MAX30101

MAX30101 - интегрированный пульсовый оксиметрический модуль и модуль мониторинга пульса.

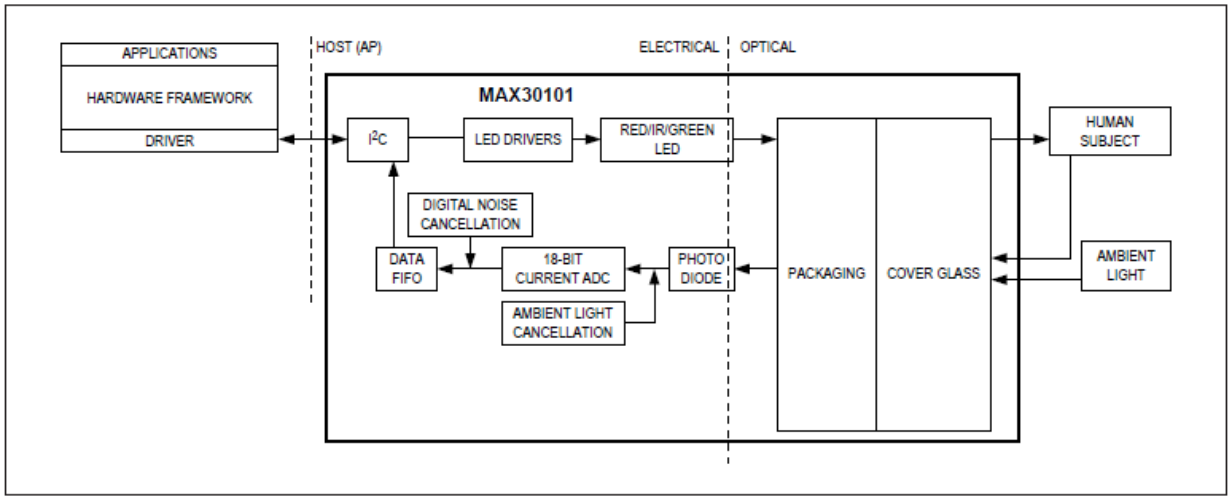
Он включает в себя внутренние светодиоды, фотодетекторы, оптические элементы и малошумную электронику с рассеиванием окружающего света.

MAX30101 обеспечивает комплексное системное решение для облегчения процесса проектирования мобильных и носимых устройств

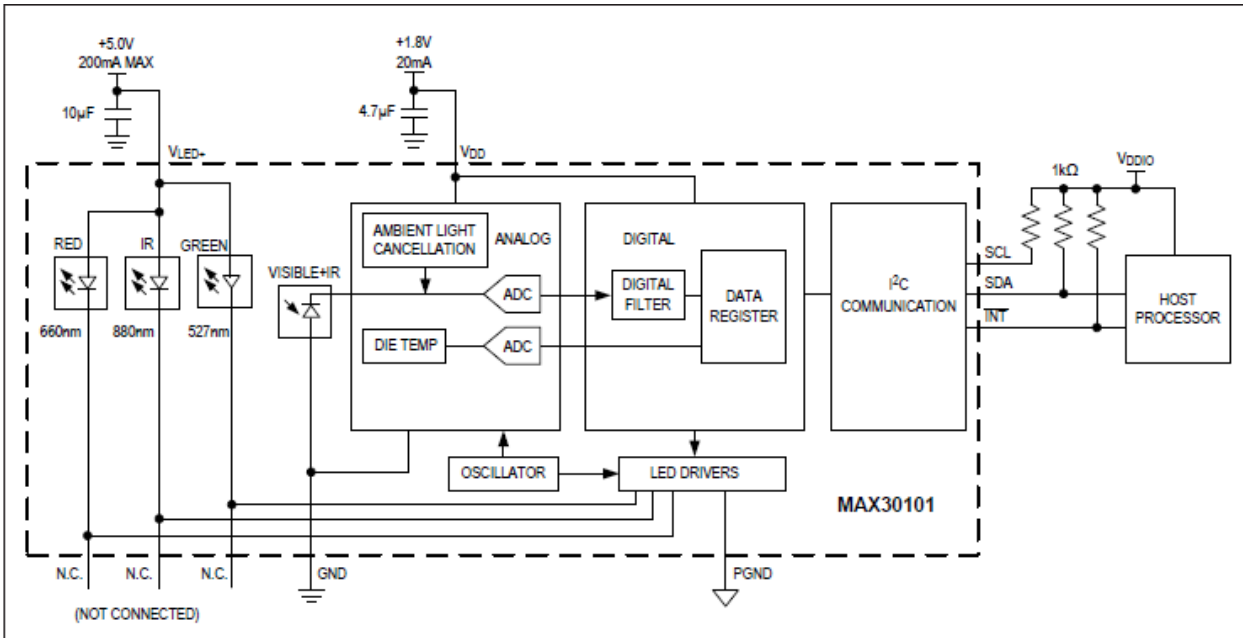
MAX30101 работает на одном источнике питания 1,8 В и отдельном источнике питания 5,0 В для внутренних светодиодов.

Связь осуществляется через стандартный I2C интерфейс.

Модуль может быть выключен с помощью программного обеспечения с нулевым резервным током, что позволяет шинам питания постоянно оставаться на питании.



Типовая схема применения



Электрические характеристики

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
POWER SUPPLY						
Power-Supply Voltage	V_{DD}	Guaranteed by RED and IR count tolerance	1.7	1.8	2.0	V
LED Supply Voltage V_{LED+} to PGND	V_{LED+}	Guaranteed by PSRR of LED driver (RED and IR LED only)	3.1	3.3	5.0	V
		Guaranteed by PSRR of LED driver (GREEN LED only). T_A = 25°C	4.5	5.0	5.5	
Supply Current	I_{DD}	SpO_2 and HR mode, PW = 215µs, 50sps		600	1100	µA
		IR only mode, PW = 215µS, 50sps		600	1100	
Supply Current in Shutdown	I_{SHDN}	T_A = +25°C, MODE = 0x80		0.7	2.5	µA

PARAMETER	SYMBOL	CONDITIONS		MIN	TYP	MAX	UNITS
PULSE OXIMETRY/HEART-RATE SENSOR CHARACTERISTICS							
ADC Resolution					18		bits
Red ADC Count (Note 2)	REDC	LED1_PA = 0x0C, LED_PW = 0x01, SPO2_SR = 0x05, ADC_RGE = 0x00			65536		Counts
IR ADC Count (Note 2)	IRC	LED2_PA = 0x0C, LED_PW = 0x01, SPO2_SR = 0x05 ADC_RGE = 0x00			65536		Counts
Green ADC Count (Note 2)	GRNC	LED3_PA = LED4_PA = 0x24, LED_PW = 0x01, SPO2_SR = 0x05, ADC_RGE = 0x00			65536		Counts
Dark Current Count	LED_DCC	LED1_PA = LED2_PA = 0x00, LED_PW = 0x03, SPO2_SR = 0x01 ADC_RGE = 0x02			30	128	Counts
					0.01	0.05	% of FS
DC Ambient Light Rejection (Note 3)	ALR	ADC counts with finger on sensor under direct sunlight (100K lux), ADC_RGE = 0x3, LED_PW = 0x03, SPO2_SR = 0x01	Red LED		2		Counts
			IR LED		2		Counts
ADC Count—PSRR (V_{DD})	$PSRRV_{DD}$	1.7V < V_{DD} < 2.0V, LED_PW = 0x00, SPO2_SR = 0x05			0.25	1	% of FS
		Frequency = DC to 100kHz, 100mV_{P-P}			10		LSB
ADC Count—PSRR (LED Driver Outputs)	$PSRR_{LED}$	3.1V < V_{LED+} < 5.0V, LED1_PA = LED2_PA = 0x0C, LED_PW = 0x01, SPO2_SR = 0x05			0.05	1	% of FS
		4.5V < V_{LED+} < 5.5V, T_A = 25°C LED3_PA = LED4_PA = 0x24, LED_PW = 0x01, SPO2_SR = 0x05					
		Frequency = DC to 100kHz, 100mV_{P-P}			10		LSB
ADC Clock Frequency	CLK			10.2	10.48	10.8	MHz
ADC Integration Time (Note 3)	INT	LED_PW = 0x00			69		µs
		LED_PW = 0x01			118		
		LED_PW = 0x02			215		
		LED_PW = 0x03			411		
Slot Timing (Timing Between Sequential Channel Samples; e.g., Red Pulse Rising Edge To IR Pulse Rising Edge)	INT	LED_PW = 0x00			427		µs
		LED_PW = 0x01			525		
		LED_PW = 0x02			720		
		LED_PW = 0x03			1107		
COVER GLASS CHARACTERISTICS (Note 3)							
Hydrolytic Resistance Class		Per DIN ISO 719			HGB 1		

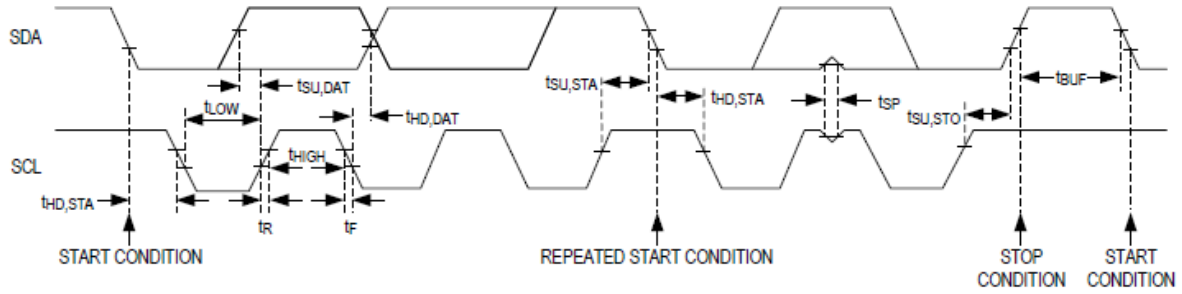
PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
IR LED CHARACTERISTICS (Note 3)						
LED Peak Wavelength	λ_P	I_{LED} = 20mA, T_A = +25°C	870	880	900	nm
Full Width at Half Max	$\Delta\lambda$	I_{LED} = 20mA, T_A = +25°C		30		nm
Forward Voltage	V_F	I_{LED} = 20mA, T_A = +25°C		1.4		V
Radiant Power	P_O	I_{LED} = 20mA, T_A = +25°C		6.5		mW
RED LED CHARACTERISTICS (Note 3)						
LED Peak Wavelength	λ_P	I_{LED} = 20mA, T_A = +25°C	650	660	670	nm
Full Width at Half Max	$\Delta\lambda$	I_{LED} = 20mA, T_A = +25°C		20		nm
Forward Voltage	V_F	I_{LED} = 20mA, T_A = +25°C		2.1		V
Radiant Power	P_O	I_{LED} = 20mA, T_A = +25°C		9.8		mW
GREEN LED CHARACTERISTICS (Note 3)						
LED Peak Wavelength	λ_P	I_{LED} = 50mA, T_A = +25°C	530	537	545	nm
Full Width at Half Max	$\Delta\lambda$	I_{LED} = 50mA, T_A = +25°C		35		nm
Forward Voltage	V_F	I_{LED} = 50mA, T_A = +25°C		3.3		V
Radiant Power	P_O	I_{LED} = 50mA, T_A = +25°C		17.2		mW
PHOTODETECTOR CHARACTERISTICS (Note 3)						
Spectral Range of Sensitivity	Λ > 30% QE	QE: Quantum Efficiency	640		980	nm
Radiant Sensitive Area	A			1.36		mm^2
Dimensions of Radiant Sensitive Area	L x W			1.38 x 0.98		mm x mm
INTERNAL DIE TEMPERATURE SENSOR						
Temperature ADC Acquisition Time	T_T	T_A = +25°C		29		ms
Temperature Sensor Accuracy	T_A	T_A = +25°C		±1		°C
Temperature Sensor Minimum Range	T_{MIN}			-40		°C
Temperature Sensor Maximum Range	T_{MAX}			85		°C
DIGITAL INPUTS (SCL, SDA)						
Input Logic-Low Voltage	V_{IL}				$0.3 \times V_{DD}$	V
Input Logic-High Voltage	V_{IH}		$0.7 \times V_{DD}$			V
Input Hysteresis	V_{HYS}			$0.5 \times V_{DD}$		V
Input Leakage Current	I_{IN}			±0.1	±1	µA
Input Capacitance	C_{IN}			10		pF
DIGITAL OUTPUTS (SDA, $\overline{INT}$)						
Output Low Voltage	V_{OL}	I_{SINK} = 3mA			0.4	V

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
I²C TIMING CHARACTERISTICS						
I²C Write Address				AE		Hex
I²C Read Address				AF		Hex
SCL Clock Frequency	f_{SCL}	Lower limit not tested	0		400	kHz
Bus Free Time Between STOP and START Condition	t_{BUF}		1.3			µs
Hold Time (Repeated) START Condition	$t_{HD,STA}$		0.6			µs
SCL Pulse-Width Low	t_{LOW}		1.3			µs
SCL Pulse-Width High	t_{HIGH}		0.6			µs
Setup Time for a Repeated START Condition	$t_{SU,STA}$		0.6			µs
Data Hold Time	$t_{HD;DAT}$		0		0.9	µs
Data Setup Time	$t_{SU;DAT}$		100			ns
Setup Time for STOP Condition	$t_{SU;STO}$		0.6			µs
Pulse Width of Suppressed Spike	t_{SP}				50	ns
Bus Capacitance	C_b				400	pF
SDA and SCL Receiving Rise Time	T_r	(Note 4)	20		300	ns
SDA and SCL Receiving Fall Time	t_{Rf}	(Note 4)	20 x V_{DD}/5.5		300	ns
SDA Transmitting Fall Time	t_{of}		20 x V_{DD}/5.5		250	ns

I2C интерфейс

MAX30101 имеет I2C/SMBus-совместимый двухпроводный последовательный интерфейс, состоящий из последовательной линии данных (SDA) и последовательной линии часов (SCL).

Диаграмма синхронизации двухпроводного интерфейса:



SDA и SCL облегчают связь между MAX30101 и мастером с тактовой частотой до 400 кГц.

Мастер генерирует SCL и инициирует передачу данных в шине. Основное устройство записывает данные в MAX30101, передавая соответствующий slave адрес с последующими данными. Каждая последовательность передачи обрамляется условием START (S) или REPEATED START (Sr) и условием STOP (P). Каждое слово, передаваемое на MAX30101, имеет длину 8 бит и сопровождается импульсом часов. Данные основного считывания от MAX30101 передают собственный адрес slave, а затем серию из девяти импульсов SCL.

MAX30101 передает данные по SDA синхронно с генерируемыми импульсами SCL. Мастер подтверждает получение каждого байта данных. Каждая последовательность чтения обрамляется условием START (S) или REPEATED START (Sr), условием STOP (P). SDA работает как в качестве входного, так и в качестве выходного отверстия

SCL работает только как входной

Один бит данных передается в течение каждого цикла SCL. Данные о SDA должны оставаться стабильными в течение высокого периода импульса SCL. Изменения в SDA при высокой SCL являются управляющими сигналами

SDA и SCL работают на холостом ходу, когда шина не используется.

Мастер инициирует связь, выдавая **условие START**. Условие START - переход с высокой на низкую SDA с высокой SCL.

Условие STOP - это переход с низкой на высокую SDA, в то время как SCL высока

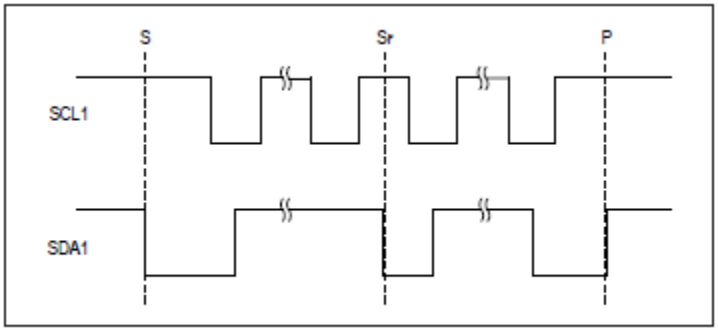


Figure 7. START, STOP, and REPEATED START Conditions

Условие START от master сигнализирует о начале передачи на MAX30101. Master прекращает передачу и освобождает автобус, выдавая условие СТОП. Шина остается активной, если вместо условия СТОП создается повторяющееся условие START.

MAX30101 распознает состояние STOP в любой момент передачи данных, за исключением случаев, когда состояние STOP происходит в том же высоком импульсе, что и условие START.

Для правильной работы не посылать условие STOP во время того же высокого импульса SCL, как условие START.

Мастер шины инициирует связь с рабочими устройствами, выдавая условие START, за которым следует 7-битный идентификатор slave

Мастер шины инициирует связь с рабочими устройствами, выдавая условие START, за которым следует 7-битный идентификатор slave. При простое MAX30101 ожидает условия START, за которым следует ID slave. Последовательный интерфейс сравнивает каждый slave идентификатор бит за битом, позволяя интерфейсу отключить и отключиться от SCL немедленно, если обнаружен неправильный идентификатор slave . После распознавания условия START, за которым следует правильный идентификатор slave, MAX30101 программируется для приема или отправки данных. LSB-слово slave ID - это бит чтения/записи (R/W).

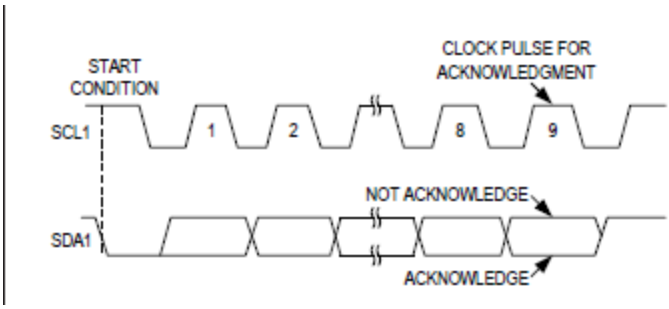
R/W указывает, что мастер записывает или считывает данные из MAX30101 (R/W = 0 выбирает условие записи, R/W = 1 выбирает условие чтения). После получения соответствующего идентификатора slave, MAX30101 выпускает ACK, вытягивая SDA на один такт.
MAX30101 slave ID состоит из семи фиксированных битов, B7-B1 (установлено в 0b10111). Сначала передается наиболее значимый бит идентификатора рабыни (B7), а затем остальные биты

Возможные идентификаторы slave устрйоства:

Table 17. Slave ID Description

B7	B6	B5	B4	B3	B2	B1	B0	WRITE ADDRESS	READ ADDRESS
1	0	1	0	1	1	1	R/$\overline{W}$	0xAE	0xAF

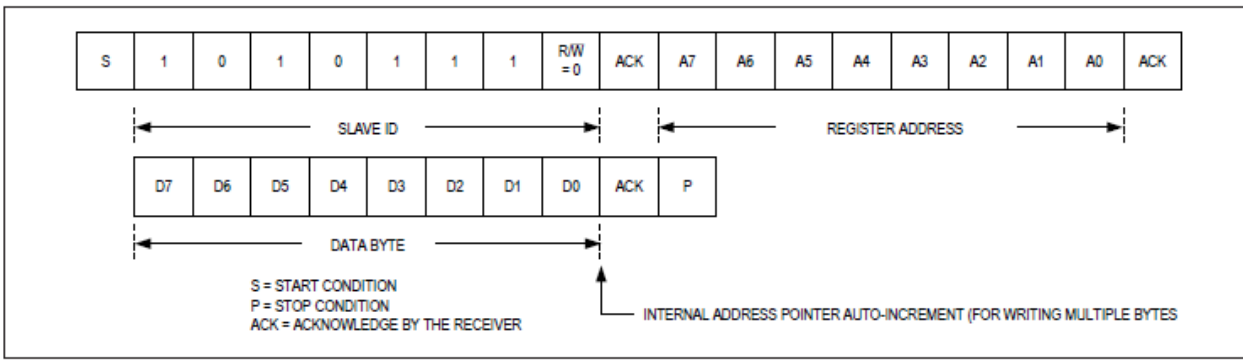
Бит подтверждения (ACK) является часовым девятым битом, который MAX30101 использует для получения каждого байта данных в режиме записи:



MAX30101 оттягивает SDA в течение всего основного 9-го импульса часов, если предыдущий байт успешно получен. Мониторинг ACK позволяет выявлять неудачные передачи данных. Неудачная передача данных происходит, если приемное устройство занято или если произошел системный сбой. В случае неудачной передачи данных мастер шины пытается наладить связь. Мастер снимает SDA во время 9-го тактового цикла, чтобы подтвердить получение данных, когда MAX30101 находится в режиме чтения

После каждого прочитанного байта мастер отправляет подтверждение для продолжения передачи данных. Не-подтверждение отправляется, когда мастер читает последний байт данных из MAX30101, за которым следует условие STOP.

Для операции записи отправьте идентификатор slave как первый байт, затем байт адреса регистра, а затем один или несколько байт данных. Указатель адреса регистра автоматически увеличивается после каждого байта полученных данных, так, например, весь регистровый банк может быть записан одновременно. Завершить передачу данных с условием STOP

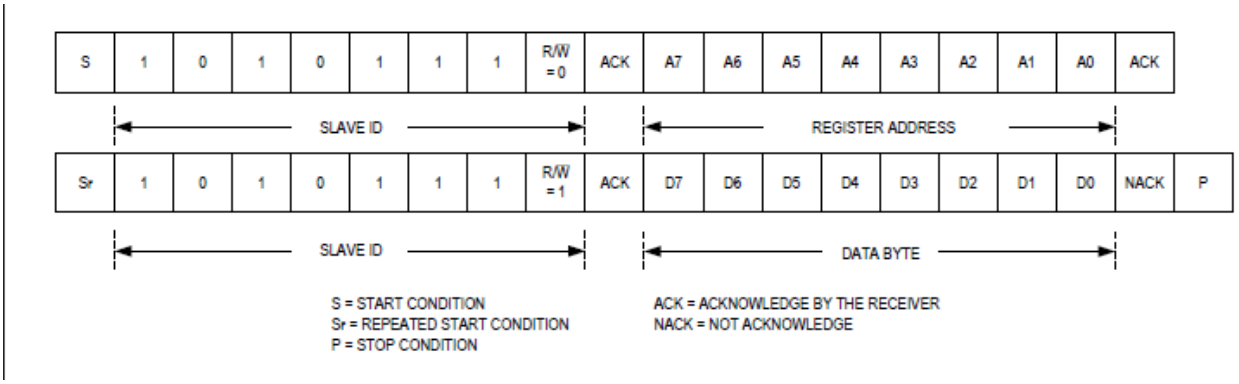


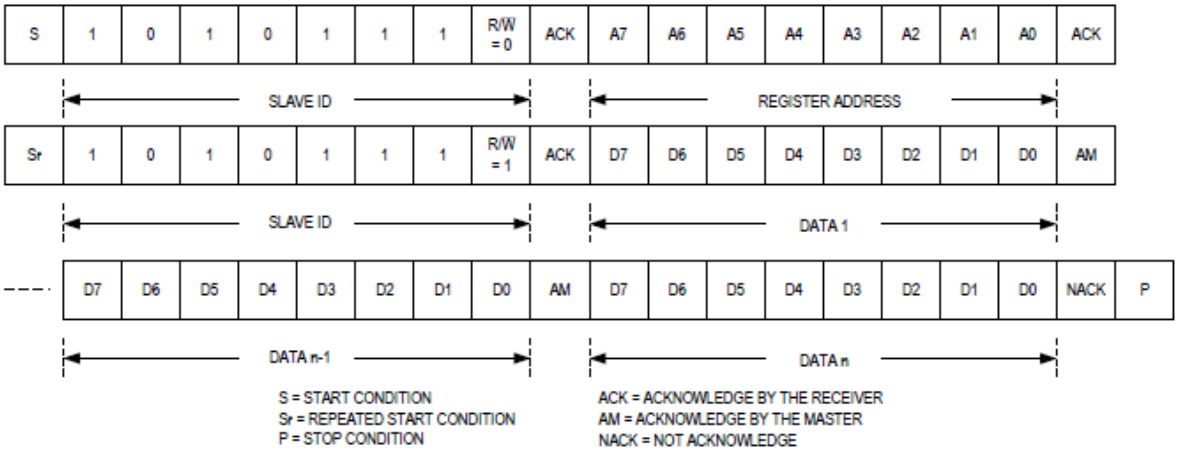
Внутренний указатель адреса регистра увеличивается автоматически, поэтому дополнительные байты данных заполняют регистры данных по порядку.

Для операции чтения необходимо выполнить две операции I2C. Сначала отправляется байт идентификатора slave, за которым следует регистр I2C, который нужно прочитать. Затем отправляется условие REPEAT START (Sr), за которым следует идентификатор работоспособного чтения. MAX30101 начинает отсылать данные, начиная с регистра, выбранного в первой операции. Указатель чтения увеличивается автоматически, поэтому MAX30101 продолжает отправлять данные из дополнительных регистров в последовательном порядке, пока не будет получено условие STOP (P). Исключеним явялется регистр данных FIFO_DATA в котором указатель чтения больше не увеличивается при чтении дополнительных байтов.

Чтобы прочитать следующий регистр после FIFO_DATA необходима команда записи I2C для изменения местоположения указателя чтения

Процесс чтения одного байта или нескольких байт данных:

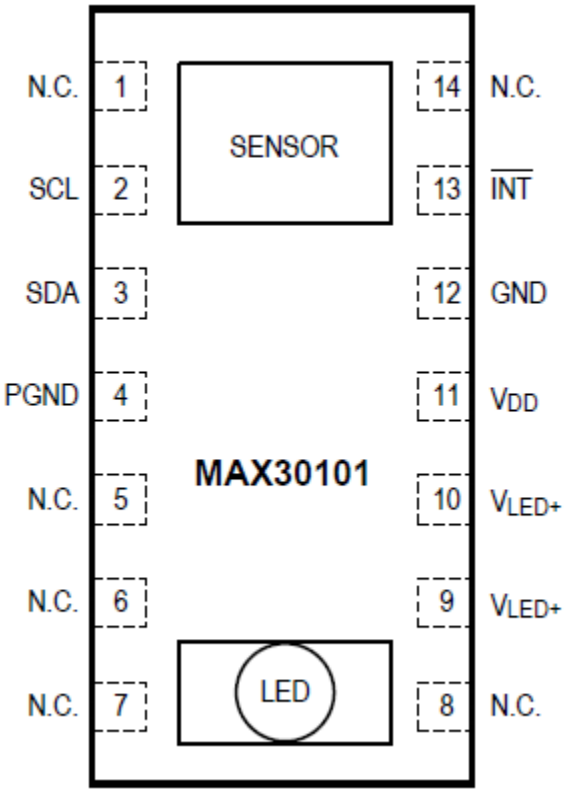




Для отправки адреса регистра чтения требуется операция начальной записи. Данные отправляются из регистров в последовательном порядке, начиная с регистра, выбранного в начальной операции записи I2C.

Если регистр FIFO_DATA прочитан то указатель чтения не будет автоматичке увеличиваться а последующие байты данных будут содержать содержимое FIFO.

Пин конфигурация



1,5,6,7,8,14 - NC нет коннекта, подключение к плате для механической стабильности

2 - SCL - I2C тактоый вход

3 - SDA - Clock data, двунаправленная (устройство с открытым стоком)

4 - PGND - Power ground светодиодных блоков драйверов

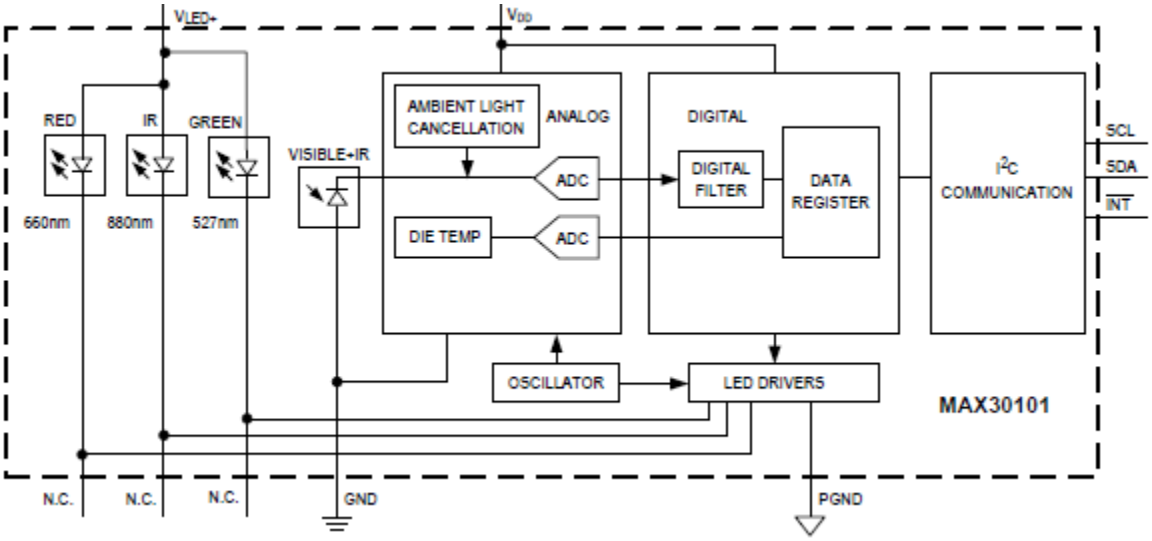
9,10 - Vled+ - светодиодный источник питания (анодное соединение), использовать bypass конденсатор для PGND для лучшей производительности

11 - Vdd - аналоговый источник питания, Вход, использавть bypass конденсатор для GND для лучшей производительности

12 - GND - Аналоговая земля

13 - INT - Active-low прерывание (устройство с открытым стоком), подключиться к внешнему напряжению с помощью резистора пульпы

Функциональная диограмма



MAX30101 полностью регулируется с помощью программных регистров

Цифровые выходные данные могут храниться в 32 deep FIFO внутри IC

FIFO

FIFO позволяет подключать MAX30101 к микроконтроллеру или процессору в общей щине где данные не считываеются непрерывано из регистров MAX30101

SpO2 подсистема

Содержит подавление окружающего света (**ALC**) , непрерывный сигма-дельта ADC и собственный фильтр времени

ALC имеет внутрениюю цепь Track / Hold чтобы отключить окружающий свет и увеличить эффективный динамический диапозон

SpO2 ADC имеет программируемые полномасштабные диапазоны от 2μA до 16μA

ALC может отменить до 200μA окружающего тока

Внутренний АЦП - конвертер сигма-дельта с 18 битным разрешением и постоянным временным превышением во времени

Температурный сенсор

MAX30101 имеет встроенный датчик температуры для калибровки температурной зависимости подсистемы SpO2

Выходные данные устройства относительно нечувствительны к длине волны IR светодиода где длина волны красного светодиона имеет решающее значение для правлиьной интерпретации данных

Алгоритм SpO2, используемый с выходным сигналом MAX30101, может компенсировать связанную с ним ошибку SpO2 изменениями температуры окружающей среды

Светодиодный драйвер

MAX30101 объединяет красные, зеленые, инфракрасные светодиодные драйверы для модуляции светодиодных импульсов для измерений SpO2 и HR (Heart Rate)

Светодиодный ток может быть запрограммирован от 0 до 50 мА при соответствующем напряжении питания.

Ширина светодиодного импульса может быть запрограммирована от 69μs до 411μs, чтобы алгоритм мог оптимизировать SpO2 и точность HR (Heart Rate) и энергопотребление на основе вариантов использования

Карта регистров и описание

REGISTER	B7	B6	B5	B4	B3	B2	B1	B0	REG ADDR	POR STATE	R/W
STATUS											
Interrupt Status 1	A_FULL	PPG_ RDY	ALC_ OVF					PWR_ RDY	0x00	0X00	R
Interrupt Status 2							DIE_TEMP _RDY		0x01	0x00	R
Interrupt Enable 1	A_FULL_ EN	PPG_ RDY_EN	ALC_ OVF_EN						0x02	0X00	R/W
Interrupt Enable 2							DIE_TEMP _RDY_EN		0x03	0x00	R/W
FIFO											
FIFO Write Pointer				FIFO_WR_PTR[4:0]					0x04	0x00	R/W
Overflow Counter				OVF_COUNTER[4:0]					0x05	0x00	R/W
FIFO Read Pointer				FIFO_RD_PTR[4:0]					0x06	0x00	R/W
FIFO Data Register	FIFO_DATA[7:0]								0x07	0x00	R/W
CONFIGURATION											
FIFO Configuration	SMP_AVE[2:0]			FIFO_ ROLL OVER_EN	FIFO_A_FULL[3:0]				0x08	0x00	R/W
Mode Configuration	SHDN	RESET				MODE[2:0]			0x09	0x00	R/W
SpO_2 Configuration	0 (Reserved)	SPO2_ADC_RGE [1:0]		SPO2_SR[2:0]			LED_PW[1:0]		0x0A	0x00	R/W
RESERVED									0x0B	0x00	R/W
LED Pulse Amplitude	LED1_PA[7:0]								0x0C	0x00	R/W
	LED2_PA[7:0]								0x0D	0x00	R/W
	LED3_PA[7:0]								0x0E	0x00	R/W
	LED4_PA[7:0]								0x0F	0x00	R/W
Multi-LED Mode Control Registers		SLOT2[2:0]				SLOT1[2:0]			0x11	0x00	R/W
		SLOT4[2:0]				SLOT3[2:0]			0x12	0x00	R/W

REGISTER	B7	B6	B5	B4	B3	B2	B1	B0	REG ADDR	POR STATE	R/W
RESERVED									0x13–0x17	0xFF	R/W
RESERVED									0x18-0x1E	0x00	R
DIE TEMPERATURE											
Die Temp Integer	TINT[7:0]								0x1F	0x00	R
Die Temp Fraction					TFRAC[3:0]				0x20	0x00	R
Die Temperature Config								TEMP_EN	0x21	0x00	R/W
RESERVED									0x22–0x2F	0x00	R/W
PART ID											
Revision ID	REV_ID[7:0]								0xFE	0xXX*	R
Part ID	PART_ID[7]								0xFF	0x15	R

Состояние прерывания 0x00 - 0x01

REGISTER	B7	B6	B5	B4	B3	B2	B1	B0	REG ADDR	POR STATE	R/W
Interrupt Status 1	A_FULL	PPG_RDY	ALC_OVF					PWR_RDY	0x00	0X00	R
Interrupt Status 2							DIE_TEMP_RDY		0x01	0x00	R

Всякий раз, когда прерывание запускается, MAX30101 выталкивает вывод активного-низкого прерывания в его низкое состояние до тех пор, пока прерывание не будет очищено.

A_FULL: FIFO Almost Full Flag - в режимах SpO2 и HR (Heart Rate) это прерываение срабатывает когда указатель записи FIFO имеет определенное количество свободных пробелов. Триггерный номер может быть задан регистром

FIFO_A_FULL[3:0]. Прерывание очищается при чтении регистра состояния прерывания1 (0x00)

PPG_RDY: New FIFO Data Ready - В режимах SpO2 и HR (Heart Rate) это прерывание срабатывает, когда появляется новый образец в данных FIFO. Прерывание очищается чтением регистра состояния прерывания 1 (0x00) или чтением регистра FIFO_DATA

ALC_OVF: Ambient Light Cancellation Overflow - Это прерывание срабатывает, когда функция подавления окружающего света фотодиода SpO2/HR (Heart Rate) достигла своего максимального предела, и поэтому окружающий свет влияет на выход ADC. Прерывание очищается при чтении регистра состояния прерывания 1 (0x00)

PWR_RDY: Power Ready Flag - При включении питания или после отключения питания, когда напряжение питания VDD переходит из-под напряжения блокировки низкого напряжения (UVLO) выше напряжения UVLO, включается готовый к питанию перерыв, чтобы подать сигнал о том, что модуль включен и готов к сбору данных.

DIE_TEMP_RDY: Internal Temperature Ready Flag - Когда внутреннее преобразование теемпературы заканчивается это прерываение запускается для того чтобы процессор смог прочитать регистры данных температуры. Прерывание очищается путем чтения регистра состояния прерывания 2 (0x01) или регистра TFRAC (0x20)

Прерывания очищаются при чтении регистра состояний прерываний или при чтении регистра, который вызвал прерывание. Например, если датчик SpO2 запускает прерывание из-за завершения преобразования, чтение данных регистра FIFO или регистра прерываний очищает вывод прерывания

(который возвращается в нормальное HIGH состояние). Это также очищает все биты в регистре состояния прерывания до нуля.

Включение прерывания 0x02-0x03

REGISTER	B7	B6	B5	B4	B3	B2	B1	B0	REG ADDR	POR STATE	R/W
Interrupt Enable 1	A_ FULL_ EN	PPG_ RDY_EN	ALC_ OVF_EN						0x02	0X00	R/W
Interrupt Enable 2							DIE_TEMP_ RDY_EN		0x03	0x00	R/W

Каждый источник аппаратного прерывания, за исключением готового питания, может быть отключен в реестре программного обеспечения в MAX30101 IC. Отключение питания невозможно из-за того, что цифровое состояние модуля сбрасывается при отключенном состоянии (низкое напряжение питания), а по умолчанию все прерывания отключаются.

Кроме того, важно, чтобы система знала, что произошло отключение, в результате чего данные внутри модуля сбрасываются.

Неиспользуемые биты всегда должны быть установлены в ноль для нормальной работы.

FIFO 0x04 - 0x07

REGISTER	B7	B6	B5	B4	B3	B2	B1	B0	REG ADDR	POR STATE	R/W
FIFO Write Pointer				FIFO_WR_PTR[4:0]					0x04	0x00	R/W
Over Flow Counter				OVF_COUNTER[4:0]					0x05	0x00	R/W
FIFO Read Pointer				FIFO_RD_PTR[4:0]					0x06	0x00	R/W
FIFO Data Register	FIFO_DATA[7:0]								0x07	0x00	R/W

FIFO Write Pointer - указыает на место где MAX30101 записывает следующий образец. Данный указатель продвигается для каждого образца который помщен в FIFO. Он может быть изменен через интерфейс I2C когда MODE[2:0] - 010, 011 или 111

FIFO Overflow Counter - Счетчик переполнения - когда FIFO заполнена образцы не перемещаются в FIFO и теряются. OVF_COUNTER подсчитывает количество потерянных образцов. Он наполняет при 0x1F. Когда полный образец "лопается" (например, удаление старых данных FIFO и перемещение образцов вниз) из FIFO (когда указатель на чтение продвигается), OVF_СЧЕТЧИК сбрасывается на ноль

FIFO Read Pointer - указывает на местоположение откуда процессор получает следующий образец от FIFO через интерфейс I2C, это происходит каждый раз когда образец изымается из FIFO. Процессор также может записывать на этот указатель после чтения образцов, чтобы разрешить повторное чтение образцов из FIFO, если есть ошибка передачи данных.

FIFO Data Register - глубина FIFO состовляет 32 и может содержать до 32 образцов данных

Размер выборки зависит от количества светодиодных каналов (a.k.a. каналы), настроенных как активные. Поскольку каждый сигнал канала хранится в виде 3-байтового сигнала, ширина FIFO может быть 3 байта, 6 байт, 9 байт или 12 байт.

Регистр FIFO_DATA в I2C карте регистра указывает на следующий образец, который будет прочитан из FIFO. **FIFO_RD_PTR** указывает на этот образец

Чтение FIFO_DATA регистра не увеличивает автоматически адрес регистра I2C

Каждый образец составляет 3 байта данных на канал (то есть 3 байта для RED, 3 байта для IR и т.д.).

Регистры FIFO (0x04-0x07) могут быть написаны и прочитаны, но на практике только регистр FIFO_RD_PTR должен быть написан. Остальные автоматически увеличиваются или заполняются данными с помощью MAX30101

При запуске нового SpO2 или преобразования сердечного ритма рекомендуется сначала очистить FIFO_WR_PTR, OVF_COUNTER и FIFO_RD_PTR регистры для всех нулей (0x00), чтобы убедиться, что FIFO пуст и в известном состоянии.

При чтении MAX30101 регистров в одной транзакции с разрывом-чтением I2C, указатель адреса регистра обычно увеличивается так, что следующий байт данных отправляется из следующего регистра и т.д. Исключением является регистр данных FIFO, регистр 0x07. При чтении этого регистра указатель адреса не увеличивается, но FIFO_RD_PTR делает это. Таким образом, следующий байт переданных данных представляет собой следующий байт данных, доступных в FIFO.

Чтение из FIFO

Обычно, чтение регистров из интерфейса I2C автоматически увеличивает указатель адреса регистра, так что все регистры могут быть прочитаны в виде разрыва чтения без события начала I2C. В MAX30101 это верно для всех регистров, за исключением регистра данных FIFO_ DATA (регистра 0x07).

При чтении регистра данных FIFO_DATA автоматически не увиличвает адрес регистра

Каждый образец содержит несколько байт данных поэтому несколько байт должны быть прочитаны из этого регистра в одной транзакции чтобы получить один полный образец

Другое исключение - 0xFF. Чтение дополнительных байтов после регистрации 0xFF не возвращает указатель адреса обратно в 0x00, и чтение данных не имеет смысла

Структура данных FIFO

Данные FIFO состоят из 32 образвогого банка памяти который может хранить данные GREEN, IR и RED ADC

Так как каждый образец состоит из трех каналов данных для каждого образца имеется 9 байт данных и поэтому в FIFO может храниться 288 полных байт данных

Данные FIFO выраниваются по левому краю то есть бит MSB всегда находится в позиции семнадцатого бита данных независимо от настройки разрешения ADC:

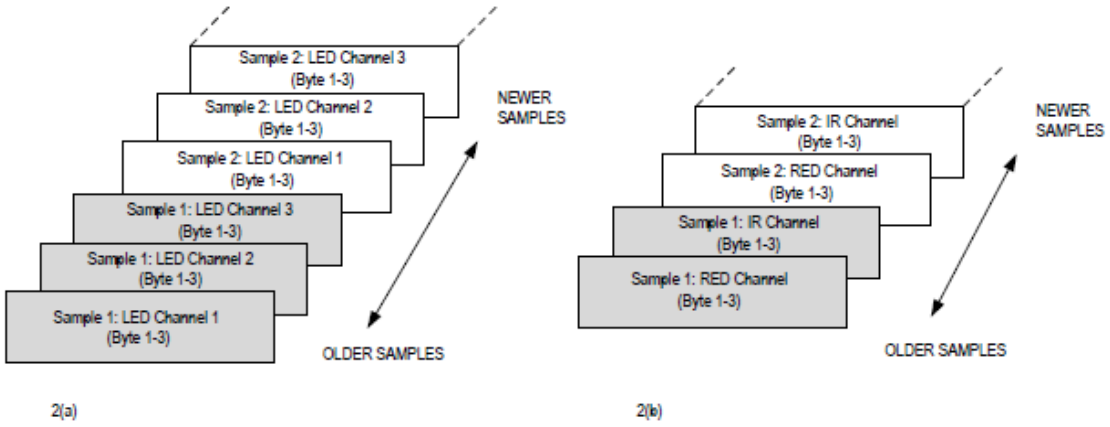
ADC Resolution	FIFO_DATA[17]	FIFO_DATA[16]	...	FIFO_DATA[12]	FIFO_DATA[11]	FIFO_DATA[10]	FIFO_DATA[9]	FIFO_DATA[8]	FIFO_DATA[7]	FIFO_DATA[6]	FIFO_DATA[5]	FIFO_DATA[4]	FIFO_DATA[3]	FIFO_DATA[2]	FIFO_DATA[1]	FIFO_DATA[0]
18-bit																
17-bit																
16-bit																
15-bit																

FIFO DATA[18] - [23] не используются

Данные FIFO содержат 3 байта на канал:

BYTE 1							FIFO_ DATA[17]	FIFO_ DATA[16]
BYTE 2	FIFO_ DATA[15]	FIFO_ DATA[14]	FIFO_ DATA[13]	FIFO_ DATA[12]	FIFO_ DATA[11]	FIFO_ DATA[10]	FIFO_ DATA[9]	FIFO_ DATA[8]
BYTE 3	FIFO_ DATA[7]	FIFO_ DATA[6]	FIFO_ DATA[5]	FIFO_ DATA[4]	FIFO_ DATA[3]	FIFO_ DATA[2]	FIFO_ DATA[1]	FIFO_ DATA[0]

Графическое представление FIFO регистра данных:



2a показывает три светодиода в multi-LED mode

2b показывает IR и Redd only in SpO2 mode

Каждый образец данных содержит две триплеты данных (по 3 байта каждый)

Для чтения одного образца требуется команда чтения I2C для каждого байта

Чтобы прочитать один образец в режиме SpO2 требуется шесть I2C байт чтения

Для чтения одного образца с тремя светодиодными каналами требуется 9 I2C байт чтения. Указатель чтения FIFO автоматически увеличивается после чтения первого байта каждого образца

Указатели на чтение и запись

Используются для управления потоком данных в FIFO

Указатель на запись увеличивается каждый раз когда новый образец добавляется в FIFO

Указтель на чтение увечилиывается кадлый раз когда образец читается из FIFO

Чтобы перечитать образец из FIFO нужно уменьшить его значение на единицу и прочитать регистр данных снова

Указатели записи/чтения FIFO должны быть очищены (обратно в режим 0x00) при входе в режим SpO2 или HR (Heart Rate)**, так что старые данные не представлены в FIFO. Указатели автоматически очищаются, если VDD является циклом питания или VDD падает ниже напряжения UVLO.**

Псевдо код чтения данных из FIFO

Получаем FIFO_WR_PTR:

```
START;
Send device address + write mode
Send address of FIFO_WR_PTR;
```

```
REPEATED_START;
Send device address + read mode
Read FIFO_WR_PTR;
STOP;
```

Центральный процессор оценивает количество образцов подлежащий считыванию из FIFO:

```
NUM_AVAILABLE_SAMPLES = FIFO_WR_PTR – FIFO_RD_PTR
(Note: pointer wrap around should be taken into account)
NUM_SAMPLES_TO_READ = < less than or equal to NUM_AVAILABLE_SAMPLES >
```

Вторая опеация - прочитать NUM_SAMPLES_TO_READ образов из FIFO:

```
START;
Send device address + write mode
Send address of FIFO_DATA;
REPEATED_START;
Send device address + read mode
for (i = 0; i < NUM_SAMPLES_TO_READ; i++) {
Read FIFO_DATA;
Save LED1[23:16];
Read FIFO_DATA;
Save LED1[15:8];
Read FIFO_DATA;
Save LED1[7:0];
Read FIFO_DATA;
Save LED2[23:16];
Read FIFO_DATA;
Save LED2[15:8];
Read FIFO_DATA;
Save LED2[7:0];
Read FIFO_DATA;
Save LED3[23:16];
Read FIFO_DATA;
Save LED3[15:8];
Read FIFO_DATA;
Save LED3[7:0];
Read FIFO_DATA;
}
STOP;
```

Третья операция - запись в регистр FIFO_RD_PTR

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

Если вторая операци была успешной то FIFO_RD_PTR указывает на следующий образец в FIFO и эта третья операция не нужна

FIFO конфигурация 0x08

REGISTER	B7	B6	B5	B4	B3	B2	B1	B0	REG ADDR	POR STATE	R/W
FIFO Configuration	SMP_AVE[2:0]			FIFO_ROLLOVER_EN	FIFO_A_FULL[3:0]				0x08	0x00	R/W

Биты 7 - 5 **Sample Averaging (SMP_AVE)** - для уменьшения пропускной способности соседние образцы (в кадом отдельном канале) могут быть усреднены и уничтожены на чипе установив этот регистр:

SMP_AVE[2:0]	NO. OF SAMPLES AVERAGED PER FIFO SAMPLE
000	1 (no averaging)
001	2
010	4
011	8
100	16
101	32
110	32
111	32

Бит 4 **FIFO Rolls on Full (FIFO_ROLLOVER_EN)** - управляет поедение FIFO когда FIFO полностью заполняется данными

Если FIFO_ROLLOVER_EN установлен (1), FIFO Адрес переходит на ноль, и FIFO продолжает заполняться новыми данными. Если бит не установлен (0), то FIFO не обновляется до тех пор, пока данные FIFO_DATA не будет прочитаны или позиции указателя WRITE/READ не будут изменены

Биты 3 - 0 **FIFO Almost Full Value (FIFO_A_FULL)** - данный регистр задает количество образцов данных (3 байта / образец) остающийхся в FIFO при выпуске прерываения.

Например если это поле установлено в 0x00 прерывание выдается когда в FIFO остается 0 образцов данных (все 32 слова FIFO имеют непрочитанные данные). Если это поле установлено в 0xF прерываение выдается когда 15 образов данных остаются в FIFO и соответственно 17 образцов данных FIFO имеют непрочитанные данные из 32

FIFO_A_FULL[3:0]	EMPTY DATA SAMPLES IN FIFO WHEN INTERRUPT IS ISSUED	UNREAD DATA SAMPLES IN FIFO WHEN INTERRUPT IS ISSUED
0x0h	0	32
0x1h	1	31
0x2h	2	30
0x3h	3	29
…	…	…
0xFh	15	17

Mode конфигурация 0x09

REGISTER	B7	B6	B5	B4	B3	B2	B1	B0	REG ADDR	POR STATE	R/W
Mode Configuration	SHDN	RESET				MODE[2:0]			0x09	0x00	R/W

Бит 7 **Shutdown Control (SHDN)** - часть может быть преведена в режим экономии энергии если установить этот бит в 1. В режиме энергосбережния все регистры сохраняют свои значения а операции записии и чтения функционируют как обычные, в данном режиме все прерывания очищаютя до нуля (?)

Бит 6 **Reset Control (Reset)** - когда данных бит установлен в 1 все регистры конфигурации, пороговые значения (?) и данные сбрасываются в состояние включения питания через сброс питания. Бит RESET автоматически очищается до нуля после завершения последовательности сброса

Установка бита RESET не вызывается событие прерывания PWR_RDY

Биты 2-0 **Mode control** - эти биты задают рабочее состояние MAX30101. Изменение режимов не меняет никаких других параметров и не стирает ранее хранившиеся данные регистрах данных:

MODE[2:0]	MODE	ACTIVE LED CHANNELS
000	Do not use	
001	Do not use	
010	Heart Rate mode	Red only
011	SpO2 mode	Red and IR
100–110	Do not use	
111	Multi-LED mode	Green, Red, and/or IR

SpO2 конфигурация 0x0A

REGISTER	B7	B6	B5	B4	B3	B2	B1	B0	REG ADDR	POR STATE	R/W
SpO_2 Configuration		SPO2_ADC_RGE[1:0]		SPO2_SR[2:0]			LED_PW[1:0]		0x0A	0x00	R/W

Биты 6-5 **SpO2 ADC Range Control** - этот регистр устанавливает полный дипозон датчика SpO2 ADC:

SPO2_ADC_RGE[1:0]	LSB SIZE (pA)	FULL SCALE (nA)
00	7.81	2048
01	15.63	4096
02	31.25	8192
03	62.5	16384

Биты 4-2 **SpO2 Sample Rate Control** - эти биты определяют эффективную частоту(rate) с помощью одного образца состоящего из одного IR pulse/ преобразования и одного RED pulse/преобразования

Rate образца и ширина pulse связаны тем что rate образца устанавливает верхнюю границу времени pulse

Если пользователь выбирает rate образца которая слишком высока для выбранного параметра LED_PW то в регистр программируется максимальная возможная rate образца

SPO2_SR[2:0]	SAMPLES PER SECOND
000	50
001	100
010	200
011	400
100	800
101	1000
110	1600
111	3200

Биты 1-0 **LED Pulse Width Control and ADC Resolution(Управление шириной импульса LED и разрешение ADC)** - задают ширину импульса светодиода (IR, Red, Green имеют одинановую ширину импульса) и следовательно косвенно определяют время интеграции АЦП в каждом образце

Разрешение АЦП напрямую сяззана со временем интеграции

LED_PW[1:0]	PULSE WIDTH (µs)	ADC RESOLUTION (bits)
00	69 (68.95)	15
01	118 (117.78)	16
10	215 (215.44)	17
11	411 (410.75)	18

Максимальная rate образца АЦП зависит от выбраннлй ширина pulse которая в свою очередь определяет разрешение АЦП

Если ширина pulse установлена в 69ms то разрешение ADC состовляет 15 бит и все rate образца можно выбирать, но если ширина pulse установлена в 411 ms то частота образца ограничена

Table 11. SpO_2 Mode (Allowed Settings)

SAMPLES PER SECOND	PULSE WIDTH (µs)			
	69	118	215	411
50	O	O	O	O
100	O	O	O	O
200	O	O	O	O
400	O	O	O	O
800	O	O	O	
1000	O	O		
1600	O			
3200				
Resolution (bits)	15	16	17	18

Table 12. HR Mode (Allowed Settings)

SAMPLES PER SECOND	PULSE WIDTH (µs)			
	69	118	215	411
50	O	O	O	O
100	O	O	O	O
200	O	O	O	O
400	O	O	O	O
800	O	O	O	O
1000	O	O	O	O
1600	O	O	O	
3200	O			
Resolution (bits)	15	16	17	18

Регистр амплитуды импульса светодиода 0x0C-0x0F

REGISTER	B7	B6	B5	B4	B3	B2	B1	B0	REG ADDR	POR STATE	R/W
LED Pulse Amplitude	LED1_PA[7:0]								0x0C	0x00	R/W
	LED2_PA[7:0]								0x0D	0x00	R/W
	LED3_PA[7:0]								0x0E	0x00	R/W
	LED4_PA[7:0]								0x0F	0x00	R/W

Эти биты задают текущий уровень каждого светодиода

Table 8. LED Current Control

LEDx_PA [7:0]	TYPICAL LED CURRENT (mA)*
0x00h	0.0
0x01h	0.2
0x02h	0.4
…	…
0x0Fh	3.0
…	…
0x1Fh	6.2
…	…
0x3Fh	12.6
…	…
0x7Fh	25.4
…	…
0xFFh	51.0

Multi-светодиодный Mode Control регистры 0x11-0x12

REGISTER	B7	B6	B5	B4	B3	B2	B1	B0	REG ADDR	POR STATE	R/W
Multi-LED Mode Control Registers		SLOT2[2:0]				SLOT1[2:0]			0x11	0x00	R/W
		SLOT4[2:0]				SLOT3[2:0]			0x12	0x00	R/W

Каждый образец разделяется на четыре временных слота: SLOT1-SLOT4

Эти управляющие регистры определяют, какой светодиод активен в каждом временном интервале, что обеспечивает очень гибкую конфигурацию

Table 9. Multi-LED Mode Control Registers

SLOTx[2:0] Setting	WHICH LED IS ACTIVE	LED PULSE AMPLITUDE SETTING
000	None (time slot is disabled)	N/A (Off)
001	LED1 (RED)	LED1_PA[7:0]
010	LED2 (IR)	LED2_PA[7:0]
011*	LED3 (GREEN)	LED3_PA[7:0]
	LED4 (GREEN)	LED4_PA[7:0]
100	None	N/A (Off)
101	RESERVED	RESERVED
110	RESERVED	RESERVED
111	RESERVED	RESERVED

Каждый слот генерирует 3байтовый выход в FIFO

Один образец содержит все активные слоты например SLOT1 и SLOT2 ненулевые то один образец составляет 2*3 = 6 Байт

Если SLOT1 - SLOT3 ненулевые то один образец равен 3*3 = 9 Байт

Слоты должны быть включены в порядке то есть SLOT1 не должен быть отключен если включены SLOT2 или SLOT3

Звездочка: LED3 и LED4 подключены к Green LED. Green LED поглотители тока из LED3_PA[7:0] и LED4_PA[7:0] конфигурация Multi-LE Mode и SLOTx[2:0] = 011

Температурные данные 0x1F-0x21

REGISTER	B7	B6	B5	B4	B3	B2	B1	B0	REG ADDR	POR STATE	R/W
Temp_Integer	TINT[7]								0x1F	0x00	R/W
Temp_Fraction					TFRAC[3:0]				0x20	0x00	R/W
Die Temperature Config								TEMP_EN	0x21	0x00	R/W

Temperatire Integer - бортовый температурные выход АЦП разделен на два регистра: один для хранения целочисленной температуры и один для хранения фракции. **Оба регистра следует учитывать при чтении данных о температуре**

Уравнение которое показыает как объединить эти два регистра: получаем измеряемую температуру

$$T_{MEASURED} = T_{INTEGER} + T_{FRACTION}$$

Этот регистр хранит данные о целочисленных температурах в формате дополнения двух где каждый бит соответствует одному градусу цельсия

Temerature Integer:

REGISTER VALUE (hex)	TEMPERATURE (°C)
0x00	0
0x00	+1
...	...
0x7E	+126
0x7F	+127
0x80	-128
0x81	-127
...	...
0xFE	-2
0xFF	-1

Temperature Fraction - этот регистр хранит данные о дробной температуре с шагом 0.00625 градусов цельсия

Если эта дробная температура сопоставляется с отрицательным целым числом то она все равно добавляет положительное дробное значение например -128C + 0.5C = -127.5C

Temperature Enable (TEMP_EN) - данный бит при установке инициирует однократное измерение температуры с датчика температуры, он автоматически возвращается к нулю при завершении чтения температуры когда бит установлен в 1

SpO2 температурные компенсации

MAX30101 имеет точный бортовой датчик температуры, который оцифровывает внутреннюю температуру IC по команде мастера I2C. Температура влияет на длину волны красных и инфракрасных светодиодов. Хотя выходные данные устройства относительно нечувствительны к длине волны инфракрасного светодиода, длина волны красного светодиода имеет решающее значение для правильной интерпретации данных.

Так как светодиодная матрица нагревается с очень короткой температурной константой времени (десятки микросекунд), длина светодиодной волны должна быть рассчитана в соответствии с текущим уровнем светодиода и температурой IC

Соотношение длины волны красного светодиода и температуры светодиода:

Table 13. RED LED Current Settings vs. LED Temperature Rise

RED LED CURRENT SETTING	RED LED DUTY CYCLE (% OF LED PULSE WIDTH TO SAMPLE TIME)	ESTIMATED TEMPERATURE RISE (ADD TO TEMP SENSOR MEASUREMENT) (°C)
00000001 (0.2mA)	8	0.1
11111010 (50mA)	8	2
00000001 (0.2mA)	16	0.3
11111010 (50mA)	16	4
00000001 (0.2mA)	32	0.6
11111010 (50mA)	32	8

Второй столбец- рабочий цикл красного светодиода (% от ширины pulse светодиода до времени отбора проб)

Третий столбец - расчетное повышение температуры (добаить к измерению датчика температуры)

Добавьте это к показаниям температуры модуля, чтобы оценить температуру светодиода и длину выходной волны. Оценка температуры светодиода

действительна даже при очень короткой ширине pulse, благодаря высокой температурной константе светодиода.

Сроки проведения измерений и сбора данных

MAX30101 может поддерживать до трех светодиодных каналов последовательной обработки (красный, IR, зеленый)

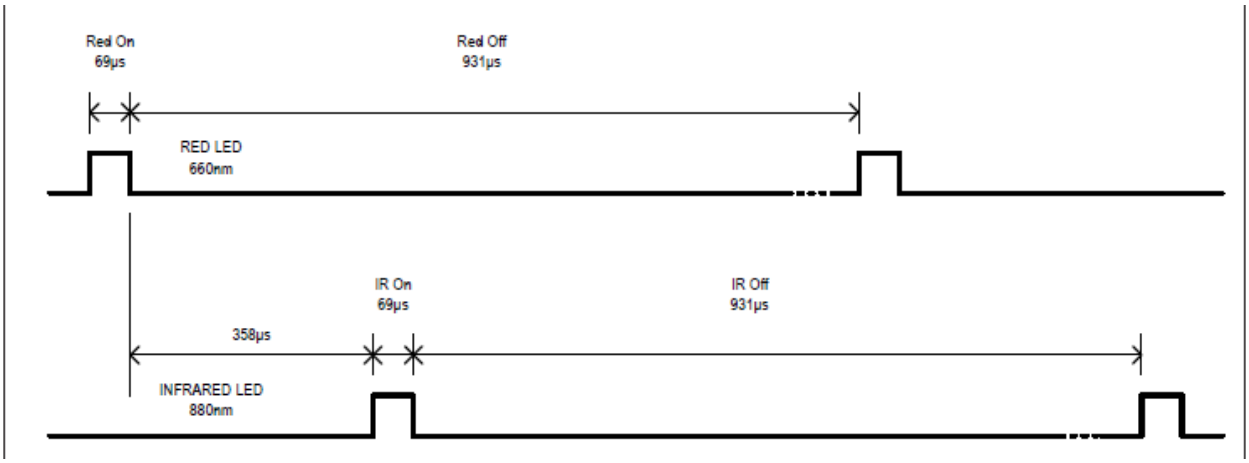
В multi-led modes временной интервал или период существует между активными последовательными каналами

Четыре возможных интервала времени связанных с каждой настройкой ширины pulse:

Table 14. Slot Timing

PULSE-WIDTH SETTING (µs)	CHANNEL SLOT TIMING (TIMING PERIOD BETWEEN PULSES) (µs)	CHANNEL-CHANNEL TIMING (RISING EDGE-TO-RISING EDGE) (µs)
69	358	427
118	407	525
215	505	720
411	696	1107

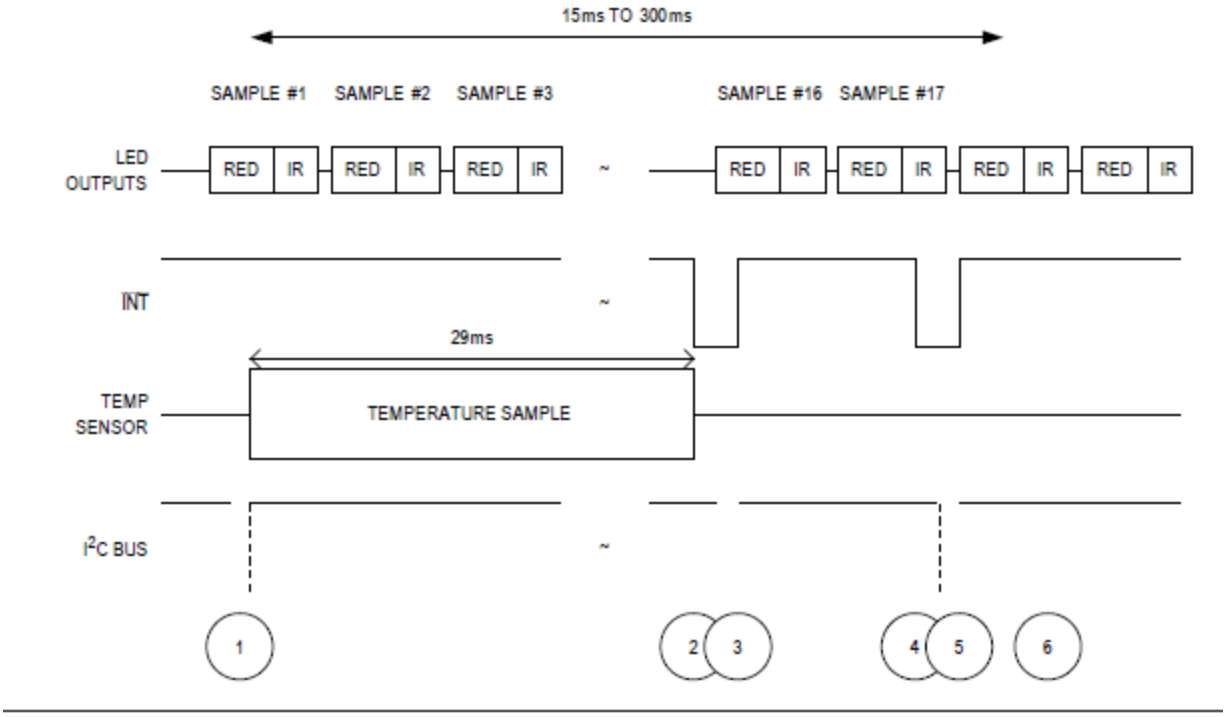
Пример времени канала слота для приложения режима SpO2 с rate образца 1kHz:



Время в SpO2 mode

SpO2 может быть откалиброван с помощью температурных данных. В этом случае температуру не нужно пробовать очень часто - раз в секунду или каждые несколько секунд должно быть достаточно.

Время получения и передачи данных в режиме SpO2:



1 событие - Переход в режим SpO2. Начало измерения температуры. I2C Write Command устанавливает MODE[2:0] = 0x03 и устанавливает A_FULL_EN. Затем, чтобы включить и инициировать единое измерение температуры, установите TEMP_EN и DIE_TEMP_RDY_EN.

2 событие - Измерение температуры завершено, генерируется прерывание. Триггеры прерывания DIE_TEMP_RDY предупреждают центральный процессор о необходимости чтения данных.

3 событие - Временные данные считываются, прерывание очищается.

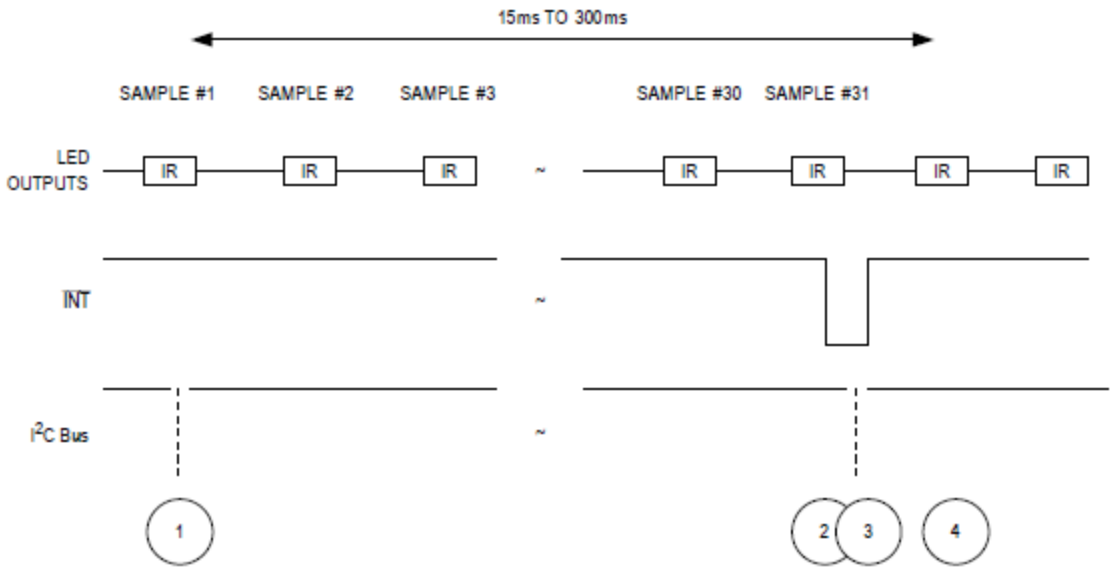
4 событие - FIFO почти заполнен, прерывание генерируется. Прерывание генерируется при достижении FIFO почти полного порога.

5 событие - Данные FIFO считываются, прерывание очищается

6 событие - Следующий образец записан. Новый образец хранится в новом месте указателя чтения. Фактически, теперь это первый образец в FIFO

Время в Heart Rate mode

Пользователь может выбрать либо красный, IR или зеленый светодиодный канал для сердечного ритма.



1 событие - переходим в режим. В I2C пишем команду MODE[2:0] = 0x02. Mask the A_FULL_EN Interrupt (?)

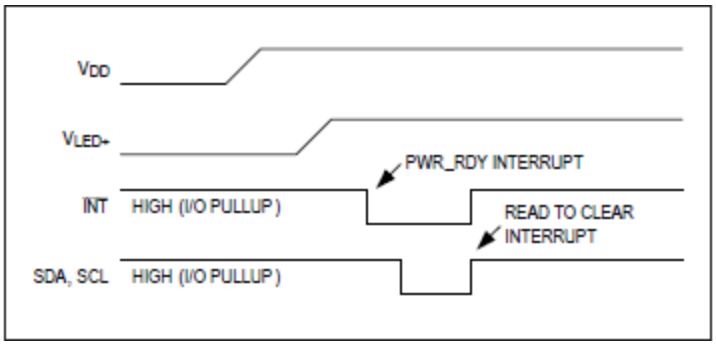
2 событие - FIFO почти заполнен, прерывание генерируется. Прерывание генерируется, когда FIFO имеет только одно свободное пространство.

3 событие - Данные FIFO считываются, прерывание очищается.

4 событие - Следующий образец записан. Новый образец хранится в новом расположении указателя чтения. Фактически, теперь это первый образец в FIFO.

Последовательность питания

Рекомендуемая последовательность включения для MAX30101:



Сначала рекомендуется включить VDD-питание, а затем светодиодные источники питания (VLED+).

Прерывания и I2C пины можно натянуть на внешнее напряжение, даже если источники питания не включены.

После установки питания происходит прерывание, чтобы предупредить систему о готовности MAX30101 к работе. Чтение регистра прерываний I2C очищает прерывание

Отключение питания может быть в любой последовательности