

## Data Sheet



Lead (Pb) Free  
RoHS 6 fully  
compliant



### Description

The ADBM-A350 sensor is a small form factor (SFF) optical finger navigation system.

The ADBM-A350 is a low- power optical finger navigation sensor. It has a new, low-power architecture and automatic power management modes, making it ideal for battery-and power-sensitive applications such as mobile phones.

The ADBM-A350 is capable of high-speed motion detection – up to 20ips. In addition, it has an on-chip oscillator and integrated LED for optical navigation to minimize external components.

There are no moving parts, thus provide high reliability and less maintenance for the end user. In addition, precision optical alignment is not required, facilitating high volume assembly.

The sensor is programmed via registers through either a serial peripheral interface or a two wire interface port. It is packaged into a 17-pin FPC module for ease of assembly via ZIF connector.

### Theory of Operation

The ADBM-A350 is based on Optical Finger Navigation (OFN) Technology, which measures changes in position by optically acquiring sequential surface images (frames) and mathematically determining the direction and magnitude of movement.

The ADBM-A350 contains an Image Acquisition System (IAS), a Digital Signal Processor (DSP), and a communication system.

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  $\Delta x$  and  $\Delta y$  relative displacement values.

The host reads the  $\Delta x$  and  $\Delta y$  information from the sensor serial port if a motion interrupt is published. The micro-controller then translates the data into cursor navigation, rocker switch, scrolling or other system dependent navigation data.

### Features

- Low power architecture
- Self-adjusting power-saving modes for longer battery life
- High speed motion detection up to 20ips
- Self-adjusting frame rate for optimum performance
- Motion detect interrupt
- Finger detect interrupt
- Soft click and Tap detect interrupt
- Single Interrupt pin
- Optional PWM output to control LED driver to enable illumination feature when finger is on the sensor
- Optional switch input for center click function
- Internal oscillator – no clock input needed
- Selectable 125, 250, 500, 750, 1000 and 1250 cpi resolution
- Single 1.8V supply voltage for analog and digital
- Internal power up reset (POR)
- Selectable Input/Output voltage at 1.8V or 2.8V nominal
- 4-wire Serial peripheral interface (SPI) or Two wire interface (TWI)
- Integrated chip-on-board LED with wavelength of 870nm
- 17-pin FPC module

### Applications

- Finger input devices
- Mobile devices
- Integrated input devices
- Battery-powered input device

Avago customers purchasing the ADBM-350 OFN product are eligible to receive a royalty free license to our US patents 6977645, 6621483, 6950094, 6172354 and 7289649, for use in their end products.

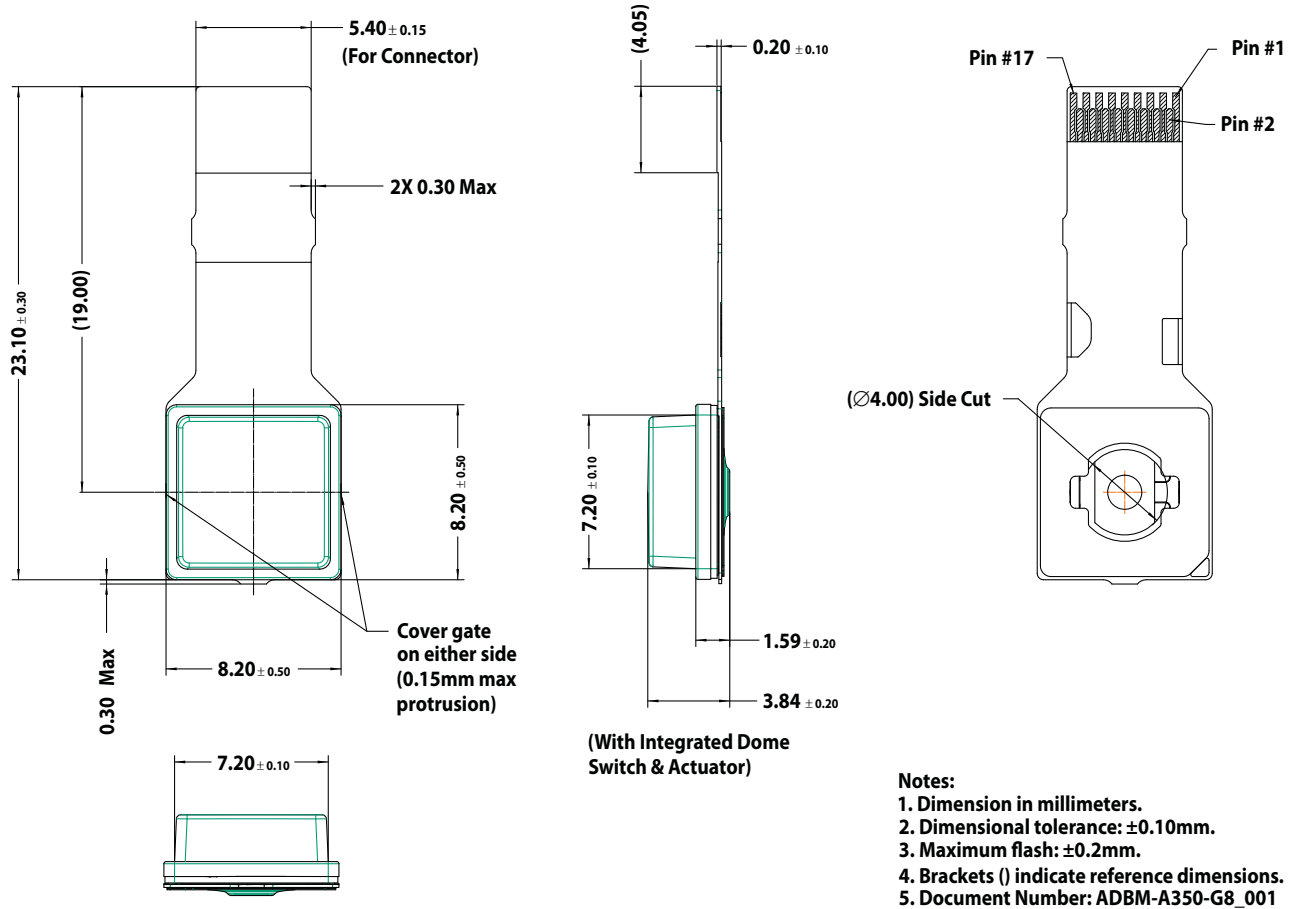
**CAUTION:** It is advised that normal static precautions be taken in handling and assembly of this component to prevent damage and/or degradation which may be induced by ESD.

## Pinout of ADBM-A350 Optical Sensor

Pin	Name	Description	Input/Output pin	Function
1	GND_SHIELD	Ground shield		
2	GPIO	General Purpose Input/ Output pin	I (Schmitt trigger input)/ O (CMOS output)	Pin can be used for FPD output, PWM output or Dome/ Button click input. If configure as input do not leave pin unconnected
3	GND	Ground		
4	IO_NCS_A1	TWI address set or Chip Select	I (Schmitt trigger input)	SPI : NCS (chip select) active low signal TWI Address Select, A1 Do no leave pin unconnected
5	IO_MISO_SDA	TWI serial data or Master In Slave Out	In SPI – CMOS output. In TWI – open drain I/O	SPI : MISO (Master Input Slave Out) signal TWI : serial data signal
6	IO_MOSI_A0	TWI address set or Master Out Slave In	I (Schmitt trigger input)	SPI : MOSI (Master Out Slave In) signal TWI Address Select, A0 Do no leave pin unconnected
7	IO_CLK	Serial clock input	I (Schmitt trigger input)	Serial clock signal
8	NRST	Hardware Chip Reset	I (Schmitt trigger input)	Set to high when not used Active low signal
9	DOME-	Dome -		
10	EVENT_INT	Event Interrupt (active low output)	O (CMOS output)	Open when not used Default active low signal, can be changed in Event control register 0x1d
11	SHTDWN	Shutdown (active high input)	I (Schmitt trigger input)	Set to low when not used Active high signal
12	NC	No Connect		No connection
13	VDD	Voltage supply		Supply 1.8V
14	DOME+	Dome +		
15	IO_SELECT	SPI / TWI Select	I (Schmitt trigger input)	TWI : GND or SPI : High
16	VDDIO	Voltage supply for Input/ Output		Supply 1.8V or 2.8V
17	NC	No Connect		No connection

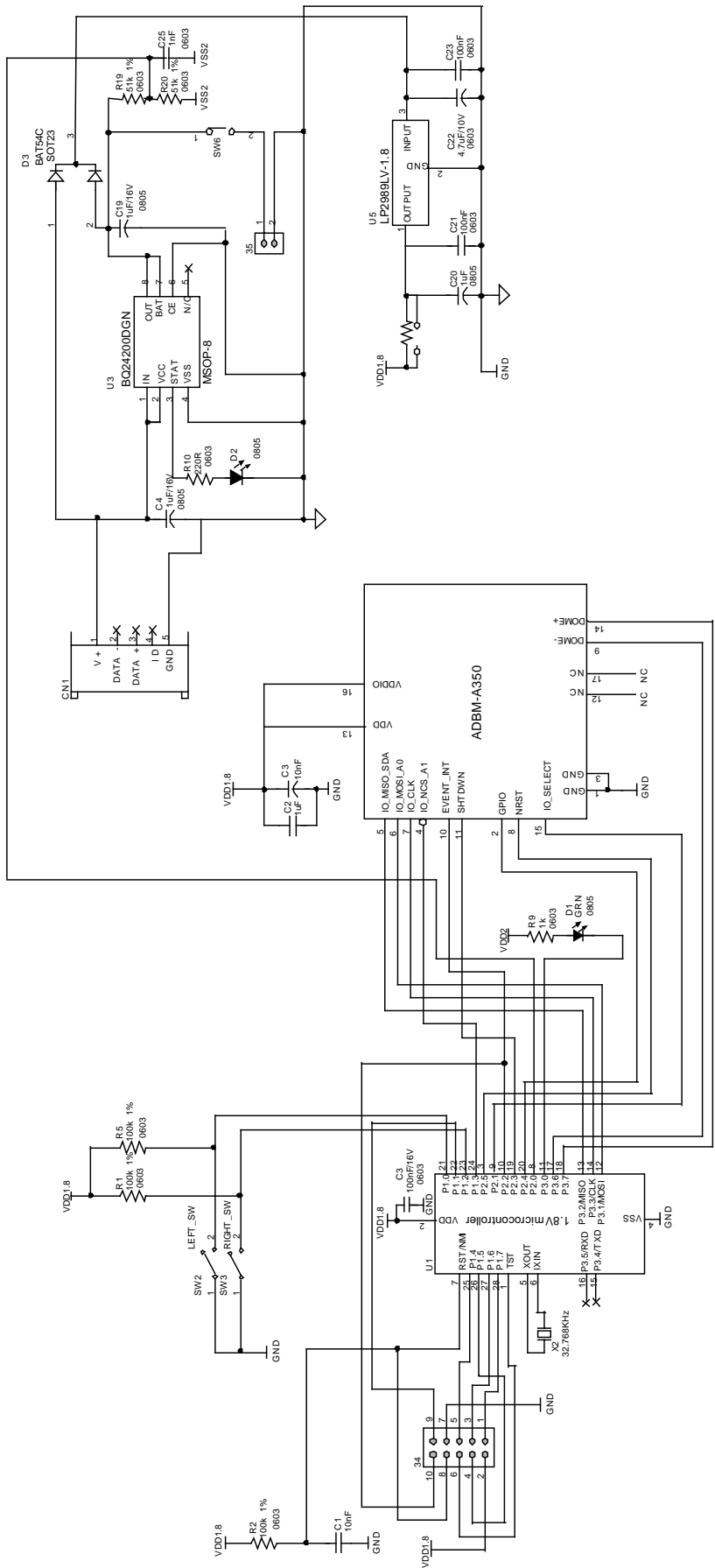
Note:

1. NC pins can be tied to VDD, GND or left open/ unconnected.



- Notes:**
1. Dimension in millimeters.
  2. Dimensional tolerance:  $\pm 0.10$ mm.
  3. Maximum flash:  $\pm 0.2$ mm.
  4. Brackets ( ) indicate reference dimensions.
  5. Document Number: ADBM-A350-G8\_001

Figure 1. Package outline drawing



Note :- Dome + must be connected to MCU to detect button change state and Dome - can be connected to GND.

Figure 2. Schematic diagram for interface between ADBM-A350 and 1.8V microcontroller via SPI

## Regulatory Requirements

- Passes FCC or CISPR 22 Class B emission limits when assembled following Avago Technologies recommendations.
- Passes IEC 61000-4-3 and IEC61000-4-6 Class A Immunity limits when assembled following Avago Technologies recommendations.

## Absolute Maximum Ratings

Parameter	Symbol	Minimum	Maximum	Units	Notes
Storage Temperature	$T_S$	-40	85	°C	
Analog and Digital Supply Voltage	VDD	-0.5	2.1	V	
I/O Supply Voltage	VDDIO	-0.5	3.7	V	
ESD (sensor only)			2	kV	All pins, human body model JESD22-A114-E
Input Voltage	$V_{IN}$	-0.5	VDD+0.5 VDDIO+0.5	V	
Latchup Current	$I_{out}$		20	mA	All Pins

Note: Stresses greater than those listed under “Absolute Maximum Ratings” may cause permanent damage to the device. These are the stress ratings only and functional operation of the device at these or any other condition beyond those indicated may affect device reliability.

## Recommended Operating Conditions

Parameter	Symbol	Minimum	Typical	Maximum	Units	Notes
Operating Temperature	$T_A$	-20		70	°C	
Analog and Digital Supply Voltage <sup>[1]</sup>	VDD	1.7	1.8	2.1	Volts	Including $V_{NA}$ noise.
I/O Supply Voltage <sup>[2]</sup>	VDDIO	1.65	1.8 or 2.8	3.6	Volts	Including $V_{NA}$ noise. Sets I/O voltages. See fig 7.
Power Supply Rise Time	$t_{VRT}$	0.001		10	ms	0 to VDD. At minimum rise time, s/
Power Supply Off Time for Valid POR (Power on Reset)	$t_{OFF}$	10			ms	Refer to section “POR During Power Cycling”
Power Off Voltage Level for Valid POR (Power on Reset)	$V_{OFF}$	0		300	mV	Refer to section “POR During Power Cycling”
Supply Noise (Sinusoidal)	$V_{NA}$			100	mV p-p	10 kHz - 50 MHz
Speed	S			20	in/sec	Using prosthetic finger as surface
Transient Supply Current	$I_{DDT}$			80	mA	Max supply current for 500 $\mu$ sec for each supply voltages ramp from 0 to 1.8 V

Notes:

1. Operating temperature of less than -20°C down to -30°C, minimum VDD of 1.8V must be met.
2. To ensure minimum leakage current, VDDIO should be more than (VDD + 0.4)V

## Timing Specifications

Electrical Characteristics over recommended operating conditions. Typical values at 25° C, VDD=VDDIO=1.8V.

Parameter	Symbol	Minimum	Typical	Maximum	Units	Notes
Motion Delay After Reset	t <sub>MOT-RST</sub>	3.5		23	ms	From Hard reset or SOFT_RESET register write to valid register write/read and motion, assuming motion is present
Shutdown	t <sub>SHTDWN</sub>			50	ms	From SHTDWN pin active to low current
Wake from Shutdown	t <sub>WAKEUP</sub>	100			ms	From SHTDWN pin inactive to valid motion. Refer to section "Notes on Shutdown", also note t <sub>MOT-RST</sub>
EVENT_INT Rise Time	t <sub>r-EVENT_INT</sub>		150	300	ns	C <sub>L</sub> = 100 pF
EVENT_INT Fall Time	t <sub>f-EVENT_INT</sub>		150	300	ns	C <sub>L</sub> = 100 pF
SHTDWN Pulse Width	t <sub>p-SHTDWN</sub>	150			ms	
NRST Pulse Width	t <sub>NRST</sub>	20			μs	From edge of valid NRST pulse
Reset Wait Time After Stable Supply Voltage	t <sub>VRT-NRST</sub>	100			ms	

## DC Electrical Specifications

Electrical Characteristics over recommended operating conditions. Typical values at 25° C, VDD=VDDIO=1.8V at default LED setting 13 mA.

Parameter		Typical	Max	Units	Notes
DC average supply current in Run mode	I <sub>VDD</sub>	2.90	4.03	mA	GPIO=SHTDWN=pull low, IO_MISO=NRST=pull high.
DC average supply current in Rest1 mode	I <sub>VDD</sub>	0.35	0.50	mA	GPIO=SHTDWN=pull low, IO_MISO=NRST=pull high.
DC average supply current in Rest2 mode	I <sub>VDD</sub>	0.07	0.12	mA	GPIO=SHTDWN=pull low, IO_MISO=NRST=pull high.
DC average supply current in Rest3 mode	I <sub>VDD</sub>	0.03	0.05	mA	GPIO=SHTDWN=pull low, IO_MISO=NRST=pull high.
Analog shutdown supply current	I <sub>DD</sub> SHTDWN VDD	1.54	26.0	mA	GPIO=pull low, SHTDWN=IO_MISO=NRST=pull high.

## DC Electrical Specifications

Electrical Characteristics over recommended operating conditions. Typical values at 25° C, VDD=VDDIO=1.8V at default LED setting 13 mA.

Parameter	Symbol	Minimum	Typical	Maximum	Units	Notes
VDDIO DC Supply Current	I <sub>VDDIO</sub>			20	μA	
Digital peak supply current	I <sub>PEAK VDD</sub>			10	mA	
LED+ peak supply current	I <sub>PEAK LED+</sub>			35	mA	At LED register setting of 27 mA
Input Low Voltage	V <sub>IL</sub>	-0.05	0	V <sub>DDIO</sub> *0.35	V	IO_MOSI_A0, IO_CLK, IO_MISO_SDA, IO_NCS_A1, NRST, SHTDWN, IO_SELECT
Input High Voltage	V <sub>IH</sub>	V <sub>DDIO</sub> * 0.7	V <sub>DDIO</sub>	V <sub>DDIO</sub> +0.05	V	IO_MOSI_A0, IO_CLK, IO_MISO_SDA, IO_NCS_A1, NRST, SHTDWN, IO_SELECT
Input hysteresis	V <sub>HYS</sub>	100			mV	
Input leakage current	I <sub>leak</sub>		±1	±10	μA	IO_MOSI_A0, IO_CLK, IO_MISO_SDA, IO_NCS_A1, NRST, SHTDWN, IO_SELECT
Output Low Voltage	V <sub>OL</sub>			0.2	V	I <sub>out</sub> = 1.2 mA
Output High Voltage	V <sub>OH</sub>	V <sub>DDIO</sub> -0.2	V <sub>DDIO</sub> -0.1		V	I <sub>out</sub> = 600 μA
Input Capacitance	C <sub>in</sub>			10	pF	MOSI, NCS, SCLK, SHTDWN

## Notes on Power-up Sequence

Below is the power up sequence for ADBM-A350:

1. Apply power.
2. Set NCS pin high, Shutdown pin low. Set IO\_Select pin to low (for TWI) or high (for SPI).
3. If in TWI mode, set A0 and A1 according to the Table TWI slave address in datasheet. This step is skipped if SPI mode is used.
4. Read Product ID to ensure sensor is power up properly.
5. Perform soft reset by writing 0x5A to address 0x3a.
6. Write 0xC9 to address 0x61.

## Note on register settings

Please refer to the OFN A350 firmware design guide for tuning of Speed Switching, Assert/De-assert, Finger Presence Detect and XY Quantization register settings.

## Notes on Shutdown and Reset

The ADBM-A350 can be set in Shutdown mode by asserting or setting SHTDWN pin high. During the shutdown state, supply voltages VDD must be maintained above the minimum level. If these conditions are not met, then the sensor must be restarted by powering down then powering up again for proper operation. Any register settings must then be reloaded.

During the shutdown state, supply voltage VDD must be maintained above the minimum level. For proper operation, SHTDWN pulse width must be at least t<sub>P-SHTDWN</sub>.

Shorter pulse widths may cause the chip to enter an undefined state. In addition, the SPI or TWI port of the sensor should not be accessed when SHTDWN is asserted. Other devices on the same SPI bus can be accessed, as long as the sensor's NCS pin is not asserted. The table below shows the state of various pins during shutdown. After deasserting SHTDWN, wait t<sub>WAKEUP</sub> before accessing the SPI port. Reinitializing the sensor from shutdown state will retain all register data that were written to the sensor prior to shutdown.

The reset of the sensor via SOFT\_RESET register or through the NRST pin would reset all registers to the default value. Any register settings must then be reloaded.

Pin	SHTDWN active
IO_NCS_A1	Functional
IO_MISO_SDA	Undefined
IO_CLK	Undefined
IO_MOSI_A0	Undefined
XY_LED	Low current
EVENT_INT	Undefined
NRST	High
IO_Select	SPI: High, TWI: Low
GPIO	Undefined

Note: There are long wakeup times from shutdown. These features should not be used for power management during normal sensor motion.

## Power on Reset (POR) During Power Cycling

$t_{VRT}$  is the power supply (VDD) rise time specification for a valid power on reset to happen when the sensor is powered up from 0V to VDD. At condition whereby the VDD of the sensor is cycled from VDD to 0V and then to VDD again, the two parameters that govern a valid power on reset are  $V_{OFF}$  and  $t_{OFF}$ . Refer to timing diagram below.

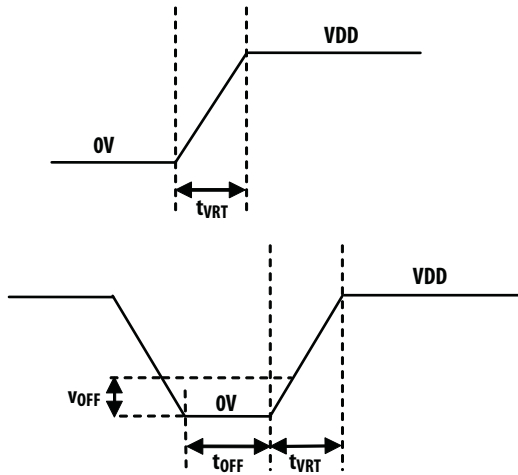


Figure 3. Power on Reset during power cycling

## Power management modes

The ADBM-A350 has three power-saving modes. Each mode has a different motion detection period, affecting response time to sensor motion (Response Time). The sensor automatically changes to the appropriate mode, depending on the time since the last reported motion (Downshift Time). The parameters of each mode are shown in the following table.

Mode	Response Time (nominal)	Downshift Time (nominal)
Rest 1	19.5 ms	250 ms
Rest 2	96 ms	9.5 s
Rest 3	482 ms	582 s

## EVENT\_INT Pin

The Event\_Int pin is a level-sensitive interrupt output that is used to trigger the host micro-controller when one of these events occurs:

- FPD – A change in finger state (finger on to finger off and vice versa) is detected
- Soft Click – Soft Click is detected
- Button – Mechanical button is asserted or de-asserted
- Motion – Motion delta is present.

A read to event register is required to determine the specific event that toggles the interrupt for user to act upon.

The EVENT\_INT will be reset after the user responds to it by reading the respective event status register:

- FPD – reading FPD\_STATUS register (0x7a)
- Soft Click – reading SC\_STATUS register (0x7f)
- Button – reading BUTTON\_STATUS register (0x12)
- Motion – reading DELTA\_X and DELTA\_Y registers until motion are cleared.

## GPIO Pin

The GPIO pin is a level-sensitive input/ output that can be used as

- FPD output – to display FPD status
- Pulse Width Modulated (PWM) output – to control LED driver to enable illumination feature in a product eg mobile phone
- Dome/ Button click input – can be connected to a dome switch that provides an input to the sensor and when a click is detected, sensor can respond by triggering button interrupt and channel the interrupt status through EVENT pin.

Refer to A350 Firmware Design Guide for more details and settings of registers for these features.

## LED Mode

For power savings, the LED will not be continuously on. ADBM-A350 will flash the LED only when needed.



## I/O Pin Status Test

This feature allows the user to verify the connectivity and the state of the I/O pin.

To run the test for input pins such as GPIO, SHUTDOWN, NRST and IO\_SELECT, first enable the PAD\_Chk\_On bit (or bit-1) of OFN\_ENGINE2 (0x61) register. Then write any value to PAD\_STATUS (0x31) register to start the test. Wait for approximately 12us before reading the actual pin status and PAD\_STATUS register. The test will be considered a PASS to indicate the sensor is responding accordingly if the actual pin status matches PAD\_STATUS register content. Refer to the table below for I/O pin status definition.

For output pins (EVENT\_INT, GPIO, MOSI and MISO) testing, first enable bit-4 of PAD\_FUNCTION (0x34) register. Then program or set the output state via PAD\_TEST\_OUT register (0x33) and do a READ on the actual pin status. Actual pin status results should match the output set in PAD\_TEST\_OUT. (Note: SPI/TWI communication will be disabled after this test is enabled. Once this test is completed, an external hardware reset on sensor is required)

Bit(s)	Name	Reset	Description	Remarks
7:6	NRST_STATE	0x0	0x0: unknown	
			0x1: Low	Invalid as the chip will be in reset state.
			0x2: High	
			0x3: Hi-Z	Indicate a floating high
5:4	SHUTDOWN_STATE	0x0	0x0: unknown	
			0x1: Low	
			0x2: High	Invalid as the chip will be in shutdown state.
			0x3: Hi-Z	Indicate a floating low
3:2	GPIO_STATE	0x0	0x0: unknown	
			0x1: Low	
			0x2: High	
			0x3: Hi-Z	
1:0	IO_SELECT_STATE	0x0	0x0: unknown	
			0x1: Low	
			0x2: High	
			0x3: Hi-Z	

## Fast Video Dump

ADBM-A350 comes with a unique feature that enables user to capture the image the optical sensor is seeing on the tracking surface. This is achieved through storing the pixel data, transferring or dumping the pixels data out to the host for processing and rebuilding the video dump image. The rebuilding of video dump image is mainly converting each 8-bit pixel data to form a grayscale digital image.

Some useful applications for this feature are sensor contamination inspection at manufacturing lines, image recognition, motion sensing applications and etc.

### Fast Video Dump Commands

The host will connect to the OFN sensor and send the following register commands to initiate video dump. The following procedure is outlined below.

1. Sensor power up.
2. Read sensor address 0x00 for correct product ID or PID.
3. Write sensor address 0x3a with 0x5a hex.
4. Provide a 24 MHz clock into GPIO pin at VDDIO level.
5. Write sensor address 0x28 with 0x01 hex.
6. Sensor will start to video dump with frame start for ~500 ms.
7. Write sensor address 0x3a with 0x5a hex to perform soft reset or perform a hard reset to reset sensor back to normal operation.

## Fast Video Dump Setup

There are several signals which need to be tapped. They are namely EVENT\_INT, MISO\_SDA and GPIO pin of the sensor. GPIO will be input while MISO\_SDA and EVENT\_INT are the output pins that the host controller must interface to the sensor, readout the data and then build the video dump image.

Pin	Status	Description
GPIO	Input	24MHz clock signal
MISO_SDA	Output	Pixel data bits 7 to 4
EVENT_INT	Output	Pixel data bits 3 to 0

Referenced to Figure 10 above, the sensor will output signals based on the 24 MHz clock signal via the GPIO pin. Executing the sequence of register setting outlined in Video Dump Commands above will initiate video dump. After step #6, each MISO\_SDA pin and EVENT\_INT pin will go high for 4 clock cycles. This indicates pixel frame start. Then after 4 clock cycles, MISO\_SDA pin will go low and EVENT\_INT pin is still high. This indicates pixel begin state.

The pixel bits will start to clock thereafter. Each pixel is represented by an 8-bit data or value. MISO\_SDA pin will output the most significant bits D7 to D4 and Motion pin will output the least significant bits D3 to D0. Then a pixel end state (a high in MISO\_SDA and low in EVENT\_INT) will follow. This completes the first pixel address 0 data. The next pixel address 1 data is then clocked out with a pixel begin state, 4 clock cycles of data followed by a pixel end state in the same manner. This will continue until all the 361 pixels (19x19 pixel array) data is read.

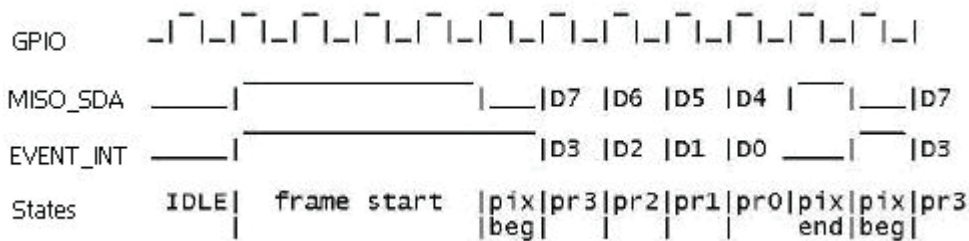


Figure 4. Output pin status and Frame Start

### Pixel byte format

Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1
D7	D6	D5	D4	D3	D2	D1	D0

Below is the pixel array address map. The figure shows the view of the chip from the top of the OFN aperture. Rows are read from top to bottom and columns from left to right.

The signal to indicate the end of the video dump frame is shown in Figure 10 below where pix end state is followed by a low in both MISO\_SDA and EVENT\_INT. This will inform the Host of the pixel end and frame end state.

Pin 1

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37
38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56
57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75
76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94
95	96	97	98	99	100	101	102	103	104	105	106	107	108	109	110	111	112	113
114	115	116	117	118	119	120	121	122	123	124	125	126	127	128	129	130	131	132
133	134	135	136	137	138	139	140	141	142	143	144	145	146	147	148	149	150	151
152	153	154	155	156	157	158	159	160	161	162	163	164	165	166	167	168	169	170
171	172	173	174	175	176	177	178	179	180	181	182	183	184	185	186	187	188	189
190	191	192	193	194	195	196	197	198	199	200	201	202	203	204	205	206	207	208
209	210	211	212	213	214	215	216	217	218	219	220	221	222	223	224	225	226	227
228	229	230	231	232	233	234	235	236	237	238	239	240	241	242	243	244	245	246
247	248	249	250	251	252	253	254	255	256	257	258	259	260	261	262	263	264	265
266	267	268	269	270	271	272	273	274	275	276	277	278	279	280	281	282	283	284
285	286	287	288	289	290	291	292	293	294	295	296	297	298	299	300	301	302	303
304	305	306	307	308	309	310	311	312	313	314	315	316	317	318	319	320	321	322
323	324	325	326	327	328	329	330	331	332	333	334	335	336	337	338	339	340	341
342	343	344	345	346	347	348	349	350	351	352	353	354	355	356	357	358	359	360

Pin 18

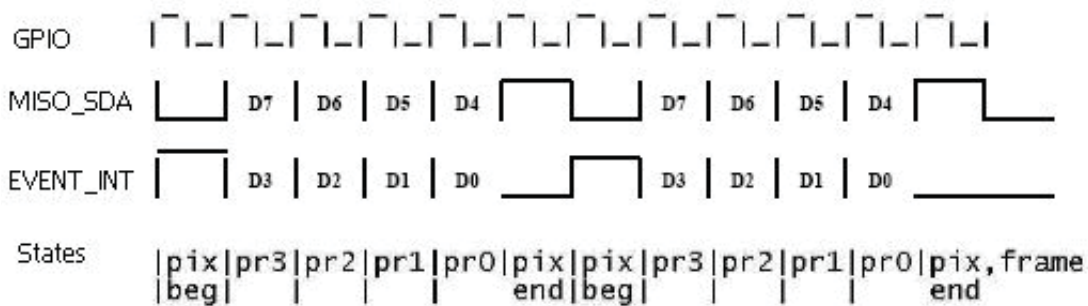


Figure 5. Frame End

Note: It is advisable to have a counter at the Host to keep count of the number of pixel addresses and data byte. If the number of pixel data byte does not correspond to the sensor number of pixel then this video dump data is invalid. In this case the sensor must be reset and a new video dump must be initiated.

## 4-wire Serial Peripheral Interface (SPI)

### SPI Specifications

Electrical Characteristics over recommended operating conditions. Typical values at 25° C, VDD = 1.8 V.

Parameter	Symbol	Minimum	Typical	Maximum	Units	Notes
Serial Port Clock Frequency	$f_{\text{sclk}}$			1	MHz	Active drive, 50% duty cycle
MISO rise time	$t_{\text{r-MISO}}$		150	300	ns	$C_L = 100 \text{ pF}$
MISO fall time	$t_{\text{f-MISO}}$		150	300	ns	$C_L = 100 \text{ pF}$
MISO delay after SCLK	$t_{\text{DLY\_MISO}}$			120	ns	From SCLK falling edge to MISO data valid, no load conditions
MISO hold time	$t_{\text{hold\_MISO}}$	0.5		$1/f_{\text{SCLK}}$	$\mu\text{s}$	Data held until next falling SCLK edge
MOSI hold time	$t_{\text{hold\_MOSI}}$	200			ns	Amount of time data is valid after SCLK rising edge
MOSI setup time	$t_{\text{setup\_MOSI}}$	120			ns	From data valid to SCLK rising edge
SPI time between write commands	$t_{\text{SWW}}$	30			$\mu\text{s}$	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	$t_{\text{SWR}}$	20			$\mu\text{s}$	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	$t_{\text{SRW}}$ $t_{\text{SRR}}$	500			ns	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	$t_{\text{SRAD}}$	4			$\mu\text{s}$	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	$t_{\text{BEXIT}}$	500			ns	Minimum NCS inactive time after motion burst before next SPI usage
NCS to SCLK active	$t_{\text{NCS-SCLK}}$	120			ns	From NCS falling edge to first SCLK falling edge
SCLK to NCS inactive (for read operation)	$t_{\text{SCLK-NCS}}$	120			ns	From last SCLK rising edge to NCS rising edge, for valid MISO data transfer
SCLK to NCS inactive (for write operation)	$t_{\text{SCLK-NCS}}$	20			us	From last SCLK rising edge to NCS rising edge, for valid MOSI data transfer
NCS to MISO high-Z	$t_{\text{NCS-MISO}}$			500	ns	From NCS rising edge to MISO high-Z state

## Synchronous Serial Port

The synchronous serial port is used to set and read parameters in the ADBM-A350, and to read out the motion information.

The port is a four wire serial port. The host micro-controller always initiates communication; the ADBM-A350 never initiates data transfers. SCLK, MOSI, and NCS may be driven directly by a micro-controller. The port pins may be shared with other SPI slave devices. When the NCS pin is high, the inputs are ignored and the output is tri-stated.

The lines that comprise the 4-wire SPI port:

**SCLK:** Clock input. It is always generated by the master (the micro-controller).

**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.

## 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. This is true for all transactions. After a transaction is aborted, the normal address-to-data or transaction-to-transaction delay is still required before beginning the next transaction. 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 ESD events 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 to terminate burst-mode. The port is not available for further use until burst-mode is terminated.

## Write Operation

Write operation, defined as data going from the micro-controller to the ADBM-A350, 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 ADBM-A350 reads MOSI on rising edges of SCLK.

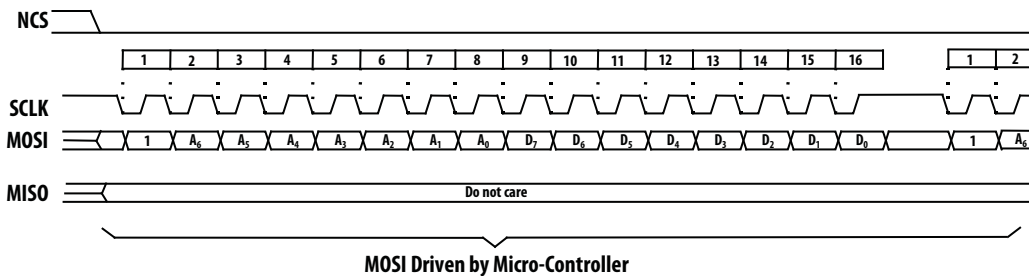


Figure 6. Write Operation

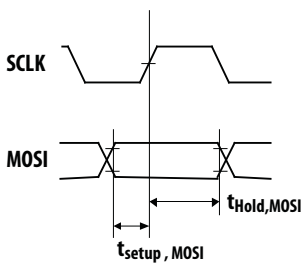


Figure 7. MOSI Setup and Hold Time

## Read Operation

A read operation, defined as data going from the ADBM-A350 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 the ADBM-A350 over MISO. The sensor outputs MISO bits on falling edges of SCLK and samples MOSI bits on every rising edge of SCLK.

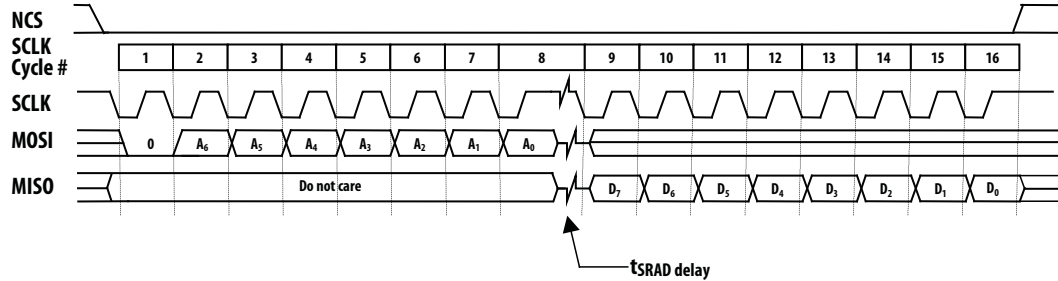


Figure 8. Read Operation

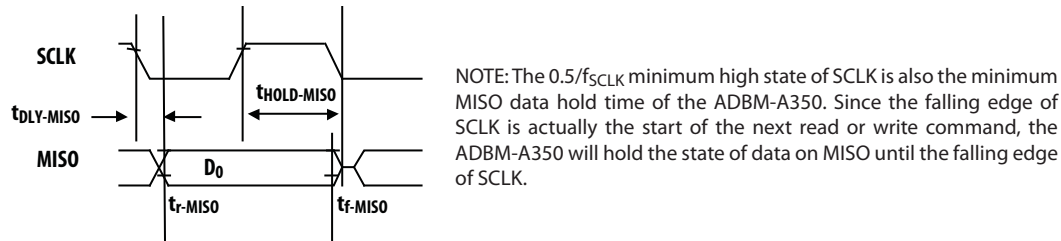


Figure 9. MISO Delay and Hold Time

## Required timing between Read and Write Commands

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

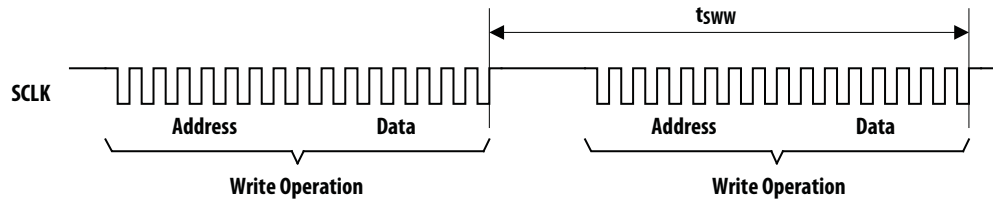


Figure 10. Timing between two write commands

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

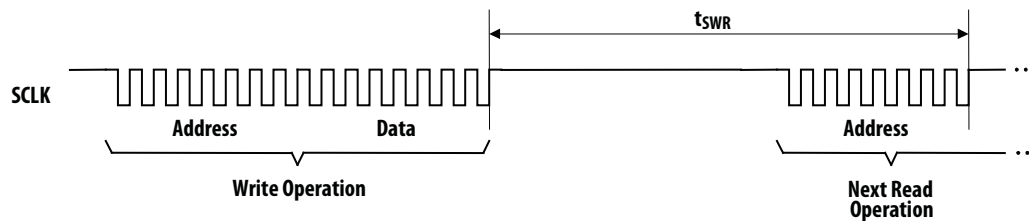


Figure 11. Timing between write and read commands

If the rising edge of SCLK for the last address bit of the read command occurs before the required delay ( $t_{SRW}$ ), the write command may not complete correctly.

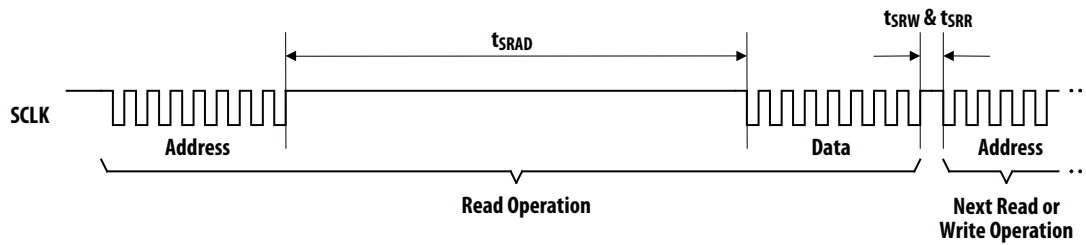


Figure 12. Timing between read and either write or subsequent read commands

During a read operation SCLK should be delayed at least  $t_{SRAD}$  after the last address data bit to ensure that the ADBM-A350 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.

### Burst Mode Operation

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

Burst mode is activated by writing 0x10 to register 0x1c IO\_MODE. Then the burst mode data can be read by reading the Motion register 0x02. The ADBM-A350 will respond with the contents of the Motion, Delta\_Y, Delta\_X, SQUAL, Shutter\_Upper, Shutter\_Lower and Maximum\_Pixel registers in that order. The burst transaction can be terminated after the first 3 bytes of the sequence are read by bringing the NCS pin high. After sending the register address, the micro-controller must wait  $t_{SRAD}$  and then begin reading data. All data bits can be read with no delay between bytes by driving SCLK at the normal rate. The data is latched into the output buffer after the last address bit is received. After the burst transmission is complete, the micro-controller 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.

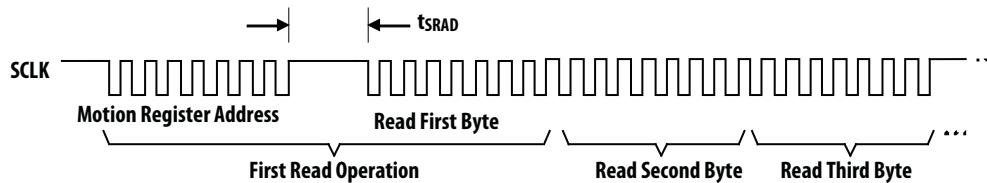


Figure 13. Motion Burst Timing

## Two – Wire Interface (TWI)

ADB350 uses a two-wire serial control interface compatible with I2C. The parameters are listed below.

### TWI Specifications

Electrical Characteristics over recommended operating conditions. Typical values at 25° C, VDD = 1.8 V.

Parameter	Symbol	Minimum	Maximum	Units	Notes
SCL clock frequency	f <sub>SCL</sub>		400	kHz	
Hold time (repeated) START condition. After this period, the first clock pulse is generated	t <sub>HD_STA</sub>	0.6	–	µs	
LOW period of the SCL clock	t <sub>LOW</sub>	1.0	–	µs	
HIGH period of the SCL clock	t <sub>HIGH</sub>	0.6	–	µs	
Set up time for a repeated START condition	t <sub>SU_STA</sub>	0.6	–	µs	
Data hold time	t <sub>HD_DAT</sub>	0 <sup>(2)</sup>	0.9 <sup>(3)</sup>	µs	
Data set-up time	t <sub>SU_DAT</sub>	100	–	ns	
Rise time of both SDA and SCL signals	t <sub>r</sub>	20+0.1C <sub>b</sub> <sup>(4)</sup>	300	ns	
Fall time of both SDA and SCL signals	t <sub>f</sub>	20+0.1C <sub>b</sub> <sup>(4)</sup>	300	ns	
Set up time for STOP condition	t <sub>SU_STO</sub>	0.6	–	µs	
Bus free time between a STOP and START condition	t <sub>BUF</sub>	1.3	–	µs	
Capacitive load for each bus line	C <sub>b</sub>	–	400	pF	
Noise margin at the LOW level for each connected device (including hysteresis)	V <sub>NL</sub>	0.1 VDD	–	V	
Noise margin at the HIGH level for each connected device (including hysteresis)	V <sub>NH</sub>	0.2 VDD	–	V	

Notes:

1. All values referred to V<sub>IHMIN</sub> and V<sub>ILMAX</sub> levels.
2. A device must internally provide a hold time of at least 300 ns for the SDA signal (referred to the V<sub>IHMIN</sub> of the SCL signal) to bridge the undefined region of the falling edge of SCL.
3. The maximum has t<sub>HD\_DAT</sub> only to be met if the device does not stretch the LOW period (t<sub>LOW</sub>) of the SCL signal.
4. C<sub>B</sub> = total capacitance of one bus line in pF.

The ADB350 responds to one of the following selectable slave device addresses depending on the IO\_MOSI\_A0 and IO\_NCS\_A1 input pin state. These pins should be set to avoid conflict with any other devices that might be sharing the bus.

**Table 1. TWI slave address**

A0	A1	Slave Address (Hex)
0	0	33
0	1	3b
1	0	53
1	1	57

### Serial Transfer Clock and Serial Data signals

The serial control interface uses two signals: a serial transfer clock (SCL) signal and a serial data (SDA) signal. Always driven by the master, SCL synchronizes the serial transmission of data bits on SDA. The frequency of SCL may vary throughout a transfer, as long as the timing is greater than the minimum timing.

SDA is bi-directional. The host (master) can read from or write to the ADB350. The host (typically a microcontroller) drives SCL and SDA in a write operation or requesting information from the ADB350. The ADB350 drives the SDA only under two conditions. First, when responding with an acknowledge (ACK) bit after receiving data from the host, or second, when sending data to the host at the host's request. Data is sent in Eight-bit packets.



## Start and Stop of Synchronous Operation

The host initiates and terminates all data transfers. Data transfers are initiated by driving SDA from high to low while holding SCL high. Data transfers are terminated by driving SDA from low to high while SCL is held high.

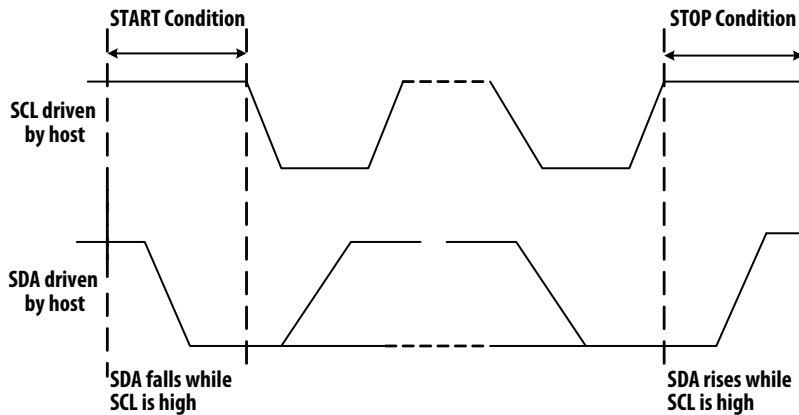


Figure 14. TWI Start and Stop operation

## Acknowledge/Not Acknowledge Bit

After a start condition, a single acknowledge/not acknowledge bit follows each Eight-bit data packet. The device receiving the data drives the acknowledge/not acknowledge signal on SDA. Acknowledge (ACK) is defined as 0 and not acknowledge (NAK) is defined as 1.

## Packet Formats

Read and write operations between the host and the ADBM-A350 use three types of host driven packets and one type of ADBM-A350 driven packet. All packets are eight bits long with the most significant bit first, followed by an acknowledge bit.

## Slave Device Address (DA)

Command packets contain a 7-bit ADBM-A350 device address and an active low read/write bit (R/W).

First bit of packet							Last bit of packet
Device Address							R/W
DA[6]	DA[5]	DA[4]	DA[3]	DA[2]	DA[1]	DA[0]	Write = 0 Read = 1

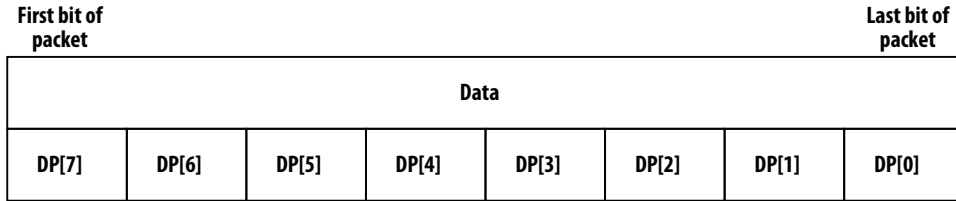
## Register Address Packets (RA)

The address packets contain an auto-increment (ai) bit and a 7-bit address. If the 'ai' bit is set, the slave will process data from successive addresses in successive bytes. For example, registers 0x01, 0x02, and 0x03 can be written by setting the 'ai' bit to one with address 0x01. The host would send three bytes of data, and the host would terminate with a P condition.

First bit of packet	Last bit of packet						
Auto increment	Register Address						
Auto increment=1, No increment=0	RA[6]	RA[5]	RA[4]	RA[3]	RA[2]	RA[1]	RA[0]

## Data Packet (DP)

Contains 8 data bits and may be sent by the host or the ADBM-A350.



## Host Driven Packets

The host initiates all data transmission with a START condition. Next, slave address and register address packets are sent. If there is a device address match, the ADBM-A350 then responds to each Eight-bit data transmission with an acknowledge signal (SDA = 0). Data is transmitted with the most significant bit first.

To terminate the transfer of host driven packets, the host follows the ADBM-A350's ACK with a STOP condition. The host can also issue a START condition after the ADBM-A350's ACK if it wants to start a new data transfer.

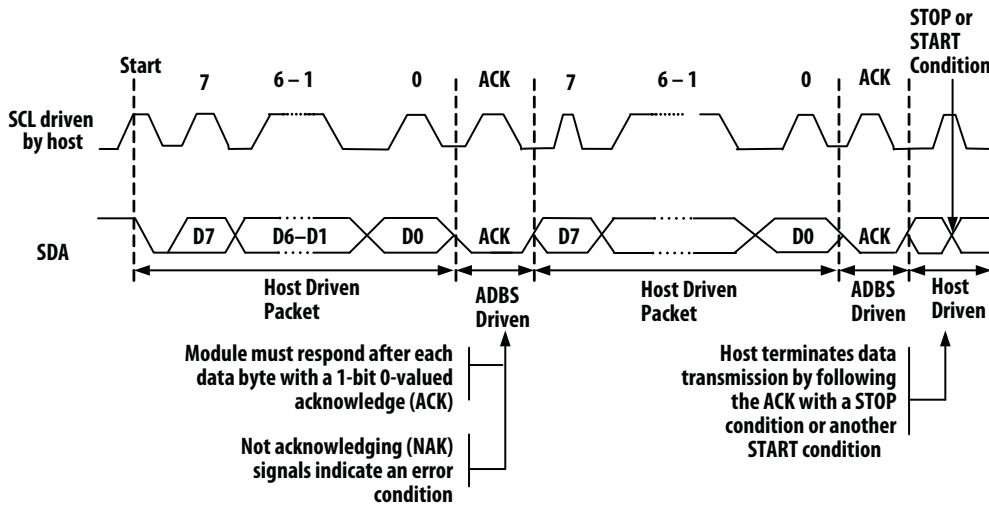


Figure 15. Host packets

## ADBM-A350 Driven Packets

By request of the host, the ADBM-A350 acknowledges a read request and then outputs a data byte transmitting the most significant bit (7) first. If the host intends to continue the data transfer, the host acknowledges the ADBM-A350. If the host intends to terminate the transfer, it responds with not acknowledge (SDA = 1), and then drives SDA to generate a STOP condition. The host can also drive a START condition if it wants to begin a new data transfer with the same ADBM-A350.

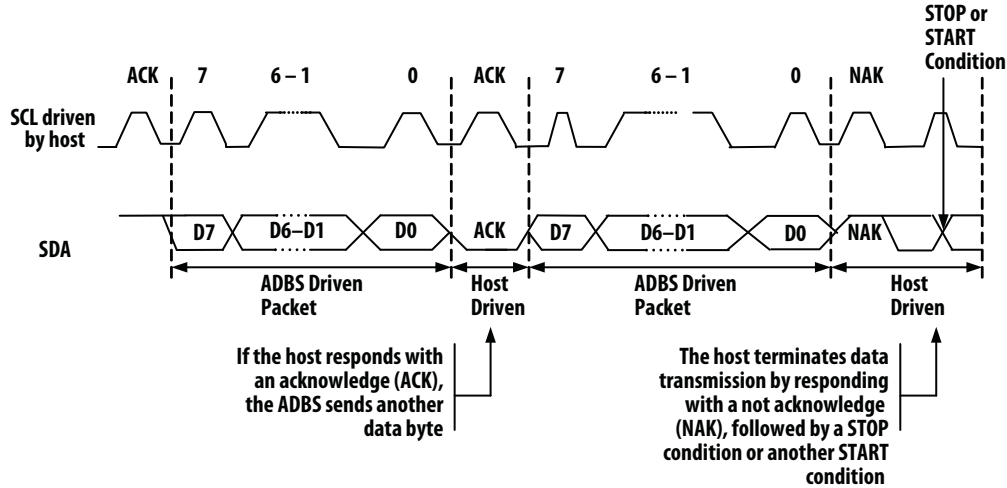


Figure 16. Sensor packets

## Example: Writing Data to Sensor Registers

The host writes a value of 0x02 to address 0x07 in the following illustration.

The example ADBM-A350 address is 0x57.

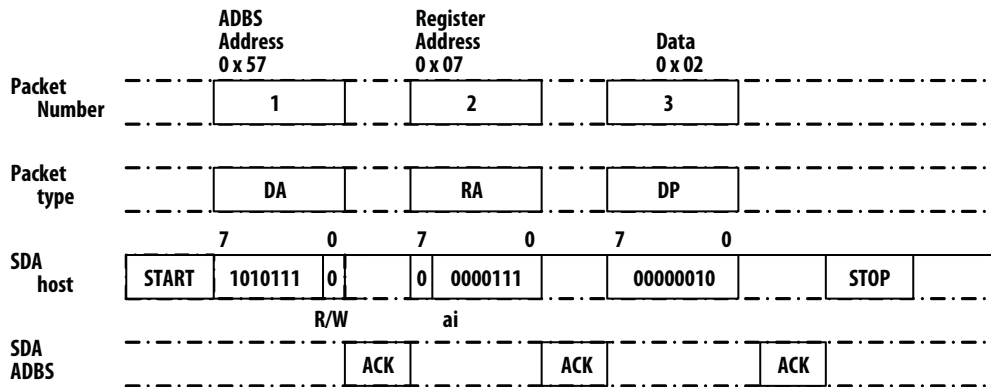


Figure 17. TWI write

### Example: Single Byte Read from Sensor Register

The sensor reads a value 0x01 from the register address 0x02 in the following illustration. Again, the example ADBM-A350 address is 0x57.

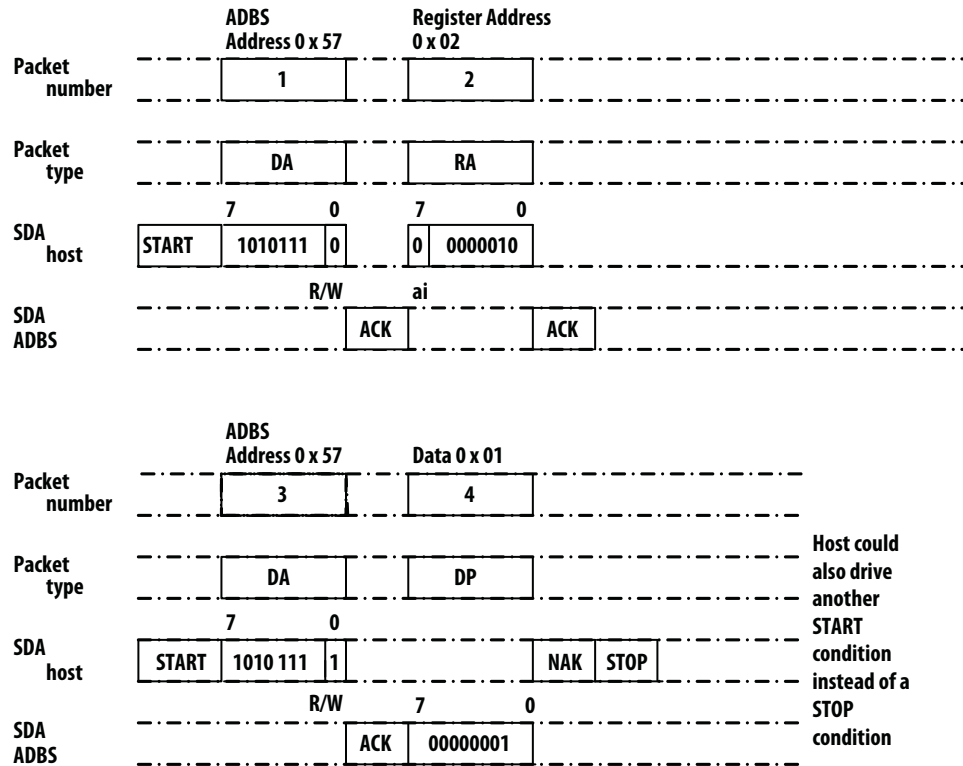


Figure 18. TWI single byte read

### Example: Polling of Status register (X-Y Motion Bit and Button bits)

To poll the STATUS register, the following structure can be used:

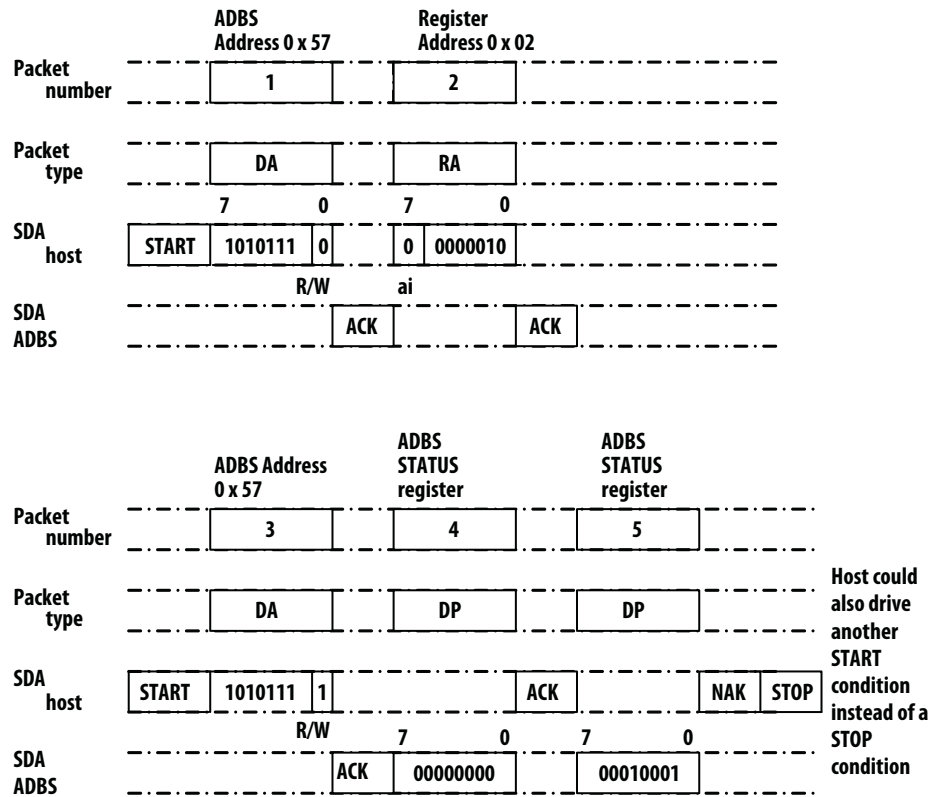


Figure 19. TWI polling

In this case, the host read ADBM-A350 data packets until the update bit (bit 4). Then the host could read successive registers using the ai bit example below.

Note: polling the Status register rather than using the DATA\_RDY pin increases power consumption

### Example: Multiple-Byte Read from Sensor Register using 'ai' bit

The ai is a useful feature, especially in the case of reading Delta\_X, Delta\_Y, and Delta\_HI in succession once either the DATA\_RDY interrupt pin and/or update bit in the STATUS register bit are set.

Once the ai bit is set, the slave will deliver data packets from successive addresses until the 'STOP' condition from the host.

In the example below, 3 bytes are read successively from registers 0x03, 0x04, and 0x05.

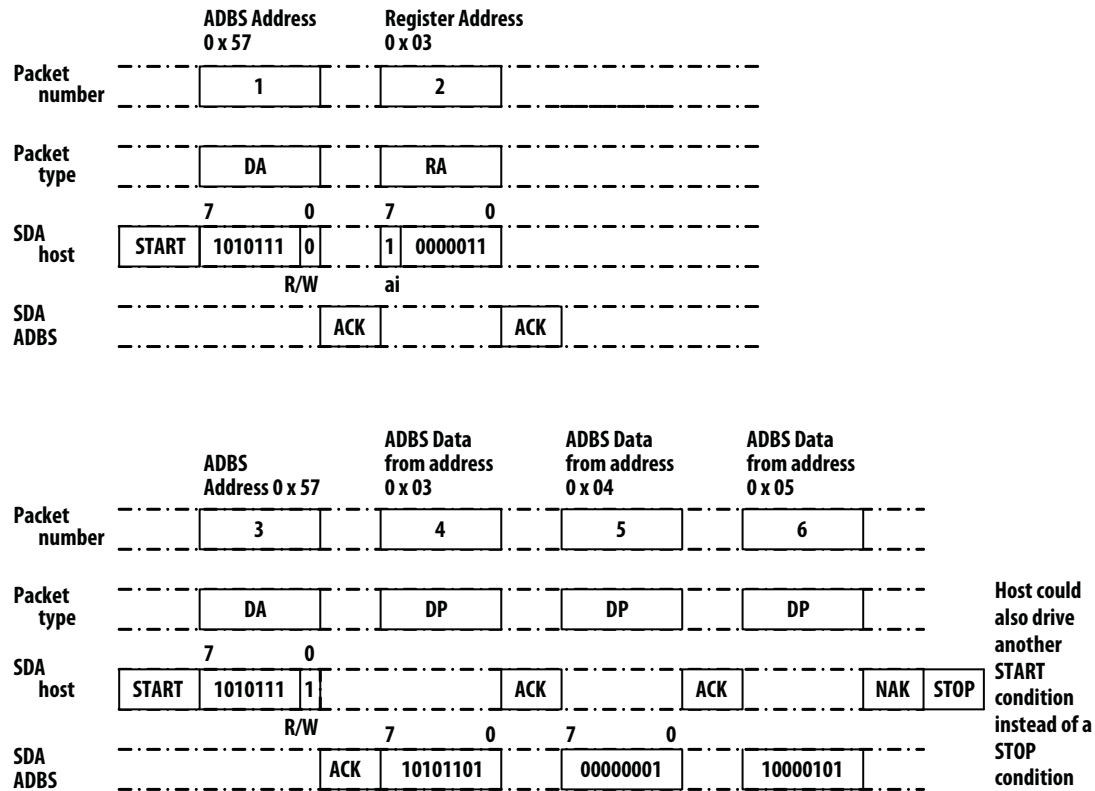


Figure 20. TWI ai bit

## SCL and SDA Timing

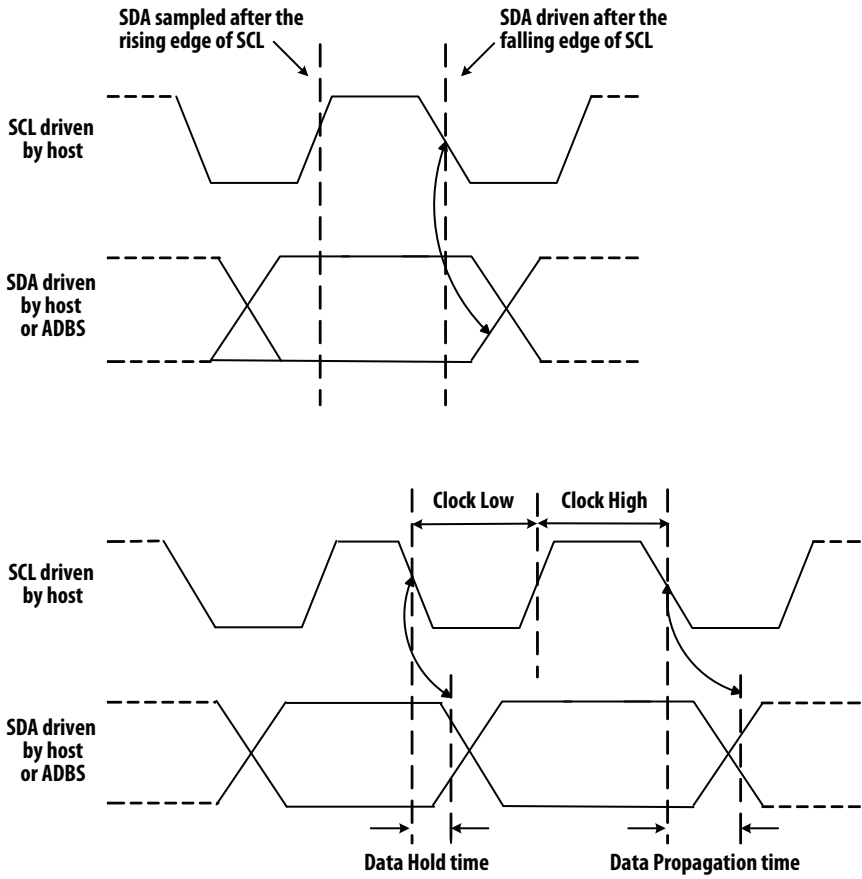


Figure 21. TWI SCL and SDA Timing

## ADBM-A350 driven SDA

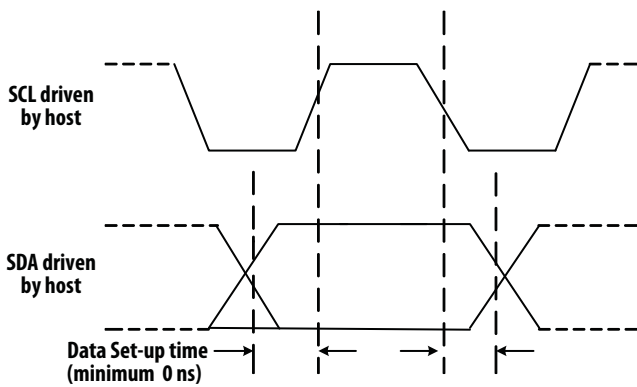


Figure 22. Sensor driven SDA

## Registers

The ADBM-A350 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.

Address	Register	Read/ Write	Default Value	Address	Register	Read/ Write	Default Value
0x00	Product_ID	R	0x88	0x40-0x5f	Reserved		
0x01	Revision_ID	R	0x00	0x60	OFN_Engine1	R/W	0x84
0x02	EVENT	R/W	Any	0x61	OFN_Engine2	R/W	0x89
0x03	Delta_X	R	Any	0x62	Resolution	R/W	0x22
0x04	Delta_Y	R	Any	0x63	Speed_Ctrl	R/W	0x0e
0x05	SQUAL	R	Any	0x64	Speed_ST12	R/W	0x08
0x06	Shutter_Upper	R	Any	0x65	Speed_ST21	R/W	0x06
0x07	Shutter_Lower	R	Any	0x66	Speed_ST23	R/W	0x40
0x08	Maximum_Pixel	R	Any	0x67	Speed_ST32	R/W	0x08
0x09	Pixel_Sum	R	Any	0x68	Speed_ST34	R/W	0x48
0x0a	Minimum_Pixel	R	Any	0x69	Speed_ST43	R/W	0x0a
0x0b	Pixel_Grab	R/W	Any	0x6a	Speed_ST45	R/W	0x50
0x0c	CRC0	R	0x00	0x6b	Speed_ST54	R/W	0x48
0x0d	CRC1	R	0x00	0x6c	GPIO_CTRL	R/W	0x80
0x0e	CRC2	R	0x00	0x6d	AD_CTRL	R/W	0xc4
0x0f	CRC3	R	0x00	0x6e	AD_ATH_HIGH	R/W	0x3a
0x10	Self_Test	W	0x00	0x6f	AD_DTH_HIGH	R/W	0x40
0x11	Reserved			0x70	AD_ATH_LOW	R/W	0x35
0x12	BUTTON_STATUS	R/W	0x00	0x71	AD_DTH_LOW	R/W	0x3b
0x13	Run_Downshift	R/W	0x04	0x72	QUANTIZE_CTRL	R/W	0x99
0x14	Rest1_Period	R/W	0x01	0x73	XYQ_THRESH	R/W	0x02
0x15	Rest1_Downshift	R/W	0x1f	0x74	MOTION_CTRL	R/W	0x00
0x16	Rest2_Period	R/W	0x09	0x75	FPD_CTRL	R/W	0xfa
0x17	Rest2_Downshift	R/W	0x2f	0x76	FPD_THRESH	R/W	0x2c
0x18	Rest3_Period	R/W	0x31	0x77	ORIENT_CTRL	R/W	0x00
0x19	Reserved			0x78	FPD_SQUAL_THRESH	R/W	0x40
0x1a	LED_CTRL	R/W	0x00	0x79	FPD_VALUE	R/W	0x00
0x1b	Reserved			0x7a	FPD_STATUS	R	0x20
0x1c	IO_Mode	R/W	0x00	0x7b	SC_CTRL	R/W	0x25
0x1d	EVENT_CTRL	R/W	0x04	0x7c	SC_T_TAPNHOLD	R/W	0x45
0x28	Fast_Video_Dump	R/W	0x00	0x7d	SC_T_DOUBLE	R/W	0x1e
0x2e	Observation	R/W	Any	0x7e	SC_DELTA_THRESH	R/W	0x19
0x31	Pad_Status	R	0x00	0x7f	SC_STATUS	R/W	0x00
0x32	Reserved						
0x33	Pad_Test_Out	RW	0x00				
0x34	Pad_Function	W	0x00				
0x3a	SOFT_RESET	W	0x00				
0x3b	Shutter_Max_Hi	R/W	0x0b				
0x3c	Shutter_Max_Lo	R/W	0x71				
0x3d	Reserved						
0x3e	Inverse_Revision_ID	R	0xFF				
0x3f	Inverse_Product_ID	R	0x77				



---

**Product\_ID** Address: 0x00  
Access: Read Reset Value: 0x88

---

<b>Bit</b>	7	6	5	4	3	2	1	0
<b>Field</b>	PID <sub>7</sub>	PID <sub>6</sub>	PID <sub>5</sub>	PID <sub>4</sub>	PID <sub>3</sub>	PID <sub>2</sub>	PID <sub>1</sub>	PID <sub>0</sub>

---

Data Type: 8-Bit unsigned integer.

USAGE: This register contains a unique identification assigned to the ADBM-A350. The value in this register does not change; it can be used to verify that the serial communications link is functional.

---

**Revision\_ID** Address: 0x01  
Access: Read Reset Value: 0x00

---

<b>Bit</b>	7	6	5	4	3	2	1	0
<b>Field</b>	RID <sub>7</sub>	RID <sub>6</sub>	RID <sub>5</sub>	RID <sub>4</sub>	RID <sub>3</sub>	RID <sub>2</sub>	RID <sub>1</sub>	RID <sub>0</sub>

---

Data Type: 8-Bit unsigned integer.

USAGE: This register contains the IC revision. It is subject to change when new IC versions are released.

<b>EVENT</b>								
Access: Read/Write	Address: 0x02 Reset Value: Any							
<b>Bit</b>	7	6	5	4	3	2	1	0
<b>Field</b>	MOT	PIXRDY	PIXFIRST	OVFL	RESET_ST	BUT_CLICK	SOFT_CLICK	FPD

Data Type: Bit field.

USAGE: Event detect register (0x02) allows user to determine if any event interrupts (FPD, Motion, Soft click or Button click) has occurred since the last time it was read. If the MOT bit is set, then the user should read registers 0x03 and 0x04 to get the accumulated motion. Read this register before reading the Delta\_Y and Delta\_X registers. Writing anything to this register clears the MOT and OVFL bits, Delta\_Y and Delta\_X registers. The written data byte is not saved.

Internal buffers can accumulate more than eight bits of motion for X or Y. If any of the internal buffers overflow, then absolute path data is lost and the OVFL bit is set. This bit is cleared once some motion has been read from the Delta\_X and Delta\_Y registers, and if the buffers are not at full scale. Since more data is present in the buffers, the cycle of reading the Event, Delta\_X and Delta\_Y registers should be repeated until the motion bit (MOT) is cleared. If the Event register has not been read for long time, at 500 cpi it may take up to 16 read cycles to clear the buffers, at 1000 cpi, up to 32 cycles. To clear an overflow, write anything to this register.

The PIXRDY bit will be set whenever a valid pixel data byte is ready and available in the Pixel\_Dump register. Check that this bit is set before reading from Pixel\_Dump. To ensure that the Pixel\_Grab pointer has been reset to pixel 0,0 on the initial write to Pixel\_Grab, check to see if PIXFIRST is set to high.

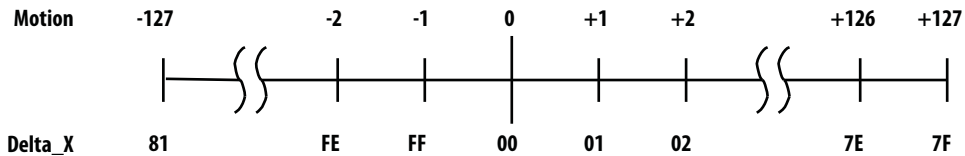
Field Name	Description
MOT	Motion since last report <b>0 = No motion</b> 1 = Motion occurred, data ready for reading in Delta_X and Delta_Y registers Bit reset when motion data in Delta_X and Delta_Y registers are cleared
PIXRDY	Pixel Dump data byte is available in Pixel_Dump register <b>0 = data not available</b> 1 = data available
PIXFIRST	This bit is set when the Pixel_Grab register is written to or when the complete pixel array has been read, initiating an increment to pixel 0,0. <b>0 = Pixel_Grab data not from pixel 0,0</b> 1 = Pixel_Grab data is from pixel 0,0
OVF	Motion overflow, ΔY and/or ΔX buffer has overflowed since last report <b>0 = no overflow</b> 1 = Overflow has occurred
RESET_ST	Reset status bit. Any internal or external reset will set this bit. Write anything to this register to clear it. 0 = No reset <b>1 = Reset occurred</b>
BUT_CLICK	Button click report <b>0 = No Button click</b> 1 = Button click occurred Bit clear or reset by reading BUTTON_STATUS (0x12) register
SOFT_CLICK	Soft click report <b>0 = No Soft click</b> 1 = Soft click occurred Bit clear or reset by reading SC_STATUS (0x7F) register
FPD	Finger presence detect bit reports a change in finger state (finger on to finger off and vice versa) <b>0 = No finger state change detected</b> 1 = Finger state change detected Bit clear or reset by reading FPD_STATUS (0x7A) register

**Delta\_X** Address: 0x03  
Access: Read Reset Value: Any

<b>Bit</b>	7	6	5	4	3	2	1	0
<b>Field</b>	Y <sub>7</sub>	Y <sub>6</sub>	Y <sub>5</sub>	Y <sub>4</sub>	Y <sub>3</sub>	Y <sub>2</sub>	Y <sub>1</sub>	Y <sub>0</sub>

Data Type: Eight bit 2's complement number.

USAGE: X movement is counts since last report. Absolute value is determined by resolution. Reading clears the register.



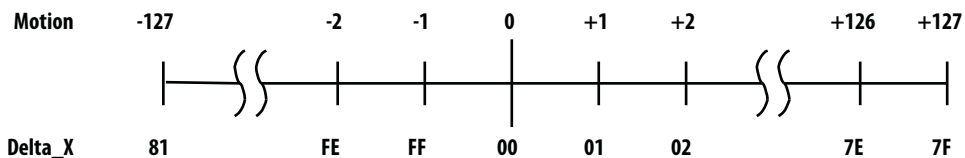
NOTES: Avago RECOMMENDS that registers 0x03 and 0x04 be read sequentially.

**Delta\_Y** Address: 0x04  
Access: Read Reset Value: Any

<b>Bit</b>	7	6	5	4	3	2	1	0
<b>Field</b>	X <sub>7</sub>	X <sub>6</sub>	X <sub>5</sub>	X <sub>4</sub>	X <sub>3</sub>	X <sub>2</sub>	X <sub>1</sub>	X <sub>0</sub>

Data Type: Eight bit 2's complement number.

USAGE: Y movement is counts since last report. Absolute value is determined by resolution. Reading clears the register.



NOTES: Avago RECOMMENDS that registers 0x03 and 0x04 be read sequentially.

**SQUAL** Address: 0x05  
Access: Read Reset Value: Any

<b>Bit</b>	7	6	5	4	3	2	1	0
<b>Field</b>	SQ <sub>7</sub>	SQ <sub>6</sub>	SQ <sub>5</sub>	SQ <sub>4</sub>	SQ <sub>3</sub>	SQ <sub>2</sub>	SQ <sub>1</sub>	SQ <sub>0</sub>

Data Type: Upper 8 bits of a 9-bit unsigned integer.

USAGE: SQUAL (Surface Quality) is a measure of the number of valid features visible by the sensor in the current frame. The maximum SQUAL register value is 167. Since small changes in the current frame can result in changes in SQUAL, variations in SQUAL when looking at a surface are expected.

<b>Shutter_Upper</b>			Address: 0x06					
Access: Read			Reset Value: Any					
<b>Bit</b>	7	6	5	4	3	2	1	0
<b>Field</b>	S <sub>15</sub>	S <sub>14</sub>	S <sub>13</sub>	S <sub>12</sub>	S <sub>11</sub>	S <sub>10</sub>	S <sub>9</sub>	S <sub>8</sub>

<b>Shutter_Lower</b>			Address: 0x07					
Access: Read			Reset Value: Undefined					
<b>Bit</b>	7	6	5	4	3	2	1	0
<b>Field</b>	S <sub>7</sub>	S <sub>6</sub>	S <sub>5</sub>	S <sub>4</sub>	S <sub>3</sub>	S <sub>2</sub>	S <sub>1</sub>	S <sub>0</sub>

Data Type: Sixteen bit unsigned integer.

USAGE: Units are clock cycles. Read Shutter\_Upper first, then Shutter\_Lower. They should be read consecutively. The shutter is adjusted to keep the average and maximum pixel values within normal operating ranges. The shutter value is automatically adjusted.

<b>Maximum_Pixel</b>			Address: 0x08					
Access: Read			Reset Value: Any					
<b>Bit</b>	7	6	5	4	3	2	1	0
<b>Field</b>	MP <sub>7</sub>	MP <sub>6</sub>	MP <sub>5</sub>	MP <sub>4</sub>	MP <sub>3</sub>	MP <sub>2</sub>	MP <sub>1</sub>	MP <sub>0</sub>

Data Type: Eight-bit number.

USAGE: Maximum Pixel value in current frame. Minimum value = 0, maximum value = 254. The maximum pixel value can vary with every frame.

<b>Pixel_Sum</b>			Address: 0x09					
Access: Read			Reset Value: Any					
<b>Bit</b>	7	6	5	4	3	2	1	0
<b>Field</b>	AP <sub>7</sub>	AP <sub>6</sub>	AP <sub>5</sub>	AP <sub>4</sub>	AP <sub>3</sub>	AP <sub>2</sub>	AP <sub>1</sub>	AP <sub>0</sub>

Data Type: High 8 bits of an unsigned 17-bit integer.

USAGE: This register is used to find the average pixel value. It reports the seven bits of a 16-bit counter, which sums all pixels in the current frame. It may be described as the full sum divided by 512. To find the average pixel value, use the following formula:

$$\text{Average Pixel} = \text{Register Value} * 128/121 = \text{Register Value} * 1.06$$

The maximum register value is 240. The minimum is 0. The pixel sum value can change every frame.

---

**Minimum\_Pixel**

Access: Read

Address: 0x0a

Reset Value: Any

---

<b>Bit</b>	7	6	5	4	3	2	1	0
<b>Field</b>	MP <sub>7</sub>	MP <sub>6</sub>	MP <sub>5</sub>	MP <sub>4</sub>	MP <sub>3</sub>	MP <sub>2</sub>	MP <sub>1</sub>	MP <sub>0</sub>

---

Data Type: Eight-bit number.

USAGE: Minimum Pixel value in current frame. Minimum value = 0, maximum value = 254. The minimum pixel value can vary with every frame.

---

**Pixel\_Grab**

Access: Read/Write

Address: 0x0b

Reset Value: Any

---

<b>Bit</b>	7	6	5	4	3	2	1	0
<b>Field</b>	PD <sub>7</sub>	PD <sub>6</sub>	PD <sub>5</sub>	PD <sub>4</sub>	PD <sub>3</sub>	PD <sub>2</sub>	PD <sub>1</sub>	PD <sub>0</sub>

---

Data Type: Eight-bit word.

USAGE: For test purposes, the sensor will read out the contents of the pixel array, one pixel per frame. To start a pixel grab, write anything to this register to reset the pointer to pixel 0,0. Then read the PIXRDY bit in the Motion register. When the PIXRDY bit is set, there is valid data in this register to read out. After the data in this register is read, the pointer will automatically increment to the next pixel. Reading may continue indefinitely; once a complete frame's worth of pixels has been read, PIXFIRST will be set to high to indicate the start of the first pixel and the address pointer will start at the beginning location again. The pixel map address and corresponding sensor orientation is shown below.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37
38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56
57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75
76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94
95	96	97	98	99	100	101	102	103	104	105	106	107	108	109	110	111	112	113
114	115	116	117	118	119	120	121	122	123	124	125	126	127	128	129	130	131	132
133	134	135	136	137	138	139	140	141	142	143	144	145	146	147	148	149	150	151
152	153	154	155	156	157	158	159	160	161	162	163	164	165	166	167	168	169	170
171	172	173	174	175	176	177	178	179	180	181	182	183	184	185	186	187	188	189
190	191	192	193	194	195	196	197	198	199	200	201	202	203	204	205	206	207	208
209	210	211	212	213	214	215	216	217	218	219	220	221	222	223	224	225	226	227
228	229	230	231	232	233	234	235	236	237	238	239	240	241	242	243	244	245	246
247	248	249	250	251	252	253	254	255	256	257	258	259	260	261	262	263	264	265
266	267	268	269	270	271	272	273	274	275	276	277	278	279	280	281	282	283	284
285	286	287	288	289	290	291	292	293	294	295	296	297	298	299	300	301	302	303
304	305	306	307	308	309	310	311	312	313	314	315	316	317	318	319	320	321	322
323	324	325	326	327	328	329	330	331	332	333	334	335	336	337	338	339	340	341
342	343	344	345	346	347	348	349	350	351	352	353	354	355	356	357	358	359	360

**Figure 23. Top view of pixel map address without lens**

360	359	358	357	356	355	354	353	352	351	350	349	348	347	346	345	344	343	342
341	340	339	338	337	336	335	334	333	332	331	330	329	328	327	326	325	324	323
322	321	320	319	318	317	316	315	314	313	312	311	310	309	308	307	306	305	304
303	302	301	300	299	298	297	296	295	294	293	292	291	290	289	288	287	286	285
284	283	282	281	280	279	278	277	276	275	274	273	272	271	270	269	268	267	266
265	264	263	262	261	260	259	258	257	256	255	254	253	252	251	250	249	248	247
246	245	244	243	242	241	240	239	238	237	236	235	234	233	232	231	230	229	228
227	226	225	224	223	222	221	220	219	218	217	216	215	214	213	212	211	210	209
208	207	206	205	204	203	202	201	200	199	198	197	196	195	194	193	192	191	190
189	188	187	186	185	184	183	182	181	180	179	178	177	176	175	174	173	172	171
170	169	168	167	166	165	164	163	162	161	160	159	158	157	156	155	154	153	152
151	150	149	148	147	146	145	144	143	142	141	140	139	138	137	136	135	134	133
132	131	130	129	128	127	126	125	124	123	122	121	120	119	118	117	116	115	114
113	112	111	110	109	108	107	106	105	104	103	102	101	100	99	98	97	96	95
94	93	92	91	90	89	88	87	86	85	84	83	82	81	80	79	78	77	76
75	74	73	72	71	70	69	68	67	66	65	64	63	62	61	60	59	58	57
56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38
37	36	35	34	33	32	31	30	29	28	27	26	25	24	23	22	21	20	19
18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

**Figure 24. Top view of pixel map address with lens**

---

**CRC0** Address: 0x0c  
Access: Read Reset Value: 0x00

---

<b>Bit</b>	7	6	5	4	3	2	1	0
<b>Field</b>	CRC0 <sub>7</sub>	CRC0 <sub>6</sub>	CRC0 <sub>5</sub>	CRC0 <sub>4</sub>	CRC0 <sub>3</sub>	CRC0 <sub>2</sub>	CRC0 <sub>1</sub>	CRC0 <sub>0</sub>

---

Data Type: Eight-bit number

USAGE: Register 0x0c reports the first byte of the system self test results. See Self Test register 0x10.

---

**CRC1** Address: 0x0d  
Access: Read Reset Value: 0x00

---

<b>Bit</b>	7	6	5	4	3	2	1	0
<b>Field</b>	CRC1 <sub>7</sub>	CRC1 <sub>6</sub>	CRC1 <sub>5</sub>	CRC1 <sub>4</sub>	CRC1 <sub>3</sub>	CRC1 <sub>2</sub>	CRC1 <sub>1</sub>	CRC1 <sub>0</sub>

---

Data Type: Eight-bit number

USAGE: Register 0x0d reports the second byte of the system self test results. See Self Test register 0x10.

---

**CRC2** Address: 0x0e  
Access: Read Reset Value: 0x00

---

<b>Bit</b>	7	6	5	4	3	2	1	0
<b>Field</b>	CRC2 <sub>7</sub>	CRC2 <sub>6</sub>	CRC2 <sub>5</sub>	CRC2 <sub>4</sub>	CRC2 <sub>3</sub>	CRC2 <sub>2</sub>	CRC2 <sub>1</sub>	CRC2 <sub>0</sub>

---

Data Type: Eight-bit number

USAGE: Register 0x0e reports the third byte of the system self test results. See Self Test register 0x10.

---

**CRC3** Address: 0x0f  
Access: Read Reset Value: 0x00

---

<b>Bit</b>	7	6	5	4	3	2	1	0
<b>Field</b>	CRC3 <sub>7</sub>	CRC3 <sub>6</sub>	CRC3 <sub>5</sub>	CRC3 <sub>4</sub>	CRC3 <sub>3</sub>	CRC3 <sub>2</sub>	CRC3 <sub>1</sub>	CRC3 <sub>0</sub>

---

Data Type: Eight-bit number

USAGE: Register 0x0f reports the fourth byte of the system self test results. See Self Test register 0x10.

---

**Self\_Test** Address: 0x10  
 Access: Write Reset Value: 0x00

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

Data Type: Bit field

USAGE: Set the TESTEN bit in register 0x10 to start the system self-test. The test takes 250ms. During this time, do not write or read through the SPI port. Results are available in the CRC0-3 registers. After self-test, reset the chip to start normal operation.

Field Name	Description
TESTEN	Enable System Self Test <b>0 = Disable</b> 1 = Enable

The procedure to start self test is as follows:-

1. Write data 0x5A to register 0x3A to initiate soft reset. Do not load OFN registers.
2. Write data 0xC1 to register 0x29.
3. Write data 0xFF to register 0x60.
4. Write data 0x88 to register 0x61.
5. Write data 0xAA to register 0x72.
6. Write data 0xC4 to register 0x63
7. Write data 0x20 to register 0x75
8. Write data 0x05 to register 0x76
9. Write data 0x04 to register 0x7C
10. Write data 0xFF to register 0x7E
11. Write data 0x01 to register 0x10 to initiate self test.
12. Wait 250ms.
13. Read CRC0 from address 0x0c, CRC1 from address 0x0d, CRC2 from address 0x0e, CRC3 from address 0x0f.

The results are as follows.

CRC#	
CRC0	0x3E
CRC1	0x5C
CRC2	0xA4
CRC3	0x24



**BUTTON\_STATUS** Address: 0x12  
 Access: Read/Write Reset Value: 0x00

Bit	7	6	5	4	3	2	1	0
Field	Assert count	Assert count	Deassert count	Deassert count	Reserved	Reserved	Reserved	But_Status

Data Type: Bit field

USAGE: Register 0x12 allows the user to read button status. Writing any value to this register will restart the counter.

Field Name	Description
Assert count 7:6	Counts for number of times the button state change from 0 to 1
Deassert count 5:4	Counts for number of times the button state change from 1 to 0
But_Status	Button current status

**Run\_Downshift** Address: 0x13  
 Access: Read/Write Reset Value: 0x04

Bit	7	6	5	4	3	2	1	0
Field	RD <sub>7</sub>	RD <sub>6</sub>	RD <sub>5</sub>	RD <sub>4</sub>	RD <sub>3</sub>	RD <sub>2</sub>	RD <sub>1</sub>	RD <sub>0</sub>

This register set the Run to Rest 1 downshift time.

Run Downshift time =  $RD[7:0] \times 8 \times Run\_rate$ .

Default value:  $4 \times 8 \times 8 \text{ ms} = 256 \text{ ms}$

Min:  $2 \times 8 \times 8 \text{ ms} = 128 \text{ ms}$

Max:  $242 \times 8 \times 8 \text{ ms} = 15,488 \text{ ms} = 15.49 \text{ s}$

All the above values are calculated base on 25 MHz System clock.

**Rest1\_Period** Address: 0x14  
 Access: Read/Write Reset Value: 0x01

Bit	7	6	5	4	3	2	1	0
Field	R1R <sub>7</sub>	R1R <sub>6</sub>	R1R <sub>5</sub>	R1R <sub>4</sub>	R1R <sub>3</sub>	R1R <sub>2</sub>	R1R <sub>1</sub>	R1R <sub>0</sub>

This register set the Rest 1 frame rate.

Rest1 frame rate =  $(R1R[7:0] + 1) \times 10 \text{ ms}$ .

Default value:  $2 \times 10 \text{ ms} = 20 \text{ ms}$

Min:  $2 \times 10 \text{ ms} = 20 \text{ ms}$

Max:  $241 \times 10 \text{ ms} = 2,410 \text{ ms} = 2.41 \text{ s}$

All the above values are calculated base on 100 Hz Hibernate clock.

**Rest1\_Downshift** Address: 0x15  
 Access: Read/Write Reset Value: 0x1f

Bit	7	6	5	4	3	2	1	0
Field	R1D <sub>7</sub>	R1D <sub>6</sub>	R1D <sub>5</sub>	R1D <sub>4</sub>	R1D <sub>3</sub>	R1D <sub>2</sub>	R1D <sub>1</sub>	R1D <sub>0</sub>

This register set the Rest 1 to Rest 2 downshift time.

Rest1 Downshift time = R1D[7:0] x 16 x Rest1\_Rate.

Default value: 31 x 16 x 20 ms (Rest1\_Rate default) = 9,920 ms = 9.92 s

Min: 1 x 16 x 20 ms (Rest1\_Rate min) = 320 ms

Max: 242 x 16 x 2.56 s (Rest1\_Rate max) = 9,912 s = 165 min = 2.75 hr

All the above values are calculated base on 100 Hz Hibernate clock.

**Rest2\_Period** Address: 0x16  
 Access: Read/Write Reset Value: 0x09

Bit	7	6	5	4	3	2	1	0
Field	R2R <sub>7</sub>	R2R <sub>6</sub>	R2R <sub>5</sub>	R2R <sub>4</sub>	R2R <sub>3</sub>	R2R <sub>2</sub>	R2R <sub>1</sub>	R2R <sub>0</sub>

This register set the Rest 2 frame rate.

Rest2 frame rate = (R2R[7:0] + 1) x 10 ms.

Default value: 10 x 10 ms = 100 ms

Min: 2 x 10 ms = 20 ms

Max: 241 x 10 ms = 2,410 ms = 2.41 s

All the above values are calculated base on 100 Hz Hibernate clock.

**Rest2\_Downshift** Address: 0x17  
 Access: Read/Write Reset Value: 0x2f

Bit	7	6	5	4	3	2	1	0
Field	R2D <sub>7</sub>	R2D <sub>6</sub>	R2D <sub>5</sub>	R2D <sub>4</sub>	R2D <sub>3</sub>	R2D <sub>2</sub>	R2D <sub>1</sub>	R2D <sub>0</sub>

This register set the Rest 2 to Rest 3 downshift time.

Rest2 Downshift time = R2D[7:0] x 128 x Rest2\_Rate.

Default value: 47 x 128 x 100 ms (Rest2\_Rate default) = 601.6 s = 10 min

Min: 1 x 128 x 20 ms (Rest2\_Rate min) = 2560 ms = 2.56 s

Max: 242 x 128 x 2.56 s (Rest2\_Rate max) = 79,298 s = 1,321 min = 22 hrs

All the above values are calculated base on 100 Hz Hibernate clock.

**Rest3\_Period** Address: 0x18  
 Access: Read/Write Reset Value: 0x31

<b>Bit</b>	7	6	5	4	3	2	1	0
<b>Field</b>	R3R <sub>7</sub>	R3R <sub>6</sub>	R3R <sub>5</sub>	R3R <sub>4</sub>	R3R <sub>3</sub>	R3R <sub>2</sub>	R3R <sub>1</sub>	R3R <sub>0</sub>

This register set the Rest 3 frame rate.

Rest3 frame rate = (R3R[7:0] + 1) x 10 ms.

Default value: 50 x 10 ms = 500 ms

Min: 2 x 10 ms = 20 ms

Max: 241 x 10 ms = 2,410 ms = 2.41 s

All the above values are calculated base on 100 Hz Hibernate clock.

**Reserved** Address: 0x19

**LED\_Control** Address: 0x1a  
 Access: Read/Write Reset Value: 0x00

<b>Bit</b>	7	6	5	4	3	2	1	0
<b>Field</b>	Reserved	Reserved	Reserved	Reserved	LED_On	LED2	LED1	LED0

Data Type: Bit field

USAGE: Register 0x1a allows the user to change the LED drive current of the sensor.

Field Name	Description
LED_On	<b>0 = LED normal operation</b> 1 = LED always On
LED2:0	<b>0x00 = 13 mA</b> 0x02 = 9.6 mA 0x07 = 27 mA

**Reserved** Address: 0x1b

**IO\_Mode** Address: 0x1c  
 Access: Read/Write Reset Value: 0x00

Bit	7	6	5	4	3	2	1	0
Field	Reserved	Reserved	Reserved	Burst	Reserved	SPI	Reserved	TWI

Data Type: Bit field

USAGE: Register 0x1c allows the user to read the Input or Output mode of the sensor.

Field Name	Description
Burst	Burst mode <b>0 = not in burst mode</b> 1 = In Burst mode
SPI	SPI mode <b>0 = not in SPI mode</b> 1 = In SPI mode
TWI	TWI mode <b>0 = not in TWI mode</b> 1 = In TWI mode

**EVENT\_CTRL** Address: 0x1d  
 Access: Read/Write Reset Value: 0x04

Bit	7	6	5	4	3	2	1	0
Field	Event_ Active_Set	Reserved	Reserved	Reserved	SFCL	FPD	BCL	MOT

Data Type: Bit field

USAGE: Register 0x1d allows the user to configure and control the Event\_Int output pin. By default, FPD interrupt is disabled while all other interrupts enabled.

Field Name	Description
Event_Active_Set	Event_Int pin setting 0 = Active-low Output 1 = Active-high Output
SFCL	1 = Disable Soft click interrupt <b>0 = Enable Soft Click interrupt</b>
FPD	<b>1 = Disable Finger Presence Detection (FPD) interrupt</b> 0 = Enable finger presence detection interrupt
BCL	1 = Disable Button click interrupt <b>0 = Enable Button Click interrupt</b>
MOT	1 = Disable Motion interrupt <b>0 = Enable Motion interrupt</b>

---

**Fast\_Video\_Dump** Address: 0x28  
Access: Read/Write Reset Value: 0x00

---

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

---

Data Type: Bit field

USAGE: This register is used to enable and initiating Fast Video Dump. See note on Fast Video Dump for more information.

---

Field Name	Description
FVD	<b>0: Disable FVD</b> 1: Enable FVD

---

---

**Observation** Address: 0x2e  
Access: Read/Write Reset Value: Any

---

Bit	7	6	5	4	3	2	1	0
Field	MODE <sub>1</sub>	MODE <sub>0</sub>	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved

---

Data Type: Bit field

USAGE: Register 0x2e provides bits that are set every frame. It can be used during ESD testing to check that the chip is running correctly. Writing anything to this register will clear the bits.

---

Field Name	Description
MODE <sub>1-0</sub>	Mode Status: Reports which mode the sensor is in. 00 = Run 01 = Rest1 10 = Rest2 11 = Rest3

---

**PAD\_STATUS**  
Access: Read

Address: 0x31  
Reset Value: 0x00

Bit	7	6	5	4	3	2	1	0
Field	NRST_ST1	NRST_ST0	SHTDWN_ST1	SHTDWN_ST0	GPIO_ST1	GPIO_ST0	IOSEL_ST1	IOSEL_ST0

Data Type: Bit field

USAGE: This register is used to read or check on pin status.

Field Name	Description
NRST_ST1-0	NRST Pin State 0x0 = Unknown 0x1 = Low (Invalid as the chip will be in reset state) 0x2 = High (Indicate a floating high) 0x3 = Hi-Z
SHTDWN_ST1-0	SHTDWN Pin State 0x0 = Unknown 0x1 = Low 0x2 = High (Invalid as chip will be in shutdown state) 0x3 = Hi-Z
GPIO_ST1-0	GPIO Pin State 0x0 = Unknown 0x1 = Low 0x2 = High 0x3 = Hi-Z
IOSEL_ST1-0	IO_SELECT Pin State 0x0 = Unknown 0x1 = Low 0x2 = High 0x3 = Hi-Z

**Reserved**

Address: 0x32

---

**PAD\_TEST\_OUT** Address: 0x33  
Access: Read/Write Reset Value: 0x00

---

<b>Bit</b>	7	6	5	4	3	2	1	0
<b>Field</b>	Reserved	Reserved	Reserved	Reserved	EVENT_O	GPIO_O	MOSI_O	MISO_O

---

Data Type: Bit field

USAGE: This register is used to set the state of the output pin.

---

<b>Field Name</b>	<b>Description</b>
EVENT_O	Set state for EVENT_INT output pin 0x0: Low 0x1: High
GPIO_O	Set state for GPIO_O output pin 0x0: Low 0x1: High
MOSI_O	Set state for MOSI_O output pin 0x0: Low 0x1: High
MISO_O	Set state for MISO_O output pin 0x0: Low 0x1: High

---

---

**PAD\_FUNCTION** Address: 0x34  
Access: Write Reset Value: 0x00

---

<b>Bit</b>	7	6	5	4	3	2	1	0
<b>Field</b>	Reserved	Reserved	Reserved	OUT_TEST_EN	Reserved	Reserved	Reserved	Reserved

---

Data Type: Bit field

USAGE: This register is used to enable output pin testing.

---

<b>Field Name</b>	<b>Description</b>
OUT_TEST_EN	Enable Output Pin Test

---

---

**SOFT\_RESET** Address: 0x3a  
Access: Write Reset Value: 0x00

---

<b>Bit</b>	7	6	5	4	3	2	1	0
<b>Field</b>	RST <sub>7</sub>	RST <sub>6</sub>	RST <sub>5</sub>	RST <sub>4</sub>	RST <sub>3</sub>	RST <sub>2</sub>	RST <sub>1</sub>	RST <sub>0</sub>

---

Data Type: 8-bit integer

USAGE: Write 0x5A to this register to reset the chip. All settings will revert to default values.

---

**Shutter\_Max\_Hi** Address: 0x3b  
Access: Read/Write Reset Value: 0x0b

---

<b>Bit</b>	7	6	5	4	3	2	1	0
<b>Field</b>	SMH <sub>7</sub>	SMH <sub>6</sub>	SMH <sub>5</sub>	SMH <sub>4</sub>	SMH <sub>3</sub>	SMH <sub>2</sub>	SMH <sub>1</sub>	SMH <sub>0</sub>

---

Data Type: 8-Bit integer

USAGE: This value is the upper 8-bit of shutter maximum open time. Shutter value represents pixel array exposure time in multiples of internal clock cycles with maximum value at 2929 decimal.

---

**Shutter\_Max\_Lo** Address: 0x3c  
Access: Read/Write Reset Value: 0x71

---

<b>Bit</b>	7	6	5	4	3	2	1	0
<b>Field</b>	SML <sub>7</sub>	SML <sub>6</sub>	SML <sub>5</sub>	SML <sub>4</sub>	SML <sub>3</sub>	SML <sub>2</sub>	SML <sub>1</sub>	SML <sub>0</sub>

---

Data Type: 8-Bit integer

USAGE: This value is the lower 8-bit of shutter maximum open time. Shutter value represents pixel array exposure time in multiples of internal clock cycles.

---

**Reserved** Address: 0x3d

---

**Inverse\_Revision\_ID** Address: 0x3e  
Access: Read Reset Value: 0xFF

---

<b>Bit</b>	7	6	5	4	3	2	1	0
<b>Field</b>	NRID <sub>7</sub>	NRID <sub>6</sub>	NRID <sub>5</sub>	NRID <sub>4</sub>	NRID <sub>3</sub>	NRID <sub>2</sub>	NRID <sub>1</sub>	NRID <sub>0</sub>

---

Data Type: Inverse 8-Bit unsigned integer

USAGE: This value is the inverse of the Revision\_ID. It can be used to test the SPI port.

---

**Inverse\_Product\_ID** Address: 0x3f  
Access: Read Reset Value: 0x77

---

<b>Bit</b>	7	6	5	4	3	2	1	0
<b>Field</b>	NPID <sub>7</sub>	NPID <sub>6</sub>	NPID <sub>5</sub>	NPID <sub>4</sub>	NPID <sub>3</sub>	NPID <sub>2</sub>	NPID <sub>1</sub>	NPID <sub>0</sub>

---

Data Type: Inverse 8-Bit unsigned integer

USAGE: This value is the inverse of the Product\_ID. It can be used to test the SPI port.

---



---

**Reserved** Address: 0x40-0x5f

---

**OFN\_Engine1** Address: 0x60  
Access: Read/Write Reset Value: 0x84

---

Bit	7	6	5	4	3	2	1	0
Field	Engine	Speed	Assert/ Deassert	XYQ	Soft_Click	FPD_En	XY_Scale	PWM

---

Data Type: Bit field

USAGE: This register is used to set several properties of the sensor.

Field Name	Description
Engine	Master control of OFN engine. Need to enable this bit to enable the rest of OFN features and properties <b>0 = Disable OFN properties</b> <b>1 = Enable OFN properties</b>
Speed_En	Speed switching enable/disable bit <b>0 = Disable speed switching</b> <b>1 = Enable speed switching</b>
Assert/ Deassert	Assert/ Deassert mode enable/disable bit <b>0 = Disable Assert/Deassert</b> <b>1 = Enable Assert/Deassert</b>
XY Q	XY quantization enable/disable bit <b>0 = Disable quantization</b> <b>1 = Enable quantization</b>
Soft_Click	Soft click enable/disable bit <b>0 = Disable soft click</b> <b>1 = Enable soft click</b>
FPD_En	Finger presence detection enable/disable bit <b>0 = Disable finger presence detection</b> <b>1 = Enable finger presence detection</b>
XY_Scale	XY scaling factor enable/disable bit <b>0 = Disable scaling</b> <b>1 = Enable scaling</b>
PWM	Pulse Width Modulation function enable/disable bit <b>0 = Disable PWM</b> <b>1 = Enable PWM</b>

**OFN\_Engine2** Address: 0x61  
 Access: Read/Write Reset Value: 0x89

Bit	7	6	5	4	3	2	1	0
Field	Reserved	Reserved	FPD_Rest_Wake1	FPD_Rest_Wake0	FPD_Mot_Cut	Reserved	PAD_Chk On	FPD_SQ_EN

Data Type: Bit field

USAGE: This register is used to set several properties of the sensor related to FPD function.

Field Name	Description
FPD_Rest_Wake1:0	Wakes up from rest with FPD <b>0 = FPD rest wake up disabled</b> 1 = Wake up when finger is present OR motion is detected 2 = Wake up when finger is present AND motion is detected
FPD_Mot_Cut	In FPD enabled mode, cut off motion when finger is not detected and shutter is below preset value. Shutter preset value can configured by SHUT_CUTOFF_THRES in FPD_CTRL (register 0x75). 0 = Disable FPD_Mot_Cut <b>1 = Enable FPD_Mot_Cut</b>
PAD_Chk_On	Enable this bit for PAD_STATUS check
FPD_SQ_EN	To improve FPD on low reflectance surface by comparing the instantaneous Squal value with a threshold set in FPD_SQUAL_THRESH (register 0x78). 0 = Disable FPD_SQ_EN 1 = Enable FPD_SQ_EN

**Resolution** Address: 0x62  
 Access: Read/Write Reset Value: 0x32

Bit	7	6	5	4	3	2	1	0
Field	WakeRES <sub>3</sub>	WakeRES <sub>2</sub>	WakeRES <sub>1</sub>	WakeRES <sub>0</sub>	RES <sub>3</sub>	RES <sub>2</sub>	RES <sub>1</sub>	RES <sub>0</sub>

Data Type: Bit field

USAGE: This register is used to set several properties of the sensor.

Field Name	Description
WakeRES <sub>3:0</sub>	Sets resolution when sensor wakes up from rest modes. Effective only when speed switching is enabled. 0x00 : 125 cpi 0x05 : 1250 cpi 0x01 : 250 cpi <b>0x02 : 500 cpi</b> 0x03 : 750 cpi 0x04 : 1000 cpi
RES <sub>3:0</sub>	Sets resolution of sensor. 0x00 : 125 cpi 0x05 : 1250 cpi 0x01 : 250 cpi <b>0x02 : 500 cpi</b> 0x03 : 750 cpi 0x04 : 1000 cpi  If Speed switching is enabled, resolution is automatically switched from 250-500-750-1000-1250. If speed switching is disabled, this register can be set to any of the available settings from 0x0 to 0x5.

**Speed\_Ctrl** Address: 0x63  
 Access: Read/Write Reset Value: 0x0E

Bit	7	6	5	4	3	2	1	0
Field	SEL_XY	XY_scale2	XY_scale1	XY_scale0	SP_IntVal1	SP_IntVal0	Reserved	Reserved

Data Type: Bit field

USAGE: This register is used to set several properties of the sensor.

Field Name	Description
Sel_XY	Select X or Y scale <b>0 = Enable X scale</b> 1 = Enable Y scale
XY_scale 2:0	Set scaling factor <b>0 = 0%</b> 4 = 50% 1 = 12.5%    5 = 62.5% 2 = 25%    6 = 75% 3 = 37.5%    7 = 87.5%
SPIntVal 1:0	Speed switching checking interval 0x00: 4 ms 0x01: 8 ms 0x02: 12 ms <b>0x03: 16 ms</b>

**Speed\_ST12** Address: 0x64  
 Access: Read/Write Reset Value: 0x08

Bit	7	6	5	4	3	2	1	0
Field	ST	ST	ST	ST	ST	ST	ST	ST

Data Type: Bit field

USAGE: This register is used to set several speed switching properties of the sensor.

Field Name	Description
ST 7:0	Sets resolution switching from step 1 to 2. Write in hexadecimal value. Formula (in decimal) = Velocity (inch per second) * 8

**Speed\_ST21** Address: 0x65  
 Access: Read/Write Reset Value: 0x06

Bit	7	6	5	4	3	2	1	0
Field	ST	ST	ST	ST	ST	ST	ST	ST

Data Type: Bit field

USAGE: This register is used to set several speed switching properties of the sensor.

Field Name	Description
ST 7:0	Sets resolution switching from step 2 to 1. Write in hexadecimal value. Formula (in decimal) = Velocity (inch per second) * 8

**Speed\_ST23**

Access: Read/Write

Address: 0x66

Reset Value: 0x40

<b>Bit</b>	7	6	5	4	3	2	1	0
<b>Field</b>	ST	ST	ST	ST	ST	ST	ST	ST

Data Type: Bit field

USAGE: This register is used to set several speed switching properties of the sensor.

Field Name	Description
ST 7:0	Sets resolution switching from step 2 to 3. Write in hexadecimal value. Formula (in decimal) = Velocity (inch per second) * 8

**OFN\_Speed\_ST32**

Access: Read/Write

Address: 0x67

Reset Value: 0x08

<b>Bit</b>	7	6	5	4	3	2	1	0
<b>Field</b>	ST	ST	ST	ST	ST	ST	ST	ST

Data Type: Bit field

USAGE: This register is used to set several speed switching properties of the sensor.

Field Name	Description
ST 7:0	Sets resolution switching from step 3 to 2. Write in hexadecimal value. Formula (in decimal) = Velocity (inch per second) * 8

**OFN\_Speed\_ST34**

Access: Read/Write

Address: 0x68

Reset Value: 0x48

<b>Bit</b>	7	6	5	4	3	2	1	0
<b>Field</b>	ST	ST	ST	ST	ST	ST	ST	ST

Data Type: Bit field

USAGE: This register is used to set several speed switching properties of the sensor.

Field Name	Description
ST 7:0	Sets resolution switching from step 3 to 4. Write in hexadecimal value. Formula (in decimal) = Velocity (inch per second) * 8

---

**Speed\_ST43** Address: 0x69  
Access: Read/Write Reset Value: 0x0a

---

<b>Bit</b>	7	6	5	4	3	2	1	0
<b>Field</b>	ST	ST	ST	ST	ST	ST	ST	ST

---

Data Type: Bit field

USAGE: This register is used to set several speed switching properties of the sensor.

---

<b>Field Name</b>	<b>Description</b>
ST 7:0	Sets resolution switching from step 4 to 3. Write in hexadecimal value. Formula (in decimal) = Velocity (inch per second) * 8

---

---

**Speed\_ST45** Address: 0x6a  
Access: Read/Write Reset Value: 0x50

---

<b>Bit</b>	7	6	5	4	3	2	1	0
<b>Field</b>	ST	ST	ST	ST	ST	ST	ST	ST

---

Data Type: Bit field

USAGE: This register is used to set several speed switching properties of the sensor.

---

<b>Field Name</b>	<b>Description</b>
ST 7:0	Sets resolution switching from step 4 to 5. Write in hexadecimal value. Formula (in decimal) = Velocity (inch per second) * 8

---

---

**Speed\_ST54** Address: 0x6b  
Access: Read/Write Reset Value: 0x48

---

<b>Bit</b>	7	6	5	4	3	2	1	0
<b>Field</b>	ST	ST	ST	ST	ST	ST	ST	ST

---

Data Type: Bit field

USAGE: This register is used to set several speed switching properties of the sensor.

---

<b>Field Name</b>	<b>Description</b>
ST 7:0	Sets resolution switching from step 5 to 4. Write in hexadecimal value. Formula (in decimal) = Velocity (inch per second) * 8

---

**GPIO\_CTRL** Address: 0x6c  
 Access: Read/Write Reset Value: 0x80

Bit	7	6	5	4	3	2	1	0
Field	GPIO_Active_Set	Reserved	GPIO_Mode5	GPIO_Mode4	PWM_Thres3	PWM_Thres2	PWM_Sel1	PWM_Sel0

Data Type: Bit field

USAGE: This register is used to set GPIO and PWM control.

Field Name	Description
GPIO_Active_Set	0 = GPIO is active LOW output <b>1 = GPIO is active HIGH output</b>
GPIO_Mode5:4	<b>0 = Output FPD</b> 1 = Output PWM 2 = Mechanical button input
PWM_Thres3:2	Start PWM only when Delta_X + Delta_Y is greater than threshold in Speed mode (Speed bit below). 0 = delta > 0 1 = delta > 8 2 = delta > 16 3 = delta > 24
PWM_Sel1:0	0 = Finger On/Off, light up in 1 sec 1 = Finger On/Off, light up in 2 sec 2 = Finger On/Off, light up in 3 sec 3 = Speed

**AD\_CTRL** Address: 0x6d  
 Access: Read/Write Reset Value: 0xc4

Bit	7	6	5	4	3	2	1	0
Field	1	1	Reserved	Reserved	ST_HIGH3	ST_HIGH2	ST_HIGH1	ST_HIGH0

Data Type: Bit field

USAGE: This register is used to control Assert De-assert. Must write 1 to bit 7 and 6.

Field Name	Description
ST_HIGH3:0	High speed definition. Any steps equal to or above this setting will use high speed A/D threshold.

---

**AD\_ATH\_HIGH** Address: 0x6e  
Access: Read/Write Reset Value: 0x3a

---

<b>Bit</b>	7	6	5	4	3	2	1	0
<b>Field</b>	ATH_H	ATH_H	ATH_H	ATH_H	ATH_H	ATH_H	ATH_H	ATH_H

---

Data Type: Bit field

USAGE: This register is used to set HIGH speed Assert shutter threshold.

---

<b>Field Name</b>	<b>Description</b>
ATH_H 7:0	Sets HIGH speed assert threshold. Write in hexadecimal value. Formula (in decimal) = Shutter value / 8. It is recommended to have hysteresis of 60 to 100 between assert and de-assert threshold.

---

---

**AD\_DTH\_HIGH** Address: 0x6f  
Access: Read/Write Reset Value: 0x40

---

<b>Bit</b>	7	6	5	4	3	2	1	0
<b>Field</b>	DTH_H	DTH_H	DTH_H	DTH_H	DTH_H	DTH_H	DTH_H	DTH_H

---

Data Type: Bit field

USAGE: This register is used to set HIGH speed De-assert shutter threshold.

---

<b>Field Name</b>	<b>Description</b>
DTH_H 7:0	Sets HIGH speed de-assert threshold. Write in hexadecimal value. Formula (in decimal) = Shutter value / 8. It is recommended to have hysteresis of 60 to 100 between assert and de-assert threshold.

---

---

**AD\_ATH\_LOW** Address: 0x70  
Access: Read/Write Reset Value: 0x35

---

<b>Bit</b>	7	6	5	4	3	2	1	0
<b>Field</b>	ATH_L	ATH_L	ATH_L	ATH_L	ATH_L	ATH_L	ATH_L	ATH_L

---

Data Type: Bit field

USAGE: This register is used to set LOW speed Assert shutter threshold.

---

<b>Field Name</b>	<b>Description</b>
ATH_L 7:0	Sets LOW speed assert threshold. Write in hexadecimal value. Formula (in decimal) = Shutter value / 8. It is recommended to have hysteresis of 60 to 100 between assert and de-assert threshold.

---

**AD\_DTH\_LOW** Address: 0x71  
 Access: Read/Write Reset Value: 0x3b

Bit	7	6	5	4	3	2	1	0
Field	DTH_L	DTH_L	DTH_L	DTH_L	DTH_L	DTH_L	DTH_L	DTH_L

Data Type: Bit field

USAGE: This register is used to set LOW speed De-assert shutter threshold.

Field Name	Description
DTH_L 7:0	Sets LOW speed de-assert threshold. Write in hexadecimal value. Formula (in decimal) = Shutter value / 8. It is recommended to have hysteresis of 60 to 100 between assert and de-assert threshold.

**QUANTIZE\_CTRL** Address: 0x72  
 Access: Read/Write Reset Value: 0x99

Bit	7	6	5	4	3	2	1	0
Field	YQ_ON	YQ_DIV <sub>6</sub>	YQ_DIV <sub>5</sub>	YQ_DIV <sub>4</sub>	XQ_ON	XQ_DIV <sub>2</sub>	XQ_DIV <sub>1</sub>	XQ_DIV <sub>0</sub>

Data Type: Bit field

USAGE: This register is used to set quatization for Delta\_X and Delta\_Y. If both X and Y quantization modes are on, then only largest quantized X or Y will be reported.

Field Name	Description
YQ_ON	0 = Y quantization off 1 = Y quantization On
YQ_DIV <sub>6:4</sub>	Quantization factor $2^{YQ\_DIV}$ . Reported $YQ = DY / 2^{YQ\_DIV}$ .
XQ_ON	0 = X quantization off 1 = X quantization On
XQ_DIV <sub>2:0</sub>	Quantization factor $2^{XQ\_DIV}$ . Reported $XQ = DX / 2^{XQ\_DIV}$ .

**XYQ\_THRESH** Address: 0x73  
 Access: Read/Write Reset Value: 0x02

Bit	7	6	5	4	3	2	1	0
Field	Reserved	Reserved	Reserved	Reserved	Reserved	XYQ_M	XYQ_C <sub>1</sub>	XYQ_C <sub>0</sub>

Data Type: Bit field

USAGE: This register is used to set quatization gradient for DeltaX and DeltaY.

Field Name	Description
XYQ_M	Gradient of linear region 0 = Gradient 1 1 = Gradient 2
XYQ_C <sub>1:0</sub>	Indicates the offset of linear region (max of C = 3 or 0x03)



**MOTION\_CTRL** Address: 0x74  
Access: Read/Write Reset Value: 0x00

Bit	7	6	5	4	3	2	1	0
Field	Reserved	Reserved	Reserved	Reserved	Reserved	Motion_Int_Thres2	Motion_Int_Thres1	Motion_Int_Thres0

Data Type: Bit field

USAGE: This register is used to control Motion interrupt.

Field Name	Description
Motion_Int_Thres2:0	Motion interrupt threshold Only trigger Motion Interrupt if $\Delta_X + \Delta_Y > \text{Threshold}$

**FPD\_CTRL** Address: 0x75  
Access: Read/Write Reset Value: 0xfa

Bit	7	6	5	4	3	2	1	0
Field	Shut_Cut_Thres7	Shut_Cut_Thres6	Shut_Cut_Thres5	Shut_Cut_Thres4	FPD_On_Hyst3	FPD_On_Hyst2	FPD_Off_Hyst1	FPD_Off_Hyst0

Data Type: Bit field

USAGE: This register is used to set sensor FPD control.

Field Name	Description
Shut_Cut_Thres7:4	Motion is cut off when shutter value is less than threshold. Valid only when FPD_Mot_Cutoff is enabled. Units are multiple of 256.
FPD_On_Hyst3:2	Hysteresis counter for finger on
FPD_Off_Hyst1:0	Hysteresis counter for finger off

**FPD\_THRESH** Address: 0x76  
Access: Read/Write Reset Value: 0x1c

Bit	7	6	5	4	3	2	1	0
Field	FPD_T	FPD_T	FPD_T	FPD_T	FPD_T	FPD_T	FPD_T	FPD_T

Data Type: Bit field

USAGE: This register is used to set the sensitivity of FPD. See OFN A350 Firmware Guide for more information on FPD operation and thresholds setting.

Field Name	Description
FPD_T 7:0	Threshold to compare with FPD Value

**ORIENT\_CTRL** Address: 0x77  
 Access: Read/Write Reset Value: 0x00

Bit	7	6	5	4	3	2	1	0
Field	XY_SWAP	Y_INV	X_INV	Reserved	Reserved	Reserved	Reserved	Reserved

Data Type: Bit field

USAGE: This register is used to configure Delta\_X and Delta\_Y reporting direction with respect to sensor orientation

Field Name	Description
XY_SWAP	<b>0 = Normal sensor reporting of Delta_X, Delta_Y. (default)</b> 1 = Swap data of Delta_X to Delta_Y and Delta_Y to Delta_X.
Y_INV	<b>0 = Normal sensor reporting of Delta_Y. (default)</b> 1 = Invert data of Delta_Y only.
X_INV	<b>0 = Normal sensor reporting of Delta_X. (default)</b> 1 = Invert data of Delta_X only.

**FPD\_SQUAL\_THRESH** Address: 0x78  
 Access: Read/Write Reset Value: 0x40

Bit	7	6	5	4	3	2	1	0
Field	FPD_S_T	FPD_S_T	FPD_S_T	FPD_S_T	FPD_S_T	FPD_S_T	FPD_S_T	FPD_S_T

Data Type: Bit field

USAGE: This register is to set the FPD Squal Threshold. It is used when bit 0 (FPD\_SQ\_EN bit) of OFN\_Engine2 (0x61) is enabled for low reflectance surface navigation.

Field Name	Description
FPD_S_T	Squal FPD Threshold to compare with Squal values

**FPD\_VALUE** Address: 0x79  
 Access: Read/Write Reset Value: 0x00

Bit	7	6	5	4	3	2	1	0
Field	SC_Dou	SC_Dou	SC_Dou	SC_Dou	SC_Dou	SC_Dou	SC_Dou	SC_Dou

Data Type: Bit field

USAGE: This register stores the FPD value that indicates surface reflectivity. Higher FPD value means higher finger or surface reflectivity.

Field Name	Description
FPD_VALUE 7:0	FPD Value

**FPD\_STATUS** Address: 0x7a  
Access: Read Reset Value: 0x00

Bit	7	6	5	4	3	2	1	0
Field	FPD_Hyst7	FPD_Hyst6	FPD_Hyst5	FPD_Hyst4	FPD_SQ_Hyst3	FPD_SQ_Hyst2	FPD_SQ_Hyst1	FPD_On

Data Type: Bit field

USAGE: This register is used to confirm FPD flag and to set FPD hysteresis threshold.

Field Name	Description
FPD_Hyst7:4	FPD hysteresis threshold boundary. Actual FPD Hysteresis Threshold = 8 x [FPD_Hyst7:4]
FPD_SQ_Hyst3:1	FPD Squal hysteresis threshold boundary. Actual FPD SQ Hysteresis Threshold = 4 x [FPD_SQ_Hyst3:1]
FPD_On	0 = finger is not present 1 = finger is present

**SC\_CTRL** Address: 0x7b  
Access: Read/Write Reset Value: 0x25

Bit	7	6	5	4	3	2	1	0
Field	Reserved	TIM_Res_Th6	TIM_Res_Th5	TIM_Res_Th4	Reserved	Tap+N_Hold_En	Double_Click_En	Single_Click_En

Data Type: Bit field

USAGE: This register is used to set sensor soft click.

Field Name	Description
TIM_Res_Th6:4	Tap and Hold timer reset threshold
Tap_N_Hold_En	0 = Disable Tap and Hold 1 = Enable Tap and Hold
Double_Click_En	0 = Disable Double Click 1 = Enable Double Click
Single_Click_En	0 = Disable Single Click 1 = Enable Single Click

**SC\_T\_TAPN\_HOLD** Address: 0x7c  
Access: Read/Write Reset Value: 0x45

Bit	7	6	5	4	3	2	1	0
Field	SC_Dou	SC_Dou	SC_Dou	SC_Dou	SC_Dou	SC_Dou	SC_Dou	SC_Dou

Data Type: Bit field

USAGE: This register is used to set the minimum time before Tap N Hold is triggered.

Field Name	Description
SC_Dou 7:0	Sets waiting time to activate Tap and Hold

**SC\_T\_DOUBLE** Address: 0x7d  
 Access: Read/Write Reset Value: 0x1e

Bit	7	6	5	4	3	2	1	0
Field	SC_Dou	SC_Dou	SC_Dou	SC_Dou	SC_Dou	SC_Dou	SC_Dou	SC_Dou

Data Type: Bit field

USAGE: This register is used to set time between first click and second click to trigger double click. Any second successive click within this time will be considered a double click.

Field Name	Description
SC_Dou 7:0	Sets waiting time after single click for double click to happen

**SC\_DELTA\_THRESH** Address: 0x7e  
 Access: Read/Write Reset Value: 0x19

Bit	7	6	5	4	3	2	1	0
Field	SC_Delta	SC_Delta	SC_Delta	SC_Delta	SC_Delta	SC_Delta	SC_Delta	SC_Delta

Data Type: Bit field

USAGE: This register is used to set sensor single click threshold.

Field Name	Description
SC_Delta 7:0	Threshold to validate single click. Single click is only valid if motion delta sum (Delta_X + Delta_Y) is less than threshold. Units are in 500 cpi.

**SC\_STATUS** Address: 0x7f  
 Access: Read/Write Reset Value: 0x00

Bit	7	6	5	4	3	2	1	0
Field	Reserved	Reserved	Reserved	Reserved	Reserved	Tap+N_Hold	Double_Click	Single_Click

Data Type: Bit field

USAGE: This register is used to confirm which soft click has been triggered when there is a soft click interrupt being triggered.

Field Name	Description
Tap_N_Hold	0 = Tap and Hold not detected 1 = Tap and Hold detected
Double_Click	0 = Double click not detected 1 = Double click detected
Single_Click	0 = Single click not detected 1 = Single click detected

## Packing information

1. ADBM-A350 : This is the standard with MOQ=1000pcs
2. ADBM-A350-200 : This is with MOQ=200pcs

For product information and a complete list of distributors, please go to our web site: [www.avagotech.com](http://www.avagotech.com)

Avago, Avago Technologies, and the A logo are trademarks of Avago Technologies in the United States and other countries.  
Data subject to change. Copyright © 2005-2011 Avago Technologies. All rights reserved.  
AV02-2905EN - May 3, 2011

