Measuring Heart Rate and SpO_2 Using the MAX32664A – A Quick Start Guide

UG7087; Rev 1; 2/20

Abstract

The MAX32664A is a variant of the MAX32664 sensor-hub family, which is specifically targeted for the finger-based measurement of heart rate and SpO_2. Combined with the MAX30101 optical sensor and a 3-axis accelerometer, it provides a sensor's raw data, as well as calculated heart-rate and SpO_2 data to a host device through its I^2C slave interface. This document provides step-by-step instructions that enable a user to communicate with the MAX32664A, and to calibrate, configure, and receive measurement and monitoring data.

Table of Contents

List of Figures

List of Tables

Introduction

The MAX32664A is a variant of the MAX32664 sensor-hub family that enables users to capture raw data as well as calculated heart-rate and SpO_2 data through finger contact. The MAX32664A is preprogrammed with the firmware, drivers, and algorithm that are required to interface with the MAX30101 sensor device through an I^2C master port. The I^2C slave interface is dedicated to establishing communication with a host microcontroller.

In order to properly capture and calculate the data, it is recommended that accelerometer data be provided to the MAX32664A. The MAX32664A firmware includes the required drivers for the Kionix® KX122 accelerometer, which is wired together with the MAX30101 to the same I^2C port. Alternatively, a host-side accelerometer can be used. In this case, the sampled accelerometer data must be periodically reported to the MAX32664A by the host microcontroller.

This document provides the instructions necessary to create a solution with the MAX32664A based on the MAXREFDES220# reference design.

1 Architecture

A typical health-sensing design includes a host microcontroller that communicates with the MAX32664A through the I2C bus. Two GPIO pins are needed to control the reset and the startup in Application or Bootloader mode through the RSTN and multifunction input/output (MFIO) pins. An MFIO pin is also used in Application mode to interrupt the host for I2C communication. The MAX32664A interfaces with the MAX30101 optical sensor through a second I2C bus.

To enter Bootloader mode:

- Set the RSTN pin to low for 10ms.
- 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 MAX32664 is in Bootloader mode.

To enter Application mode:

- Set the RSTN pin to low for 10ms.
- While RSTN is low, set the MFIO pin to high.
- After the 10ms has elapsed, set the RSTN pin to high. (The MFIO pin should be set to high at least 1ms before the RSTN pin is set to high.)
- After an additional 50ms has elapsed, the MAX32664 is in Application mode and the application performs its initialization of the application software.
- After approximately 1 second from when the RSTN pin was set to high, the application completes the initialization and the device is ready to accept I2C commands.

Figure 1 shows the top-level architecture.

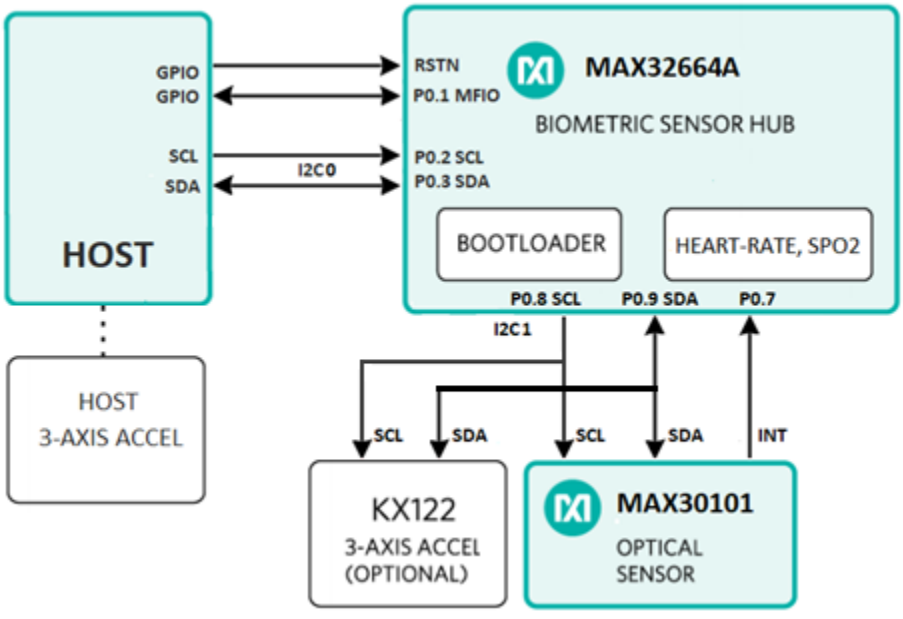

Figure 1. Architecture diagram for health-sensing applications.

1.1 Communicating with the MAX32664A

A host should use the I2C bus to communicate with the MAX32664A (slave) using a series of commands. A generic write command includes the following fields:

```
Slave_WriteAddress(1 byte)|Command_Family(1 byte)|Command_Index(1 byte)|Value(multiple bytes)
```

A generic response includes the following fields:

```
Slave_ReadAddress(1 byte)|Status(1 byte)|Value(multiple bytes)
```

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

The read status byte is an indicator of success (0x00) or failure, as shown **Table 1**.

Table 1. Read Status Byte Value

STATUS BYTE VALUE	DESCRIPTION
0x00	The write transaction was successful.
0x01	Illegal Family Byte and/or Command Byte was used.
0x02	This function is not implemented.
0x03	Incorrect number of bytes sent for the requested Family Byte.
0x04	Illegal configuration value was attempted to be set.
0x05	Incorrect mode specified. (In bootloader: Device is busy. Try again.)
0x80	General error while receiving/flashing a page during the bootloader sequence.
0x81	Checksum error while decrypting/checking page data.
0x82	Authorization error.
0x83	Application not valid.
0xFE	Device is busy. Try again.
0xFF	Unknown error.

This document provides examples of commands for establishing communication with the MAX32664A. For a complete list of commands and instructions for the I2C interface, see the **MAX32664 User Guide**.

1.2 Accelerometer

For best results, it is recommended that accelerometer data be provided to the MAX32664A. 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.

A sensor hub accelerometer can be integrated through the I²C port of the MAX32664A. In this case, the required driver for KX122 is already included. 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.

Alternatively, a host-side accelerometer can be used. In order to use the host-side accelerometer:

1. 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 I²C command should be discarded.
2. The host is required to use a 3-axis accelerometer at a 100Hz sampling rate. If a higher sampling rate is chosen, samples should be decimated to be synchronized with a 10ms PPG sampling time.
3. The host must queue five accelerometer samples and feed them at the same time to the MAX32664A using the commands shown in **Table 2**. The period of feeding samples should be 200ms. Because the sensor and the host accelerometer use different clock sources, exact synchronization between them is not possible.

Table 2. Host-Side Accelerometer—Sending Data to the MAX32664A

HOST COMMAND (HEX)	DESCRIPTION	MAX32664 RESPONSE (HEX)	DESCRIPTION
AA 44 04 01 01	Enable the host accelerometer.	AB 00	Success
AA 13 00 04	Read the sensor sample size for the accelerometer (optional).	AB 00 06	Success; 6 is the number of bytes per samples in FIFO
The following should be executed periodically at 200ms:			
AA 14 00 [Sample 1 values] … [Sample N values]	Write data to the input FIFO of the sensor hub. Each sample has three 2-byte integer values for X, Y, and Z in milli-g. N = 20	AB 00	Success
AA 00 00	Read the sensor hub status.	AB 00 00	Success; sensor hub not busy

2 Calibration of the SpO_2 Algorithm

2.1 Calibration of SpO_2 Coefficients for the Final Product

Due to variations in the physical design and optical shield of the final product, a calibration procedure 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 with a reference SpO_2 device to determine three calibration coefficients: a, b, and c. The details of the calibration procedure are described in the **Guidelines for SpO2 Measurement Using the Maxim MAX32664 Sensor Hub** application note.

Once three calibration coefficients are obtained, they need to be loaded to the MAX32664A every time prior to starting the algorithm. But first, they are required 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 calibration coefficients are:

- a = 1.5958422
- b = -34.659664
- c = 112.68987

They are sent to the MAX32664A in integer format after conversion:

- A_{int32} = round (10^5 x a) = 0x00026F60
- B_{int32} = round (10^5 x b) = 0xFFCB1D12
- C_{int32} = round (10^5 x c) = 0x00ABF37B

The calibration coefficients may be stored in the host flash separately and loaded to the MAX32664A after every reset. **Table 4** shows the sequence of commands for the calibration process. **Table 5** shows the format of received samples. Typically, R values are needed for the calibration process, as described in the **Guidelines for SpO2 Measurement Using the Maxim MAX32664 Sensor Hub** application note.

2.2 Algorithm Settings and Configurations

Table 3 shows the settings that are available for the heart-rate (HR)/SpO_2 algorithm. To update the algorithm settings, be sure to send the appropriate commands BEFORE enabling the algorithm, as shown in **Table 8**.

Table 3. Configurations and Settings—HR/SpO_2

FAMILY BYTE	ALGORITHM INDEX	CONFIGURATION INDEX	DESCRIPTION	DEFAULT VALUE
0x50 for write 0x51 for read	0x02	0x0B	SpO_2 calibration coefficients* 100,000 (12 bytes comprised of three 32-bit signed values)	A = 1.5958422 (0x00026f60) B = -34.659664 (0xffcb1d12) C = 112.68987 (0x00abf37b)

Table 4. Host Commands—SpO_2 Calibration

	#	HOST COMMAND (HEX)	COMMAND DESCRIPTION	RESPONSE (HEX)
START ALGORITHM	Host initializes the MAX32664A in calibration mode and starts the algorithm using following commands:			
	1.1	AA 10 00 03	Set the output mode to sensor + algorithm data (0x03) (streamed data will include PPG, accelerometer, and algorithm data).	AB 00
	1.2	AA 10 01 0F	Set the sensor hub interrupt threshold.	AB 00
	1.3	AA 52 00 01	Enable the AGC.	AB 00
	1.4	AA 44 04* 01 00 (if sensor hub accelerometer is used) AA 44 04* 01 01 (if Host accelerometer is used)	Enable the accelerometer with the sensor hub or host-side accelerometer.* (Do not use this command if there is no accelerometer.)	AB 00
	1.5	AA 44 03* 01	Enable the AFE (e.g., the MAX30101).	AB 00
	1.6	AA 52 02 02 (SpO_2 calibration report)	Enable the HR/SpO_2 algorithm. The format of the samples is shown in **Table 5.**	AB 00
	1.7	Wait for 100ms before sending the next command. Any command to change sensor registers should appear AFTER enabling the algorithm or they will be overwritten.		
READING SAMPLES	Host reads the samples upon receiving the MFIO interrupt from the MAX32664A. For SpO_2 calibration, continue as needed to capture the R value. See **Table 5**.			
	2.1	AA 00 00	Read the 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: Sensor hub busy (DevBusy) Bit 7: Reserved If DataRdyInt is set, proceed to the next step.	AB 00 08
	2.2	AA 12 00	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 (30 bytes each) will be read. The format of the samples is shown in **Table 5**.	AB 00 data_for_ nn_samples
STOP	Host ends the procedure:			
	3.1	AA 44 03* 00	Disable the AFE (e.g., the MAX30101).*	AB 00
	3.2	AA 44 04* 00	Disable the accelerometer.* (Do not use this command if there is no accelerometer.)	AB 00
	3.3	AA 52 02 00	Disable the algorithm.	AB 00

**Provided indexes are examples for sensors such as the MAX30101 AFE or KX122 accelerometer.*

Table 5. Format of Received Samples—SpO_2 Calibration Mode

DATA SOURCE	BYTE INDEX	DATA ITEM	# OF BYTES (MSB FIRST)	DESCRIPTION
MAX30101 (12 Bytes)*	0	LED1	3	IR counter
	3	LED2	3	Red counter
	6	LED3	3	N/A
	9	LED4	3	N/A
Accelerometer (6 Bytes)*	12	accelX	2	Two's complement. LSB = 0.001g
	14	accelY	2	Two's complement. LSB = 0.001g
	16	accelZ	2	Two's complement. LSB = 0.001g
HR/SpO_2 Algorithm (14 Bytes)**	18	Heart rate	2	10x heart-rate value
	20	Heart-rate confidence	1	Calculated confidence level in %
	21	SpO_2	2	10x SpO_2 value
	23	Algorithm state	1	Algorithm current state: 0: No object is detected 1: Something is on sensor 2: Another object is detected 3: Finger is detected
	24	R	2	1000x calculated R value
	26	Algorithm status	1	Algorithm current status: 0: Success 1: Not ready -1: Something is on sensor -2: Device excessive motion -3: No object -4: Pressing too hard -5: Object instead of finger -6: Finger excessive motion
	27	Motion flag	1	Shows excessive motion: 0: No motion 1: Excessive motion
	28	Perfusion	2	10x perfusion value
	30	Interbeat interval (IBI)	2	1000x IBI value (in ms)

*If the output mode includes the sensor.
**If the output mode includes the algorithm.

3 Measuring Heart Rate and SpO_2 on Finger

3.1 Raw Data Collection Mode

For hardware testing purposes, the user may choose to start the MAX32664A to collect raw PPG samples. In this case, the host configures the MAX32664A to work in Raw Data mode (no algorithm report). **Table 6** lists the set of commands that are needed to work in this mode. In Raw Data mode, only raw PPG samples and accelerometer data are included in the received samples.

The AGC must be turned off to collect raw PPG data, as shown in step 1.6 in **Table 6**. In this case, LED currents will not be adjusted automatically. Although the algorithm is running, it will not affect the PPG samples. If the reported PPG data is saturated, you can reduce the LED currents as shown. Note that updating MAX30101 registers should appear AFTER enabling the algorithm and the MAX30101, or they will be overwritten during initialization. By setting the output mode to sensor data in step 1.1, only the 12-byte PPG data of the MAX30101 and 6-byte accel data will be reported in received samples.

Table 6. Host Commands—Raw Data Mode

	#	HOST COMMAND (HEX)	COMMAND DESCRIPTION	RESPONSE (HEX)
START ALGORITHM	Host initializes the MAX32664A:			
	1.1	AA 10 00 01	Set the output mode to sensor data (0x01, streamed data will include only PPG and accelerometer data).	AB 00
	1.2	AA 10 01 0F	Set the sensor hub interrupt threshold.	AB 00
	1.3	AA 44 04* 01 00 (if sensor hub accelerator is used) AA 44 04* 01 01 (if host accelerator is used)	Enable the accelerometer with the sensor hub or host-side accelerometer.* (Do not use this command if there is no accelerometer.)	AB 00
	1.4	AA 44 03* 01	Enable the AFE (e.g., the MAX30101).	AB 00
	1.5	AA 52 02 01	Enable the HR/SpO_2 algorithm	AB 00
	1.6	AA 52 00 00	Disable the AGC.	AB 00
	1.7	Wait for 100ms before sending the next command. Any command to change the sensor registers should appear AFTER enabling the algorithm or they will be overwritten.		
	1.8	AA 40 03 0C [7F]	Set the MAX30101 LED1 (red) current to half of full scale. Reduce [7F] if the signal is saturated.	AB 00
	1.9	AA 40 03 0D [7F]	Set the MAX30101 LED2 (IR) current to half of full scale. Reduce [7F] if signal is saturated.	AB 00
READING SAMPLES	Host reads samples upon receiving the MFIO interrupt by the MAX32664A. For raw data, repeat as needed to collect PPG counters.			
	2.1	AA 00 00	Read the 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: Sensor hub busy (DevBusy) Bit 7: Reserved If DataRdyInt is set, proceed to the next step.	AB 00 08
	2.2	AA 12 00	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 (18 bytes each) will be read. The format of samples is shown in **Table 7.**	AB 00 data_for_ nn_samples

STOP	Host ends the procedure:			
	3.1	AA 44 03* 00	Disable the AFE (e.g., the MAX30101).*	AB 00
	3.2	AA 44 04* 00	Disable the accelerometer.* (Do not use this command if there is no accelerometer.)	AB 00
	3.2	AA 52 02 00	Disable the algorithm.	AB 00

**Provided indexes are examples for sensors such as the MAX30101 AFE or KX122 accelerometer.*

Table 7. Format of Received Samples—Raw Data Mode

DATA SOURCE	BYTE INDEX	DATA ITEM	# OF BYTES (MSB FIRST)	DESCRIPTION
MAX30101 (12 Bytes)*	0	LED1	3	IR counter
	3	LED2	3	Red counter
	6	LED3	3	N/A
	9	LED4	3	N/A
Accelerometer (6 Bytes)*	12	accelX	2	Two's complement. LSB = 0.001g
	14	accelY	2	Two's complement. LSB = 0.001g
	16	accelZ	2	Two's complement. LSB = 0.001g

*If the output mode includes the sensor.

3.2 Algorithm Mode: Heart Rate and SpO_2

Table 8 shows the list of commands to start the HR/SpO_2 algorithm.

Table 8. Host Commands—HR/SpO_2 Algorithm

	#	HOST COMMAND (HEX)	COMMAND DESCRIPTION	RESPONSE (HEX)
START ALGORITHM	Host initializes the MAX32664A:			
	1.1	AA 50 02 0B 00 02 6F 60 (example for A) FF CB 1D 12 (example for B) 00 AB F3 7B (example for C)	Set SpO_2 calibration coefficients derived from the procedure in section **2.1**. Provided example for: A = 1.5958422, B = -34.659664, C = 112.68987.	AB 00
	1.2	AA 10 00 03	Set output mode to sensor + algorithm data (0x03, streamed data will include PPG, accelerometer, and algorithm data).	AB 00
	1.3	AA 10 01 0F	Set sensor hub interrupt threshold.	AB 00
	1.4	AA 52 00 01	Enable the AGC.	AB 00
	1.5	AA 44 04* 01 00 (if sensor hub accelerator is used) AA 44 04* 01 01 (if host accelerator is used)	Enable the accelerometer with the sensor hub or host-side accelerometer.* (Do not use this command if there is no accelerometer.)	AB 00
	1.6	AA 44 03* 01	Enable the AFE (e.g., the MAX30101).	AB 00
	1.7	AA 52 02 01 (normal algorithm report)	Enable the HR/SpO_2 algorithm. The format of the samples is shown in **Table 9**.	AB 00
READING SAMPLES	Host reads the samples upon receiving the MFIO interrupt by the MAX32664A.			
	2.1	AA 00 00	Read the 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: Sensor hub busy (DevBusy) Bit 7: Reserved If DataRdyInt is set, proceed to next step.	AB 00 08
	2.2	AA 12 00	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 (24 bytes each) will be read. The format of samples is shown in **Table 9**.	AB 00 data_for_ nn_samples
STOP	Host ends the procedure:			
	3.1	AA 44 03* 00	Disable the AFE (e.g., the MAX30101).*	AB 00
	3.2	AA 44 04* 00	Disable the accelerometer.* (Do not use this command if there is no accelerometer.)	AB 00
	3.3	AA 52 02 00	Disable the algorithm.	AB 00

**Provided indexes are examples for sensors such as the MAX30101 AFE or KX122 accelerometer.*

Table 9. Format of Received Samples—HR/SpO_2 Algorithm

DATA SOURCE	BYTE INDEX	DATA ITEM	# OF BYTES (MSB FIRST)	DESCRIPTION
MAX30101 (12 Bytes)*	0	LED1	3	IR counter
	3	LED2	3	Red counter
	6	LED3	3	N/A
	9	LED4	3	N/A
Accelerometer (6 Bytes)*	12	accelX	2	Two's complement. LSB = 0.001g
	14	accelY	2	Two's complement. LSB = 0.001g
	16	accelZ	2	Two's complement. LSB = 0.001g
HR/SpO_2 Algorithm (9 Bytes)**	18	Heart rate	2	10x heart-rate value
	20	Heart rate confidence	1	Calculated confidence level in %
	21	SpO_2	2	10x SpO_2 value
	23	Algorithm state	1	Algorithm current state: 0: No object is detected 1: Something is on sensor 2: Another object is detected 3: Finger is detected
	24	Algorithm status	1	Algorithm current status: 0: Success 1: Not ready -1: Something is on sensor -2: Device excessive motion -3: No object -4: Pressing too hard -5: Object instead of finger -6: Finger excessive motion
	25	Interbeat interval (IBI)	2	1000x IBI value in ms

*If the output mode includes the sensor.
**If the output mode includes the algorithm.

Revision History

REVISION NUMBER	REVISION DATE	DESCRIPTION	PAGES CHANGED
0	08/19	Initial release	—
1	02/20	Updated Tables 5 and 9	11, 15