ADUXxx Technical Specification

aka: adux1022, apdp103,5,6,7… and various modules

Proximity and Gesture Sensor

Van Buren Edition

Rev 6.03

5/22/2017

Table of Contents

Revision

Revision	Date	Description
0.0	6/27/2013	Initial version (gk)
0.1	7/10/2013	• Changed afe_mask to be afe_icmask_a,b • Added register to control cmclock during test mode • Added 1 bit to pulse_offset registers • Removed afe_ref_enb and sw2n_en unused registers • Added operating mode registers for a,b and global control • Added data formats for FIFO data • Specified mode change methodology • Described new gesture mask/interrupt feature • Clarified/corrected interrupt pin control feature • Clarified R_MODULATE_A,B register function • Fixed incorrect “example flow with modulate pulse” diagram • Changed AOCANCEL control fields to match fine pulse width control with only course offset control • Clarified VDD to PDs during sleep function • Specified qreset delay register width • Specified vcm_width register widths • Added timing diagrams to illustrate various modes and relationships
0.2	7/12/13	• Minor update to the block diagram • Changed r_op_mode_a codes and separated auto mode switch like adux1020 • Corrected names for R_MAX_GESTURE_LENGTH and R_GEST_MASK registers • Added R_PROX_SAMPLE_EN feature to output sample data between prox on and prox off. Removed “add new output sample data between prox on/off” from the not done list. • Corrected typo in r_op_mode_b table. • R_AFE_CAL_EN is not duplicated.
0.3	7/15/13	• Added missing digital gain/offset registers for timeslot a and b
0.4	7/21/13	• Added more signals to timing diagrams • Added photodiode diagram • Included re-mapped register map • Added comments to dual-led drive mode • Described saturation to 20 bits of digital sample integration • Clarified gesture dataout modes • Added fifoclock enable bit • Clarified aocancel pulse description and signal name • Corrected qreset offset edge timing vs afe_offset

0.5	8/7/13	• Added r_hold_regs_a and r_hold_regs_b feature. • Added r_vcm_float_en feature • Clarified offset values • Revised pulse mask register • Revised register defaults • Added PD diagram/pins • Added sample interrupt timing control bits • Added better description of data registers and their usage • Added register to allow separate trim values for LED1 and LED2
1.1	11/6/13	• Corrected various typos. • Adjusted to match actual PD-2 type sensor. • This version covers R0 and R1 silicon.
2.0	12/6/13	• New release to cover R2 silicon functionality • Added gain calibration mode • Added efuse block • Added cmclk only during vcm option • Added better handling of unused photodiodes • Allow longer qreset, afe, and aoc delays to allow longer float times • Added additional afe power control options • Added difference mode for two pulse with different float time accesses for ALS • Added independent vcm, qreset timing controls for timeslot b • Added additional modulate types • Added additional “connect” switch to provide less leakage during ALS float cycles • Minor default changes • Typo fixes • New control bits for new bias channel for the afe reference • Changed afe reference to be independent of power supply voltage • Changed ESD structures to low leakage diodes • Bug fixes for R0/R1
2.1	1/7/14	• Various minor typo fixes
2.2	2/10/14	• Various minor typo fixes
3.0	4/23/14	• New release to cover R3 silicon functionality • New Float during Sleep mode to allow much longer sleep times • New Float gesture mode to allow float type accesses to be combined with LED flashes • Digital integrate mode for 221 compatibility • New cathode voltage control options and voltages • I2C read automatic address increment • FIFO now supports 32bit data registers • Shorten timeslot a calculate time when not doing gesture operations • Better control of cmclk during precondition mode • Support separate tia gain resistor settings • Periodic qreset mode

3.1	5/15/14	• Minor typo fixes. • Better description of LED testmodes
4.0	12/4/14	• New release to cover R4 functionality • Provide ESD Diodes on all photodiode inputs • Provide sum of channels output for sample mode • Add features to provide background value in digital integrate mode • Option in digital integration mode to add space between sample regions • Option in digitial integration mode to do a single access region (one-shot mode) as well as the existing two region method • Option in digital integration mode to use only one photodiode input/channel • Method to allow changing the I2C address for devices on the same bus using INT or TCLI pins • Add separate control to allow using the integrator as a buffer when not using digital integration mode • Allow use of INT or TCLI pins as a sample trigger • Other test feature to provide a new data test pattern • Option to write the FIFO with the 16 lsbs of the 32 bit value rather than the normal shifted/saturated value. • Removed non-functional data and packet flag functions (reg 0x11)
4.03	3/5/15	• Various typo fixes and clarifications

5.00	2/1/16	• New release to cover R5 functionality
		  ○ Added SPI interface.
		  ○ Random Sleep mode.
		  ○ ADC cycle removal.
		  ○ Pulsed Vref mode.
		  ○ TCLI/INT pin GPIO functionality.
		  ○ New FIFO design without read/clear clocking requirement.
		  ○ Cathode and Anode float options.
		  ○ Cathode toggle options.
		  ○ Additional AFE power controls
		  ○ New lower current Power-On Reset circuit.
		  ○ Better 32MHz oscillator.
		  ○ The integrator input resistor (R4) has been doubled.
		  ○ The analog gain setting for digital integration has been changed from (1 or 2) to (1 or .6875).
		  ○ The LED driver has been redesigned to have better 1/f noise performance.
		  ○ Software reset now ACK's on the I2C interface
		  ○ Integrator as buffer mode is fixed.
		  ○ Internal 32KHz oscillator does not need to be enabled while using an external 32KHz clock.
		• Various typos fixed

5.01	3/21/16	• Fixed additional typos • Removed R_ADC_CHAN_DISABLE[3:0] as not implemented

6.00	2/10/17	• Initial version for Van Buren tape out
		• Digital changes
		  ○ Digital control changes to lower power by turning off LED amplifiers, Bandgap, Bias Distribution, and Vref buffer when not needed.
		  ○ New “four pulse normal” mode including option to reverse integrate clock order.
		  ○ Logic to support new simpler pd mux switches and pd modes. Includes combining the sw_mod signal with the mux selection value.
		  ○ Provide indication for “ambient” value greater than preset threshold for a configurable number of occurrences. Indicator can be read in a register and optionally driven to a GPIO pin.
		  ○ FIFO clear operation now also clears the FIFO interrupt.
		  ○ ADC continuous mode bug fix.
		  ○ Remove glitches from “timeslot a” and “timeslot b” output signals on gpio pins.
		  ○ Remove need for TIA1 to be on for other channels to work in TIA/BUF/ADC mode.
		  ○ Increment chip-id.
		  ○ Default r_ed_qreset_en to 1.
		  ○ Remove digital gain to free register space
		• Analog changes
		  ○ Total resistance goal of 5K ohm with +/- 2K variance. Now using an ESD resistor of 6.125k ohm in series with a single (approximately 200 ohm) pd mux switch.
		  ○ Design changes to support better PSRR. Targeting 60dB
		  ○ Increase TIA FB cap from 0.9pf to 2.4pf
		  ○ Hardcode afe_trim_bpf_ to code 1.
		  ○ Hardcode INT feedback cap to 6.3pf
		  ○ Current multiplier for LED current change from .4 to .1
		• Discussed but not included:
		  ○ 3 point median filter
		  ○ CIC 2nd order
		  ○ Positive offset for noise measurement

6.02	5/19/17	• Fixed various typos • Changed defaults for adc dummy cycle and reset operations • Added INT/TCLI codes for bigdark plus interrupt function to match R7 silicon
6.03	5/22/17	• Fixed typo in integrator gain value to the correct value for R5 and up. (registers 0x42,0x44)

Introduction

Description of this document

This manual describes the ADUX10xx in terms of the differences from the ADUX1020. It does not fully describe the ADUX1020 feature set. It contains all the intended changes to create the new device.

Disclaimer

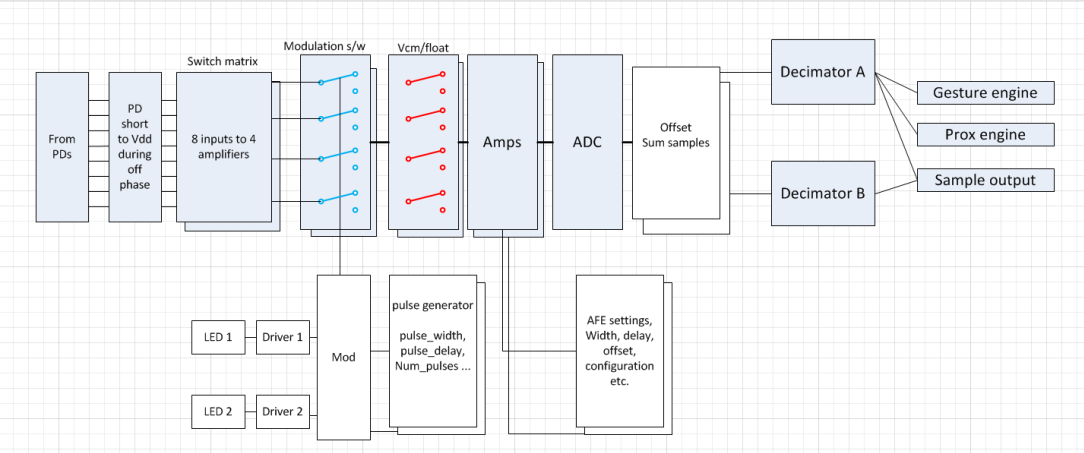
Naming Conventions

Name/Symbol	Description
<name>_PROX	Register names with PROX suffix are used when Proximity Mode is operating in Timeslot A
<name>_A	Register names with _A suffix control device during Timeslot A
<name>_B	Register names with _B suffix control device during Timeslot B

The registers that have been changed from the adux1020 are shown in the text. A full register map is also included.

Architecture Overview

Block Diagram



The ADUX10xx is an enhanced version of the ADUX1020 product. It is modified to allow programmable connection to two sensor PDs and provides timeslots for accessing either or both sensors. The above diagram represents the internal architecture. A shadow block in the diagram indicates that there are independent settings for each timeslot. Each timeslot also has an independent decimation filter block. The main additions are the additional data path, switch matrix, additional LED driver and timing generator, control for supporting multiple timeslots, larger FIFO, and bug fixes.

Timeslots

To support the use of two sensors the operating cycle has been broken into two individual timeslots. The operation that occurs in each timeslot is programmable. Timeslot A supports all operations from the ADUX1020. Timeslot B supports a more limited set of operations. The timeslot function cycles can be selected from the following choices:

Mode	Sequence
A only mode	Sleep – Timeslot A – Sleep –Timeslot A etc.
B only mode	Sleep – Timeslot B – Sleep –Timeslot B etc.
A and B mode	Sleep – Timeslot A – Timeslot B – Sleep – Timeslot A – Timeslot B – Sleep

Each Timeslot will have its own switch matrix, AFE, and LED control timings. The sample rate will be controlled by the Timeslot A control registers R_GEST_FREQ and R_PROX_FREQ, whichever is active. The sample rate control has been expanded to a full 16-bit count value to allow finer rate control. We will not support the predefined constants used on the adux1020. The formula for sample rate is:

Sample rate = 32kHz / (R_GEST_FREQ * 4)	When TIMESLOT A is not in Proximity mode
Sample rate = 32kHz / (R_PROX_FREQ * 4)	When TIMESLOT A is in Proximity mode

AFE timing controls

The adux10xx allows separate independent control of the AFE settings for each timeslot. Timeslot A also supports alternate settings for proximity mode.

Old adux1020 name	New adux10xx registers	Comment
R_AFE_FINE_OFFSET	R_AFE_FINE_OFFSET_A R_AFE_FINE_OFFSET_B	
R_AFE_OFFSET	R_AFE_OFFSET_A R_AFE_OFFSET_B	
R_AFE_WIDTH	R_AFE_WIDTH_A R_AFE_WIDTH_B	
R_AFE_FINE_OFFSET_PROX	R_AFE_FINE_OFFSET_PROX_A	no prox in timeslot b
R_AFE_OFFSET_PROX	R_AFE_OFFSET_PROX_A	no prox in timeslot b
R_AFE_WIDTH_PROX	R_AFE_WIDTH_PROX_A	no prox in timeslot b
R_AFE_MULTI_SAMPLE	R_AFE_MULTI_SAMPLE_A R_AFE_MULTI_SAMPLE_B	
	R_AFE_ICMASK_A R_AFE_ICMASK_B	8 bits. Integration clock masks. Even bits mask positive pulses, odd mask negative. Bits 0,1 are first pulses.

For all offset values in the above table, the actual offset used is the register value plus 1. AFE_WIDTH values of 0 are not allowed.

Some AFE control registers control both timeslots:

Old adux1020 name	New adux10xx registers	Comment
R_CMCLK_SEL	R_CMCLK_SEL	Same values as adux1020
R_AFE_MASK	R_AFE_MASK	Same values as adux1020
R_AFE_CAL_EN	R_AFE_CAL_EN	Same values as adux1020

New registers were added to allow per channel integration clock masking. Also a register was added to control the cmclk during test.

Also note that the timing calibration features should only be used with timeslot a only operating.

Timeslot Timing for LED or sensor modulation

The timing controls for the LED pulse generation and or switch modulation are set on a **per timeslot basis**. The registers have been duplicated to allow each timeslot to be independent. The modulation registers have been renamed from the adux1020 according to this table:

Old adux1020 name	New adux10xx registers	Comment
R_LED_OFFSET	R_PULSE_OFFSET_A R_PULSE_OFFSET_B	This has been expanded to 7 bits in R0/R1, and 8 in R2
R_LED_WIDTH	R_PULSE_WIDTH_A R_PULSE_WIDTH_B	
R_LED_PERIOD	R_PULSE_PERIOD_A R_PULSE_PERIOD_B	
R_LED_NUMBER (6 bits)	R_PULSE_NUMBER_A (8 bits) R_PULSE_NUMBER_B (8 bits)	This has been expanded to 8 bits for 255 pulses.
R_LED_OFFSET_PROX	R_PULSE_OFFSET_PROX_A	This has been expanded to 7 bits in R0/R1 and 8 in R2, no prox in timeslot b
R_LED_WIDTH_PROX	R_PULSE_WIDTH_PROX_A	no prox in timeslot b
R_LED_PERIOD_PROX	R_PULSE_PERIOD_PROX_A	no prox in timeslot b
R_LED_FORCE_NUM	R_PULSE_FORCE_OPCYCLE_NUM	Test mode – timeslot a
R_LED_FORCE_EN	R_PULSE_FORCE_OPCYCLE_EN	Test mode – timeslot a
R_LED_MASK	R_PULSE_MASK_A R_PULSE_MASK_B	Test mode to mask led pulses.
n/a	R_PULSE_PERIOD_EXT_A R_PULSE_PERIOD_EXT_PROX_A R_PULSE_PERIOD_EXT_B	Added in R2 to provide two additional pulse offset bits.

For all offset values in the above table, the actual offset used is the register value plus 1. Although the number of pulses allowed has increased, the digital sample integration is still limited to 20 bits internally and will saturate on overflow.

In R2, the new R_PULSE_PERIOD_EXT register has been added to allow extension of the pulse offsets to a total of 10 bits. There are separate extension fields for each timeslot and proximity mode.

For each timeslot the modulation timing circuit can flash LED1, LED2, or pulse the connection of the pd array (through the switch matrix) to the AFE. This is selected by programming the R_MODULATE_A and R_MODULATE_B registers with the codes shown below:

Code	Description	Comment
0	Pulse PD connection to AFE	No LEDs will flash
1	Select LED1	LED1 to flash
2	Select LED2	LED2 to flash
3	Select LED1 and LED2	Both LEDs flash This may not be disclosed to customers as the LED current is reduced by half in this mode.

When “pulsing” the PD connection to AFE, the switch will be open during the pulse_offset time and connected during the pulse_width time, and opened afterwards.

AFE control registers

Some AFE control registers are duplicated for the two timeslots. These may not want to disclose these to customers.

Old adux1020 name	New adux10xx registers	Comment
R_AFE_TRIM_TIA	R_AFE_TRIM_TIA_A R_AFE_TRIM_TIA_B	
R_AFE_TRIM_BPF	R_AFE_TRIM_BPF_A R_AFE_TRIM_BPF_B	
R_AFE_TRIM_VREF	R_AFE_TRIM_VREF_A R_AFE_TRIM_VREF_B	
R_AFE_TRIM_INT	R_AFE_TRIM_INT_A R_AFE_TRIM_INT_B	
R_AFE_MUX_TEST	R_AFE_MUX_TEST_A R_AFE_MUX_TEST_B	

The values of these registers will be sent to the AFE during the appropriate timeslots.

Digital Gain/Offset per timeslot

The digital offset and gain settings are also configurable per timeslot as shown:

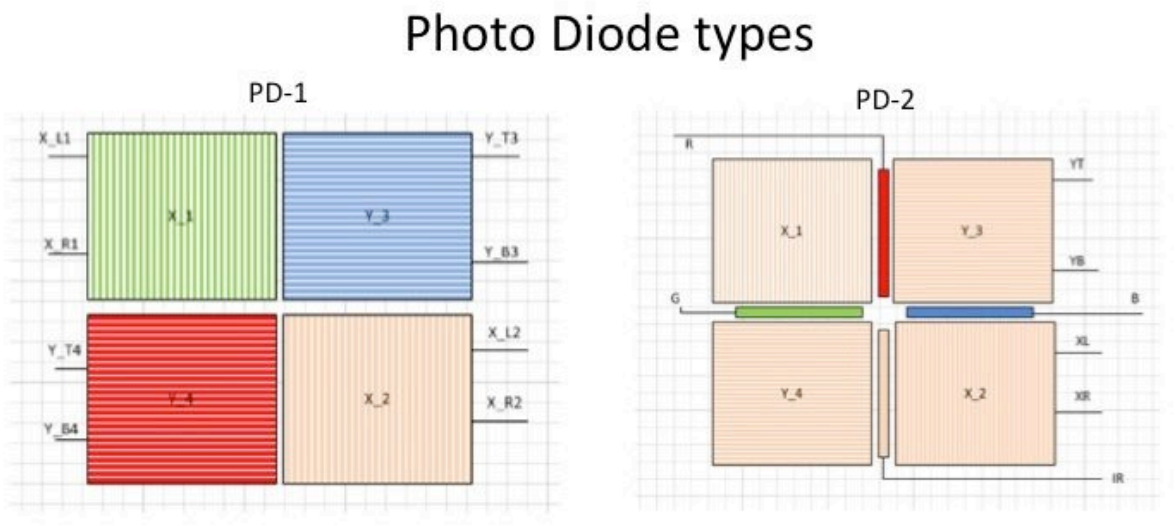
Old adux1020 name	New adux10xx registers	Comment
R_GAIN1	R_GAIN1_A R_GAIN1_B	same gain settings as adux1020
R_GAIN2	R_GAIN2_A R_GAIN2_B	
R_GAIN3	R_GAIN3_A R_GAIN3_B	
R_GAIN4	R_GAIN4_A R_GAIN4_B	
R_CH1_OFFSET	R_CH1_OFFSET_A R_CH1_OFFSET_B	default 0x2000 same as adux1020
R_CH2_OFFSET	R_CH2_OFFSET_A R_CH2_OFFSET_B	default 0x2000 same as adux1020
R_CH3_OFFSET	R_CH3_OFFSET_A R_CH3_OFFSET_B	default 0x2000 same as adux1020
R_CH4_OFFSET	R_CH4_OFFSET_A R_CH4_OFFSET_B	default 0x2000 same as adux1020

Switch Matrix and sensor types

A switching matrix is added to the input of the AFE to allow connection to a primary and secondary sensor. The switch settings have been selected to connect to either a single shared mode (PD-1) sensor or to two independent (PD-2) sensors. The switch matrix can be programmed to support connection of either type of sensor in either mode during each timeslot. The following table shows the connections and modes. A-D are the 4 AFE channels, 1-8 are the photodiode connections.

Code	Sensor type	Which sensor	A	B	C	D	Use
7	PD-1	Primary	1+3	2+4	5+7	6+8	Gesture
6	Other	n/a	5	6	N/C	N/C	
5	PD-2	Primary	1	2	3	4	ALS
4	PD-2	Secondary	5	6	7	8	Gesture
3	PD-1	Secondary	1+2	3+4	5+6	7+8	ALS
2	Other	n/a	N/C	N/C	N/C	N/C	
1	PD-1	Secondary	1+2	3+4	5+6	7+8	ALS
0	Other	n/a	N/C	N/C	N/C	N/C	

Timeslot A and B can independently select which switch mapping is done by using these codes in the R_SW_MATRIX_A for Timeslot A and R_SW_MATRIX_B for Timeslot B. The default settings assume a type PD-2 sensor doing Gesture in Timeslot A and ALS in Timeslot B.



AFE PD Inputs vs Sensor Type

Type	1	2	3	4	5	6	7	8
PD-1	XL1	XR1	XL2	XR2	YT3	YB3	YT4	YB4
PD-2	R	B	G	IR	YB	YT	XL	XR

Three LED Drivers

There are now three LED drivers (R3). The driver settings for each LED retains the same controls as in the adux1020. There is also an additional control to allow scaling by 40% to allow 10mA drive. The operating current for each LED is programmable in the registers shown:

Register	Description	**Comment**
R_LED1_GIREF	LED1 driver current	Same settings as adux1020
R_LED1_ITAIL	Slew rate control	Same as adux1020
R_LED1_EN250	LED1 current scale factor	1=same as adux1020, 0=40% of adux1020. This allows 10ma drive. **This is applied to both sample and proximity mode drive currents**
R_LED1_PIREF	LED1 proximity mode drive current	
R_LED1_PIREF_EN	LED1 proximity mode value enable	Proximity mode drive current 0: use LED1_IREF 1: use LED1_PIREF
R_LED2_GIREF	LED2 driver current	Same settings as adux1020
R_LED2_ITAIL	Slew rate control	Same as adux1020
R_LED2_EN250	LED2 current scale factor	1=same as adux1020, 0=40% of adux1020. This allows 10ma drive. **This is applied to both sample and proximity mode drive currents**
R_LED2_PIREF	LED2 proximity mode drive current	
R_LED2_PIREF_EN	LED2 proximity mode value enable	Proximity mode drive current 0: use LED2_GIREF 1: use LED2_PIREF
R_LED3_GIREF	LED3 driver current	Same settings as adux1020
R_LED3_ITAIL	Slew rate control	Same as adux1020
R_LED3_EN250	LED3 current scale factor	1=same as adux1020, 0=40% of adux1020. This allows 10ma drive. **This is applied to both sample and proximity mode drive currents**
R_LED3_PIREF	LED3 proximity mode drive current	
R_LED3_PIREF_EN	LED3 proximity mode value enable	Proximity mode drive current 0: use LED3_GIREF 1: use LED3_PIREF

If enabled, the proximity settings are applied during both timeslots when TIMESLOT A is operating in proximity mode.

The LED_TRIM register has been added to provide separate current trim values for LED1, LED2, and LED3.

Operating Modes

Mode selection method

The R_OP_MODE register is used to control the overall state of the chip. It supports the modes shown in the following table:

Code	Mode	Description
0	OFF	All sections powered down. The chip powers up in this mode.
1	IDLE	No timeslot operations occur. Put the chip in this mode before changing any operating settings.
2	GO	This allows the chip to start the enabled timeslot operations.

The required method of configuring the operations during the timeslots is as follows:

1. Write the R_OP_MODE register to IDLE
2. Make changes to all of the AFE, Pulse generator, data formats, for timeslot a gesture and proximity values, and op_mode_a.
3. Make changes to all of the AFE, Pulse generator, data formats, and op_mode_b for timeslot b.
4. Write the R_OP_MODE register to GO and the part will start operations as selected
5. If a change of operating mode is required, set R_OP_MODE to IDLE before making changes, repeat the sequence above.

No mode settings should be changed while the part is operating in GO. Make sure to enable Timeslot A, Timeslot B, or both before entering GO.

Timeslot A

The register R_OP_MODE_A selects the operation mode during timeslot A according to the values in the following table:

Code	Mode	Data_out_modes supported
0	Disabled	Mode 0 only
1	Sample	Modes 0, 3, 4, 6 allowed – others illegal Modes 0, 1, 2 allowed in digital integrate mode (added in R3)
2	Proximity	Modes 0, 1, 3
3	Gesture	Modes 0, 2, 5 allowed – others illegal

Smart sample mode continues to operate in timeslot A as it functioned on the adux1020.

When setting the R_PROX_AUTO_GESTURE bit, the part operates in proximity mode until the switch criteria (PROX_ON2) is met. When this occurs, timeslot B (if enabled) will complete its operations, and then the switch will occur. The R_GEST_FREQ sample rate will then be applied, with a new sample cycle commencing within a few 32kHz clock cycles. At this point the part will be operating with all the gesture mode settings for Timeslot A. Timeslot B will be unaffected except for the new sample rate.

Timeslot B

The register R_OP_MODE_B selects the operation mode during timeslot B according to the values in the following table:

Code	Mode	Data_out_modes supported
0	Disabled	Mode 0 only
1	Sample	Mode 0 allowed Mode 4 or 6 in sample mode, Mode 1 or 2 allowed only if B is in Digital Integrate sample mode, ONLY if Timeslot A is disabled – OR – if these conditions are met: • Timeslot A in sample mode • Timeslot A in data out format 4 or 6 • Timeslot A and B have the same final sample output rates. This requires the same multi-sample settings and same decimation factors. • Timeslot A must not be in smart sample mode

The data sizes for Timeslot A and B sample mode data can be different but the data rates must match. Care must be taken to read the appropriate data from the FIFO to maintain data alignment.

Data Formats

The R_DATAOUT_MODE_A and R_DATAOUT_MODE_B registers are used to select the data formats for data going to the FIFO. The following table shows the supported data formats:

Code	Bytes	Format	Description	Available for
0	0	Nothing in FIFO	Data only available in registers	Timeslot A and B
1	2	I (16bits)	Proximity mode	Timeslot A only
1	2	Channel 1 16 bit data	1 Data channel	Sample mode Digital Integrate only (R3)
1	2	Sum of channels (16 bits)	4 channel sum	Sample mode if not digital integrate (R4)
2	4	G (16bits) I (16bits)	Gesture mode (in proximity mode, no data will be output to fifo is this dataout mode is selected)	Timeslot A only
2	4	Channel 1 32 bit data	1 Data channel	Sample mode Digital Integrate only (R3)
2	4	Sum of channels (32 bit)	4 channel sum	Sample mode if not digital integrate (R4)
3	6	X (16 bits) Y (16 bits) I (16 bits)	Proximity sample values	Timeslot A only for Proximity or Sample mode
3	4	X1 (16 bits) X2 (16 bits)	Digital integrate signal and background value	Sample mode digital integrate only (R4)
4	8	X1 (16 bits) X2 (16 bits) Y1 (16 bits) Y2 (16 bits)	Sample mode data	Sample mode if not digital integrate
4	8	X1L (16 bits) X1H (16 bits) X2L (16 bits) X2H (16 bits)	Digital integrate signal and background value in 32 bit format	Sample mode digital integrate only (R4)
5	8	G (16 bits) I (16 bits) Maxdidt (16 bits) Mindidt (16 bits)	Gesture mode (in proximity mode, no data will be output to fifo is this dataout mode is selected)	Timeslot A only
6	16	X1L (16 bits) X1H (16 bits) X2L (16 bits) X2H (16 bits) Y1L (16 bits) Y1H (16 bits) Y2L (16 bits) Y2H (16 bits)	Sample mode data 32 bit/channel data	Timeslot A and B with restrictions if both selected

In the above table G and I are the same as output on the adux1020. Timeslot B only supports sample mode data formats.

The format for G is as shown:

Bits	value
15:8	# of gesture sample points
7	1 indicates “click”
6	1 indicates “up”
5	1 indicates “down”
4	1 indicates “right”
3	1 indicates “left”
2:0	0

FIFO details

The FIFO has been expanded to 128 bytes. It can be used to store the following combinations data types:

Data	**Comment**	**Timeslot A data mode**	**Timeslot B data mode**
Timeslot A gesture values	Stores 4 or 8 byte gesture value	2,5	0
Timeslot A proximity values	2 bytes I value –or– 6 bytes X,Y, I value	1,3	0
Timeslot A sample values	8 bytes of sample data x1,x2,y1,y2 –or– 6 bytes of sample data X,Y,I	4,3	0
Timeslot B sample values	Timeslot A must not use FIFO in this mode	0	4
Timeslot A and Timeslot B sample values	2,4,8 or 16 bytes of Timeslot A sample data followed by 2,4,8 or 16 bytes of Timeslot B sample data	1,2,4,6 **Both Timeslots must have same decimation rate.**	1,2,4,6 **Both Timeslots must have same decimation rate.**

Other combinations of data modes are not permitted.

The adux1020 test mode which shows packet start info in LSBs is retained. This is controlled by the r_packet_start_en bit. Setting this bit to 1 puts a packet marker on the lsb of all data. A 1 indicates the first word of a packet.

The FIFO can be programmed to not push incomplete packages setting the r_fifo_prevent_en bit. When putting both A and B sample data in the FIFO, and this bit is set, the data will only be written if all 16 bytes of sample data will fit.

Care must be taken to read the packets from the FIFO as complete groups. When both A and B timeslot data is being written, always read the whole packet. Using either the B fifo-timed sample interrupt or an appropriate value for the FIFO threshold will indicate when to read the packet.

Proximity sample on/off mode

Added the R_PROX_SAMPLE_EN register bit. When this bit is set to 0, the proximity mode operates as it did on the adux1020, output a single proximity output at each interrupt. When set to 1, proximity outputs will be written at each point between the on-1 and off-1 interrupts. The value output depends on the dataout_mode_a value which must be set to either mode 1 or mode 3. There is no interrupt associated with the samples in this case. Use the fifo_th setting to generate a FIFO interrupt or poll the FIFO depth.

Gesture mask features

The new 5-bit register R_GESTURE_MASK will be used to limit gesture interrupts and data to only those corresponding to selected gestures. Setting the mask bit will allow the gesture to be reported and generate an interrupt if enabled. If the associated mask bit is 1, that particular gesture will be ignored. There are bits for up, down, left, right, and click as shown:

Bit	Meaning
4	Click
3	Y positive (up)
2	Y negative (down)
1	X positive (right)
0	X negative (left)

In addition the 8-bit R_MAX_GESTURE_LENGTH register has been added with a default of 0x40. This sets the maximum gesture length that will be identified. Gestures longer than this will be dropped.

Interrupts

R_INT_STATUS will be expanded as shown:

Bit	Description
0	Prox on1 interrupt
1	Prox off1 interrupt
2	Prox on2 interrupt
3	Prox off2 interrupt
4	Gesture interrupt
5	Sample interrupt_a
6	Sample interrupt_b
7	Watchdog interrupt

This provides a sample interrupt for timeslot B in addition to the previous interrupts. The fifo count greater than threshold interrupt doesn't have a separate status bit, the count is directly accessible and that interrupt automatically gets cleared when the fifo goes below the threshold value.

Interrupt pin control

The R_INT_POL, R_INT_OD and the R_INT_OE register bits allow control of the polarity, open drain, and output enable of the INT pin to allow sharing of the signal. These register bits are combined with the data value to drive the oe and data signals to the AFE to achieve the desired drive/polarity. The following table shows the drive of the pin based on the mode bits:

Internal Int value	R_INT_POL	R_INT_OD	R_INT_OE	Pad drive	Comment
X	X	X	0	Float	Disabled
0	0	0	1	0	non inv drive 0
1	0	0	1	1	non inv drive 1
0	1	0	1	1	inv drive
1	1	0	1	0	inv drive
0	0	1	1	Float	non inv open drain 0
1	0	1	1	1	non inv open drain 1
0	1	1	1	Float	inv open drain 0
1	1	1	1	0	inv open drain 1

Data registers only updated when final decimated values arrive

The datapath will only update the data registers (0x60-0x7f) when fully decimated values arrive. The adux1020 would update these after each adc sample, regardless of decimation value.

New bits to control sample interrupt timing

The new control bits r_samp_int_type_a and r_samp_int_type_b allow the selection of when sample interrupts occur. When set to 0, the sample interrupt occurs when the data is updated in the data registers. When set to 1, the sample interrupt occurs when the data is written to the fifo. If the datamode selected doesn't write to the fifo, be careful to use the use the data register timing or an interrupt will not be generated.

New bits to lock register updates during i2c reads

The new bits r_hold_regs_a and r_hold_regs_b have been added. This feature was added to allow random time accesses to the registers rather than using interrupts to access registers between samples. While these set, the chip will not update the data register outputs for readback over the i2c bus. This will ensure that the registers can be read as a consistent group, rather than being possibly partially updated. The timeslot A and B registers can be prevented from being updated independently by the appropriate bit.

To use this feature, simply set the appropriate hold_regs bit, read the desired data registers, and then clear the bit. No updates will occur when the bit is set. If samples arrive during the period when updates are blocked, the update of only the data register will be lost. These control bits have no effect on what gets written to or read from the FIFO, or used by the gesture and proximity engines.

Data registers

The following data registers are added:
X1_B ,X2_B, Y1_B, Y2_B,
READ_X1L_B, READ_X2L_B, READY1L_B, READY2L_B
READ_X1H_B, READ_X2H_B, READY1H_B, READY2H_B
READ_MAX_DIDT, READ_MIN_DIDT

The registers X1_a, X2_a, Y1_a, Y2_a, X1_b, X2_b, Y1_b, Y2_b, I, X, and Y always contain the 16 bit decimated and saturated sample values. These registers are intended for customer use to read samples when not using the FIFO. Either use the sample interrupt to access the registers at a safe time, or use the hold_regs feature to get a consistent group of values. These data registers get updated as a group per timeslot (A all at once, B all at once) so using either method works correctly.

The extended range registers READ_X1L_a, READ_X1H_a ….. READ_Y2H_b maintain the 27 bit accumulated value from the decimation function. Their value is controlled by the r_rdout_mode control bit. Set r_rdout_mode to 0 to get unshifted data. Set it to 1 to get the appropriate shift based on the decimation setting.

Miscellaneous changes

Die shape requirement

The die needs to have an approximate die size of 1.4mm x 2.5mm to fit targeted 5.6mm x 2.0mm module.

Chip id and rev code

This will be programmable as a metal change only. The chip id and rev code is shown in the detailed register map.

New bit to control clock for FIFO access

A new bit r_fifoclk_force has been added as a user mode register. This bit is used by users to enable the clock for the fifo, without having to use or disclose our test register clock power bits. Write 1 to enable the clock when doing fifo accesses. Set it to 0 to have the clock run only when internally required.

Decimate blocks

The decimate blocks have been duplicated for timeslot B. In addition, the decimation will be increased to support 64 and 128 samples if die space permits. The decimate blocks will reset on mode changes. If using the FIFO to store both Timeslot A and Timeslot B samples, the decimation rate must be set the same for both time slots. Here are the registers:

Old adux1020 register	New adux10xx registers	Comment
R_PROX_DEC_MODE (1,2,4,8,16,32)	R_PROX_DEC_MODE (1,2,4,8,16,32,64,128)	For timeslot A only. Same number of bits, additional settings
R_GEST_DEC_MODE (1,2,4,8,16,32)	R_GEST_DEC_MODE_A R_GEST_DEC_MODE_B (1,2,4,8,16,32,64,128)	Each timeslot has separate register

R_Dsample_time and R_Digital_time

The r_dsample_time and r_digital_time registers have been removed. The timing functions previously provided by these registers will be automatically generated by the hardware.

VDD preconditioning of PD during sleep

The register R_pd_vdd_disable configures the function as shown in this table:

Value	PD Inputs 1-4 during sleep	PD Inputs 5-8 during sleep	Comment
0	Connected to VDD	Connected to VDD	Default
1	Connected to matrix (floating)	Connected to VDD	
2	Connected to VDD	Connected to matrix (floating)	
3	Always connected to matrix	Always connected to matrix	Operates like adux1020

Switches controlled by sw_5to8_vddi and sw_1to4_vddi are used to connect the PDs to Vdd during sleep times and are driven according to the mode selected in the above table.

Earlier qreset

The new register R_QRESET_OFFSET is a 6-bit register in R0/R1 and 8 bits in R2 which is used to specify the delay to the start of the qreset pulse in 1µs increments. The qreset signal is 1µs long. The offset is used for both timeslots in R0/R1 and separately controlled per timeslot in R2. The qreset signal will pulse at the beginning and at the end of sampling.

Analog Offset Cancellation

A new analog circuit is added to optionally cancel offset. It is controlled by these registers:

R_AOCANCEL_OFFSET_A	7 bits (R0/R1) 8 bits (R2) (1µs steps)
R_AOCANCEL_WIDTH_A	7 bits (0 = no pulse) 31.25ns steps otherwise
R_AOCANCEL_OFFSET_PROX_A	7 bits (R0/R1) 8 bits (R2) (1µs steps)
R_AOCANCEL_WIDTH_PROX_A	7 bits (0 = no pulse) 31.25ns steps otherwise
R_AOCANCEL_OFFSET_B	7 bits (R0/R1) 8 bits (R2) (1µs steps)
R_AOCANCEL_WIDTH_B	7 bits (0 = no pulse) 31.25ns steps otherwise

The aocancel signal is pulsed once for each LED or modulate pulse. The pulse is delivered on the afe_qilp signal. It is recommended to align the AOC pulse with the AFE integrate clock. **THIS FEATURE IS DISABLED IN R3.**

PD array conditioning with Vcm

Switches in the matrix have been added to allow the pre-conditioning of the PDs with the Vcm to help reduce sampling offset. The signal to drive the switch matrix is sw_vcmi. It is controlled by the R_VCM_WIDTH register. The same value is used for both timeslots. The register indicates the number of µs that the array is held to Vcm. R_VCM_WIDTH is a five bit register with 1µs resolution. If set to 0, no pulse is made. The array connection signal sw_modi overrides this signal.

The register R_VCM_FLOAT_EN can be used in R0/R1 to select either clamping the PD to VCM when not modulating or have it float. In R2 the r_modulate_type_a and r_modulate_type_b registers are used to control the preconditioning type.

In the case of LED operation, it doesn't matter whether this is set, since the PD is connected to the AFE except during the preconditioning time. In the case of R0/R1 modulate operation, if R_VCM_FLOAT_EN is set to 1, the PD will float between the preconditioning pulse and the access. If set to 0, the PD is held to VCM except when the PD is connected to the AFE which only occurs during the modulate pulse. See the table below for R2 preconditioning.

New registers R_DISABLE_SWMOD_A and R_DISABLE_SWMOD_B can be used to hold the sw_modi signal high during the selected timeslot to fully disable this feature.

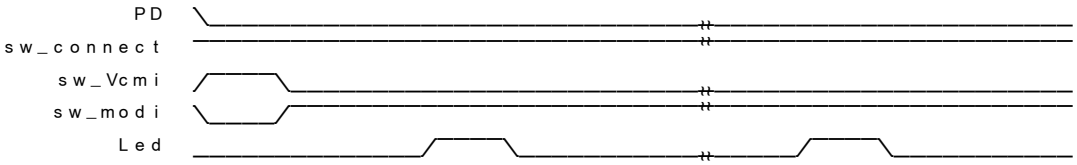
The following diagrams have been updated for R2.

In R2 there is a new signal (sw_connect), which is used to allow the photodiodes to float during certain access modes. The vcm_float_en bitfield has been replaced by the r_modulate_type_a and r_modulate_type_b bitfields. These new bits control which type of led/modulate function is used in timeslot a and timeslot b as shown in the following table.

r_modulate_a or modulate_b	r_modulate_type_a or r_modulate_type_b	description
01, 10, 11	xx	Precharge PD with VCM voltage. Leave pd connected. Flash selected Led.
00	00	ALS VCM type operation. Precondition with VCM voltage. Hold PD to VCM until modulate pulse. Pulse connection to PD with the sw_mod signal.
00	01	ALS float type operation. Precondition with VCM voltage. Float after vcm pulse. Pulse connection to PD using the sw_connect signal.
00	11	ALS float type operation. Precondition with TIA amplifier. Float after precondition. Pulse sw_connect to make PD connection.
00	10	reserved

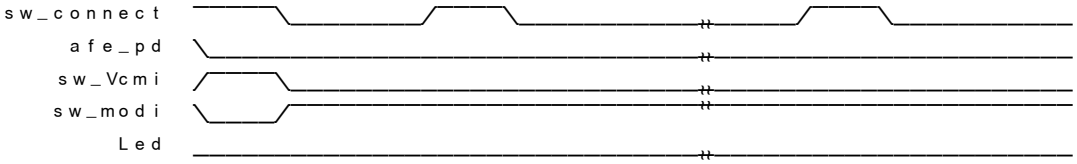
Here are the relevant cases for operation of the sw_connect, sw_vcmi, and sw_modi signals:

Case 1: Preconditioning, access PD after precondition. Typical gesture operation.
Vcm_width !=0 and modulate set to 01,10,or 11.

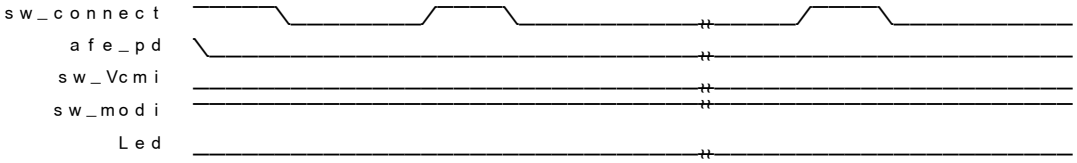


Case 2a: Preconditioning, access PD with modulate pulses. ALS – VCM type operation.
Modulate_type = 00, Vcm_width != 0 and modulate = 00 (no led flash)

Case 2b: Preconditioning to VCM, access PD with modulate pulses. ALS – float operation.
Modulate_type = 01, Vcm_width != 0 and modulate = 00 (no led flash)

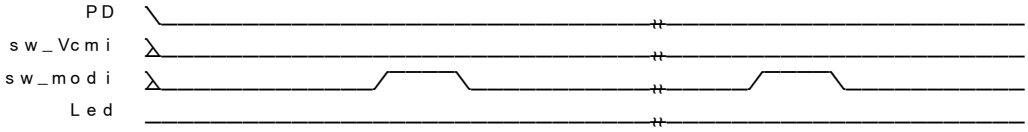


Case 2c: Preconditioning to TIA amplifier, access PD with modulate pulses. ALS – float operation.
Modulate_type = 11, Vcm_width != 0 and modulate = 00 (no led flash)

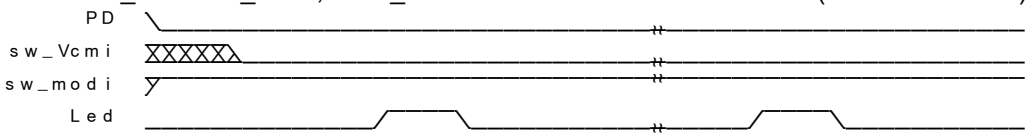


Case 3: No preconditioning, access PD. Operation like adux1020
Vcm_width==0 and modulate is not set (led flash instead)

Case 4: No preconditioning, access PD with modulate pulses. ALS with no precondition.
Vcm_width == 0 and modulate is set (no led flash)



Case 5: DISABLE_SWMOD_A set, VCM_Width != 0 and modulate is not set (led flash instead)



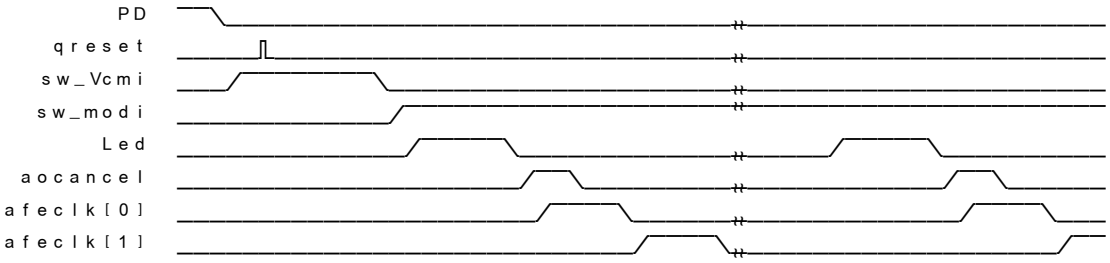
Cases 1 and 2 are the most likely useful cases, with the preconditioning enabled for both timeslots, one with LED flash and the other with modulate pulses. This is the expected use case.

Cases 3 and 4 are fall backs which allow adux_1020 like functionality. Case 5 is useful to allow the ability to not use the vcm/swmod function in one timeslot (like adux1020) and still use the modulate function with vcm conditioning in the other timeslot.

Example Timing Diagrams

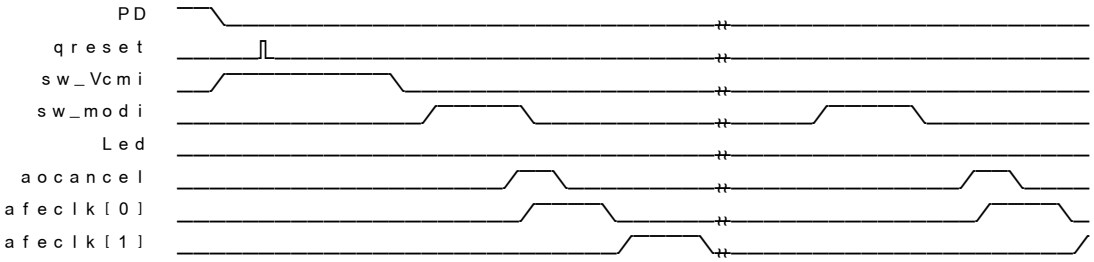
Here are a few example diagrams to show typical cycles of operation with the additional new signals.

Example Flow with LED Pulse(s)



In the ALS type operation, the LED is not pulsed but instead the connection of the PD to the AFE is pulsed.

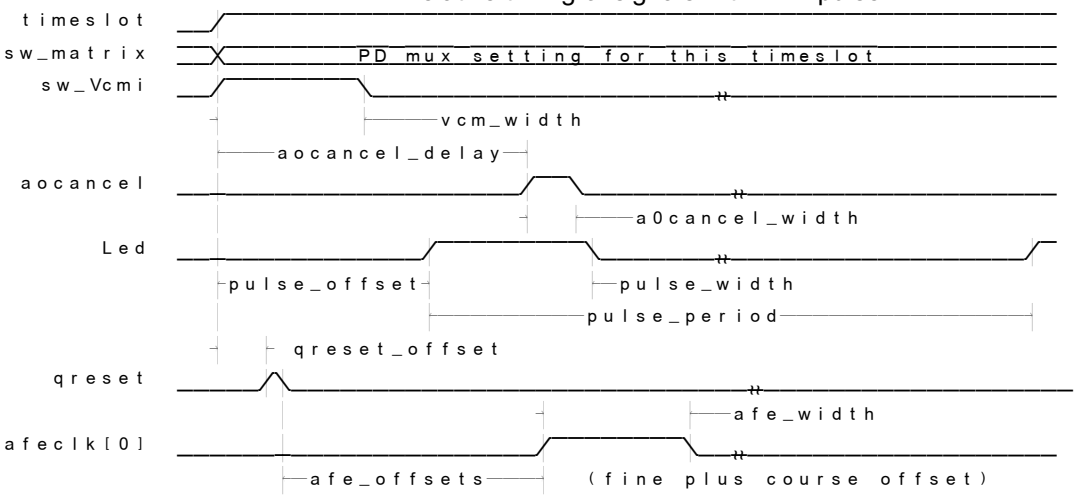
Example Flow with Modulate Pulse(s)



Relative timing of generated signals:

This diagram shows the relationship of the different generated pulse signals versus the start of a timeslot.

Relative timing of signals with LED pulse



Additional AFE power controls

There is a new 6 bit register R_AFE_PD_MASK. The three lsbs control channel 1, the 3 remaining bits control channels 2-4 together. The mapping of the control bits to specific op-amps is detailed in the register map

New Functionality in R2

Alternate subtract function

A new feature has been added to allow the calculation of the difference of the sample values for the first two pulses in a timeslot. This is used both for the new calibration function or to allow a two pulse ALS difference calculation to be used with different float times associated with the first and second pulses. Since the first float is controlled by the offsets, and the second float is controlled by the period setting, the difference between the two can be calculated.

New calibration function

A new feature has been added to allow gain calibration. When enabled by the r_gaincalibrate_a_en or r_gaincalibrate_b_en bitfields in the CALIBRATE register, a circuit is enabled which which cause additional charge to be injected into the first pulse sample during the selected timeslot. When combined with the r_alternate_sub_a_en or r_alternate_sub_b_en function, the amount of signal caused by the added chage can be measured. This fixed amount of charge can be used to determine a gain correction term.

Efuse

The ADI efuse 16x8 block has been added to optionally allow part specific information to be stored. The EFUSE_CTRL register has been added to control the use of the Efuse block. The register r_efuse_en controls the use of the Efuse block. When it is set to 0, the Efuse block remains held in the low power reset state.

To enable the Efuse block, first enable the 32MHz clock by asserting the r_fifoclk_force bit. Next the Efuse block is enabled by setting r_efuse_en to 3. Register access to the Efuse shadow and control registers is controlled by the r_efuse_reg_en bit. The registers for the Efuse block, when enabled appear in the same locations as the data registers.

As the Efuse block exits the reset state it copies the current fuse values to the shadow registers. The Efuse block must be in the on state to allow any operations. Please refer to pages 7-10 if the Efuse integration guide for instructions about proper use of programming, bist, and other features.

The shadow registers will hold their value even if the efuse register access is disabled. Also, the standby mode can be used to put the efuse block in a low power state without triggering a refresh operation when enabling the Efuse block.

The programming operation would follow a procedure like this:

- Enable Efuse block (r_efuse_en=3, r_efuse_reg_en=1)
- Enable 32MHz clock (r_fifoclk_force=1)
- Wait for refresh to complete (look for REF_DONE in NVM_STATUS0 register)
- Issue efuse command test unprogrammed fuses (bist operation)
- Unlock shadow registers
- Write shadow registers with desired contents for fuses
- Issue a program command

Normal read operations would just follow the first three steps shown above.

Timeslot B qreset, vcm controls.

The QRESET_VCM_B register has been added to allow timeslot b to have different vcm and qreset controls.

Additional AFE test power controls

The AFE_POWER_CTRL register has been added to allow some additional test options for the AFE power. See the register description for details

Photodiodes driven to cathode voltage

Unused photodiodes are connected to the driven cathode voltage when not being accessed or intentionally floating.

CMCLK changes

The r_cmtype control register has been added to control when the CMCLK runs. The new default is for it to run only during the VCM pulse time.

New Functionality in R3

The R3 release includes a number of fully compatible changes to increase functionality and performance. The changes are made such new features need to be actively enabled, R2 register settings should still remain consistently functional in R3.

Third LED added

A third LED driver and related control was added. Please see the LED driver section for details.

Cathode voltage flexible control.

The AFE_POWER_CTL register (0x54) has new control signals which allow more control over the cathode voltage during operations and sleep. The new feature is enabled with the r_flex_cathode control bit. The default for this bit is 0, which causes the cathode to be selected by the r_cathode_sel bit in the AFE_CTRL2 (0x3c) register. When this bit is set to 1, the new functionality is enabled and the voltage for the cathode pin is controlled by the r_tsa_cathode, r_tsb_cathode, and r_sleep_cathode register fields according to the following table:

code	voltage
00	VDD
01	AFE_VREF
10	V_DELTA
11	GND

I2C read register auto-increment

The I2C interface has been modified so that the register address increments automatically if additional byte read cycles occur during normal I2C register read operations. If reading addresses 0x5f, 0x60, or 0x7f the address will not increment. This feature allows for much lower overhead in reading results from the data registers. The write access does not have the auto increment feature.

Extended precision FIFO operations

The datamodes for sample mode FIFO operations have been expanded to include 32 bit sample data. The r_dataout_mode_a or r_dataout_mode_b fields in the OP_MODE_CFG (0x11) register have been

expanded to include 32 bit formats. The previous restrictions on data rates still apply when both timeslots are writing to the FIFO, but they can have different dataout_mode formats. Both timeslots must write the fifo at the same data rate. The user must ensure they read the entire packet of data consistently. The extended precision data written to the fifo follows the r_rdout_mode bit in register OP_MODE_CFG (0x11) to control if the data is output as the block average or block sum.

CM Clock change

A new option has been added to allow limiting the CM clock operation to the first 0-7 µsec of the precondition time. It has been determined that there is a settling transient caused by the shutting off of the CM clock which can cause the first pulse in a float type operation to have a slightly different starting point than subsequent pulses. This new mode by default enables the cmclock to be 32Mhz for the first 2 µsec of the precondition time. The new functionality is controlled by the r_cmclk_width field in the AFE_POWER_CTL (0x54) register which defaults to 2. The previous controls for the CM clock in the AFE_CTRL2 (0x3c) register function as previously described.

TIA trim per channel

An option to allow the 4 input channels to have different TIA resistor selections has been added in the new AFE_CHANNEL_TRIM (0x55) register. This new functionality is enabled by the r_afe_trim_tia_pc_en_a in the AFE_TRIM_A (0x42) and r_afe_trim_tia_pc_en_b in AFE_TRIM_B (0x44) register fields. The new functionality is disabled by default. When the enable bit is 0, all four channels follow the r_afe_trim_tia_x field in the associated AFE_TRIM_x register. When the enable bit is 1, channel 2-4 get their settings from the new AFE_CHANNEL_TRIM register fields and channel 1 uses the original bits.

TIA new gain setting

A 25K resistor setting has been added to the tia trim settings replacing one of the duplicate 50K resistor settings. This is available in the original register and the per channel settings.

New add/subtract options

The two pulse add/subtract options have been expanded to allow different add and subtract choices for up to 4 pulses. The first two pulses will operate according to the previously defined r_alternate_sub_a_en and r_alternate_sub_b_en fields in the CALIBRATE (0x58) register. This allows current “two-pulse” subtract and “calibrate” operations to operate as before. In addition the r_alternate_sub_ext_a_en and r_alternate_sub_ext_b_en allow the choice of operations for the third and fourth pulses. By appropriately setting these two fields any combination of adds and subtracts can be performed on groups of four pulses. This is useful for the newly supported float-gesture mode to allow the sub-add-add-sub sequence desired for that operation. When there are more than four pulses in a given timeslot, the pattern repeats.

Shorten delay between timeslot A and B

In sample mode the calculation time after timeslot A has been reduced by 64 µsec. The calculation time remains unchanged if using proximity or gesture modes. No register changes are necessary.

Timeslot order control and Go mode sleep option

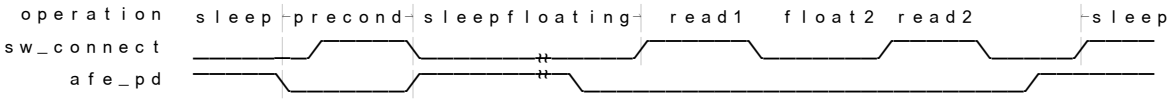
An option has been added to allow the order of timeslot operations to be swapped. This is necessary mainly for the new “sleep-float” operations, but is separately enabled by the r_bfirst field in the SLEEP_FLOAT (0x38) register. When set, the order of timeslot operations is timeslot b then timeslot a. The r_go_sleep field controls whether the first access when entering go mode starts with sleep cycle.

When this bit is set to 1, the part will do a sleep cycle before the first sample. This is mainly to support getting the first sample to be valid in the sleep float operation. When this bit is set to 0 (the default) the part immediately starts a sample cycle after entering go mode, like the previous revisions.

Sleep Float function

The sleep float function has been added in order to provide longer float times without incurring a power penalty. The sleep float function operates by doing a photodiode precondition cycle followed by floating during the end of the sleep cycle. By keeping the photodiode floating during this part of sleep and also what would have been the normal precondition cycle, the float time for the first pulse is extended beyond the normal offset. The analog section remains powered down during the float time so there is no additional power burden. A sleep float cycle looks like this:

Preconditioning to TIA amplifier, ALS – sleep float operation.



The duration of the precondition time is a total of five 32KHz clock cycles, with four cycles being used for our normal startup sequence and one for the actual photodiode preconditioning. The amount of sleep-float time is determined by the r_sleep_float_time field in the SLEEP_FLOAT (0x38) register. It is specified in increments of 125 µsec. There will be a small up to 1 µsec uncertainty of the total float time in this mode caused by clock domain crossing synchronization. The sleep time is determined by this formula:

Sleep Float time = (125us * r_sleep_float_time) + pulse_offset + (9 to 10) us

The sleep float function is enabled by setting either the r_sleep_float_a_en or r_sleep_float_b_en fields in the SLEEP_FLOAT (0x38) register. There are some requirements to allow sleep float operation:

- The timeslot enabled must be the first to operate. Use the r_bfirst field to change the timeslot order if necessary, or only enable one timeslot.
- There must be adequate sleep time to allow the selected amount of sleep float. This is controlled by both r_sleep_float_time and the sample rate.
- The enabled timeslot must be configured to do a float type access
- For the first sample taken after enabling the mode, the r_go_sleep field should be set.

There are no restrictions to the operations in the timeslot that is not operating in sleep float mode.

Uniform qreset to integration delay

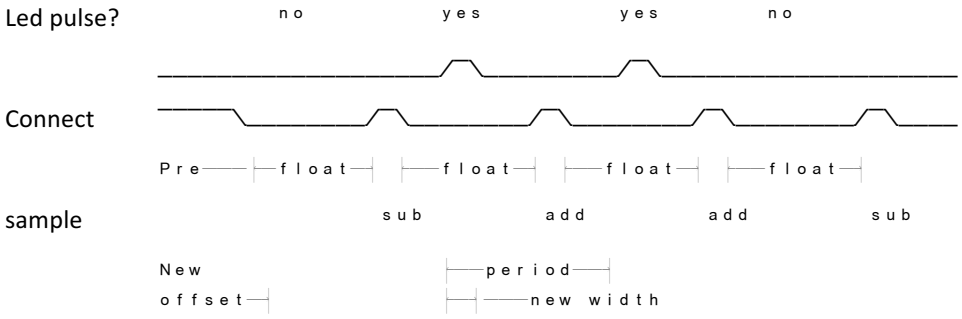
A new mode has been added to make the qreset to integration delay uniform for all pulses. This mode, which is disabled by default, is selected by setting the r_ed_qreset_en register field in the TIMING_OPTIONS (0x5a) register. When this bit is set, the 1 µsec qreset occurs 1usec before the afe_clk for each pulse cycle. The delay between the end of the qreset pulse and the afe integration clock is (r_afe_fine_offset_x+1) * 31.25ns. Contrary to what was previously reported, there are no additional timing limitations cause by the use of this mode.

Float Gesture mode

Float gesture mode was added as a way to combine doing a float type access with a flashing LED. This is intended to be used with the gesture sensors to allow more sensitivity which could provide a better range vs LED power tradeoff. The basic operation in this mode is a float style access being done in a four pulse

sequence. A new float-gesture LED pulse generator allows a the generation of an LED pulse with controlled offset and width only during selected pulses in this four pulse sequence. Combining this with the four pulse add/subtract feature allows the partial removal of background signal.

An example of a float gesture sequence:



The new AFE_FGL_A (0x3e) and AFE_FGL_B (0x3f) registers have been added to specify parameters for float gesture mode. The AFE_FGL_A register contains the r_fgl_pulse_width_a, r_fgl_pulse_offset_a and r_fgl_led_sel_a fields to control the width, offset, and which LED to flash in this mode. Like named fields exist in AFE_FGL_B to control float gesture in timeslot b. In addition the register fields r_fgl_pulse_mask_a and r_fgl_pulse_mask_b fields have been added to the TIMING_OPTIONS (0x5a) register to select which of the four float periods will have an LED flash enabled. The add/subtract fields are controlled by the fields previously described in the CALIBRATE (0x58) register. Be careful that the order of bits that describe the LED pulses is not the same as the order of bits which describe the add subtract fields.

The LED flash specified by the above registers will only occur if the selected timeslot is operating in modulate=00 mode and if an LED is selected in the appropriate r_fgl_led_sel_a or r_fgl_led_sel_b field.

Digital Integrate and support for the 221 amplifier

Digital integrate mode has been added to support applications using the 221 amplifier. In this mode, the 221 amplifier is attached to all four photodiode inputs, all four channels operate in parallel and provide one sample channel of data on channel 1. In this mode, the bandpass filter is bypassed, the analog integrators are converted to buffer amplifiers, and the digital engine is used to do the integration function.

To enable the digital integrate function:

- Set the r_digital_integrate_a_en or r_digital_integrate_b_en bits in the CALIBRATE (0x58) register.
- Set the r_afe_mux_test_a or r_afe_mux_test_b field to 0xAE65.
- Set r_afe_trim_int_a or r_afe_trim_int_b to 00101, 00110, or 00111
- Set the pulse and modulate values as in normal mode. Must use “multi-sample” (digital accumulate) mode
- Set the afe_offset and width to specify the digital integration window.

In digital integration mode, the ADC does a conversion every 1µs during the integration window and the digital engine adds or subtracts the sample. At the end of the digital integration window the resulting sum is sent along to the decimate unit as the sample for that pulse. There is a total of one sample per pulse.

The digital integration window starts at the afe_offset + qreset_offset +1. The digital integration window is made up of three regions, the first is a subtract region of width afe_width during which samples are

subtracted. Next two addition regions, each afe_width long occur. This is followed by an additional subtract region also with a duration of afe_width.

New Functionality in R4

The R4 release includes a number of fully compatible changes to increase functionality and performance. The changes are made so that new features need to be actively enabled, R2 and R3 register settings should still remain consistently functional in R4.

Address feature to allow multiple devices on same I2C bus

The I2C_ID_KEY register has been changed to allow changing the I2C slave address dependent on either or both of the INT and TCLI pins. This allows multiple devices to reside on the I2C bus by having the appropriate value placed on the INT or TCLI when writing the key. The KEY values are as follows:

KEY	Condition for address change	Function
0x04AD	Ignore state of INT,TCLI	change slave address in all devices
0x44AD	INT must be high, ignore TCLI	only change devices with INT high
0x84AD	TCLI must be high, ignore INT	only change devices with TCLI high
0xC4AD	INT and TCLI must be high	only change devices with INT and TCLI high

To use this function follow this sequence:

- Enable the associated input buffer (using the MISC register at address 0x4F)
- Set the INT and/or TCLI pins to select the device to modify
- Write the key to match the desired function
- With no intervening writes, write the I2CS_ID register with the desired address in bits 7:1, 0xAD in the upper bits, and the lsb set to 0.
- Repeat as necessary to modify all devices in the system

External Sample Sync support

An external sample sync can be provided on either the INT or TCLI pins. This functionality is controlled by the R_EXT_SYNC_SEL field in the MISC register. The following table shows the function:

R_EXT_SYNC_SEL	Function
00	Use on chip 32KHz sample timer (default)
01	Use rising edge on INT pin to trigger timeslot start
10	Use rising edge on TCLI pin to trigger timeslot start
11	reserved

When enabled, a rising edge on the selected input will be used to specify when next sample cycle occurs, after a synchronization and the usual startup delay. This overrides the usual 32KHz sample timer. To use this feature:

- Enter IDLE mode
- Write the R_EXT_SYNC_SEL and the matching input enable in the MISC register to select the desired input
- Set the R_GO_SLEEP field to 1 in the SLEEP_FLOAT register. This avoids having the first sample start immediately when entering GO.
- Enter GO mode

An async edge detector is used on this input to capture the rising edge to avoid requiring a multiple 32KHz cycle pulse. The output of this detector is then synchronized to the 32KHz clock domain causing 1

or 2 cycles of delay. This is then followed by the normal start up sequence of multiple 32KHz clock cycles. This sequence is the same as if the normal sample timer had triggered it.

The limit to sample period is controlled by the timeslot operations (in 32MHz domain) plus the 32KHz wakeup and sleep start cycles. The sleep float operations are not avalible in this mode.

New Data Modes

New data mode selects have been added to allow the following:

- 16 Bit sum of all 4 channels in sample mode
- 32 Bit sum of all 4 channels in sample mode
- 16 bit support for digital integration background value
- 32 bit support for digital integration background value

These data selections are detailed in the Data Modes table earlier in this document.

Digital Integration background values

The digital integration mode has been enhanced to provide background data samples. During all digital integration operations, the background (aka dark) sample sum will be written to the X2 registers and optionally written to the FIFO. The X1 register continues to contain the difference value. Both 16 and 32 bit output values are available in the data registers or can be written to the FIFO using the appropriate data output mode. The channel 1 gain and offset values are used for the background channel in X2 as well as for the signal value in X1.

Digital Integration single channel mode

The R_SINGLE_CHAN_DIG_INT_A and R_SINGLE_CHAN_DIG_INT_B fields are used to select single channel digital integrate mode. When these bits are set, only the X1 channel (pd1 or pd5) is used for digital integration. The other channels are powered down and ignored. The new mode bits R_DIGINT_TIA234_PD_EN, R_DIGINT_INT234_PD_EN, R_DIGINT_BPF234_PD_EN, and R_DIGINT_BPF1_PD_EN have been added to allow powering down the various amplifiers during single channel and four channel digital integration. The BPF bits control the power down of the band pass filters during all digital integrate cycles, the INT and TIA bits control the power for the integrators and tias during single channel mode digital integrate only.

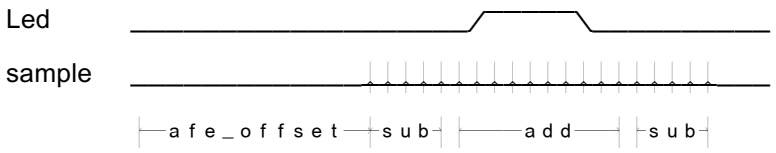
Digital Integration sample modes

The digital integration mode has been enhanced to allow additional sample modes. The number of sample regions and the space between the sample regions are now configurable.

The R_DIGITAL_INTEGRATE_GAP_EN field has been added to allow the specification of the space between the negative and positive summing regions. When enabled, the space is specified by the AFE_FINE_OFFSET fields.

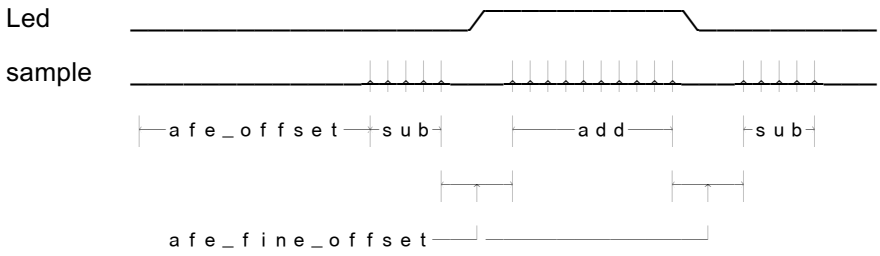
The R_DIG_INTEGRATE_ONESHOT_A and R_DIG_INTEGRATE_ONESHOT_B have been added to allow the specification of a single negative and positive sample region in place of the usual two. The following diagrams show the various sample modes:

Digital integrate with gap not enabled:



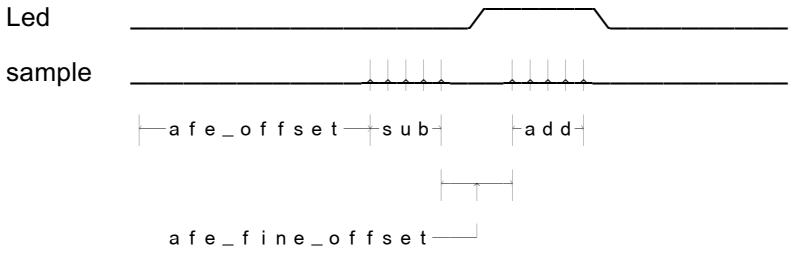
In this example there are two 5 sample dark subtract regions and one 10 sample add region. With this setup, the pulse response is captured including the rise and fall times. The add region must be set to capture the entire response.

Digital integrate with gap enabled:



In this example there are also two 5 sample dark subtract regions and one 10 sample add region. A gap is configured to avoid the rising and falling transistions.

Digital integrate with gap enabled in one-shot mode:



In this example there is a 5 sample dark subtract region and a 5 sample add region. A gap is also used to avoid the rising and falling transistions. There is no dark sample after the pulse. This mode can be used to gain the equivalent function of TIA-ADC mode.

Integrator as buffer option for TIA-ADC modes

The R_AFE_INT_AS_BUFFER_A and R_AFE_INT_AS_BUFFER_B control bits have been added to allow the integrator to be configured as a buffer with gain of 1 or 2 to be used in TIA-ADC mode, without having to use the digital integration function.

Data test pattern mode

The R_USE_DATA_TESTPATTERN field has been added to the TESTMODE register to select a constant data field rather than the actual ADC data. This is useful, in combination with the offset, gain, and data mode fields to provide a constant data pattern for software debug. When set, the contents of the

r_prox_th_on1 register is used for all ADC accesses in place of the real data. This is only useful in sampling modes.

Led testmode method

The steps to use the INT pin to drive the LEDs in test mode are as follows:

- Enable 32K oscillator
- Enter IDLE
- Enable INT pin to be used as an input
- Set desired LED settings (iref, itail values etc.) using registers LED1's settings 0x23 and 0x25
- Select LED to drive using TEST_PD (0x52) register bits [13:12] to powerup LED1, LED2 or LED3
- Configure TEST_MODE (0x50) register bits [15:14] to select the same LED pin (LED1/2/3) to be driven

LED1's settings are used in this mode to control the selected LED pin.

Digital to AFE signals

The sw_vcmi and sw_modi signals operate as in this table:

Sw_vcmi	Sw_modi	Connection
0	0	No PD connection No AFE connection
0	1	PD matrix -> AFE
1	0	Vcm -> PD matrix No AFE connection
1	1	PD matrix -> AFE

The control for the two LED signals are sent on the led_en[1:0] signals. The switch matrix controls are sent on the pdmux[6:0] bus = {sw_1to4_vddi, sw_5to8_vddi, sw_matrix[0], sw_matrix[1], sw_matrix[2], sw_modi, sw_vcmi}. The aoc pulse is sent on the afe_qilp signal. The control to select the cathode voltage is sent on the r_cathode_sel signal.

The control and trim settings which are Timeslot specific are muxed within the digital section and are sent on the previous used signals. The LED signals which are LED specific have LED1 and LED2 names. All other control signals remain the same.

R2 Bug fixes

The following bug fixes were made in the R2 release:

- Fifo clock force function fixed. The FIFO clock will now operate properly using the R_FIFOCLK_FORCE control bit in regiser ACCESS_CTRL (0x5F). It is no longer necessary to change the control bits in the TEST_PD (0x52) register to control the clock for the FIFO.
- Data mode 0 for gesture now correctly updates register, generates the gesture interrupt, and does not write to the FIFO.
- READMAXDIDT, READMINDIDT registers now update at the correct time and correctly follow the r_hold_regs_a control bit.
- The LED drive strengths are now correct and no longer require forcing either LED to be powered on or off.
- The proximity mode sample rate is now correct for all sample times.
- INT pin test outputs have are corrected and enhanced. See the r_test_mode bitfield in the TEST_MODE register (0x50) for more details.

Please consult the issue manager to see more details about these functional problems and workarounds for earlier revisions.

R3 Bug fixes

The following bug fixes were made in the R3 release:

- The problem preventing use of adc_spacing=0 in two pulse difference mode was fixed.

R4 Bug fixes

The following bug fixes were made in the R4 release:

- The problem restricting the pulse counts to be a multiple of 4 in float gesture mode has been fixed.

Not fixed in R2 or R3 or R4

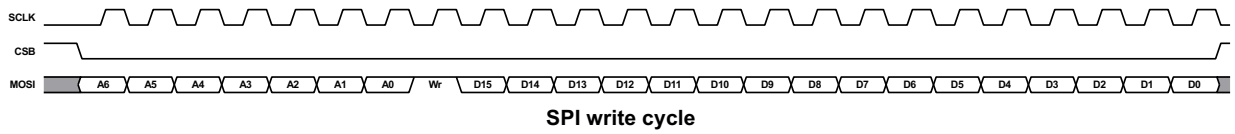
- Problems with the data flag and packet marking features were not fixed. These were removed in R4.

New Functionality in R5

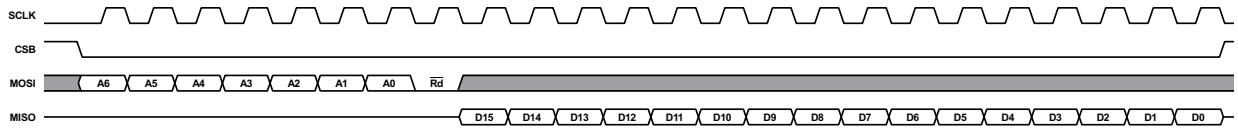
The R5 release includes a number of fully compatible changes to increase functionality and performance. The changes are made so that new features need to be actively enabled, R2, R3, and R4 register settings should still remain consistently functional in R5.

SPI interface has been added

An SPI optional interface has been added. In order to use the SPI interface, the I2C interface pins must both be held high. In order to use the I2C interface, the SPI pins should be held high. The expectation is that the RDL layer for the WLCSP type package will do these tie-offs to support two different part pinouts. If both are bonded out in a different package, the user must tie off one or the other as unpredictable results will occur if both interfaces are used simultaneously. The SPI write and read cycles are shown below:



SPI write cycle



SPI read cycle

Additional bytes read after the first pair will automatically increment the address except for addresses 0x5F, 0x60(FIFO), or 0x7F. The I2C interface functionality has not changed.

Random sleep mode

Random sleep mode has been added. When R_RANDOM_SLEEP is set to 1, the wakeup time is changed each cycle based on the output of an LFSR. This provides a pseudorandom sequence with each sample time being offset by +/- 7 cycles.

Ability to disable unneeded ADC cycles.

An ADC channel mask, R_ADC_CHAN_DISABLE[3:0] has been added to allow the unnecessary ADC cycles when using fewer than 4 channels to not interfere with having a shorter pulse period. An example use of this feature is when only a single channel is enabled, set the R_ADC_CHAN_DISABLE to 0xE to disable all but the first channel. Also the R_ED_QRESET_EN bit must be set to use this mode. This would then allow the pulse operations to start sooner, overlapping where the ADC cycles would have occurred. The PULSE_PERIOD_ registers in this example would only need to be greater than 2*AFE_WIDTH_+2 rather than the previously required 2*AFE_WIDTH_+5. Disabling a channel does not change the sample time order, it always goes in order from channel 1 to channel 4. This feature is only really useful for 1 and 2 channel operation when a shorter pulse period is desired.

Pulsed Vref mode

An additional mode which allows the vref value to pulse to a different value using the R_PULSE_WIDTH_ and R_PULSE_OFFSET_ values has been added. When enabled by R_PULSE_VREF_A or R_PULSE_VREF_B, the vref value will change to the value specified in the R_PULSE_VREF_VAL_A or R_PULSE_VREF_VAL_B registers.

GPIO configurations for INT and TCLI pins

The TCLI and INT pins now act as GPIO type pins. They identically support additional output values, in addition to their previous use as input and output signals. The R_INT_ALT_CFG and R_TCLI_ALT_CFG registers have been added to specify a variety of new output signals which can be used to trigger external events. Both pins can support any of the functions available. In addition, R_TCLI_OD and R_TCLI_POL have been added to provide the ability to invert or have open drain operations on the TCLI pin, as the INT pad had in previous revisions.

When one of the new output value is selected using R_INT_ALT_CFG or R_TCLI_ALT_CFG, the open drain and polarity values are also applied. The R4 functionality remains in place, if no new output value is selected. The following table shows the new options:

Code	Function	Clock Domain
0x0	0x0: do R0-R4 function	varies
0x1	INT function	1MHz
0x2	PD[9] (afe_vref powerdown)	32KHz(R5) 1MHz(R6)
0x3	ts_active_a	1MHz
0x4	ts_active_b	1MHz
0x5	pulse output (led or modulate) timeslot a only	1MHz
0x6	pulse output (led or modulate) timeslot b only	1MHz
0x7	pulse output (led or modulate) both timeslots	1MHz
0x8	float led pulse timeslot a	1MHz
0x9	float led pulse timeslot b	1MHz
0xa	float led pulse both timeslots	1MHz
0xb	aoc pulse	1MHz
0xc	output data cycle occurred in timeslot a	32KHz
0xd	output data cycle occurred in timeslot b	32KHz
0xe	output data cycle occurred	32KHz
0xf	odd timeslot (toggle)	32KHz
0x10	0	n/a
0x11	1	n/a
0x12	PD[0] (tia powerdown)	1MHz
0x13	32KHz osc output	32KHz
0x14	32MHz osc output	1MHz
0x15	1MHz (32MHz/32)	1MHz
0x16	dclk	1MHz
0x17	afe integrate clock 0	1MHz
0x18	afe integrate clock 1	1MHz
0x19	OR of r_bigdark_a bits	1MHz (R6 only)
0x1a	OR of r_bigdark_b bits	1MHz (R6 only)
0x1b	OR of r_bigdark_a, r_bigdark_b bits	1MHz (R6 only)
0x1c	OR of r_bigdark_a, r_bigdark_b, INT function	1MHz (R6 only)

No Clocking needed for FIFO read or clear

The FIFO circuit was replaced with a new version which does not require running a clock. Reading and clearing the FIFO both work without needing to turn on the 32MHz oscillator. It also works properly if the clock is running. There is a minor change to the interrupt logic for the fifo_th interrupt. This interrupt now gets cleared by any FIFO read, and gets set again only when the FIFO is written and the number of words is above the threshold. (R5 limitation: The FIFO interrupt is not cleared when the FIFO is cleared. To clear the FIFO in R5, stop sampling, read one or more bytes from the FIFO, and write the FIFO clear bit. This is not necessary in R6)

Cathode float during sleep option

The R_CATHODE_SLEEP register has been redefined to include a floating option. This is specified by using code 0x10.

Anode float option

The R_FLOAT_ANODE_SLEEP control bit was added to allow photodiode inputs to be disconnected from the input mux. This also requires setting the appropriate R_PD_VDD_DISABLE bit.

Cathode toggle option

The R_CATHODE_TOGGLE_EN_A and R_CATHODE_TOGGLE_EN_B register bits enable toggling the cathode voltage during the timeslot between the normally selected value and the value selected by R_CATHODE_TOGGLE_VAL_A or R_CATHODE_TOGGLE_VAL_B on alternating timeslots. For example, if R_FLEX_CATHODE is set to 1, R_TSA_CATHODE is set to 0, R_CATHODE_TOGGLE_EN_A is 1, and R_CATHODE_TOGGLE_VAL_A is set to 3, the cathode voltage will alternate between VDD and GND in alternating timeslot A periods.

Additional afe power control options

The R_AFE_AMPS2_DISABLE[2:0] and R_AFE_AMPS34_DISABLE have been added to allow separate control of the channel 2 amplifiers. Channel 3 and 4 are still controlled together. These disable bits work in concert with the previous control bits, which retain their functions.

Analog Feature changes:

The following changes have been made:

- Power-On Reset circuit changed to lower power version to reduce standby power from 4µA to below 1µA.
- The 32MHz oscillator has been replaced with a better design which uses less power, starts faster, and is more stable with voltage.
- The integrator input resistor (R4) has been doubled.
- The analog gain setting for digital integration has been changed from (1 or 2) to (1 or .6875). These gains also apply to “integrator as buffer” mode.
- The LED driver has been redesigned to have better 1/f noise performance.

R_AFE_TRIM_INT_[1:0] setting	R3,R4 gain	R5 Gain
00	1	1
01	1	1
10	2	.6875
11	2	.6875

R5 Bug fixes

The following bug fixes were made in the R5 release:

- Applying software reset in I2C mode now ACK's the cycle rather than stopping before the ACK cycle.
- It is no longer required to enable the internal 32KHz oscillator while using an external 32KHz oscillator input.
- "Integrator as buffer" mode now operates properly. However, in R5 the channel 1 integrator power controls the power for the other channels. (this is fixed in R6)

New Functionality in R6 (Van Buren)

The R6 release includes a number of mostly compatible changes to increase functionality and performance. Although the changes are made so that new features need to be actively enabled and is mostly compatible with old versions, the PD mux change and LED gain change is not fully compatible. Customers switching to this version may need to change the R_SW_MATRIX_A and R_SW_MATRIX_B register fields to account for the PD mux change and the R_LED1_en250, R_LED2_en250 or R_LED3_en250 registers to account for the 0.4x to 0.1x LED current multiplier change.

PD Mux change

The PD mux input logic has been replaced with a simpler matrix which uses a single switch to connect the various pd inputs to the various TIA channel inputs. This single switch connections replace the functionality of the SW_MATRIX, SW_MOD and CONNECT switches. This change provides better PSRR performance by using fewer series switches. The mapping of the PD inputs to the AFE channels is as shown in this table:

R_SW_MATRIX_	Connections	Comments
0	x1=n/c x2=n/c y1=n/c y2=n/c	Same as previous revs – No Connect
1	x1=pd3+pd4 x2=pd1+pd2 y1=n/c y2=n/c	Different from old revs
2	x1=pd7+pd8 x2=pd5+pd6 y1=n/c y2=n/c	Different from old revs
3	x1=pd1+pd2+pd3+pd4 x2=n/c y1=n/c y2=n/c	Different from old revs
4	x1=pd5 x2=pd6 y1=pd7 y2=pd8	Same as previous revs – straight through
5	x1=pd1 x2=pd2 y1=pd3 y2=pd4	Same as previous revs – straight through
6	x1=pd3+pd4 x2=pd5+pd6 c=n/c d=n/c	Different from old revs
7	x1=pd5+pd6+pd7+pd8 x2=n/c y1=n/c y2=n/c	Different from old revs

These mappings were chosen to meet the expected use models for wlcsp, lfcsp and module use. These R_SW_MATRIX_ values are combined with the previous modulate and connect signals to control the new single switch type pd mux.

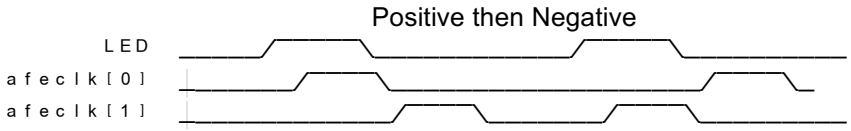
4 Pulse Integrator clock reversal option

This feature is added to provide what has been referred to as "new normal mode" which bypasses the bandpass filter. In this mode, the user uses an even number of pulses and sets up a sequence of positive and negative integrations by specifying which of each group of 2 or 4 pulses integrate in the opposite direction. This combined with the previously available 2 and 4 pulse subtract modes allows the user to effectively "chop" the signal path by adding the positive integrated cycles and subtracting the negative ones. This also has the benefit of removing the offsets.

The R_REVERSE_INT_ registers provide a four bit pattern specifying whether the respective integrate pulse pair in a four pulse group would integrate in the negative direction. When set to 0, the integration happens positive first then negative. When set to 1, the integration order is reversed with the negative integration happening first. With the bandgap filter bypassed, this has the effect of inverting the input

signal when the bit is set. By matching these inversions with the existing four pulse add/subtract controls, it is straightforward to add the positives and subtract the negatives and thus cancel offsets and reduce some types of noise. When there are more than four pulses in a given timeslot, the pattern repeats. The user must take care to only do a matching number of positive and negative integrates (and matching adds and subtracts) in order for this mode to be useful.

Example of positive and negative integrate operations controlled by the R_REVERSE_INT_ register



This diagram shows the effect of having the second pulse's R_REVERSE_INT_ bit set to 1. This would cause the second pulses current to be integrated in the negative direction. For this to be useful in normal mode, the user would select "add first pulse, subtract second pulse" for the math operation.

Ambient threshold checking during "subtract" cycles

One challenge of using the Float-LED modes has been the problem of not being able to observe a large amount of ambient light saturating the value and the inherent bad effect of the signal appearing to go down as the system saturates. This has been typically solved by doing a separate tia-adc mode reading of the ambient value in a second timeslot. A new feature has been added to solve this problem.

This new feature allows the user to be notified that the ambient value has risen above a user specified level for a specified number of ADC samples. After each conversion during a region which will be subtracted, the 14 bit ADC value, minus the channel's offset, is compared to the user specified value. If it is greater, the count is incremented. If the count advances to the user specified value, that channel's R_BIGDARK_ bit is set. This allows the user to simply read the R_BIGDARK_ registers to determine if the ambient level is getting too large and needs to be mitigated. Since only the subtracted samples are checked, it only operates on the ambient values. This feature is meant to be used in multi-pulse float modes using the difference feature.

The R_DARK_TH_ fields have been added to specify the threshold value to trigger a count. The R_DARK_CNT_ fields specify the number of samples over the threshold value to set that timeslot and channel's R_BIGDARK_ bit. The sample counters get reset when the R_BIGDARK_ register is read or in IDLE mode.

R_DARK_CNT_	Function
0	Never set R_BIGDARK_ bits
1	1 or more times (sample > R_DARK_TH_) sets R_BIGDARK_ bit
2	4 or more times (sample > R_DARK_TH_) sets R_BIGDARK_ bit
3	16 or more times (samples > R_DARK_TH_) sets R_BIGDARK_ bit

The R_BIGDARK_ register contain a bit for each input channel and timeslot that indicate if that channel has been over the threshold value enough times. Reading the R_BIGDARK_ register clears both the bits and the counters, as does reset and IDLE mode. This allows the user to poll that register at a given rate and detect if the specified number of over threshold values have occurred during the polling time. Also the R_BIGDARK_ registers can be output on the GPIO pin using these new output codes in the R_TCLI_ALT_CFG or R_INT_ALT_CFG register.

Code	Function	Clock Domain
0x19	OR of R_BIGDARK_A register field	1MHz
0x1a	OR of R_BIGDARK_B register field	1MHz
0x1b	OR of all R_BIGDARK_A and R_BIGDARK_B fields	1MHz
0x1c	OR of all R_BIGDARK_A and R_BIGDARK_B and INT	1MHz

Reduce power during digital processing time and sleep entry

The various analog circuitry which is not used during digital processing time will be shut off. In addition, the bandgap, bias, ref buffer, and led amplifiers will be turned off sooner prior to entering sleep.

Changes to reduce inter-channel coupling in ADC

Two new optional features are added which will reduce inter-channel coupling during ADC cycles and improve settling times.

The first is controlled by r_adc_dummy_cycle. This defaults to off. When set to 1, it enables an ADC cycle to occur in the cycle prior to the first channel. This will cause a missing “boost” cycle in the ADC to occur and should help settling of the next ADC cycle. No channel is selected during this cycle and the digital output is discarded.

The other change is controlled by r_adc_input_short. When set to 1, a reset pulse is asserted at the end of each ADC cycle which shorts the ADC inputs together and removes the previous conversions differential signal. This two bit field specifies the number of 32MHz cycles are used to short the input between ADC cycles, prior to asserting the next ADC select. This does not occur during digital integrate cycles.

The adc dummy cycle and short functions are enabled by default.

Default value changes

The following register bitfields have new defaults:

- r_ed_qreset (in reg 0x5A) now defaults to 1. This changes the qreset pulse timing to be the same for all pulses
- r_tsa_cathode and r_tsb cathode (in reg 0x54) default to v_delta
- r_flex_cathode (in reg 0x54) now defaults to 1. This causes the values of r_tsa_cathode (in reg 0x54) and r_tsb_cathode (in reg 0x54) to override the value of r_cathode cell (in reg 0x3c)

Analog Feature changes:

The following changes have been made:

- The switches have been redesigned to have both smaller resistance and less variation over voltage.
- A single switch between each pd input and tia channel input is used to combine the functionality of the connect, mod, and pd_mux switches.
- New PD mux selection codes as shown in the digital section.
- The lower led range has been changed from 40% to 10%.
- Removed unnecessary debug switches.
- Design changes to support better PSRR. Targeting 60dB
- Increase TIA FB cap from 0.9pf to 2.4pf
- Hardcode afe_trim_bpf_ to code 1.
- Hardcode INT cap to 6.3pf

R6 Bug fixes

The following bug fixes were made in the R6 release:

- Continuous ADC test mode has been fixed. It now uses the TCLI pin for adc output data and works for either spi or i2c modes
- FIFO threshold interrupt is now automatically cleared when clearing the FIFO by writing the INT_STATUS REGISTER.
- It is no longer required to enable the TIA in channel 1 to use integrator as buffer mode.
- Remove glitches in the ts_active_a and ts_active_b output signals.

Scan Information

To enter scan mode set the r_scan_comp and r_scan_method bits to match the mode desired and set r_scan_mode bit to 1. Set the r_scan_chain_length to 0x039d for compressed mode or 0x1209 for non-compressed mode. These are the R7 values. (R6 had a length of 0x1208 for non-compressed mode.)

The following pins are used for scan in R5 and R6 and R7

I/O Mode	Scan Clock	Scan Input	Scan Output
I2C	SCL	SDA	INT
SPI	SCLK	MOSI	INT

Register map summary:

Registers not mentioned above retain their previous adux1020 function.

Address	Name	Reset	Access
0x00	INT_STATUS	0x0000	R/W
0x01	INT_MASK	0x00FF	R/W
0x02	PAD_IO_CTRL	0x0000	R/W
0x03	SCAN_MODE	0x0000	R/W
0x04	DARK_STATUS	0x0000	R
0x06	I2CS_CTL_MATCH	0x0000	R/W
0x07	INTERNAL_STATE	0x0000	R
0x08	CHIPID	0x0916	R/W
0x09	I2CS_ID	0x00C8	R
0x0A	OS32M_CAL_OUT	0x0000	R
0x0B	GPIO_CTRL	0x0000	R/W
0x0D	I2CS_ID_KEY	0x0000	R/W
0x0F	SW_RESET	0x0000	R/W
0x10	OP_MODE	0x0000	R/W
0x11	OP_MODE_CFG	0x1000	R/W
0x12	GEST_FREQ	0x0028	R/W
0x13	PROX_FREQ	0x0320	R/W
0x14	PD_SELECT	0x0541	R/W
0x15	DEC_MODE	0x0600	R/W
0x16	DARK_TH_A	0x3000	R/W
0x17	INTEG_ORDER_A	0x0000	R/W
0x18	CH1_OFFSET_A	0x2000	R/W
0x19	CH2_OFFSET_A	0x2000	R/W
0x1A	CH3_OFFSET_A	0x2000	R/W
0x1B	CH4_OFFSET_A	0x2000	R/W
0x1C	DARK_TH_B	0x3000	R/W
0x1D	INTEG_ORDER_B	0x0000	R/W
0x1E	CH1_OFFSET_B	0x2000	R/W
0x1F	CH2_OFFSET_B	0x2000	R/W
0x20	CH3_OFFSET_B	0x2000	R/W
0x21	CH4_OFFSET_B	0x2000	R/W
0x22	LED3_DRV	0x3000	R/W
0x23	LED1_DRV	0x3000	R/W
0x24	LED2_DRV	0x3000	R/W
0x25	LED_TRIM	0x630C	R/W
0x26	GEST_SLOPE_TH	0x0000	R/W
0x27	GEST_CTRL	0x0700	R/W
0x28	GEST_THRESH	0x0000	R/W
0x29	GEST_SIZE	0x4000	R/W
0x2A	PROX_ON_TH1	0x0000	R/W
0x2B	PROX_OFF_TH1	0x0000	R/W

Address	Name	Reset	Access
0x2C	PROX_ON_TH2	0x0000	R/W
0x2D	PROX_OFF_TH2	0x0000	R/W
0x2E	PROX_TH1_HBYTE	0x0000	R/W
0x2F	PROX_TH2_HBYTE	0x0000	R/W
0x30	PULSE_OFFSET_A	0x0320	R/W
0x31	PULSE_PERIOD_A	0x0818	R/W
0x32	PULSE_OFFSET_PROX_A	0x0320	R/W
0x33	PULSE_PERIOD_PROX_A	0x0818	R/W
0x34	PULSE_MASK	0x0000	R/W
0x35	PULSE_OFFSET_B	0x0320	R/W
0x36	PULSE_PERIOD_B	0x0818	R/W
0x37	PULSE_PERIOD_EXT	0x0000	R/W
0x38	SLEEP_FLOAT	0x0000	R/W
0x39	AFE_CTRL_A	0x22FC	R/W
0x3A	AFE_CTRL_PROX_A	0x22FC	R/W
0x3B	AFE_CTRL_B	0x22FC	R/W
0x3C	AFE_CTRL2	0x3006	R/W
0x3D	AFE_MASKS	0x0000	R/W
0x3E	AFE_FGL_A	0x0320	R/W
0x3F	AFE_FGL_B	0x0320	R/W
0x40	ADC_CTRL	0x1010	R/W
0x41	ADC_REG1	0x004C	R/W
0x42	AFE_TRIM_A	0x1C38	R/W
0x43	AFE_TEST_A	0xADA5	R/W
0x44	AFE_TRIM_B	0x1C38	R/W
0x45	AFE_TEST_B	0xADA5	R/W
0x46	OP_TIME1	0x0000	R/W
0x47	REF_CTRL	0x0080	R/W
0x48	BIAS_PD1	0x0000	R/W
0x49	TRIM1	0x0000	R/W
0x4A	BTRIM2	0x0000	R/W
0x4B	OSC32K	0x2612	R/W
0x4C	OSC32M	0x0004	R/W
0x4D	OSC32M_TRIM	0x0098	R/W
0x4E	ADC_POST	0x0060	R/W
0x4F	MISC	0x2090	R/W
0x50	TEST_MODE	0x0000	R/W
0x51	TEST_USE_I2C	0x0000	R/W
0x52	TEST_PD	0x0040	R/W
0x53	TEST_AFE_SEL_MODE	0xE400	R/W
0x54	AFE_POWER_CTL	0x0AA0	R/W
0x55	AFE_CHANNEL_TRIM	0x0000	R/W
0x56	R5_OPTIONS	0x0000	R/W
0x57	EFUSE_CTRL	0x0000	R/W
0x58	CALIBRATE	0x0000	R/W
0x59	QRESET_VCM_B	0x0808	R/W

Address	Name	Reset	Access
0x5A	TIMING_OPTIONS	0x0010	R/W
0x5B	AO_CANCEL_A	0x0000	R/W
0x5C	AO_CANCEL_PROX_A	0x0000	R/W
0x5D	AO_CANCEL_B	0x0000	R/W
0x5E	QRESET_VCM	0x0808	R/W
0x5F	ACCESS_CTRL	0x0000	R/W
0x60	DATA_BUFFER	0x0000	R
0x61	I	0x0000	R
0x62	X	0x0000	R
0x63	Y	0x0000	R
0x64	X1_a	0x0000	R
0x65	X2_a	0x0000	R
0x66	Y1_a	0x0000	R
0x67	Y2_a	0x0000	R
0x68	X1_b	0x0000	R
0x69	X2_b	0x0000	R
0x6A	Y1_b	0x0000	R
0x6B	Y2_b	0x0000	R
0x6C	READ_MINDIDT	0x0000	R
0x6D	READ_MAXDIDT	0x0000	R
0x70	READ_X1L_a	0x0000	R
0x71	READ_X2L_a	0x0000	R
0x72	READ_Y1L_a	0x0000	R
0x73	READ_Y2L_a	0x0000	R
0x74	READ_X1H_a	0x0000	R
0x75	READ_X2H_a	0x0000	R
0x76	READ_Y1H_a	0x0000	R
0x77	READ_Y2H_a	0x0000	R
0x78	READ_X1L_b	0x0000	R
0x79	READ_X2L_b	0x0000	R
0x7A	READ_Y1L_b	0x0000	R
0x7B	READ_Y2L_b	0x0000	R
0x7C	READ_X1H_b	0x0000	R
0x7D	READ_X2H_b	0x0000	R
0x7E	READ_Y1H_b	0x0000	R
0x7F	READ_Y2H_b	0x0000	R

Detailed Register map with bitfield descriptions

Addr	Name	Bits	Bit Name	Description	Reset	Access
0x00	INT_STATUS	[15:8]	fifo_status	FIFO status, number of available data (in bytes) to be read out. fifo_status[7:0] is cleared when write bit[15] to 1. Clear FIFO only when not operating.	0x0	R/W1C
		[7:0]	int_status	status of interrupt, each bit is cleared when 1 written to that bit. [0] proximity ON1 interrupt. [1] proximity OFF1 interrupt. [2] proximity ON2 interrupt. [3] proximity OFF2 interrupt [4] gesture interrupt. [5] timeslot a sample interrupt [6] timeslot b sample interrupt	0x0	R/W1C
0x01	INT_MASK	15	r_samp_int_type_b	0: trigger interrupt when sample data is written to data registers 1: trigger interrupt when data is being written into FIFO.	0x0	R/W
		14	r_samp_int_type_a	0: trigger interrupt when sample data is written into data registers 1: trigger interrupt when data is being written into FIFO.	0x0	R/W
		[13:9]	RESERVED	Reserved.	0x0	R
		[8:0]	r_int_mask	interrupt mask (high active) for [0] proximity ON1 interrupt. [1] proximity OFF1 interrupt. [2] proximity ON2 interrupt. [3] proximity OFF2 interrupt [4] gesture interrupt. [5] timeslot a sample interrupt [6] timeslot b sample interrupt [7]: reserved [8] FIFO valid data more than threshold (r_fifo_th, reg0x06 bit[11:8])	0xFF	R/W
0x02	PAD_IO_CTRL	[15:10]	RESERVED	Reserved.	0x0	R
		9	r_tcli_od	TCLI share. If output enabled with r_tcli_oe: 0: drive always 1: drive only when asserted	0x0	R/W
		8	r_tcli_pol	TCLI polarity. 0: high active 1: low active	0x0	R/W
		7	r_sda_slew	reserved	0x0	R/W
		[6:5]	r_io_slew		0x0	R/W
		[4:3]	r_io_drv		0x0	R/W
		2	r_int_oe	INT oe. 0: disable INT pin drive 1: enable drive according to pol, od, and value	0x0	R/W
		1	r_int_od	INT share. If output enabled with r_int_oe: 0: drive always 1: drive only when asserted	0x0	R/W
		0	r_int_pol	INT polarity. 0: high active 1: low active	0x0	R/W
0x03	SCAN_MODE	15	r_scan_mode		0x0	W1
		14	r_scan_comp		0x0	R/W
		13	r_scan_method		0x0	R/W
		[12:0]	r_scan_chain_length		0x0	R/W

Addr	Name	Bits	Bit Name	Description	Reset	Access
0x04	DARK_STATUS **(R6)**	[15:8]	RESERVED	Reserved.	0x0	R
		[7:4]	bigdark_b	status of dark compare timeslot B, bits and counts are cleared when register is read [0] x1 exceeded dark threshold count [1] x2 exceeded dark threshold count [2] y1 exceeded dark threshold count [3] y2 exceeded dark threshold count	0x0	R
		[3:0]	bigdark_a	status of dark compare timeslot A, bits and counts are cleared when register is read [0] x1 exceeded dark threshold count [1] x2 exceeded dark threshold count [2] y1 exceeded dark threshold count [3] y2 exceeded dark threshold count	0x0	R
0x06	I2CS_CTL_MATCH	[15:14]	RESERVED	Reserved.	0x0	R
		[13:8]	r_fifo_th	minimum FIFO words to trigger interrupt	0x0	R/W
		[7:0]	RESERVED	Reserved.	0x0	R
0x07	INTERNAL_STATE	[15:0]	internal_state	Internal states for debug. bit[0]: data flag (even/odd sample flag) bit[1]: gesture sequence valid flag. bit[2]: gesture sequence end flag. bit[3]: gesture sequence start flag. bit[4]: p_didtmax bit[5]: n_didtmax bit[7:6]: internal op_mode. bit[9:8] 0 bit[13:10]: current state of mode_ctrl block. same definitions as reg 0x10	0x0	R
0x08	CHIPID	[15:8]	r_version	R0: 0x0 R1: 0x1 R2: 0x2 R3: 0x3 R4: 0x4 R5: 0x5 R6: 0x8	0x8	R
		[7:0]	r_chip_id		0x16	R
0x09	I2CS_ID	[15:8]	RESERVED	Reserved.	0x0	R
		[7:1]	slave_address	i2c slave address	0x64	R
		0	RESERVED	Reserved.	0x0	R
0x0A	OS32M_CAL_OUT	[15:12]	RESERVED	Reserved.	0x0	R
		[11:0]	osc_cal_out	Counter value of 2cycle 32KHz pulses with 32Mhz clock.	0x0	R

Addr	Name	Bits	Bit Name	Description	Reset	Access
0x0B	GPIO_CTRL	[15:13]	RESERVED	Reserved.	0x0	R
		[12:8]	r_tcli_alt_cfg	TCLI Pin alternate function. 0x0: do R4 function 0x1: INT function 0x2: PD signal for AFE_VREF (active2,active3) 32KHz domain 0x3: ts_active_a 0x4: ts_active_b 0x5: pulse output timeslot a only 0x6: pulse output timeslot b only 0x7: pulse output both timeslots 0x8: float led pulse timeslot a 0x9: float led pulse timeslot b 0xa: float led pulse both timeslots 0xb: aoc pulse 0xc: output data cycle occurred in a 0xd: output data cycle occurred in b 0xe: output data cycle occurred 0xf: odd timeslot (toggle) 0x10: 0 0x11: 1 0x12: PD[0] (tia powerdown) 0x13: 32KHz osc output 0x14: 32MHz osc output 0x15: 1MHz (32MHz/32) 0x16: dclk 0x17 afe integrate clock 0 0x18 afe integrate clock 1 0x19: OR of r_bigdark_a 0x1a: OR of r_bigdark_b 0x1b: OR of r_bigdark_a,r_bigdark_b 0x1c: OR of r_bigdark_a, r_bigdark_b, INT	0x0	R/W
		[7:5]	RESERVED	Reserved.	0x0	R
		[4:0]	r_int_alt_cfg	INT Pin alternate function. 0x0: do R4 function 0x1: INT function 0x2: PD signal for AFE_VREF (active2,active3) 32KHz domain 0x3: ts_active_a 0x4: ts_active_b 0x5: pulse output timeslot a only 0x6: pulse output timeslot b only 0x7: pulse output both timeslots 0x8: float led pulse timeslot a 0x9: float led pulse timeslot b 0xa: float led pulse both timeslots 0xb: aoc pulse 0xc: output data cycle occurred in a 0xd: output data cycle occurred in b 0xe: output data cycle occurred 0xf: odd timeslot (toggle) 0x10: 0 0x11: 1 0x12: PD[0] (tia powerdown) 0x13: 32KHz osc output 0x14: 32MHz osc output 0x15: 1MHz (32MHz/32) 0x16: dclk 0x17 afe integrate clock 0 0x18 afe integrate clock 1 0x19: OR of r_bigdark_a 0x1a: OR of r_bigdark_b 0x1b: OR of r_bigdark_a,r_bigdark_b 0x1c: OR of r_bigdark_a, r_bigdark_b, INT	0x0	R/W

Addr	Name	Bits	Bit Name	Description	Reset	Access
0x0D	I2CS_ID_KEY	[15:0]	I2CS_ID_KEY	interlock for slave address. Write the following key values to enable changing the I2C address in i2cs_id: 0x04AD: enable address update independent of INT or TCLI (all devices change) 0x44AD: enable address update if INT input is high, ignore TCLI 0x84AD: enable address update if TCLI input is high, ignore INT 0xC4AD: enable address update if both INT and TCLI are high	0x0	R/W
0x0F	SW_RESET	[15:1]	RESERVED	Reserved.	0x0	R
		0	sw_reset	software reset (self clear)	0x0	R/W
0x10	OP_MODE	15	r_force_mode_1m_en	force 1mhz mode enable. 0: normal operation mode 1: overwrite operation state machine with r_force_mode_1m	0x0	R/W
		[14:12]	r_force_mode_1m	force 1mhz mode. Force state values: 0: idle 1: TS_A Pulse Gen 2: TS_A Processing 3: TS_B Pulse Gen 4: TS_B Processing	0x0	R/W
		[11:9]	RESERVED	Reserved.	0x0	R
		8	r_force_mode_en	force mode control enable. 0: normal operation mode 1: overwrite operation state machine with r_force_mode	0x0	R/W
		[7:4]	r_force_mode	force mode. 0: OFF 1: IDLE 2: SLEEP 3: WAKEUP0 4: WAKEUP 5: ACTIVE1 6: ACTIVE2 7: ACTIVE3 8: ACTIVE4 11: SFW0 12: SFWUP 13: SFA1 14: SFA2 15: SFA3 others: N/A	0x0	R/W
		[3:2]	RESERVED	Reserved.	0x0	R
		[1:0]	r_op_mode	global operation mode. Global operation: 0: OFF. All sections powered down 1: IDLE. AKA Program mode. No timeslot operations occur. Use this mode for changing configurations 2: GO. Allows the enabled timeslot based operations to occur	0x0	R/W

Addr	Name	Bits	Bit Name	Description	Reset	Access
0x11	OP_MODE_CFG	15	r_raw16_mode_b	use 16lsbs of 32bit register for data writes to FIFO timeslot b. 0: normal operation. Always write saturated average value to FIFO in 16 bit modes. 1: Write 16 lsb's of sum/average with no saturation to FIFO in 16 bit modes. (sum/average chosen by r_rdout_mode)	0x0	R/W
		14	r_raw16_mode_a	use 16lsbs of 32bit register for data writes to FIFO timeslot a. 0: normal operation. Always write saturated average value to FIFO in 16 bit modes. 1: Write 16 lsb's of sum/average with no saturation to FIFO in 16 bit modes. (sum/average chosen by r_rdout_mode)	0x0	R/W
		13	r_rdout_mode	readback data mode. Format for extended data registers 1: read block average value (decimation with shift) 0: read block sum value (decimation without shift) (default)	0x0	R/W
		12	r_fifo_prevent_en	prevent fifo overrun enable. 0: write data into FIFO. 1: write data into FIFO when FIFO has enough space, drop whole package if FIFO doesn't have enough space for whole package.	0x1	R/W
		11	r_prox_sample_en	sample enable for proximity mode. Output proximity value according to dataout_mode_a 0: output to FIFO on each on/off event only 1: output proximity value to FIFO for every sample time between proximity 1 on and proximity 1 off.	0x0	R/W
		10	r_prox_auto_gesture	enable auto mode switch between prox and gest. 0: normal operation 1: auto switch from prox to gesture when triggered	0x0	R/W
		9	r_samp_out_mode	smart sample mode enable. 0: output every sample. 1: output samples only between gesture start and gesture end.	0x0	R/W
		[8:6]	r_dataout_mode_b Timeslot B supports modes other than 0 only: If timeslot A is not writing the fifo, or If timeslot A is also in sample mode, has output set to sample data, has the same decimation and also is not in smart sample mode.	data format mode timeslot a. **0:** no data to fifo **1:** 2 bytes (16 bit) X1 (digital integrate mode) or 2 bytes (16 bit) sum of all channels (R4) **2:** 4 bytes x1l, x1h (digital integrate mode) or 4 bytes (32bit) sum of all channels (R4) **3:** 4 bytes (16 bit) X1,X2 (digital integrate mode) **4:** 8 bytes Sample data X1,X2,Y1,Y2 data or 8 bytes x1l,x1h,x2l,x2h (digital integrate mode) **6:** 16 bytes extended Sample data x1l,x1h,x2l,x2h,y1l,y1h,y2l,y2h	0x0	R/W
		5	r_op_mode_b	operation mode timeslot b. 0: Disabled 1: Sample mode	0x0	R/W
		[4:2]	r_dataout_mode_a	data format mode timeslot a. **0:** no data to fifo **1:** 2 bytes Proximity I data to fifo or 2 bytes X1 in (digital integrate mode) or 2 bytes (16 bit) sum of all channels (R4) **2:** 4 bytes Gesture G,I data to fifo or 4 bytes x1l, x1h, (digital integrate mode) or 4 bytes (32bit) sum of all channels (R4) **3:** 6 bytes Proximity X,Y,I data to fifo or 4 bytes (16 bit) X1,X2 (digital integrate mode) **4:** 8 bytes Sample data X1,X2,Y1,Y2 data or 8 bytes x1l,x1h,x2l,x2h (digital integrate mode) **5:** 8 bytes Gesture data G,I,Maxdidt,Mindidt **6:** 16 bytes extended Sample data x1l,x1h,x2l,x2h,y1l,y1h,y2l,y2h	0x0	R/W
		[1:0]	r_op_mode_a	operation mode timeslot a. 0: Disabled 1: Sample mode 2: Proximity mode 3: Gesture mode.	0x0	R/W

Addr	Name	Bits	Bit Name	Description	Reset	Access
0x12	GEST_FREQ	[15:0]	r_gest_freq	sample rate for gesture mode. Divider for gesture frequency rate: Formula: Sample Rate = 32KHz / (r_gest_freq *4)	0x28	R/W
0x13	PROX_FREQ	[15:0]	r_prox_freq	sample rate for proximity mode. Divider for proximity sample rate Formula: Sample rate = 32KHz / (r_prox_freq * 4)	0x320	R/W

Addr	Name	Bits	Bit Name	Description	Reset	Access
0x14	PD_SELECT	15	r_disable_swmod_b	disable precondition/modulate for timeslot b. Disable modulation/vcm precondition for timeslot b. 1: connect to PD and don't precondition 0: use vcm conditioning and allow modulation of connection to PD	0x0	R/W
		14	r_disable_swmod_a	disable precondition/modulate for timeslot a. Disable modulation/vcm precondition for timeslot a. 1: connect to PD and don't precondition 0: use vcm conditioning and allow modulation of connection to PD	0x0	R/W
		[13:12]	r_pd_vdd_disable	control Vdd connections to PD during sleep. Connect PD to CATHODE during sleep mode control: 0: Connect PD terminals 1-8 to CATHODE during sleep (default) 1: Connect PD terminals 5-8 to CATHODE during sleep, leave 1-4 connected to switch matrix 2: Connect PD terminals 1-4 to CATHODE during sleep, leave 5-8 connected to switch matrix 3: don't connect any PD terminals to CATHODE during sleep, leave connect to switch matrix	0x0	R/W
		11	r_float_anode_sleep	float pd anodes during sleep. Combined with pd_vdd_disable. Added in R5. 0: anodes not connected to cathode during sleep remain connected to input mux 1: anodes not connected to cathode during sleep are left floating by disconnecting connect switch	0x0	R/W
		[10:8]	r_sw_matrix_b **NEW CODES for R6** See rev 5 spec for Pre-R6 connections	select connection of PD for timeslot b. Code selects connections of 8 PD terminals (1-8) into 4 AFE inputs x1,x2,y1,y2) code: connections: usage 0: a=n/c b=n/c c=n/c d=n/c (no connect) 1: x1=3+4 x2=1+2 y1=n/c y2=n/c 2: x1=7+8 x2=5+6 y1=n/c y2=n/c 3: x1=1+2+3+4 x2=n/c y1=n/c y2=n/c 4: x1=5 x2=6 y1=7 y2=8 5: x1=1 x2=2 y1=3 y2=4 6: x1=3+4 x2=5+6 c=n/c d=n/c 7: x1=5+6+7+8 x2=n/c y1=n/c y2=n/c	0x5	R/W
		7	r_random_sleep	Enable random sleep LSFR. Set this bit to 1 to turn off the bandpass filter in channel 1 during all digital integrate cycles	0x0	R/W
		[6:4]	r_sw_matrix_a **NEW CODES for R6** See rev 5 spec for Pre-R6 connections	select connection of PD for timeslot a. Code selects connections of 8 PD terminals (1-8) into 4 AFE inputs x1,x2,y1,y2) code: connections: usage 0: a=n/c b=n/c c=n/c d=n/c (no connect) 1: x1=3+4 x2=1+2 y1=n/c y2=n/c 2: x1=7+8 x2=5+6 y1=n/c y2=n/c 3: x1=1+2+3+4 x2=n/c y1=n/c y2=n/c 4: x1=5 x2=6 y1=7 y2=8 5: x1=1 x2=2 y1=3 y2=4 6: x1=3+4 x2=5+6 c=n/c d=n/c 7: x1=5+6+7+8 x2=n/c y1=n/c y2=n/c	0x4	R/W
		[3:2]	r_modulate_b	select modulation for timeslot b. Timeslot B modulation select 0: pulse PD connection to AFE 1: pulse LED1 2: pulse LED2 3: pulse LED3 (R3) (pulse both in R0-R2)	0x0	R/W
		[1:0]	r_modulate_a	select modulation for timeslot a. Timeslot A modulation select 0: pulse PD connection to AFE 1: pulse LED1 2: pulse LED2 3: pulse LED3 (R3) (both leds in r0-r2)	0x1	R/W

Addr	Name	Bits	Bit Name	Description	Reset	Access
0x15	DEC_MODE	[15:11]	RESERVED	Reserved.	0x0	R
		[10:8]	r_dec_mode_b	decimation rate for timeslot b. 0: 1 1: 2 2: 4 3: 8 4: 16 5: 32 6: 64 7: 128	0x6	R/W
		7	RESERVED	Reserved.	0x0	R
		[6:4]	r_gest_dec_mode_a	decimation rate for gesture mode for timeslot a. 0: 1 1: 2 2: 4 3: 8 4: 16 5: 32 6: 64 7: 128	0x0	R/W
		3	RESERVED	Reserved.	0x0	R
		[2:0]	r_prox_dec_mode_a	decimation rate for proximity mode for timeslot a. 0: 1 1: 2 2: 4 3: 8 4: 16 5: 32 6: 64 7: 128	0x0	R/W
0x16	DARK_TH_A **(R6)**	[15:14]	r_dark_cnt_a	Dark count for Timeslot A. Number of values greater than dark_th required to set bigdark bit 0: never set bit 1: 1 sample 2: 4 samples 3: 16 samples	0x0	R/W
		[13:0]	r_dark_th_a	Dark threshold for Timeslot A. ADC value is compared to this value during "subtract" regions and increments the count if ADC is greater than this value.	0x3000	R/W
0x17	INTEG_ORDER_A **(R6)**	[15:4]	RESERVED	Reserved.	0x0	R
		[3:0]	r_reverse_int_a	Integration clock order A. Bit to control integration clock order in groups of four pulses. Bit 0 controls the first pulse's clock and each subsequent bit matches the corresponding next pulses. After four pulses the sequence repeats. 0: normal integration clock 1: reversed integration clock	0x0	R/W
0x18	CH1_OFFSET_A	[15:14]	RESERVED	Reserved.	0x0	R
		[13:0]	r_ch1_offset_a	default: 0x2000, the offset of ADC output.	0x2000	R/W
0x19	CH2_OFFSET_A	[15:14]	RESERVED	Reserved.	0x0	R
		[13:0]	r_ch2_offset_a	default: 0x2000, the offset of ADC output.	0x2000	R/W

Addr	Name	Bits	Bit Name	Description	Reset	Access
0x1A	CH3_OFFSET_A	[15:14]	RESERVED	Reserved.	0x0	R
		[13:0]	r_ch3_offset_a	default: 0x2000, the offset of ADC output.	0x2000	R/W
0x1B	CH4_OFFSET_A	[15:14]	RESERVED	Reserved.	0x0	R
		[13:0]	r_ch4_offset_a	default: 0x2000, the offset of ADC output.	0x2000	R/W
0x1C	DARK_TH_B **(R6)**	[15:14]	r_dark_cnt_b	Dark count for Timeslot B. Number of values greater than dark_th required to set bigdark bit 0: never set bit 1: 1 sample 2: 4 samples 3: 16 samples	0x0	R/W
		[13:0]	r_dark_th_b	Dark threshold for timeslot B. ADC value is compared to this value during "subtract" regions and increments the count if ADC is greater than this value.	0x3000	R/W
0x1D	INTEG_ORDER_B **(R6)**	[15:4]	RESERVED	Reserved.	0x0	R
		[3:0]	r_reverse_int_b	Integration clock order B. Bit to control integration clock order in groups of four pulses. Bit 0 controls the first pulse's clock and each subsequent bit matches the corresponding next pulses. After four pulses the sequence repeats. 0: normal integration clock 1: reversed integration clock	0x0	R/W
0x1E	CH1_OFFSET_B	[15:14]	RESERVED	Reserved.	0x0	R
		[13:0]	r_ch1_offset_b	default: 0x2000, the offset of ADC output.	0x2000	R/W
0x1F	CH2_OFFSET_B	[15:14]	RESERVED	Reserved.	0x0	R
		[13:0]	r_ch2_offset_b	default: 0x2000, the offset of ADC output.	0x2000	R/W
0x20	CH3_OFFSET_B	[15:14]	RESERVED	Reserved.	0x0	R
		[13:0]	r_ch3_offset_b	default: 0x2000, the offset of ADC output.	0x2000	R/W
0x21	CH4_OFFSET_B	[15:14]	RESERVED	Reserved.	0x0	R
		[13:0]	r_ch4_offset_b	default: 0x2000, the offset of ADC output.	0x2000	R/W

Addr	Name	Bits	Bit Name	Description	Reset	Access
0x22	LED3_DRV	[15:14]	RESERVED	Reserved.	0x0	R
		13	r_led3_en250	LED3 current scale factor. 1: 100% strength 0: 10% strength **(R6)** 0: 40% strength **(R0-R5)**	0x1	R/W
		12	r_led3_piref_en	LED3 driver current for proximity enable. 0: use R_LED3_IREF[3:0] for proximity mode. 1: use R_LED3_PIREF[3:0] for proximity mode. (default)	0x1	R/W
		[11:8]	r_led3_piref	LED3 driver current for proximity mode	0x0	R/W
		7	RESERVED	Reserved.	0x0	R
		[6:4]	r_led3_itail	LED3 slew rate control	0x0	R/W
		[3:0]	r_led3_giref	LED3 driver current. 0: 25, 1: 40, 2: 55, 3: 70, 4: 85, 5: 100, 6: 115, 7: 130, 8: 145, 9: 160, 10: 175, 11: 190, 12: 205, 13: 220, 14: 235, 15: 250mA	0x0	R/W

Addr	Name	Bits	Bit Name	Description	Reset	Access
0x23	LED1_DRV	[15:14]	RESERVED	Reserved.	0x0	R
		13	r_led1_en250	LED1 current scale factor. 1: 100% strength 0: 10% strength**(R6)** 0: 40% strength **(R0-R5)**	0x1	R/W
		12	r_led1_piref_en	LED1 driver current for proximity enable. 0: use R_LED1_IREF[3:0] for proximity mode. 1: use R_LED1_PIREF[3:0] for proximity mode. (default)	0x1	R/W
		[11:8]	r_led1_piref	LED1 driver current for proximity mode	0x0	R/W
		7	RESERVED	Reserved.	0x0	R
		[6:4]	r_led1_itail	LED1 slew rate control	0x0	R/W
		[3:0]	r_led1_giref	LED1 driver current. 0: 25, 1: 40, 2: 55, 3: 70, 4: 85, 5: 100, 6: 115, 7: 130, 8: 145, 9: 160, 10: 175, 11: 190, 12: 205, 13: 220, 14: 235, 15: 250mA	0x0	R/W
0x24	LED2_DRV	[15:14]	RESERVED	Reserved.	0x0	R
		13	r_led2_en250	LED2 current scale factor. 1: 100% drive strength 0: 10% drive strength**(R6)** 0: 40% strength **(R0-R5)**	0x1	R/W
		12	r_led2_piref_en	LED2 driver current for proximity enable. 0: use R_LED2_IREF[3:0] for proximity mode. 1: use R_LED2_PIREF[3:0] for proximity mode. (default)	0x1	R/W
		[11:8]	r_led2_piref	LED2 driver current for proximity mode	0x0	R/W
		7	RESERVED	Reserved.	0x0	R
		[6:4]	r_led2_itail	LED2 slew rate control	0x0	R/W
		[3:0]	r_led2_giref	LED2 driver current. 0: 25, 1: 40, 2: 55, 3: 70, 4: 85, 5: 100, 6: 115, 7: 130, 8: 145, 9: 160, 10: 175, 11: 190, 12: 205, 13: 220, 14: 235, 15: 250mA	0x0	R/W

Addr	Name	Bits	Bit Name	Description	Reset	Access
0x25	LED_TRIM	[15:11]	r_led3_trim	Trim the amount of current being multiplied for LED driver	0xC	R/W
		[10:6]	r_led2_trim	Trim the amount of current being multiplied for LED driver	0xC	R/W
		5	RESERVED	Reserved.	0x0	R
		[4:0]	r_led1_trim	Trim the amount of current being multiplied for LED driver	0xC	R/W
0x26	GEST_SLOPE_TH	[15:0]	r_slope_th	slope threshold, used in X/Y direction detection. If abs (dX) + abs(dY) < r_slope_th, gesture is Z direction movement.	0x0	R/W
0x27	GEST_CTRL	[15:13]	RESERVED	Reserved.	0x0	R
		12	r_iout_mode	I output mode. 0: output lower 16bit internal I 1: output higher 16bits of internal I	0x0	R/W
		[11:8]	r_gest_didt_m	m used for I(n) - I(n-m)	0x7	R/W
		[7:5]	r_orientation	orientation control. bit[0] 1: flip X0 and Y0, X1 and Y1 after X/Y flip controlled by bit[2:1] bit[1] 1: flip X direction. bit[2] 1: flip Y direction.	0x0	R/W
		[4:0]	r_gest_mask	Gesture mask. Set to 1 ignore this gesture type. Set to 0 to enable bit[0]: x negative (left) bit[1]: x positive (right) bit[2]: y negative (down) bit[3]: y positive (up) bit[4]: click	0x0	R/W
0x28	GEST_THRESH	[15:0]	r_gest_di_th	dI/dt threshold	0x0	R/W
0x29	GEST_SIZE	[15:8]	r_max_gest_length	maximum length of valid gesture	0x40	R/W
		[7:0]	r_gest_n_pts	minimum length of valid gesture	0x0	R/W
0x2A	PROX_ON_TH1	[15:0]	r_prox_th_on1	bit[15:0] of proximity ON1 threshold	0x0	R/W
0x2B	PROX_OFF_TH1	[15:0]	r_prox_th_off1	bit[15:0] of proximity OFF1 threshold	0x0	R/W
0x2C	PROX_ON_TH2	[15:0]	r_prox_th_on2	bit[15:0] of proximity ON2 threshold	0x0	R/W
0x2D	PROX_OFF_TH2	[15:0]	r_prox_th_off2	bit[15:0] of proximity OFF2 threshold	0x0	R/W
0x2E	PROX_TH1_HBYTE	[15:14]	RESERVED	Reserved.	0x0	R
		[13:8]	r_prox_th_off1_high	bit[21:16] of proximity OFF1 threshold	0x0	R/W
		[7:6]	RESERVED	Reserved.	0x0	R
		[5:0]	r_prox_th_on1_high	bit[21:16] of proximity ON1 threshold	0x0	R/W
0x2F	PROX_TH2_HBYTE	15	r_prox_type	proximity trigger type. 0: trigger when I crosses threshold. 1: trigger when I above or below threshold	0x0	R/W
		14	RESERVED	Reserved.	0x0	R
		[13:8]	r_prox_th_off2_high	bit[21:16] of proximity OFF2 threshold	0x0	R/W
		[7:6]	RESERVED	Reserved.	0x0	R
		[5:0]	r_prox_th_on2_high	bit[21:16] of proximity ON2 threshold	0x0	R/W
0x30	PULSE_OFFSET_A	[15:13]	RESERVED	Reserved.	0x0	R
		[12:8]	r_pulse_width_a	pulse width for gesture mode (in 1us step) for timeslot a	0x3	R/W
		[7:0]	r_pulse_offset_a	pulse offset for gesture mode (in 1us step) timeslot a	0x20	R/W
0x31	PULSE_PERIOD_A	[15:8]	r_pulse_number_a	pulse count for gesture mode	0x8	R/W
		[7:0]	r_pulse_period_a	pulse period for gesture mode (in 1us step)	0x18	R/W

Addr	Name	Bits	Bit Name	Description	Reset	Access
0x32	PULSE_OFFSET_PROX_A	[15:13]	RESERVED	Reserved.	0x0	R
		[12:8]	r_pulse_width_prox_a	pulse width for timeslot a proximity mode (in 1us step)	0x3	R/W
		[7:0]	r_pulse_offset_prox_a	pulse offset for timeslot a proximity mode(in 1us step)	0x20	R/W
0x33	PULSE_PERIOD_PROX_A	[15:8]	r_pulse_num_prox_a	pulse count for proximity mode	0x8	R/W
		[7:0]	r_pulse_period_prox_a	pulse period for proximity mode (in 1us step)	0x18	R/W
0x34	PULSE_MASK	[15:11]	RESERVED	Reserved.	0x0	R
		10	r_force_opcycle_en	Test mode enable signal to force the pulse generation timeslot a. 0: Normal Mode 1: Test mode	0x0	R/W
		9	r_pulse_mask_b	pulse masking signal timeslot b. 0: Normal Mode 1: Test mode	0x0	R/W
		8	r_pulse_mask_a	pulse masking signal timeslot a. 0: Disable 1: Enable	0x0	R/W
		[7:0]	r_force_opcycle_num	Operation cycle counts for which pulses are forced ON in test mode timeslot a	0x0	R/W
0x35	PULSE_OFFSET_B	[15:13]	RESERVED	Reserved.	0x0	R
		[12:8]	r_pulse_width_b	pulse width (in 1us step) for timeslot b	0x3	R/W
		[7:0]	r_pulse_offset_b	pulse offset (in 1us step) timeslot b	0x20	R/W
0x36	PULSE_PERIOD_B	[15:8]	r_pulse_number_b	pulse count	0x8	R/W
		[7:0]	r_pulse_period_b	pulse period for (in 1us step)	0x18	R/W
0x37	PULSE_PERIOD_EXT	[15:13]	r_afe_amps34_disable	Disable control for channels 3 and 4 only. 0: channel 3,4 TIA opamp 1: channel 3,4 BPF opamp 2: channel 3,4 INT opamp	0x0	R/W
		[12:10]	r_afe_amps2_disable	Disable control for channel 2 only. 0: channel 2 TIA opamp 1: channel 2 BPF opamp 2: channel 2 INT opamp	0x0	R/W
		[9:8]	r_pulse_period_ext_b	pulse period upper bits timeslot b	0x0	R/W
		[7:6]	RESERVED	Reserved.	0x0	R
		[5:4]	r_pulse_period_ext_prox_a	pulse period upper bits proximity mode timeslot a	0x0	R/W
		[3:2]	RESERVED	Reserved.	0x0	R
		[1:0]	r_pulse_period_ext_a	pulse period upper bits timeslot a	0x0	R/W

Addr	Name	Bits	Bit Name	Description	Reset	Access
0x38	SLEEP_FLOAT	15	r_bfirst	Do Timeslot B before A. Set this bit to 1 to enable timeslot b to happen before timeslot a. This must be set to allow float during sleep for timeslot b.	0x0	R/W
		14	r_go_sleep	Sleep before first sample time. Set this bit to force a sleep (and possible float during sleep) before first sample after setting go. 0: go causes immediate sample (default) 1: go causes sleep first then sample	0x0	R/W
		13	r_sleep_float_b_en	Enable timeslot B float during sleep. This bit enables the sleep during float mode for timeslot b. This causes a precondition cycle to occur during sleep for timeslot b and enables floating the pd input after the precondition. Either the r_bfirst bit must be set, OR timeslot a must be disabled for this mode to work properly.	0x0	R/W
		12	r_sleep_float_a_en	Enable timeslot A float during sleep. This bit enables the sleep during float mode for timeslot a. This causes a precondition cycle to occur during sleep for timeslot a and enables floating the pd input after the precondition. The r_bfirst bit must not be set, OR timeslot b must disabled for this mode to work properly.	0x0	R/W
		[11:0]	r_sleep_float_time	Duration of float during sleep. Formula: Sleep Float time = (125us * r_sleep_float_time) + pulse_offset + (9 to 10) us Values of 0 or 1 are not allowed.	0x0	R/W
0x39	AFE_CTRL_A	[15:11]	r_afe_width_a	AFE clock width for gesture mode (in 1us step) timeslot a	0x4	R/W
		[10:5]	r_afe_offset_a	AFE clock coarse offset for gesture mode (in 1us step) timeslot a	0x17	R/W
		[4:0]	r_afe_fine_offset_a	AFE clock fine offset for gesture mode (in 31.25ns step) , also space between digital integrate regions in 1 us steps (r4) for timeslot a	0x1C	R/W
0x3A	AFE_CTRL_PROX_A	[15:11]	r_afe_width_prox_a	AFE clock width for timeslot a proximity mode (in 1us step)	0x4	R/W
		[10:5]	r_afe_offset_prox_a	AFE clock coarse offset for timeslot a proximity mode (in 1us step)	0x17	R/W
		[4:0]	r_afe_fine_offset_prox_a	AFE clock fine offset for proximity mode (in 31.25ns step) , also space between digital integrate regions in 1 us steps (r4) timeslot a	0x1C	R/W
0x3B	AFE_CTRL_B	[15:11]	r_afe_width_b	AFE clock width for gesture mode (in 1us step) timeslot b	0x4	R/W
		[10:5]	r_afe_offset_b	AFE clock coarse offset for gesture mode (in 1us step) timeslot b	0x17	R/W
		[4:0]	r_afe_fine_offset_b	AFE clock fine offset for gesture mode (in 31.25ns step), also space between digital integrate regions in 1 us steps (r4) for timeslot b	0x1C	R/W

Addr	Name	Bits	Bit Name	Description	Reset	Access
0x3C	AFE_CTRL2	15	RESERVED	Reserved.	0x0	R
		14	r_cmclk_type	CM Clocking method. Common mode clock method control. 0: Run CMCLK only during VCM periods (default) 1: Run CMCLK when AFE amps are powered up (like 1020 and r0/r1 1022)	0x0	R/W
		[13:11]	r_cmclk_sel	Common mode clock frequency select bits. 7: Ground 6: 32MHz 5: 16MHz 4: 8MHz 3: 4MHz 2: 2MHz 1: 1MHz 0: Ground	0x6	R/W
		10	RESERVED	Reserved	0x0	R/W
		9	r_cathode_sel	PD cathode voltage select. 0: PD cathode connected to AFE reference voltage 1: PD cathode connected to VDD	0x0	R/W
		[8:3]	r_afe_pd_mask	Mask bits for afe power. 0: channel 1 TIA opamp 1: channel 1 BPF opamp 2: channel 1 INT opamp 3: channel 2,3,4 TIA opamp 4: channel 2,3,4 BPF opamp 5: channel 2,3,4 INT opamp	0x0	R/W
		2	r_afe_multi_sample_b	Multi-sample mode enable bit timeslot b. 0: Single sample mode (analog integrate) 1: Multiple sample mode (digital accumulate)	0x1	R/W
		1	r_afe_multi_sample_a	Multi-sample mode enable bit timeslot a. 0: Single sample mode (analog integrate) 1: Multiple sample mode (digital accumulate)	0x1	R/W
		0	r_afe_mask	Mask bit for AFE clock. 0: Disable masking 1: Enable masking	0x0	R/W
0x3D	AFE_MASKS	[15:8]	r_afe_icmask_b	Integration clock mask bits timeslot b. 0: Disable masking 1: Enable masking	0x0	R/W
		[7:0]	r_afe_icmask_a	Integration clock mask bits timeslot a. 0: Disable masking 1: Enable masking	0x0	R/W
0x3E	AFE_FGL_A	[15:14]	r_fgl_ledsel_a	led select for float gesture mode timeslot a. 00: No Led (default) 01: LED1 selected 10: LED2 selected 11: LED3 selected	0x0	R/W
		13	RESERVED	Reserved.	0x0	R
		[12:8]	r_fgl_pulse_width_a	pulse width for float gesture mode (in 1us step) for timeslot a	0x3	R/W
		[7:0]	r_fgl_pulse_offset_a	pulse offset for float gesture mode (in 1us step) timeslot a	0x20	R/W

Addr	Name	Bits	Bit Name	Description	Reset	Access
0x3F	AFE_FGL_B	[15:14]	r_fgl_ledsel_b	led select for float gesture mode timeslot b. 00: No Led (default) 01: LED1 selected 10: LED2 selected 11: LED3 selected	0x0	R/W
		13	RESERVED	Reserved.	0x0	R
		[12:8]	r_fgl_pulse_width_b	pulse width for float gesture mode (in 1us step) for timeslot b	0x3	R/W
		[7:0]	r_fgl_pulse_offset_b	pulse offset for float gesture mode (in 1us step) timeslot b	0x20	R/W
0x40	ADC_CTRL	[15:8]	r_adc_short	thermometric code to control how long the comparator is shorted.	0x10	R/W
		[7:0]	r_adc_gain	thermometric code to control how long the comparator is amplifying	0x10	R/W
0x41	ADC_REG1	[15:8]	r_pd_test	photodiode test mux	0x0	R/W
		7	r_adc_c_del		0x0	R/W
		6	r_adc_b_del		0x1	R/W
		5	r_adc_a_del		0x0	R/W
		4	r_adc_delay	test-mode related to band-limiting	0x0	R/W
		[3:2]	r_adc_band	band-limit the ADC to reduce noise	0x3	R/W
		[1:0]	r_adc_range	control input range of ADC	0x0	R/W

Addr	Name	Bits	Bit Name	Description	Reset	Access
0x42	AFE_TRIM_A	[15:13]	RESERVED	Reserved.	0x0	R
		[12:8]	r_afe_trim_int_a	integrator register and capacitor select. Upper three bits select capacitors: Cfb = 3.62pF x trim_int[4] + 1.81pF x trim_int[3] + 0.9pF x trim_int[2] Lower two bits select resistors: Rin = 200KΩ (00), 100KΩ (01), 50KΩ (10, 11) (R0-R4) **Rin = 400KΩ (00), 200KΩ (01), 100KΩ (10, 11) (R5+)** Digital Integrate Gain: Gain = n/aB (00), 1 (01), 2 (10, 11) (R0-R4) **Gain = 1 (00), 1 (01), .6875 (10, 11) (R5+)**	0x1C	R/W
		7	r_afe_int_as_buffer_a	convert integrator to buffer. Useful for tia-adc mode. 0: normal integrator configuration (default) 1: convert integrator to buffer amplifier DIgital integration mode automatically does this conversion	0x0	R/W
		6	r_afe_trim_tia_pc_en_a	TIA feedback register per-channel enable. When this bit is set, channel 1 during timelot a gets its value from this value and channels 2-4 get their values from the fields in register TIA_CHANNEL_TRIM (0x55)	0x0	R/W
		[5:4]	r_afe_trim_vref_a	voltage trim for 1.6V ref buffer. afe vref settings (set by 10uA current into resistors): 0: 1.1385V (R2) 1: 1.012V (R2) 2: 0.8855V (R2) 3: 1.265V (R2) Older versions: 13/16 vdd (R0/R1) 12/16 vdd (R0/R1) 15/16 vdd (R0) 8/16 vdd (R1) 14/16 vdd (R0/R1)	0x3	R/W
		[3:2]	r_afe_trim_bpf_a	BPF input and feedback network register select. 0: 100KHz/100KHz (Zin = 350KΩ // 4.52pF, Zfb = 353KΩ + 4.52pF) 1: 390KHz/100KHz (Zin = 90KΩ // 4.52pF, Zfb = 353KΩ + 4.52pF) 2: 407KHz/100KHz (Zin = 350KΩ // 4.52pF, Zfb = 87KΩ + 4.52pF) 3: 407KHz/390KHz (Zin = 90KΩ // 4.52pF, Zfb = 87KΩ + 4.52pF)	0x2	R/W
		[1:0]	r_afe_trim_tia_a	TIA feedback resister select (50K ~ 200K) 0:200KΩ 1:100KΩ 2:50KΩ 3:25KΩ If per-channel enable is set, this controls channel 1, otherwise it all 4 channels.	0x0	R/W

Addr	Name	Bits	Bit Name	Description	Reset	Access
0x43	AFE_TEST_A	[15:0]	r_afe_mux_test_a	bypass and input mux select. b0-b4: AFE input MUX switch control b5-b7: bypass MUX switch control between TIA, BPF. b8-b10: bypass MUX switch control between BPF, INT b11-b12: bypass MUX switch control between INT, ADC b13-b14: clock source select for AFE charge pump bit[14:13] 00: use 32KHz oscillator is used to clock AFE charge pump. 01: use 1MHz oscillator is used to clock AFE charge pump (default) 1X: disable clock input. b15: AFE charge pump enable 0: enable charge pump 1: disable charge pump. Set this to AE65 for digital integrate or integrator as buffer modes	0xADA5	R/W

Addr	Name	Bits	Bit Name	Description	Reset	Access
0x44	AFE_TRIM_B	[15:13]	RESERVED	Reserved.	0x0	R
		[12:8]	r_afe_trim_int_b	integrator register and capacitor select. Upper three bits select capacitors: Cfb = 3.62pF x trim_int[4] + 1.81pF x trim_int[3] + 0.9pF x trim_int[2] Lower two bits select resistors: Rin = 200KΩ (00), 100KΩ (01), 50KΩ (10, 11) (R0-R4) **Rin = 400KΩ (00), 200KΩ (01), 100KΩ (10, 11) (R5+)** Digital Integrate Gain: Gain = n/aB (00), 1 (01), 2 (10, 11) (R0-R4) **Gain = 1 (00), 1 (01), .6875 (10, 11) (R5+)**	0x1C	R/W
		7	r_afe_int_as_buffer_b	convert integrator to buffer. Useful for tia-adc mode. 0: normal integrator configuration (default) 1: convert integrator to buffer amplifier DIgital integration mode automatically does this conversion	0x0	R/W
		6	r_afe_trim_tia_pc_en_b	TIA feedback register per-channel enable b. When this bit is set, channel 1 during timelot b gets its value from this value and channels 2-4 get their values from the fields in register TIA_CHANNEL_TRIM (0x55)	0x0	R/W
		[5:4]	r_afe_trim_vref_b	voltage trim for 1.6V ref buffer. afe vref settings (set by 10uA current into resistors): 0: 1.1385V (R2) 1: 1.012V (R2) 2: 0.8855V (R2) 3: 1.265V (R2) Older versions: 13/16 vdd (R0/R1) 12/16 vdd (R0/R1) 15/16 vdd (R0) 8/16 vdd (R1) 14/16 vdd (R0/R1)	0x3	R/W
		[3:2]	r_afe_trim_bpf_b	BPF input and feedback network register select. 0: 100KHz/100KHz (Zin = 350KΩ // 4.52pF, Zfb = 353KΩ + 4.52pF) 1: 390KHz/100KHz (Zin = 90KΩ // 4.52pF, Zfb = 353KΩ + 4.52pF) 2: 407KHz/100KHz (Zin = 350KΩ // 4.52pF, Zfb = 87KΩ + 4.52pF) 3: 407KHz/390KHz (Zin = 90KΩ // 4.52pF, Zfb = 87KΩ + 4.52pF)	0x2	R/W
		[1:0]	r_afe_trim_tia_b	TIA feedback resister select (50K ~ 200K) 0:200KΩ 1:100KΩ 2:50KΩ 3:25KΩ If per-channel enable is set, this controls channel 1, otherwise it all 4 channels.	0x0	R/W

Addr	Name	Bits	Bit Name	Description	Reset	Access
0x45	AFE_TEST_B	[15:0]	r_afe_mux_test_b	bypass and input mux select. b0-b4: AFE input MUX switch control b5-b7: bypass MUX switch control between TIA, BPF. b8-b10: bypass MUX switch control between BPF, INT b11-b12: bypass MUX switch control between INT, ADC b13-b14: clock source select for AFE charge pump bit[14:13] 00: use 32KHz oscillator is used to clock AFE charge pump. 01: use 1MHz oscillator is used to clock AFE charge pump (default) 1X: disable clock input. . b15: AFE charge pump enable 0: enable charge pump 1: disable charge pump. Set this to AE65 for digital integrate or integrator as buffer modes	0xADA5	R/W
0x46	OP_TIME1	[15:12]	r_afe_time	afe operation time in 32kHz cycles	0x0	R/W
		[11:8]	r_os32m_time	Fast OSC startup time	0x0	R/W
		[7:0]	r_ref_time	REF startup time	0x0	R/W
0x47	REF_CTRL	[15:12]	RESERVED	Reserved.	0x0	R
		[11:10]	r_ref_tm	mux test points	0x0	R/W
		9	r_ref_limit	Control charging time. 0: fast charge. 1: slow charge	0x0	R/W
		8	RESERVED	Reserved.	0x0	R
		[7:0]	r_ref_abs	trim REF output value. default: 0x80 makes 1.2v. 2mV/LSB	0x80	R/W
0x48	BIAS_PD1	15	RESERVED	Reserved.	0x0	R
		[14:12]	r_bias_trim50u	Trim 50uA output. 0: 50uA 1: 45.8uA 2: 41.5uA 3: 37.5uA 4: 66.5uA 5: 62.5uA 6: 58.35uA 7: 54.15uA	0x0	R/W
		11	RESERVED	Reserved.	0x0	R
		10	r_bias_test	output of 10uA test current	0x0	R/W
		9	r_bias_i10u5_pd	power down of 10uA output 5	0x0	R/W
		8	r_bias_i50u_pd	power down of 50uA output	0x0	R/W
		7	r_bias_i10u4_pd	power down of 10uA output4	0x0	R/W
		6	r_bias_i10u3_pd	power down of 10uA outupt3	0x0	R/W
		5	r_bias_i10u2_pd	power down of 10uA otuput2	0x0	R/W
		4	r_bias_i10u1_pd	power down of 10uA output1	0x0	R/W
		3	r_bias_i5u4_pd	power down of 5uA output4	0x0	R/W
		2	r_bias_i5u3_pd	power down of 5uA output3	0x0	R/W
		1	r_bias_i5u2_pd	power down of 5uA output2	0x0	R/W
		0	r_bias_i5u1_pd	power down of 5uA outpu1	0x0	R/W

Addr	Name	Bits	Bit Name	Description	Reset	Access
0x49	TRIM1	[15:9]	RESERVED	Reserved.	0x0	R
		[8:7]	r_bias_trim10u5	Bias trim 10uA for AFE vref	0x0	R/W
		[6:4]	r_bias_atrim	Trim all bias current outputs (50uA, 10uA and 5uA) 0: default. 1: 9.1% increase 2: 20% increase 3: 33.3% increase 4: 38.3% decrease 5: 29.4% decrease 6: 20% decrease 7: 11.1% decrease	0x0	R/W
		[3:0]	r_bg_trim	Trim bandgap current output. 0: 50uA 1: 51.57uA 2: 53.14uA 3: 54.71uA 4: 56.28uA 5: 57.85uA 6: 59.42uA 7; 60.99uA 8: 37.44uA 9: 39.01uA 10: 40.58uA 11: 42.15uA 12: 43.72uA 13: 45.29uA 14: 46.86uA 15: 48.43uA	0x0	R/W

Addr	Name	Bits	Bit Name	Description	Reset	Access
0x4A	BTRIM2	[15:14]	r_bias_trim10u4	trim bias<1> for LED driver. 0: 10uA 1: 8uA 2: 14uA 3: 12uA	0x0	R/W
		[13:12]	r_bias_trim10u3	trim bias<0> for LED driver. 0: 10uA 1: 8uA 2: 14uA 3: 12uA	0x0	R/W
		[11:10]	r_bias_trim10u2	trim bias for OS32M. 0: 10uA 1: 8uA 2: 14uA 3: 12uA	0x0	R/W
		[9:8]	r_bias_trim10u1	trim bias for OS32M reference. 0: 10uA 1: 8uA 2: 14uA 3: 12uA	0x0	R/W
		[7:6]	r_bias_trim5u4	trim bias<1> for REF. 0: 5uA 1: 4uA 2: 7uA 3: 6uA	0x0	R/W
		[5:4]	r_bias_trim5u3	trim bias<0> for REF. 0: 5uA 1: 4uA 2: 7uA 3: 6uA	0x0	R/W
		[3:2]	r_bias_trim5u2	trim bias<1> for ADC. 0: 5uA 1: 4uA 2: 7uA 3: 6uA	0x0	R/W
		[1:0]	r_bias_trim5u1	trim bias<0> for ADC. 0: 5uA 1: 4uA 2: 7uA 3: 6uA	0x0	R/W

Addr	Name	Bits	Bit Name	Description	Reset	Access
0x4B	OSC32K	15	RESERVED	Reserved.	0x0	R
		14	r_os32k_test4	32KHz oscillator frequency range. 0: normal. 1: increased frequency range.	0x0	R/W
		[13:12]	r_os32k_test3	capacitor current trim. 0: 1.0x current 1: 1.25x current 2: 1.5x current 3: 1.75x current	0x2	R/W
		11	r_os32k_test2	comparator current trim. 0: normal operation 1: 2x current	0x0	R/W
		[10:9]	r_os32k_test1	Vref current trim. 0: 1x current 1: 2x current 2: 3x current 3: 4x current	0x3	R/W
		8	r_os32k_bypass	bypass 32KHz oscillator. 0: normal operation 1: external clock (TCLI)	0x0	R/W
		7	r_os32k_pdb	32KHz oscillator power down (low active) 0: power down 1: normal operation	0x0	R/W
		6	RESERVED	Reserved.	0x0	R
		[5:0]	r_os32k_trim	32KHz oscillator trim. 00 0000: max frequency 01 0010 default frequency 11 1111: min frequency	0x12	R/W
0x4C	OSC32M	[15:12]	r_os32m_ctat_trim	32MHz oscillator CTAT trim. 32MHz oscillator CTAT trim. New in R5	0x0	R/W
		[11:8]	r_os32m_ptat_trim	32MHz oscillator PTAT trim. Ptat trim for 32MHz oscillator. New in R5	0x0	R/W
		[7:5]	RESERVED	Reserved.	0x0	R
		4	r_os32m_start	start select. 0: normal start-up (R0-R4) 1: old start-up (R0-R4) This bit is ignored in R5	0x0	R/W
		3	r_os32m_fast	fast lock detect. 0: normal operation (R0-R4) 1: fast lock detect (R0-R4) **This bit is ignored starting in R5**	0x0	R/W
		2	r_os32m_enb_16m	enable 16MHz(0.5x) oscillator (low active) 0: 16MHz (R0-R4) 1: 32MHz (R0-R4) **This bit is ignored starting in R5**	0x1	R/W
		1	r_os32m_bypass	bypass 32MHz oscillator. 0: normal operation 1: external clock (TCLI)	0x0	R/W
		0	RESERVED	Reserved.	0x0	R
0x4D	OSC32M_TRIM	[15:8]	RESERVED	Reserved.	0x0	R
		[7:0]	r_os32m_freq_adj	32MHz oscillator frequency adjust. 0x00: min frequency 0x98: default frequency (32MHz) 0xFF: max frequency	0x98	R/W

Addr	Name	Bits	Bit Name	Description	Reset	Access
0x4E	ADC_POST	15	RESERVED	Reserved.	0x0	R
		14	r_adc_dummy_cycle	Insert extra dummy cycle before first channel conversion. 0: no extra cycle 1: add 1 operation with no select	0x1	RW
		[13:12]	r_adc_input_short	ADC input short options. Optionally apply a shorting pulse at the end of each adc cycle. (except in digital integrate mode) 0: no pulse 1: 1 32MHz cycle 2: 2 32MHz cycles 3: 3 32MHz cycles	0x3	R/W
		[11:10]	r_adc_testmode	ADC testmode. 0: normal operation 1: use conv_start as sample enable signal.	0x0	R/W
		[9:8]	r_adc_pipe	adjust sample clock timing	0x0	R/W
		7	r_adc_force_clken_tm	forces test mode ADC clock to always stay on	0x0	R/W
		6	r_adc_clksel_tm	ADC clock select. selects normal or test mode ADC clock: 0: test mode, adc clock runs all during ACTIVE3 mode 1: normal mode, adc clock only pulses during conversion cycles (default)	0x1	R/W
		5	r_adc_spacing	insert 1 cycle spacing between ADC samples	0x0	R/W
		4	r_adc_retime_edge	retime edge selection. 0: rising edge of adc_clk 1: falling edge of adc_clk	0x0	R/W
		3	r_conv_fromint	conv_start select control. 1: conv_start from INT pin. 0: conv_start from register r_conv_start.	0x0	R/W
		2	r_conv_start	register control for conv_start	0x0	R/W
		1	RESERVED	Reserved.	0x0	R
		0	r_adc_tm_diffn_pdiff		0x0	R/W
0x4F	MISC	15	RESERVED	Reserved.	0x0	R
		[14:8]	r_div		0x20	R/W
		7	r_div_en		0x1	R/W
		6	r_tcli_oe	TCLI pin output enable	0x0	R/W
		5	r_tcli_ie	TCLI pin input buffer enable	0x0	R/W
		4	r_sda_slope_en	SDA pad slope control. 0: disable slew rate control. 1: enable slew rate control (default)	0x1	R/W
		[3:2]	r_ext_sync_sel	External Sample Sync 00: use internal 32Khz timer with r_gest_freq, r_prox_freq to select sample timing (default) 01: use INT input high to trigger sample 10: use TCLI input high to trigger sample 11: reserved	0x0	R/W
		1	r_int_ie	INT pin input enable	0x0	R/W
		0	r_bypass	SDA input i2c glitch buffer bypass 0: i2c 1: bypass glitch buffer	0x0	R/W

Addr	Name	Bits	Bit Name	Description	Reset	Access
0x50	TEST_MODE	[15:14]	r_int_led_en	use INT pin to control LED. 0; normal operation 1: drive INT pin value to LED1 2: drive INT pin value to LED2 3: drive INT pin value to LED3	0x0	R/W
		[13:8]	r_pin_test	PIN mux during test mode. connect analog signal to INT pad mux controlled by bits: 8: itest_bg (bandgap test current) 9: itest_bias (bias test current) 10: vref_bg 11: gnd 12: gnd 13: gnd	0x0	R/W
		7	r_use_data_testpattern	use r_prox_th_on1 in place of adc data. 0: normal operation (default) 1: use r_prox_th_on1 in place of all adc data for software dataflow debug	0x0	R/W
		6	r_tcli_out_test	controls tcli output. 0: driven low when enabled by r_tcli_oe 1: driven by channel 1 TIA powerdown signal	0x0	R/W
		5	r_osc32m_cal_en	osc32m calibration enable	0x0	R/W
		4	r_data_mask	mask ADC data output. 0: normal. 1: mask ADC output by disabling sample of ADC output.	0x0	R/W
		[3:0]	r_test_mode	test mode setting. 0: normal INT pin function 1: observe OS32K on INT pin. 2: observe OS32M/32 (1MHz) on INT pin. 3: observe OS32M/(2^16) on INT pin. 4: observe OS32K/(2^11) on INT pin. 5: observe 0 on INT pin. 6: observe 1 on INT pin. 7 observe adc_busy on INT pin. 8: observe afe_clk[0] on INT pin. 9: observe afe_clk[1] on INT pin. 10: observe afe_qreset on INT pin. 11: observe aoc_pulse on INT pin. 12: observe swmod_pulse on INT pin. 13: observe afe_pd[0] (tia on channel1) R2 14: observe afe_pd[6] (afe vrefs) R2 others: normal function.	0x0	R/W
0x51	TEST_USE_I2C	[15:14]	RESERVED	Reserved.	0x0	R
		13	test_timeout_en	continuous mode timeout enable	0x0	R/W
		12	test_use_i2c_enable	enable using I2C pins for data output	0x0	R/W
		[11:0]	test_i2c_time	timer of using I2C pin for data output. (counted by SCL divided by r_div	0x0	R/W

Addr	Name	Bits	Bit Name	Description	Reset	Access
0x52	TEST_PD	15	RESERVED	Reserved.	0x0	R
		14	r_adc_pdb	power down of ADC (low active)	0x0	R/W
		[13:12]	r_led_pdb	power down of LED driver (low active	0x0	R/W
		11	r_os32m_pdb	power down of 32MHz oscillator (low active)	0x0	R/W
		10	r_ref_pdb	power down of REF(low active)	0x0	R/W
		9	r_bg_pdb	power down of bandgap (low active)	0x0	R/W
		8	r_bias_pdb	power down of bias distribution (LOW active)	0x0	R/W
		7	r_dev_config_enable	device slave address configure enable. This has no function since TEST1 and TEST2 are hardcoded to 0 on 1022 0: slave address (I2C) = 0x64. 1: slave address (I2C) is based on TEST2, TEST1 pin status. 00: 0x64, 01: 0x6C, 10: 0x44, 11: 0x4C,	0x0	R/W
		[6:0]	r_pd_force	pown down control force mode. each bit control component bit[0]: Bias bit[1]: BG bit[2]: REF bit[3]: OS32M bit[4]: LED bit[5]: ADC bit[6]: AFE value = 0; power control comes from state machine. value = 1; power control comes from this register (except AFE) for AFE, when r_pd_force[6] is 1, power down control signals come from internal AFE control logic (default). When set to 0, the AFE is powered up for all of the ACTIVE3 state	0x40	R/W
0x53	TEST_AFE_SEL_MODE	[15:8]	r_afech_reorder	AFE channel select reorder bits. [1:0] - AFE channel0 reorder select signal [3:2] - AFE channel1 reorder select signal [5:4] - AFE channel2 reorder select signal [7:6] - AFE channel3 reorder select signal	0xE4	R/W
		[7:5]	RESERVED	Reserved.	0x0	R
		4	r_afe_sel_test_en	AFE select test mode enable bit. 'AFE select' test mode enable signal 0 - disable AFE select test mode 1 - enable AFE select test mode	0x0	R/W
		[3:0]	r_afe_sel	AFE channel select bits in AFE select test mode. bit[0]:AFE channel 1 select signal in 'AFE select' test mode bit[1]:AFE channel 2 select signal in 'AFE select' test mode bit[2]:AFE channel 3 select signal in 'AFE select' test mode bit[3]:AFE channel 4 select signal in 'AFE select' test mode	0x0	R/W

Addr	Name	Bits	Bit Name	Description	Reset	Access
0x54	AFE_POWER_CTL	15	r_single_chan_dig_int_b	single channel digital integrate select timeslot b. 0: use all four photodiode channels in parallel (default) 1: use only the first photodiode channel	0x0	R/W
		14	r_single_chan_dig_int_a	single channel digital integrate select timeslot a. 0: use all four photodiode channels in parallel (default) 1: use only the first photodiode channel	0x0	R/W
		[13:12]	r_sleep_cathode	cathode voltage during sleep or idle. Select cathode voltage during sleep or idle 00: VDD 01: AFE Vref -- VDD during sleep 10: FLOATING (new in R5) 11: GND	0x0	R/W
		[11:10]	r_tsb_cathode	cathode voltage for timeslot b. Select cathode voltage for timeslot b: 00: VDD 01: AFE Vref 10: V_Delta 11: GND	0x2	R/W
		[9:8]	r_tsa_cathode	cathode voltage for timeslot a. Select cathode voltage for timeslot a: 00: VDD 01: AFE Vref 10: V_Delta 11: GND	0x2	R/W
		7	r_flex_cathode	enable flexible cathode control. Enable flexible cathode control. 0: control cathode according to afe_ctrl2 (0x3c) 1: select cathode voltages based on the bits in this register for sleep, timeslot a and timeslot b	0x1	R/W
		[6:4]	r_cmclk_width	cmclk duration in us. Number of us that cmclk will run during preconditioning time. A value of zero will disable the cmclk during this time. The frequency is still set by the r_cmclk_sel field in register 0x3c	0x2	R/W
		[3:2]	r_afe_vref_power	afe amplifier reference power control. 00: follow state machine, on for active 2,3 (default) 01: on if any amp is on (like r0/r1) 10: always on 11: always off	0x0	R/W
		[1:0]	r_afe_amps_power	afe amplifier power control. 00: follow state machine (default) 01: force on active 2, active 3 10: always on 11: always off these modes do not override the power masks	0x0	R/W

Addr	Name	Bits	Bit Name	Description	Reset	Access
0x55	AFE_CHANNEL_TRIM	15	r_digint_tia234_pd_en	Digital integrate tia234 powerdown enable. Set this bit to 1 to turn off the tia amplifiers in channels 2,3,4 during single channel digital integrate cycles	0x0	R/W
		14	r_digint_int234_pd_en	Digital integrate int234 powerdown enable. Set this bit to 1 to turn off the integrators in channels 2,3,4 during single channel digital integrate cycles	0x0	R/W
		13	r_digint_bpf234_pd_en	Digital integrate bpf234 powerdown enable. Set this bit to 1 to turn off the bandpass filter in channels 2,3,4 during all digital integrate cycles	0x0	R/W
		12	r_digint_bpf1_pd_en	Digital integrate bpf1 powerdown enable. Set this bit to 1 to turn off the bandpass filter in channel 1 during all digital integrate cycles	0x0	R/W
		[11:10]	r_afe_trim_tia_4_b	TIA feedback resister select (25K ~ 200K) 0:200KΩ 1:100KΩ 2:50KΩ 3:25KΩ	0x0	R/W
		[9:8]	r_afe_trim_tia_3_b	TIA feedback resister select (25K ~ 200K) 0:200KΩ 1:100KΩ 2:50KΩ 3:25KΩ	0x0	R/W
		[7:6]	r_afe_trim_tia_2_b	TIA feedback resister select (25K ~ 200K) 0:200KΩ 1:100KΩ 2:50KΩ 3:25KΩ	0x0	R/W
		[5:4]	r_afe_trim_tia_4_a	TIA feedback resister select (25K ~ 200K) 0:200KΩ 1:100KΩ 2:50KΩ 3:25KΩ	0x0	R/W
		[3:2]	r_afe_trim_tia_3_a	TIA feedback resister select (25K ~ 200K) 0:200KΩ 1:100KΩ 2:50KΩ 3:25KΩ	0x0	R/W
		[1:0]	r_afe_trim_tia_2_a	TIA feedback resister select (25K ~ 200K) 0:200KΩ 1:100KΩ 2:50KΩ 3:25KΩ	0x0	R/W

Addr	Name	Bits	Bit Name	Description	Reset	Access
0x56	R5_OPTIONS	[15:14]	r_cathode_toggle_val_b	Cathode value for timeslot b in alternate timeslots	0x0	R/W
		[13:12]	r_cathode_toggle_val_a	Cathode value for timeslot a in alternate timeslots	0x0	R/W
		11	r_cathode_toggle_en_b	Enable cathode toggle for timeslot b. Set this bit to 1 to have cathode toggle between the timeslot b value and cathode_toggle_val_b on alternating timeslots	0x0	R/W
		10	r_cathode_toggle_en_a	Enable cathode toggle for timeslot a. Set this bit to 1 to have cathode toggle between the timeslot a value and cathode_toggle_val_a on alternating timeslots	0x0	R/W
		[9:6]	r_adc_chan_disable	ADC channel disable. Used to disable channel to allow tighter pulse spacing. Setting the bit disables that channel's adc conversion. Must set r_ed_qreset_en to use this feature.	0x0	R/W
		5	r_pulse_vref_b	Enable pulsed vref mode timeslot b. 0: off 1: on	0x0	R/W
		4	r_pulse_vref_a	Enable pulsed vref mode timeslot a. 0: off 1: on	0x0	R/W
		[3:2]	r_pulse_vref_val_b	Vref value for timeslot b in pulse vref mode. 0: 1.1385V 1: 1.012V 2: 0.8855V 3: 1.265V	0x0	R/W
		[1:0]	r_pulse_vref_val_a	Vref value for timeslot a in pulse vref mode 0: 1.1385V 1: 1.012V 2: 0.8855V 3: 1.265V	0x0	R/W
0x57	EFUSE_CTRL	[15:3]	RESERVED	Reserved.	0x0	R
		[2:1]	r_efuse_en	Efuse enable: 00: off (efuse held in reset) 01: reserved 10: standby (efuse in low power state 11: on (efuse fully operational) Transistions from 00 to 11 cause shadow registers to update from fuses. Must be in on state to do refresh, bist, or programming. Off and standby states have lowest power.	0x0	R/W
		0	r_efuse_reg_en	Efuse register enable bit. 0: access normal data registers 1: Enable EFuse registers accessible from address x61-x7f	0x0	R/W

Addr	Name	Bits	Bit Name	Description	Reset	Access
0x58	CALIBRATE	15	r_cal_enb	gain cal disable. master disable for gain calibration circuit 0: enable functions as configured by other bits 1: force disabled	0x0	R/W
		14	r_cal_switch_low	gain cal force switch low. 0: sw_cal high when not operating, switches low during cal cycle (default, normal operation) 1: sw_cal low always	0x0	R/W
		13	r_digital_integrate_b_en	digital integrate mode enable timeslot b. Digital integrate mode enable for timeslot b 0: normal analog integration mode 1: digital 1MHz integrate mode, single channel output	0x0	R/W
		12	r_digital_integrate_a_en	digital integrate mode enable timeslot a. Digital integrate mode enable for timeslot a 0: normal analog integration mode 1: digital 1MHz integrate mode, single channel output	0x0	R/W
		[11:10]	r_alternate_sub_ext_b_en	subract alternate samples 3,4 b. New in R3: Mode control for subtracting or adding samples. These bits control the sample in the third and fourth pulse positions for every group of 4 pulses. 00: add third and fourth samples (default) 01: add third, then subtract fourth 10: subtract third, then add fourth 11: subtract both third and fourth	0x0	R/W
		[9:8]	r_alternate_sub_ext_a_en	subract alternate samples 3,4 a. New in R3: Mode control for subtracting or adding samples. These bits control the sample in the third and fourth pulse positions for every group of 4 pulses. 00: add third and fourth samples (default) 01: add third, then subtract fourth 10: subtract third, then add fourth 11: subtract both third and fourth	0x0	R/W
		7	r_digital_int_invert	digital integration analog inversion. 0: non-inverting analog signal path. Current in causes increasing ADC value. 1:inverting analog signal path: Current in causes decreasing ADC value.	0x0	R/W
		[6:5]	r_alternate_sub_b_en	subract alternate samples 1,2 b. Mode control for subtracting or adding samples. These bits control the sample in the first two pulse positions for every group of 4 pulses. 00: add first two samples (default) 01: add first sample, then subtract second 10: subtract first, then add second 11: subtract both first and second	0x0	R/W
		4	r_gaincalibrate_b_en	gain calibration mode enable b. Enable bit for gain calibration in timeslot B 0: normal operation (default) 1: enable gain calibration function	0x0	R/W
		3	RESERVED	Reserved.	0x0	R
		[2:1]	r_alternate_sub_a_en	subract alternate samples 1,2 a. Mode control for subtracting or adding samples. These bits control the sample in the first two pulse positions for every group of 4 pulses. 00: add first two samples (default) 01: add first sample, then subtract second 10: subtract first, then add second 11: subtract both first and second	0x0	R/W
		0	r_gaincalibrate_a_en	gain calibration mode enable a. Enable bit for gain calibration in timeslot A 0: normal operation (default) 1: enable gain calibration function	0x0	R/W

Addr	Name	Bits	Bit Name	Description	Reset	Access
0x59	QRESET_VCM_B	15	RESERVED	Reserved.	0x0	R
		[14:13]	r_modulate_type_b	modulation type for timeslot b. When pd modulate is set for timeslot a choose type of pd access: 00: hold at vcm voltage between accesses (no float) 01: precondition to vcm voltage, float until modulate pulse 10: reserved 11: precondition using afe amplifier, float until modulate pulse in R0/R1 bit 13 was r_vcm_float_en: 0: vcm between accesses 1: float between accesses	0x0	R/W
		[12:8]	r_vcm_width_b	vcm pulse width in 1us steps for timeslot b	0x8	R/W
		[7:0]	r_qreset_offset_b	qreset delay in 1us steps for timeslot b	0x8	R/W

Addr	Name	Bits	Bit Name	Description	Reset	Access
0x5A	TIMING_OPTIONS	[15:12]	r_fgl_pulse_mask_b	float gesture led pulse mask timeslot b. Pulse mask for float gesture alternate LED pulses. Spans four pulses. 0s in the bit position cause the LED flash in the associated float period. The LSB is the first pulse time. Pulses are made based on the FGL_WIDTH and offset and repeat with PULSE_PERIOD.	0x0	R/W
		[11:8]	r_fgl_pulse_mask_a	float gesture led pulse mask timeslot a. Pulse mask for float gesture alternate LED pulses. Spans four pulses. 0s in the bit position cause the LED flash in the associated float period. The LSB is the first pulse time. Pulses are made based on the FGL_WIDTH and offset and repeat with PULSE_PERIOD.	0x0	R/W
		7	r_digital_integrate_gap_en	Digital integrate mode gap enable. 0: no gap between subtract and add regions (default) 1: use afe_fine_offset to specify gap in us	0x0	R/W
		6	r_dig_integrate_oneshot_b	digital integrate oneshot mode timeslot b. 0: use two light and dark regions (default) 1: use one light and dark region	0x0	R/W
		5	r_dig_integrate_oneshot_a	digital integrate oneshot mode timeslot a. 0: use two light and dark regions (default) 1: use one light and dark region	0x0	R/W
		4	r_ed_qreset_en	enable equidistant qreset timing. 0: use standard (r0-r2 type) qreset timing (default) 1: use equidistant qreset timing (generated just before afe clocks)	0x1	R/W
		3	r_read_sync_data	read sync data. There should be no reason to use this 0: normal operation. (default) 1: force fifo_clock latch of registers. Requires enable of fifo_clock for all register reads.	0x0	R/W
		2	r_regop_enb	afe signal register disable. There should be no reason to use this 0: use latches on all afe controls to prevent glitches 1: disable latches	0x0	R/W
		1	r_afe_optperiod_enb	disable optimized led-afe timing. There should be no reason to use this 0: use new optimized timing for afe 1: use old 1020 timing	0x0	R/W
		0	r_qreset_opt_enb	new qreset timing disable. 0: use new qreset timing 1: use old qreset timing (ignore new controls) There should be no reason to use this.	0x0	R/W

Addr	Name	Bits	Bit Name	Description	Reset	Access
0x5B	AO_CANCEL_A	15	RESERVED	Reserved.	0x0	R
		[14:8]	r_aocancel_width_a	timeslot a aocancel pulse with in 31.25ns steps 0=disable	0x0	R/W
		[7:0]	r_aocancel_offset_a	timeslot a aocancel pulse offset in 1us steps	0x0	R/W
0x5C	AO_CANCEL_PROX_A	15	RESERVED	Reserved.	0x0	R
		[14:8]	r_aocancel_width_prox_a	timeslot a aocancel pulse with in 31.25ns steps 0=disable	0x0	R/W
		[7:0]	r_aocancel_offset_prox_a	timeslot a aocancel pulse offset in 1us steps	0x0	R/W
0x5D	AO_CANCEL_B	15	RESERVED	Reserved.	0x0	R
		[14:8]	r_aocancel_width_b	timeslot a aocancel pulse with in 31.25ns steps 0=disable	0x0	R/W
		[7:0]	r_aocancel_offset_b	timeslot a aocancel pulse offset in 1us steps	0x0	R/W
0x5E	QRESET_VCM	15	RESERVED	Reserved.	0x0	R
		[14:13]	r_modulate_type_a	modulation type for timeslot a. When pd modulate is set for timeslot a choose type of pd access: 00: hold at vcm voltage between accesses (no float) 01: precondition to vcm voltage, float until modulate pulse 10: reserved 11: precondition using afe amplifier, float until modulate pulse in R0/R1 bit 13 was r_vcm_float_en: 0: vcm between accesses 1: float between accesses	0x0	R/W
		[12:8]	r_vcm_width_a	vcm pulse width in 1us steps for timeslot a	0x8	R/W
		[7:0]	r_qreset_offset_a	qreset delay in 1us steps for timeslot a	0x8	R/W
0x5F	ACCESS_CTRL	[15:3]	RESERVED	Reserved.	0x0	R
		2	r_hold_regs_b	Prevent Timeslot B data register update. 0: allow updates 1: hold current contents, delay updates	0x0	R/W
		1	r_hold_regs_a	Prevent Timeslot A data register update. 0: allow updates 1: hold current contents, delay updates	0x0	R/W
		0	r_fifoclk_force	Fifo Clock enable. Set to 1 to force FIFO clocking on. Starting in R5 it is not necessary to set this to operate the FIFO. This enables the 32MHz oscillator in all versions.	0x0	R/W

Addr	Name	Bits	Bit Name	Description	Reset	Access
0x60	DATA_BUFFER	[15:0]	data_buffer_out	data of next available word (16bit) in FIFO.	0x0	R
0x61	I	[15:0]	sampleI	I value	0x0	R
0x62	X	[15:0]	sampleX	X value	0x0	R
0x63	Y	[15:0]	sampleY	Y value	0x0	R
0x64	X1_a	[15:0]	readX1_a	X1 value	0x0	R
0x65	X2_a	[15:0]	readX2_a	X2 value	0x0	R
0x66	Y1_a	[15:0]	readY1_a	Y1 value	0x0	R
0x67	Y2_a	[15:0]	readY2_a	Y2 value	0x0	R
0x68	X1_b	[15:0]	readX1_b	X1 value	0x0	R
0x69	X2_b	[15:0]	readX2_b	X2 value	0x0	R
0x6A	Y1_b	[15:0]	readY1_b	Y1 value	0x0	R
0x6B	Y2_b	[15:0]	readY2_b	Y2 value	0x0	R
0x6C	READ_MINDIDT	[15:0]	read_mindidt		0x0	R
0x6D	READ_MAXDIDT	[15:0]	read_maxdidt		0x0	R
0x70	READ_X1L_a	[15:0]	read_x1l_a		0x0	R
0x71	READ_X2L_a	[15:0]	read_x2l_a		0x0	R
0x72	READ_Y1L_a	[15:0]	read_y1l_a		0x0	R
0x73	READ_Y2L_a	[15:0]	read_y2l_a		0x0	R
0x74	READ_X1H_a	[15:0]	read_x1h_a		0x0	R
0x75	READ_X2H_a	[15:0]	read_x2h_a		0x0	R
0x76	READ_Y1H_a	[15:0]	read_y1h_a		0x0	R
0x77	READ_Y2H_a	[15:0]	read_y2h_a		0x0	R
0x78	READ_X1L_b	[15:0]	read_x1l_b		0x0	R
0x79	READ_X2L_b	[15:0]	read_x2l_b		0x0	R
0x7A	READ_Y1L_b	[15:0]	read_y1l_b		0x0	R
0x7B	READ_Y2L_b	[15:0]	read_y2l_b		0x0	R
0x7C	READ_X1H_b	[15:0]	read_x1h_b		0x0	R
0x7D	READ_X2H_b	[15:0]	read_x2h_b		0x0	R
0x7E	READ_Y1H_b	[15:0]	read_y1h_b		0x0	R
0x7F	READ_Y2H_b	[15:0]	read_y2h_b		0x0	R

Efuse register map:

These efuse control registers appear in these locations when the r_efuse_reg_en bit is set in the EFUSE_CTRL register.

Address	Name	Description	Reset	Access
0x61	NVM_KEY_R	NVM_KEY (Read Functionality) : Read-back values of the unlock status of the shadow registers and the refreshed lockout bit.	0x00	R
0x61	NVM_KEY_W	NVM_KEY (Write Functionality) : Notional write register address for locking/unlocking the shadow registers, or activating fuse programming.	0x00	W
0x62	CONTROL_BMR	CONTROL_BMR : Control register used to invoke BIST, margining and refreshes	0x00	R/W
0x63	CONTROL_PS	CONTROL_PS : Control register which defines the start address of a fuse region for programming or BIST	0x00	R/W
0x64	CONTROL_PE	CONTROL_PE : Control register which defines the end address of a fuse region for programming or BIST.	0x0F	R/W
0x65	CONTROL_PP	CONTROL_PP : Width minus one in system clock cycles of the fuse programming pulse.	0x14	R/W
0x66	CONTROL_PI	CONTROL_PI : Width minus one in system clock cycles of half of the idle interval between pulses during normal (non-BIST) fuse programming.	0x14	R/W
0x67	NVM_STATUS0	NVM_STATUS0 : Status diagnostics register.	0x00	R
0x68	NVM_STATUS1	NVM_STATUS1 : Status diagnostics register.	0x00	R
0x69	NVM_STATUS2	NVM_STATUS2 : Status diagnostics register.	0x00	R
0x6A	NVM_STATUS3	NVM_STATUS3 : Status diagnostics register.	0x00	R
0x70 - 0x7F	SHADOW[n]	SHADOW : Shadow registers for the NVM fuses	0x00	R/W

Detailed E-fuse Register map with bitfield descriptions

Addr	Name	Bits	Bit Name	Description	Reset	Access
0x61	NVM_KEY_R	[7:2]	RESERVED	Reserved.	0x0	R
		1	LOCKOUT	Refreshed value of the lockout fuse bit. 0: The NVM fuse banks are not locked out from fuse programming. 1: Fuse programming is permanently disabled.	0x0	R
		0	NVM_UNLOCK	Unlock status of the shadow registers, determined by writes of key values to NVM_KEY. 0: The shadow registers are currently locked. 1: The shadow registers are currently unlocked.	0x0	R
0x61	NVM_KEY_W	[7:0]	KEY	KEY : 8-bit write data of values NVM_UNLOCK_KEY, NVM_LOCK_KEY, or NVM_PROG_KEY - A write of value NVM_UNLOCK_KEY (0xe5) unlocks the shadow registers to enable virtual trimming and also allows subsequent fuse programming. - A write of value NVM_LOCK_KEY (0x00) locks the shadow registers out from virtual trimming and also prevents fuse programming. This is the defaults. - A write of value NVM_PROG_KEY (0xf0) invokes fuse programming on the configured fuse region.	0x0	W
0x62	CONTROL_BMR	[7:3]	RESERVED	Reserved.	0x0	R
		2	BIST	Invocation signal for built-in self-test in the ADI eFuse IP 0: Don't invoke BIST. 1: Invoke BIST. Note that BIST will run to completion if invoked.	0x0	R/W
		1	MARGIN	Enable margining during fuse reads 0: Don't enable margining. Use normal reference resistance to determine if a fuse is blown. 1: Enable margining. Use a higher reference resistance to determine if a fuse is blown.	0x0	R/W
		0	REFRESH	Invoke a fuse refresh into the shadow registers. 0: Don't invoke a refresh 1: Invoke a refresh of the entire fuse array into the shadow registers.	0x0	R/W
0x63	CONTROL_PS	[3:0]	CONTROL_PS	Start address for a fuse region, used during programming or BIST	0x0	R/W
0x64	CONTROL_PE	[3:0]	CONTROL_PE	End address for a fuse region, used during programming or BIST.	0xF	R/W
0x65	CONTROL_PP	[7:0]	CONTROL_PP	Width minus one in system clock cycles of the fuse programme pulse. **Change this to 0x1d for normal programming.**	0x14	R/W
0x66	CONTROL_PI	[7:0]	CONTROL_PI	Width minus one in system clock cycles of half of the idle interval between pulses during normal (non-BIST) fuse programming. **Change this to 0x1d for normal programming.**	0x14	R/W

Addr	Name	Bits	Bit Name	Description	Reset	Access
0x67	NVM_STATUS0	7	BIST_FAIL	Pass/Fail status of the most recently invoked BIST run. 0x0: BIST passed when most recently run on the fuse region defined by [CONTROL_PS, CONTROL_PE] 0x1: BIST failed when most recently run on the fuse region defined by [CONTROL_PS, CONTROL_PE]	0x0	R
		[6:5]	BIST_STATE	Value of the BIST FSM state vector in the "bankfuse_control" module. 0x0: FSM is in the BIST_IDLE state. 0x1: FSM is in the BIST_START state. 0x2: FSM is in the BIST_RUN state. 0x3: FSM is in the BIST_DONE state.	0x0	R
		[4:3]	PROGRAM_STATE	Value of the program FSM state vector in the "bankfuse_control" module. 0x0: FSM is in the PGM_IDLE state. 0x1: FSM is in the PGM_BYTE state. 0x2: FSM is in the PGM_BIT state. 0x3: FSM is in the PGM_WAIT state.	0x0	R
		[2:1]	REFRESH_STATE	Value of the refresh FSM state vector in the "bankfuse_control" module. 0x0: FSM is in the REF_START state. 0x1: FSM is in the REF_READ state. 0x2: FSM is in the REF_DONE state. 0x3: FSM is in the REF_HOLDOFF state.	0x0	R
		0	NVM_BUSY	Flag which indicates that the IP is busy or idle. 0: The IP is idle. No programming, BIST or refreshes are currently beng carried out. 1: The IP is busy due to programming, BIST or refresh operations.	0x0	R
0x68	NVM_STATUS1	[7:1]	NVMADDR	Fuse byte pointer used in programmng, refresh and BIST operations.	0x0	R
		0	PROGRAM_BUSY	Flag which indicates that a fuse programming sequence is currently under way. 0x0: Fuse programming is not currently occurring. 0x1: Fuse programming is currently occurring	0x0	R
0x69	NVM_STATUS2	[7:0]	BIST_FAIL_LOC	Diagnostic information on the location and nature of the most recently encountered BIST failure.	0x0	R
0x6A	NVM_STATUS3	[7:3]	RESERVED	Reserved.	0x0	R
		[2:0]	BIST_FAIL_LOC	Location and nature of the first of any BIST failures encountered.	0x0	R
0x70 0x7F	SHADOW[n]	[7:0]	SHADOW	Shadow register 0x00: Shadow register value.	0x0	R/W