General Description

MAX32664C is a sensor hub product which provides the following innovative features:

- AFE sensor drivers.
- Biometric algorithms developed by a team of data scientist and machine learning experts.
- Embedded algorithms.
- Low-powered sensor hub handles the complexity of sensor management :
    - AFE, accelerometer configuration, FIFO access and synchonization.
    - Sensor hub enters deep sleep when idle.
- OTA update compatibility allows the sensor hub to receive the latest algorithm version.
- Faster time to market; development time cut by at least six months.
- Sample host code reduces integration time.
- Reference design includes a host processor, enclosure, and Android app which allows for wrist-band evaluation.

When using the MAXREFDES103, the following three software and firmware must all be updated using the latest and same software release package.

1. Install the Maxim DeviceStudio PC GUI using the **.msi** file
2. Flash the micro board **.bin** file using drag & drop to the DAPLINK folder
3. Flash the algorithm **.msbl** file to MAX32664 using the Maxim viceStudio PC GUI

NOTE: The instructions in this document are compatible with the MAX32664C firmware version 30.13.19+ (MAX86141), 32.9.23 (MAXM86161+KX122), 32.9.33 (MAXM86161+LIS2DS12) , or 33.13.31 (MAXM86146) and later. If you are using older firmware, make sure to upgrade the firmware.

Evaluation Platforms

The MAXREFDE103# may be used to evaluate wrist-based biometric algorithms. The MAXREFDES103# platform includes the following major components:

- MAX32630 host microcontroller board
  - MAX20303 power-management IC (PMIC)
  - Dual-mode Bluetooth® connection
  - Tricolor status LED
- MAX32664 MAX86141 Sensor Hub board
  - MAX86141 analog frontend and optical heart rate, SpO_2 sensor with one green LED, one red LED, one IR LED, and two photodiodes
  - MAX32664 sensor hub microcontroller with embedded heart-rate, SpO_2, skin contact algorithm
  - Three-axis accelerometer
- MAXDAP-TYPE-C Adapter board to be used during a firmware upgrade for the micro board
- Health sensor band enclosure
- Battery
- Two USB Micro-B cables for firmware upgrade of the micro board or for PC communication with the micro board or charging of the health sensor band
- One USB Type-C™ cable for PC communication with the micro board (Windows® 7 or Windows 10) or charging of the health sensor band

Figure 1. MAXREFDES103#

The Bluetooth word mark and logos are registered trademarks owned by Bluetooth SIG, Inc. and any use of such marks by Maxim is under license.

The MAXM86161EVSYS# may be used to evaluate ear-based biometric algorithms. The MAXM86161EVSYS# platform includes the following major components:

- MAXSensorBLE_EVKIT host microcontroller board
  - NRF52832 microcontroller
- MAXM86161OSBFLEX sensor hub board
  - MAXM86161 analog frontend and integrated optical heart rate sensor (three optical readout channels that can be individually controlled (Green, Red, and IR))
  - MAX32664 microcontroller with embedded heart-rate algorithm
  - Three-axis accelerometer
- Two programmer boards
  - MAXREFDES100HDK
  - MAXM86161_ADPTR_EVKIT_B
- Cy Smart CY5677 BLE USB Dongle
- Health sensor band enclosure
- Battery
- One USB Type-C™ cable

The MAXM86146EVSYS# may be used to evaluate chest-based biometric algorithms. The MAXM86146EVSYS# platform includes the following major components:

- MAXSensorBLE_EVKIT host microcontroller board
  - NRF52832 microcontroller
- MAXM86146OSB sensor hub board
  - MAXM86146 integrated microcontroller (MAX32664C), analog frontend (MAX86141) and two photodiodes
  - Three-axis accelerometer
  - Two green LEDs and one combination IR/red LED
- Cy Smart CY5677 BLE USB Dongle
- Health sensor band enclosure
- Battery
- One USB Type-C™ cable

Contents

Architecture

An accelerometer is mandatory for MAX32664C heart rate monitoring. A KX122 or LIS2DS12 accelerometer may be connected directly to the sensor hub. The interrupt line of the accelerometer may be connected to the sensor hub to support power saving using motion detection. Alternatively, an external 3-axis host-side accelerometer can be used. In this case, the host needs to periodically provide accelerometer readings to the sensor hub using the commands provided in this document.

The optical sensor utilizes green, infrared (IR), and red LEDs to transmit pulses and one or more photodiodes (PD) to collect reflected or residual light. By default, the heart rate monitoring algorithm for MAXREFDES103 uses green1 LED (LED1) and two PDs (PD1 and PD2). The SpO_2 algorithm employs one IR LED (LED2) and one red LED (LED3) with one PD (PD1).

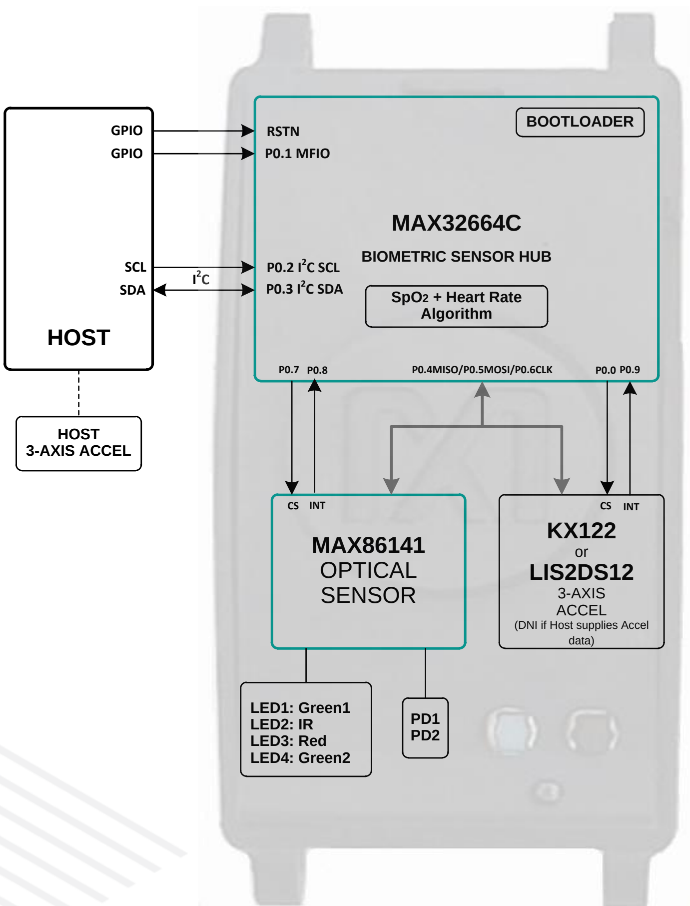


Figure 2. Architecture diagram for health-sensing applications using the MAX32664C sensor hub and the MAX86141 sensor (SPI). Kionix KX122 or STM LIS2DS12 accel are supported (30.13.12+)

Kionix is a registered trademark of Kionix, Inc.

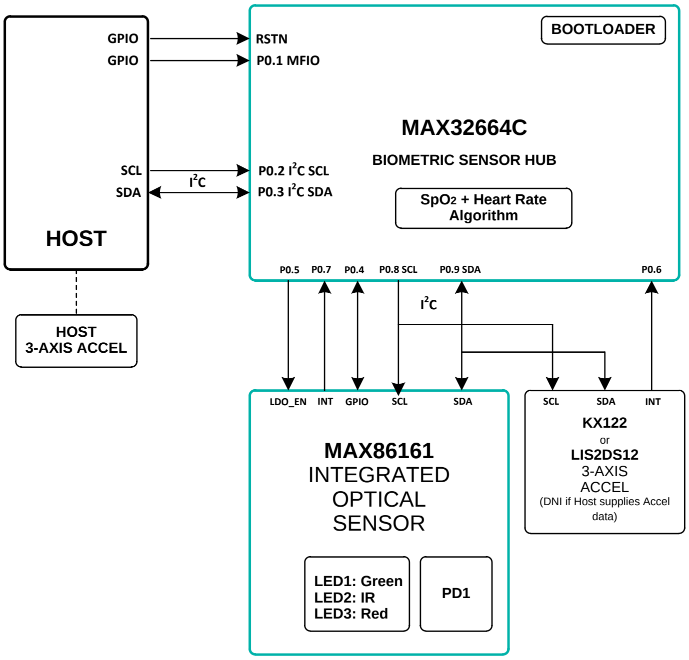


Figure 3. Architecture diagram for health-sensing applications using the MAX32664C sensor hub and the MAX86161 sensor (I^2C). Kionix KX122 (32.9.23) or STM LIS2DS12 (32.9.33) accel are supported.

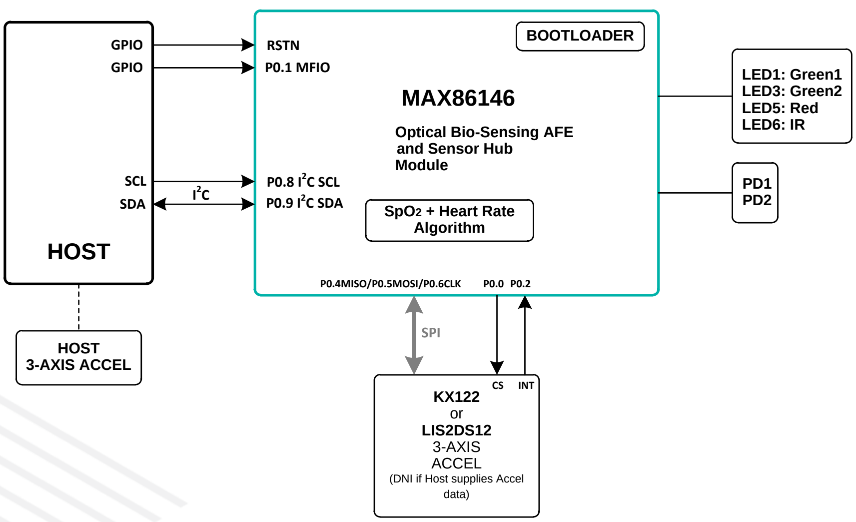


Figure 4. Architecture diagram for health-sensing applications using MAXM86146 Module. Kionix KX122 or STM LIS2DS12 accel (SPI) are supported (33.13.12+).

MAX32664C Reset

In order to reset the device into bootloader or application mode, two GPIO pins from the host are needed to control the RSTN and multifunction input/output (MFIO) pins.

Reset to Bootloader Mode

To enter Bootloader mode:

- Set the RSTN pin low.
- While RSTN is low, set the MFIO pin to low. (The MFIO pin should be set to low at least 1ms before the RSTN pin is set to high.)
- After the 10ms has elapsed, set the RSTN pin to high.
- After an additional 50ms has elapsed, the sensor hub is in Bootloader mode.

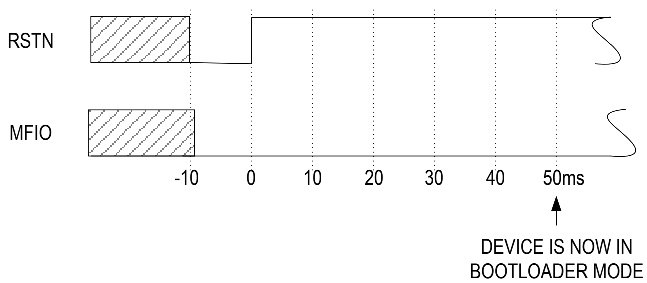


Figure 5 Entering bootloader mode using the RSTN pin and the MFIO GPIO pin.

Reset to Application Mode

To enter Application mode:

- Set the RSTN pin low.
- While RSTN is low, set the MFIO pin to high (The MFIO pin should be set to high at least 1ms before the RSTN pin is set to high).
- After the 10ms has elapsed, set the RSTN pin to high
- After an additional 50ms has elapsed, the sensor hub is in Application mode and the application performs its initialization of the application software.
- After approximately 1.5s from when the RSTN pin was set to high, the application completes the initialization, and the device is ready to accept I^{2}C commands.

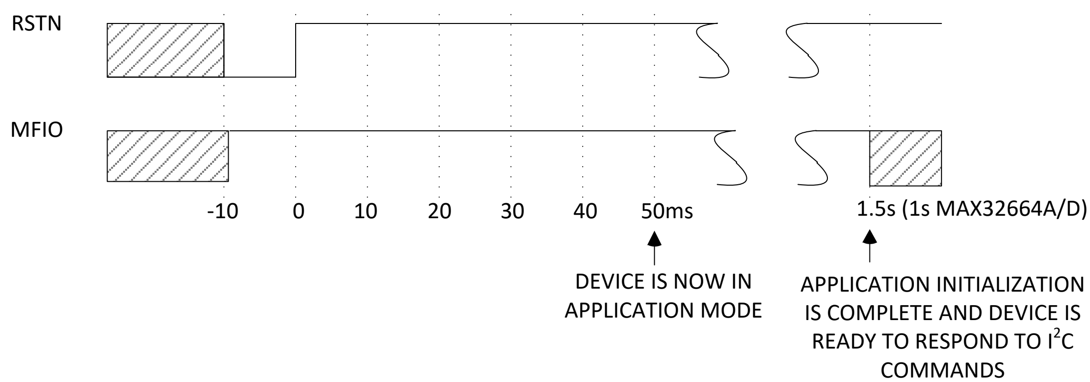


Figure 6 Entering application mode using the RSTN pin and MFIO pin.

Reset to Bootloader and then Timing Out to Application Mode

To enter Application mode by timing out from Bootloader mode:

- Set the RSTN pin low.
- While RSTN is low, set the MFIO pin to low (The MFIO pin should be set to low at least 1ms before the RSTN pin is set to high).
- After the 10ms has elapsed, set the RSTN pin to high.
- After an additional 50ms has elapsed, the sensor hub is in Bootloader mode.
- If no I²C commands are sent to the sensor hub within the next 1s, then the sensor hub will automatically switch to application mode.

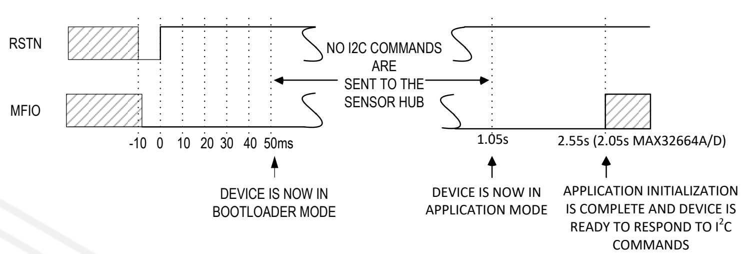


Figure 7 Entering application mode by timing out from Bootloader mode.

Sensor Hub Handshaking: I²C and MFIO pin

Normal MFIO Mode (default, host periodically empties the FIFO)

When the sensor hub is idle, it switches to deep sleep mode to save power. An external interrupt-like sensor, host MFIO, or RTC alarm forces the sensor hub to wake up. Do not keep the MFIO pin at a constant low – when the MFIO pin is high, the sensor hub wakes up to service the AFE.

The host is required to wake up the sensor hub prior to any I²C communication by:

- Setting the MFIO pin to low at least 300µs before the beginning of an I^2C transaction to wake the sensor hub.
- Keeping the MFIO pin low during the I^2C transaction.
- Setting MFIO to high after the end of I^2C communication to allow the sensor hub to switch back to deep sleep/AFE servicing mode.

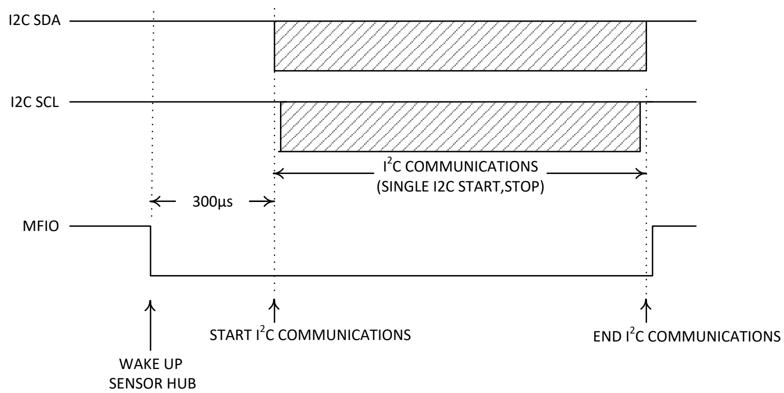


Figure 8. Normal MFIO Mode: host uses MFIO for enabling I^2C host communications.

MFIO Interrupt Mode

MFIO interrupt mode is mode where FIFO pin is used by the sensor hub to interrupt Host when: measurement data is ready, report FIFO overflow events and skin contact detection event (when SCD state machine is enabled – see section SCD State Machine for MFIO Interrupt Mode). (NA for 32.x.x).

- To set up for MFIO interrupt mode, the host sets the MFIO pin to input mode and assigns a GPIO interrupt handler to this pin to receive event reporting from the sensor hub.
- The host initializes the sensor hub to AEC mode.
- Sensor hub enters MFIO interrupt mode with command sequence 0xB8, 0x01.
- After the interrupt is received by the host, the host should wake the sensor hub. When in MFIO interrupt mode, the sensor hub is woken up by:
  - Host sends byte 0x00 to I^2C slave address 0x00 and waits for 200us
  - Host sends bytes 0x00 to I^2C slave address 0x00 and waits for 150us
- Host sends commands to read samples in the output FIFO.

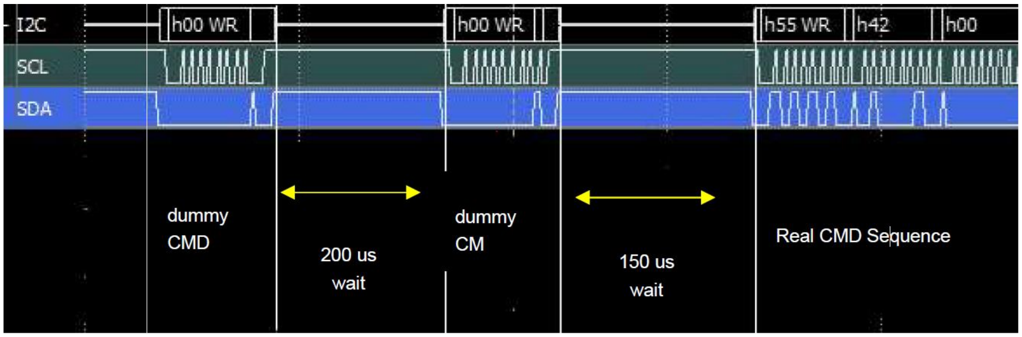


Figure 9. SDA wake-up sequence for sensor hub.

Sensor Hub I²C Communications

A host uses the I²C bus to communicate with the sensor hub (slave) using a series of commands. The default CMD_DELAY is 2ms. The I²C data rate 400Kbps has been tested; the 1000Kbps data rate has not been tested.

`Slave_WriteAddress` and `Slave_ReadAddress` are set to 0xAA and 0xAB, respectively.

A generic write command includes the following fields:
`Slave_WriteAddress(1 byte)|Command_Family(1 byte)|[Index byte]|[Write byte]|[additional command byte(s)]`

A generic response includes the following fields:
`Slave_ReadAddress(1 byte)|Read Status Byte|Value (multiple bytes)`

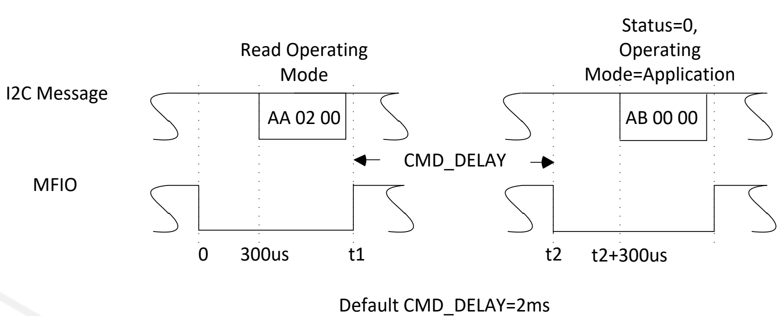


Figure 10. Example I²C, MFIO example command response.

MAX32664 I²C Bit Transfer Process.

The read status byte is an indicator of success (0x00) or failure, as detailed in the following table.

The defined bit transfer process is described below. It is recommended that I²C GPIO 'bit-bang' software be implemented on the host if the host MCU I²C hardware/HAL is not compatible with sensor hub protocol.

Both SDA and SCL signals are open-drain circuits. Each has an external pullup resistor that ensures each circuit is high when idle. The I²C specification states that during data transfer, the SDA line can change state only when SCL is low, and that SDA is stable and able to be read when SCL is high. Typical I²C write/read transactions are shown below.

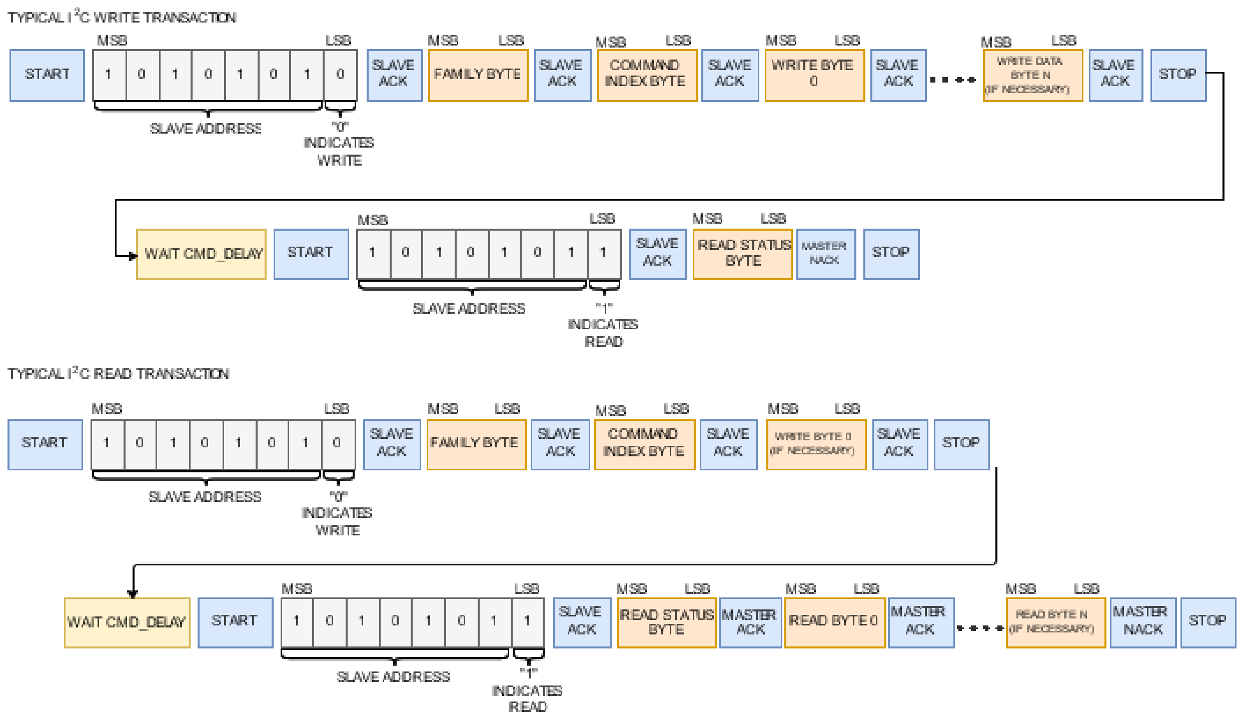


Figure 1. I²C Write/Read data transfer from host microcontroller.

The read status byte is an indicator of the success or failure of the Write Transaction. The read status byte must be accessed after each write transaction to the device. This ensures that write transaction processing is understood and any errors in the device command handling can be corrected. The value of the read status byte is summarized in the next table.

Read Status Byte

The read status byte is an indicator of success (0x00) or failure, as detailed in the following table.

Table 1. Read Status Byte Definition

READ STATUS BYTE VALUE	DESCRIPTION
0x00	The write transaction was successful.
0x01	Illegal Family Byte and/or Index Byte was used. Verify that the Family Byte, Index Byte are valid for the host command sent.
0x02	Illegal Index Byte and/or Write Byte was used. Verify that the Index Byte and Write Byte(s) are valid for the host command sent.
0x03	Incorrect number of bytes sent for the requested Family Byte. Verify that the correct number of bytes are sent for the host command.
0x04	Illegal configuration value was attempted to be set. Verify that the Family Byte, Index Byte, and Write Byte are correct.
0x05	Not used in application mode. (In bootloader: Device is busy. Insert delay and resend the host command.)

0x80	Not used. General error while receiving/flashing a page during the bootloader sequence. Not used.
0x81	Bootloader checksum error while decrypting/checking page data. Verify that the keyed .msbl file is compatible with MAX32664.
0x82	Bootloader authorization error. Verify that the keyed .msbl file is compatible with MAX32664.
0x83	Bootloader detected that the application is not valid.
0xFF	Unknown error. Verify that the communications to the AFE/KX122/LIS2DS12 are correct by reading the PART_ID/WHO_AM_I register. For MAX32664B/C, the sensor hub is in deep sleep unless the host sets the MFIO pin low 300us before and during the I²C communications. When switching to bootloader mode, allow 50ms for initialization. When switching to application mode, allow 1.5s for initialization.
NAK	NAK received. Sensor hub was busy. Resend command after 1ms with a maximum of five retries. If this issue persists, then empty the FIFO by reading all the data or reduce the report rate. Verify that the hardware I²C /MFIO rise times, voltage levels, and grounding are correct. For MAX32664B/C, verify that the MFIO line fall time is clean; increase the MFIO pin low time to wake to 300us.

Sensor Hub I²C Command Response Definitions

The following table defines the host to MAX32647C I²C command response definitions. The Read Status Byte is not listed in this table.

Table 2. Sensor Hub I2C Command Response Definitions

FAMILY BYTE	INDEX BYTE	WRITE BYTE(S)	DESCRIPTION	READ BYTE(S) RESPONSE (Read Status Byte not listed)
0x00 read	0x00		Read sensor hub status	**Err0[bit0]**: 0 = No error; 1 = Sensor communication problem **Err1[bit1]**: Not used **Err2[bit2]**: Not used **DataRdyInt[bit3]**: 0 = FIFO below threshold; 1 = FIFO filled to threshold or above. **FifoOutOvrInt[bit4]**: 0 = No FIFO overflow; 1 = Sensor hub output FIFO overflowed, data lost. **FifoInOvrInt[bit5]**: 0 = No FIFO overflow; 1 = Sensor hub Input FIFO overflowed, data lost. **HostAccelUfInt[bit6]**: 0 = No FIFO overflow; 1 = host data to input FIFO is slow and the input FIFO has underflowed.

				SCDMonSkin[bit7]: 0 = Skin not detected. 1 = Skin detected.
0x01 write	0x00	**0x00**: Exit bootloader mode, enter application mode (CMD_DELAY = 1.5s) **0x01**: Shutdown the MAX32664. Restart by power cycling or pulsing RSTN. **0x08**: Invalidate Application and Enter bootloader mode. CMD_DELAY 4.25s (this command uses two of the 10,000 write cycles). Reset the MAX32664 after the 4.25s by setting RSTN low for 10ms and waiting 50ms (33.13.31, NA for 32.9.x, NA for 30.x.x)	Set the device operating mode.	
0x02 read	0x00		Read the device operating mode.	0x00: Application operating mode. 0x02: Reset. 0x08: Bootloader operating mode.
0x10 write	0x00	**0x00**: (no data) **0x01**: Sensor Data **0x02**: Algorithm Data **0x03**: Sensor Data and	Enable/disable the inclusion of sample counter, raw data or algorithm data in the output FIFO. See I²C Sensor Hub Output FIFO Samples Report Format.	

		Algorithm Data **0x04**: (no data) **0x05**: Sample Counter byte, Sensor Data **0x06**: Sample Counter byte, Algorithm Data **0x07**: Sample Counter byte, Sensor Data and Algorithm Data		
0x11 read	0x00		Read the settings for the inclusion of sample counter, raw data or algorithm data in the output FIFO.	**0x00**: Pause (no data) **0x01**: Sensor Data **0x02**: Algorithm Data **0x03**: Sensor Data and Algorithm Data **0x04**: Pause (no data) **0x05**: Sample Counter byte, Sensor Data **0x06**: Sample Counter byte, Algorithm Data **0x07**: Sample Counter byte, Sensor Data, and Algorithm Data
0x10 write	0x01	0x01 to 0xFF: Sensor Hub DataRdyInt Threshold for FIFO.	Set the threshold for the FIFO DataRdyInt bit. The bit DataRdyInt of the sensor hub status byte is set when this threshold is reached.	
0x11 read	0x01	-	Read the threshold for the FIFO interrupt bit. The bit DataRdyInt of the sensor hub status byte is set when this threshold is reached.	**0x01 to 0xFF**: Sensor Hub DataRdyInt Interrupt Threshold for FIFO.
0x11 read	0x02	-	Read the samples report period (e.g., a value of 25 means a samples report is generated once every 25 samples).	**0x01 (default) to 0xFE:** lsb is 40ms. N, where a samples report is generated once every N samples.
0x10 write	0x02	**0x01 to 0xFE:** lsb is 40ms. N, where a samples report is generated once every N samples.	Set the samples report period (e.g., a value of 25 means a samples report is generated once every 25 samples).	-
0x10 write	0x03	**0x02 to 0xFF**: New I^{2}C	Change I^{2}C address of the MAX32664.	

		address (8-bit I^2C write address)		
0x10 write	0x04	**0x00 to 0xFF**: Counter	Set the sensor hub counter.	
0x10 write	0x12	**0x00**: 0, Disable **0x01**: 1, Enable	MAXM86161: Disable/enable LDO_EN, P0.5, WLP-D4. (32.9.23, 32.9.33)	
0x10 write	0x13	**0x00**: 0, Disable **0x01**: 1, Enable	MAXM86161: Disable/enable MAXM86161 GPIO, P0.4, WLP-D3. Not used by .msbl algorithm – GPIO_CTRL is 0; could be used in raw data mode. (32.9.23, 32.9.33)	
0x11 read	0x04		Read the sensor hub counter.	0x00 to 0xFF: Counter
0x11 read	0x05		Read PPG output FIFO samples report size. (32.9.23, 32.9.33, 33.13.19+)	Number of bytes in the PPG samples report
0x11 read	0x06	**0x01**: Normal Algorithm Report **0x02**: Extended Algorithm Report **0x03**: SCD Only Algorithm Report (N/A 32.9.x) **0x04**: Packed Normal Algorithm Report (32.9.33, 33.13.31)	Read algorithm output FIFO samples report size. (32.9.23, 32.9.33, 33.13.19+)	Number of bytes in the algorithm samples report
0x12 read	0x00		Read the number of samples available in the output FIFO	Number of samples available in the FIFO.
0x20			Read the number of samples available in the output FIFO (one byte command synonym of 0x12 0x00) (32.9.33, 33.13.31)	Number of samples available in the FIFO.
0x12 read	0x01		Read data stored in output FIFO.	Samples Report from Output FIFO. The internal FIFO read pointer increments once the sample size bytes have been read. See the Read Status Byte and Samples Report tables for further details.
0x21 read			Read data stored in output FIFO. (one byte command synonym of 0x12 0x01) (32.9.33, 33.13.31)	Samples Report from Output FIFO.

				The internal FIFO read pointer increments once the sample size bytes have been read. See the Read Status Byte and Samples Report tables for further details.
0x13 read	0x01	0x04	Read the sensor sample size for the accelerometer input FIFO	0x00 0x06 Six bytes
0x13 read	0x01		Read the input FIFO size for the maximum number of samples that the input FIFO can hold.	Input FIFO size (16-bit unsigned)
0x14 write	0x00	N*[LSBx MSBx, LSBy MSBy, LSBy MSBz] N sets of data are sent, where N is the 40 msec samples report period.	Write data to the input FIFO. If the host has been enabled (0x44 0x04 0x01 0x01), then this command may be used supply host accel data to the sensor hub. Example, send five sets of host accel [LSBx MSBx, LSBy MSBy, LSBy MSBz] data to the sensor hub: 14 00 84 FE 1C 00 55 FC 85 FE 1B 00 57 FC 84 FE 1A 00 56 FC 85 FE 19 00 54 FC 87 FE 1B 00 57 FC -380=FE84, 28=001C, -939=FC55; -379=85FE, 27=001B, -938=FC56; -380=FE84, , 26=001A, -938=FC56; -379=85FE, 25=0019, -940=FC54; -377=87FE, 27=001B, -937=FC57	[MSB LSB] Number of bytes received.
0x40 write	0x00	[reg addr] [reg value]	Write a value to a writable MAX86140/MAX86141/MAXM86161 register.	
0x41 read	0x00	[reg addr]	Read the value of a MAX86140/MAX86141/MAXM86161 register.	Register value (byte)
0x40 write	0x04	[reg addr] [reg value]	Write a value to a writable accelerometer sensor register.	
0x41 read	0x04	[reg addr]	Read accelerometer sensor register.	Register value (byte)
0x42 read	0x00		Read the attributes of the MAX86140/MAX86141/MAXM86146/MAXM86161 AFE.	Number of bytes in a word for this sensor, Number of registers available for this sensor.
0x42 read	0x04		Read the attributes of the accelerometer.	Number of bytes in a word for this sensor, Number of registers available for this sensor.
0x43 read	0x00		Read all the MAX86140/ MAX86141/ MAXM86161 registers.	Register address 0, register value 0, register address 1, register value 1, …, register address n, register value n.

0x43 read	0x04		Read all the accelerometer registers.	Register address 0, register value 0, register address 1, register value 1, …, register address n, register value n.
0x44 write	0x00	**0x00**: Disable **0x01**: Enable CMD_DELAY = 465ms	Write the enable/disable the MAX86140/MAX86141/MAXM86146/ MAXM86161 sensor.	
0x45 read	0x00		Read the enable/disable of the MAX86140/MAX86141/ MAXM86146/MAXM86161	**0x00**: Disable **0x01**: Enable
0x44 write	0x04	**0x00, 0x00:** Disable sensor hub accelerometer **0x00, 0x01:** Disable external host accelerometer **0x01, 0x00:** Enable sensor hub accelerometer **0x01, 0x01:** Enable external host accelerometer CMD_DELAY = 20ms	Write the enable/disable, sensor hub accel/external accel sensor.	
0x45 read	0x04		Read the enable/disable, sensor hub accel/external accel sensor.	**0x00**, **0x00:** Sensor hub accelerometer disabled **0x00**, **0x01:** External host accelerometer disabled **0x01**, **0x00:** Sensor hub accelerometer enabled **0x01**, **0x01:** External host accelerometer enabled
0x44 write	0xFF	0x44	Single command to enable multiple sensors. CMD_DELAY = 465ms for AFE+accel sensor. Example: Enable [MAX86140/MAX86141/MAXM86146/MAXM86161] and accel. 44 FF 02 04 00 00 00 00 00 Exceptions: 1. If any sensor in the list is already enabled, it turns off and enables again. 2. If enabling one of the sensors in the list fails, the sensor hub disables all the sensors in the command list.	**N, SI, SM, SE, … SI, SM, SE:** Enable multiple sensors, where: • N is the number of sensors • SI is the sensor index • SM is the sensor mode • SE is 1 if the sensor is an external host or 0 if the sensor is connected to the sensor hub Sensor indices are defined as:

			3. All sensors in this command list must be valid available hardware, otherwise, the sensor hub disables all the sensors listed in this command.	**0x00:** MAX86140/ MAX86141/ MAXM86146/ MAXM86161 **0x01:** MAX30205 **0x02:** MAX30001 **0x03:** MAX30101/MAX30102 **0x04:** Accelerometer Sensor modes are defined in the first byte of Write Bytes field of the Sensor Mode Enable commands, 0x44 0x00 to 0x44 0x04
0x46 write	0x04	00 [byte1] [byte2] [byte3]	Write enable/disable wake up on motion detection (3-byte value): byte1: **0x00:** Disable **0x01:** Enable byte2: **0x01 to 0xFE:** Wake up filter period (seconds). Motion must be present during this period time before a wake-up is generated. **0xFF:** Disable byte3: **0x01 to 0x80** lsb = 0.0625g (1/16g. For example, 0x08 is 0.5g. *As defined in the KX122/LIS2DS12 data sheet. **To disable wake up on motion, use 0x00FFFF.**	
0x50 for write	0x07	0x00 [A_MSB .. .. A_LSB] [B_MSB .. .. B_LSB] [C_MSB .. .. C_LSB]	Write SpO_2 calibration coefficients (12 bytes comprised of three 32-bit signed values (A, B, C), scaled up by 100,000).	
0x51 for read	0x07	0x00	Read SpO_2 calibration coefficients (12 bytes comprised of three 32-bit signed values, scaled up by 100,000). Default: 0x00000000 FFD7FBDD 00AB61FE A = 0 (0x00000000) B = -26.224999 (0xFFD7FBDD)	32-bit signed integer A, 32-bit signed integer B, 32-bit signed integer C Values scaled up by 100,000

			C = 112.317421 (0x00AB61FE)	
0x50 for write	0x07	0x01 [MSB LSB]	Write SpO_2 motion-detection period (unsigned 16-bit int, seconds). The algorithm will consider the state to be motionless if the motion is below the threshold for this duration of time.	
0x51 for read	0x07	0x01	Read SpO_2 motion-detection period. The algorithm will consider the state to be motionless if the motion is below the threshold for this duration of time. Default: 0x0002	MSB of period, LSB of period (16-bit unsigned integer**,** seconds)
0x50 for write	0x07	0x02 [MSB .. .. LSB]	Write SpO_2 motion-detection threshold (signed 32-bit int, equal to 10^5 x milli-g threshold value).	
0x51 for read	0x07	0x02	Read SpO_2 motion-detection threshold (signed 32-bit int [MSB .. .. LSB], equal to 10^5 x milli-g threshold value). Default: 0x01C9C380 (0.3g)	4 bytes (32-bit signed integers which are the milli-g motion threshold times 100,000)
0x50 for write	0x07	0x03 [byte]	Write SpO_2 AGC Timeout (sec).	
0x51 for read	0x07	0x03	Read SpO_2 AGC Timeout (sec). Default: 0x3C	SpO_2 AGC Timeout (sec, 8-bit unsigned)
0x50 for write	0x07	0x04 [byte]	Write the timeout duration for SpO_2 measurement in seconds (1 byte).	
0x51 for read	0x07	0x04	Read the timeout duration for SpO_2 measurement in seconds. Default: 0x5A	SpO_2 algorithm timeout (sec, 8-bit unsigned)
0x50 for write	0x07	0x05 [byte]	Write initial HR algorithm value (8-bit unsigned).	
0x51 for read	0x07	0x05	Read initial HR algorithm value. Default: 0x3C	Initial heart rate setting (8-bit unsigned)
0x50 for write	0x07	0x06 [MSB] [LSB]	Write height (16-bit unsigned, cm).	
0x51 for read	0x07	0x06	Read height. Default: 0x00AF	Height (16-bit unsigned, cm)
0x50 for write	0x07	0x07 [MSB] [LSB]	Write weight (16-bit unsigned, kg).	
0x51 for read	0x07	0x07	Read weight. Default: 0x004E	Weight (16-bit unsigned, kg)
0x50 for write	0x07	0x08 [byte]	Write age (8-bit unsigned, years).	
0x51 for read	0x07	0x08	Read age. Default: 0x1E	Age (8-bit unsigned, years)
0x50 for write	0x07	0x09 [byte]	Write gender. byte: 0x00: Male 0x01: Female	
0x51 for read	0x07	0x09	Read gender	Gender **0x00**: Male (default) **0x01**: Female

0x50 for write	0x07	0x0A [algo operation mode byte]	Set the algorithm operation mode (can be switched in runtime): **0x00**: Continuous HR + Continuous SpO_2 **0x01**: Continuous HR + One-Shot SpO_2 **0x02**: Continuous HR **0x03**: Sampled HR (One-Shot) **0x04**: Sampled HR (One-Shot) + One-Shot SpO_2 **0x05**: Activity Tracking ONLY (Applicable to wrist form factor only, MAX86141/0, MAXM86146) **0x06**: SpO_2 Calibration Data Collection	-
0x51 for read	0x07	0x0A	Read the algorithm operation mode.	**0x00**: Continuous HR + Continuous SpO_2 **0x01**: Continuous HR + One-Shot SpO_2 **0x02**: Continuous HR **0x03**: Sampled HR (One-Shot) **0x04**: Sampled HR (One-Shot) + One-Shot SpO_2 **0x05**: Activity Tracking ONLY (Applicable to wrist form factor only, MAX86141/0, MAXM86146) **0x06**: SpO_2 Calibration Data Collection
0x50 for write	0x07	0x0B [byte]	Write the enable/disable AEC byte: **0x00**: Disable **0x01**: Enable	
0x51 for read	0x07	0x0B	Read the enable/disable AEC.	**0x00**: Disabled **0x01**: Enabled (default)
0x50 for write	0x07	0x0C [byte]	Write the enable/disable Skin Contact Detection (SCD) algorithm. byte: **0x00**: Disable **0x01**: Enable	
0x51 for read	0x07	0x0C	Read the enable/disable Skin Contact Detection (SCD) algorithm.	**0x00**: Disabled **0x01**: Enabled (default)
0x50 for write	0x07	0x0D [MSB] [LSB]	Write adjusted target PD current period (16-bit unsigned, seconds)	
0x51 for read	0x07	0x0D	Read adjusted target PD current period in seconds. Default: 0x0708	Adjusted target PD current period (16-bit unsigned, seconds)
0x50 for write	0x07	0x0E [MSB] [LSB]	Write HR motion magnitude threshold (16-bit unsigned, 0.001g)	
0x51 for read	0x07	0x0E	Read HR motion magnitude threshold. Default: 0x0032 (0.05g)	[MSB] [LSB] motion magnitude threshold (16-bit unsigned, 0.001g)

0x50 for write	0x07	0x0F [MSB] [LSB]	Write minimum PD current (16-bit unsigned, 0.1uA).	
0x51 for read	0x07	0x0F	Read minimum PD current. Default: 0x0032	minimum PD current (16-bit unsigned, 0.1uA)
0x50 for write	0x07	0x10 [MSB] [LSB]	Write initial PD current (16-bit unsigned, 0.1uA). This sets the target PD current you would like AEC algorithm to maintain initially. It does not correspond to any register. Once you set what PD current you need, algorithm will calculate the appropriate LED current.	
0x51 for read	0x07	0x10	Read initial PD current. Default: 0x0064	initial PD current (16-bit unsigned, 0.1uA)
0x50 for write	0x07	0x11 [MSB] [LSB]	Write target PD current (16-bit unsigned, 0.1uA). Applicable only if Auto Target PD Current Calculation is enabled.	
0x51 for read	0x07	0x11	Read target PD current. Default: 0x0064	Target PD current (16-bit unsigned, 0.1uA).
0x50 for write	0x07	0x12 [byte]	Write enable/disable automatic calculation of target PD current. byte: **0x00**: Disable **0x01**: Enable	
0x51 for read	0x07	0x12	Read enable/disable automatic calculation of target PD current	**0x00**: Disable **0x01**: Enable (default)
0x50 for write	0x07	0x13 [byte]	Write minimum integration time. byte: **0x00**: 14.8µs **0x01**: 29.4µs **0x02**: 58.7µs **0x03**: 117.3µs	
0x51 for read	0x07	0x13	Read minimum integration time.	Minimum integration time. **0x00**: 14.8µs (default) **0x01**: 29.4µs **0x02**: 58.7µs **0x03**: 117.3µs
0x50 for write	0x07	0x14 [byte]	Write minimum sampling rate and averaging. byte: 0x00: 25sps, avg = 1 0x01: 50sps, avg = 2 0x02: 100sps, avg = 4 0x03: 200sps, avg = 8 0x04: 400sps, avg = 16	
0x51 for read	0x07	0x14	Read minimum sampling rate and averaging.	Minimum sampling rate and averaging. **0x00**: 25sps, avg = 1 **0x01**: 50sps, avg = 2 (default) **0x02**: 100sps, avg = 4 **0x03**: 200sps, avg = 8 **0x04**: 400sps, avg = 16

0x50 for write	0x07	0x15 [byte]	Write maximum integration time: byte: **0x00**: 14.8µs **0x01**: 29.4µs **0x02**: 58.7µs **0x03**: 117.3µs	
0x51 for read	0x07	0x15	Read maximum integration time:	Maximum integration time: **0x00**: 14.8µs **0x01**: 29.4µs **0x02**: 58.7µs **0x03**: 117.3µs (default)
0x50 for write	0x07	0x16 [byte]	Write maximum sampling rate and averaging: byte: **0x00**: 25sps, avg = 1 **0x01**: 50sps, avg = 2 **0x02**: 100sps, avg = 4 **0x03**: 200sps, avg = 8 **0x04**: 400sps, avg = 16	
0x51 for read	0x07	0x16	Read maximum sampling rate and averaging:	Maximum sampling rate and averaging: **0x00**: 25sps, avg = 1 **0x01**: 50sps, avg = 2 **0x02**: 100sps, avg = 4 (default) **0x03**: 200sps, avg = 8 **0x04**: 400sps, avg = 16
0x50 for write	0x07	0x17, 0xWX, 0xYZ	Write slot and PD configuration for the two HR inputs to the HR algorithm WX is input 1 of the HR algorithm. W = 0 for Slot 1 W = 1 for Slot 2 W = 2 for Slot 3 W = 3 for Slot 4 W = 4 for Slot 5 W = 5 for Slot 6 W = 7 for Slot not used X = 0 for PD1 X = 1 for PD2 X = 3 for PD not used. YZ is input 2 of the HR algorithm. Y = 0 for Slot 1 Y = 1 for Slot 2 Y = 2 for Slot 3 Y = 3 for Slot 4 Y = 4 for Slot 5 Y = 5 for Slot 6 Y = 7 for Slot not used Z = 0 for PD1 Z = 1 for PD2 Z = 3 for PD not used.	-

			The LED # that is fired in each slot is defined in 0x19 command. For 32.9.33 (MAXM86161) the only acceptable setting is 0x0073 (default setting)	
0x51 for read	0x07	0x17	Read Slot and PD configuration for the two HR inputs to the HR algorithm. (Not implemented in 32.9.33 (MAXM86161)).	**0xWX 0xYZ** MAX86141 default: 0x0001 MAX86140, MAXM86146, MAXM86161 default: 0x0073 WX is the Slot, PD used for HR algorithm input 1: W = 0 for Slot 1 W = 1 for Slot 2 W = 2 for Slot 3 W = 3 for Slot 4 W = 4 for Slot 5 W = 5 for Slot 6 W = 7 for Slot not used X = 0 for PD1 X = 1 for PD2 X = 3 for PD not used. YZ is the Slot, PD used for HR algorithm input 2: HR algorithm. Y = 0 for Slot 1 Y = 1 for Slot 2 Y = 2 for Slot 3 Y = 3 for Slot 4 Y = 4 for Slot 5 Y = 5 for Slot 6 Y = 7 for Slot not used Z = 0 for PD1 Z = 1 for PD2 Z = 3 for PD not used
0x50 for write	0x07	0x18 0xWX 0xYZ	Write Slot and PD configuration for the IR, red inputs to the SpO_2 algoritm. WX is the LED/PD used for IR for the SpO_2 algorithm. W = 0 for Slot 1 W = 1 for Slot 2 W = 2 for Slot 3 W = 3 for Slot 4 W = 4 for Slot 5 W = 5 for Slot 6	-

			W = 7 for Slot not used X = 0 for PD1 X = 1 for PD2 X = 3 for PD not used. YZ is the LED/PD used for red for the SpO_2 algorithm. Y = 0 for Slot 1 Y = 1 for Slot 2 Y = 2 for Slot 3 Y = 3 for Slot 4 Y = 4 for Slot 5 Y = 5 for Slot 6 Y = 7 for Slot not used Z = 0 for PD1 Z = 1 for PD2 Z = 3 for PD not used	
0x51 for read	0x07	0x18	Read Slot and PD configuration for the IR, red inputs to the SpO_2 algorithm.	**0xWX 0xYZ** MAX86141/40 default: 0x1020 MAXM86146 default: 0x2111 WX is the LED/PD used for IR for the SpO_2 algorithm. W = 0 for Slot 1 W = 1 for Slot 2 W = 2 for Slot 3 W = 3 for Slot 4 W = 4 for Slot 5 W = 5 for Slot 6 W = 7 for Slot not used X = 0 for PD1 X = 1 for PD2 X = 3 for PD not used. YZ is the LED/PD used for red for the SpO_2 algorithm. Y = 0 for Slot 1 Y = 1 for Slot 2 Y = 2 for Slot 3 Y = 3 for Slot 4 Y = 4 for Slot 5 Y = 5 for Slot 6 Y = 7 for Slot not used Z = 0 for PD1 Z = 1 for PD2 Z = 3 for PD not used
0x50 for write	0x07	0x19 0xUV 0xWX 0xYZ (3x.12.x+)	Write slots used for LED firing sequence U is Slot 1 V is Slot 2	-

			W is Slot 3 X is Slot 4 Y is Slot 5 Z is Slot 6 U, V, W, X, Y, Z are defined as: 0: No LED firing 1: LED1 firing 2: LED2 firing 3: LED3 firing 4: LED4 firing 5: LED5 firing 6: LED6 firing 7: LED1 and LED2 firing 8: LED1 and LED3 firing 9: LED2 and LED3 firing LED firing sequence in firing slots 1-3 (slot 1-6 in case of MAXM86146) PPGs are reported in the same order. **The non-firing slots should appear to the end of sequence. Each LED can only be fired once (e.g., if one slot is set to LED1_AND_LED2, LED1 or LED 2, no other firing slot can be set to LED1)**	
0x51 for read	0x07	0x19	Read slots used for LED firing sequence.	**0xUV 0xWX 0xYZ** MAX86141/40, MAXM86161 default: 0x123000 MAXM86146 default: 0x123456 U is Slot 1 V is Slot 2 W is Slot 3 X is Slot 4 Y is Slot 5 Z is Slot 6 U, V, W, X, Y, Z are defined as: 0: No LED firing 1: LED1 firing 2: LED2 firing 3: LED3 firing 4: LED4 firing 5: LED5 firing 6: LED6 firing 7: LED1 and LED2 firing 8: LED1 and LED3 firing 9: LED2 and LED3 firing
0x50 for write	0x07	0x1A [byte]	Write initial integration time.(30.13.19+, 33.13.19+)	

			byte: **0x00**: 14.8µs **0x01**: 29.4µs **0x02**: 58.7µs **0x03**: 117.3µs	
0x51 for read	0x07	0x1A	Read initial integration time. (30.13.19+, 33.13.19+)	**0x00**: 14.8µs **0x01**: 29.4µs **0x02**: 58.7µs **0x03**: 117.3µs (default)
0x50 for write	0x07	0x1B [byte]	Write initial sampling rate and averaging. (30.13.19+, 33.13.19+) byte: 0x00: 25sps, avg = 1 0x01: 50sps, avg = 2 0x02: 100sps, avg = 4 0x03: 200sps, avg = 8 0x04: 400sps, avg = 16	
0x51 for read	0x07	0x1B	Read initial sampling rate and averaging. (30.13.19+, 33.13.19+)	**0x00**: 25sps, avg = 1 **0x01**: 50sps, avg = 2 (default) **0x02**: 100sps, avg = 4 (default) **0x03**: 200sps, avg = 8 **0x04**: 400sps, avg = 16
0x50 for write	0x07	0x1D [byte]	Write initial SpO_2 perfusion index threshold (red, IR). (30.13.19+, 33.13.19+) byte: 0x01 – 0xFF: 0.001 to 0.255	
0x51 for read	0x07	0x1D	Read initial SpO_2 perfusion index threshold (red, IR). (30.13.19+, 33.13.19+)	byte: 0x01 – 0xFF: 0.001 to 0.255 (default is 0.05, 0x32)
0x50 for write	0x07	0x20 [byte1] [byte2] [byte3]	Write Packed Normal output FIFO mask. Must be sent before enable algorithm command 0x52 0x07 0x04. At least one byte must be enabled. (33.13.31, 32.9.33) byte1, bit 8 (MSb): 0, 1 = disable/enable algorithm operation mode byte in the Packed Normal output. … byte3, bit 0 (LSb): 0, 1 = disable/enable IBI offset byte in the Packed Normal output.	
0x51 for read	0x07	0x20	Read Packed Normal output FIFO mask (33.13.31, 32.9.33) default: 0xFF 0xFF 0x00	[byte1] [byte2] [byte3] byte1, bit 8 (MSb): 0, 1 = algorithm operation mode byte is disabled/enabled in the Packed Normal output. …

				byte3, bit 0 (LSb): 0, 1 = IBI offset byte is disabled/enabled in the Packed Normal output.
0x52	0x07	**0x00**: Disable (CMD_DELAY = 120ms) **0x01**: Enable Normal Algorithm Samples Report (CMD_DELAY = 465ms) **0x02**: Enable Extended Algorithm Samples Report (CMD_DELAY = 465ms) **0x03**: Enable SCD Only Algorithm Samples Report (NA 32.9.x) (CMD_DELAY = 465ms) **0x04**: Enable Packed Normal Algorithm Samples Report (32.9.33) (CMD_DELAY = 465ms)	Write enable/disable the Wearable Algorithm Suite (WAS) (HR+SpO_2) algorithm. For further details, see I^2C Sensor Hub Output FIFO Samples Report Format.	
0x80	0x00	Use bytes 0x28 to 0x32 from the .msbl file as the IV bytes.	Bootloader mode flash the application .msbl: Set the initialization vector (IV) bytes.	
0x80	0x01	Use bytes 0x34 to 0x43 from the .msbl file.	Bootloader mode flash the application .msbl: Set the authentication bytes.	
0x80	0x02	0x00, Number of pages located at byte 0x44 from the .msbl file.	Bootloader mode flash the application .msbl: Set the number of pages to flash.	

0x80	0x03	- (CMD_DELAY = 1400ms)	Bootloader mode flash the application .msbl: Erase the application flash memory.	
0x80	0x04	The first page is specified by byte 0x4C from the .msbl file. The total bytes for each message protocol are the page size plus 16 bytes of CRC. (CMD_DELAY = 680ms)	Bootloader mode flash the application .msbl: Send the page values. Each page sent includes 16 CRC bytes for that page, so there are 8208 bytes per page sent in the payload of the message.	
0x81	0x00	-	Bootloader mode flash the application .msbl: Get bootloader version.	
0x81	0x01	-	Bootloader mode flash the application .msbl: Get the page size in bytes.	
0xB3 read			Read the sensor hub initial authentication vector for Maxim Wellness Library Suite	[ARRAY0] ARRAY0: six bytes of sensor hub initialization vector data which are inputs to mxm_algosuite_manager_getauthinitials()
0XB5 write	[ARRAY 1] twelve bytes of library public key which are output from mxm_algosuite_manager_getauthinitials()		Send the public key from the Maxim Wellness Library Suite to the sensor hub	
0xB4 read			Read the 12-byte sensor hub authentication public key for the Maxim Wellness Library Suite which will form the first 12 bytes of ARRAY2.	Twelve bytes of the authentication data

0xB2 read			Read the 32-byte sensor hub authentication public key for Maxim Wellness Library Suite which will form the last 32 bytes of ARRAY2.	
0xB8 write	0x01		Set the sensor hub to use MFIO to wake the host, MFIO Interrupt Mode.	
0xFF	0x03		Read the sensor hub version.	Major version byte, Minor version byte, Revision byte
0xFF	0x07		Read the algorithm: version. Deprecated - no longer supported.	Major version byte, Minor version byte, Revision byte. Deprecated

LEDs and PDs Configuration

The sensor hub firmware for MAX86141/40 and MAXM86161 use default LEDs and PDs configuration which are appropriate for the reference design. The user may change the configuration if other combinations of LEDs and PDs provide superior performance according to the hardware and optomechanical design. The following 3 steps are required to change LEDs and PDs configuration:

- *Map LEDs to configuration slots and firing order (Configuration index = 0x19):* The user needs to select which LED (or LEDs) are fired in each slot. The firing starts from LEDs selected in slot 1 and continues to slot 6. For example, if LED 1 is Green, LED 2 is IR and LED 3 is red, the firing sequence 0x123000 configures the firmware to fire them in that order. PPGs are also reported in the same order.
- *Map the slots and PDs for the two HR algorithm input channels (Configuration index = 0x17):* The user should configure which PD in which firing slot (as defined in A) is used as each of the two inputs of the algorithm. For example, 0x0001 means PD1 and PD2 of the LED fired in slot 1 (Green as configured in A) are selected for HR algorithm inputs.
- *Map the slots and PDs for the IR and Red SpO_2 algorithm input channels (Configuration index = 0x18):* The user should configure which PD in which firing slot (as defined in A) is used as two inputs of the algorithm. For example, 0x1020 means PD1 of the LEDs fired in slot 2 (IR as configured in A) and slot 3 (Red as configured in A) are selected for IR and Red inputs of SpO_2 algorithm.

Examples:

1) Configuration settings for two PDs, slot 1 green, slot 2 IR, slot 3 red (default for **MAXREFDES103 (MAX86141)**)
   0x19: 0x123000 [slot 1 use LED1 (green); slot 2 use LED2 (IR); slot 3 use LED3 (red)]
   0x17: 0x0001 [HR input 1 uses slot 1, PD1; HR input 2 uses slot 1, PD2]
   0x18: 0x1020 [SpO_2 IR input uses slot 2 PD1; SpO_2 red input uses slot 3 PD1]
2) Configuration settings for two PDs, slot 1 green, slot 2 Red, slot 3 IR (Host configures sensor hub for **Maslak D6W space optimized schematic, MAX86141**)
   0x19: 0x123000 [slot 1 use LED1 (green);slot 2 use LED2 (Red); slot 3 use LED3 (IR)]
   0x17: 0x0001 [HR input 1 uses slot 1, PD1; HR input 2 uses slot 1, PD2]
   0x18: 0x2010 [SpO_2 IR input uses slot 3 PD1; SpO_2 red input uses slot 2 PD1]
3) Configuration settings for one PD, slot 1 green, slot 2 IR, slot 3 red (default for **MAX86140, MAXM86161EVSYS**).
   0x19: 0x123000 [slot 1 use LED1 (green); slot 2 use LED2 (IR); slot 3 use LED3 (red)]
   (0x17: 0x0073) default: [HR Input 1 uses Slot 1 (green), PD1; HR Input 2 not used]
   0x18: 0x1020 [SpO_2 IR input uses slot 2 PD1; SpO_2 red input uses slot 3 PD1]
4) Configuration settings for LED1 green and LED2 green fired simultaneously using slot 1; LED4 and LED5 are IR and Red and are fired using slot 2 and 3; HR uses PD1 and PD2 when both LED1, LED2 are fired; $WSpO_2$ uses PD2 of LED4, LED5:
   0x19: 0x745000 [slot 1 uses LED1&2 (green) together; slot 2 uses LED4 (IR); slot 3

uses LED5 (red)]
0x17: 0x0001 [HR input 1 uses IR slot 1, PD1; HR input 2 uses slot 1, PD2]
0x18: 0x1121 [SpO_2 IR input uses slot 2 PD2; SpO_2 red input uses slot 3 PD2]

5) Configuration for one PD, no Green LED1; slot 2 IR, slot 3 red. (Host configures **MAXREFDES103** to the settings below)
0x19: 0x230000 [slot 1 use LED2 (IR); slot 2 use LED3 (red)]
0x17: 0x0073 [HR input 1 uses IR slot 1, PD1; HR Input 2 not used]
0x18: 0x0010 [SpO_2 IR input uses slot 1 PD1; SpO_2 red input uses slot 2 PD1]
6) Configuration settings for two PDs, slot 1 green2, slot 2 IR, slot 3 red (Host configures **MAXREFDES103** to the settings below)
0x19: 0x423000 [slot 2 use LED4 (green2); slot 2 use LED2 (IR); slot 3 use LED3 (red)]
0x17: 0x0001 [HR input 1 uses slot 1, PD1; HR input 2 uses slot 1, PD2]
0x18: 0x1020 [SpO_2 IR input uses slot 2 PD1; SpO_2 red input uses slot 3 PD1]
MAXM86146EVSYS configuration settings for two PDs, slot 1 green1, slot 2 green2, slot 3 red, slot 4 IR.
0x19: 0x135600 [slot 1 use LED1 (green1); slot 2 use LED3 (green2); slot 3 use LED5 (red); slot 4 LED6 (IR)]
0x17: 0x0011 [HR input 1 uses slot 1, PD1; HR input 2 uses slot 2, PD2]
0x18: 0x3020 [SpO_2 IR input uses slot 4 PD1; SpO_2 red input uses slot 3 PD1]
7) **MAXM86146EVSYS** configuration settings for two PDs, slot 1 green1, slot 2 red, slot 3 IR.
0x19: 0x156000 [slot 1 use LED1 (green1); slot 2 use LED5 (red); slot 3 LED6 (IR)]
0x17: 0x0001 [HR input 1 uses slot 1, PD1; HR input 2 uses slot 1, PD2]
0x18: 0x2010 [SpO_2 IR input uses slot 3 PD1; SpO_2 red input uses slot 2 PD1]
8) **MAXREFDES103(MAX86141)** Configuration settings for HR only two PDs, slot 1 green 0x19: 0x100000 [slot 1 use LED1 (green)]
0x17: 0x0001 [HR input 1 uses slot 1, PD1; HR input 2 uses slot 1, PD2]
0x18: 0x7373 [SpO_2 IR not used; SpO_2 red input not used]
Set the algorithm operating mode to continuous HR: 0x50 0x07 0x0A 0x02
9) **MAXM86146EVSYS** configuration settings for HR only: two PDs, slot 1 green1, slot 2 green2
0x19: 0x130000 [slot 1 use LED1 (green1); slot 2 use LED3 (green2)
0x17: 0x0011 [HR input 1 uses slot 1, PD1; HR input 2 uses slot 2, PD2]
0x18: 0x7373 [SpO_2 IR not used; SpO_2 red input not used]
Set the algorithm operating mode to continuous HR: 0x50 0x07 0x0A 0x02.

Modifying the LED, PD configurations will affect which LED counts will in the description column of the PPG raw data in samples report.

Reading FIFO Samples Report

I^2C Sensor Hub Output FIFO Samples Report Format

The sensor hub output FIFO samples report may contain Sample Counter byte and/or Sensor Data, and/or WAS Algorithm Data and/or. The inclusion of these categories of output data is controlled by the “Set the output format of the sensor hub samples report”, 0x10 0x00 command and the formats are listed below (The initial response byte, “Read Status Byte” is not shown).

Table 3. Sample Counter Byte Output FIFO Samples Report Format

DATA ITEM	# OF BYTES (MSB FIRST)	DESCRIPTION
Sample Counter byte	1	Counter which cycles from 0-0xFF.

Table 4. Sensor Data Output FIFO Samples Report Format MAXM86141/40, MAXM86161

DATA SOURCE (24 bytes total)	DATA ITEM	# OF BYTES (MSB FIRST)	DESCRIPTION
MAX86141/40, MAXM86161 PPG Data (18 Bytes)	PPG1 (PD1)	3	Green1 counts
	PPG2 (PD1)	3	IR LED counts
	PPG3 (PD1)	3	Red LED counts
	PPG4 (PD2)	3	Green2 counts
	PPG5 (PD2)	3	N/A
	PPG6 (PD2)	3	N/A
Accelerometer (6 Bytes)*	accelX	2	Two's complement. lsb = 0.001g
	accelY	2	Two's complement. lsb = 0.001g
	accelZ	2	Two's complement. lsb = 0.001g

Table 5. Sensor Data Output FIFO Samples Report Format MAXM86146

DATA SOURCE (42 bytes total)	DATA ITEM	# OF BYTES (MSB FIRST)	DESCRIPTION
MAXM86146 PPG Data (36 Bytes)	PPG1 (PD1)	3	Green counts
	PPG2 (PD1)	3	N/A
	PPG3 (PD1)	3	N/A
	PPG4 (PD1)	3	N/A
	PPG5 (PD1)	3	N/A
	PPG6 (PD1)	3	N/A
	PPG7 (PD2)	3	N/A
	PPG8 (PD2)	3	Red counts
	PPG9 (PD2)	3	IR counts
	PPG10 (PD2)	3	N/A
	PPG11 (PD2)	3	N/A
	PPG12 (PD2)	3	N/A
Accelerometer (6 Bytes)	accelX	2	Two's complement. lsb = 0.001g
	accelY	2	Two's complement. lsb = 0.001g
	accelZ	2	Two's complement. lsb = 0.001g

Table 6. Normal Algorithm Data Output FIFO Samples Report Format

DATA ITEM (24 bytes total) (32.9.x, 33.13.31: 20 bytes)	# OF BYTES (MSB FIRST)	DESCRIPTION
Algorithm operation mode	1	Algorithm operation mode: 0: Continuous HR and Continuous SpO_2 1: Continuous HR and One-Shot SpO_2 2: Continuous HR 3: Sampled HR

		4: Sampled HR and One-Shot SpO_2 5: Activity tracking (applicable to wrist form factor only) 6: SpO_2 Calibration Data Collection
HR	2	10x Calculated heart rate
HR confidence	1	Confidence level in %, >40 is for consumer devices, >80,90 is for medical devices.
RtoR	2	10x RtoR – inter-beat interval in ms Only shows a nonzero value when a new value is calculated.
RtoR confidence	1	Calculated confidence level of RtoR in % Only shows a nonzero value when a new value is calculated.
Activity class	1	Activity class (applicable to wrist form factor only) 0: Rest 1: Other 2: Walk 3: Run 4: Bike
R	2	1000x Calculated SpO_2 R value
SpO_2 confidence	1	SpO_2 confidence level in %, >40 is for consumer devices, >80,90 is for medical devices.
SpO_2	2	10x Calculated SpO_2 %
SpO_2 valid, % complete	1	Calculation progress in % in one-shot mode of algorithm. In continuous mode, it is reported as zero and only jumps to 100 when the SpO_2 value is updated. Bit[7]: SpO_2 valid Bit[6..0]: Percent complete
SpO_2 low signal quality flag	1	Shows the low quality of the PPG signal: 0: Good quality 1: Low quality
SpO_2 motion flag	1	Shows excessive motion: 0: No motion 1: Excessive motion
SpO_2 low PI flag	1	Shows the low perfusion index (PI) of the PPG signal: 0: Normal PI 1: Low PI
SpO_2 unreliable R flag	1	Shows the reliability of R: 0: Reliable 1: Unreliable
SpO_2 state	1	Reported status of the SpO_2 algorithm: 0: LED adjustment 1: Computation 2: Success 3: Timeout
SCD state	1	Skin contact state: 0: Undetected 1: Off skin 2: On some subject 3: On skin
IBI Offset (Not sent in 32.9.x, 33.13.31)	1	Reported when IBI is calculated. Defines number of samples between current algo sample and previous algo sample where IBI is calculated (for Maxim Wellness library)
Unreliable	1	Flag reporting not appropriate orientation of wrist for reliable SpO_2 measurement (for Maxim Wellness library, sleep quality) (the third accelerometer axis (accel[2]), should be the

orientation flag (Not sent in 32.9.x, 33.13.31)		axis that is aligned with the gravity vector for the orientation detection algo) 0: correct orientation 1: wrong orientation
RESERVED (Not sent in 32.9.x, 33.13.31)	2	Reserved for future use

Table 7. Extended Algorithm Data Output FIFO Samples Report Format

DATA ITEM (56 bytes total) (32.9.x, 33.13.31: 52 bytes)	# OF BYTES (MSB FIRST)	DESCRIPTION
Algorithm operation mode	1	Algorithm operation mode: 0: Continuous HR and Continuous SpO_2 1: Continuous HR and One-Shot SpO_2 2: Continuous HR 3: Sampled HR 4: Sampled HR and One-Shot SpO_2 5: Activity tracking (applicable to wrist form factor only) 6: SpO_2 Calibration Data Collection
HR	2	10x Calculated heart rate
HR confidence	1	Confidence level in %, >40 is for consumer devices, >80,90 is for medical devices.
RtoR	2	10x RtoR – inter-beat interval in ms Only shows a nonzero value when a new value is calculated.
RtoR confidence	1	Calculated confidence level of RtoR in % Only shows a nonzero value when a new value is calculated.
Activity class	1	Activity class (applicable to wrist form factor only) 0: Rest 1: Other 2: Walk 3: Run 4: Bike
Total walk steps	4	Total number of walking steps since the last reset
Total run steps	4	Total number of running steps since the last reset. (applicable to wrist form factor only)
Total energy exp in kcal	4	10x total energy expenditure since the last reset in kcal. (applicable to wrist form factor only)
Total AMR in kcal	4	10x total active energy expenditure since the last reset in kcal. (applicable to wrist form factor only)
Is LED current adjustment requested in first time slot	1	Flag to notify if the LED current adjustment is requested or not in the first time slot
Adjusted LED current in first time slot	2	10x value of the adjusted LED current (mA) in the first time slot, valid only if “Is LED current adjustment requested in first time slot” flag is true
Is LED current adjustment requested in second time slot	1	Flag to notify if the LED current adjustment is requested or not in the second time slot

Adjusted LED current in second time slot	2	10x value of the adjusted LED current (mA) in the second time slot, valid only if the "Is LED current adjustment requested in second time slot" flag is true
Is LED current adjustment requested in third time slot	1	Flag to notify if the LED current adjustment is requested or not in the third time slot
Adjusted LED current in third time slot	2	10x value of the adjusted LED current (mA) in third time slot, valid only if the "Is LED current adjustment requested in third time slot" flag is true
Is integration time adjustment requested	1	Flsag to notify if the integration time adjustment is requested or not
Requested integration time	1	Value of the requested integration time option, valid only if the "Is integration time adjustment requested" flag is true. 0 = 14.8µs 1 = 29.4µs 2 = 58.7µs 3 = 117.3µs
Is sampling rate adjustment requested	1	Flag to notify if the sampling rate adjustment is requested or not
Requested sampling rate	1	Value of the requested sampling rate option, valid only if the "Is sampling rate adjustment requested" flag is true. 0: 25 samples/second, average 1 1: 50 samples/second, average 2 2: 100 samples/second, average 4 3: 200 samples/second, average 8 4: 400 samples/second, average 16
Requested sampling rate	1	Value of the requested sampling rate option, valid only if the "Is sampling rate adjustment requested" flag is true. 0: 25 samples/second, average 1 1: 50 samples/second, average 2 2: 100 samples/second, average 4 3: 200 samples/second, average 8 4: 400 samples/second, average 16
AFE controller state for HR channels	1	State of the AFE manager (for HR channels)
Is high motion for HR	1	Flag to notify if the motion is considered high for heart rate measurement
SCD state	1	Skin contact state: 0: Undetected 1: Off skin 2: On some subject 3: On skin
R	2	1000x Calculated SpO_2 R value
SpO_2 confidence	1	SpO_2 confidence level in %, >40 is for consumer devices, >80,90 is for medical devices.
SpO_2	2	10x Calculated SpO_2 %
SpO_2 valid, % complete	1	Calculation progress in % in one-shot mode of algorithm. In continuous mode, it is reported as zero and only jumps to 100 when the SpO_2 value is updated. Bit[7]: SpO_2 valid

		Bit[6..0]: Percent complete
SpO_2 low signal quality flag	1	Shows the low quality of the PPG signal: 0: Good quality 1: Low quality
SpO_2 motion flag	1	Shows excessive motion: 0: No motion 1: Excessive motion
SpO_2 low PI flag	1	Shows the low perfusion index (PI) of the PPG signal: 0: Normal PI 1: Low PI
SpO_2 unreliable R flag	1	Shows the reliability of R: 0: Reliable 1: Unreliable
SpO_2 state	1	Reported status of the SpO_2algorithm: 0: LED adjustment 1: Computation 2: Success 3: Timeout
IBI Offset (Not sent in 32.9.x, 33.13.31)	1	Reported when IBI is calculated. Defines number of samples between current algo sample and previous algo sample where IBI is calculated (for Maxim Wellness library)
Unreliable orientation flag (Not sent in 32.9.x, 33.13.31)	1	Flag reporting not appropriate orientation of wrist for reliable SpO_2 measurement (for Maxim Wellness library, sleep quality) (the third accelerometer axis (accel[2]), should be the axis that is aligned with the gravity vector for the orientation detection algo) 0: correct orientation 1: wrong orientation
RESERVED (Not sent in 32.9.x, 33.13.31)	2	Reserved for future use

Table 8. Packed Normal Algorithm Data Output FIFO Samples Report Format

DATA ITEM (16 bytes total) Available in 32.9.33 33.9.31	# OF BYTES (MSB FIRST)	DESCRIPTION
Algorithm operation mode	1	Algorithm operation mode: 0: Continuous HR and Continuous SpO_2 1: Continuous HR and One-Shot SpO_2 2: Continuous HR 3: Sampled HR 4: Sampled HR and One-Shot SpO_2 5: Activity tracking (applicable to wrist form factor only) 6: SpO_2 Calibration Data Collection
HR	2	10x Calculated heart rate
HR confidence	1	Confidence level in %, >40 is for consumer devices, >80,90 is for medical devices.

RtoR	2	10x RtoR – inter-beat interval in ms Only shows a nonzero value when a new value is calculated.
RtoR confidence	1	Calculated confidence level of RtoR in % Only shows a nonzero value when a new value is calculated.
R	2	1000x Calculated SpO_2 R value
SpO_2 confidence	1	SpO_2 confidence level in %, >40 is for consumer devices, >80,90 is for medical devices.
SpO_2	2	10x Calculated SpO_2 %
SpO_2 valid, % complete	1	Calculation progress in % in one-shot mode of algorithm. In continuous mode, it is reported as zero and only jumps to 100 when the SpO_2 value is updated. Bit[7]: SpO_2 valid Bit[6..0]: Percent complete
SpO_2 Discrete1	1	Bit 0: SpO_2 low signal quality flag 0: Good quality 1: Low quality Bit 1: SpO_2 excessive motion flag 0: No motion 1: Excessive motion Bit 2: SpO_2 low perfusion index (PI) flag 0: Normal PI 1: Low PI Bit 3: SpO_2 unreliable R flag 0: Reliable R 1: Unreliable R Bit 4,5: SpO_2 state 0: LED adjustment 1: Computation 2: Success 3: Timeout
Discrete2	1	Bit 0, 1: Skin Contact Detection (SCD) state 0: Undetected 1: Off skin 2: On some subject 3: On skin Bit 2, 3, 4: Activity class (applicable to wrist form factor only) 0: Rest 1: Other 2: Walk 3: Run 4: Bike Bit 5: Flag reporting not appropriate orientation of wrist for reliable SpO_2 measurement (for Maxim Wellness library, sleep quality) (the third accelerometer axis (accel[2]), should be the axis that is aligned with the gravity vector for the orientation detection algo) 0: correct orientation 1: wrong orientation
IBI Offset	1	Reported when IBI is calculated. Defines number of samples between current algo sample and previous algo sample where IBI is calculated (for Maxim Wellness library)

Heart Rate, SpO_2, AEC, SCD Report

Automatic Exposure Control (AEC) is Maxim's gain control algorithm that is superior to AGC. The AEC algorithm optimally maintains the best SNR range and power optimization. The targeted SNR range is maintained regardless of skin color or ambient temperature within the limits of the LED currents configurations; The AEC dynamically manages the appropriate register settings for sampling rate, LED current, pulse width and integration time. SpO_2 requires higher sensitivity, so the AEC controls for sampling rate, pulse width and integration time are not active for modes that output SpO_2 data/ AEC controls for sampling rate, pulse width and integration time is only active for HR only operational modes and AEC will only actively control the LED current for SpO_2 operational modes. If the host is trying to optimize power in this manner, then it's recommended that continuous HR operation mode be used and then continuous HR+One-Shot SpO_2 operation mode be used every couple of minutes (or adjusted per user setting).

In the example below, both the AEC and SCD are enabled. The algorithm mode of operation can be selected as described in previous section. The sequence of commands is shown in the table below.

Table 9. Host Commands: HR, SpO_2, AEC, SCD Algorithm Report

	#	HOST COMMAND (HEX)	COMMAND DESCRIPTION	RESPONSE (HEX)
START ALGORITHM	Host initializes the sensor hub in AEC-SCD mode using the following commands:			
	1.0	AA 02 00 (optional)	Read the device operating mode	AB 00 00 application mode
	1.1	AA 50 07 00 [00000000FFD7FBDD00AB61FE] (default, optional)	Bracketed SpO_2 coefficients are for example only. Values used in the final form factor may be different per section 2.	AB 00
	1.2	AA 10 00 03 (default, optional)	Set the output format to sensor + algorithm data (streamed data will include PPG, accelerometer, and algorithm data).	AB 00
	1.3	AA 10 01 01 (default, optional)	Set the sensor hub DataRdyInt threshold.	AB 00
	1.4	AA 10 02 01 (default, optional)	Set the samples report period to 40ms. Samples report rate to be one report per every sensor sample.	AB 00
	1.5	AA 50 07 0A 00 (default, optional)	Set the algorithm operation mode to Continuous HRM and Continuous SpO_2.	AB 00
	1.6	AA 50 07 0B 01 (default, optional)	Enable AEC [enabled by default].	AB 00
	1.7	AA 50 07 12 01 (default, optional)	Enable Auto PD Current Calculation [enabled by default].	AB 00
	1.8	AA 50 07 0C 01 (default, optional)	Enable SCD [enabled by default].	AB 00
	Optional: Any command to change the algorithm settings and configurations from default should appear here before enabling the algorithm.			
	1.9	AA FF 03	Read the sensor hub version for [MAX86141/MAX86140/MAXM86146/MAXM86161] (30.x.y/30.x.y/33.x.y/32.x.y)	AB 00 [1E/1E/21/20] XX YY
	1.10	AA 50 07 20 7F FF 00 (optional)(32.9.33, 33.13.31)	Mask out the first byte of the Packed Algorithm Data Output FIFO Samples Report so that it's not included int the Packed Algorithm Data Output FIFO Samples Report.	AB 00

	For 33.x.y (MAXM86146), configure HR inputs to use Green1 and Green2; configure SpO_2inputs to use IR, red. See "LEDs and PDs Configuration" for other example configurations.			
	1.20	For 33.x.y (MAXM86146) AA 50 07 19 13 56 00	Map configuration slot 1 to use LED1 (green1); map slot 2 to use LED3 (green2); map slot 3 to use LED5 (red); map slot 4 to use LED6 (IR)]	AB 00
	1.21	For 33.x.y (MAXM86146) AA 50 07 17 00 11	Map HR input 1 to use slot 1, PD1; map HR input 2 to use slot 2, PD2	AB 00
	1.22	For 33.x.y (MAXM86146) AA 50 07 18 30 20	Map SpO_2IR input to use slot 4 PD1; map SpO_2 red input to use slot 3 PD1	AB 00
	1.30	AA 52 07 01 (for normal algorithm report, CMD_DELAY = 465ms) AA 52 07 04 (for packed normal algorithm report, CMD_DELAY = 465ms, 32.9.33, 33.13.31) AA 52 07 02 (for extended algorithm report, CMD_DELAY = 465ms)	Enable the HR and SpO_2 algorithm; the analog front-end (AFE) and sensor hub accelerometer will be enabled automatically. Note: If the accel is not connected to the sensor hub and if host accel data is not configured and provided, the algorithm will not produce the samples report - Accel data is required.	AB 00
	1.31	AA 41 00 FF (optional sanity check)	Read register FF (PART_ID) of [MAX86141/MAX86140//MAXM86161]	AB 00 [25/24/36]
	1.32	AA 41 04 0F (optional sanity check) (accel data is required from either host or KX122/KX11S/LIS2DS12I)	Read register 0F (WHO_AM_I) of [KX122/KX112/LIS2DS12] if accel is connected to sensor hub.	AB 00 [1B/22/43]
READING SAMPLES REPORT IN OUTPUT FIFO	Host reads samples periodically (do not execute at a faster rate than the samples report period):			
	2.1	AA 00 00 (optional)	Read sensor hub status byte: Bit 0: Sensor comm error Bits 1 and 2: Reserved Bit 3: FIFO filled to threshold (DataRdyInt) Bit 4: Output FIFO overflow (FifoOutOvrInt) Bit 5: Input FIFO overflow (FifoInOverInt) Bit 6: Host input FIFO underflow (HostAccelUflint) Bit 7: SCDSM detected skin contact (SCDMonSkin) If DataRdyInt is set, proceed to next step.	AB 00 08
	2.2	AA 12 00 (optional, but recommended every 25th iteration)	Get the number of samples (nn) in the FIFO.	AB 00 nn
	2.3	AA 12 01	Read the data stored in the FIFO; nn sample reports will be read. Samples report = [Sensor Data Output FIFO Samples Report] [Normal (or Extended or Packed Normal (less the bytes masked in step 1.10)) Algorithm Data Output FIFO Samples Report]	AB 00 data_for_ nn_sample_repo rts
S T	Host disables sensors and algorithm:			

	3.1	AA 44 00 00 (CMD_DELAY = 465ms)	Disable the AFE (e.g., MAX86141)	AB 00
	3.2	AA 44 04 00 (CMD_DELAY = 20ms)	Disable the accelerometer.	AB 00
	3.3	AA 52 07 00 (CMD_DELAY = 120ms)	Disable the Wearable Algorithm Suite (HR+ SpO_2) algorithm	AB 00

Heart Rate, SpO_2, AGC Report

In this configuration, the wearable algorithm suite (HR, SpO_2) is enabled and the R value, SpO_2, SpO_2 confidence level, heart rate, heart rate confidence level, RtoR value, and activity class are reported. Furthermore, automatic gain control (AGC) is enabled. Because AGC is a subset of AEC functionality, to enable AGC, AEC still needs to be enabled. However, automatic calculation of target PD should be turned off, and the desired level of AGC target PD current is set by the user.

Table 10. Host Commands: HR, SpO_2, AGC Normal Algorithm Report

	#	HOST COMMAND (HEX)	COMMAND DESCRIPTION	RESPONSE (HEX)
START ALGORITHM	Host initializes the sensor hub in AGC mode using the following commands:			
	1.0	AA 02 00 (optional)	Read the device operating mode	AB 00 00 application mode
	1.1	AA 50 07 00 [00000000FFD7FBDD00AB61FE] (default, optional)	Bracketed SpO_2 coefficients are for example only. Values used in the final form factor may be different per section 2.	AB 00
	1.2	AA 10 00 03 (default, optional)	Set the output format to sensor + algorithm data (streamed data will include PPG, accelerometer, and algorithm data).	AB 00
	1.3	AA 10 01 01 (default, optional)	Set the sensor hub DataRdyInt threshold.	AB 00
	1.4	AA 10 02 01 (default, optional)	Set the samples report period to 40ms. Samples report rate to be one report per every sensor sample.	AB 00
	1.5	AA 50 07 0A 00 (default, optional)	Set the algorithm operation mode to Continuous HRM and Continuous SpO_2.	AB 00
	1.6	AA 50 07 0B 01 (default, optional)	Enable AEC [enabled by default].	AB 00
	1.7	AA 50 07 12 00	Disable Auto PD Current Calculation.	AB 00
	1.8	AA 50 07 0C 00	Disable SCD.	AB 00
	1.9	AA 50 07 11 00 64	Set AGC Target PD Current to 10µA.	
	Optional: Any command to change the algorithm settings and configurations from default should appear here before enabling the algorithm.			
	1.10	AA FF 03 (optional)	Read the sensor hub version for [MAX86141/MAX86140/MAXM86146/MAXM86161] (30.x.y/30.x.y/33.x.y/32.x.y)	AB 00 [1E/1E/21/20] XX YY
	For 33.x.y (MAXM86146), configure HR inputs to use Green1 and Green2; configure SpO_2 inputs to use IR, red. See "LEDs and PDs Configuration" for other example configurations.			
	1.20	For 33.x.y (MAXM86146) AA 50 07 19 13 56 00	For 33.x.y (MAXM86146) AA 50 07 19 13 56 00	AB 00
	1.21	For 33.x.y (MAXM86146)	For 33.x.y (MAXM86146)	AB 00

	1.22	AA 50 07 17 00 11 For 33.x.y (MAXM86146) AA 50 07 18 30 20	AA 50 07 17 00 11 For 33.x.y (MAXM86146) AA 50 07 18 30 20	AB 00
	1.30	AA 52 07 01 (for normal algorithm report, CMD_DELAY = 465ms) AA 52 07 02 (for extended algorithm report, CMD_DELAY = 465ms) AA 52 07 04 (for packed normal algorithm report, CMD_DELAY = 465ms, 32.9.33, 33.13.31)	Enable the HR and SpO_2 algorithm; the analog front-end (AFE) and sensor hub accelerometer will be enabled automatically. Note: If the accel is not connected to the sensor hub and if host accel data is not configured and provided, the algorithm will not produce the samples report – Accel data is required.	AB 00
	1.31	AA 41 00 FF (optional sanity check)	Read register FF (PART_ID) of [MAX86141/MAX86140//MAXM86161]	AB 00 [25/24/36]
	1.32	AA 41 04 0F (optional sanity check if accel is connected to sensor hub)	Read register 0F (WHO_AM_I) of [KX122/KX112/LIS2DS12] if accel is connected to sensor hub	AB 00 [1B/22/43]
READING SAMPLES REPORT IN OUTPUT FIFO	Host reads samples periodically (do not execute at a faster rate than the samples report period):			
	2.1	AA 00 00 (optional)	Read sensor hub status byte: Bit 0: Sensor comm error Bits 1 and 2: Reserved Bit 3: FIFO filled to threshold (DataRdyInt) Bit 4: Output FIFO overflow (FifoOutOvrInt) Bit 5: Input FIFO overflow (FifoInOverInt) Bit 6: Host input FIFO underflow (HostAccelUflint) Bit 7: SCDSM detected skin contact (SCDMonSkin) If DataRdyInt is set, proceed to next step.	AB 00 08
	2.2	AA 12 00 (optional, but recommended every 25th iteration)	Get the number of samples (nn) in the FIFO.	AB 00 nn
	2.3	AA 12 01	Read the data stored in the FIFO; nn sample reports will be read. Samples report = [Sensor Data Output FIFO Samples Report] [Normal (or Extended or Packed Normal) Algorithm Data Output FIFO Samples Report]	AB 00 data_for_ nn_sample_repo rts
STOP	Host disables sensors and algorithm:			
	3.1	AA 44 00 00 (CMD_DELAY = 465ms)	Disable the AFE (e.g., MAX86141)	AB 00
	3.2	AA 44 04 00 (CMD_DELAY = 20ms)	Disable the accelerometer.	AB 00
	3.3	AA 52 07 00 (CMD_DELAY = 120ms)	Disable the Wearable Algorithm Suite (HR+ SpO_2) algorithm	AB 00

SpO_2 Calibration Data Report

Due to variations in the physical design and optical cover lens of the final product, a calibration data collection procedure for SpO_2 is required to be performed once in a controlled environment. This procedure is important to ensure the quality of the SpO_2 calculation. This step is typically performed in a standard lab using the final form factor (with cover lens) with a reference SpO_2 device to determine three SpO_2 calibration coefficients: a, b, and c. The details of the SpO_2 calibration data collection and SpO_2 coefficient derivation procedure are described in the Guidelines for SpO_2 Measurement Using the Maxim MAX32664 Sensor Hub application note.

Once the three SpO_2 calibrations coefficients are obtained, they need to be loaded to the sensor hub every time prior to starting the algorithm.

The SpO_2 calibrations coefficients need to be converted to a 32-bit integer format using the following:

- A_{int32} = round (10^5 x a)
- B_{int32} = round (10^5 x b)
- C_{int32} = round (10^5 x c)

For example, the default measured SpO_2calibration coefficients (32-bit signed integer) for

- A_{int32} = round (10^5 x a) = 0x00000000 (0.0)
- B_{int32} = round (10^5 x b) = 0xFFD7FBDD (-26.224999)
- C_{int32} = round (10^5 x c) = 0x00AB61FE (112.317421)

The SpO_2 calibration coefficients may be stored in the host flash separately and loaded to the sensor hub after every reset.

The table below shows the sequence of commands for the enabling the SpO_2data collection report.

Table 11. Host Commands: SpO_2 Calibration Data Collection Report

	#	HOST COMMAND (HEX)	COMMAND DESCRIPTION	RESPONSE (HEX)
START ALGORITHM	Host initializes the sensor hub to SpO_2 calibration data collection mode and starts the algorithm using following commands:			
	1.0	AA 02 00 (optional)	Read the device operating mode	AB 00 00 application mode
	1.1	AA 10 00 03	Set the output format to sensor + algorithm data (streamed data will include PPG, accelerometer, and algorithm data).	AB 00
	1.2	AA 10 01 01	Set the sensor hub DataRdyInt threshold.	AB 00
	1.3	AA 10 02 01	Set the samples report period to 40ms. Samples report rate to be one report per every sensor sample.	AB 00
	1.4	AA 50 07 0A 06	Set the mode to SpO_2 Calibration Data Collection mode	AB 00
	Optional: Any command to change the algorithm settings and configurations from the default setting should appear here BEFORE enabling the algorithm.			
	1.5	AA FF 03 (optional)	Read the sensor hub version for [MAX86141/MAX86140/MAXM86146/MAXM86161] (30.x.y/30.x.y/33.x.y/32.x.y)	AB 00 [1E/1E/21/20] XX YY

	1.6	AA 52 07 01 (CMD_DELAY = 465ms)	Enable the HR and SpO_2 algorithm; the analog front-end (AFE) and sensor hub accelerometer will be enabled automatically.	AB 00
	1.7	AA 41 00 FF (optional sanity check)	Read register FF (PART_ID) of [MAX86141/MAX86140//MAXM86161]	AB 00 [25/24/36]
	1.8	AA 41 04 0F (optional sanity check if accel is connected to sensor hub)	Read register 0F (WHO_AM_I) of [KX122/KX112/LIS2DS12] if accel is connected to sensor hub.	AB 00 [1B/22/43]
READING SAMPLES REPORT IN OUTPUT FIFO	Host reads samples periodically (do not execute at a faster rate than the samples report period):			
	2.1	AA 00 00 (optional)	Read sensor hub status byte: Bit 0: Sensor comm error Bits 1 and 2: Reserved Bit 3: FIFO filled to threshold (DataRdyInt) Bit 4: Output FIFO overflow (FifoOutOvrInt) Bit 5: Input FIFO overflow (FifoInOverInt) Bit 6: Host input FIFO underflow (HostAccelUflint) Bit 7: SCDSM detected skin contact (SCDMonSkin) If DataRdyInt is set, proceed to next step.	AB 00 08
	2.2	AA 12 00 (optional, but recommended every 25th iteration)	Get the number of samples (nn) in the FIFO.	AB 00 nn
	2.3	AA 12 01	Read the data stored in the FIFO; nn sample reports will be read. Samples report = [Sensor Data Output FIFO Samples Report] [Normal Algorithm Data Output FIFO Samples Report]	AB 00 data_for_nn_sample_reports
STOP	Host disables sensors and algorithm:			
	3.1	AA 44 00 00 (CMD_DELAY = 465ms)	Disable the AFE (e.g., MAX86141)	AB 00
	3.2	AA 44 04 00 (CMD_DELAY = 20ms)	Disable the accelerometer.	AB 00
	3.3	AA 52 07 00 (CMD_DELAY = 120ms)	Disable the Wearable Algorithm Suite (HR+ SpO_2) algorithm	AB 00

PPG Raw Report Configuration

The host may configure the sensor hub to output raw rata (no algorithm) by enabling the AFE. The table below lists the set of commands that are needed to obtain the PPG raw data.

Table 12. Host Commands: Raw Data Report

	#	HOST COMMAND (HEX)	COMMAND DESCRIPTION	RESPONSE (HEX)
START ALGORITH	Host initializes the sensor hub in Raw Data mode using following commands:			
	1.0	AA 02 00 (optional)	Read the device operating mode	AB 00 00 application mode
	1.1	AA FF 03 (optional)	Read the sensor hub version for [MAX86141/MAX86140/MAXM86146/MAXM86161] (30.x.y/30.x.y/33.x.y/32.x.y)	AB 00 [1E/1E/21/20] XX YY

	1.2	AA 10 00 01	Set the output format to Sensor data only.	AB 00
	1.3	AA 10 01 01	Set the sensor hub DataRdyInt threshold.	AB 00
	1.4	AA 44 04 01 00 (if sensor hub accelerometer is used, CMD_DELAY = 20ms)	Enable the accelerometer.	AB 00
	1.5	AA 44 00 01 (CMD_DELAY = 465ms)	Enable AFE (MAX86141/40, MAXM86146, or MAXM86161).*	AB 00
	Any command to change the sensor registers should appear AFTER enabling the sensor or they will be overwritten. By default, the algorithm sets the following AFE registers: Sample rate: 100Hz, 1-sample averaging (samples report period: 10 msec) Integration time: 117µs ADCs 1 and 2 range: 32µA LEDs 1, 2, and 3 full range: 124mA			
	1.6	AA 41 00 FF (optional sanity check)	Read register FF (PART_ID) of [MAX86141/MAX86140//MAXM86161]	**AB 00 [25/24/36]**
	1.7	AA 41 04 0F (optional sanity check if accel is connected to sensor hub)	Read register 0F (WHO_AM_I) of [KX122/KX112/LIS2DS12] if accel is connected to sensor hub.	**AB 00 [1B/22/43]**
	1.8	AA 40 00 12 18	Set the sample rate of the MAX86141 to 100Hz with 1-sample averaging by modifying register 0x12 of the AFE. (100Hz is the maximum for 3x.12.x or earlier versions).	AB 00
		AA 40 00 12 20	Set the sample rate of the MAX86141 to 200Hz with 1-sample averaging by modifying register 0x12 of the AFE. (200 Hz is the maximum for 3x.13.x+).	
	1.9	AA 40 00 23 7F	Set the MAX86141 LED1 current to half of full scale. Reduce [7F] if the signal is saturated.	AB 00
	1.10	AA 40 00 24 7F	Set the MAX86141 LED2 current to half of full scale. Reduce [7F] if the signal is saturated.	AB 00
	1.11	AA 40 00 25 7F	Set the MAX86141 LED3 current to half of full scale. Reduce [7F] if the signal is saturated.	AB 00
READING	Host reads samples periodically (do not execute at a faster rate than the samples report period):			
	2.1	AA 00 00 (optional)	Read sensor hub status byte: Bit 0: Sensor comm error Bits 1 and 2: Reserved	AB 00 08

			Bit 3: FIFO filled to threshold (DataRdyInt) Bit 4: Output FIFO overflow (FifoOutOvrInt) Bit 5: Input FIFO overflow (FifoInOverInt) Bit 6: Host input FIFO underflow (HostAccelUflint) Bit 7: SCDSM detected skin contact (SCDMonSkin) If DataRdyInt is set, proceed to next step.	
	2.2	AA 12 00 (optional, but recommended every 25th iteration)	Get the number of samples (nn) in the FIFO.	AB 00 nn
	2.3	AA 12 01	Read the data stored in the FIFO; nn sample reports will be read. Samples report = [Sensor Data Output FIFO Samples Report]	AB 00 data_for_nn_sample_reports
STOP	Host disables sensors and algorithm:			
	3.1	AA 44 00 00 (CMD_DELAY = 465ms)	Disable the AFE (e.g., MAX86141)	AB 00
	3.2	AA 44 04 00 (CMD_DELAY = 20ms)	Disable the accelerometer.	AB 00

Accelerometer

The sensor hub requires accelerometer data to function properly. An accelerometer is mandatory for a heart rate monitor to be able to compensate for the user's motion. Otherwise, the reported heart rate will not be correct during movement.

SpO_2 calculation requires a resting condition, and the algorithm uses accelerometer data to detect excessive motion. In such a condition, computation is paused, and the user is informed with a motion flag.

Sensor Hub Accelerometer

A sensor hub accelerometer can be integrated through the SPI port of the sensor hub. In this case, the required driver for KX122/KX112/LIS2DS12 is included in the latest application firmware. The user only needs to follow the reference schematics to connect the accelerometer and enable it before starting the algorithm, as described later in this document. Normally, the accelerometer is polled to collect samples. The interrupt line is only needed if the SCD-based power saving procedure is implemented in the host.

Host Accelerometer

Alternatively, a host-side accelerometer can be used. However, this option requires strict timing synchronization between the sampled accelerometer data and PPG samples of ±40ms or less. To use the host-side accelerometer:

- The host should start the accelerometer just before enabling the algorithm to maximize the initial synchronization between the PPG and accelerometer samples. However, accelerometer samples collected prior to receiving the confirmation of the algorithm enable I^2C command should be discarded.
- The host is required to use a 3-axis accelerometer at a 25Hz sampling rate. If a higher sampling rate is chosen, samples should be decimated to be synchronized with a 40ms PPG sampling time.

- The host must queue five accelerometer samples and feed them at the same time to the sensor hub using the commands shown in the table below. The period of feeding samples should be 200ms. This is the longest delay that the sensor hub can tolerate to receive accelerometer samples.

Because the AFE sensor and the host accelerometer use different clock sources, exact synchronization between them is not possible. The sensor hub internally decimates or interpolates accelerometer samples to compensate for drift.

Piping in Host Accelerometer Data to the Sensor Hub for Heart Rate, SpO_2,, AEC, SCD Report

The table below provides the sequence of commands for writing external (host connected) accelerometer data to the input FIFO of the sensor hub; The KX122/LIS2DS12 is not connected to the connected to the MAX32664 sensor hub.

Table 13. Sequence of Commands to Write External Accelerometer Data to the Input FIFO for HR, SpO_2, AEC, SCD Report

	#	HOST COMMAND (HEX)	COMMAND DESCRIPTION	RESPONSE (HEX)
START ALGORITHM	Host initializes the sensor hub in AEC-SCD mode using the following commands:			
	1.0	AA 02 00 (optional)	Read the device operating mode	AB 00 00 application mode
	1.1	AA 50 07 00 [00000000FFD7FBDD00AB61 FE] (default, optional)	Bracketed SpO_2 coefficients are for example only. Values used in the final form factor may be different per section 2.	AB 00
	1.2	AA 10 00 03 (default, optional)	Set the output format to sensor + algorithm data (streamed data will include PPG, accelerometer, and algorithm data).	AB 00
	1.3	AA 10 01 05 (recommended)	Set the sensor hub DataRdyInt threshold. N=5	AB 00
	1.4	AA 10 02 01 (default, optional)	Set the samples report period to 40ms. Samples report rate to be one report per every sensor sample.	AB 00
	1.5	AA 50 07 0A 00 (default, optional)	Set the algorithm operation mode to Continuous HRM and Continuous SpO_2.	AB 00
	1.6	AA 50 07 0B 01 (default, optional)	Enable AEC [enabled by default].	AB 00
	1.7	AA 50 07 12 01 (default, optional)	Enable Auto PD Current Calculation [enabled by default].	AB 00
	1.8	AA 50 07 0C 01 (default, optional)	Enable SCD [enabled by default].	AB 00
	1.9	AA 44 04 01 01	Enable the input FIFO for host supplied accelerometer data.	AB 00
	Optional: Any command to change the algorithm settings and configurations from default should appear here before enabling the algorithm.			
	1.10	AA FF 03 (optional)	Read the sensor hub version for [MAX86141/MAX86140/MAXM86146/MAXM86161] (30.x.y/30.x.y/33.x.y/32.x.y)	AB 00 [1E/1E/21/20] XX YY
	For 33.x.y (MAXM86146), configure HR inputs to use Green1 and Green2; configure SpO_2 inputs to use IR, red. See "LEDs and PDs Configuration" for other example configurations.			

	1.20	For 33.x.y (MAXM86146) AA 50 07 19 13 56 00	Map configuration slot 1 to use LED1 (green1); map slot 2 to use LED3 (green2); map slot 3 to use LED5 (red); map slot 4 to use LED6 (IR)]	AB 00
	1.21	For 33.x.y (MAXM86146) AA 50 07 17 00 11	Map HR input 1 to use slot 1, PD1; map HR input 2 to use slot 2, PD2	AB 00
	1.22	For 33.x.y (MAXM86146) AA 50 07 18 30 20	Map SpO_2IR input to use slot 4 PD1; map SpO_2 red input to use slot 3 PD1	AB 00
	1.3.0	AA 52 07 01 (for normal algorithm report, CMD_DELAY = 465ms) AA 52 07 02 (for extended algorithm report, CMD_DELAY = 465ms)	Enable the HR and SpO_2 algorithm; the analog front-end (AFE).	AB 00
	1.31	AA 41 00 FF (optional sanity check)	Read register FF (PART_ID) of [MAX86141/MAX86140//MAXM86161]	AB 00 [25/24/36]
	1.32	AA 41 04 0F (optional sanity check) (accel data is required from either host or KX122/KX11S/LIS2DS12l)	Read register 0F (WHO_AM_I) of [KX122/KX112/LIS2DS12] if accel is connected to sensor hub.	AB 00 [1B/22/43]
READING SAMPLES REPORT IN OUTPUT FIFO	Host pipes in accel data and reads samples every 200ms:			
	2.0	AA 14 00 *[84 FE 1C 00 55 FC] [85 FE 1B 00 57 FC] [84 FE 1A 00 56 FC] [85 FE 19 00 54 FC] [87 FE 1B 00 57 FC]*	Write the N=5 sets of accel data to the input FIFO: N*[LSBx MSBx, LSBy MSBy, LSBy MSBz] [Sample 1 values] … [Sample N values] Each sample has three 2-byte integer values for X, Y, and Z in milli-g. In this example, the following accel data is sent: -380=FE84, 28=001C, -939=FC55; -379=85FE, 27=001B, -938=FC56; -380=FE84, , 26=001A, -938=FC56; -379=85FE, 25=0019, -940=FC54; -377=87FE, 27=001B, -937=FC57	AB 00 00 1E Thirty bytes received.
	2.1	AA 00 00 (optional)	Read sensor hub status byte: Bit 0: Sensor comm error Bits 1 and 2: Reserved Bit 3: FIFO filled to threshold (DataRdyInt) Bit 4: Output FIFO overflow (FifoOutOvrInt) Bit 5: Input FIFO overflow (FifoInOverInt) Bit 6: Host input FIFO underflow (HostAccelUflint) Bit 7: SCDSM detected skin contact (SCDMonSkin) If DataRdyInt is set, proceed to next step.	AB 00 08
	2.2	AA 12 00 (optional, but recommended every 25th iteration)	Get the number of samples (nn) in the FIFO.	AB 00 nn

	2.3	AA 12 01	Read the data stored in the FIFO; nn sample reports will be read. Samples report = [Sensor Data Output FIFO Samples Report] [Normal (or Extended) Algorithm Data Output FIFO Samples Report]	AB 00 data_for_nn_sample_reports
STOP	Host disables sensors and algorithm:			
	3.1	AA FF 02 04 00 00 00 00 00 (CMD_DELAY = 270ms)	Disable the AFE (e.g., MAX86141) and the accel input.	AB 00
	3.2	AA 52 07 00 (CMD_DELAY = 120ms)	Disable the Wearable Algorithm Suite (HR+ SpO_2) algorithm	AB 00

Down-Sampling Host Accelerometer Data to 25Hz

If the host accelerometer cannot be configured to 25Hz, then the host should configure the host accel to a higher frequency and down-sample the data to 25Hz. Example code for down-sampling may be found on the web:

https://stackoverflow.com/questions/31836598/subsampling-an-array-of-numbers

The pseudo code below illustrates how to down-sample 5Hz to 4 Hz and it may be used to test the down-sampling code.

```
o[0] = i[0]
o[1] = .75*i[1] + .25*i[2]
o[2] = .5*i[2] + .5*i[3]
o[3] = .25*i[3] + .75*i[4] ….
```

Converting Gs into Counts

The G's may be converted back to counts with the following pseudo code. The accelerometer resolution is set to 16 bits, 8G.

```
float G_s
int plus_minus_counts
plus_minus_counts = (int)((float)(((G_s * 32768.0)+4.0)/8.0))
if (plus_minus_counts > 32767)
  plus_minus_counts =32767
else if (plus_minus_counts < -32768)
  plus_minus_counts = -32768
```

E.g., (int)((7.9997559*32768)+4)/8 =(int)(32767.5) =32767

Power-Saving Options

Using Algorithm Data Output Format to Reduce the Size of the Samples Report

If the sensor data such as accelerometer and photoplethysmogram (PPG) signals are not required, the host may choose to request only algorithm data in the samples report. This reduces the I^2C communication time and affects power consumption. This is performed by configuring the output format to Algorithm Data.

Increasing the Samples Report Period

The sensor hub goes into deep sleep in Idle mode and wakes up on internal or external interrupts. To maximize the benefits of low power, the host may configure the samples report period of the algorithm to a longer time. This samples report period may be configured through an I^2C command.

The host is required to regularly empty the data in the sensor hub output FIFO at a periodic rate. The periodic rate depends on the rate that the sensor hub samples report is generated. When the samples report period is reduced, the host will not need to empty the FIFO as often. If the FIFO becomes full, then the FIFO should be emptied.

Allowing Sample Reports to Accumulate in the FIFO

The host may use the FIFO to accumulate sample reports. For instance, the host may read samples in the output FIFO at a period (*host reading FIFO period*) five times the length of the samples report period (*samples report period*) to avoid FIFO overflow. In this example, an average of five samples will be in the output FIFO.

By default, the samples report period (read samples report period, 0x11 0x02) is set to 40ms. In this case, it is recommended that the host read samples from the output FIFO every 200ms (*host reading period*). At these rates, on average there will be five samples in the output FIFO for the host to read.

Power Saving Mode: Algorithm Data Output Format, Increased Samples Report Period, Samples Report Accumulating in the FIFO and Sampled HR Example

Sampled HR may be used to reduce power usage. Configure the sensor hub as detailed in Table 9. Host Commands: HR, SpO_2, AEC, SCD Algorithm Report and use the additional configurations below:

- Change the output format in step 1.2 to Algorithm Data (0x10 0x00 0x02).
- Change the samples report period in step 1.4 to 25 (0x10 0x02 0x19); samples report period is (25*40ms = 1s).
- Host reads the samples in output FIFO at a period of five times the samples report period. For example, host reads output FIFO every 5 seconds (host reading FIFO period).
- Choose the Sampled HR or Sampled HR + One-Shot SpO_2 algorithm operation mode in step 1.5 (0x50 0x07 0x0A 0x03 or 0x04). The Sampled HR mode saves more power as it automatically switches to Activity Tracking mode once the heart rate is measured (~15s). The host may manage this configuration and switch to a continuous algorithm operation mode if there's motion
- Reduce message traffic by use the Packed Normal Algorithm Report in step 1.30. (32.9.33, 33.13.31).
- It's not required to execute step 2.0. The frequency of executing step 2.1 may also be reduced.
- Use the 0x50 0x07 0x20 to mask out unnecessary bytes from the Packed Normal Algorithm Report. (32.9.33, 33.13.31)
- Use the one-byte commands 0x20, 0x21 instead of 0x12 0x00, 0x12 0x01 to reduce I^2C traffic (32.9.33, 33.13.31).
- Allow sample reports to accumulate in the FIFO before emptying the FIFO.

This configuration helps the sensor hub to wake up less often, and I^2C communication time is minimized. For the MAXREFDES103, a fully charged battery will last about 8 hours in this configuration. It is believed that the battery of the MAXREFDES103 may be updated to a 200 mAH of size 401530 and still have some clearance for expansion. Alternatively, the host may also run the algorithm in the sampled one-shot algorithm operation mode say every 5 minutes to conserve power.

- 401530 200maH 3.7V Lithium-Ion Polymer Battery, Alibaba

Note: The above example is not appropriate for monitoring inter-beat interval (RtoR) value. RtoR and RtoR Confidence are reported whenever a new value is calculated by the algorithm and shown as zero for the rest of the time. Therefore, the last reported value may be missed if the samples report period is not set to 1.

SCD Only Algorithm Report

In this configuration, SCD is enabled and only SCD state is reported in the algorithm samples report. Before enabling SCD Only mode, the host should specify which LED is to be used by the SCD algorithm. The sequence of commands is shown below. (NA 32.x.x)

Table 14. SCD Only Data Output FIFO Samples Report Format

DATA ITEM (1 byte total)	# OF BYTES (MSB FIRST)	DESCRIPTION
SCD State	1	Skin contact state: 0: Undetected 1: Off skin 2: On some subject 3: On skin

Table 15. Host Commands: SCD Only Algorithm Report

	#	HOST COMMAND (HEX)	COMMAND DESCRIPTION	RESPONSE (HEX)
START ALGORITHM	Host initializes the sensor hub in SCD mode using the following commands:			
	1.0	AA E5 LED_IDX	Sets which LED index is to be used by the SCD algorithm. (Colors are for MAXREFDES103) LED_IDX = 0 for LED1 (Green) = 1 for LED2 (IR) = 2 for LED3 (Red)	AB 00 00 application mode
	1.1	AA 10 00 02	Set output mode to algorithm data (SCD Only when used with 1.7).	AB 00
	1.2	AA 10 01 01 (default, optional)	Set sensor hub DataRdyInt threshold.	AB 00
	1.3	AA 10 02 01 (default, optional)	Set the samples report period to 40ms. Samples report rate to be one report per every sensor sample.	AB 00
	1.4	AA 44 00 01 00	Enable AFE (e.g., MAX86141) with sensor hub samples.	AB 00
	1.5	AA E5 LED_IDX	Sets which LED index is to be used by the SCD algorithm. (Colors are for MAXREFDES103) LED_IDX = 0 for LED1 (Green) = 1 for LED2 (IR) = 2 for LED3 (Red)	AB 00
	1.6	AA 44 04 01 00 (if sensor hub accelerometer is used, CMD_DELAY = 20ms)	Enable accelerometer with sensor hub	AB 00

	Optional: Any command to change the algorithm settings and configurations from default should appear here before enabling the algorithm.			
	1.7	AA FF 03 (optional)	Read the sensor hub version for [MAX86141/MAX86140/MAXM86146/MAXM86161] (30.x.y/30.x.y/33.x.y/32.x.y)	AB 00 [1E/1E/21/20] XX YY
	For 33.x.y (MAXM86146), configure HR inputs to use Green1 and Green2; configure SpO_2 inputs to use IR, red. See “LEDs and PDs Configuration” for other example configurations.			
	1.8	For 33.x.y (MAXM86146) AA 50 07 19 13 56 00	Map configuration slot 1 to use LED1 (green1); map slot 2 to use LED3 (green2); map slot 3 to use LED5 (red); map slot 4 to use LED6 (IR)]	AB 00
	1.9	For 33.x.y (MAXM86146) AA 50 07 17 00 11	Map HR input 1 to use slot 1, PD1; map HR input 2 to use slot 2, PD2	AB 00
	1.10	For 33.x.y (MAXM86146) AA 50 07 18 30 20	Map SpO_2 IR input to use slot 4 PD1; map SpO_2 red input to use slot 3 PD1	AB 00
	1.3.0	AA 52 07 03 (CMD_DELAY = 465ms)	Enable SCD Only algorithm. The format of samples report is shown in the previous table.	AB 00
	1.31	AA 41 00 FF (optional sanity check)	Read register FF (PART_ID) of [MAX86141/MAX86140//MAXM86161]	AB 00 [25/24/36]
	1.32	AA 41 04 0F (optional sanity check if accel is connected to sensor hub)	Read register 0F (WHO_AM_I) of [KX122/KX112/LIS2DS12] if accel is connected to sensor hub.	AB 00 [1B/22/43]
READING SAMPLES REPORT IN OUTPUT FIFO	Host reads samples periodically (do not execute at a faster rate than the samples report period):			
	2.1	AA 00 00 (optional)	Read sensor hub status byte: Bit 0: Sensor comm error Bits 1 and 2: Reserved Bit 3: FIFO filled to threshold (DataRdyInt) Bit 4: Output FIFO overflow (FifoOutOvrInt) Bit 5: Input FIFO overflow (FifoInOverInt) Bit 6: Host input FIFO underflow (HostAccelUflint) Bit 7: SCDSM detected skin contact (SCDMonSkin) If DataRdyInt is set, proceed to next step.	AB 00 08
	2.2	AA 12 00 (optional, but recommended every 25th iteration)	Get the number of samples (nn) in the FIFO.	AB 00 nn
	2.3	AA 12 01	Read the data stored in the FIFO; nn sample reports will be read. Samples report = [Sensor Data Output FIFO Samples Report] [Normal (or Extended) Algorithm Data Output FIFO Samples Report]	AB 00 data_for_nn_sample_reports
S T	Host disables sensors and algorithm:			

	3.1	AA 44 00 00 (CMD_DELAY = 465ms)	Disable the AFE (e.g., MAX86141)	AB 00
	3.2	AA 44 04 00 (CMD_DELAY = 50ms)	Disable the accelerometer.	AB 00
	3.3	AA 52 07 00 (CMD_DELAY = 120ms)	Disable algorithm	AB 00

Using SCD State and Motion Detection (SCDSMD) for Power Savings

For the case of when the device is not on the skin, a motion-detection-enabled state machine can be implemented in the host to reduce power consumption. In this case, the sensor hub stays in sleep mode until a motion event is reported by the accelerometer, or an I^2C command is received from the host. Below is an example of such a state machine.

- **Active State:** Normally, the sensor hub runs in Active state in HR, SpO_2, AEC, SCD Report mode or Power Saving mode. If the SCD state in the report shows off-skin for certain time, the state machine switches to Probing state.
- **Probing State:** In this state, the host periodically turns the algorithm on and off. If an On-Skin state is reported while the algorithm is running, it will switch back to Active state and continue running the algorithm. Otherwise, after several attempts of turning the algorithm on and off (the off period can be increased after each attempt), it will switch to Off-Skin state. In Active and Probing states, the procedure to start, read report, or stop are like the sequence described in HR, SpO_2, AEC, SCD Report, or as described in Power Saving Mode.
- **Off-Skin State**: In Off-Skin state, the goal is to save more power by allowing the sensor hub to stay in sleep mode, so long as there is no motion.
- If the KX122/LIS2DS12 is connected to the sensor hub, the sensor hub must be configured to wake up on motion. In this case, the accelerometer is enabled in the interrupt mode; the motion threshold and the duration of motion are configured using the wake up on motion configuration command, as shown in the flow chart below.

To support this feature, the interrupt line of the accelerometer is required to be connected to the sensor hub. Once the sensor hub is configured, the host should start only the accelerometer. As soon as a motion interrupt occurs, the sensor hub will wake up and read accelerometer samples and store them in the sensor hub FIFO. The host should periodically read the sensor hub FIFO to check if any accelerometer sample has been captured since the last polling period. If there is a sample, the host should switch to Active state by first disabling the wake up on motion configuration and then restarting the algorithm.

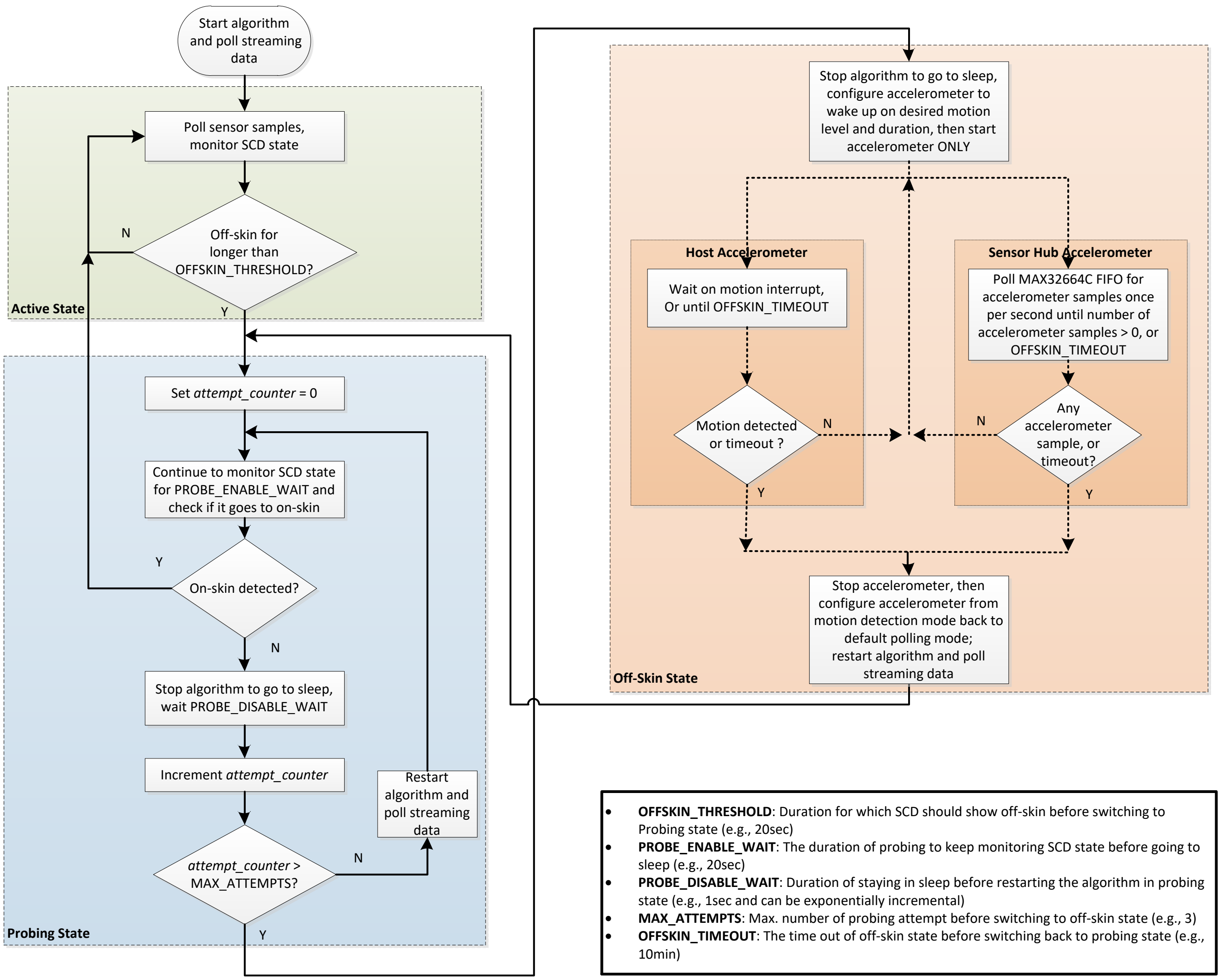


Figure 2. Example of an SCD-enabled, power saving state machine.

Table 16. Host Commands to Enable/Disable Wake Up on Motion Configuration of Sensor Hub Accelerometer for Off-Skin State

	#	HOST COMMAND (HEX)	COMMAND DESCRIPTION	RESPONSE (HEX)
START OFF-SKIN STATE	Host initializes the sensor hub in SCD mode using the following commands:			
	1.0	AA E5 LED_IDX	Sets which LED index is to be used by the SCD algorithm. (Colors are for MAXREFDES103) LED_IDX = 0 for LED1 (Green) = 1 for LED2 (IR) = 2 for LED3 (Red)	AB 00 00 application mode
	Host initializes the sensor hub to wake up on accelerometer motion detection to go into off-skin state.			
	Disable the sensor, accelerometer, and algorithm if enabled			

	1.1	AA 46 04 00 01 [05] [08]	Set the sensor hub accelerator in wake up on motion mode if motion is greater than a threshold for a certain duration, for example: [05]: 0.2s motion duration [08]: 0.5g motion	AB 00
	1.2	AA 10 00 01	Set the output FIFO mode to accelerometer data only.	AB 00
	1.3	AA 44 00 01 (CMD_DELAY = 465ms)	Enable AFE (e.g., MAX86141) with sensor hub samples.	AB 00
	1.4	AA 44 04 01 00 (CMD_DELAY = 20ms)	Enable the sensor-hub accelerometer. It will generate a report only if there is motion, according to step 1.4.	AB 00
READING SAMPLES REPORT IN OUTPUT FIFO	Host reads samples periodically repeatedly during off-skin state:			
	2.1	AA 00 00 (optional)	Read sensor hub status byte: Bit 0: Sensor comm error Bits 1 and 2: Reserved Bit 3: FIFO filled to threshold (DataRdyInt) Bit 4: Output FIFO overflow (FifoOutOvrInt) Bit 5: Input FIFO overflow (FifoInOverInt) Bit 6: Host input FIFO underflow (HostAccelUflint) Bit 7: SCDSM detected skin contact (SCDMonSkin) If DataRdyInt is set, proceed to next step.	AB 00 08
	2.2	AA 12 00 (optional, but recommended every 25th iteration)	Get the number of samples (nn) in the FIFO.	AB 00 nn
	2.3	AA 12 01	Read the data stored in the FIFO; nn samples will be read. The samples report will only include accelerometer data (6 bytes).	AB 00 data_for_nn_sample_reports
END OFF-SKIN	Host ends the wake up on motion configuration.			
	3.1	AA 44 04 00 (CMD_DELAY = 20ms)	Disable the accelerometer.	AB 00
	3.2	AA 46 04 00 00 FF FF	Disable wake up on motion	AB 00
	3.3	Proceed to start algorithm in HR, SpO_2, AEC, SCD Report or Power Saving mode		

Host Implemented Power Savings Using Accelerometer Wake

If the host has spare GPIOs, the accel interrupt can also be connected to the host. In this case, when SCD detects "off skin", the sensor hub/AFE may be put to deep sleep/shutdown and the accel can be configured to wake on motion via register writes from the host using commands. Alternatively MAX86141 interrupt and proximity detection may be used in a similar manner.

SCD State Machine for MFIO Interrupt Mode

SCD state machine for MFIO interrupt mode (SCDSM) is sensor hub application mode firmware algorithm which is only enabled in the MFIO interrupt mode. SCDSM is enabled via the command sequence AA BC 01, and it is disabled via the command sequence AA BB 00. SCDSM is also disabled when the algorithm is disabled or after a reset. Figure 11 is the flow chart for SCDSM. (SCDSM is not available for LIS2DS12, and 32.x.x)

- SCDSM is valid only when sensor hub has direct connection to accelerometer.
- SCDSM is only available in MFIO interrupt mode.
- SCDSM is only available for continuous HRM modes.
- Enable SCDSM after the algorithm is enabled.

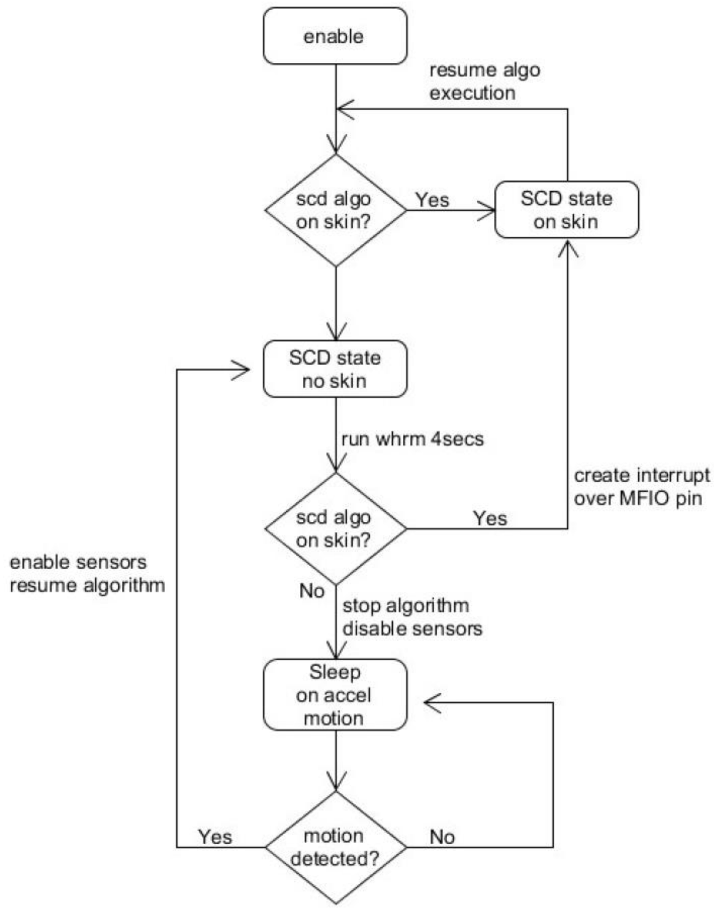


Figure 3. SCDSM flowchart.

Table 17. SCDSM Sensor Hub Commands

HOST I²C COMMAND (HEX)	DESCRIPTION	RESPONSE (HEX)
0xAA 0xBB 0x01	Disable skin contact state machine (SCDSM) on sensor hub. (NA for 32.x.x)	AB 00
0xAA 0xBC 0x01	Enable skin contact state machine (SCDSM) on sensor hub. (NA for 32.x.x) - SCDSM is valid only when sensor hub has direct connection to accelerometer.	AB 00

	- SCDSM is only available in MFIO interrupt mode. - SCDSM is only available for continuous HRM modes. - Enable SCDSM after the algorithm is enabled.	
0xAA 0xBD 0x00	Get the status of skin contact state machine (SCDSM) (NA for 32.x.x)	AB 00 <scdm_status> scdm_status: **0x00:** Disabled **0x01:** Enabled

Authentication Process for Maxim Wellness Suite Library

The wrist-based Maxim Wellness Suite Library (HRV, respiration rate, sleep quality, stress, sports coaching) is available as a separate library. Algorithms within in the Wellness Suite Library will not operate without proper authentication from the sensor hub. The authentication sequence with sensor hub is listed below. See also Wellness Library Integration Guide distributed with MAXREFDES103 .zip.

Table 18. Authentication Commands

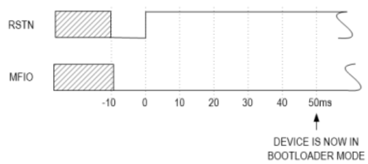
	#	HOST COMMAND (HEX)	COMMAND DESCRIPTION	RESPONSE (HEX)
START OFF-SKIN STATE	Host executes the authentication steps below.			
	1.1	AA B3 00	Read authentication ARRAY0, 6 bytes, from sensor hub.	AB 00 <ARRAY0, six bytes>
	1.2		ARRAY0 is an input to mxm_algosuite_manager_getauthinitials()*, wellness library suite. mxm_algosuite_manager_getauthinitials()* outputs ARRAY1, 12 bytes	
	1.3	AA B5 00 <ARRAY1 twelve bytes>	Send ARRAY1, 12 bytes, to sensor hub.	AB 00
	1.4	AA B4 00	Read authentication ARRAY2, 12 bytes, from sensor hub.	AB 00 <ARRAY2, twelve bytes >
	1.5	AA B2 00	Read authentication ARRAY3, 32 bytes, from sensor hub.	AB 00 <ARRAY2, thirty-two bytes >
	1.6		To finalize authentication in wellness library suite call mxm_algosuite_manager_authenticate()* with the inputs ARRAY2 (12 bytes) and ARRAY3. (32 bytes)	

Application .msbl Programming Sequence

To program the MAX32664C application .msbl, the host microprocessor may implement the software to flash the .msbl file. The MAX32664C uses the 8-bit slave address of 0xAA. Each page sent includes 16 CRC bytes for that page, so there are 8208 bytes per page sent in the payload of the 0x80 0x04 message. The number of pages is located at address 0x44 in the .msbl file. Values for the number of pages, initialization vector, authorization bytes, and page bytes may be different for the latest .msbl, but the locations of these values in the .msbl file remain the same. There are additional bytes in the .msbl past the last page; these are the file checksum bytes. Since the bootloader uses the commands listed below and it does not accept files, the file checksum bytes are not used by the bootloader.

It is recommended that I²C GPIO 'bit-bang' software be implemented on the host if the host MCU I²C hardware/HAL does not support messages that are 8210 bytes. The MAX32664 does not support partial page loading; MAX32674C/MAXREFDES104 does support partial pages.

Table 19. Annotated I²C Trace for Flashing the Application

HOST COMMAND	COMMAND DESCRIPTION	READ MAX32664C RESPONSE	RESPONSE DESCRIPTION
Sequence the MAX32664 to enter bootloader mode. *  *Figure 4. Sequence to enter bootloader mode.*			
0xAA 0x02 0x00†	Read mode.	0xAB 0x00 0x08	No error. Mode is bootloader.
0xAA 0x81 0x00+	Read bootloader firmware version.	0xAB 0x00 0x03 0xYY 0xZZ	No error. Version is 3.YY.ZZ.
0xAA 0x81 0x01†	Read bootloader page size.	0xAB 0x00 0x20 0x00	No error. Page size is 8192.
0xAA 0x80 0x02 0x00 *0x1A**	Bootloader flash. Set the "number of pages" to 31 based on the value at byte *0x44* from the application .msbl file.	0xAB 0x00	No error.
00000044 02 ed 27 af 1a 00 00 20 04 00 00 00 c2 31 90 2c *Figure 5. Page number byte 0x44 from the .msbl file.*			
0xAA 0x80 0x00 *0x1A 0xDB 0xE5 0x0D 0x90 0x79 0xE6 0xC6 0x13 0x87 0xB9**	Bootloader flash. Set the initialization vector bytes to the *0x28* to *0x32* values from the .msbl file.	0xAB 0x00	No error.
00000020 00 00 00 00 00 00 00 00 1a db e5 0d 90 79 e6 c6 00000032 13 87 b9 00 2b f5 ad cd 2e 47 d2 83 23 88 37 63 *Figure 6. Initialization vector bytes 0x28 to 0x32 from the .msbl file.*			
0xAA 0x80 0x01 *0x2B 0xF5 0xAD 0xCD 0x2E 0x47*	Bootloader flash. Set the authentication bytes to the *0x34 to*	0xAB 0x00	No error.

*0xD2 0x83 0x23 0x88 0x37 0x63 0x02 0xED 0x27 0xAF**	*0x43* values from the .msbl file.		

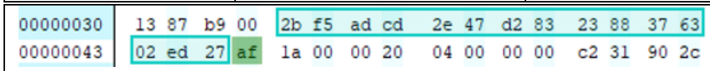


Figure 7. Authentication bytes 0x34 to 0x43 from the .msbl file.

0xAA 0x80 0x03* (CMD_DELAY = 1400ms)	Bootloader flash. Erase application.	0xAB 0x00	No error.
0xAA 0x80 0x04 *0xC2 0x31 0x90 ... 0x9E 0x6A 0x0E** (CMD_DELAY = 680ms)	Bootloader flash. Send page bytes *0x4C to 0x205B* from the .msbl file (8028 bytes).	0xAB 0x00	No error.

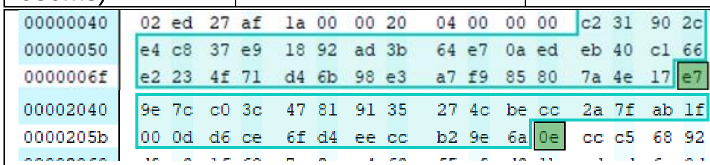


Figure 8. Send page bytes 0x4C to 0x205B from the .msbl file.

0xAA 0x80 0x04 *0xCC 0xC5 0x68 ... 0xF7 0xD6 0x4C** (CMD_DELAY = 680ms)	Bootloader flash. Send page bytes *0x205C* to *0x406B* from the .msbl file.	0xAB 0x00	No error.
0xAA 0x80 0x04 *0x2E 0xA6 0x13 ... 0x84 0xF7 0xCF** (CMD_DELAY = 680ms)	Bootloader flash. Send page bytes *0x406C* to *0x607B* from the .msbl file.	0xAB 0x00	No error.
0xAA 0x80 0x04 *0xD7 0x1F 0x7F ... 0x55 0xAB 0xB8** (CMD_DELAY = 680ms)	Bootloader flash. Send page bytes *0x607C* to *0x808B* from the .msbl file.	0xAB 0x00	No error.
0xAA 0x80 0x04 *0xC4 0x63 0x2B ... 0x48 0xCD 0x52** (CMD_DELAY = 680ms)	Bootloader flash. Send page bytes *0x808C to 0xA09B* from the .msbl file.	0xAB 0x00	No error.
0xAA 0x80 0x04 *0x89 0x33 0x22 ... 0x31 0xAD 0x19**	Bootloader flash. Send page bytes *0xA09C to 0xC0AB* from the .msbl file.	0xAB 0x00	No error.

(CMD_DELAY = 680ms)			
0xAA 0x80 0x04 *0x8B 0x97 0x18 ... 0xF3 0xCF 0x90** (CMD_DELAY = 680ms)	Bootloader flash. Send page bytes *0xC0AC to 0xE0BB* from the .msbl file.	0xAB 0x00	No error.
0xAA 0x80 0x04 *0xD0 0x78 0x38 ... 0x1F 0x7F 0x92** (CMD_DELAY = 680ms)	Bootloader flash. Send page bytes *0xE0BC* to *0x100CB* from the .msbl file.	0xAB 0x00	No error.
0xAA 0x80 0x04 *0xB1 0xE9 0x8F ... 0xF4 0x23 0xD8** (CMD_DELAY = 680ms)	Bootloader flash. Send page bytes *0x100CC* to *0x120DB* from the .msbl file.	0xAB 0x00	No error.
0xAA 0x80 0x04 *0xF8 0xC6 0x83 ... 0xF4 0x24 0xE2** (CMD_DELAY = 680ms)	Bootloader flash. Send page bytes *0x120DC* to *0x140EB* from the .msbl file.	0xAB 0x00	No error.
0xAA 0x80 0x04 *0x1F 0x4F 0x5C ... 0xCC 0x2E 0xCD** (CMD_DELAY = 680ms)	Bootloader flash. Send page bytes *0x140EC* to *0x160FB* from the .msbl file.	0xAB 0x00	No error.
0xAA 0x80 0x04 *0x40 0x1F 0x03 ... 0x26 0xEB 0xB9** (CMD_DELAY = 680ms)	Bootloader flash. Send page bytes *0x160FC* to *0x1810B* from the .msbl file.	0xAB 0x00	No error.
0xAA 0x80 0x04 *0x2F 0xD9 0xB2 ... 0xEE 0x2A 0x8F** (CMD_DELAY = 680ms)	Bootloader flash. Send page bytes *0x1810C* to *0x1A11B* from the .msbl file.	0xAB 0x00	No error.
0xAA 0x80 0x04 *0x51 0x32 0x47 ... 0x41 0xE6 0x47** (CMD_DELAY = 680ms)	Bootloader flash. Send page bytes *0x1A11C* to *0x1C12B* from the .msbl file.	0xAB 0x00	No error.

0xAA 0x80 0x04 *0x22 0xA6 0x06 ... 0x2A 0xCB 0x44** (CMD_DELAY = 680ms)	Bootloader flash. Send page bytes *0x1C12C* to *0x1E13B* from the .msbl file.	0xAB 0x00	No error.
0xAA 0x80 0x04 *0x68 0x9E 0x1E ... 0x53 0x89 0xE8** (CMD_DELAY = 680ms)	Bootloader flash. Send page bytes *0x1E13C* to *0x2014B* from the .msbl file.	0xAB 0x00	No error.
0xAA 0x80 0x04 *0x5F 0x1A 0x6A ... 0x14 0xA1 0x85** (CMD_DELAY = 680ms)	Bootloader flash. Send page bytes *0x2014C* to *0x2215B* from the .msbl file.	0xAB 0x00	No error.
0xAA 0x80 0x04 *0xE8 0xDE 0xC9 ... 0x81 0xD8 0x00** (CMD_DELAY = 680ms)	Bootloader flash. Send page bytes *0x2215C* to *0x2416B* from the .msbl file.	0xAB 0x00	No error.
0xAA 0x80 0x04 *0x0E 0xD2 0x16 ... 0x8D 0x69 0xEE** (CMD_DELAY = 680ms)	Bootloader flash. Send page bytes *0x2416C* to *0x2617B* from the .msbl file.	0xAB 0x00	No error.
0xAA 0x80 0x04 *0x2F 0x4B 0x38 ... 0x02 0xA7 0xDC** (CMD_DELAY = 680ms)	Bootloader flash. Send page bytes *0x2617C* to *0x2818B* from the .msbl file.	0xAB 0x00	No error.
0xAA 0x80 0x04 *0xA5 0xFE 0xFD ... 0xE3 0x38 0x89** (CMD_DELAY = 680ms)	Bootloader flash. Send page bytes *0x2818C* to *0x2A19B* from the .msbl file.	0xAB 0x00	No error.
0xAA 0x80 0x04 *0x52 0x88 0x9A ... 0xF0 0xC5 0x9D** (CMD_DELAY = 680ms)	Bootloader flash. Send page bytes *0x2A19C* to *0x2C1AB* from the .msbl file.	0xAB 0x00	No error.
0xAA 0x80 0x04 *0xA3 0xA6*	Bootloader flash. Send page bytes *0x2C1AC* to	0xAB 0x00	No error.

*0x92 ... 0xA0 0x4D 0xBE** (CMD_DELAY = 680ms)	*0x2E1BB* from the .msbl file.		
0xAA 0x80 0x04 *0x47 0x09 0x75 ... 0x24 0xBD 0x3D** (CMD_DELAY = 680ms)	Bootloader flash. Send page bytes *0x2E1BC* to *0x301CB* from the .msbl file.	0xAB 0x00	No error.
0xAA 0x80 0x04 *0x44 0xEC 0xE6 ... 0xBC 0xC9 0x5E** (CMD_DELAY = 680ms)	Bootloader flash. Send page bytes *0x301CC* to *0x321DB* from the .msbl file.	0xAB 0x00	No error.
0xAA 0x80 0x04 *0xD3 0x58 0x34 ... 0x62 0x00 0x37** (CMD_DELAY = 680ms)	Bootloader flash. Send page bytes *0x321DC* to *0x341EB* from the .msbl file.	0xAB 0x00	No error.

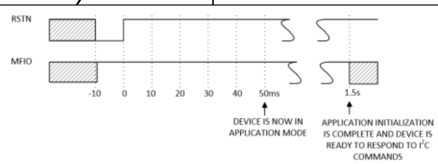


Figure 9. Sequence to enter application mode.

Alternately, the MAX32664C can be commanded to application mode.+			
0xAA 0x01 0x00 0x00+ (CMD_DELAY = 1.5s)	Set mode to 0x00 for application mode.	0xAB 0x00	No error.
0xAA 0x02 0x00+	Read mode.	0xAB 0x00 0x00	No errors. Mode is application.

**Mandatory*
†Recommended
+Optional

APIs for Sleep, Shutdown, Reset

Summarized below are the commands and methods to place the MAX32664, AFE, accelerometer into sleep, shutdown or reset.

Table 20. Sleep, Shutdown, Reset Protocol Definitions

COMMAND NAME	HOST COMMAND TO MAX32664	DESCRIPTION
MAX32664A/B/C/D shutdown	0xAA 0x01 0x00 0x01	Place the MAX32664 into shutdown (MAX32660 "Backup" mode with RAM

		disabled). Restart by power cycling or executing the reset to application mode sequence.
MAX32664 sleeps when idle and MFIO is low.		20.2.0+, 30.2.4+, 32.1.2+, 33.6.0+ uses deep-sleep for low-powered mode when idle and MFIO low.
Use write register command for MAX86140/MAX86141/ MAXM86146/MAXM86161 to turn off LED1, LED2, and LED3.	0x40 0x00 0x23 0x00 0x40 0x00 0x24 0x00 0x40 0x00 0x25 0x00 0x40 0x00 0x20 0x99 0x40 0x00 0x21 0x09	Set the current of LED1, LED2, and LED3 to 0. Set the sequence control registers LEDC1, LEDC2, and LEDC3 to ambient.
MAX86140/MAX86141/ MAXM86146/MAXM86161 AFE shutdown. Use AFE write register command.	0xAA 0x40 0x00 0x0D 0x02	Write 0x02 to MAX86140/1, MAXM86146/61 register 0x0D (System Control) to put the MAX86140/1, MAXM86146/61 into shutdown (SHDN) mode. The AFE must be enabled using the enable command when using the read, write AFE register command)
KX122, KX112 standby. Use accel write register command.	0xAA 0x40 0x04 0x18 0x00	Write 0x00 to KX122 register 0x018 (CNTL1) to put the KX122 into "Standby" mode. The KX122 must be enabled using the enable command when using the read, write KX122 register command)
LIS2DS12 wake on activity or wake on double/single tap. Use accel write register command.	0xAA 0x40 0x04 [reg_addr] [value]	See AN4748 LIS2DS12 for register settings. Accel interrupt pin can be connected to the host GPIO.
MAX32664 hard reset	Use MFIO and RSTN pins according to section 1.1	
WDT in MAX32664		Not implemented.

MAX32664 Power Consumption Estimate

The MAX32664 sensor hub family runs in two distinct operating modes. The Active mode is the mode in which the execution of the firmware occurs. The Deep Sleep mode is enabled by the sensor hub to save power when the processor is idle or there is no need for any processing. It makes all internal clocks of the MAX32664 gated off. In this mode, only RTC is enabled as a source of backup for wakeup. As soon as a sensor interrupt is received, the MAX32664 wakes up, completes the processing, and goes back to sleep. It also must wake up prior to I²C communication by setting MFIO low.

Table 21. Comparison of Active and Deep Sleep Power—Single Supply (V_{DD} Only)

MAX32664 OPERATIONAL MODE	POWER CONSUMPTION
Active	15.5664mW
Deep Sleep	0.00756mW

The tables below show the power consumption in each mode. To estimate the power consumption while running the algorithm, the percentage of time that the MAX32664 is in Active mode is measured. For this measurement, the report interval is set to 1 second and only algorithm data is reported.

Table 22. Estimated Power Consumption for the MAX32664C

WEARABLE SUITE ALGORITHM	MEASURED CPU ACTIVE TIME (AVERAGE) %	CALCULATED POWER CONSUMPTION (AVERAGE)*
		SINGLE-SUPPLY V_{DD} + INTERNAL LDO
Continuous HRM + Continuous SPO_2 mode	4.7%	0.74mW
Continuous HRM	4.3%	0.68mW
Sampled HRM	4.3%	0.68mW
Activity Tracking Only	4.2%	0.66mW

*V_{DD}: 1.8V and CPU clock: 96MHz.

Sensor Hub .msbl Version Numbering Convention

The MAX32664 is pre-programmed with bootloader software that accepts in-application programming of Maxim application code which consists of algorithms and the associated sensor driver. The MAX32664 is used as a sensor hub controller.

The algorithm/application code provides processed and/or raw data through the I^2C interface. Several variants of the MAX32664 exist based on the target application. These variants come pre-programmed with a bootloader that only accepts the matching encryption keys for the part (e.g., the MAX32664C bootloader is pre-programmed with the C encryption key, reference designs are programmed with Z keying, etc.). Designers should use the table below to select the correctly C keyed .msbl to flash the MAX32664GWEC part.

Table 23. Sensor Hub .msbl Version Numbering Convention

MAX32664+ Optical Sensor	.msbl versions	IC	Form Factor
MAX32664C+ MAX86141/0	MAX32664C_HSP2_WHRM_AEC_SCD_WSPO2_C_30.x.y .msbl (MAX32664 Website)	MAX32664GWEC	Wrist
	MAX32664C_HSP2_WHRM_AEC_SCD_WSPO2_Z_30.x.y .msbl (MAXREFDES103 Website or MAX32664 Website)	MAX32664GWEZ	
MAXM86146	MAX32664C_OB07_WHRM_AEC_SCD_WSPO2_C_33.x.y.msbl (MAX86146EVSYS Website)	MAXM86146CFU	Chest
MAX32664C+ MAXM86161	MAX32664C_MAXM86161_WHRM_AEC_SCD_WSPO2_C_32.9.23. msbl (MAX32664 Website)	MAX32664GWEC, KX122	Ear
	MAX32664C_MAXM86161_WHRM_AEC_SCD_WSPO2_C_32.9.33. msbl (MAX32664 Website)	MAX32664GWEC, LIS2DS12	

	MAX32664C_MAXM86161_WHRM_AEC_SCD_WSPO2_Z_32.x.y.msbl (MAX86161EVSYS Website or MAX32664 Website)	MAX32664GWEZ	

Default Application .msbl Versions Pre-Programmed on the MAX32664C

The MAX32664C is pre-programmed with the bootloader and the application .msbl application/sensor hub version listed in the table below. It is recommended that the sensor hub be updated with the latest application .msbl available on the Maxim Integrated website to be compatible with the latest sensor hub documentation.

Table 24. MAX32664C/MAXM86146 Pre-Programmed .msbl Version

MAXIM PART	PRE-PROGRAMMED .msbl APPLICATION/SENSOR HUB VERSION
MAX32664C	Version 30.2.2 (deprecated)

MAX32664 Processing Capabilities

The MAX32664 IC hardware is the same as the MAX32660.

- MIPS: Arm Cortex-M4 with FPU: 1.27 Dhrystone MIPS/MHz
- RAM: 96kB SRAM
- Flash: 256kB Flash Memory
- CPU Frequency: 96MHz

Heart Rate Algorithm Perfomance

Table 25. Heart Rate Algorithm Perfomance

Wearable Heart Rate Monitoring – WHRM		
Category	**Features**	**Specifications**
Algorithm	Measurement principle:	Optical PPG signal from wrist, 3D Axis Accelerometer
	Measurement range:	HR: [30 240] BPM Cadence (steps per minute): [90 360]
	Measurement accuracy:	HR: Accuracy Definition -> within +/-10% error band vs. reference (chest strap) Resting: 94 – 100% Walking: 93 - 99% Biking: 91 - 97% Running: 91 - 97% Daily Life: 90 - 100% Step counting: Accuracy Definition: 100% – (Absolute Percent Error) Treadmill walking: 89 – 95% Treadmill running: 86 – 92% Outdoor walking: 80 – 90% Activity Classification: Accuracy Definition: 100% – (Absolute Percent Error) Rest: 87 - 93% Treadmill walking: 93 - 99% Treadmill running: 90 - 95% Outdoor walking: 91 - 97% Outdoor biking: 80 - 90% Energy consumption Kcal: calculated according to ACSM & ADA
	Reference measurement device:	HR: ECG based chest strap IBI: ECG based chest strap (sampling rate min 1kHz)
	Average response time	25Hz, first response time 15sec
	Inputs:	Single/Multiple Channel PPG signal 3-axis accelerometer signals
	Built-in features:	Activity Classifier Built-in Step Counter Motion compensation of PPG for accurate HR estimation Inter-beat interval estimator Energy expenditure estimation
Measurement Positions	Wrist, Ear, Finger, Chest, Abdomen	Sports and daily life activities
Sensor & Signal Requirements	LED requirements:	Please refer to "Reference Design Document" for details
	Perfusion index range:	Minimum AC to calculate HR is 20nA with average 0.8% PI
	Sampling rate:	25 Hz
Calibration	Calibration:	Algorithm activity classifier is tuned for the sensor placement on wrist, a calibration might be required to train algorithm to improve its performance for another body location.

SpO_2 Algorithm Perfomance

Table 26. SpO_2 Algorithm Perfomance

SpO_2 on Wrist		
Category	**Features**	**Specifications**
Algorithm	Measurement principle:	Optical PPG signal from wrist
	Measurement range:	70 – 100% SpO_2
	Measurement accuracy:	RMSE ≤ 3.5% as required by FDA for reflective mode pulse oximeters
	Clinical test:	Certified calibration lab
	Measurement time:	30 – 60 sec, on-demand operations, one shot
	Inputs:	Red and Infrared PPG signals 3-axis accelerometer signals
	Built-in features:	Precise motion detector Automatic AFE setting adjustment for optimum PPG quality Signal conditioning Signal selection according to signal quality for discarding noisy signal portions Adjustable time-out duration Adjustable confidence threshold
Measurement Positions	Standing:	Arm is kept horizontal at the level of heart; palm is facing the floor
	Sitting	Arms are placed on a table Arms are crossed, arm with wristwatch is above the other arm
	Lying down:	Arms are horizontal
Sensor & Signal Requirements	LED requirements:	Center wavelength shift ≤ ±5nm LED full width at half maximum (FWHM) ≤ 20nm
	Perfusion index range:	PI ≥ 0.05%
	Sampling rate:	25 Hz
	Optical layout design:	Please refer to “Design Guide for SpO2 Measurement”
Calibration	Calibration lab:	Lab calibration is required for wearable’s finished industrial design. Please refer to lab calibration procedure guide “Design Guide for SpO2 Measurement”

References

- MAX32664 website: MAX32664 user guides; C-keyed .msbl for MAX32664C; sample host code; Guidelines For Spo2 Measurement, Guidelines For The Opto-mechanical Integration Of Heart-rate Monitors In Wearable Wrist Devices; Guidelines For The Opto-mechanical Integration Of Heart-rate Monitors In Wearable Earbud Devices:

  MAX32664 Design Resources Website
- Application Note 7148, protocol definition between sample host (MAX32630) and PC UART/BLE:

  Interface Guide for MAX32664 Sensor Hub-Based Reference Design Platforms
- MAXREFDES103# hardware, software files:

  MAXREFDES103#: Wrist-Based SpO_2, HR, and HRV Health Sensor Platform
- MAXM86146EVSYS# hardware, software files:

  MAXM86146: Evaluation System for the MAXM86146
- MAXM86161EVSYS# hardware, software files:

  MAXM86161: Evaluation System for the MAXM86161
- Validation And Performance Of A Wearable Heart rate Monitoring Algorithm
- Frequently Asked Questions: Maxim Support Center
- Technical Support