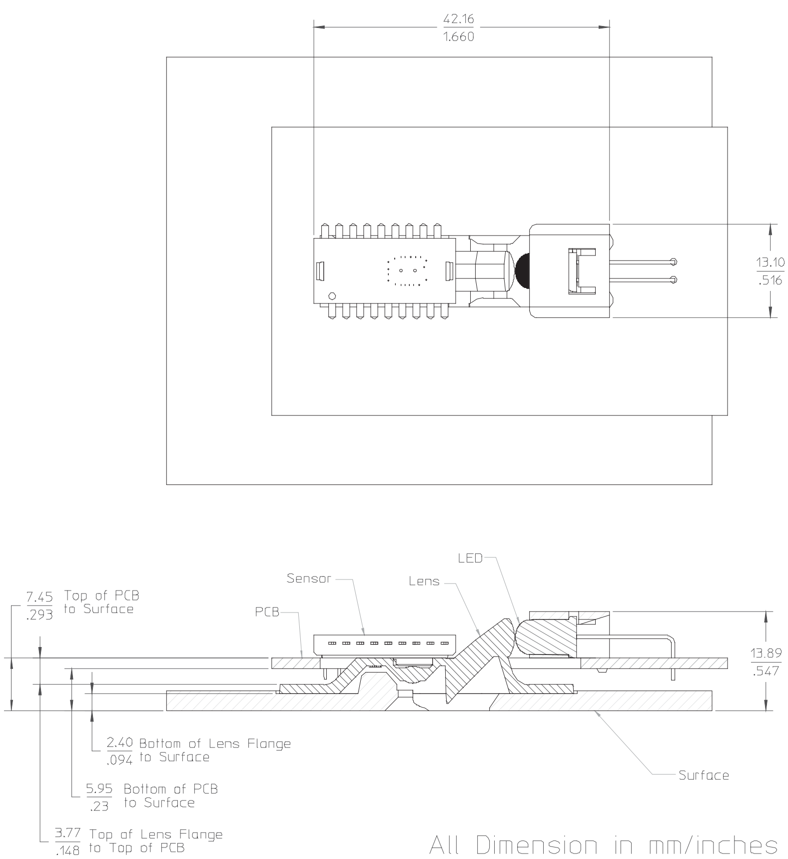


Figure 4. 2D assembly drawing of ADNS-5000

The components interlock as they are mounted onto defined features on the base plate.

The ADNS-5000 sensor is designed for mounting on a through hole PCB, looking down. The aperture stop and features on the package align it to the lens (See figure 3).

The ADNS-5100 Round lens provides optics for the imaging of the surface as well as illumination of the surface at the optimum angle. Lens features align it to the sensor, base plate, and clip with the LED. The lens also has a large round flange to provide a long creepage path for any ESD events that occur at the opening of the base plate (See figure 4).

The ADNS-5200 clip holds the LED in relation to the lens. The LED must be inserted into the clip and the LED's leads formed prior to loading on the PCB.

The HLMP-ED80-XX000 LED is recommended for illumination. If used with the bin table, sufficient illumination can be guaranteed.

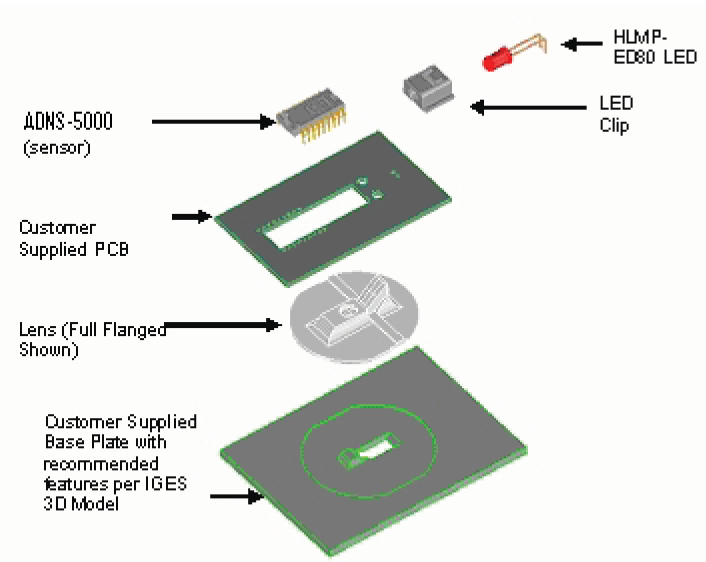


Figure 5. Exploded view drawing

Block Diagram

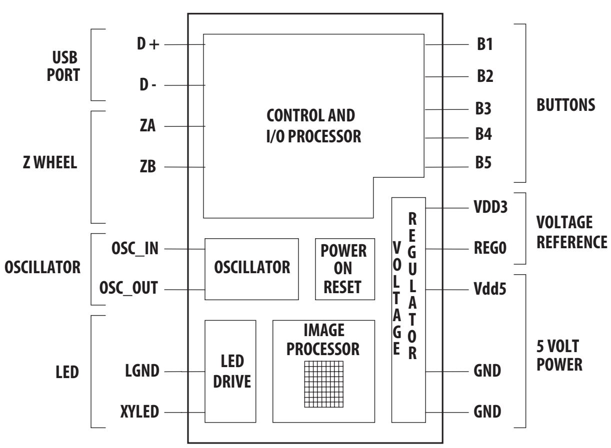


Figure 6. Block Diagram

PCB Assembly Considerations

1. Insert the sensor and all other electrical components into PCB.
2. Bend the LED leads 90 degrees and then insert the Led into the assembly clip until the snap feature locks the Led base.
3. Insert the LED/clip assembly into PCB.
4. Wave solder the entire assembly in a no-wash solder process utilizing solder fixture. The solder fixture is needed to protect the sensor during the solder process. The fixture should be designed to expose the sensor leads to solder while shielding the optical aperture from direct solder contact.
5. Place the lens onto the base plate.
6. Remove the protective Kapton tape from optical aperture of the sensor. Care must be taken to keep contaminants from entering the aperture. Recommend not placing the PCB facing up during the entire mouse assembly process. Recommend to hold the PCB first vertically for the Kapton removal process.
7. Insert PCB assembly over the lens onto base plate aligning post to retain PCB assembly. The sensor aperture ring should self-align to the lens.
8. 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.
9. Install mouse top case.

Design considerations for improving ESD Performance

The table below shows typical values assuming base plate construction per the Avago Technologies supplied IGES file and ADNS-5100 Round lens.

Typical distance	A5100	A5100-001
Creepage	40.5mm	17.9mm
Clearance	32.6mm	9.2mm

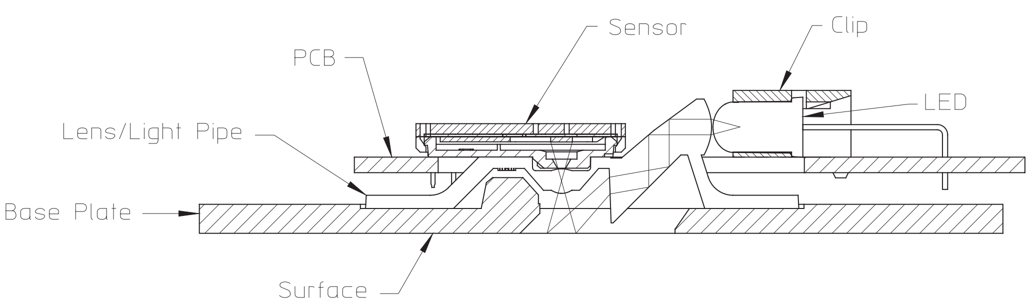


Figure 7. Typical Application

Typical Application

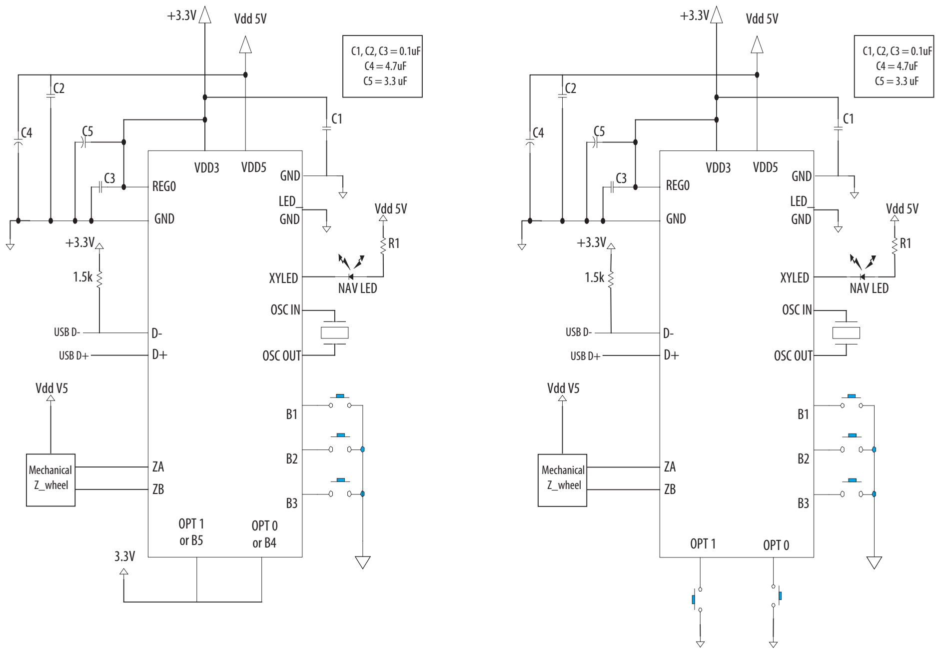


Figure 8. Application Schematic for 3 buttons and 5 buttons

R1 value (ohm)	LED Bin
59.0	K
59.0	L
59.0	M
59.0	N
59.0 to 66.5	P
59.0 to 78.7	Q
59.0 to 93.1	R
59.0 to 110	S
59.0 to 143	T

Notes on bypass capacitors:

- All caps (except C4) MUST be as close to the sensor pins as possible.
- Caps should be ceramic.
- Caps should have less than 5 nH of self inductance
- Caps connected to VDD3 MUST have less than 0.2Ω ESR
- 1.5kΩ resistor should be ± 1% tolerance.
- Z-wheel connections are detailed in Figure 20
- Buttons B1-B5 can be used as button or VID/PID straps (see strap table on page 14). For VID/PID connections, parts must be connected to Vdd3 on 'high' connection, preferably near pin 10

Surface mount parts are recommended

Regulatory Requirements

- Passes FCC B and worldwide analogous emission limits when assembled into a mouse with unshielded cable and following Avago Technologies recommendations.
- Passes EN61000-4-4/IEC801-4 EFT tests when assembled into a mouse with shielded cable and following Avago Technologies recommendations.
- UL flammability level UL94 V-0.
- Provides sufficient ESD creepage/clearance distance to avoid discharge up to 15kV when assembled into a mouse according to usage instructions above.

Absolute Maximum Ratings

Parameter	Symbol	Minimum	Maximum	Units	Notes
Storage Temperature	T_S	-40	85	°C	
Operating Temperature	T_A	-15	55	°C	
Lead Solder Temp			260	°C	For 10 seconds, 1.6mm below seating plane.
Supply Voltage	V_{DD}	-0.5	5.5	V	
ESD			2	kV	All pins, human body model MIL 883 Method 3015
Input Voltage	V_{IN}	-0.5	V_{DD}+0.5	V	All I/O pins except OSC_IN and OSC_OUT, D+, D-
Input Voltage	V_{IN}	-1.0	4.6	V	D+, D-, AC waveform, see USB specification (7.1.1)
Input Voltage	V_{IN}	-0.5	3.6	V	OSC_IN and OSC_OUT
Input Short Circuit Voltage	V_{SC}	0	V_{DD}	V	D+, D-, see USB specification (7.1.1)

Recommended Operating Conditions

Parameter	Symbol	Minimum	Typical	Maximum	Units	Notes
Operating Temperature	T_A	0		40	°C	
Power supply voltage	V_{DD}	4.0	5.0	5.25	Volts	For accurate navigation and proper USB operation
Power supply voltage	V_{dd}	3.8	5.0	5.25	Volts	Maintains communication to USB host and internal register contents.
Power supply rise time	V_{RT}	0.1		100	ms	
Supply noise	V_N			100	mV	Peak to peak within 0-100 MHz bandwidth
Velocity	Vel			16	ips	
Acceleration	Acc			2	G	
Clock Frequency	f_{clk}	23.64	24	24.36	MHz	Due to USB timing constraints
Resonator Impedance	X_{RES}			55	Ω	
Distance from lens reference plane to surface	Z	2.3	2.4	2.5	mm	See Figure 9
Light Level onto IC	IRR_{INC}	80 100		25,000 30,000	mW/m^2	=639nm =875nm

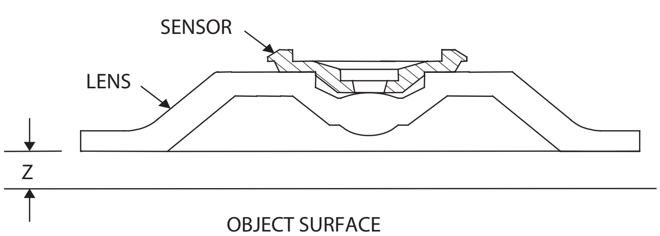


Figure 9. Distance from lens reference plane to object surface

AC Electrical Specifications

Electrical Characteristics over recommended operating conditions. Typical values at 25 °C, V_{DD}=5.0 V, 24MHz

Parameter	Symbol	Min.	Typ.	Max.	Units	Notes
Power up delay	T_{PUP}			50	ms	
Debounce delay on button inputs	T_{DBB}	5	9	17	ms	"Maximum" specified at 8ms polling rate.
Mechanical Z-Wheel						Internally pulled down with 20k resistors and debounced
Transient Supply Current	I_{DDT}			60	mA	Max. supply current during a VDD ramp from 0 to 5.0 V with > 500 s rise time. Does not include charging currents for bypass capacitors.
Input Capacitance (OSC Pins)	C_{OSC_IN}		50		pF	OCS_IN, OSC_OUT to GND

USB Electrical Specifications

Electrical Characteristics over recommended operating conditions.

Parameter	Symbol	Min.	Max.	Units	Notes
Output Signal Crossover Voltage	V_{CRS}	1.3	2.0	V	C_L = 200 to 600 pF (see Figure 10)
Input Signal Crossover Voltage	V_{ICRS}	1.2	2.1	V	C_L = 200 to 600 pF (see Figure 10)
Output High	V_{OH}	2.8	3.6	V	with 15 kohm to Ground and 7.5 k to Vbus on D- (see Figure 11)
Output Low	V_{OL}	0.0	0.3	V	with 15 kohm to Ground and 7.5 k to Vbus on D- (see Figure 11)
Single Ended Output	V_{SE0}		0.8	V	
Input High (Driven)	VI_H	2.0		V	
Input High (Floating)	V_{IHZ}	2.7	3.6	V	
Input Low	V_{IL}		0.8	V	7.5k to Vdd5
Differential Input Sensitivity	V_{DI}	0.2		V	|(D+)-(D-)| See Figure 12
Differential Input Common Mode Range	V_{CM}	0.8	2.5	V	Includes V_{DI}, See Figure 12
Single Ended Receiver Threshold	V_{SE}	0.8	2.0	V	
Transceiver Input Capacitance	C_{IN}		12	pF	D+ to V_{BUS}, D- to V_{BUS}

USB Timing Specifications

Timing Specifications over recommended operating conditions.

Parameter	Symbol	Min.	Max.	Units	Notes
D+/D- Transition rise time	T_{LR}	75		ns	C_L = 200 pF (10% to 90%), see Figure 10
D+/D- Transition rise time	T_{LR}		300	ns	C_L = 600 pF (10% to 90%), see Figure 10
D+/D- Transition fall time	T_{LF}	75		ns	C_L = 200 pF (90% to 10%), see Figure 10
D+/D- Transition fall time	T_{LF}		300	ns	C_L = 600 pF (90% to 10%), see Figure 10
Rise and Fall time matching	T_{LRFM}	80	125	%	T_R/T_F; C_L = 200 pF; Excluding the first transition from the Idle State
Wakeup delay from USB suspend mode due to buttons push	T_{WUPB}		17	ms	Delay from button push to USB operation Only required if remote wakeup enabled
Wakeup delay from USB suspend mode due to buttons push until accurate navigation	T_{WUPN}		50	ms	Delay from button push to navigation operation Only required if remote wakeup enabled
USB reset time	T_{reset}	18.7		s	
Data Rate	t_{LDRATE}	1.4775	1.5225	Mb/s	Average bit rate, 1.5 Mb/s +/- 1.5%
Receiver Jitter Tolerance	t_{DJR1}	-75	75	ns	To next transition, see Figure 13
Receiver Jitter Tolerance	t_{DJR2}	-45	45	ns	For paired transitions, see Figure 13
Differential to EOP Transition Skew	t_{LDEOP}	-40	100	ns	See Figure 14
EOP Width at Receiver	t_{LEOPR}	670		ns	Accepts EOP, see Figure 14
Source EOP Width	t_{LEOPT}	1.25	1.50	s	
Width of SE0 interval during Differential Transition	t_{LST}		210	ns	See Figure 11.
Differential Output Jitter	t_{UDJ1}	-95	95	ns	To next transition, see Figure 15
Differential Output Jitter	t_{UDJ2}	-150	150	ns	For paired transitions, see Figure 15

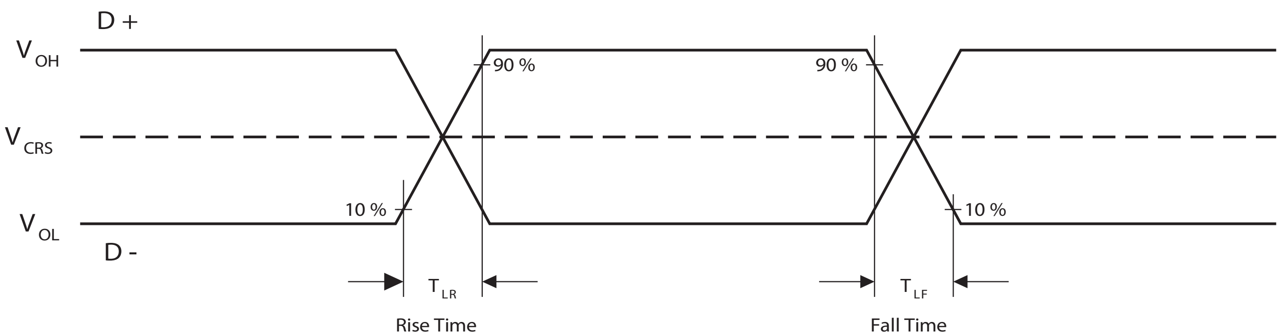


Figure 10. Data Signal Rise and Fall Times

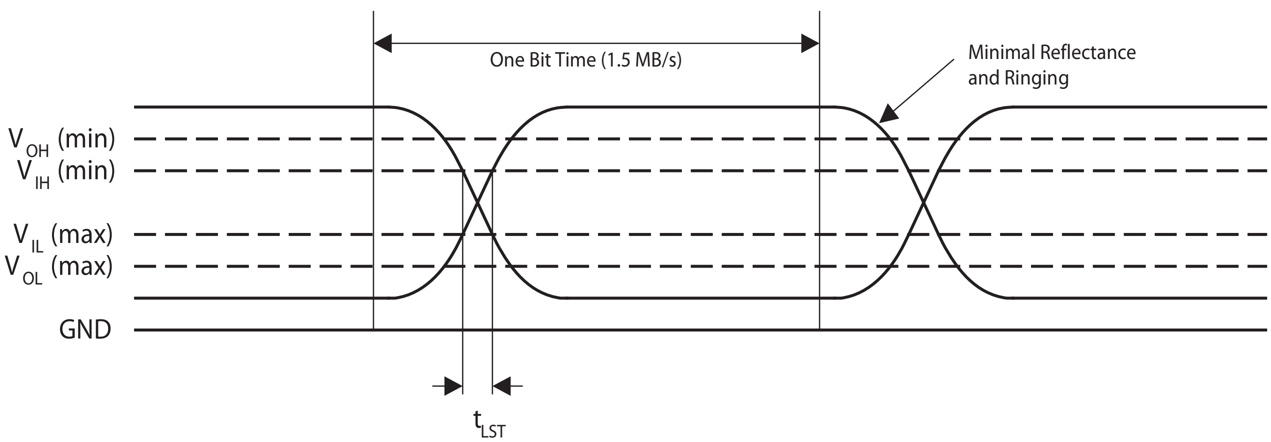


Figure 11. Data Signal Voltage Levels

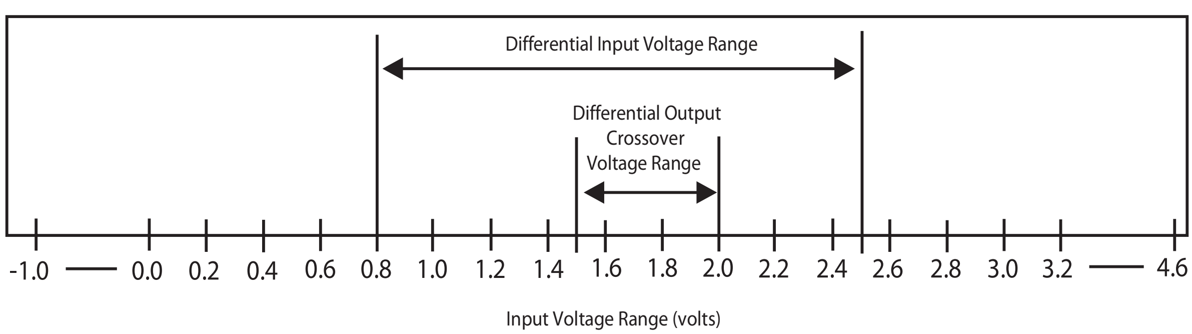


Figure 12. Differential Receiver Input Sensitivity vs. Common Mode Input Range

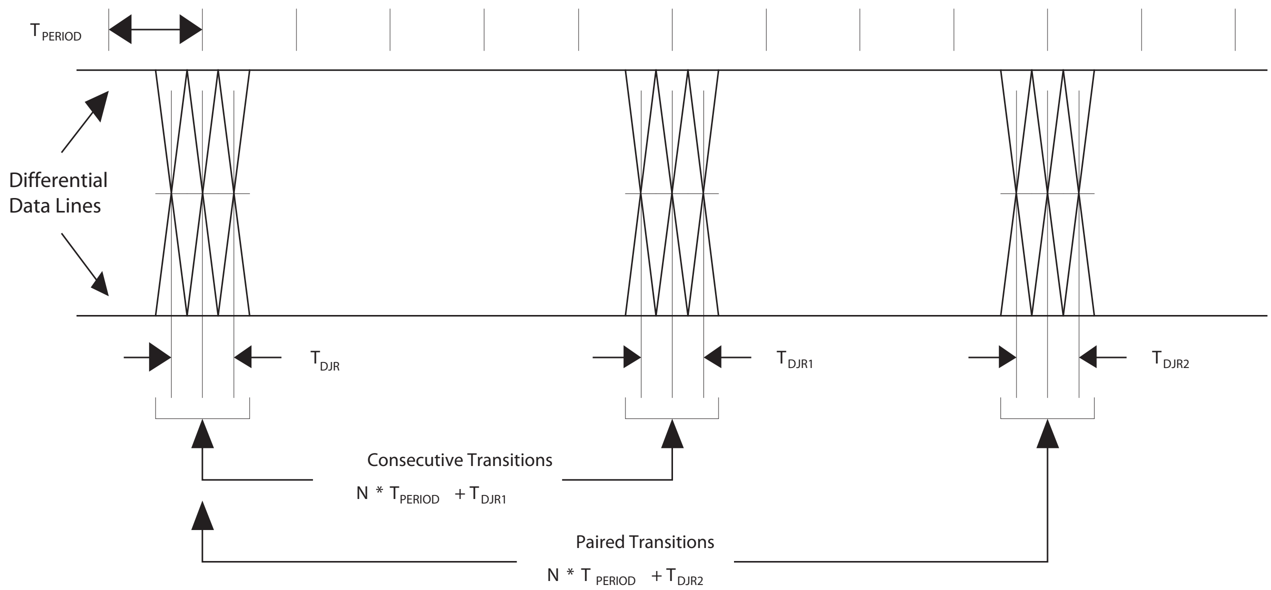


Figure 13. Receiver Jitter Tolerance

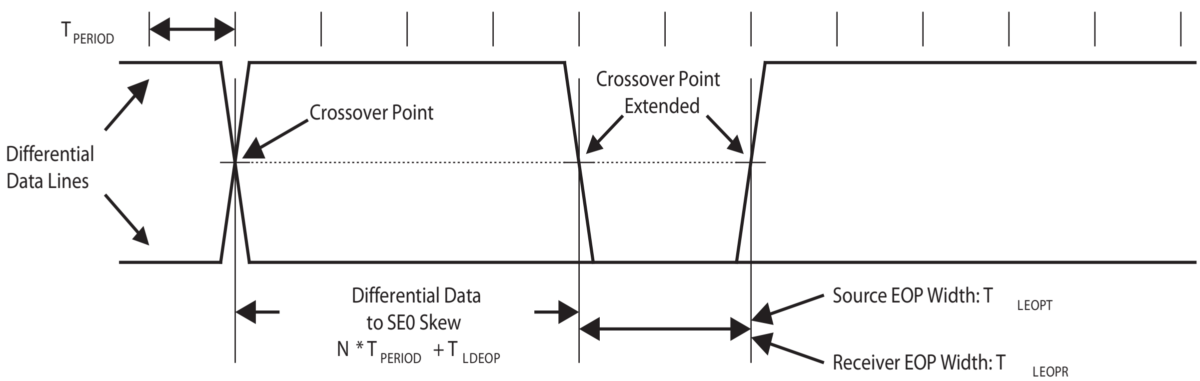


Figure 14. Differential to EOP Transition Skew and EOP Width

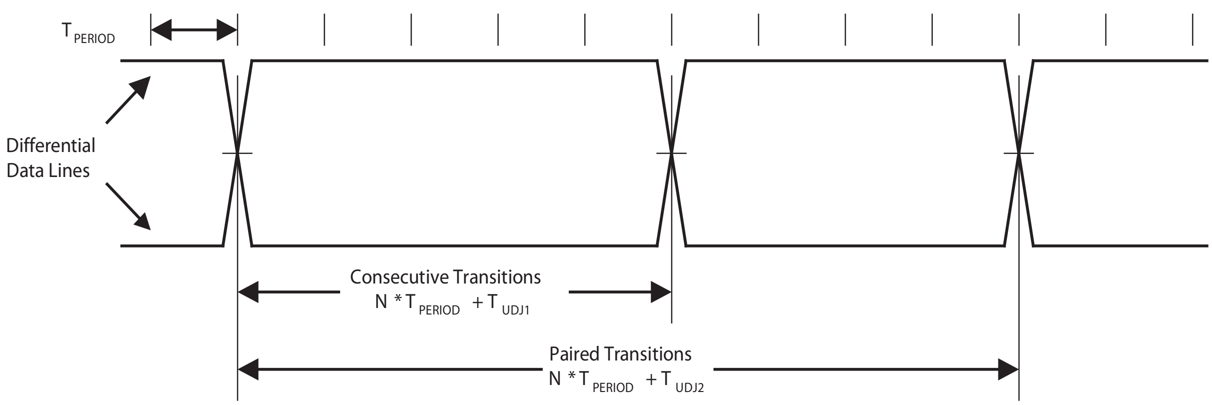


Figure 15. Differential Output Jitter

DC Electrical Specifications

Electrical Characteristics over recommended operating conditions. Typical values at 25 °C, V_{DD}=5.0 V, 24MHz

Parameter	Symbol	Minimum	Typical	Maximum	Units	Notes
Supply current (Sensor only), mouse moving	I_{DDS}		7.2		mA	No load on B1-B3, Z-LED, XYLED ZA, ZB, D+, D-
Supply current (Sensor only), mouse not moving	I_{DDSN}		6.2		mA	No load on B1-B3, Z-LED, XYLED ZA, ZB, D+, D-
Supply current, USB suspend mode	I_{DDSS}			250	µA	No load on B1-B3, Z-LED, XYLED ZA, ZB, D+, D-
XYLED current	I_{LED}			30	mA	
XYLED Output Low Voltage	V_{OL}			1.1	V	Refer to Figure 16
Input Low Voltage	V_{IL}			0.5	V	Pins: ZA, ZB, B1, B2, B3, V_{IL} max of $0.5V_{DC}$ is at V_{DD} min of $4V_{DC}$, with a typical of $0.8V_{DC}$ at V_{DD} of $5V_{DC}$
Input High Voltage	V_{IH}	0.6*V_{DD}			V	Pins: ZA, ZB, B1, B2, B3
Input Hysteresis	V_{HYST}		285		mV	Pins: ZA, B1, B2,
Input Hysteresis	V_{HYST}		200		mV	Pins: ZB
Button Pull Up Current	B_{IOUT}	125	275	500	A	Pins: B1, B2, B3

Typical Performance Characteristics

Performance Characteristics over recommended operating conditions. Typical values at 25 °C, V_{DD}=5.0 V, 24MHz

Parameter	Symbol	Minimum	Typical	Maximum	Units	Notes
Path Error (Deviation)	P_{Error}		0.5		%	Average path error as percent of total 2.5" travel on various standard surfaces

Typical Performance Characteristics

Performance Characteristics over recommended operating conditions. Typical values at 25 °C, V_{DD}=5.0 V, 24MHz

The following graphs are the typical performance of the ADNS-5000 sensor, assembled as shown in the 2D assembly drawing with the ADNS-5100 Round Lens/Prism, the ADNS-5200 clip, and the HLMP-ED80-XX000 LED.

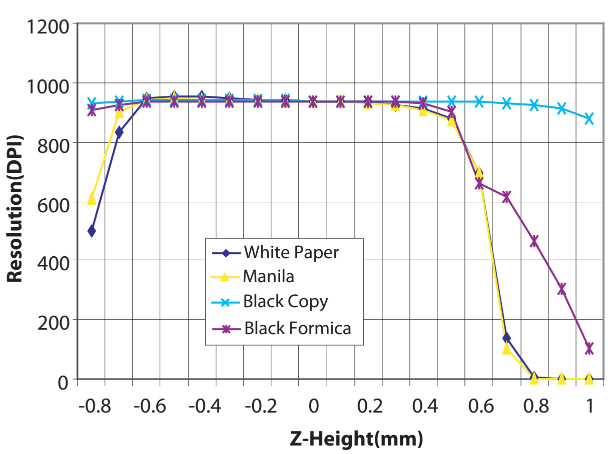


Figure 16. Typical Resolution vs. Z [2,3]

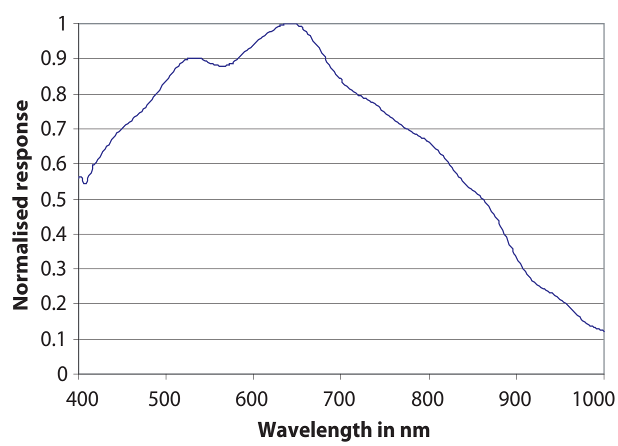


Figure 17. Wavelength responsivity.[1] (Comparative Surfaces)

Notes:
1. The ADNS-5000 is designed for optimal performance when used with the HLMP-ED80-XX000 (Red LED 639nm).
2. Z = distance from Lens Reference Plane to Surface.
3. DOF = Depth of Field

Configuration after Power up (Data Values)

Signal Function	State from Figure 9-1 of USB spec: Powered or Default Address or Configured	State from Figure 9-1 of USB spec: Suspended from any other state
B1	Hi-Z if tied to VDD3 else pullup active	Hi-Z if tied to VDD3 else pullup active
B2	Hi-Z if tied to VDD3 else pullup active	Hi-Z if tied to VDD3 else pullup active
B3	Hi-Z if tied to VDD3 else pullup active	Hi-Z if tied to VDD3 else pullup active
B4	Hi-Z if tied to VDD3 else pullup active	Hi-Z if tied to VDD3 else pullup active
B5	Hi-Z if tied to VDD3 else pullup active	Hi-Z if tied to VDD3 else pullup active
D-	USB I/O	Hi-Z input
D+	USB I/O	Hi-Z input
OSC_IN	24MHz	pulled low
OSC_OUT	24MHz	
XYLED	low (on) or pulsing	Pulled high (off)
ZB/Z_LED	Hi-Z input	Hi-Z input
ZA	Hi-Z if ZA tied to GND	Hi-Z input

Strap (Jumper) Table

The PID/string strap matrix is the following:

Mouse type	VID	PID	Manuf str.	Product string	B1	B2	B3	OPT 0	OPT 1	ZA	ZB
3-button mse	0x192F	0x0116	“”	“USB Optical Mouse”	sw1	sw2	sw3	Vdd3	Vdd3	mech Z-wheel	mech Z-wheel
5-button mse	0x192F	0x0216	“”	“USB Optical Mouse”	sw1	sw2	sw3	sw4	sw5	mech Z-wheel	mech Z-wheel

X & Y Directions

(Looking through an ADNS-5100 Lens)

The positive and negative X and Y directions with respect to the mouse case are shown in the diagram below.

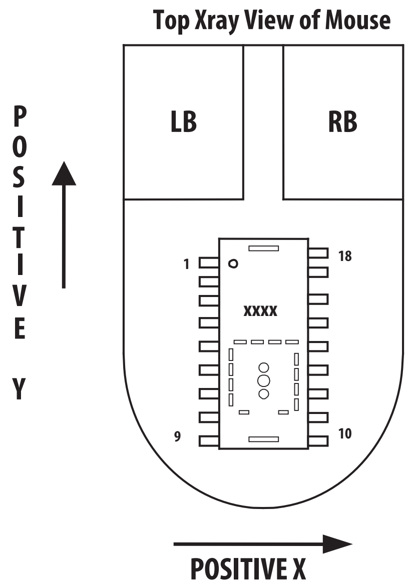


Figure 18. Directions are for a complete mouse, with the ADNS-5100 lens

XY LED

- The peak current values are 30 mA if R1 59ohm and the part meets the IEC 825-1 eye safety regulations.

Buttons

The minimum time between button presses is T_{DBB}. Buttons B1 through B3 are connected to a Schmidt trigger input with 100 uA current sources pulling up to +5 volts during normal, sleep and USB suspend modes.

Z-Wheel

The mechanical Z-Wheel connections (A,B) are determined below.

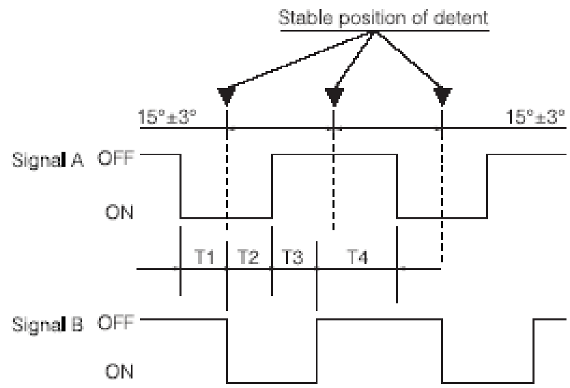


Figure 19. Z-Wheel A and B connections

Notes:

For mechanical Z-wheels the following must be implemented:

- Use a rotary switch equivalent to the Panasonic part EVQVX at http://industrial.panasonic.com/www-data/pdf/ATC0000/ATC0000CE20.pdf (The key point is stable "A" switch state in all detent positions).
- Solder the rotary switch into the PCB such that the common pin is closest to the cable end of the mouse. (Metal plate faces to left)
- Connect the "A" terminal of the rotary switch to "ZA" and the "B" terminal to "ZB". ZA MUST be connected to "Signal A" in Figure 19 where the z-wheel detents are mechanically stable.

USB Commands

Mnemonic	Command	Notes
USB_RESET	D+/D- low > 18.6 us	Device Resets; Address=0
USB_SUSPEND	Idle state > 3mS	Device enters USB low-power mode
USB_RESUME	Non-idle state	Device exits USB low-power mode
Get_Status_Device	80 00 00 00 00 00 02 00	Normally returns 00 00, Self powered 00 00, Remote wakeup 02 00
Get_Status_Interface	81 00 00 00 00 00 02 00	Normally returns 00 00
Get_Status_Endpt0	82 00 00 00 xx 00 02 00	OUT: xx=00, IN: xx=80Normally returns 00 00
Get_Status_Endpt1	82 00 00 00 81 00 02 00	Normally returns 00 00, Halt 00 01
Get_Configuration	80 08 00 00 00 00 01 00	Return: 00=not config., 01=configured
Get_Interface	81 0A 00 00 00 00 01 00	Normally returns 00
Get_Protocol	A1 03 00 00 00 00 01 00	Normally returns 01, Boot protocol 00
Get_Desc_Device	80 06 00 01 00 00 nn 00	See USB command details
Get_Desc_Config	80 06 00 02 00 00 nn 00	See USB command details
Get_Desc_String	80 06 xx 03 00 00 nn 00	See USB command details
Get_Desc_HID	81 06 00 21 00 00 09 00	See USB command details
Get_Desc_HID_Report	81 06 00 22 00 00 nn 00	See USB command details
Get_HID_Input	A1 01 00 01 00 00 nn 00	Return depends on motion & config
Get_Idle	A1 02 00 00 00 00 01 00	Returns rate in multiples of 4ms
Get_Vendor_Test	C0 01 00 00 xx 00 01 00	Read register xx
Set_Address	00 05 xx 00 00 00 00 00	xx = address
Set_Configuration	00 09 xx 00 00 00 00 00	Not configured: xx=00Configured: xx=01
Set_Interface	01 0B 00 00 00 00 00 00	Only one interface supported
Set_Protocol	21 0B xx 00 00 00 00 00	Boot: xx=00, Report: xx=01
Set_Feature_Device	00 03 01 00 00 00 00 00	Enable remote wakeup
Set_Feature_Endpt0	02 03 00 00 xx 00 00 00	Halt. OUT: xx=00, IN: xx=80
Set_Feature_Endpt1	02 03 00 00 81 00 00 00	Halt
Clear_Feature_Device	00 01 01 00 00 00 00 00	Disable Remote wakeup
Clear_Feature_Endpt0	02 01 00 00 xx 00 00 00	Clear Halt; OUT: xx=00, IN: xx=80
Clear_Feature_Endpt1	02 01 00 00 81 00 00 00	Clear Halt
Set_Idle	21 0A 00 rr 00 00 00 00	rr = report rate in multiples of 4ms
Set_Vendor_Test	40 01 00 00 xx yy 00 00	Write yy to address xx
Poll_Endpt1		Read buttons, motion, & Z-wheel

Note:
The last two bytes in a command shown as "nn 00" specify the 16-bit data size in the order of "LowByte HighByte." For example a two-byte data size would be specifed as "02 00." ADNS-5000 will not provide more bytes than the number requested in the command, but it will only supply up to a maximum of 8 bytes at a time. The ADNS-5000 will re-send the last packet if the transfer is not acknowledged properly.

USB COMMAND DETAILS

USB_RESET		D+/D- low for an extended period	
	USB Spec:		A device may reset after seeing an SE0 for more than 18.6 uS, and definitely after 10mS.
	Notes:		After power up and prior to Reset, the device will not respond to any USB commands. After the device has been given a USB Reset, the device's address will be reset to zero and the device will be in the Default state. The chip will default to Report protocol and any pending output will be flushed.

USB_SUSPEND			Idle state for an extended period
	USB Spec:		A device may suspend after seeing an idle for more than 3mS, and definitely after 10mS.
	Notes:		The chip will take a minimum of 5mS to start Suspend, though will definitely start after 6mS. The chip may finish the current frame if necessary before stopping the clock. Thus, an additional frame time may be used to reach Suspend mode.

USB_RESUME		Non-idle state	
	USB Spec:		Remote Resume signalling from a device must be between 1mS and 15mS. The host is required to send Resume signaling for 20mS plus 10mS of resume recovery time in which it does not access any devices. This allows devices enough time to wake back up.
	Notes:		The chip can cause a Resume if Remote Wakeup is enabled and a button has been pressed. Remote resume signalling from the chip will last 11.45mS to 12.45mS.

Get_Status_Device			
			80 00 00 00 00 00 02 00
	Returns:		xx yy xx[0] = Self Powered xx[1] = Remote Wakeup xx[7:2] = 0 yy = 00 (Reserved)
	Default:		Accept (undefined in USB Spec)
	Addressed:		Accept
	Configured:	Accept	
	Notes:		Use Set_Feature_Device/Clear_Feature_Device to set/clear remote wakeup.

Get_Status_Interface			
			81 00 00 00 00 00 02 00
	Returns:		00 00
	Default:		Stall (undefined in USB Spec)
	Addressed:		Stall
	Configured:	Accept	
	Notes:		Both return bytes are reserved and currently 00.

Get_Status_Endpt0			82 00 00 00 xx 00 02 00 82 00 00 00 00 00 02 00 82 00 00 00 80 00 02 00 xx = 00 = Endpt0 OUT xx = 80 = Endpt0 IN
	Returns:		xx yy xx[0] = Halt xx[7:1] = 0 yy = 00 (Reserved)
	Default:		Accept (undefined in USB Spec)
	Addressed:		Accept
	Configured:	Accept	
	Notes:		Use Set_Feature_Endpt0/Clear_Feature_Endpt0 to (try to) set/clear Halt bit. According to USB, "It is neither required or recommended that the Halt feature be implemented for the Default Control Pipe." Since a new SETUP command will clear any Endpt0 halt bit, it is impossible to tell if there really is a halt bit.

Get_Status_Endpt1 82 00 00 00 81 00 02 00

Returns: xx yy
- xx[0] = Halt
- xx[7:1] = 0
- yy = 00 (Reserved)

Default: Stall (undefined in USB Spec)
Addressed: Stall
Configured: Accept
Notes: Use Set_Feature_Endpt1/Clear_Feature_Endpt1 to set/clear Halt bit.

Get_Configuration

80 08 00 00 00 00 01 00

Returns: xx
xx = config value

Default: Accept (undefined in USB Spec) — returns 00
Addressed: Accept — returns 00
Configured: Accept — returns 01
Notes: Use Set_Configuration to change.

Get_Interface

81 0A 00 00 00 00 01 00

Returns: 00
Default: Undefined in USB Spec
Addressed: Stall
Configured: Accept — returns 00
Notes: Command has no alternate interfaces, so only valid value is 00

Get_Protocol

A1 03 00 00 00 00 01 00

Returns: xx
- xx = 00 = Boot protocol
- xx = 01 = Report protocol

Default: Accept
Addressed: Accept
Configured: Accept
Notes: Defaults to Report protocol after USB Reset. Use Set_Protocol to change.

Get_Desc_Device 80 06 00 01 00 00 nn00
80 06 00 01 00 00 12 00

Returns: 12 01 00 02 00 00 00 08
vv vv pp pp dd dd mm PP
ss 01

- vv vv = vendor id
- pp pp = product id (vendor specified)
- dd dd = device id (vendor specified) (bcd rev_id byte)
- mm = iManufacturer
- PP = iProduct
- ss = iSerialNumber (00 - no string)

Default: Accept
Addressed: Accept
Configured: Accept

Get_Desc_Config 80 06 00 02 00 00 nn 00
80 06 00 02 00 00 22 00

Returns:
09 02 22 00 01 01 00 A0
32 09 04 00 00 01 03 01
02 00 09 21 10 01 00 01
22 rr 00 07 05 81 03 04
00 0A

rr = HID Report descriptor length

These values are determined by jumper configuration see strap table.

Without Z-Wheel:

09 02 22 00 01 01 00 A0
32 09 04 00 00 01 03 01
02 00 09 21 10 01 00 01
22 32 00 07 05 81 03 04
00 0A

	//	Config Descriptor
| 09	//	bLength
| 02	//	bDescriptorType
| 22	//	wTotalLength (34 decimal)
| 00	//	high byte of WTotalLength
| 01	//	bNumInterfaces
| 01	//	bConfigurationValue
| 00	//	iConfiguration
| A0	//	bmAttributes (bus powered/remote wakeup)
| 32	//	MaxPower (in 100mA in 2mA units)
	//	Interface Descriptor
| 09	//	bLength
| 04	//	bDescriptorType
| 00	//	bInterfaceNumber
| 00	//	bAlternateSetting
| 01	//	bNumEndpoints
| 03	//	bInterfaceClass (HID Class)
| 01	//	bInterfaceSubClass
| 02	//	bInterfaceProtocol
| 00	//	iInterface
	//	HID Descriptor
| 09	//	bLength
| 21	//	bDescriptorType
| 11	//	bcdHID (HID Release ##.##; HID 1.1 compliant)
| 01	//	
| 00	//	bCountry
| 01	//	bAvailable
| 22	//	bType
| 32	//	wLength (Length of HID Report below)
| 00	//	
	//	Endpoint Descriptor
| 07	//	bLength
| 05	//	bDescriptorType
| 81	//	bEndpointAddress (IN & #=1)
| 03	//	bmAttributes (Interrupt)
| 04	//	wMaxPacketSize
| 00	//	
| 0A	//	bInterval (10mS)

With Z-Wheel:

09 02 22 00 01 01 00 A0
32 09 04 00 00 01 03 01
02 00 09 21 10 01 00 01
22 34 00 07 05 81 03 04
00 0A

	//	Config Descriptor
| 09	//	bLength
| 02	//	bDescriptorType
| 22	//	wTotalLength (34 decimal)
| 00	//	high byte of WTotalLength
| 01	//	bNumInterfaces
| 01	//	bConfigurationValue
| 00	//	iConfiguration
| A0	//	bmAttributes (bus powered/remote wakeup)
| 32	//	MaxPower (in 100mA in 2mA units)
	//	Interface Descriptor
| 09	//	bLength
| 04	//	bDescriptorType
| 00	//	bInterfaceNumber
| 00	//	bAlternateSetting
| 01	//	bNumEndpoints
| 03	//	bInterfaceClass (HID Class)
| 01	//	bInterfaceSubClass
| 02	//	bInterfaceProtocol
| 00	//	iInterface
	//	HID Descriptor
| 09	//	bLength
| 21	//	bDescriptorType
| 11	//	bcdHID (HID Release ##.##; HID 1.1 compliant)
| 01	//	
| 00	//	bCountry
| 01	//	bAvailable
| 22	//	bType
| 34	//	wLength (Length of HID Report below)
| 00	//	
	//	Endpoint Descriptor
| 07	//	bLength
| 05	//	bDescriptorType
| 81	//	bEndpointAddress (IN & #=1)
| 03	//	bmAttributes (Interrupt)
| 04	//	wMaxPacketSize
| 00	//	
| 0A	//	bInterval (10mS)

Default: Accept
Addressed: Accept
Configured: Accept
Notes: This is the concatenation of 4 descriptors:
Configuration
Interface
HID
Endpt

Get_Desc_String		80 06 xx 03 00 00 nn 00
xx= 00 Language String
02 Product String

Returns: ss 03 "unicode string"
ss = String descriptor length

These values are determined by jumper configuration on page 14:

For xx=00:

04 03 09 04 // Language ID

For xx=02:

20 00 55 00 53 00 42 00 //" USB"
20 00 4f 00 70 00 74 00 //"Opt"
69 00 63 00 61 00 6c 00 //"ical"
20 00 4d 00 6f 00 75 00 //"Mou"
73 00 65 00 //"se

Get_Desc_HID

81 06 00 21 00 00 09 00

Returns: 09 21 10 01 00 01 22 rr 00
rr = HID Report descriptor length

These values are determined by jumper configuration see table on page 14:

Without Z-wheel:

09 21 10 01 00 01 22 32 00

	//	HID Descriptor
| 09	//	bLength
| 21	//	bDescriptorType
| 10	//	bcdHID (HID Release ##.##; HID 1.1 compliant)
| 01	//	
| 00	//	bCountry
| 01	//	bAvailable
| 22	//	bType
| 32	//	wLength (Length of HID Report below)
| 00	//	

With Z-wheel:

09 21 10 01 00 01 22 34 00

	//	HID Descriptor
| 09	//	bLength
| 21	//	bDescriptorType
| 10	//	bcdHID (HID Release ##.##; HID 1.1 compliant)
| 01	//	
| 00	//	bCountry
| 01	//	bAvailable
| 22	//	bType
| 34	//	wLength (Length of HID Report below)
| 00	//	

Default: Accept
Addressed: Accept
Configured: Accept

Get_Desc_HID_Report 81 06 00 22 00 00 nn 00

Returns: This returns a report descriptor that describes how many buttons and x, y, z data.

These values are determined by jumper configuration see table on page 14:

Without Z-wheel:

```
05 01 09 02 A1 01 09 01
A1 00 05 09 19 01 29 xx          // xx = # buttons
15 00 25 01 75 01 95 xx          // xx = # buttons
81 02 75 yy 95 01 81 01          // yy = 8 - # buttons
05 01 09 30 09 31 15 81
25 7F 75 08 95 02 81 06
C0 C0
          //     HID Report
| 05      //     USAGE_PAGE (Generic Desktop)
| 01      //
| 09      //     USAGE (Mouse)
| 02      //
| A1      //     COLLECTION (Application)
| 01      //
| 09      //      USAGE (Pointer)
| 01      //
| A1      //      COLLECTION (Physical)
| 00      //
| 05      //     USAGE_PAGE (Button)
| 09      //
| 19      //     USAGE_MINIMUM (Button 1)
| 01      //
| 29      //     USAGE_MAXIMUM (Button #)
| xx      //
| 15      //     LOGICAL_MINIMUM (0)
| 00      //
| 25      //     LOGICAL_MAXIMUM (1)
| 01      //
| 75      //     REPORT_SIZE (1)
| 01      //
| 95      //     REPORT_COUNT (Button #)
| xx      //
| 81      //     INPUT (Data,Var,Abs)
| 02      //
| 75      //     REPORT_SIZE (8 - Button #)
| yy      //
| 95      //     REPORT_COUNT (1)
| 01      //
| 81      //     INPUT (Cnst,Ary,Abs)
| 01      //
| 05      //     USAGE_PAGE (Generic Desktop)
| 01      //
| 09      //     USAGE (X)
| 30      //
| 09      //     USAGE (Y)
| 31      //
| 15      //     LOGICAL_MINIMUM (-127)
| 81      //
| 25      //     LOGICAL_MAXIMUM (127)
| 7F      //
| 75      //     REPORT_SIZE (8)
| 08      //
| 95      //     REPORT_COUNT (2)
| 02      //
| 81      //     INPUT (Data,Var,Rel)
| 06      //
| C0      //     END_COLLECTION
| C0      //     END_COLLECTION
```

With Z-wheel:

```
05 01 09 02 A1 01 09 01
A1 00 05 09 19 01 29 xx          // xx = # buttons
15 00 25 01 75 01 95 xx          // xx = # buttons
81 02 75 yy 95 01 81 01          // yy = 8 - # buttons
05 01 09 30 09 31 09 38
15 81 25 7F 75 08 95 03
81 06 C0 C0
          //    HID Report
| 05      //    USAGE_PAGE (Generic Desktop)
| 01      //
| 09      //    USAGE (Mouse)
| 02      //
| A1      //    COLLECTION (Application)
| 01      //
| 09      //     USAGE (Pointer)
| 01      //
| A1      //     COLLECTION (Physical)
| 00      //
| 05      //            USAGE_PAGE (Button)
| 09      //
| 19      //            USAGE_MINIMUM (Button 1)
| 01      //
| 29      //            USAGE_MAXIMUM (Button #)
| xx      //
| 15      //            LOGICAL_MINIMUM (0)
| 00      //
| 25      //            LOGICAL_MAXIMUM (1)
| 01      //
| 75      //            REPORT_SIZE (1)
| 01      //
| 95      //            REPORT_COUNT (Button #)
| xx      //
| 81      //            INPUT (Data,Var,Abs)
| 02      //
| 75      //            REPORT_SIZE (8 - Button #)
| yy      //
| 95      //            REPORT_COUNT (1)
| 01      //
| 81      //            INPUT (Cnst,Ary,Abs)
| 01      //
| 05      //            USAGE_PAGE (Generic Desktop)
| 01      //
| 09      //            USAGE (X)
| 30      //
| 09      //            USAGE (Y)
| 31      //
| 09      //            USAGE (Wheel)
| 38      //
| 15      //            LOGICAL_MINIMUM (-127)
| 81      //
| 25      //            LOGICAL_MAXIMUM (127)
| 7F      //
| 75      //            REPORT_SIZE (8)
| 08      //
| 95      //            REPORT_COUNT (3)
| 03      //
| 81      //            INPUT (Data,Var,Rel)
| 06      //
| C0      //            END_COLLECTION
| C0      //            END_COLLECTION
```

Default: Accept
Addressed: Accept
Configured: Accept
Notes: The length of this report is needed in the HID descriptor.

Get_HID_Input A1 01 00 01 00 00 nn 00
A1 01 00 01 00 00 04 00 OR
A1 01 00 01 00 00 03 00 (if no Z-wheel present)

Returns: bb xx yy zz OR
bb xx yy (if no Z-wheel present)
bb = button byte
xx = X motion byte
yy = Y motion byte
zz = Z motion byte

Default: Stall
Addressed: Stall
Configured: Accept
Notes: If the device is configured, it will always respond with a report for this command, even if no motion or button changes have occurred. In this case, it would report 00 for motion and simply report the current button state. If a report is pending on endpt1, the data there will be reported and the report on endpt1 cleared.

The mouse will only create new button/motion packets when it is in the Configured state.

Get_Idle
A1 02 00 00 00 00 01 00

Returns: rr
rr = rate in multiples of 4mS
Default: Accept
Addressed: Accept
Configured: Accept
Notes: The third byte of the command is to select the Report ID. There is only one for the mouse — so, using 00 or 01 will work.
See also Set_Idle.

Get_Vendor_Test C0 01 00 00 xx 00 01 00
ii = ignore
xx = address of register to read

Returns: rr (depends on register read)

Default: Accept
Addressed: Accept
Configured: Accept
Notes: Address range (xx) should be x00 to x3F. Behavior for addresses above 0x3F is not guaranteed and may be modified in the future.
See page 28 for description of registers.

Set_Address 00 05 xx 00 00 00 00 00
xx = new device address, from 00 to 7F
Default: Accept
Addressed: Accept
Configured: Undefined in USB Spec

Set_Configuration 00 09 xx 00 00 00 00 00
xx = 00 = not configured
xx = 01 = configured
Default: Undefined in USB Spec
Addressed: Accept
Configured: Accept

Set_Interface 01 0B 00 00 00 00 00 00

	Default:		Stall (undefined in USB Spec)
	Addressed:		Stall
	Configured:	Accept	
	Notes:		Mouse has only one valid interface (00) and alternate setting (00). Invalid values will cause stall. Chip retains previous (valid) interface state after executing this command in configured mode even if invalid values are given and command was stalled.

Set_Protocol 21 0B xx 00 00 00 00 00

xx = 00 = Boot protocol

xx = 01 = Report protocol

	Default:		Accept (Not in USB Spec)
	Addressed:		Accept (Not in USB Spec)
	Configured:	Accept	
	Notes:		3 byte data packets will be reported in boot mode.

Set_Feature_Device 00 03 01 00 00 00 00 00

	Default:		Accept (undefined in USB Spec)
	Addressed:		Accept
	Configured:	Accept	
	Notes:		This sets the remote wakeup bit.

Set_Feature_Endpt0

02 03 00 00 xx 00 00 00

02 03 00 00 00 00 00 00

02 03 00 00 80 00 00 00

xx = 00 = Endpt0 OUT

xx = 80 = Endpt0 IN

	Default:		Stall (undefined in USB Spec)
	Addressed:		Stall
	Configured:	Stall	
	Notes:		This (tries to) sets the halt bit. The chip always stalls the status stage for this command. The chip never reports the halt bit set for Endpt0 with the Get_Status_Endpt0 command, as any new SETUP command will clear Endpt0 stall.

Set_Feature_Endpt1 02 03 00 00 81 00 00 00

	Default:		Stall (undefined in USB Spec)
	Addressed:		Stall
	Configured:	Accept	
	Notes:		Sets the halt bit for Endpt1.

Clear_Feature_Device 00 01 01 00 00 00 00 00

	Default:		Accept (undefined in USB Spec)
	Addressed:		Accept
	Configured:	Accept	
	Notes:		This clears the remote wakeup bit.

Clear_Feature_Endpt0 02 01 00 00 xx 00 00 00

02 01 00 00 00 00 00 00

02 01 00 00 80 00 00 00

xx = 00 = Endpt0 OUT

xx = 80 = Endpt0 IN

	Default:		Accept (undefined in USB Spec)
	Addressed:		Accept
	Configured:	Accept	
	Notes:		The chip does NOT stall like it does for Set_Feature_Endpt0.

Clear_Feature_Endpt1	02 01 00 00 81 00 00 00	
Default:		Stall (undefined in USB Spec)
Addressed:		Stall
Configured:	Accept	
Notes:		See Set_Feature_Endpt1.

Set_Idle		21 0A 00 rr 00 00 00 00
		rr = rate in multiples of 4mS
Default:		Accept
Addressed:		Accept
Configured:	Accept	
Notes:		The third byte of the command is to select the Report ID. There is only one for the mouse — so, using either 00 or 01 will work. The fourth byte of the command sets the rate in multiples of 4mS. The initial value for mice will be x00 which means "infinite" — that is packets only come out when there is a change in data. Data will only be allowed to come out when the device is configured. However, the chip will accept the command in Default or Addressed mode and use that value when the device is later configured.

Set_Vendor_Test		40 01 00 00 xx yy 00 00
		xx = address yy = data
Default:		Accept
Addressed:		Accept
Configured:	Accept	
Notes:		Address range for "xx" should be 0x00 to 0x3F. Addresses above this are reserved for possible future use. See also Get_Vendor_Test.

Poll_Endpt1		
Returns:		bb xx yy zz bb = button byte xx = X motion byte yy = Y motion byte zz = Z motion byte (if Z-Wheel)
Default:		Ignore request
Addressed:		Ignore request
Configured:	Accept (NAK if no data; Send packet if available)	
Notes:		See also Get_HID_Input. Endpt will only stall if halt bit is set by Set_Feature_Endpt1. Details of data packet are below. Endpt1 should be polled at least every 10 frames (mS). It is typically polled every 8 frames on Windows machines. For internal testing, Endpt1 can be continuously polled if desired. The chip will not generate any report packets unless in the Configured state. If Endpt1 is currently empty, any motion or button change will be loaded into the Endpt1 buffers. Once the Endpt1 buffers are full, any further motion events will get accumulated. When the Endpt1 buffers are later polled and emptied, the current accumulated X/Y/Z values will be loaded into the Endpt1 buffers. After transferring their data, the accumulation registers are reset so they are ready to start accumulating new motion events.

Button information is handled a bit differently. If the Endpt1 buffers are empty, and a button change event occurs, the new button state is put into the Endpt1 buffers. At the same time, the button state that is put in Endpt1 is copied for later use. While Endpt1 is full, changes in button state are essentially ignored. When Endpt1 is emptied, if the current button state is different than that which was last loaded into Endpt1, then the new state will be loaded and a new copy saved. Basically, the button state that is loaded into Endpt1 is always the current button state at that point in time. It should also be noted that there is hardware on the chip to help de-bounce the buttons.

USB Data Packet Format

Bit	7	6	5	4	3	2	1	0
Byte 1	0	0	0	B5	B4	B3(MB)	B2(RB)	B1(LB)
Byte 2	X7	X6	X5	X4	X3	X2	X1	X0
Byte 3	Y7	Y6	Y5	Y4	Y3	Y2	Y1	Y0
Byte 4	Z7	Z6	Z5	Z4	Z3	Z2	Z1	Z0

Special note on wLength:

The wLength paramater in commands specifies the maximum number of bytes a device should send back. The commands listed below are not able to handle a wLength of 0 correctly.

Get_Status_Device
Get_Status_Interface
Get_Status_Endpt0
Get_Status_Endpt1
Get_Configuration
Get_Interface

This chip will send one byte of data rather than none when wLength=0 is requested for the above commands.

USB Data Packet Format, With Z wheel

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Byte 1	0	0	0	Button 5	Button 4	Button 3	Button 2	Button 1
Byte 2	X[7]	X[6]	X[5]	X[4]	X[3]	X[2]	X[1]	X[0]
Byte 3	Y[7]	Y[6]	Y[5]	Y[4]	Y[3]	Y[2]	Y[1]	Y[0]
Byte 4	Z[7]	Z[6]	Z[5]	Z[4]	Z[3]	Z[2]	Z[1]	Z[0]

USB Data Packet Format, Without Z wheel

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Byte 1	0	0	0	Button 5	Button 4	Button 3	Button 2	Button 1
Byte 2	X[7]	X[6]	X[5]	X[4]	X[3]	X[2]	X[1]	X[0]
Byte 3	Y[7]	Y[6]	Y[5]	Y[4]	Y[3]	Y[2]	Y[1]	Y[0]

Registers

The sensor can be programmed through registers, via the USB port, and configuration and motion data can be read from these registers. Certain registers must be "enabled" after power up but before first read or write to that register. The registers will be "disabled" by VDD going low or sending a USB reset command.

Address	Register
0x00	Product_ID
0x01	Revision_ID
0x02	Motion
0x03	DeltaX
0x04	DeltaY
0x05	SQUAL
0x06	Shutter_Upper
0x07	Shutter_Lower
0x08	Maximum_Pixel
0x09	Average_Pixel
0x0a	Minimum_Pixel
0x0b	Pix_Grab
0x0c	Dz
0x0d	Configuration_bits
0x3f	InvRevID

Product_ID Address: 0x00

Access: Read Reset Value: 0x05

Bit	7	6	5	4	3	2	1	0
Field	PID_7	PID_6	PID_5	PID_4	PID_3	PID_2	PID_1	PID_0

Data Type: Eight bit number with the product identifier.

USAGE: The value in this register does not change; it can be used to verify that the sensor communications link is OK.

Revision_ID Address: 0x01

Access: Read Reset Value: 0xNN

Bit	7	6	5	4	3	2	1	0
Field	RID_7	RID_6	RID_5	RID_4	RID_3	RID_2	RID_1	RID_0

Data Type: Eight bit number with current revision of the IC.

USAGE: NN is a value between 00 and FF which represent the current design revision of the device.

For example, NN for IC revision 3.4 is 34.

Motion	Address: 0x02
Access: Read	Reset Value: Undefined

Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Field	MOT	Reserved	Reserved	BUT_5	BUT_4	BUT_3	BUT_2	BUT_1

Data Type: Bit field

Field Name	**Description**
MOT	Motion since last report **0 = no motion** 1 = motion has occurred
BUT_5	When used as a 5 button mouse, reports the status of OPT 1/B5 button pin 0 = pin at logic 1 (Vdd3) 1 = pin at logic 0 (GND)
BUT_4	When used as a 5 button mouse, reports the status of OPT 0/B4 button pin 0 = pin at logic 1 (Vdd3) 1 = pin at logic 0 (GND)
BUT_3	Reports the status of B3 0 = pin at logic 1 (Vdd3) 1 = pin at logic 0 (GND)
BUT_2	Reports the status of B2 0 = pin at logic 1 (Vdd3) 1 = pin at logic 0 (GND)
BUT_1	Reports the status of B1 0 = pin at logic 1 (Vdd3) 1 = pin at logic 0 (GND)
Reserved	Reserved for future use **Reset Value = 0**

USAGE: A “1” in the motion bit indicates that the USB endpoint has valid data. This register is included for test purposes only. For navigation use, use the USB HID defined commands. The button status bits reported are for the debounce signals.

DeltaX	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

Data Type: Bit field

USAGE: The value in this register reflects the last USB delta X data output or data queued for output. This register is included for test purposes only. For navigation use, use the HID defined commands. Data is 8 bit, 2’s complement. Absolute value is determined by the currently set resolution.

Register 0x03 must be read before register 0x04 (Delta Y)

DeltaY Address: 0x04

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

Data Type: Bit field

USAGE: The value in this register reflects the last USB delta Y data output or data queued for output.

Register 0x03 should be read before register 0x04 (Delta Y), else Delta Y will return 0.

This register is included for test purposes only. For navigation use, use the HID defined commands. Data is 8 bit, 2's complement. Absolute value is determined by the currently set resolution.

SQUAL Address: 0x05

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

Data Type: Eight bit number.

USAGE: SQUAL is a measure of the number of features visible by the sensor in the current frame. The maximum value is 144. Since small changes in the current frame can result in changes in SQUAL, slight variations in SQUAL on one surface is expected.

Shutter_Upper Address: 0x06

Access: Read Reset Value: 0x01

Bit	7	6	5	4	3	2	1	0
Field	SH_7	SH_6	SH_5	SH_4	SH_3	SH_2	SH_1	SH_0

Data Type: Eight bit number.

USAGE: The combination of Shutter_Upper and Shutter_Lower is a 16-bit number. This is the number of clocks the shutter was open for the last image taken. The units are in main clocks ticks (nominally 24MHz). To avoid split read issues, read Shutter_Upper first.

Shutter_Lower	Address: 0x07
Access: Read	Reset Value: 0x64

Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Field	SL_7	SL_6	SL_5	SL_4	SL_3	SL_2	SL_1	SL_0

Data Type: Eight bit number.

USAGE: The combination of Shutter_Upper and Shutter_Lower is a 16-bit number. This is the number of clocks the shutter was open for the last image taken. The units are in main clocks ticks (nominally 24MHz). To avoid split read issues, read Shutter_Upper first.

Maximum_Pixel	Address: 0x08
Access: Read	Reset Value: 0x00

Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Field	MX_7	MX_6	MX_5	MX_4	MX_3	MX_2	MX_1	MX_0

Data Type: Eight bit number.

USAGE: This is the maximum pixel value from the last image taken.

Average_Pixel	Address: 0x09
Access: Read	Reset Value: 0x00

Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Field	AC_7	AC_6	AC_5	AC_4	AC_3	AC_2	AC_1	AC_0

Data Type: Eight bit number.

USAGE: This is the accumulated pixel value from the last image taken. For the 15X15 raw image, only the 8 most interesting bits are reported ([14:7]). To get the true average pixel value, multiply this register value by 1.75.

Minimum_Pixel	Address: 0x0a
Access: Read	Reset Value: 0x00

Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Field	MN_7	MN_6	MN_5	MN_4	MN_3	MN_2	MN_1	MN_0

Data Type: Eight bit number.

USAGE: This is the minimum pixel value from the last image taken.

Pix_Grab	Address: 0x0b
Access: Read/Write	Reset Value: 0x00

Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Field	PG_7	PG_6	PG_5	PG_4	PG_3	PG_2	PG_1	PG_0

Data Type:	Bit field.
USAGE:	The pixel grabber captures 1 pixel per frame. If there is a valid pixel in the grabber when this is read, the MSB will be set, an internal counter will incremented to captured the next pixel and the grabber will be armed to capture the next pixel. It will take 225 reads to upload the completed image. Any write to this register will reset and arm the grabber to grab pixel 0,0 on the next image.

Pixel Address Map (Looking through the aperture of the sensor)

0	15	30	45	60	75	90	105	120	135	150	165	180	195	210
1	16	31	46	61	76	91	106	121	136	151	166	181	196	211
2	17	32	47	62	77	92	107	122	137	152	167	182	197	212
3	18	33	48	63	78	93	108	123	138	153	168	183	198	213
4	19	34	49	64	79	94	109	124	139	154	169	184	199	214
5	20	35	50	65	80	95	110	125	140	155	170	185	200	215
6	21	36	51	66	81	96	111	126	141	156	171	186	201	216
7	22	37	52	67	82	97	112	127	142	157	172	187	202	217
8	23	38	53	68	83	98	113	128	143	158	173	188	203	218
9	24	39	54	69	84	99	114	129	144	159	174	189	204	219
10	25	40	55	70	85	100	115	130	145	160	175	190	205	220
11	26	41	56	71	86	101	116	131	146	161	176	191	206	221
12	27	42	57	72	87	102	117	132	147	162	177	192	207	222
13	28	43	58	73	88	103	118	133	148	163	178	193	208	223
14	29	44	59	74	89	104	119	134	149	164	179	194	209	224

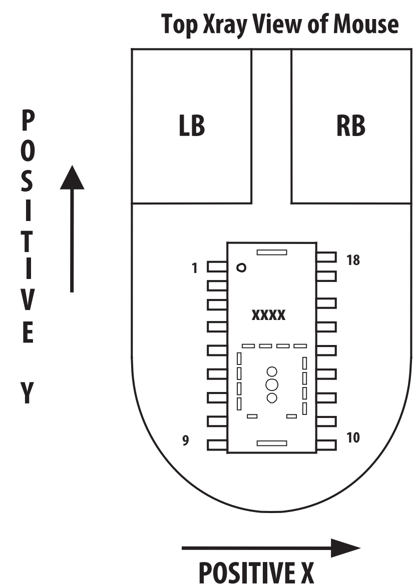
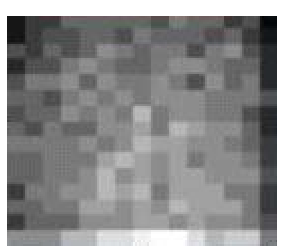


Figure 20. Pixel Map.

The following images are the output of the pixel dump command. The data ranges from zero for complete black, to 63 for complete white. An internal AGC circuit adjusts the shutter value to keep the brightest feature (max pixel) in the mid 50's.

White Paper



Manila Folder

Figure 21. Pixel Dump Pictures

Dz Address: 0x0c

Access: Read Reset Value: 0x00

Bit	7	6	5	4	3	2	1	0
Field	Z_7	Z_6	Z_5	Z_4	Z_3	Z_2	Z_1	Z_0

Data Type: Bit field

USAGE: If mouse is configured to contain a Z-wheel, this register contains the Z-wheel count. Range is from –127 to 127 decimal.

Configuration_bits Address: 0x0d

Access: Read/Write Reset Value: 0x00

Bit	7	6	5	4	3	2	1	0
Field	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	RES

Data Type: Bit field

Field Name	Description
RES	Resolution in counts per inch **0 = 500** 1 = 1000

USAGE: Allows configuration of cpi in sensor

InvRevID Address: 0x03f

Access: Read Reset Value: 0xf0

Bit	7	6	5	4	3	2	1	0
Field	$RRID_7$	$RRID_6$	$RRID_5$	$RRID_4$	$RRID_3$	$RRID_2$	$RRID_1$	$RRID_0$

Data Type: Eight bit number with current revision of the IC.

USAGE: Contains the inverse of the revision ID which is located in register 0x01.

IC Register state after Reset (power up)

Ordering Information

Specify part number as follows:

ADNS-5000 = Sensor IC in a 18-pin staggered DIP, 22 per tube.

ADNS-5100 = Lens

ADNS-5100-001 = Trim Lens

ADNS-5200 = LED clip

AV02-1006EN - December 4, 2009