

Click [here](#) for production status of specific part numbers.

MAX86160

Integrated Heart-Rate Sensor for In-Ear Applications

General Description

The MAX86160 is an integrated heart rate monitor sensor module designed for the demanding requirements of mobile, wearable, and hearable devices. It includes internal LEDs, photo-detector, and low-noise electronics with high-dynamic-range ambient light rejection. This integrated product has evaluation software available that includes algorithms for detecting heart rate. The MAX86160 sensor module takes care of the most challenging parts of the design for fast time-to-market in mobile and wearable devices.

The MAX86160 operates on a 1.8V supply voltage, with a separate 3.3V/5.0V power supply for the internal LEDs. Communication to and from the module occurs entirely through a standard I²C-compatible interface. The module can be shut down through software with near zero standby current, allowing the power rails to remain powered at all times.

Benefits and Features

- Miniature 4.3mm x 2.8mm x 1.45mm 18-pin Optical Module
 - Optical-Grade, Robust Glass Eliminates Customer Cover Glass
- High In-Band 13Hz Signal-to-Noise Ratio (SNR) Reflective Heart Rate Monitor and Medical-Grade Pulse Oximeter
- Ultra Low-Power Operation for Mobile Device
 - Zero-Power Shutdown Current (0.7μA, typ)
- -40°C to +85°C Operating Temperature Range

Applications

- Wearable and Hearable Devices
- Smartphones/Tablets
- Disposable Patch Sensors
- Fitness Assistant Devices

Ordering Information appears at end of data sheet.

Simplified Block Diagram

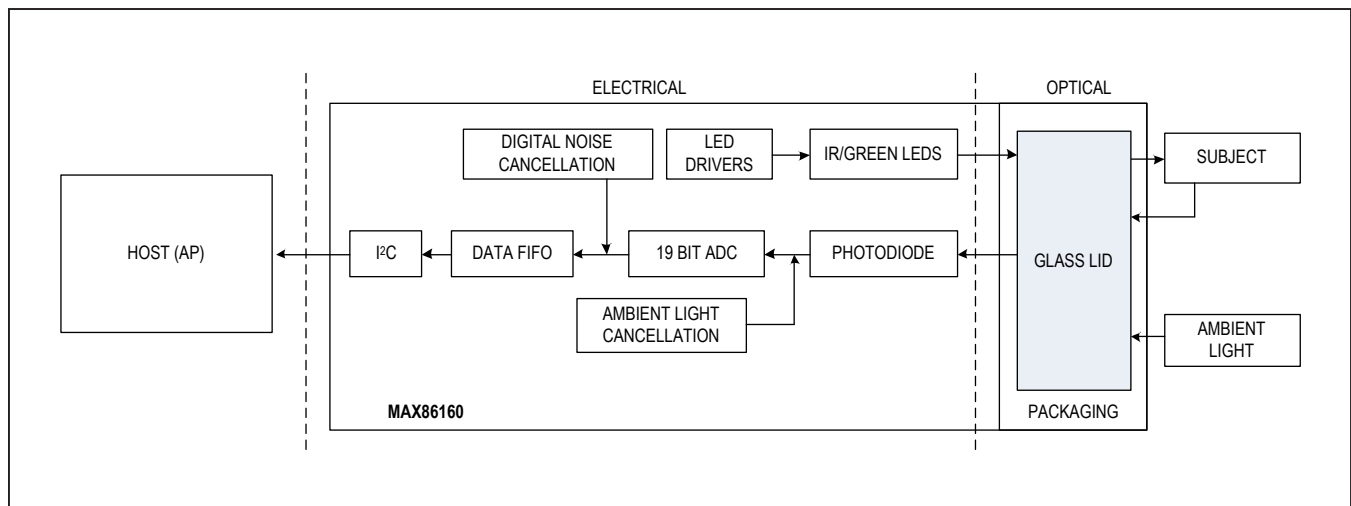


TABLE OF CONTENTS

| | |
|--|----|
| General Description | 1 |
| Benefits and Features | 1 |
| Applications | 1 |
| Simplified Block Diagram | 1 |
| Absolute Maximum Ratings | 6 |
| Package Information | 6 |
| 18-Lead OESIP | 6 |
| Electrical Characteristics | 7 |
| Typical Operating Characteristics | 10 |
| Pin Configurations | 11 |
| Pin Description | 11 |
| Functional Diagram | 12 |
| Detailed Description | 13 |
| HRM Subsystem | 13 |
| LED Driver | 13 |
| Proximity Function | 13 |
| Register Map | 13 |
| Interrupt Status 1 (0x00) | 15 |
| A_FULL: FIFO Almost Full Flag | 15 |
| PPG_RDY: New PPG FIFO Data Ready | 15 |
| ALC_OVF: Ambient Light Cancellation Overflow | 15 |
| PROX_INT: Proximity interrupt | 15 |
| PWR_RDY: Power Ready Flag | 15 |
| Interrupt Status 2 (0x01) | 16 |
| VDD_OOR: VDD Out-of-Range flag | 16 |
| Interrupt Enable 1 (0x02) | 16 |
| A_FULL_EN: FIFO Almost Full Flag enable | 16 |
| PPG_RDY_EN: New PPG FIFO Data Ready Interrupt enable | 16 |
| ALC_OVF_EN: Ambient Light Cancellation (ALC) Overflow Interrupt enable | 16 |
| PROX_INT_EN: Proximity Interrupt enable | 17 |
| Interrupt Enable 2 (0x03) | 17 |
| VDD_OOR_EN: VDD Out-of-Range Indicator enable | 17 |
| FIFO Write Pointer (0x04) | 17 |
| FIFO_WR_PTR: FIFO Write Pointer | 17 |
| OVF_COUNTER: FIFO Overflow Counter | 18 |
| FIFO Read Pointer (0x06) | 18 |
| FIFO_RD_PTR: FIFO Read Pointer | 18 |

TABLE OF CONTENTS (CONTINUED)

| | |
|---|----|
| FIFO Data Register (0x07) | 18 |
| FIFO_DATA: FIFO Data Register | 18 |
| FIFO Configuration (0x08) | 18 |
| A_FULL_CLR: FIFO Almost Full Interrupt Options | 18 |
| A_FULL_TYPE: FIFO Almost Full Flag Options | 19 |
| FIFO_ROLLS_ON_FULL: FIFO Rolls on Full Options | 19 |
| FIFO_A_FULL: FIFO Almost Full Value | 19 |
| FIFO Data Control Register 1 (0x09) | 19 |
| FD2: FIFO Data Time Slot 2 | 20 |
| FD1: FIFO Data Time Slot 1 | 20 |
| FD4: FIFO Data Time Slot 4 | 20 |
| FD3: FIFO Data Time Slot 3 | 21 |
| System Control (0x0D) | 21 |
| FIFO_EN: FIFO Enable | 21 |
| SHDN: Shutdown Control | 21 |
| RESET: Reset Control | 22 |
| PPG Configuration 1 (0x0E) | 22 |
| PPG_ADC_RGE: PPG ADC Range Control | 22 |
| PPG_SR: PPG Sample Rate Control | 22 |
| PPG Sample Rate Control | 22 |
| Maximum Sample rates Supported for all the Pulse Widths and Number of LEDs: | 23 |
| PPG_LED_PW: LED Pulse Width Control | 23 |
| PPG Configuration 2 (0x0F) | 23 |
| SMP_AVE: Sample Averaging Options | 23 |
| Prox Interrupt Threshold (0x10) | 24 |
| PROX_INT_THRESH: Proximity Mode Interrupt Threshold | 24 |
| LED1 PA (0x11) | 24 |
| LED1_PA: LED 1 (IR) Current Pulse Amplitude | 24 |
| LED3 PA (0x13) | 25 |
| LED3_PA: LED 3 (Green) Current Pulse Amplitude | 25 |
| LED Range (0x14) | 25 |
| LED3_PA: LED 3 (Green) Current Pulse Amplitude | 25 |
| LED1_RGE: LED 1 (IR) Current Control | 26 |
| LED PILOT PA (0x15) | 26 |
| PILOT_PA: Proximity Mode LED Pulse Amplitude | 26 |
| Part ID (0xFF) | 26 |
| PART_ID: Part Identifier | 26 |

TABLE OF CONTENTS (CONTINUED)

| | |
|--|----|
| Applications Information | 27 |
| Power Sequencing and Requirements | 27 |
| Power-Up Sequencing | 27 |
| Power-Down Sequencing | 27 |
| I ² C Interface | 27 |
| Bit Transfer | 27 |
| START and STOP Conditions | 28 |
| Early STOP Conditions | 28 |
| Slave Address | 28 |
| Write Data Format | 29 |
| Read Data Format | 30 |
| FIFO Description | 31 |
| Overview | 31 |
| FIFO Data Types | 32 |
| FIFO Data Control Registers | 32 |
| Write Pointer to the FIFO, FIFO_WR_PTR[4:0]: | 34 |
| Read Pointer to the FIFO, FIFO_RD_PTR[4:0]: | 34 |
| FIFO Data Read, FIFO_DATA[7:0]: | 34 |
| Reading from the FIFO | 34 |
| FIFO Flush | 37 |
| FIFO Organization | 38 |
| Typical Application Circuits | 40 |
| Ordering Information | 40 |
| Revision History | 41 |

LIST OF FIGURES

Figure 1. I²C-Compatible Interface Timing Diagram 9

Figure 2: Power-Up Sequence of the Power Supply Rails 27

Figure 3. START, STOP, and REPEATED START Conditions 28

Figure 4. I²C Acknowledge 29

Figure 5. Writing One Data Byte to MAX86160. 29

Figure 6. Reading One Byte of Data from MAX86160 30

Figure 7. Reading Multiple Bytes of Data from the MAX86160. 31

Figure 8. Example of FIFO Organization with Four Active Elements 38

Figure 9. Example of FIFO Organization with Two Active Elements. 39

LIST OF TABLES

Table 1: MAX86160 I²C Slave Address. 28

Table 2. FIFO Data Control Registers 32

Table 3. FDx Format Configurations 32

Table 4. FIFO Data Format 33

 Example 1: Configurations for 2 elements: PPG (LED1) + PPG (LED3) 33

 Example 2: Configurations for 1 elements: PPG (LED1) 33

Table 5. Sample of FIFO Data Index 33

Table 6. FIFO Handling Registers 33

Table 7. FIFO Sample Elements Order with Four Active Elements 34

Table 8. FIFO Sample Elements Order with Two Active Elements 35

Absolute Maximum Ratings

| | | | |
|---------------------------------------|----------------|---|-----------------|
| V _{DD} to GND | -0.3V to +2.2V | Continuous Power Dissipation | 440mW |
| PGND to GND | -0.3V to +0.3V | SDA, SCL, INTB, GPIO to GND | -0.3V to +6.0V |
| LED_DRVx, VLED to PGND | -0.3V to +6.0V | OESIP (derate 5.5mW/°C above +70°C) | -40°C to +85°C |
| V _{REF} to GND..... | -0.3V to +2.2V | Operating Temperature Range..... | -40°C to +85°C |
| Output Short-Circuit Duration..... | Continuous | Junction Temperature..... | +150°C |
| Continuous Input Current Into Any Pin | | Storage Temperature Range | -40°C to +105°C |
| (Except LED_DRVx Pins)..... | ±20mA | Soldering Temperature (Reflow)..... | +260°C |

Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated in the operational sections of the specifications is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

Package Information

18-Lead OESIP

| | |
|--|---------------------------|
| Package Code | F182A4+1 |
| Outline Number | 21-100099 |
| Land Pattern Number | 90-100030 |
| Thermal Resistance, Four Layer Board: | |
| Junction-to-Ambient (θ _{JA}) | 174°C/W |
| Junction-to-Case Thermal Resistance (θ _{JC}) | 150°C/W |

For the latest package outline information and land patterns (footprints), go to www.maximintegrated.com/packages. Note that a "+", "#", or "-" in the package code indicates RoHS status only. Package drawings may show a different suffix character, but the drawing pertains to the package regardless of RoHS status.

Package thermal resistances were obtained using the method described in JEDEC specification JESD51-7, using a four-layer board. For detailed information on package thermal considerations, refer to www.maximintegrated.com/thermal-tutorial.

Electrical Characteristics

($V_{DD} = 1.8V$, $V_{LED} = 3.3V$ (IR), $V_{LED} = 5.0V$ (GREEN), $GND = PGND = 0V$, $T_A = +25^\circ C$, min/max are from $T_A = -40^\circ C$ to $+85^\circ C$, unless otherwise noted.) (Note 1)

| PARAMETER | SYMBOL | CONDITIONS | MIN | TYP | MAX | UNITS |
|---|-------------|---|---------|---------|---------|---------|
| POWER SUPPLY | | | | | | |
| Power Supply Voltage | V_{DD} | | 1.7 | 1.8 | 2.0 | V |
| LED Supply Voltage for IR | V_{LED} | | 3.1 | 3.3 | 5.5 | V |
| LED Supply Voltage for Green | V_{LED} | | 4.0 | 5.0 | 5.5 | V |
| V_{DD} Supply Current | I_{DD} | Heart Rate Mode; PW = 50 μ s; SPS = 100; LED Driver = 0mA | | 400 | 750 | μ A |
| | | Heart Rate Mode; PW = 50 μ s; SPS = 10; LED Driver = 0mA | | 400 | 750 | |
| V_{DD} Current in Shutdown | | $T_A = 25^\circ C$ | | 0.5 | 12 | μ A |
| V_{LED} Current in Shutdown | | $T_A = 25^\circ C$ | | 0 | 1 | μ A |
| Internal Voltage Reference (Note 2) | V_{REF} | Bypass to GND with 1 μ F | 1.192 | 1.204 | 1.215 | V |
| PULSE OXIMETRY/HEART RATE SENSOR CHARACTERISTICS | | | | | | |
| ADC Resolution | | | | 19 | | bits |
| IR ADC Count | IR_C | Proprietary ATE Setup. IR_PA = 0x13, PW = 50 μ S, SPS = 1000, $T_A = +25^\circ C$ | 121,072 | 131,072 | 141,072 | Counts |
| Green ADC Count | GREEN_C | Proprietary ATE Setup. GREEN_PA = 0x80, PW = 50 μ S, SPS = 1000, $T_A = +25^\circ C$ | 111,072 | 131,072 | 151,072 | Counts |
| Dark Current Counts | DC_C | ALC = ON, IR_PA = 0x00, PW = 50 μ S, SPS = 1000, PPG_ADC_RGE<1:0> = 8 μ A, $T_A = +25^\circ C$ | | 0.0001 | 0.02 | % of FS |
| IR/GREEN ADC Count - PSRR (V_{DD}) | PSRR_VDD | Proprietary ATE setup, 1.7V < V_{DD} < 2.0V, IR_PA = 0x12, GREEN_PA = 0x80, PW = 50 μ S, SPS = 1000 | | 0.5 | 1 | % of FS |
| IR/GREEN ADC Count - PSRR (LED Driver Outputs) | PSRR_LED | Proprietary ATE setup, 3.1V < V_{LED} < 5V(IR), 4V < V_{LED} < 5.5V (GREEN), IR_PA = 0x12, GREEN_PA = 0x80, PW = 50 μ S, SPS = 1000 | | 0.05 | 0.5 | % of FS |
| ADC Clock Frequency | CLK | | 9.649 | 9.846 | 10.043 | MHz |
| IR LED CHARACTERISTICS (Note 5) | | | | | | |
| LED Peak Wavelength | λ_P | $I_{LED} = 20mA$, $T_A = +25^\circ C$ | 870 | 880 | 900 | nm |
| GREEN LED CHARACTERISTICS (Note 5) | | | | | | |
| LED Peak Wavelength | λ_P | $I_{LED} = 20mA$, $T_A = +25^\circ C$ | 520 | 527 | 540 | nm |

Electrical Characteristics (continued)

($V_{DD} = 1.8V$, $V_{LED} = 3.3V$ (IR), $V_{LED} = 5.0V$ (GREEN), $GND = PGND = 0V$, $T_A = +25^\circ C$, min/max are from $T_A = -40^\circ C$ to $+85^\circ C$, unless otherwise noted.) (Note 1)

| PARAMETER | SYMBOL | CONDITIONS | MIN | TYP | MAX | UNITS |
|--|---------------|---|-----|------|-----|---------|
| LED DRIVERS | | | | | | |
| LED Current Resolution | | | | 8 | | bits |
| LED Drive Current Range (Note 3) | I_{LED} | $V_{LED} = 5.0V$ (for GREEN), $V_{LED} = 3.3V$ (for IR ONLY), $LEDx_PA = 0xFF$, $LEDx_RGE[1:0] = 00$ | | 50 | | mA |
| | | $V_{LED} = 5.0V$ (for GREEN), $V_{LED} = 3.3V$ (for IR ONLY), $LEDx_PA = 0xFF$, $LEDx_RGE[1:0] = 01$ | | 100 | | |
| | | $V_{LED} = 5.0V$ (for GREEN), $V_{LED} = 3.3V$ (for IR ONLY), $LEDx_PA = 0xFF$, $LEDx_RGE[1:0] = 10$ | | 150 | | |
| | | $V_{LED} = 5.0V$ (for GREEN), $V_{LED} = 3.3V$ (for IR ONLY), $LEDx_PA = 0xFF$, $LEDx_RGE[1:0] = 11$ | | 200 | | |
| DIGITAL CHARACTERISTICS (SDA, SCL, INT) | | | | | | |
| Output Low Voltage SDA, INTB | V_{OL} | $I_{SINK} = 6mA$ | | | 0.4 | V |
| I ² C Input Voltage Low | V_{IL_I2C} | SDA, SCL | | | 0.4 | V |
| I ² C Input Voltage High | V_{IH_I2C} | SDA, SCL | 1.4 | | | V |
| Input Hysteresis (Note 5) | V_{HYS} | SDA, SCL | | 200 | | mV |
| Input Capacitance (Note 5) | C_{IN} | SDA, SCL | | 10 | | pF |
| Input Leakage Current | I_{IN} | $V_{IN} = 0V$, $T_A = +25^\circ C$ (SDA, SCL) | | 0.01 | 1 | μA |
| | | $V_{IN} = V_{DD}$, $T_A = +25^\circ C$ (SDA, SCL) | | 0.01 | 1 | |
| I²C TIMING CHARACTERISTICS (SDA, SCL) (Note 4, Figure 1) | | | | | | |
| I ² C Write Address | | | | BC | | Hex |
| I ² C Read Address | | | | BD | | Hex |
| Serial Clock Frequency | f_{SCL} | | 0 | | 400 | kHz |
| Bus Free Time Between STOP and START Conditions | t_{BUF} | | 1.3 | | | μs |
| Hold Time (Repeated) START Condition | $t_{HD,STA}$ | | 0.6 | | | μs |
| SCL Pulse-Width Low | t_{LOW} | | 1.3 | | | μs |
| SCL Pulse-Width High | t_{HIGH} | | 0.6 | | | μs |
| Setup Time for a Repeated START Condition | $t_{SU,STA}$ | | 0.6 | | | μs |
| Data Hold Time | $t_{HD,DAT}$ | | 0 | | 900 | ns |

Electrical Characteristics (continued)

($V_{DD} = 1.8V$, $V_{LED} = 3.3V(IR)$, $V_{LED} = 5.0V$ (GREEN), $GND = PGND = 0V$, $T_A = +25^{\circ}C$, min/max are from $T_A = -40^{\circ}C$ to $+85^{\circ}C$, unless otherwise noted.) (Note 1)

| PARAMETER | SYMBOL | CONDITIONS | MIN | TYP | MAX | UNITS |
|---------------------------------|--------------|------------|---------------|-----|-----|---------|
| Data Setup Time | $t_{SU,DAT}$ | | 100 | | | ns |
| Setup Time for STOP Condition | $t_{SU,STO}$ | | 0.6 | | | μs |
| Pulse Width of Suppressed Spike | t_{SP} | | 0 | 50 | | ns |
| Bus Capacitance | C_B | | | | 400 | pF |
| SDA and SCL Receiving Rise Time | t_R | | $20 + 0.1C_B$ | | 300 | ns |
| SDA and SCL Receiving Fall Time | t_F | | $20 + 0.1C_B$ | | 300 | ns |
| SDA Transmitting Fall Time | t_F | | $20 + 0.1C_B$ | | 300 | ns |

Note 1: All devices are 100% production tested at $T_A = +25^{\circ}C$. Specifications over temperature limits are guaranteed by Maxim Integrated's bench or proprietary automated test equipment (ATE) characterization.

Note 2: Internal Reference Voltage only.

Note 3: Whenever Green LED is used, V_{LED} must be 4.0V or above. For LED Current range more than 100mA ($LEDx_RGE = 2'b1X$), V_{LED} must be 4.5V or above.

Note 4: For design guidance only. Not production tested.

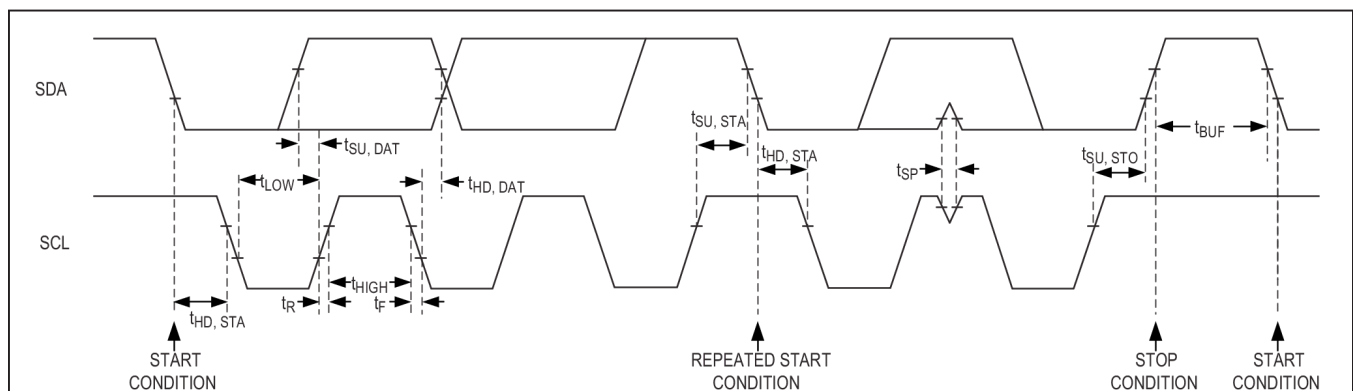
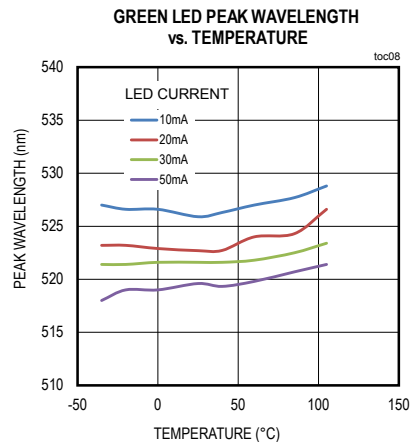
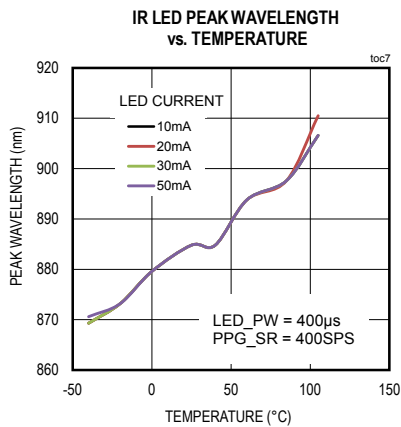
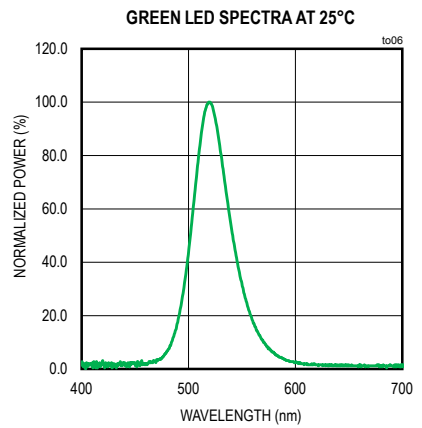
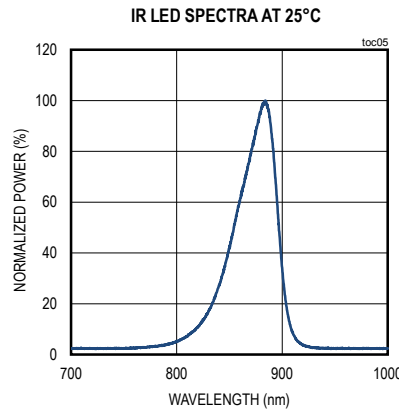
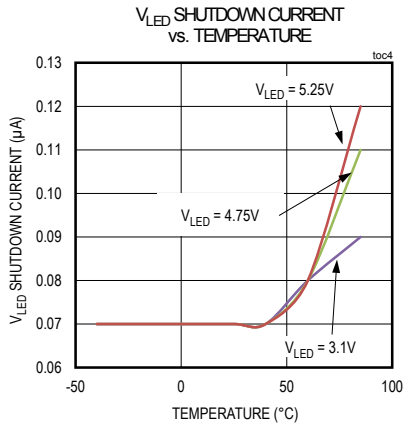
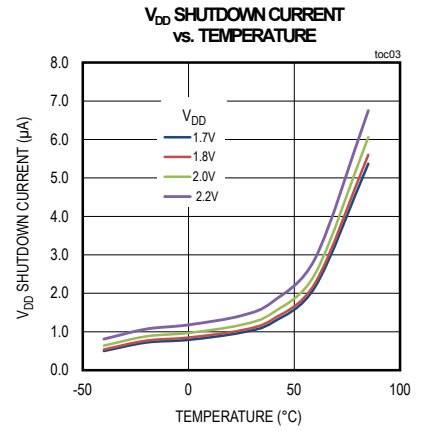
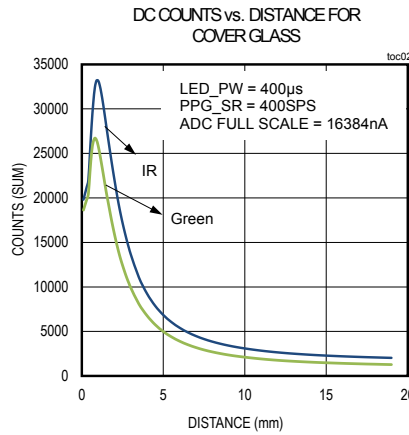
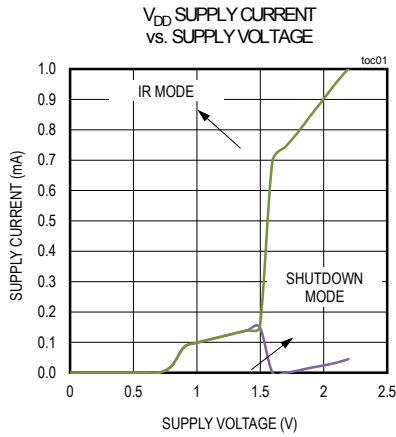


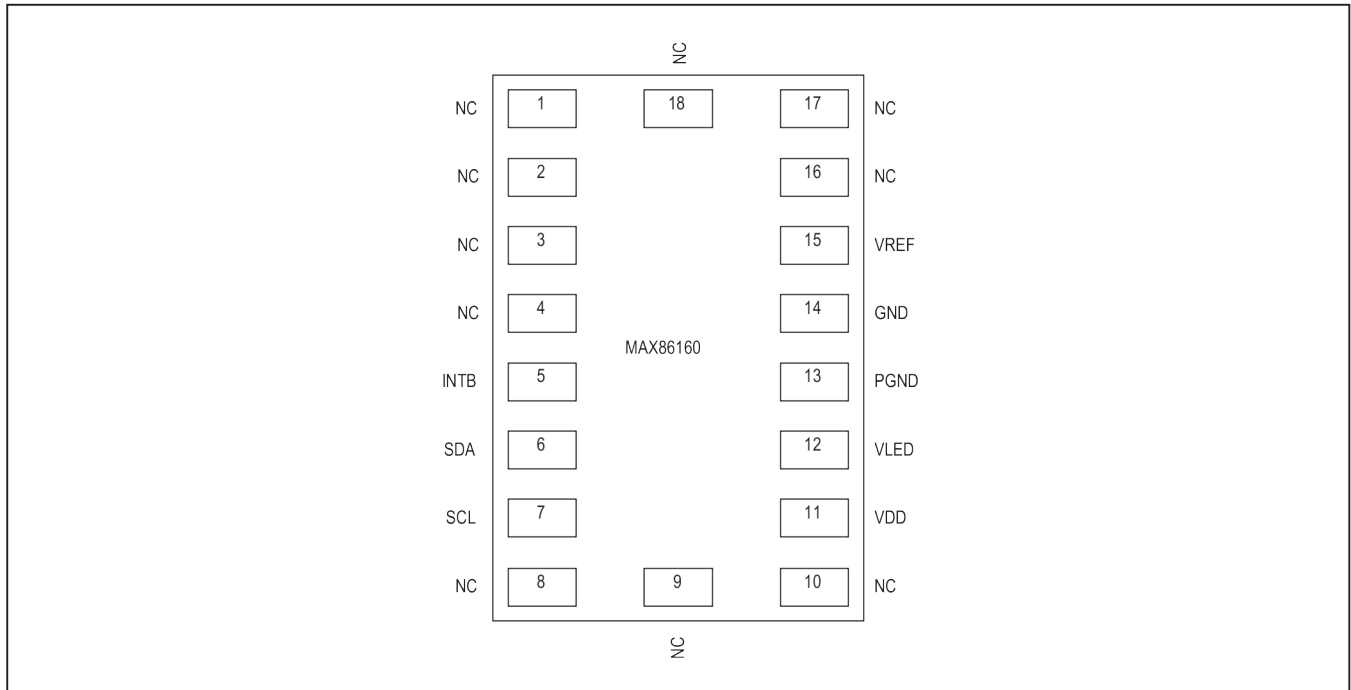
Figure 1. I²C-Compatible Interface Timing Diagram

Typical Operating Characteristics

($V_{DD} = 1.8V$, $V_{LED} = 3.3V$ (IR), $V_{LED} = 5.0V$ (GREEN), $GND = PGND = 0V$, $T_A = +25^\circ C$, unless otherwise noted.)($T_A = +25^\circ C$, unless otherwise noted.)



Pin Configurations

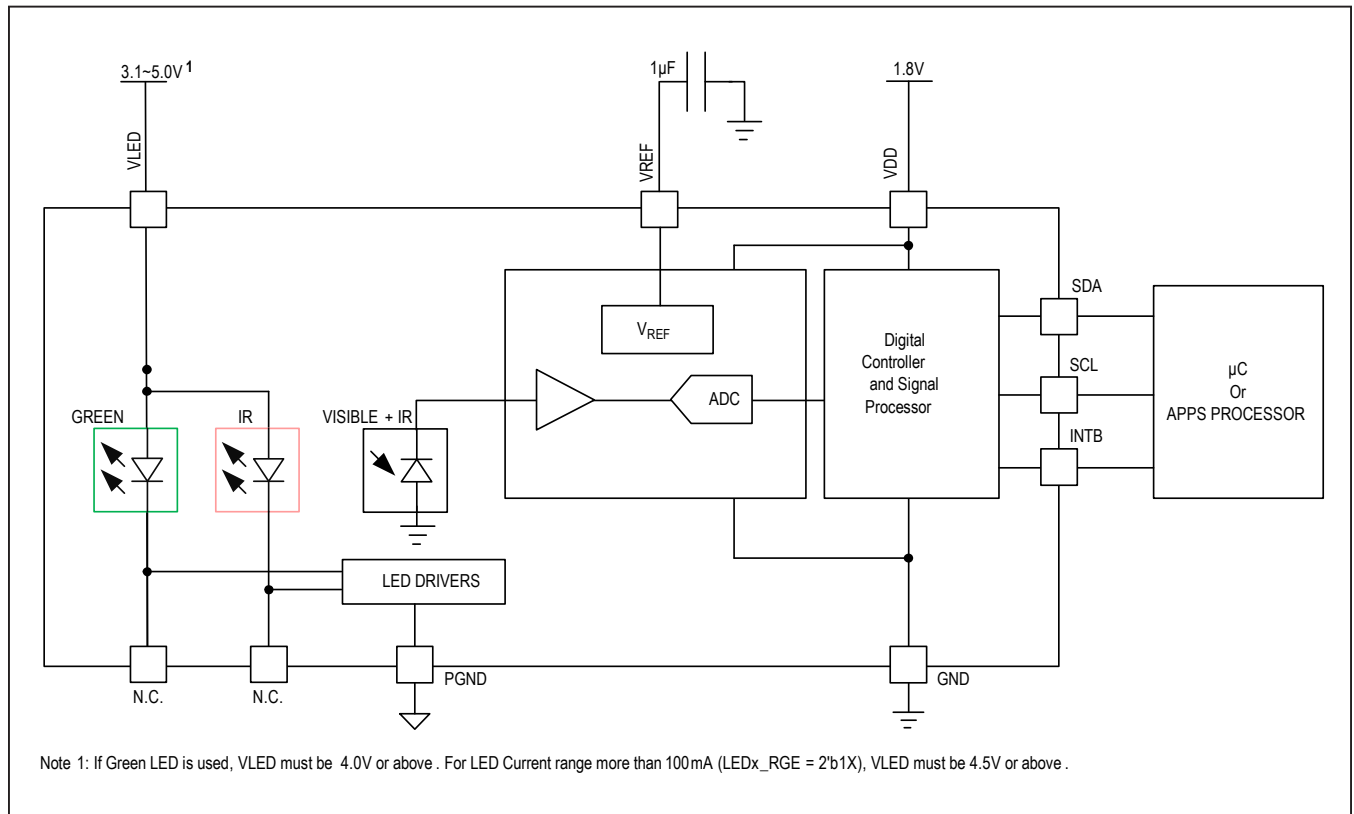


Pin Description

| PIN | NAME | FUNCTION |
|--------------------------|------------------|---|
| POWER | | |
| 11 | V _{DD} | Analog Supply. Connect to externally-regulated supply. Bypass to GND |
| 12 | V _{LED} | LED Power Supply Input. Connect to external battery supply. Bypass to PGND. |
| 13 | PGND | LED Power Return. Connect to GND. |
| 14 | GND | Analog Power Return. Connect to GND. |
| CONTROL INTERFACE | | |
| 5 | INTB | Open-Drain Interrupt |
| 6 | SDA | I ² C Data |
| 7 | SCL | I ² C Clock |
| REFERENCE | | |
| 15 | VREF | Internal Reference Decoupling Point. Bypass to GND. |
| N.C. | | |
| 1 | N.C. | No Connection. Connect to unconnected PCB pad for mechanical stability. N.C. pins should not be connected to any signal, power, or ground pins. |
| 2 | N.C. | No Connection. Connect to unconnected PCB pad for mechanical stability. N.C. pins should not be connected to any signal, power, or ground pins. |

| PIN | NAME | FUNCTION |
|----------|------|---|
| MAX86160 | | |
| 3 | N.C. | No Connection. Connect to unconnected PCB pad for mechanical stability. N.C. pins should not be connected to any signal, power, or ground pins. |
| 4 | N.C. | No Connection. Connect to unconnected PCB pad for mechanical stability. N.C. pins should not be connected to any signal, power, or ground pins. |
| 8 | N.C. | No Connection. Connect to unconnected PCB pad for mechanical stability. N.C. pins should not be connected to any signal, power, or ground pins. |
| 9 | N.C. | No Connection. Connect to unconnected PCB pad for mechanical stability. N.C. pins should not be connected to any signal, power, or ground pins. |
| 10 | N.C. | No Connection. Connect to unconnected PCB pad for mechanical stability. N.C. pins should not be connected to any signal, power, or ground pins. |
| 16 | N.C. | No Connection. Connect to unconnected PCB pad for mechanical stability. N.C. pins should not be connected to any signal, power, or ground pins. |
| 17 | N.C. | No Connection. Connect to unconnected PCB pad for mechanical stability. N.C. pins should not be connected to any signal, power, or ground pins. |
| 18 | N.C. | No Connection. Connect to unconnected PCB pad for mechanical stability. N.C. pins should not be connected to any signal, power, or ground pins. |

Functional Diagram



Detailed Description

The MAX86160 is a heart rate sensor system solution module designed for the demanding requirements of mobile and wearable devices. The MAX86160 maintains a very small total solution size without sacrificing optical or electrical performance. Minimal external hardware components are necessary for integration into a mobile device. The device is fully adjustable through software registers, and the digital output data is stored in a 32-deep FIFO within the device. The FIFO allows the device to be connected to a micro-controller or processor on a shared bus, where the data is not being read continuously from the MAX86160's registers.

HRM Subsystem

The HRM subsystem in the MAX86160 is composed of ambient light cancellation (ALC), a continuous-time sigma delta ADC, and proprietary discrete time filter. The ALC has an internal DAC to cancel ambient light and increase the effective dynamic range. The internal ADC is a continuous time oversampling sigma delta converter with 19-bit resolution. The ADC output data rate can be programmed from 10sps (samples per second) to 3200sps. The MAX86160 includes a proprietary discrete time filter to reject 50Hz/60Hz interference and slow moving residual ambient noise.

LED Driver

The MAX86160 integrates green and infrared LED drivers to modulate LED pulses for HR measurements. The LED current can be programmed from 0mA to 200mA with proper V_{LED} supply voltage. The LED pulse width can be programmed from 50 μ s to 400 μ s to allow the algorithm to optimize HR accuracy and power consumption based on use cases.

Proximity Function

The MAX86160 includes a proximity function to save power and reduce visible light emission when the user's finger is not on the sensor. Proximity function is enabled by setting PROX_INT_EN to 1. When the HR function is initiated, the IR LED is turned on in proximity mode with a drive current set by the PILOT_PA register. When an object is detected by exceeding the IR ADC count threshold (set in the PROX_INT_THRESH register), PROX_INT interrupt is asserted and the part transitions automatically to the normal HR Mode. To reenter PROX mode, a new HR reading must be initiated (even if the value is the same). The proximity function can be disabled by resetting PROX_INT_EN to 0. In that case, when the HR function is initiated in the FIFO Data Control registers, the HR mode begins immediately.

Register Map

| ADDRESS | NAME | MSB | | | | | | | LSB |
|-------------------------|-------------------------|-------------|-------------|-------------|--------------|---|---|---|----------|
| STATUS REGISTERS | | | | | | | | | |
| 0x00 | Interrupt Status 1[7:0] | A_FULL_ | PPG_RDY_ | ALC_OVF_ | PROX_INT_ | - | - | - | PWR_RDY_ |
| 0x01 | Interrupt Status 2[7:0] | VDD_OOR_ | - | - | - | - | - | - | - |
| 0x02 | Interrupt Enable 1[7:0] | A_FULL_EN_ | PPG_RDY_EN_ | ALC_OVF_EN_ | PROX_INT_EN_ | - | - | - | - |
| 0x03 | Interrupt Enable 2[7:0] | VDD_OOR_EN_ | - | - | - | - | - | - | - |

Register Map (continued)

| ADDRESS | NAME | MSB | | | | | | | LSB |
|----------------------------|-----------------------------------|-----------------------|-------------|----------------|---------------------|-------------------|------------------|-------|--------|
| FIFO REGISTERS | | | | | | | | | |
| 0x04 | FIFO Write Pointer[7:0] | - | - | - | FIFO_WR_PTR_[4:0] | | | | |
| 0x05 | Overflow Counter[7:0] | - | - | - | OVF_COUNTER_[4:0] | | | | |
| 0x06 | FIFO Read Pointer[7:0] | - | - | - | FIFO_RD_PTR_[4:0] | | | | |
| 0x07 | FIFO Data Register[7:0] | FIFO_DATA_[7:0] | | | | | | | |
| 0x08 | FIFO Configuration[7:0] | - | A_FULL_CLR_ | A_FULL_TYPE_ | FIFO_ROLLS_ON_FULL_ | FIFO_A_FULL_[3:0] | | | |
| FIFO DATA CONTROL | | | | | | | | | |
| 0x09 | FIFO Data Control Register 1[7:0] | FD2_[3:0] | | | FD1_[3:0] | | | | |
| 0x0A | FIFO Data Control Register 2[7:0] | FD4_[3:0] | | | FD3_[3:0] | | | | |
| - | | | | | | | | | |
| SYSTEM CONTROL | | | | | | | | | |
| 0x0D | System Control [7:0] | - | - | - | - | - | FIFO_EN_ | SHDN_ | RESET_ |
| PPG Configuration | | | | | | | | | |
| 0x0E | PPG Configuration 1 [7:0] | PPG_ADC_RGE_[1:0] | | PPG_SR_[3:0] | | | PPG_LED_PW_[1:0] | | |
| 0x0F | PPG Configuration 2 [7:0] | - | - | - | - | - | SMP_AVE_[2:0] | | |
| 0x10 | Prox Interrupt Threshold [7:0] | PROX_INT_THRESH_[7:0] | | | | | | | |
| LED Pulse Amplitude | | | | | | | | | |
| 0x11 | LED1 PA[7:0] | LED1_PA_[7:0] | | | | | | | |
| - | | | | | | | | | |
| 0x13 | LED3 PA[7:0] | LED3_PA_[7:0] | | | | | | | |
| 0x14 | LED Range[7:0] | - | - | LED3_RGE_[1:0] | - | - | LED1_RGE_[1:0] | | |
| 0x15 | LED PILOT PA[7:0] | PILOT_PA_[7:0] | | | | | | | |
| - | | | | | | | | | |
| Part ID | | | | | | | | | |
| 0xFF | Part ID[7:0] | PART_ID_[7:0] | | | | | | | |

Interrupt Status 1 (0x00)

| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|-------------|-----------|-----------|-----------|-----------|---|---|---|-----------|
| Field | A_FULL | PPG_RDY | ALC_OVF | PROX_INT | – | – | – | PWR_RDY |
| Reset | 0x0 | 0x0 | 0x0 | 0x0 | – | – | – | 0x0 |
| Access Type | Read Only | Read Only | Read Only | Read Only | – | – | – | Read Only |

A_FULL: FIFO Almost Full Flag

| VALUE | ENUMERATION | DECODE |
|-------|-------------|--|
| 0 | OFF | Normal Operation |
| 1 | ON | Indicates that the FIFO buffer will overflow the threshold set by FIFO_A_FULL<3:0> on the next sample. This bit is cleared when the Interrupt Status 1 Register is read. It is also cleared when FIFO_DATA register is read, if A_FULL_CLR = 1 |

PPG_RDY: New PPG FIFO Data Ready

| VALUE | ENUMERATION | DECODE |
|-------|-------------|---|
| 0 | OFF | Normal Operation |
| 1 | ON | In HR modes, this interrupt triggers when there is a new sample in the data FIFO. The interrupt is cleared by reading the Interrupt Status 1 register (0x00), or by reading the FIFO_DATA register. |

ALC_OVF: Ambient Light Cancellation Overflow

| VALUE | ENUMERATION | DECODE |
|-------|-------------|---|
| 0 | OFF | Normal Operation |
| 1 | ON | This interrupt triggers when the ambient light cancellation function of the HR photodiode has reached its maximum limit due to overflow, and therefore, ambient light is affecting the output of the ADC. The interrupt is cleared by reading the Interrupt Status 1 register (0x00). |

PROX_INT: Proximity interrupt

If PROX_INT is masked then the prox mode is disabled and the select PPG begins immediately. This bit is cleared when the Interrupt Status 1 Register is read.

| VALUE | ENUMERATION | DECODE |
|-------|-------------|---|
| 0 | OFF | Normal Operation |
| 1 | ON | Indicates that the proximity threshold has been crossed when in proximity mode. |

PWR_RDY: Power Ready Flag

| VALUE | ENUMERATION | DECODE |
|-------|-------------|--|
| 0 | OFF | Normal Operation |
| 1 | ON | Indicates that VBATT went below the UVLO threshold. This bit is not triggered by a soft reset. This bit is cleared when Interrupt Status 1 Register is read. |

Interrupt Status 2 (0x01)

| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|-------------|-----------|---|---|---|---|---|---|---|
| Field | VDD_OOR | – | – | – | – | – | – | – |
| Reset | 0x0 | – | – | – | – | – | – | – |
| Access Type | Read Only | – | – | – | – | – | – | – |

VDD_OOR: VDD Out-of-Range flag

This flag checks if the VDD_ANA supply voltage is outside supported range.

| VALUE | ENUMERATION | DECODE |
|-------|-------------|--|
| 0 | OFF | VDD_ANA between range. |
| 1 | ON | Indicated that VDD_ANA is greater than 2.05V or less than 1.65V. This bit is automatically cleared when the Interrupt Status 2 register is read. The detection circuitry has a 10ms delay time, and will continue to trigger as long as the VDD_ANA is out of range. |

Interrupt Enable 1 (0x02)

| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|-------------|-------------|-------------|-------------|-------------|---|---|---|---|
| Field | A_FULL_EN | PPG_RDY_EN | ALC_OVF_EN | PROX_INT_EN | – | – | – | – |
| Reset | 0x0 | 0x0 | 0x0 | 0x0 | – | – | – | – |
| Access Type | Write, Read | Write, Read | Write, Read | Write, Read | – | – | – | – |

A_FULL_EN: FIFO Almost Full Flag enable

| VALUE | ENUMERATION | DECODE |
|-------|-------------|------------------------------|
| 0 | OFF | A_FULL interrupt is disabled |
| 1 | ON | A_FULL interrupt is enabled |

PPG_RDY_EN: New PPG FIFO Data Ready Interrupt enable

| VALUE | ENUMERATION | DECODE |
|-------|-------------|-------------------------------|
| 0 | OFF | PPG_RDY interrupt is disabled |
| 1 | ON | PPG_RDY interrupt is enabled. |

ALC_OVF_EN: Ambient Light Cancellation (ALC) Overflow Interrupt enable

The ALC_OVF flag will be triggered when the HRM photodiode has reached it's maximum limit due to overflow. At this point, the ADC output will be affected by the ambient light.

| VALUE | ENUMERATION | DECODE |
|-------|-------------|-------------------------------|
| 0 | OFF | ALC_OVF interrupt is disabled |
| 1 | ON | ALC_OVF interrupt is enabled |

PROX_INT_EN: Proximity Interrupt enable

When the HR function is initiated, the IR LED is turned on in proximity mode with a drive current set by the PILOT_PA register. When an object is detected by exceeding the IR ADC count threshold (set in the PROX_INT_THRESH register), PROX_INT interrupt is asserted and the part transitions automatically to the normal HR mode.

| VALUE | ENUMERATION | DECODE |
|-------|-------------|--------------------------------|
| 0 | OFF | PROX_INT interrupt is disabled |
| 1 | ON | PROX_INT interrupt is enabled |

Interrupt Enable 2 (0x03)

| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|-------------|-------------|---|---|---|---|---|---|---|
| Field | VDD_OOR_EN | – | – | – | – | – | – | – |
| Reset | 0x0 | – | – | – | – | – | – | – |
| Access Type | Write, Read | – | – | – | – | – | – | – |

VDD_OOR_EN: VDD Out-of-Range Indicator enable

| VALUE | ENUMERATION | DECODE |
|-------|-------------|--------------------------------|
| 0 | OFF | Disables the VDD_OVR interrupt |
| 1 | ON | Enables the VDD_OVR interrupt |

FIFO Write Pointer (0x04)

| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|-------------|---|---|---|------------------|---|---|---|---|
| Field | – | – | – | FIFO_WR_PTR[4:0] | | | | |
| Reset | – | – | – | 0x00 | | | | |
| Access Type | – | – | – | Write, Read | | | | |

FIFO_WR_PTR: FIFO Write Pointer

This points to the location where the next sample will be written. This pointer advances for each sample pushed on to the FIFO.

| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|-------------|---|---|---|------------------|---|---|---|---|
| Field | – | – | – | OVF_COUNTER[4:0] | | | | |
| Reset | – | – | – | 0x00 | | | | |
| Access Type | – | – | – | Read Only | | | | |

OVF_COUNTER: FIFO Overflow Counter

When FIFO is full any new samples will result in new or old samples getting lost depending on FIFO_ROLLS_ON_FULL. OVF_COUNTER counts the number of samples lost. It saturates at 0x1F.

FIFO Read Pointer (0x06)

| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|-------------|---|---|---|------------------|---|---|---|---|
| Field | – | – | – | FIFO_RD_PTR[4:0] | | | | |
| Reset | – | – | – | 0x00 | | | | |
| Access Type | – | – | – | Write, Read | | | | |

FIFO_RD_PTR: FIFO Read Pointer

The FIFO Read Pointer points to the location from where the processor gets the next sample from the FIFO through the I²C interface. This advances each time a sample is popped from the FIFO. The processor may also write to this pointer after reading the samples. This allows rereading (or retrying) samples from the FIFO.

FIFO Data Register (0x07)

| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|-------------|----------------|---|---|---|---|---|---|---|
| Field | FIFO_DATA[7:0] | | | | | | | |
| Reset | 0x00 | | | | | | | |
| Access Type | Write, Read | | | | | | | |

FIFO_DATA: FIFO Data Register

This is a read-only register and is used to get data from the FIFO. See [FIFO Description](#) for more details.

FIFO Configuration (0x08)

| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|-------------|---|-------------|-------------|--------------------|------------------|---|---|---|
| Field | – | A_FULL_CLR | A_FULL_TYPE | FIFO_ROLLS_ON_FULL | FIFO_A_FULL[3:0] | | | |
| Reset | – | 0x0 | 0x0 | 0x0 | 0xF | | | |
| Access Type | – | Write, Read | Write, Read | Write, Read | Write, Read | | | |

A_FULL_CLR: FIFO Almost Full Interrupt Options

This defines whether the A-FULL interrupt should get cleared by FIFO_DATA register read.

| VALUE | ENUMERATION | DECODE |
|-------|---------------|--|
| 0 | RD_DATA_NOCLR | A_FULL interrupt does not get cleared by FIFO_DATA register read. It gets cleared by status register read. |
| 1 | RD_DATA_CLR | A_FULL interrupt gets cleared by FIFO_DATA register read or status register read. |

A_FULL_TYPE: FIFO Almost Full Flag Options

This defines the behavior of the A_FULL interrupt.

| VALUE | ENUMERATION | DECODE |
|-------|-------------|--|
| 0 | AFULL_RPT | A_FULL interrupt gets asserted when the a_full condition is detected. It is cleared by status register read, but re-asserts for every sample if the a_full condition persists. |
| 1 | AFULL_ONCE | A_FULL interrupt gets asserted only when the a_full condition is detected. The interrupt gets cleared on status register read, and does not re-assert for every sample until a new a-full condition is detected. |

FIFO_ROLLS_ON_FULL: FIFO Rolls on Full Options

This bit controls the behavior of the FIFO when the FIFO becomes completely filled with data.

- When the device is in PROX mode, the FIFO always rolls on full.
- Push to FIFO is enabled when FIFO is full if FIFO_ROLLS_ON_FULL = 1 and old samples are lost. Both FIFO_WR_PTR and FIFO_RD_PTR increment for each sample after the FIFO is full.
- Push to FIFO is disabled when FIFO is full if FIFO_ROLLS_ON_FULL = 0 and new samples are lost. FIFO_WR_PTR does not increment for each sample after the FIFO is full.

| VALUE | ENUMERATION | DECODE |
|-------|-------------|--|
| 0 | OFF | The FIFO stops on full. |
| 1 | ON | The FIFO automatically rolls over on full. |

FIFO_A_FULL: FIFO Almost Full Value

These bits indicate how many new samples can be written to the FIFO before the interrupt is asserted. For example, if set to 0xF, the interrupt triggers when there is 17 empty space left (15 data samples), and so on.

| FIFO_A_FULL<3:0> | FREE SPACE BEFORE INTERRUPT | # OF SAMPLES IN FIFO |
|------------------|-----------------------------|----------------------|
| 0000 | 0 | 32 |
| 0001 | 1 | 31 |
| 0010 | 2 | 30 |
| 0011 | 3 | 29 |
| ---- | ---- | ---- |
| 1110 | 14 | 18 |
| 1111 | 15 | 17 |

FIFO Data Control Register 1 (0x09)

| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|-------------|-------------|---|---|---|-------------|---|---|---|
| Field | FD2[3:0] | | | | FD1[3:0] | | | |
| Reset | 0x0 | | | | 0x0 | | | |
| Access Type | Write, Read | | | | Write, Read | | | |

FD2: FIFO Data Time Slot 2

These bits set the data type for element 2 of the FIFO.

The FIFO can hold up to 32 samples. Each sample can hold up to four elements and each element is 3 bytes wide. The data type that gets stored in the 3 bytes is configured by FD1, FD2, FD3 and FD4 according to the table below. For restriction on data type sequences please see the FLEX FIFO document.

| FD2<3:0> | DATA TYPE | FD2<3:0> | DATA TYPE | FD2<3:0> | DATA TYPE | FD2<3:0> | DATA TYPE |
|----------|-----------|----------|------------|----------|-----------|----------|-----------|
| 0000 | Reserved | 0100 | Reserved | 1000 | Reserved | 1100 | Reserved |
| 0001 | PPG_LED1 | 0101 | Pilot LED1 | 1001 | Reserved | 1101 | Reserved |
| 0010 | Reserved | 0110 | Reserved | 1010 | Reserved | 1110 | Reserved |
| 0011 | PPG_LED3 | 0111 | Pilot LED3 | 1011 | Reserved | 1111 | Reserved |

FD1: FIFO Data Time Slot 1

These bits set the data type for element 1 of the FIFO.

The FIFO can hold up to 32 samples. Each sample can hold up to four elements and each element is 3 bytes wide. The data type that gets stored in the 3 bytes is configured by FD1, FD2, FD3, and FD4 according to the table below. For restriction on data type sequences please refer to the [FIFO Description](#) section.

| FD1<3:0> | DATA TYPE | FD1<3:0> | DATA TYPE | FD1<3:0> | DATA TYPE | FD1<3:0> | DATA TYPE |
|----------|-----------|----------|-------------|----------|-----------|----------|-----------|
| 0000 | Reserved | 0100 | Reserved | 1000 | Reserved | 1100 | Reserved |
| 0001 | PPG_LED1 | 0101 | Pilot LED 1 | 1001 | Reserved | 1101 | Reserved |
| 0010 | Reserved | 0110 | Reserved | 1010 | Reserved | 1110 | Reserved |
| 0011 | PPG_LED3 | 0111 | Pilot LED3 | 1011 | Reserved | 1111 | Reserved |

FIFO Data Control Register 2 (0x0A)

| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|-------------|-------------|---|---|---|-------------|---|---|---|
| Field | FD4[3:0] | | | | FD3[3:0] | | | |
| Reset | 0x0 | | | | 0x0 | | | |
| Access Type | Write, Read | | | | Write, Read | | | |

FD4: FIFO Data Time Slot 4

These bits set the data type for element 4 of the FIFO.

The FIFO can hold up to 32 samples. Each sample can hold up to four elements and each element is 3 bytes wide. The data type that gets stored in the 3 bytes is configured by FD1, FD2, FD3, and FD4 according to the table below. For restriction on data type sequences please see the FLEX FIFO document.

| FD4<3:0> | DATA TYPE | FD4<3:0> | DATA TYPE | FD4<3:0> | DATA TYPE | FD4<3:0> | DATA TYPE |
|----------|-----------|----------|------------|----------|-----------|----------|-----------|
| 0000 | Reserved | 0100 | Reserved | 1000 | Reserved | 1100 | Reserved |
| 0001 | PPG_LED1 | 0101 | Pilot LED1 | 1001 | Reserved | 1101 | Reserved |
| 0010 | Reserved | 0110 | Reserved | 1010 | Reserved | 1110 | Reserved |
| 0011 | PPG_LED3 | 0111 | Pilot LED3 | 1011 | Reserved | 1111 | Reserved |

FD3: FIFO Data Time Slot 3

These bits set the data type for element 3 of the FIFO.

The FIFO can hold up to 32 samples. Each sample can hold up to four elements and each element is 3 bytes wide. The data type that gets stored in the 3 bytes is configured by FD1, FD2, FD3, and FD4 according to the table below. For restriction on data type sequences please refer to the [FIFO Description](#) section.

| FD3<3:0> | DATA TYPE | FD3<3:0> | DATA TYPE | FD3<3:0> | DATA TYPE | FD3<3:0> | DATA TYPE |
|----------|-----------|----------|------------|----------|-----------|----------|-----------|
| 0000 | Reserved | 0100 | Reserved | 1000 | Reserved | 1100 | Reserved |
| 0001 | PPG_LED1 | 0101 | Pilot LED1 | 1001 | Reserved | 1101 | Reserved |
| 0010 | Reserved | 0110 | Reserved | 1010 | Reserved | 1110 | Reserved |
| 0011 | PPG_LED3 | 0111 | Pilot LED3 | 1011 | Reserved | 1111 | Reserved |

System Control (0x0D)

| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|-------------|---|---|---|---|---|-------------|-------------|-------------|
| Field | – | – | – | – | – | FIFO_EN | SHDN | RESET |
| Reset | – | – | – | – | – | 0x0 | 0x0 | 0x0 |
| Access Type | – | – | – | – | – | Write, Read | Write, Read | Write, Read |

FIFO_EN: FIFO Enable

| VALUE | ENUMERATION | DECODE |
|-------|-------------|---|
| 0 | OFF | Push to FIFO is disabled, but the read and write pointers and the data in the FIFO are all held at their values before FIFO_EN is set to 0. |
| 1 | ON | The FIFO is enabled. When this bit is set the FIFO is flushed of all old data and the new samples start loading from pointer zero. |

SHDN: Shutdown Control

The part can be put into a power-save mode by setting this bit to one. While in power-save mode, all registers retain their values, and write/read operations function as normal. All interrupts are cleared to zero in this mode.

| VALUE | ENUMERATION | DECODE |
|-------|-------------|---|
| 0 | OFF | The part is in normal operation. No action taken. |
| 1 | ON | The part can be put into a power-save mode by writing a '1' to this bit. While in this mode all registers remain accessible and retain their data. ADC conversion data contained in the registers are previous values. Writeable registers also remain accessible in shutdown. All interrupts are cleared. In this mode the oscillator is shutdown and the part draws minimum current. If this bit is asserted during a active conversion then the conversion completes before the part shuts down. |

RESET: Reset Control

When this bit is set, the part initiates a forced power-on-reset sequence. All configuration, threshold and data registers including distributed registers are reset to their power-on-state. This bit then automatically becomes '0' after the reset sequence is completed.

| VALUE | ENUMERATION | DECODE |
|-------|-------------|--|
| 0 | OFF | The part is in normal operation. No action taken. |
| 1 | ON | The part initiates a forced power-on-reset sequence. All configuration, threshold and data registers including distributed registers are reset to their power-on-state. This bit then automatically becomes '0' after the reset sequence is completed. |

PPG Configuration 1 (0x0E)

| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|-------------|------------------|---|-------------|---|---|-----------------|---|---|
| Field | PPG_ADC_RGE[1:0] | | PPG_SR[3:0] | | | PPG_LED_PW[1:0] | | |
| Reset | 0x0 | | 0x0 | | | 0x0 | | |
| Access Type | Write, Read | | Write, Read | | | Write, Read | | |

PPG_ADC_RGE: PPG ADC Range Control

These bits set the ADC range of the PPG sensor as shown in the table below.

| PPG_ADC_RGE<1:0> | LSB [PA] | FULL SCALE [NA] |
|------------------|----------|-----------------|
| 00 | 7.8125 | 4096 |
| 01 | 15.625 | 8192 |
| 10 | 31.25 | 16384 |
| 11 | 62.5 | 32768 |

PPG_SR: PPG Sample Rate Control**PPG Sample Rate Control**

These bits set the effective sampling rate of the PPG sensor as shown in the table below.

Note: If a sample rate is set can not be supported by the selected pulse width and LED mode then the highest available sample rate will be automatically set. The user can read back this register to confirm the sample rate.

| PPG_SR<3:0> | SAMPLES PER SECOND | PULSES PER SAMPLE, N |
|-------------|--------------------|----------------------|
| 0000 | 10 | 1 |
| 0001 | 20 | 1 |
| 0010 | 50 | 1 |
| 0011 | 84 | 1 |
| 0100 | 100 | 1 |
| 0101 | 200 | 1 |
| 0110 | 400 | 1 |
| 0111 | 800 | 1 |

| PPG_SR<3:0> | SAMPLES PER SECOND | PULSES PER SAMPLE, N |
|-------------|--------------------|----------------------|
| 1000 | 1000 | 1 |
| 1001 | 1600 | 1 |
| 1010 | 3200 | 1 |
| 1011 | 10 | 2 |
| 1100 | 20 | 2 |
| 1101 | 50 | 2 |
| 1110 | 84 | 2 |
| 1111 | 100 | 2 |

Maximum Sample Rates Supported for all the Pulse Widths and Number of LEDs:

| NUMBER OF ADC CONVERSIONS PER SAMPLE | PPG_LED_PW = 0 (50US) | PPG_LED_PW = 1 (100US) | PPG_LED_PW = 2 (200US) | PPG_LED_PW = 3 (400US) |
|--------------------------------------|-----------------------|------------------------|------------------------|------------------------|
| 1 LED, N = 1 | 3200 | 1600 | 1000 | 1000 |
| 2 LED, N = 1 | 1600 | 800 | 800 | 400 |
| 1 LED, N = 2 | 100 | 100 | 100 | 100 |
| 2 LED, N = 2 | 100 | 100 | 100 | 84 |

PPG_LED_PW: LED Pulse Width Control

These bits set the pulse width of the LED drivers and the integration time of PPG ADC as shown in the table below.

| PPG_LED_PW<1:0> | PULSE WIDTH [US] | INTEGRATION TIME [US] | RES BITS |
|-----------------|------------------|-----------------------|----------|
| 00 | 50 | 50 | 19 |
| 01 | 100 | 100 | 19 |
| 10 | 200 | 200 | 19 |
| 11 | 400 | 400 | 19 |

PPG Configuration 2 (0x0F)

| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|-------------|---|---|---|---|---|--------------|---|---|
| Field | - | - | - | - | - | SMP_AVE[2:0] | | |
| Reset | - | - | - | - | - | 0x0 | | |
| Access Type | - | - | - | - | - | Write, Read | | |

SMP_AVE: Sample Averaging Options

To reduce the amount of data throughput, adjacent samples (in each individual channel) can be averaged and decimated on the chip by setting this register.

These bits set the number of samples that are averaged on chip before being written to the FIFO.

| SMP_AVE[2:0] | SAMPLE AVERAGE |
|--------------|------------------|
| 000 | 1 (No Averaging) |
| 001 | 2 |
| 010 | 4 |
| 011 | 8 |
| 100 | 16 |
| 101 | 32 |
| 110 | 32 |
| 111 | 32 |

Prox Interrupt Threshold (0x10)

| | | | | | | | | |
|-------------|----------------------|----------|----------|----------|----------|----------|----------|----------|
| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Field | PROX_INT_THRESH[7:0] | | | | | | | |
| Reset | 0x00 | | | | | | | |
| Access Type | Write, Read | | | | | | | |

PROX_INT_THRESH: Proximity Mode Interrupt Threshold

This register sets the IR ADC count that will trigger the beginning of HR mode. The threshold is defined as the 8 MSB bits of the ADC count. For example, if PROX_INT_THRESH[7:0] = 0x01, then an ADC value of 1023 (decimal) or higher triggers the PROX interrupt. If PROX_INT_THRESH[7:0] = 0xFF, then only a saturated ADC triggers the interrupt.

LED1 PA (0x11)

| | | | | | | | | |
|-------------|--------------|----------|----------|----------|----------|----------|----------|----------|
| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Field | LED1_PA[7:0] | | | | | | | |
| Reset | 0x00 | | | | | | | |
| Access Type | Write, Read | | | | | | | |

LED1_PA: LED 1 (IR) Current Pulse Amplitude.

These bits set the nominal current pulse amplitude of LED 1, as shown in the table below.

| LED1_RGE<1:0> | 00 (50MA) | 01 (100MA) | 10 (150MA) | 11 (200MA) |
|----------------------------|------------------|-------------------|-------------------|-------------------|
| LED1_PA<7:0> | LED Current[mA] | LED Current[mA] | LED Current[mA] | LED Current[mA] |
| 00000000 | 0 | 0 | 0 | 0 |
| 00000001 | 0.2 | 0.4 | 0.6 | 0.8 |
| 00000010 | 0.4 | 0.8 | 1.2 | 1.6 |
| 00000011 | 0.6 | 1.2 | 1.8 | 2.4 |
| | | | | |
| 11111100 | 50.4 | 100.8 | 151.2 | 201.6 |
| 11111101 | 50.6 | 101.2 | 151.8 | 202.4 |
| 11111110 | 50.8 | 101.6 | 152.4 | 203.2 |
| 11111111 | 51 | 102 | 153 | 204 |
| LSB | 0.2 | 0.4 | 0.6 | 0.8 |

Note: For LED Current more than 100mA, VLED must be 4.5V or above.

LED3 PA (0x13)

| | | | | | | | | |
|--------------------|--------------|----------|----------|----------|----------|----------|----------|----------|
| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Field | LED3_PA[7:0] | | | | | | | |
| Reset | 0x00 | | | | | | | |
| Access Type | Write, Read | | | | | | | |

LED3_PA: LED 3 (Green) Current Pulse Amplitude

These bits set the nominal current pulse amplitude of LED 3, as shown in the table below.

| LED3_RGE<1:0> | 00 (50MA) | 01 (100MA) | 10 (150MA) | 11 (200MA) |
|---------------|-----------------|-----------------|-----------------|-----------------|
| LED3_PA<7:0> | LED Current[mA] | LED Current[mA] | LED Current[mA] | LED Current[mA] |
| 00000000 | 0 | 0 | 0 | 0 |
| 00000001 | 0.2 | 0.4 | 0.6 | 0.8 |
| 00000010 | 0.4 | 0.8 | 1.2 | 1.6 |
| 00000011 | 0.6 | 1.2 | 1.8 | 2.4 |
| | | | | |
| 11111100 | 50.4 | 100.8 | 151.2 | 201.6 |
| 11111101 | 50.6 | 101.2 | 151.8 | 202.4 |
| 11111110 | 50.8 | 101.6 | 152.4 | 203.2 |
| 11111111 | 51 | 102 | 153 | 204 |
| LSB | 0.2 | 0.4 | 0.6 | 0.8 |

Note: For LED Current more than 100mA, VLED must be 4.5V or above.

LED Range (0x14)

| | | | | | | | | |
|--------------------|----------|----------|---------------|----------|----------|----------|---------------|----------|
| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Field | – | – | LED3_RGE[1:0] | | – | – | LED1_RGE[1:0] | |
| Reset | – | – | 0x00 | | – | – | 0x00 | |
| Access Type | – | – | Write, Read | | – | – | Write, Read | |

LED3_PA: LED 3 (Green) Current Pulse Amplitude

These bits set the nominal current pulse amplitude of LED 3, as shown in the table below.

| LED3_RGE<1:0> | LED CURRENT[MA] |
|---------------|-----------------|
| 00 | 50 |
| 01 | 100 |
| 10 | 150 |
| 11 | 200 |

LED1_RGE: LED 1 (IR) Current Control

Range selection of the LED current. Please refer to LED1_PA[7:0] for more details. For LED Current more than 100mA, VLED must be 4.5V or above.

| LED1_RGE<1:0> | LED CURRENT[MA] |
|---------------|-----------------|
| 00 | 50 |
| 01 | 100 |
| 10 | 150 |
| 11 | 200 |

LED PILOT PA (0x15)

| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|-------------|---------------|---|---|---|---|---|---|---|
| Field | PILOT_PA[7:0] | | | | | | | |
| Reset | 0x00 | | | | | | | |
| Access Type | Write, Read | | | | | | | |

PILOT_PA: Proximity Mode LED Pulse Amplitude.

The purpose of PILOT_PA<7:0> is to set the LED power during the PROX mode, as well as in Multi-LED mode. These bits set the current pulse amplitude for the pilot mode as shown in the table below.

When LED x is used, the respective LEDx_RGE<1:0> is used to control the range of the LED driver in conjunction with PILOT_PA<7:0>. For instance, if LED1 is used in the PILOT mode, then, LED1_RGE<1:0> together with PILOT_PA<7:0> will be used to set the LED1 current.

| LEDX_RGE<1:0> | 00 (50MA) | 01 (100MA) | 10 (150MA) | 11 (200MA) |
|---------------|-----------------|-----------------|-----------------|-----------------|
| PILOT_PA<7:0> | LED Current[mA] | LED Current[mA] | LED Current[mA] | LED Current[mA] |
| 00000000 | 0 | 0 | 0 | 0 |
| 00000001 | 0.2 | 0.4 | 0.6 | 0.8 |
| 00000010 | 0.4 | 0.8 | 1.2 | 1.6 |
| 00000011 | 0.6 | 1.2 | 1.8 | 2.4 |
| | | | | |
| 11111100 | 50.4 | 100.8 | 151.2 | 201.6 |
| 11111101 | 50.6 | 101.2 | 151.8 | 202.4 |
| 11111110 | 50.8 | 101.6 | 152.4 | 203.2 |
| 11111111 | 51 | 102 | 153 | 204 |
| LSB | 0.2 | 0.4 | 0.6 | 0.8 |

Note: For LED Current more than 100mA, VLED must be 4.5V or above.

Part ID (0xFF)

| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|-------------|--------------|---|---|---|---|---|---|---|
| Field | PART_ID[7:0] | | | | | | | |
| Reset | 0x1E | | | | | | | |
| Access Type | Read Only | | | | | | | |

PART_ID: Part Identifier

This register stores the Part identifier for the chip.

Applications Information

Power Sequencing and Requirements

Power-Up Sequencing

It is recommended to power the VDD_ANA supply first, then VDD_DIG before the LED power supplies (V_LED). VDD_ANA and VDD_DIG can be powered on at the same time. The interrupt and I²C pins can be pulled up to an external voltage even when the power supplies are not powered up.

After the power is established, an interrupt occurs to alert the system that the sensor is ready for operation. Reading the I²C interrupt register clears the interrupt, as shown in [Figure 2](#).

Power-Down Sequencing

The sensor is designed to be tolerant of any power supply sequencing on power-down.

I²C Interface

The MAX86160 features an I²C/SMBus-compatible, 2-wire serial interface consisting of a serial data line (SDA) and a serial clock line (SCL). SDA and SCL facilitate communication between the MAX86160 and the master at clock rates up to 400kHz. The master generates SCL and initiates data transfer on the bus. The master device writes data to the MAX86160 by transmitting the proper slave address followed by data. Each transmit sequence

is framed by a START (S) or REPEATED START (Sr) condition and a STOP (P) condition. Each word transmitted to the MAX86160 is 8 bits long and is followed by an acknowledge clock pulse. A master reading data from the MAX86160 transmits the proper slave address followed by a series of nine SCL pulses.

The MAX86160 transmits data on SDA in sync with the master-generated SCL pulses. The master acknowledges receipt of each byte of data. Each read sequence is framed by a START (S) or REPEATED START (Sr) condition, a not acknowledge, and a STOP (P) condition. SDA operates as both an input and an open-drain output. A pullup resistor, typically greater than 1000Ω, is required on SDA. SCL operates only as an input. A pullup resistor, typically greater than 1000Ω, is required on SCL if there are multiple masters on the bus, or if the single master has an open-drain SCL output. Series resistors in line with SDA and SCL are optional. Series resistors protect the digital inputs of the MAX86160 from high voltage spikes on the bus lines and minimize crosstalk and undershoot of the bus signals.

Bit Transfer

One data bit is transferred during each SCL cycle. The data on SDA must remain stable during the high period of the SCL pulse. Changes in SDA while SCL is high are control signals. See the START and STOP Conditions section.

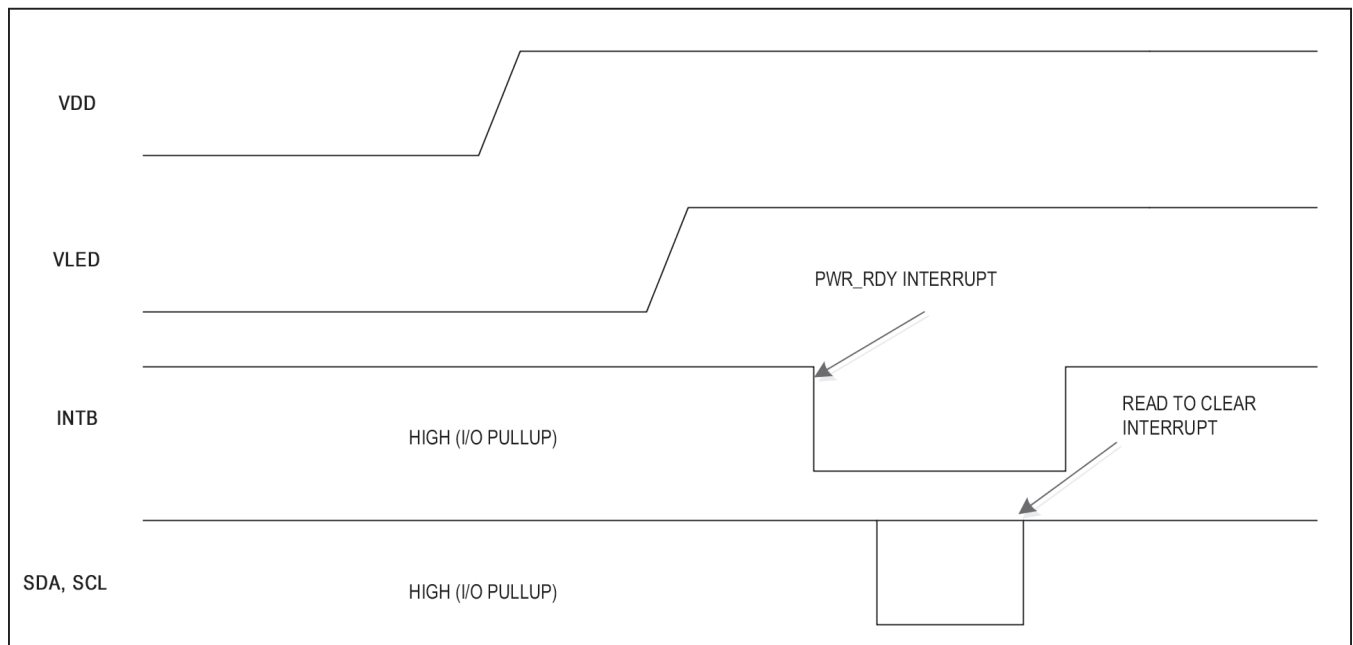


Figure 2: Power-Up Sequence of the Power Supply Rails

START and STOP Conditions

SDA and SCL idle high when the bus is not in use. A master initiates communication by issuing a START condition. A START condition is a high-to-low transition on SDA with SCL high. A STOP condition is a low-to-high transition on SDA while SCL is high. A START condition from the master signals the beginning of a transmission to the MAX86160. The master terminates transmission, and frees the bus, by issuing a STOP condition. The bus remains active if a REPEATED START condition is generated instead of a STOP condition.

Early STOP Conditions

The MAX86160 recognizes a STOP condition at any point during data transmission, except when the STOP condition occurs in the same high pulse as a START condition. For proper operation, do not send a STOP condition during the same SCL high pulse as the START condition.

Slave Address

A bus master initiates communication with a slave device by issuing a START condition followed by the 7-bit slave ID. When idle, the MAX86160 waits for a START condition followed by its slave ID. The serial interface compares each slave ID bit by bit, allowing the interface to power down and disconnect from SCL immediately if an incorrect slave ID is detected. After recognizing a START condition followed by the correct slave ID, the MAX86160 is programmed to accept or send data. The LSB of the slave ID word is the read/write (R/W) bit. R/W indicates whether the master is writing to or reading data from the MAX86160 (R/W = 0 selects a write condition, R/W = 1 selects a read condition). After receiving the proper slave ID, the MAX86160 issues an ACK by pulling SDA low for one clock cycle.

The MAX86160 slave ID consists of seven fixed bits, B7–B1 (set to 0b1011110). The most significant slave ID bit (B7) is transmitted first, followed by the remaining bits.

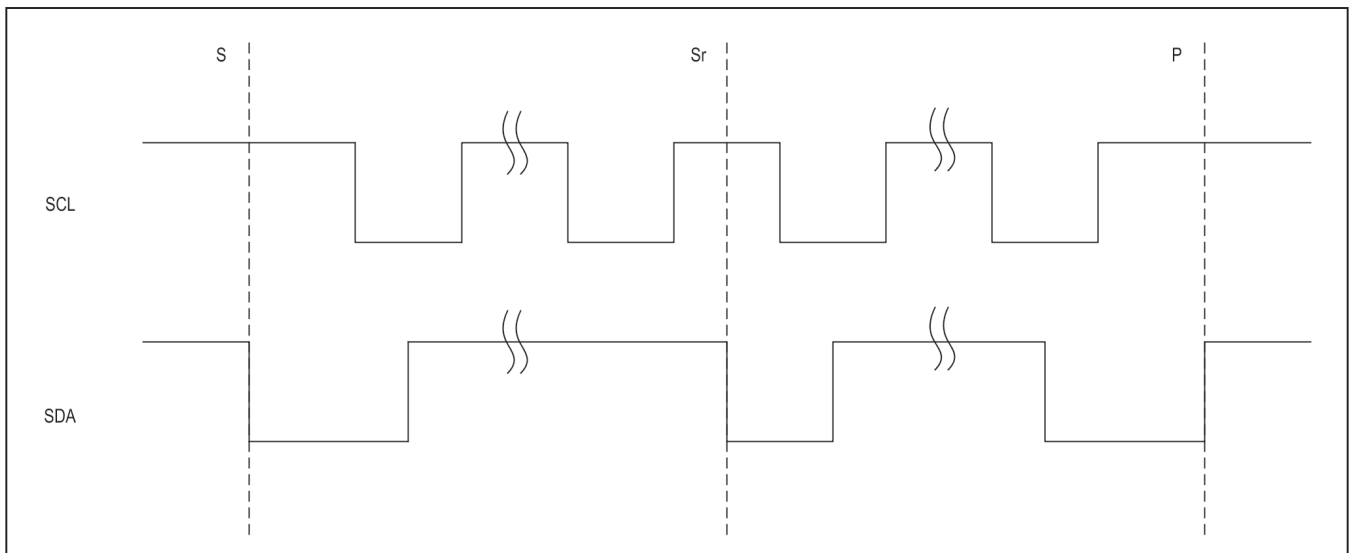


Figure 3. START, STOP, and REPEATED START Conditions

Table 1: MAX86160 I²C Slave Address

| B7 | B6 | B5 | B4 | B3 | B2 | B1 | B0 | WRITE ADDRESS | READ ADDRESS |
|----|----|----|----|----|----|----|-----|---------------|--------------|
| 1 | 0 | 1 | 1 | 1 | 1 | 0 | R/W | 0xBC | 0xBD |

Acknowledge

The acknowledge bit (ACK) is a clocked 9th bit that the MAX86160 uses to handshake receipt each byte of data when in write mode. The MAX86160 pulls down SDA during the entire master-generated 9th clock pulse if the previous byte is successfully received. Monitoring ACK allows for detection of unsuccessful data transfers. An unsuccessful data transfer occurs if a receiving device is busy or if a system fault has occurred. In the event of an unsuccessful data transfer, the bus master retries communication. The master pulls down SDA during the 9th clock cycle to acknowledge receipt of data when the MAX86160 is in read mode. An acknowledge is sent by

the master after each read byte to allow data transfer to continue. A not-acknowledge is sent when the master reads the final byte of data from the MAX86160, followed by a STOP condition.

Write Data Format

For the write operation, send the slave ID as the first byte followed by the register address byte and then one or more data bytes. The register address pointer increments automatically after each byte of data received, so for example the entire register bank can be written by at one time. Terminate the data transfer with a STOP condition. The write operation is shown in the following figure.

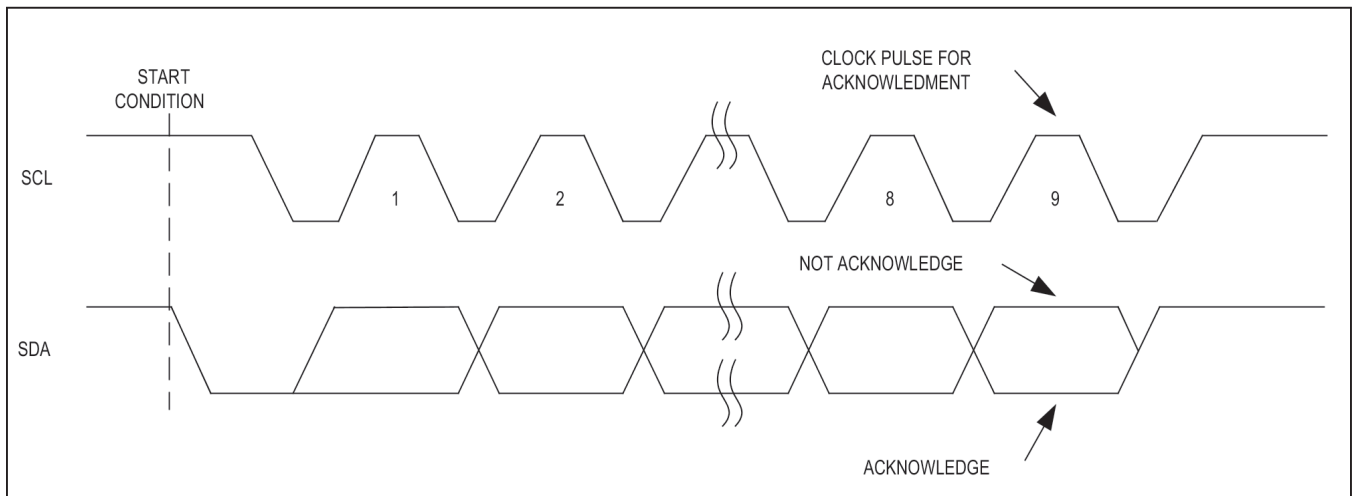


Figure 4. I²C Acknowledge

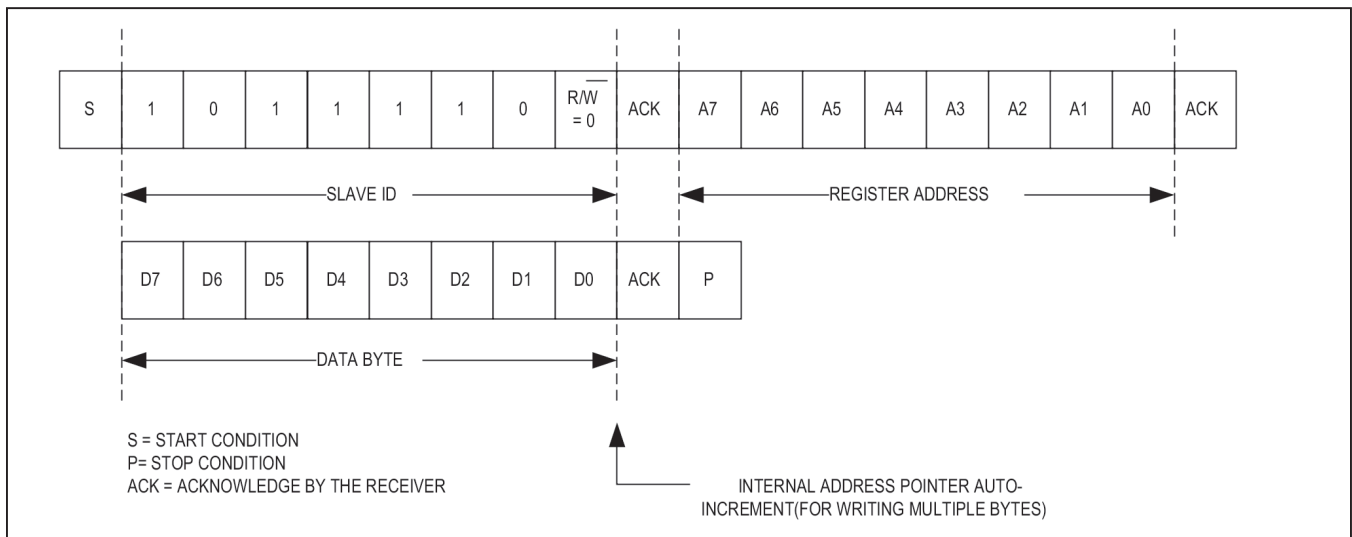


Figure 5. Writing One Data Byte to MAX86160

Read Data Format

For the read operation, two I²C operations must be performed. First, the slave ID byte is sent followed by the I²C register that you wish to read. Then a REPEAT START (Sr) condition is sent, followed by the read slave ID. The MAX86160 then begins sending data beginning with the register selected in the first operation. The read pointer increments automatically, so the MAX86160 continues sending data from additional registers in sequential order until a STOP (P) condition is received. The exception to this is the FIFO_DATA register, at which the read pointer no longer increments when reading additional bytes. To

read the next register after FIFO_DATA, an I²C write command is necessary to change the location of the read pointer.

Figure below show the process of reading one byte or multiple bytes of data.

An initial write operation is required to send the read register address.

Data is sent from registers in sequential order, starting from the register selected in the initial I²C write operation. If the FIFO_DATA register is read, the read pointer will not automatically increment, and subsequent bytes of data contain the contents of the FIFO.

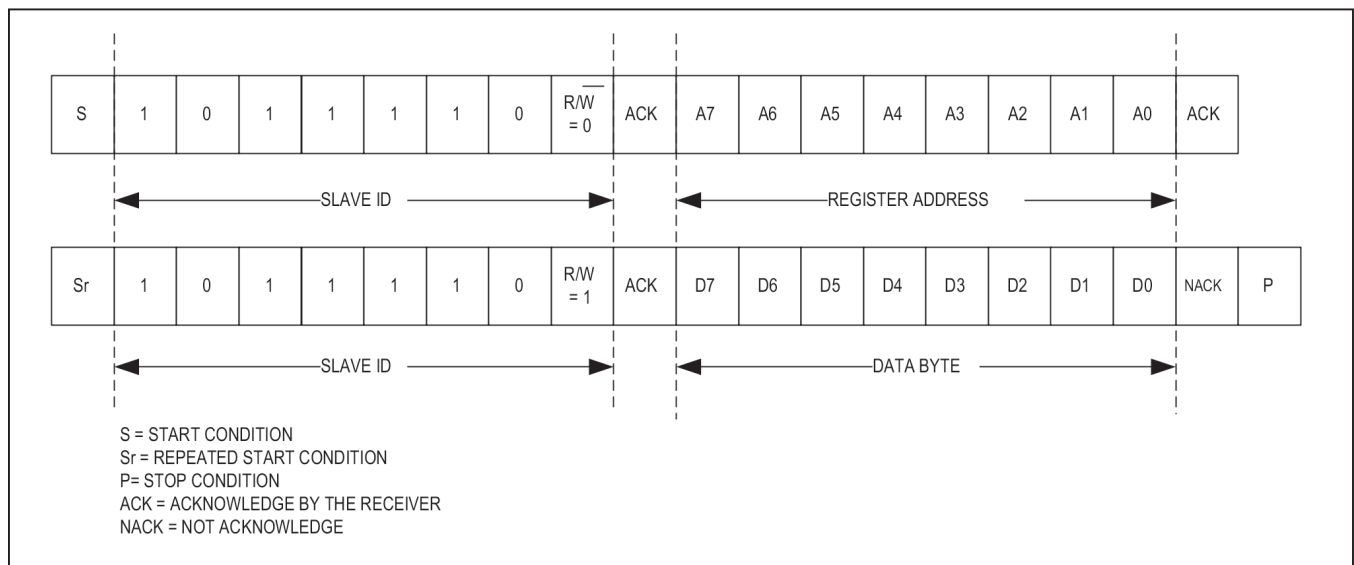


Figure 6. Reading One Byte of Data from MAX86160

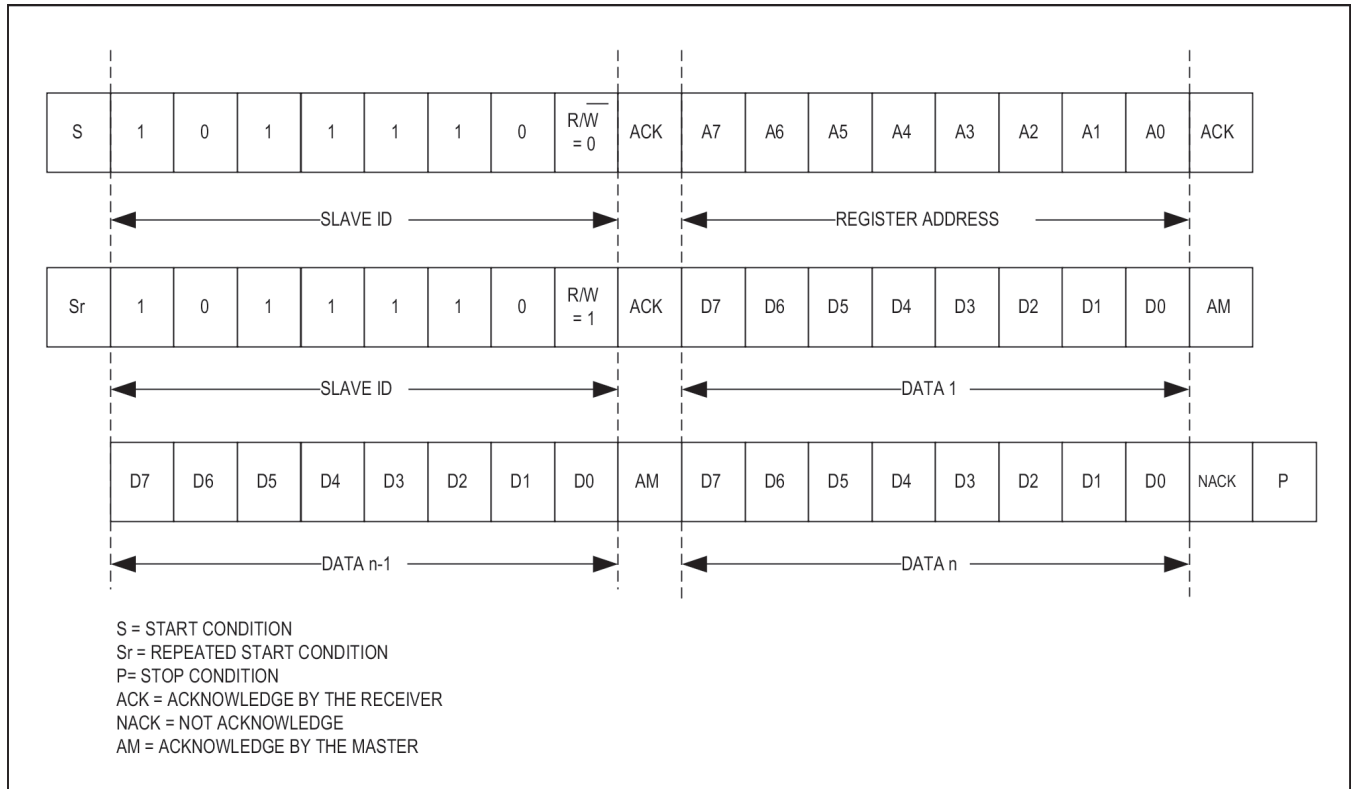


Figure 7. Reading Multiple Bytes of Data from the MAX86160

FIFO Description

Overview

The FLEX FIFO is designed to support configurable number of elements. So the number of elements in each sample is configurable. All elements are of same width, but may be interpreted differently, depending on how the FIFO data is configured. MS bits of any element that is smaller than this width is padded with zeroes. Reading FIFO through the I²C returns only the active FIFO Data corresponding to the current configuration.

The design is also scalable, to support any

- Element width, in number of bits
- Sample length, in number of elements
- FIFO depth, in number of samples.

Each sample is comprised of up to four elements. The actual number of elements in a sample depends on:

- FIFO Data Control Registers 1
- FIFO Data Control Registers 2

FIFO Data Types

FIFO Data Control Registers

Table 2 shows FIFO data control registers which are used for enabling any of the PPG mode.

Table 2. FIFO Data Control Registers

| ADDRESS | REGISTER NAME | HARDWIRED VALUE | B7 | B6 | B5 | B4 | B3 | B2 | B1 | B0 |
|---------|------------------------------------|-----------------|----------|----|----|----|-----------|----|----|----|
| 0x09 | FIFO Data Configuration Register 1 | 00 | FD2<3:0> | | | | FD1<3:0> | | | |
| 0x0A | FIFO Data Configuration Register 2 | 00 | FD4<3:0> | | | | FDS3<3:0> | | | |

FD1, FD2, FD3, and FD4 (FDx<3:0> are configured as shown in Table 3 to hold data as programmed. It also shows the format of the data in the FIFO.

Table 3. FDx Format Configurations

| FDX<3:0> | DATA TYPE | FIFO CONTENT OR DATA DESCRIPTION | NOTE |
|----------|-----------|---------------------------------------|--------------------------|
| 0000 | Reserved | Reserved | |
| 0001 | PPG | PPG_DATA[18:0] for LED1 (IR) | MS bits should be masked |
| 0010 | Reserved | Reserved | |
| 0011 | PPG | PPG_DATA[18:0] for LED3 (Green) | MS bits should be masked |
| 0100 | Reserved | Reserved | |
| 0101 | Reserved | PPG_DATA[18:0] for Pilot LED1 (IR) | MS bits should be masked |
| 0110 | Reserved | Reserved | |
| 0111 | PPG | PPG_DATA[18:0] for Pilot LED3 (Green) | MS bits should be masked |
| 1000 | Reserved | Reserved | |
| 1001 | Reserved | Reserved | |
| 1010 | Reserved | Reserved | |
| 1011 | Reserved | Reserved | |
| 1100 | Reserved | Reserved | |
| 1101 | Reserved | Reserved | |
| 1110 | Reserved | Reserved | |
| 1111 | Reserved | Reserved | |

- If a configuration uses only one element, FD2, FD3 and FD4 are programmed as zeroes, and FD1 is programmed to the required data type.
- If a configuration uses only two elements, FD3 and FD4 are programmed as zeroes, and FD1 and FD2 are programmed to the required data types.
- If a configuration uses only three elements, FD4 is programmed as zeroes, and FD1, FD2 and FD3 are programmed to the required data types.
- If a configuration uses all four elements, FD1, FD2, FD3 and FD4 are programmed to the required Data Types.

PPG Data is left justified as shown in the [Table 4](#). In other words, the MSB bit is always in the bit 18 position regardless of ADC resolution setting. FIFO_DATA[23:19] are "don't care" and should be masked.

Table 4. FIFO Data Format

| | FIFO_DATA[23:0] | | | | | | | | | | | | | | | | | | | | | | | |
|----------------|-----------------|------|------|------|------|--------|------|------|------|------|--------|------|------|------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| | BYTE 1 | | | | | BYTE 2 | | | | | BYTE 3 | | | | | | | | | | | | | |
| ADC Resolution | [23] | [22] | [21] | [20] | [19] | [18] | [17] | [16] | [15] | [14] | [13] | [12] | [11] | [10] | [9] | [8] | [7] | [6] | [5] | [4] | [3] | [2] | [1] | [0] |
| PPG(19-bit) | x | x | x | x | x | | | | | | | | | | | | | | | | | | | |

Example 1: Configurations for 2 elements: PPG (LED1) + PPG (LED3)

| FD1<3:0> | FD2<3:0> | FD3<3:0> | FD4<3:0> |
|------------|------------|-------------|-------------|
| 0001 (PPG) | 0011 (PPG) | 0000 (None) | 0000 (None) |

Example 2: Configurations for 1 elements: PPG (LED1)

| FD1<3:0> | FD2<3:0> | FD3<3:0> | FD4<3:0> |
|------------|-------------|-------------|-------------|
| 0001 (PPG) | 0000 (None) | 0000 (None) | 0000 (None) |

A sample in the FIFO is shown in [Table 5](#).

Table 5. Sample of FIFO Data Index

| INDEX WITHIN A SAMPLE | FIFO_DATA[23:0] |
|-----------------------|----------------------|
| 0 | FD1 data, if enabled |
| 1 | FD2 data, if enabled |
| 2 | FD3 data, if enabled |
| 3 | FD4 data, if enabled |

FIFO Handling

Only the elements corresponding to the active FIFO data are pushed onto the FIFO, and only these are read through the I²C interface. The unused FIFO data are not read through the I²C, so they are "don't care" and not padded with zeroes.

The FIFO handling registers are shown in [Table 6](#).

Table 6. FIFO Handling Registers

| ADDRESS | REGISTER NAME | HARDWIRED VALUE | B7 | B6 | B5 | B4 | B3 | B2 | B1 | B0 |
|---------|--------------------|-----------------|----------------|----|----|--------------------|------------------|----|----|----|
| 0x04 | FIFO Write Pointer | 00 | | | | FIFO_WR_PTR<4:0> | | | | |
| 0x05 | Overflow Counter | 00 | | | | OVF_COUNTER<4:0> | | | | |
| 0x06 | FIFO Read Pointer | 00 | | | | FIFO_RD_PTR<4:0> | | | | |
| 0x07 | FIFO Data Register | 00 | FIFO_DATA<7:0> | | | | | | | |
| 0x08 | FIFO Configuration | 00 | | | | FIFO_ROLLS_ON_FULL | FIFO_A_FULL<3:0> | | | |

Write Pointer to the FIFO, FIFO_WR_PTR[4:0]:

This points to the location where the next sample will be written. This pointer advances for each sample pushed on to the FIFO.

Read Pointer to the FIFO, FIFO_RD_PTR[4:0]:

This points to the location from where the AP gets the next sample from the FIFO through the I²C interface. This advances each time a sample is popped from the FIFO. The AP may also write to this pointer after reading the samples. This allows rereading (or retrying) samples from the FIFO.

FIFO Data Read, FIFO_DATA[7:0]:

This is a read-only register and is used to get data from the FIFO. Reading FIFO_DATA register does not automatically increment the register address. So burst reading this register reads the same address over and over. The length of a sample is determined by the number of active elements in the sample. Each element is three bytes long, so, to read one complete sample, the FIFO_DATA register has to be read N times, where

$$N = (\text{Number of Active Elements}) \times (\text{Number of Bytes}, 3)$$

Reading from the FIFO

Normally, reading registers from the I²C interface auto-increments the register address pointer, so that all the registers can be read in a burst read without an I²C restart event. In this case, this holds true for all registers except for the FIFO_DATA register (register 0x07).

Reading the FIFO_DATA register does not automatically increment the register address. Burst reading this register reads data from the same address over and over. Each sample comprises multiple bytes of data, so multiple bytes should be read from this register (in the same transaction) to get one full sample.

FIFO_RD_PTR advances only after burst reading the entire sample.

Each sample is read from the FIFO in the following order, when all **four** elements are active.

Table 7. FIFO Sample Elements Order with Four Active Elements

| | | | FIFO_RD_PTR[4:0] |
|-----------|-----------|------------------|------------------|
| n Sample: | 1st read | Element 1[23:16] | n |
| | 2nd read | Element 1[15:8] | n |
| | 3rd read | Element 1[7:0] | n |
| | 4th read | Element 2[23:16] | n |
| | 5th read | Element 2[15:8] | n |
| | 6th read | Element 2[7:0] | n |
| | 7th read | Element 3[23:16] | n |
| | 8th read | Element 3[15:8] | n |
| | 9th read | Element 3[7:0] | n |
| | 10th read | Element 4[23:16] | n |
| | 11th read | Element 4[15:8] | n |
| | 12th read | Element 4[7:0] | n |

Table 7. FIFO Sample Elements Order with Four Active Elements (continued)

| | | | FIFO_RD_PTR[4:0] |
|-------------|-----------|------------------|------------------|
| n+1 Sample: | 13th read | Element 1[23:16] | n+1 |
| | 14th read | Element 1[15:8] | n+1 |
| | 15th read | Element 1[7:0] | n+1 |
| | 16th read | Element 2[23:16] | n+1 |
| | 17th read | Element 2[15:8] | n+1 |
| | 18th read | Element 2[7:0] | n+1 |
| | 19th read | Element 3[23:16] | n+1 |
| | 20th read | Element 3[15:8] | n+1 |
| | 21st read | Element 3[7:0] | n+1 |
| | 22nd read | Element 4[23:16] | n+1 |
| | 23rd read | Element 4[15:8] | n+1 |
| | 24th read | Element 4[7:0] | n+1 |
| . | . | . | |
| . | . | . | |
| . | . | . | |
| . | . | . | |

Each sample is read from the FIFO in the following order, when any **two** elements are active.

Table 8. FIFO Sample Elements Order with Two Active Elements

| | | | FIFO_RD_PTR[4:0] |
|-------------|-----------|------------------|------------------|
| n Sample: | 1st read | Element 1[23:16] | n |
| | 2nd read | Element 1[15:8] | n |
| | 3rd read | Element 1[7:0] | n |
| | 4th read | Element 2[23:16] | n |
| | 5th read | Element 2[15:8] | n |
| | 6th read | Element 2[7:0] | n |
| n+1 Sample: | 7th read | Element 1[23:16] | n+1 |
| | 8th read | Element 1[15:8] | n+1 |
| | 9th read | Element 1[7:0] | n+1 |
| | 10th read | Element 2[23:16] | n+1 |
| | 11th read | Element 2[15:8] | n+1 |
| | 12th read | Element 2[7:0] | n+1 |
| . | . | . | |
| . | . | . | |
| . | . | . | |
| . | . | . | |

Enable Push on FIFO FULL, FIFO_ROLLS_ON_FULL: This bit determines whether samples get pushed on to the FIFO when it is full. If push is enabled when FIFO is full, old samples are lost. Otherwise, new samples are lost.

Overflow counter, OVF_COUNTER[4:0]: When the FIFO is full, samples are lost. OVF_COUNTER counts the number of samples lost. It saturates at 0x1F. When a complete sample is popped from the FIFO (when the read pointer advances), and OVF_COUNTER is reset to zero.

FIFO Almost Full Counter, FIFO_AFULL_COUNT[3:0]: This determines the amount of space available in the FIFO, to declare that it is Almost Full.

FIFO Almost Full status, and Interrupt Enable, A_FULL and MSK_A_FULL: When the FIFO is almost full, Almost Full interrupt is asserted if it is enabled by the MSK_A_FULL bit. This prompts the AP to read some samples before the FIFO gets full. A_FULL bit is cleared when the status register is read.

The AP reads the FIFO_WR_PTR and FIFO_RD_PTR to calculate the number of samples available in the FIFO, and read as many samples as it needs up to a maximum of available samples. The AP may then choose to write the new read pointer to the FIFO_RD_PTR register. If necessary to retry, the AP updates the FIFO_RD_PTR register with appropriate value.

Example: Following is an example of the pseudo code:

First transaction: Get the FIFO_WR_PTR and FIFO_RD_PTR:

```
START;
Send device address + write mode
Send address of FIFO_WR_PTR;
REPEATED_START;
Send device address + read mode
Read FIFO_WR_PTR;
Read OVF_COUNTER;
Read FIFO_RD_PTR;
STOP;
```

AP evaluates the number of samples to be read from the FIFO:

```
If OVF_COUNTER is zero,
NUM_AVAILABLE_SAMPLES = FIFO_WR_PTR – FIFO_RD_PTR
(Note: pointer wrap around should be taken into account)
If OVF_COUNTER is non-zero some samples are lost, and
NUM_AVAILABLE_SAMPLES = 32
NUM_SAMPLES_TO_READ = < less than or equal to NUM_AVAILABLE_SAMPLES >
```

Next transaction: Read NUM_SAMPLES_TO_READ samples from the FIFO:

```
START;
Send device address + write mode
Send address of FIFO_DATA;
REPEATED_START;
Send device address + read mode
for (i = 0; i < NUM_SAMPLES_TO_READ; i++) {
Read FIFO_DATA;
Save Data_Item1[23:16];
Read FIFO_DATA;
Save Data_Item1[15:8];
Read FIFO_DATA;
Save Data_Item1[7:0];
}
```

```

Read FIFO_DATA;
Save Data_Item2[23:16];
Read FIFO_DATA;
Save Data_Item2[15:8];
Read FIFO_DATA;
Save Data_Item2[7:0];
Read FIFO_DATA;
Save Data_Item3[23:16];
Read FIFO_DATA;
Save Data_Item3[15:8];
Read FIFO_DATA;
Save Data_Item3[7:0];
}
STOP;

```

Next transaction: Write to FIFO_RD_PTR register. If the pervious transaction was successful, FIFO_RD_PTR points to the next sample in the FIFO, and this transaction is not necessary. Otherwise, the AP will update the FIFO_RD_PTR appropriately to New_FIFO_RD_PTR, so that the samples will be re-read.

```

START;
Send device address + write mode
Send address of FIFO_RD_PTR;
Write New_FIFO_RD_PTR;
STOP;

```

FIFO Flush

The FIFO gets flushed if FIFO_EN = 1, and if any of the following conditions are met:

- I²C write to any of the PPG configuration registers
- I²C write to any of the FIFO data control registers
- At the rising-edge of FIFO_EN
- Enter and exit PROX mode

When the FIFO gets flushed, FIFO_WR_PTR and FIFO_RD_PTR are reset to zero, and the contents of the FIFO are lost.

If FIFO contents should not be lost, set FIFO_EN = 0, before writing to any of the registers listed above.

Note: FIFO_EN bit is in the System Control Register. Data will be pushed to the FIFO, when FIFO_EN = 1. When FIFO_EN = 0, push to FIFO is disabled, but it holds the status of the FIFO (FIFO pointers and the actual data).

FIFO Organization

Figure 8 shows how the samples are organized in the FIFO, when all four elements in a sample are active.

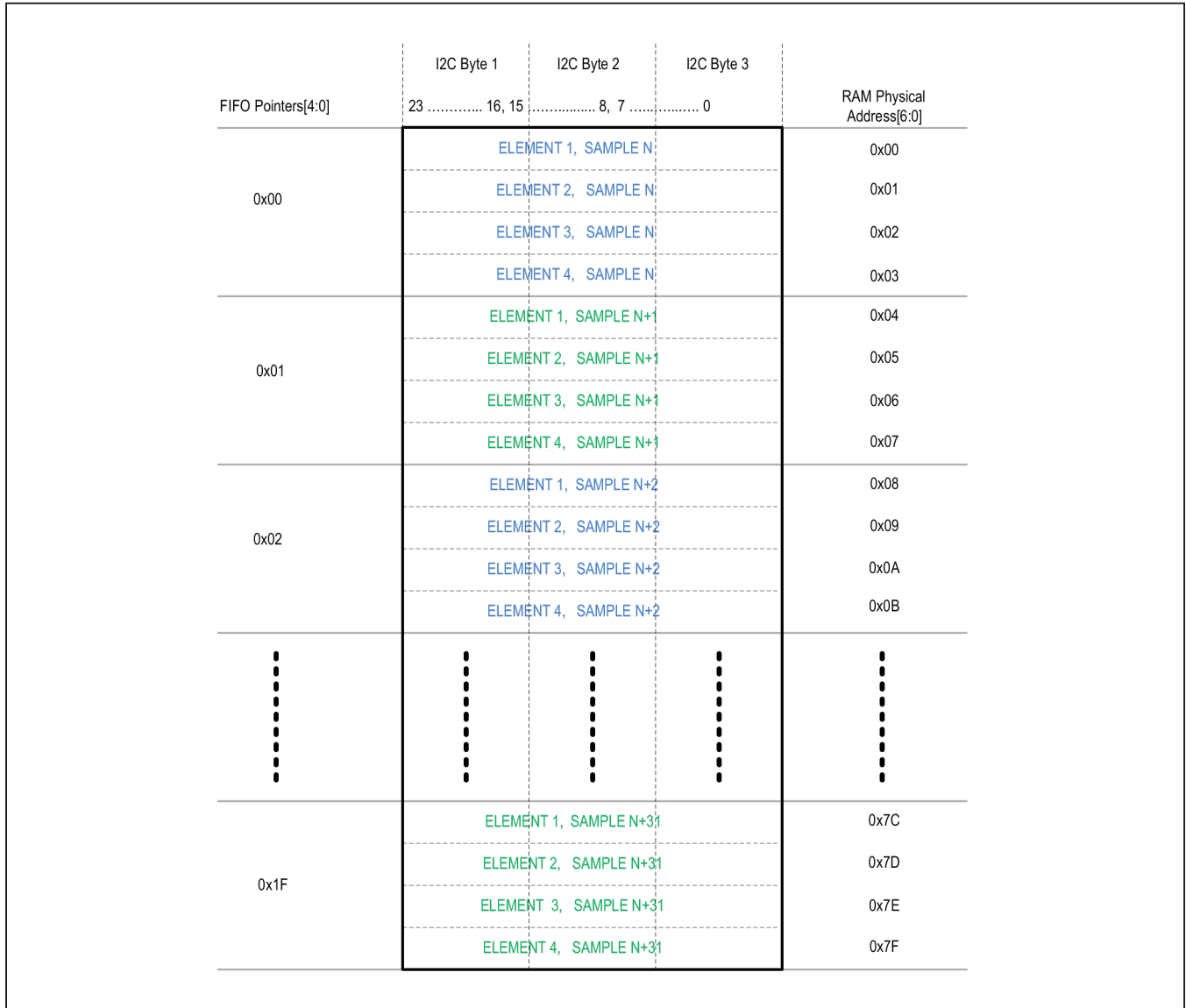


Figure 8. Example of FIFO Organization with Four Active Elements

Figure 9 shows how the samples are organized in the FIFO, when only two elements in a sample are active.

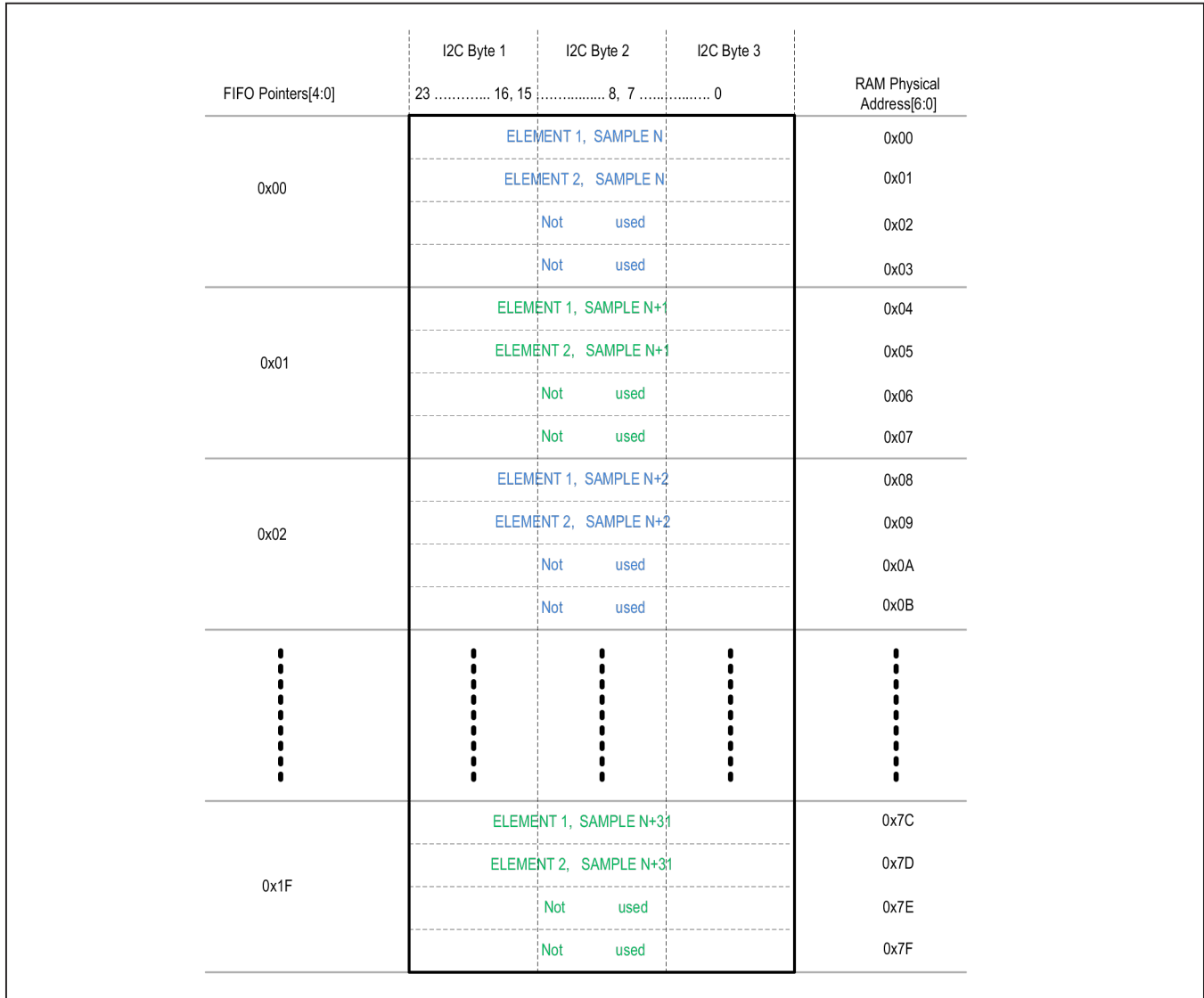
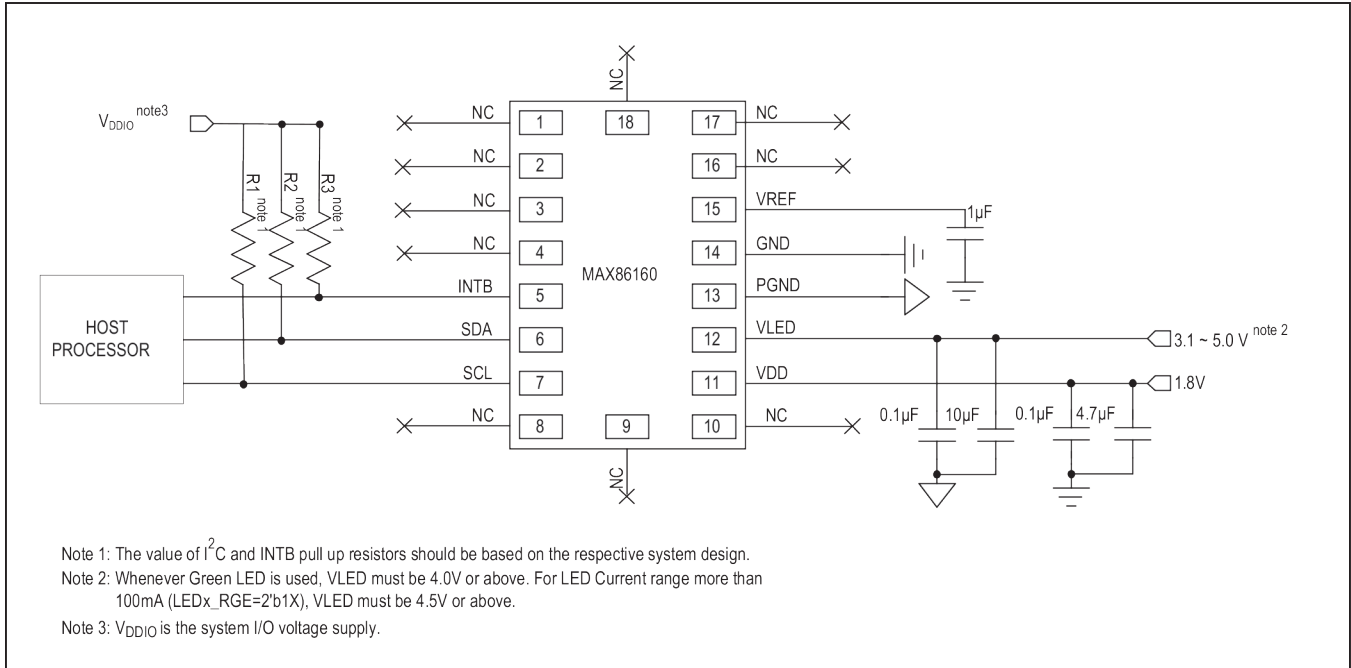


Figure 9. Example of FIFO Organization with Two Active Elements

Typical Application Circuits



Ordering Information

| PART NUMBER | TEMP RANGE | PIN-PACKAGE | [TOP MARKING] |
|---------------|----------------|-------------|---------------|
| MAX86160EFN+T | -40°C to +85°C | 18 PIN LGA | |

+ Denotes a lead(Pb)-free/RoHS-compliant package.

T Denotes tape-and-reel.

Revision History

| REVISION NUMBER | REVISION DATE | DESCRIPTION | PAGES CHANGED |
|-----------------|---------------|------------------------------------|---------------|
| 0 | 9/16 | Initial release | — |
| 1 | 9/19 | Updated <i>General Description</i> | 1 |

For pricing, delivery, and ordering information, please visit Maxim Integrated's online storefront at <https://www.maximintegrated.com/en/storefront/storefront.html>.

Maxim Integrated cannot assume responsibility for use of any circuitry other than circuitry entirely embodied in a Maxim Integrated product. No circuit patent licenses are implied. Maxim Integrated reserves the right to change the circuitry and specifications without notice at any time. The parametric values (min and max limits) shown in the Electrical Characteristics table are guaranteed. Other parametric values quoted in this data sheet are provided for guidance.