

3 Обзор продукта

3.1 MPU-60X0 Обзор

MotionInterface™ становится обязательной функцией, используемой производителями смартфонов и планшетов из-за огромной ценности, которую он добавляет к удобству конечного пользователя. В смартфонах он находит применение в таких приложениях, как жестовые команды для приложений и управления телефоном, улучшенные игры, дополненная реальность, панорамный захват и просмотр фотографий, а также навигация пешеходов и транспортных средств. Обладая способностью точно и точно отслеживать движения пользователей, технология MotionTracking может преобразовывать телефоны и планшеты в мощные интеллектуальные 3D-устройства, которые могут использоваться в приложениях, начиная от мониторинга состояния здоровья и фитнеса и заканчивая услугами на основе определения местоположения. Ключевыми требованиями для устройств с поддержкой MotionInterface являются небольшой размер упаковки, низкое энергопотребление, высокая точность и повторяемость, высокая устойчивость к ударам и программируемость производительности в зависимости от приложения - и все это по низкой потребительской цене.

MPU-60X0 - это первое в мире интегрированное 6-осевое устройство MotionTracking, которое объединяет 3-осевой гироскоп, 3-осевой акселерометр и цифровой процессор движения (DMP) в небольшом корпусе 4x4x0,9 мм. Благодаря выделенной сенсорной шине I2C, он напрямую принимает входы от внешнего 3-осевого компаса, чтобы обеспечить полный 9-осевой выход MotionFusion™. Устройство MPU-60X0 MotionTracking с его 6-осевой интеграцией, встроенным MotionFusion™ и встроенным программным обеспечением для калибровки позволяет производителям исключить дорогостоящий и сложный выбор, квалификацию и интеграцию на уровне системы дискретных устройств, гарантируя оптимальное движение производительность для потребителей. MPU-60X0 также предназначен для взаимодействия с несколькими неинерциальными цифровыми датчиками, такими как датчики давления, на его вспомогательном порте I2C. MPU-60X0 совместима с семейством MPU-30X0.

MPU-60X0 имеет три 16-разрядных аналого-цифровых преобразователя (АЦП) для оцифровки выходов гироскопа и три 16-разрядных АЦП для оцифровки выходов акселерометра. Для точного отслеживания как быстрых, так и медленных движений детали оснащены программируемым пользователем масштабным диапазоном гироскопа ± 250 , ± 500 , ± 1000 и ± 2000 °/сек (dps) и программируемым пользователем акселерометром в полном масштабе диапазон ± 2 g, ± 4 g, ± 8 g и ± 16 g.

Встроенный 1024-байтовый буфер FIFO помогает снизить энергопотребление системы, позволяя системному процессору считывать данные датчика в пакетах и затем переходить в режим пониженного энергопотребления, когда MPU собирает больше данных. Со всеми необходимыми компонентами обработки на чипе и сенсорами, необходимыми для поддержки многих сценариев использования на основе движения, MPU-60X0 уникальным образом позволяет использовать приложения MotionInterface с низким энергопотреблением в портативных приложениях с уменьшенными требованиями к обработке для системного процессора. Благодаря встроенному выходу MotionFusion, DMP в MPU-60X0 снимает

с компьютера системного процессора требования к интенсивным вычислениям MotionProcessing, сводя к минимуму необходимость частого опроса выходного сигнала датчика движения.

Связь со всеми регистрами устройства осуществляется с использованием I2C на частоте 400 кГц или SPI на частоте 1 МГц (только MPU-6000). Для приложений, требующих более быстрой связи, сенсор и регистры прерываний могут считываться с использованием SPI на частоте 20 МГц (только для MPU-6000). Дополнительные функции включают в себя встроенный датчик температуры и встроенный генератор с отклонением $\pm 1\%$ в диапазоне рабочих температур.

Используя свою запатентованную и проверенную на практике платформу Nasiri-Fabrication, которая объединяет пластины MEMS с сопутствующей электроникой CMOS посредством соединения на уровне пластин, InvenSense позволил уменьшить размер пакета MPU-60X0 до революционной площади 4x4x0,9 мм (QFN), в то время как обеспечивает наивысшую производительность, низкий уровень шума и самую низкую стоимость полупроводниковой упаковки, необходимой для портативных потребительских электронных устройств. Деталь обладает устойчивостью к ударным нагрузкам в 10 000 g и имеет программируемые фильтры нижних частот для гироскопов, акселерометров и встроенного датчика температуры.

Для обеспечения гибкости электропитания MPU-60X0 работает в диапазоне напряжений питания VDD от 2,375 до 3,46 В. Кроме того, MPU-6050 имеет опорный вывод VLOGIC (в дополнение к его аналоговому выводу питания: VDD), который устанавливает логические уровни интерфейса I2C. Напряжение VLOGIC может составлять $1,8 \text{ В} \pm 5\%$ или VDD.

MPU-6000 и MPU-6050 идентичны, за исключением того, что MPU-6050 поддерживает только последовательный интерфейс I2C и имеет отдельный опорный вывод VLOGIC. MPU-6000 поддерживает интерфейсы I2C и SPI и имеет один вывод питания VDD, который является как логическим эталонным источником питания устройства, так и аналоговым источником питания детали. В таблице ниже приведены эти различия:

Primary Differences between MPU-6000 and MPU-6050

Part / Item	MPU-6000	MPU-6050
VDD	2.375V-3.46V	2.375V-3.46V
VLOGIC	n/a	1.71V to VDD
Serial Interfaces Supported	I ² C, SPI	I ² C
Pin 8	/CS	VLOGIC
Pin 9	AD0/SDO	AD0
Pin 23	SCL/SCLK	SCL
Pin 24	SDA/SDI	SDA

4 Приложения

- Технология BlurFree™ (для стабилизации видео / фото)
- Технология AirSign™ (для безопасности / аутентификации)
- Технология TouchAnywhere™ (для управления приложениями пользовательского интерфейса «без прикосновения» / навигации)
- Технология MotionCommand™ (для ярлыков жестов)

- Игра с поддержкой движений и инфраструктура приложений
- InstantGesture™ iG™ распознавание жестов
- Услуги на основе местоположения, достопримечательности и безрассудный расчет
- Телефон и портативные игры
- Игровые контроллеры на основе движений
- 3D-пульты дистанционного управления для подключенных к Интернету телевизоров и телевизионных приставок, 3D-мыши
- Носимые датчики для здоровья, фитнеса и спорта
- Игрушки

5 Особенности

5.1 Особенности гироскопа

Трехосный MEMS-гироскоп в MPU-60X0 имеет широкий спектр функций:

- Цифровые выходные датчики угловой скорости осей X, Y и Z (гироскопы) с программируемым пользователем диапазоном полной шкалы ± 250 , ± 500 , ± 1000 и ± 2000 ° / сек.
- Внешний сигнал синхронизации, подключенный к выводу FSYNC, поддерживает синхронизацию изображения, видео и GPS
- Интегрированные 16-битные АЦП обеспечивают одновременную выборку гироскопов
- Улучшенное смещение и температурная стабильность чувствительности уменьшают потребность в пользовательской калибровке
- Улучшенные характеристики низкочастотного шума
- Цифровой программируемый фильтр нижних частот
- Рабочий ток гироскопа: 3,6 мА
- ток в режиме ожидания: 5 мкА
- Откалиброванный на заводе коэффициент чувствительности
- Самопроверка пользователя

5.2 Особенности акселерометра

Трехосный MEMS-акселерометр в MPU-60X0 обладает широким спектром возможностей:

- Трехосевой акселерометр с цифровым выходом и программируемым диапазоном полной шкалы ± 2 g, ± 4 g, ± 8 g и ± 16 g
- Интегрированные 16-разрядные АЦП обеспечивают одновременную выборку акселерометров, не требуя внешнего мультиплексора
- Нормальный рабочий ток акселерометра: 500 мкА
- Ток в режиме акселерометра малой мощности: 10 мкА при 1,25 Гц, 20 мкА при 5 Гц, 60 мкА при 20 Гц, 110 мкА при 40 Гц
- Обнаружение ориентации и сигнализации
- Обнаружение касания
- Программируемые пользователем прерывания
- High-G прерывание
- Самопроверка пользователя

5.3 Дополнительные функции

MPU-60X0 включает в себя следующие дополнительные функции:

- 9-осевой MotionFusion с помощью встроенного цифрового процессора движения (DMP)
 - Вспомогательная главная шина I2C для считывания данных с внешних датчиков (например, магнитометр)
 - Рабочий ток 3,9 мА, когда все 6 осей измерения движения и DMP включены
 - Диапазон напряжения питания VDD 2,375 В - 3,46 В
 - Гибкая VLOGIC опорного напряжения поддерживает множественные напряжения I2C интерфейса (MPU-6050 только)
 - Самый маленький и самый тонкий пакет QFN для портативных устройств: 4x4x0,9 мм
 - Минимальная перекрестная чувствительность между осями акселерометра и гироскопа
 - 1024-байтовый буфер FIFO снижает энергопотребление, позволяя хост-процессору читать данные пакетами и затем переходить в режим пониженного энергопотребления, поскольку MPU собирает больше данных
 - Датчик температуры с цифровым выходом
 - Программируемые пользователем цифровые фильтры для гироскопа, акселерометра и датчика температуры
 - 10 000 g толерантного к шоку
 - Быстрый режим I2C 400 кГц для связи со всеми регистрами
 - Последовательный интерфейс SPI 1 МГц для связи со всеми регистрами (только MPU-6000)
 - Последовательный интерфейс SPI 20 МГц для считывания датчиков и регистров прерываний (только MPU-6000)
 - Структура MEMS герметично закрыта и скреплена на уровне пластин
 - Соответствует RoHS и Green

5.4 MotionProcessing

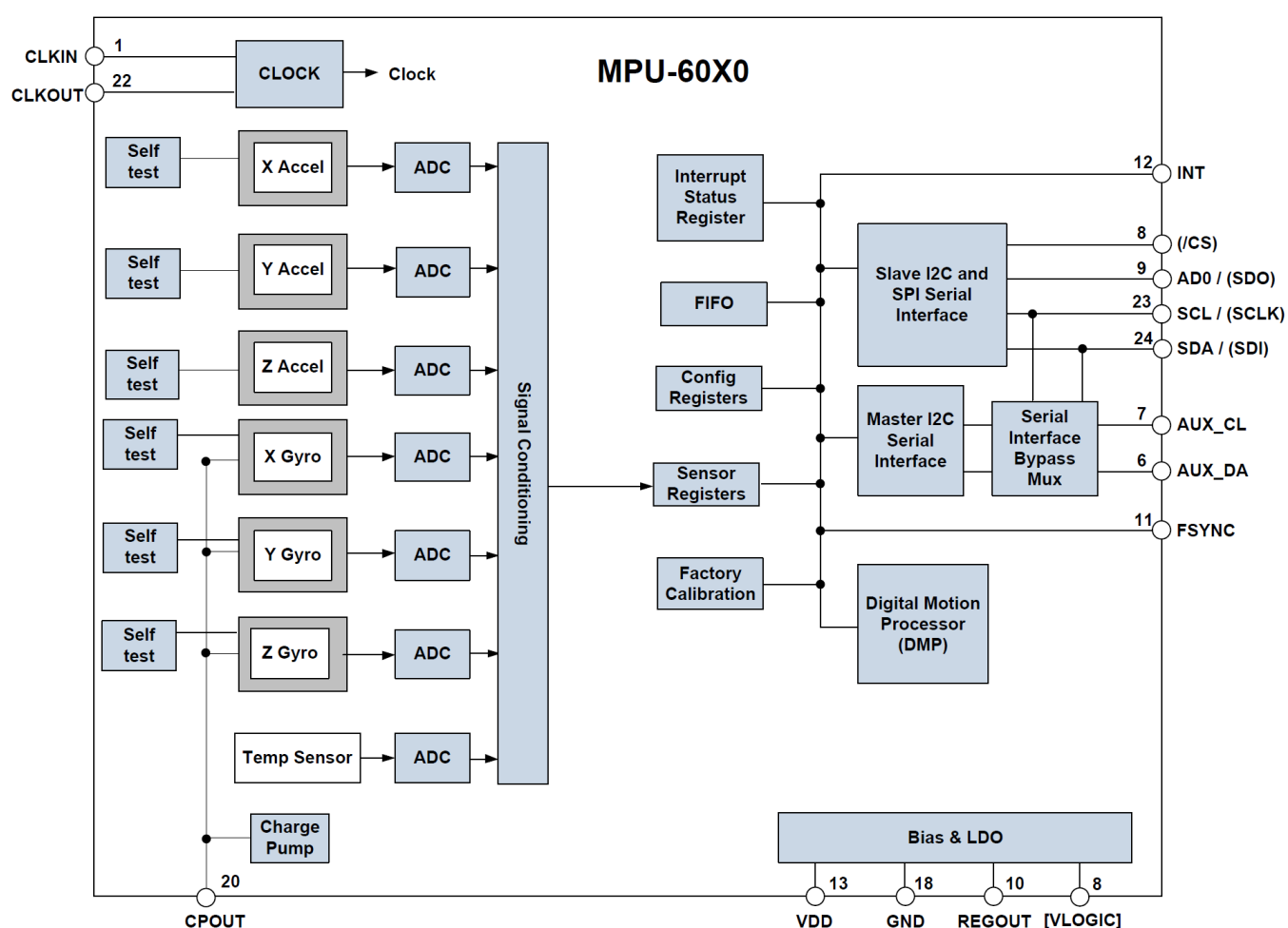
- Механизм внутренней цифровой обработки движения™ (DMP™) поддерживает алгоритмы трехмерной обработки движения и распознавания жестов.
- MPU-60X0 собирает данные гироскопа и акселерометра, синхронизируя выборку данных с определенной пользователем скоростью. Полный набор данных, полученный с помощью MPU-60X0, включает в себя данные 3-осевого гироскопа, данные 3-осевого акселерометра и данные о температуре. Расчетный выход MPU для системного процессора может также включать данные о курсе с цифрового 3-осевого стороннего магнитометра.
- FIFO буферизирует полный набор данных, уменьшая требования к синхронизации системного процессора, позволяя пакетному процессору считывать данные FIFO. После пакетного чтения данных FIFO системный процессор может экономить электроэнергию, переходя в режим ожидания с низким энергопотреблением, в то время как MPU собирает больше данных.
- Программируемое прерывание поддерживает такие функции, как распознавание жестов, панорамирование, масштабирование, прокрутка, обнаружение касаний и обнаружение сотрясений

- Цифровые программируемые фильтры нижних частот
- Функция шагомера с низким энергопотреблением позволяет хост-процессору спать, пока DMP поддерживает количество шагов.

5.5 Синхронизация

- Встроенный генератор синхронизации $\pm 1\%$ изменения частоты во всем диапазоне температур
- Дополнительные входы внешней тактовой частоты 32,768 кГц или 19,2 МГц

7.5 Block Diagram



Note: Pin names in round brackets () apply only to MPU-6000
Pin names in square brackets [] apply only to MPU-6050

7.6 Обзор

MPU-60X0 состоит из следующих ключевых блоков и функций:

- Трехосевой MEMS-датчик скорости гироскопа с 16-разрядными АЦП и формированием сигнала
- Трехосевой датчик MEMS-акселерометр с 16-разрядными АЦП и формированием сигнала
- Процессор цифрового движения (DMP)
- Первичные интерфейсы последовательной связи I2C и SPI (только MPU-6000)
- Вспомогательный последовательный интерфейс I2C для стороннего магнитометра и других датчиков
- Синхронизация
- Регистры данных датчика
- FIFO
- Прерывания
- Датчик температуры с цифровым выходом
- Само тестирование гироскопа и акселерометра
- Смещение и LDO
- Зарядный насос

7.7 Трехосевой MEMS гироскоп с 16-разрядными АЦП и формированием сигнала

MPU-60X0 состоит из трех независимых вибрационных MEMS-скоростных гироскопов, которые обнаруживают вращение вокруг осей X, Y и Z. Когда гироскопы вращаются вокруг любой из осей чувствительности, эффект Кориолиса вызывает вибрацию, которая обнаруживается емкостным датчиком. Результирующий сигнал усиливается, демодулируется и фильтруется для получения напряжения, пропорционального угловой скорости. Это напряжение оцифровывается с использованием отдельных встроенных 16-разрядных аналого-цифровых преобразователей (АЦП) для выборки каждой оси. Полномасштабный диапазон гироскопических датчиков может быть запрограммирован в цифровой форме до ± 250 , ± 500 , ± 1000 или ± 2000 градусов в секунду (dps). Частота дискретизации АЦП программируется от 8000 выборок в секунду до 3,9 выборок в секунду, а выбираемые пользователем фильтры нижних частот обеспечивают широкий диапазон частот среза.

7.8 Трехосевой MEMS-акселерометр с 16-разрядными АЦП и формированием сигнала

Трехосевой акселерометр MPU-60X0 использует отдельные контрольные массы для каждой оси. Ускорение вдоль определенной оси вызывает смещение на соответствующей контрольной массе, а емкостные датчики определяют смещение по-разному. Архитектура MPU-60X0 снижает восприимчивость акселерометров к изменениям изготовления, а также к тепловому дрейфу. Когда устройство расположено на плоской поверхности, оно будет измерять 0g по осям X и Y и + 1g по оси Z. Масштабный коэффициент акселерометров откалиброван на заводе и номинально не зависит от напряжения питания. Каждый датчик имеет специальный сигма-дельта АЦП для обеспечения цифровых выходов. Диапазон полной шкалы цифрового выхода может быть настроен на $\pm 2g$, $\pm 4g$, $\pm 8g$ или $\pm 16g$.

7.9 Цифровой процессор движения

Встроенный цифровой процессор движения (DMP) расположен в MPU-60X0 и выгружает вычисления алгоритмов обработки движения из главного процессора. DMP получает данные от акселерометров, гироскопов и дополнительных сторонних датчиков, таких как магнитометры, и обрабатывает данные. Полученные данные могут быть считаны из регистров DMP или могут быть помещены в буфер в FIFO. DMP имеет доступ к одному из внешних выводов MPU, который можно использовать для генерации прерываний.

Назначение DMP - снять с хост-процессора как требования к синхронизации, так и вычислительную мощность. Как правило, алгоритмы обработки движения должны выполняться с высокой частотой, часто около 200 Гц, чтобы обеспечить точные результаты с низкой задержкой. Это необходимо, даже если приложение обновляется с гораздо меньшей скоростью; например, пользовательский интерфейс с низким энергопотреблением может обновляться с частотой всего 5 Гц, но обработка движения все равно должна выполняться на частоте 200 Гц. DMP можно использовать в качестве инструмента для минимизации энергопотребления, упрощения синхронизации, упрощения архитектуры программного обеспечения и сохранения ценных MIPS на главном процессоре для использования в приложении.

7.10 Первичные интерфейсы последовательной связи I2C и SPI

MPU-60X0 связывается с системным процессором, используя SPI (только MPU-6000) или последовательный интерфейс I2C. MPU-60X0 всегда действует как подчиненное устройство при связи с системным процессором. Младший бит адреса подчиненного устройства I2C устанавливается контактом 9 (AD0).

Логические уровни для связи между MPU-60X0 и его ведущим устройством следующие:

- MPU-6000: логический уровень для связи с ведущим задается напряжением на VDD
- MPU-6050: логический уровень для связи с мастером задается напряжением на VLOGIC

Для получения дополнительной информации о логических уровнях MPU-6050, пожалуйста, обратитесь к Разделу 10.

7.11 Вспомогательный последовательный интерфейс I2C

MPU-60X0 имеет вспомогательную шину I2C для связи с 3-осевым цифровым выходным магнитометром или другими датчиками. Эта шина имеет два режима работы:

- Основной режим I2C: MPU-60X0 действует как главный для любых внешних датчиков, подключенных к вспомогательной шине I2C.
- Режим сквозного прохождения: MPU-60X0 напрямую соединяет первичную и вспомогательную шины I2C, что позволяет системному процессору напрямую связываться с любыми внешними датчиками.

Режимы работы вспомогательной шины I2C:

- Основной режим I2C: позволяет MPU-60X0 напрямую обращаться к регистрам данных внешних цифровых датчиков, таких как магнитометр. В этом ре-

жиме MPU-60X0 напрямую получает данные от вспомогательных датчиков, что позволяет встроенному DMP генерировать данные объединения датчиков без вмешательства со стороны процессора системных приложений.

Например, в режиме I2C Master MPU-60X0 может быть сконфигурирован для выполнения пакетного считывания, возвращая следующие данные из магнитометра:

X данные магнитометра (2 байта)

Y данные магнитометра (2 байта)

Z данные магнитометра (2 байта)

I2C Master может быть сконфигурирован для считывания до 24 байтов с 4 дополнительных датчиков. Пятый датчик может быть настроен для работы однобайтового режима чтения / записи.

- Сквозной режим: позволяет внешнему системному процессору действовать в качестве главного и напрямую связываться с внешними датчиками, подключенными к вспомогательным контактам шины I2C (AUX_DA и AUX_CL). В этом режиме логика управления вспомогательной шиной I2C (интерфейсный блок датчика стороннего производителя) MPU-60X0 отключена, а вспомогательные выводы I2C AUX_DA и AUX_CL (контакты 6 и 7) подключены к главной шине I2C (контакты 23 и 24) через аналоговые переключатели.

Сквозной режим полезен для настройки внешних датчиков или для поддержания MPU-60X0 в режиме пониженного энергопотребления, когда используются только внешние датчики.

В сквозном режиме системный процессор все еще может получать доступ к данным MPU-60X0 через интерфейс I2C.

Уровни логики вспомогательной шины I2C IO

- MPU-6000: логический уровень вспомогательной шины I2C - VDD
- MPU-6050: логический уровень вспомогательной шины I2C может быть запрограммирован на VDD или VLOGIC.

Для получения дополнительной информации о логических уровнях MPU-6050, пожалуйста, обратитесь к Разделу 10.2.

7.12 Самопроверка

Пожалуйста, обратитесь к документу «Карта регистров MPU-6000 / MPU-6050» и «Описания регистров» для более подробной информации о самопроверке.

Самопроверка позволяет проверять механические и электрические части датчиков. Самопроверка для каждой оси измерения может быть активирована с помощью регистров самотестирования гироскопа и акселерометра (регистры с 13 по 16).

Когда активируется самопроверка, электроника запускает датчики и выдает выходной сигнал. Выходной сигнал используется для наблюдения за реакцией самопроверки.

Ответ самотестирования определяется следующим образом:

Ответ самотестирования = Выход датчика с включенным самотестированием
- Выход датчика без включенного самотестирования

Реакция самопроверки для каждой оси акселерометра определяется в таблице характеристик акселерометра (раздел 6.2), а реакция для каждой оси гироскопа - в таблице спецификаций гироскопа (раздел 6.1).

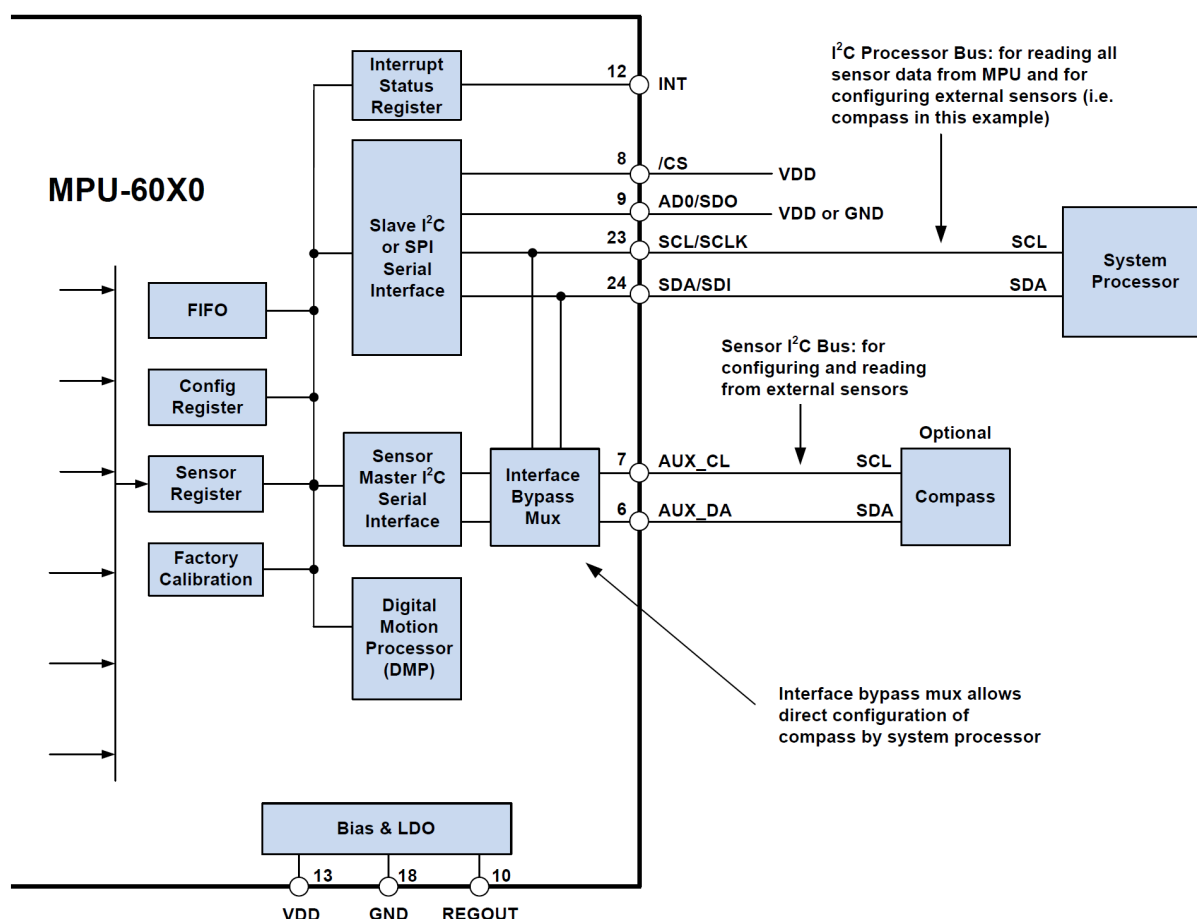
Когда значение ответа самопроверки находится в пределах мин / макс пределов спецификации продукта, деталь прошла самотестирование. Когда отклик самопроверки превышает минимальное / максимальные значения, считается, что деталь не прошла самотестирование. Код для выполнения кода самопроверки включен в программное обеспечение MotionApps, предоставляемое InvenSense.

7.13 Решение MPU-60X0 для 9-осевого объединения датчиков с использованием интерфейса I2C

На рисунке ниже системный процессор является ведущим I2C для MPU-60X0. Кроме того, MPU-60X0 является ведущим устройством I2C для дополнительного внешнего датчика компаса. MPU-60X0 имеет ограниченные возможности в качестве ведущего устройства I2C и зависит от системного процессора для управления начальной конфигурацией любых вспомогательных датчиков. MPU-60X0 имеет мультиплексор обхода интерфейса, который соединяет выходы 23 и 24 шины I2C системного процессора (SDA и SCL) непосредственно с выводами 6 и 7 шины I2C вспомогательного датчика (AUX_DA и AUX_CL).

После того, как вспомогательные датчики сконфигурированы системным процессором, мультиплексор обхода интерфейса должен быть отключен, чтобы вспомогательный мастер I2C MPU-60X0 мог управлять шиной I2C датчика и собирать данные со вспомогательных датчиков.

Для получения дополнительной информации о главном управлении I2C, пожалуйста, обратитесь к Разделу 10.



7.14 MPU-6000 с использованием интерфейса SPI

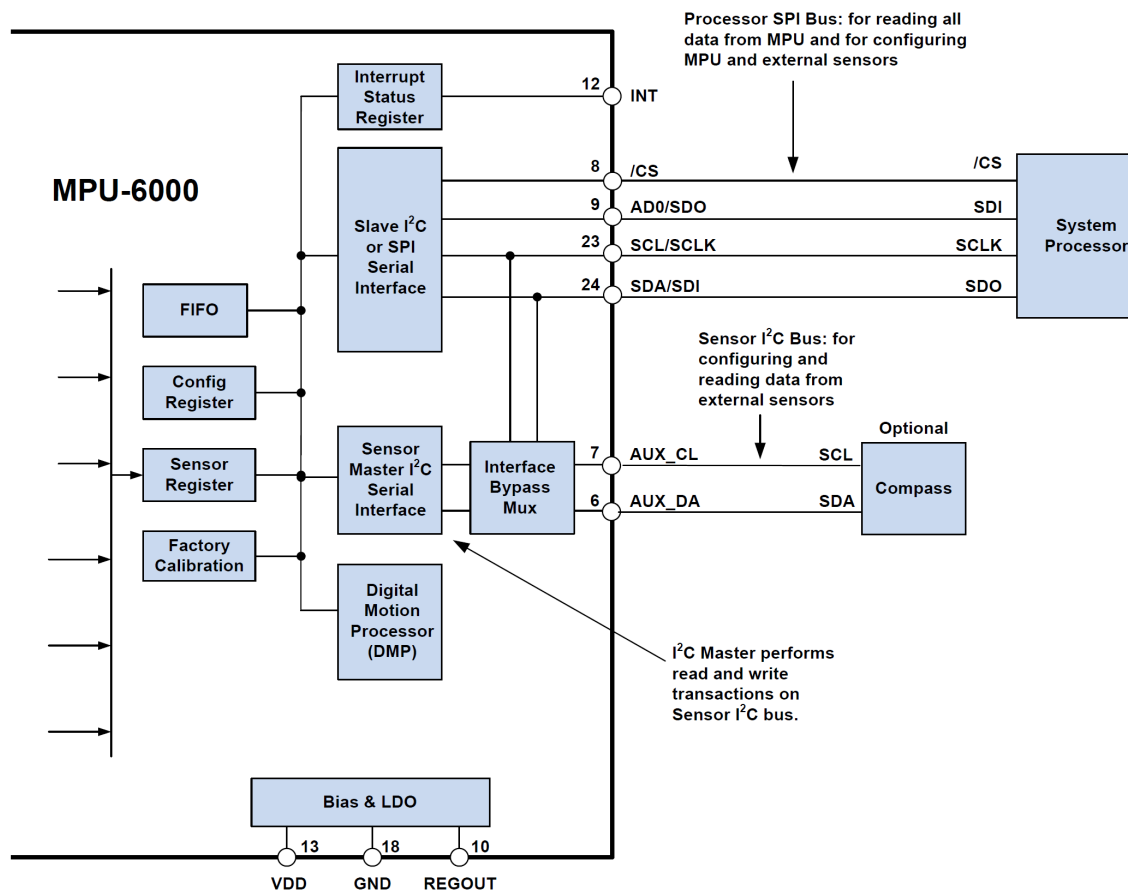
На рисунке ниже системный процессор является ведущим SPI для MPU-6000. Контакты 8, 9, 23 и 24 используются для поддержки сигналов /CS, SDO, SCLK и SDI для связи SPI. Поскольку эти контакты SPI используются совместно с подчиненными контактами I2C (9, 23 и 24), системный процессор не может получить доступ к вспомогательной шине I2C через мультиплексор обхода интерфейса, который соединяет интерфейсные контакты I2C процессора с интерфейсными контактами I2C датчика.

Поскольку MPU-6000 имеет ограниченные возможности в качестве ведущего устройства I2C и зависит от системного процессора для управления начальной конфигурацией любых вспомогательных датчиков, необходимо использовать другой метод для программирования датчиков на выводах 6 и 7 шины I2C вспомогательного датчика (AUX_DA и AUX_CL).

При использовании SPI-связи между MPU-6000 и системным процессором конфигурация устройств на вспомогательной шине датчика I2C может быть достигнута с помощью ведомых устройств I2C 0-4 для выполнения операций чтения и записи на любом устройстве и регистрации на вспомогательной шине I2C. Интерфейс I2C Slave 4 может использоваться для выполнения только однобайтовых операций чтения и записи.

После настройки внешних датчиков MPU-6000 может выполнять однобайтовые или многобайтовые операции чтения с использованием шины I2C датчика. Результаты считывания с контроллеров Slave 0-3 могут быть записаны в буфер FIFO, а также в регистры внешнего датчика.

Для получения дополнительной информации относительно управления вспомогательным интерфейсом I2C MPU-60X0, пожалуйста, обратитесь к документу «Карта регистров и описания регистров» MPU-6000 / MPU-6050



7.15 Генерация внутренних часов

MPU-60X0 имеет гибкую схему синхронизации, позволяющую использовать различные внутренние или внешние источники тактовых импульсов для внутренней синхронной схемы. Эта синхронная схема включает в себя преобразование сигнала и АЦП, DMP и различные схемы управления и регистры. Встроенная ФАПЧ обеспечивает гибкость в допустимых входах для генерации этих тактовых импульсов.

Допустимые внутренние источники для генерации внутренних часов:

- Внутренний генератор релаксации
- Любой из гироскопов X, Y или Z (MEMS-осцилляторы с отклонением $\pm 1\%$ от температуры)

Допустимые внешние источники синхронизации:

- 32,768 кГц прямоугольная волна
- 19,2 МГц прямоугольная волна

Выбор источника для генерации внутренних синхронных часов зависит от доступности внешних источников и требований к потребляемой мощности и точности часов. Эти требования, скорее всего, будут зависеть от режима работы. Например, в одном режиме, где наибольшее беспокойство вызывает энергопотребление, пользователь может захотеть использовать цифровой процессор движения MPU-60X0 для обработки данных акселерометра, не выключая гироскопы. В этом случае внутренний генератор релаксации является хорошим выбором часов. Однако в другом режиме, где гироскопы активны, выбор гироскопов в качестве источника синхронизации обеспечивает более точный источник синхронизации.

Точность синхронизации важна, поскольку ошибки синхронизации напрямую влияют на расчеты расстояния и угла, выполняемые цифровым процессором движения (и, соответственно, любым процессором).

Есть также условия запуска, чтобы рассмотреть. При первом запуске MPU-60X0 устройство использует свои внутренние часы до тех пор, пока не запрограммируется на работу от другого источника. Это позволяет пользователю, например, дожидаться стабилизации генераторов MEMS, прежде чем они будут выбраны в качестве источника синхронизации.

7.16 Регистры данных датчика

Регистры данных датчика содержат последние данные гироскопа, акселерометра, вспомогательного датчика и данные измерения температуры. Они доступны только для чтения и доступны через последовательный интерфейс. Данные из этих регистров могут быть прочитаны в любое время. Тем не менее, функция прерывания может использоваться для определения доступности новых данных.

Таблицу источников прерываний см. В разделе 8.

7.17 FIFO

MPU-60X0 содержит 1024-байтовый регистр FIFO, доступный через последовательный интерфейс. Регистр конфигурации FIFO определяет, какие данные записываются в FIFO. Возможные варианты выбора включают данные гироскопа, данные акселерометра, показания температуры, показания вспомогательного датчика и вход FSYNC. Счетчик FIFO отслеживает, сколько байтов действитель-

ных данных содержится в FIFO. Регистр FIFO поддерживает пакетное чтение. Функция прерывания может использоваться для определения доступности новых данных.

Для получения дополнительной информации о FIFO, пожалуйста, обратитесь к документу «Карта регистров и описания регистров MPU-6000 / MPU-6050».

7.18 Прерывания

Функциональность прерывания настраивается через регистр конфигурации прерываний. Настраиваемые элементы включают в себя конфигурацию контактов INT, метод фиксации и очистки прерываний и триггеры для прерывания. Элементы, которые могут вызвать прерывание являются (1) Часы генератор заблокированы для нового опорного генератора (используется при переключении источника часов); (2) новые данные доступны для чтения (из регистров FIFO и данных); (3) прерывание события акселерометра; и (4) MPU-60X0 не получил подтверждение от вспомогательного датчика на вторичной шине I2C. Состояние прерывания можно прочесть из регистра состояния прерывания.

Для получения дополнительной информации о прерываниях см. Документ «Карта регистров MPU-60X0» и «Описание регистров».

Информацию о прерываниях события акселерометра MPU-60X0 см. В разделе 8.

7.19 Датчик температуры с цифровым выходом

Встроенный датчик температуры и АЦП используются для измерения температуры матрицы MPU-60X0. Показания с АЦП могут быть считаны из FIFO или регистров данных датчика.

7.20 Смещение и LDO

Секция смещения и LDO генерирует внутреннее питание и опорные напряжения и токи, требуемые MPU-60X0. Его вторым входом является нерегулируемым VDD от 2,375 до 3.46V и VLOGIC логика опорного напряжения питания 1.71V к VDD (только MPU-6050). Выход LDO обойден конденсатором на выходе. Для получения дополнительной информации о конденсаторе, пожалуйста, обратитесь к Спецификации для внешних компонентов (Раздел 7.3).

7.21 Зарядный насос

Встроенный зарядный насос генерирует высокое напряжение, необходимое для генераторов MEMS. Его выход обойден конденсатором в SPOUT. Для получения дополнительной информации о конденсаторе, пожалуйста, обратитесь к Спецификации для внешних компонентов (Раздел 7.3).

8 Программируемые прерывания

MPU-60X0 имеет программируемую систему прерываний, которая может генерировать сигнал прерывания на выводе INT. Флаги состояния указывают на источник прерывания. Источники прерывания могут быть включены и отключены индивидуально.

Таблица источников прерываний

Interrupt Name	Module
FIFO Overflow	FIFO
Data Ready	Sensor Registers
I ² C Master errors: Lost Arbitration, NACKs	I ² C Master
I ² C Slave 4	I ² C Master

Для получения информации относительно регистров включения / отключения прерываний и регистров флагов, пожалуйста, обратитесь к документу «Карта регистров и описания регистров» MPU-6000 / MPU-6050. Некоторые источники прерываний описаны ниже.

9 цифровой интерфейс

9.1 Последовательные интерфейсы I2C и SPI (только MPU-6000)

Внутренние регистры и память MPU-6000 / MPU-6050 могут быть доступны с использованием I2C на частоте 400 кГц или SPI на частоте 1 МГц (только для MPU-6000). SPI работает в четырехпроводном режиме.

Serial Interface

Pin Number	MPU-6000	MPU-6050	Pin Name	Pin Description
8	Y		/CS	SPI chip select (0=SPI enable)
8		Y	VLOGIC	Digital I/O supply voltage. VLOGIC must be ≤ VDD at all times.
9	Y		AD0 / SDO	I ² C Slave Address LSB (AD0); SPI serial data output (SDO)
9		Y	AD0	I ² C Slave Address LSB
23	Y		SCL / SCLK	I ² C serial clock (SCL); SPI serial clock (SCLK)
23		Y	SCL	I ² C serial clock
24	Y		SDA / SDI	I ² C serial data (SDA); SPI serial data input (SDI)
24		Y	SDA	I ² C serial data

Примечание:

Чтобы предотвратить переключение в режим I2C при использовании SPI (MPU-6000), интерфейс I2C следует отключить, установив бит конфигурации I2C_IF_DIS. Установка этого бита должна выполняться сразу после ожидания времени, указанного в «Время запуска для чтения / записи в регистре» в разделе 6.3.

Для получения дополнительной информации о бите I2C_IF_DIS, пожалуйста, обратитесь к документу «Карта регистров и описания регистров MPU-6000 / MPU-6050».

9.2 Интерфейс I2C

I2C является двухпроводным интерфейсом, состоящим из сигналов последовательных данных (SDA) и последовательных тактовых импульсов (SCL). В целом,

линии являются открытыми и двунаправленными. В обобщенной реализации интерфейса I2C подключенные устройства могут быть ведущим или ведомым. Ведущее устройство помещает подчиненный адрес на шину, а ведомое устройство с совпадающим адресом подтверждает ведущий.

MPU-60X0 всегда работает как подчиненное устройство при связи с системным процессором, который, таким образом, действует как ведущий. Для линий SDA и SCL обычно нужны подтягивающие резисторы для VDD. Максимальная скорость шины составляет 400 кГц.

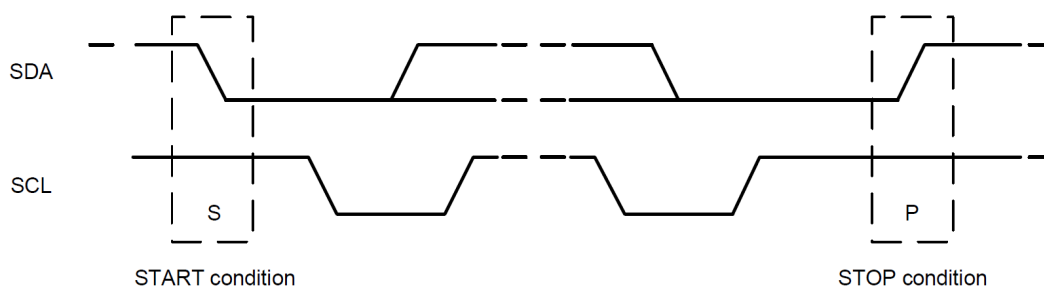
Адрес ведомого устройства MPU-60X0 - b110100X, длина которого 7 бит. Бит LSB 7-битного адреса определяется логическим уровнем на выводе AD0. Это позволяет двум MPU-60X0 подключаться к одной шине I2C. При использовании в этой конфигурации адрес одного из устройств должен быть b1101000 (вывод AD0 - низкий логический уровень), а адрес другого - b1101001 (вывод AD0 - высокий логический уровень).

9.3 Протокол связи I2C

Условия START (S) и STOP (P)

Связь по шине I2C начинается, когда мастер устанавливает на шину условие START (S), которое определяется как переход HIGH-LOW линии SDA, в то время как линия SCL HIGH (см. Рисунок ниже). Шина считается занятой до тех пор, пока мастер не установит на шине условие STOP (P), которое определяется как переход от низкого уровня к высокому на линии SDA, в то время как SCL является высоким (см. Рисунок ниже).

Кроме того, шина остается занятой, если вместо условия STOP генерируется повторный START (Sr).



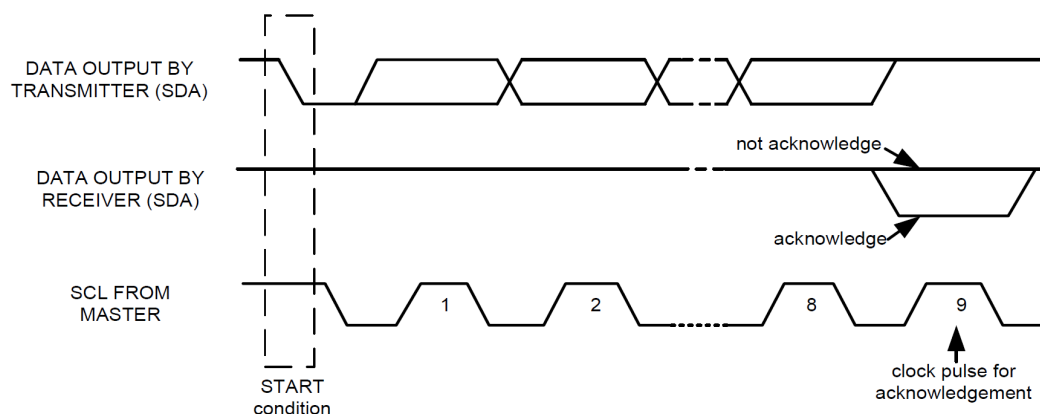
START and STOP Conditions

Формат данных / Подтверждение

Байты данных I2C определены как 8-битные. Нет ограничений на количество байтов, передаваемых за одну передачу данных. За каждым переданным байтом должен следовать сигнал подтверждения (ACK). Синхронизация для сигнала подтверждения генерируется ведущим устройством, в то время как приемник генерирует фактический сигнал подтверждения, опуская SDA и удерживая его на низком уровне в течение ВЫСОКОЙ части тактового импульса подтверждения.

Если ведомое устройство занято и не может передавать или принимать другой байт данных до тех пор, пока не будет выполнена какая-либо другая задача, он может удерживать SCL LOW, тем самым приводя мастер в состояние ожидания.

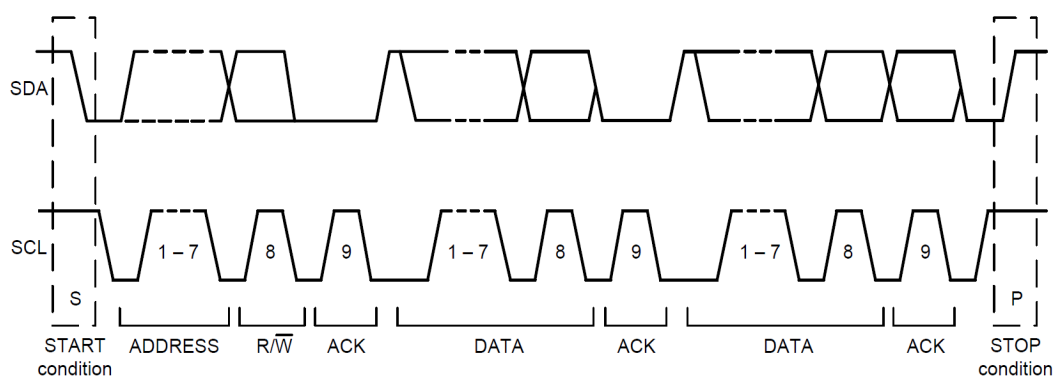
Нормальная передача данных возобновляется, когда ведомое устройство готово, и освобождает линию часов (см. Следующий рисунок).



Acknowledge on the I²C Bus

Связь

После начала связи с условием START (S) ведущий отправляет 7-битный адрес ведомого, за которым следует 8-й бит, бит чтения / записи. Бит чтения / записи указывает, получает ли мастер данные от или записывает на ведомое устройство. Затем мастер освобождает линию SDA и ожидает сигнала подтверждения (ACK) от подчиненного устройства. Каждый переданный байт должен сопровождаться битом подтверждения. Чтобы подтвердить, подчиненное устройство тянет линию SDA НИЗКОЙ и поддерживает ее НИЗКОЙ в течение высокого периода линии SCL. Передача данных всегда прерывается ведущим с условием STOP (P), освобождая тем самым линию связи. Однако ведущее устройство может генерировать повторяющееся условие START (Sr) и обращаться к другому ведомому без предварительной генерации условия STOP (P). Переход НИЗКИЙ к ВЫСОКОМУ на линии SDA, в то время как SCL является ВЫСОКИМ, определяет условие остановки. Все изменения SDA должны иметь место, когда SCL низкий, за исключением условий запуска и остановки.



Complete I²C Data Transfer

Для записи внутренних регистров MPU-60X0 мастер передает условие запуска (S), за которым следуют адрес I2C и бит записи (0). На 9-м тактовом цикле (когда часы высокие) MPU-60X0 подтверждает передачу. Затем мастер помещает адрес регистра (RA) на шину. После того, как MPU-60X0 подтвердит получение адреса регистра, мастер помещает данные регистра на шину. За этим следует сигнал ACK, и передача данных может быть завершена условием остановки (P). Чтобы

записать несколько байтов после последнего сигнала АСК, мастер может продолжить вывод данных вместо передачи сигнала остановки. В этом случае MPU-60X0 автоматически увеличивает адрес регистра и загружает данные в соответствующий регистр. На следующих рисунках показаны одно- и двухбайтовые последовательности записи.

Однобайтовая последовательность записи

Master	S	AD+W		RA		DATA		P
Slave			ACK		ACK		ACK	

Последовательность записи пакета

Master	S	AD+W		RA		DATA		DATA		P
Slave			ACK		ACK		ACK		ACK	

Для чтения внутренних регистров MPU-60X0 мастер отправляет условие запуска, за которым следует адрес I2C и бит записи, а затем адрес регистра, который будет считан. После получения сигнала АСК от MPU-60X0 мастер передает сигнал запуска, за которым следует адрес ведомого и бит чтения. В результате MPU-60X0 отправляет сигнал АСК и данные. Связь заканчивается сигналом не подтверждения (NACK) и стоп-битом от мастера. Условие NACK определяется так, чтобы линия SDA оставалась высокой в 9-м тактовом цикле. На следующих рисунках показаны одно- и двухбайтовые последовательности чтения.

Single-Byte Read Sequence

Master	S	AD+W		RA		S	AD+R			NACK	P
Slave			ACK		ACK			ACK	DATA		

Burst Read Sequence

Master	S	AD+W		RA		S	AD+R			ACK		NACK	P
Slave			ACK		ACK			ACK	DATA		DATA		

9.4 I²C Terms

Signal	Description
S	Start Condition: SDA goes from high to low while SCL is high
AD	Slave I ² C address
W	Write bit (0)
R	Read bit (1)
ACK	Acknowledge: SDA line is low while the SCL line is high at the 9 th clock cycle
NACK	Not-Acknowledge: SDA line stays high at the 9 th clock cycle
RA	MPU-60X0 internal register address
DATA	Transmit or received data
P	Stop condition: SDA going from low to high while SCL is high

9.5 Интерфейс SPI (только MPU-6000)

SPI - это 4-проводной синхронный последовательный интерфейс, который использует две линии управления и две линии данных. MPU-6000 всегда работает как ведомое устройство во время стандартной работы Master-Slave SPI.

Что касается ведущего устройства, последовательный тактовый выход (SCLK), последовательный выход данных (SDO) и последовательный ввод данных (SDI) совместно используются ведомыми устройствами. Каждое ведомое устройство SPI требует своей собственной линии выбора чипа (/CS) от ведущего устройства.

/CS становится низким (активным) в начале передачи и возвращается обратно высоким (неактивным) в конце. Только одна линия /CS активна одновременно, гарантируя, что только один ведомый выбран в любой момент времени. Линии /CS невыбранных подчиненных устройств поддерживаются на высоком уровне, в результате чего их линии SDO остаются в состоянии высокого импеданса (с высоким Z), чтобы они не мешали никаким активным устройствам.

SPI Эксплуатационные Особенности

1. Данные доставляются сначала MSB, а LSB последними
2. Данные фиксируются на переднем крае SCLK
3. Данные должны быть переданы по заднему фронту SCLK
4. Максимальная частота SCLK составляет 1 МГц.
5. Операции чтения и записи SPI выполняются за 16 или более тактов (два или более байтов). Первый байт содержит адрес SPI, а следующие байты содержат данные SPI. Первый бит первого байта содержит бит чтения / записи и указывает операции чтения (1) или записи (0).

Следующие 7 бит содержат адрес регистра. В случае многобайтовых операций чтения / записи данные составляют два или более байтов:

SPI Address format

MSB							LSB
R/W	A6	A5	A4	A3	A2	A1	A0

SPI Data format

MSB							LSB
D7	D6	D5	D4	D3	D2	D1	D0

6. Поддерживает одиночное или пакетное чтение / запись.

10. Особенности последовательного интерфейса (MPU-6050)

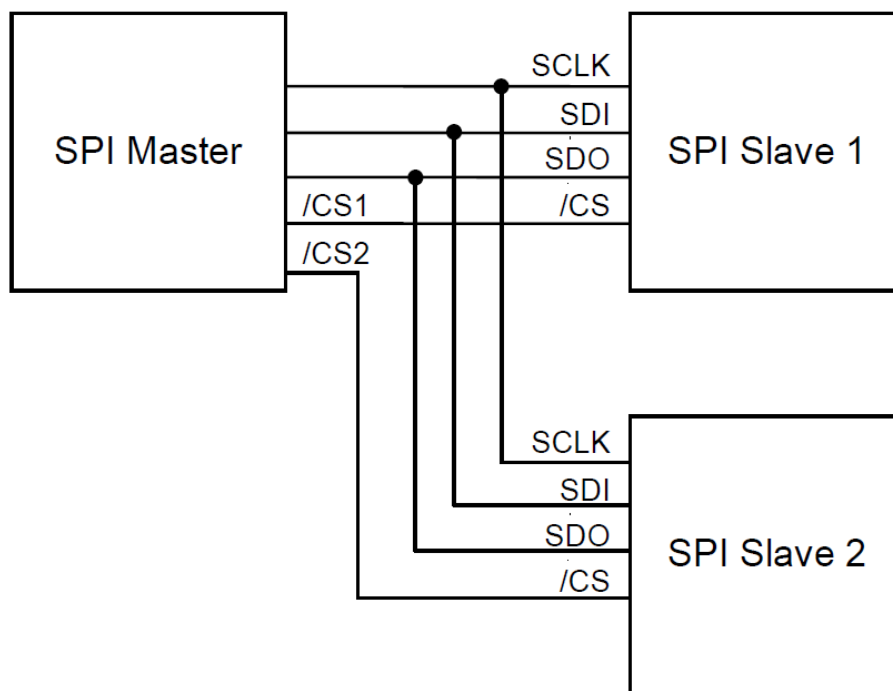
10.1 Поддерживаемые интерфейсы MPU-6050

MPU-6050 поддерживает связь I2C как по первичному (микропроцессорному) последовательному интерфейсу, так и по вспомогательному интерфейсу.

10.2 Уровни логики

Уровни логики ввода-вывода MPU-6050 установлены на VLOGIC, как показано в таблице ниже. AUX_VDDIO должен быть установлен в 0.

AUX_VDDIO	MICROPROCESSOR LOGIC LEVELS (Pins: SDA, SCL, AD0, CLKIN, INT)	AUXILIARY LOGIC LEVELS (Pins: AUX_DA, AUX_CL)
0	VLOGIC	VLOGIC



Typical SPI Master / Slave Configuration

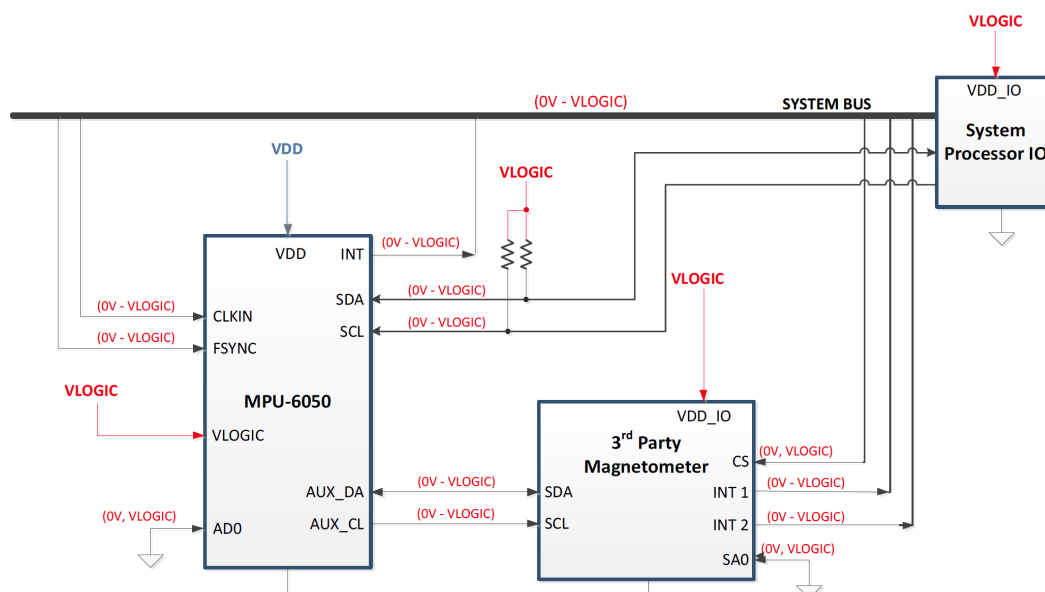
Примечание.

Значение сброса при включении питания для AUX_VDDIO равно 0.

Когда для AUX_VDDIO установлено значение 0 (значение сброса при включении питания), VLOGIC - это напряжение питания как для системной шины микропроцессора, так и для вспомогательной шины I2C, как показано на рисунке в разделе 10.3.

10.3 Диаграмма логических уровней для AUX_VDDIO = 0

На рисунке ниже показан пример схемы со сторонним магнитометром, подключенным к вспомогательной шине I2C. Он показывает логические уровни и подключения напряжения для AUX_VDDIO = 0. Примечание: Фактическая конфигурация будет зависеть от используемых вспомогательных датчиков.



Уровни ввода / вывода и соединения для AUX_VDDIO = 0

Примечания:

1. AUX_VDDIO определяет уровни напряжения ввода-вывода AUX_DA и AUX_CL
(0 = установить уровни выхода относительно VLOGIC)
2. Все другие логические операции ввода-вывода MPU-6050 относятся к VLOGIC.

МПУ-6000 и МПУ-6050

Карта регистров и описание

Редакция 4.2

2 Цель и сфера применения

В этом документе содержится предварительная информация о карте регистров и описание модулей обработки движения TM MPU-6000 TM и MPU-6050 TM, которые в совокупности называются MPU-60X0 TM или MPU TM.

Устройства MPU обеспечивают первое в мире интегрированное 6-осевое решение для процессора движений, которое устраняет перекося оси гироскопа и акселерометра, связанный с дискретными решениями. Устройства объединяют 3-осевой гироскоп и 3-осевой акселерометр на одной и той же кремниевой матрице вместе со встроенным цифровым процессором Motion TM (DMP TM), способным обрабатывать сложные 9-осевые алгоритмы слияния датчиков с использованием проверенных и запатентованных MotionFusion TM движка.

Интегрированные в MPU-6000 и MPU-6050 9-осевые алгоритмы MotionFusion обеспечивают доступ к внешним магнитометрам или другим датчикам через вспомогательную главную шину I2C, позволяя устройствам собирать полный набор данных датчиков без вмешательства системного процессора. Устройства предлагаются с тем же размером и разводкой QFN 4x4x0,9 мм, что и текущее семейство интегрированных 3-осевых гироскопов MPU-3000 TM, обеспечивая простой путь обновления и облегчая размещение на уже ограниченных по объему платах.

Для точного отслеживания как быстрых, так и медленных движений, MPU-60X0 имеет программируемый пользователем гироскоп с полной шкалой диапазона ± 250 , ± 500 , ± 1000 и ± 2000 ° / сек (dps). Детали также имеют программируемый пользователем полный диапазон шкалы акселерометра: ± 2 г, ± 4 г, ± 8 г и ± 16 г.

Семейство MPU-6000 состоит из двух частей: MPU-6000 и MPU-6050. Эти части идентичны друг другу с двумя исключениями. MPU-6050 поддерживает связь I2C на частоте до 400 кГц и имеет вывод VLOGIC, который определяет уровни напряжения его интерфейса; MPU-6000 поддерживает SPI на частоте до 20 МГц в дополнение к I2C и имеет один вывод питания VDD, который является как логическим эталонным источником питания устройства, так и аналоговым источником питания детали.

Для получения более подробной информации об устройствах MPU-60X0, пожалуйста, обратитесь к «Спецификациям продуктов MPU-6000 и MPU-6050».

3 Карта регистров
Карта регистров для MPU-60X0 приведена ниже.

Addr (Hex)	Addr (Dec.)	Register Name	Serial I/F	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0		
0D	13	SELF_TEST_X	R/W		XA_TEST[4-2]				XG_TEST[4-0]				
0E	14	SELF_TEST_Y	R/W		YA_TEST[4-2]				YG_TEST[4-0]				
0F	15	SELF_TEST_Z	R/W		ZA_TEST[4-2]				ZG_TEST[4-0]				
10	16	SELF_TEST_A	R/W	RESERVED		XA_TEST[1-0]		YA_TEST[1-0]		ZA_TEST[1-0]			
19	25	SMPLRT_DIV	R/W	SMPLRT_DIV[7:0]									
1A	26	CONFIG	R/W	-	-	EXT_SYNC_SET[2:0]			DLPF_CFG[2:0]				
1B	27	GYRO_CONFIG	R/W	-	-	-	FS_SEL [1:0]			-	-		
1C	28	ACCEL_CONFIG	R/W	XA_ST	YA_ST	ZA_ST	AFS_SEL[1:0]						
23	35	FIFO_EN	R/W	TEMP_FIFO_EN	XG_FIFO_EN	YG_FIFO_EN	ZG_FIFO_EN	ACCEL_FIFO_EN	SLV2_FIFO_EN	SLV1_FIFO_EN	SLV0_FIFO_EN		
24	36	I2C_MST_CTRL	R/W	MULT_MST_EN	WAIT_FOR_ES	SLV_3_FIFO_EN	I2C_MST_P_NSR	I2C_MST_CLK[3:0]					
25	37	I2C_SLV0_ADDR	R/W	I2C_SLV0_RW	I2C_SLV0_ADDR[6:0]								
26	38	I2C_SLV0_REG	R/W	I2C_SLV0_REG[7:0]									
27	39	I2C_SLV0_CTRL	R/W	I2C_SLV0_EN	I2C_SLV0_BYTE_SW	I2C_SLV0_REG_DIS	I2C_SLV0_GRP	I2C_SLV0_LEN[3:0]					
28	40	I2C_SLV1_ADDR	R/W	I2C_SLV1_RW	I2C_SLV1_ADDR[6:0]								
29	41	I2C_SLV1_REG	R/W	I2C_SLV1_REG[7:0]									
2A	42	I2C_SLV1_CTRL	R/W	I2C_SLV1_EN	I2C_SLV1_BYTE_SW	I2C_SLV1_REG_DIS	I2C_SLV1_GRP	I2C_SLV1_LEN[3:0]					
2B	43	I2C_SLV2_ADDR	R/W	I2C_SLV2_RW	I2C_SLV2_ADDR[6:0]								
2C	44	I2C_SLV2_REG	R/W	I2C_SLV2_REG[7:0]									
2D	45	I2C_SLV2_CTRL	R/W	I2C_SLV2_EN	I2C_SLV2_BYTE_SW	I2C_SLV2_REG_DIS	I2C_SLV2_GRP	I2C_SLV2_LEN[3:0]					

Addr (Hex)	Addr (Dec.)	Register Name	Serial I/F	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
42	66	TEMP_OUT_L	R				TEMP_OUT[7:0]				
43	67	GYRO_XOUT_H	R				GYRO_XOUT[15:8]				
44	68	GYRO_XOUT_L	R				GYRO_XOUT[7:0]				
45	69	GYRO_YOUT_H	R				GYRO_YOUT[15:8]				
46	70	GYRO_YOUT_L	R				GYRO_YOUT[7:0]				
47	71	GYRO_ZOUT_H	R				GYRO_ZOUT[15:8]				
48	72	GYRO_ZOUT_L	R				GYRO_ZOUT[7:0]				
49	73	EXT_SENS_DATA_00	R				EXT_SENS_DATA_00[7:0]				
4A	74	EXT_SENS_DATA_01	R				EXT_SENS_DATA_01[7:0]				
4B	75	EXT_SENS_DATA_02	R				EXT_SENS_DATA_02[7:0]				
4C	76	EXT_SENS_DATA_03	R				EXT_SENS_DATA_03[7:0]				
4D	77	EXT_SENS_DATA_04	R				EXT_SENS_DATA_04[7:0]				
4E	78	EXT_SENS_DATA_05	R				EXT_SENS_DATA_05[7:0]				
4F	79	EXT_SENS_DATA_06	R				EXT_SENS_DATA_06[7:0]				
50	80	EXT_SENS_DATA_07	R				EXT_SENS_DATA_07[7:0]				
51	81	EXT_SENS_DATA_08	R				EXT_SENS_DATA_08[7:0]				
52	82	EXT_SENS_DATA_09	R				EXT_SENS_DATA_09[7:0]				
53	83	EXT_SENS_DATA_10	R				EXT_SENS_DATA_10[7:0]				
54	84	EXT_SENS_DATA_11	R				EXT_SENS_DATA_11[7:0]				
55	85	EXT_SENS_DATA_12	R				EXT_SENS_DATA_12[7:0]				
56	86	EXT_SENS_DATA_13	R				EXT_SENS_DATA_13[7:0]				
57	87	EXT_SENS_DATA_14	R				EXT_SENS_DATA_14[7:0]				
58	88	EXT_SENS_DATA_15	R				EXT_SENS_DATA_15[7:0]				
59	89	EXT_SENS_DATA_16	R				EXT_SENS_DATA_16[7:0]				

Примечание.

Имена регистров, заканчивающиеся на _H и _L, содержат старший и младший байты, соответственно, значения внутреннего регистра.

В приведенных ниже подробных таблицах регистров имена регистров прописными буквами, а значения регистров прописными буквами и курсивом. Например, регистр ACCEL_XOUT_H (регистр 59) содержит 8 старших значащих битов, ACCEL_XOUT [15: 8], 16-разрядного измерения акселерометра по оси X, ACCEL_XOUT.

Значение сброса составляет 0x00 для всех регистров, кроме регистров ниже.

Регистр 107: 0x40.

Регистр 117: 0x68.

4 Описание регистров

В этом разделе описывается функция и содержимое каждого регистра в MPU-60X0.

Примечание.

При включении устройство перейдет в спящий режим.

4.1 Регистры с 13 по 16 - Регистры самопроверки

SELF_TEST_X, SELF_TEST_Y, SELF_TEST_Z и SELF_TEST_A

Тип: чтение / запись

Register (Hex)	Register (Decimal)	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
0D	13	XA_TEST[4-2]			XG_TEST[4-0]				
0E	14	YA_TEST[4-2]			YG_TEST[4-0]				
0F	15	ZA_TEST[4-2]			ZG_TEST[4-0]				
10	16	RESERVED		XA_TEST[1-0]		YA_TEST[1-0]		ZA_TEST[1-0]	

Описание:

Эти регистры используются для самотестирования гироскопа и акселерометра, которые позволяют пользователю проверить механические и электрические части гироскопа и акселерометра. В следующих разделах описывается процесс самопроверки.

1. Аппаратное тестирование гироскопа: относительный метод

Самодиагностика гироскопа позволяет пользователям тестировать механические и электрические части гироскопа. Код для самопроверки включен в программное обеспечение MotionApps™, предоставляемое InvenSense. Пожалуйста, обратитесь к следующему разделу (Получение значения заводской настройки гироскопа (FT)), если не используете программное обеспечение MotionApps.

Когда активируется самопроверка, бортовая электроника активирует соответствующий датчик. Это срабатывание будет перемещать контрольные массы датчика на расстояние, эквивалентное предварительно определенной силе Кориолиса. Это контрольное смещение массы приводит к изменению выходного сигнала датчика, что отражается в выходном сигнале. Выходной сигнал используется для наблюдения за реакцией самопроверки.

Ответ самотестирования (STR) определяется следующим образом:

$$\text{SelfTest Response} = \text{Gyroscope Output with Self-Test Enabled} - \text{Gyroscope Output with Self-Test Disabled}$$

Этот ответ самодиагностики используется для определения того, прошла ли деталь самопроверку или нет, путем обнаружения изменения заводской настройки ответа самопроверки следующим образом:

$$\text{Change from Factory Trim of the Self-Test Response(\%)} = \frac{(STR - FT)}{FT}$$

где

FT = Factory trim value of selftest response, available via MotionApps software

Это изменение от заводской настройки отклика самопроверки должно быть в пределах, указанных в документе со спецификацией изделия MPU-6000 / MPU-6050, чтобы деталь прошла самотестирование. В противном случае считается, что деталь не прошла самопроверку.

Получение значения заводской настройки гироскопа (FT)

Если программное обеспечение InvenSense MotionApps не используется, необходимо выполнить процедуру, описанную ниже, чтобы получить заводскую настройку отклика самопроверки (FT), упомянутого выше. Для конкретных регистров, упомянутых ниже, пожалуйста, обратитесь к регистрам 13-15.

Заводская подстройка отклика самопроверки (FT) рассчитывается, как показано ниже. FT [Xg], FT [Yg] и FT [Zg] относятся к заводским значениям (FT) для осей гироскопа X, Y и Z соответственно. XG_TEST - это десятичная версия XG_TEST [4-0], YG_TEST - это десятичная версия YG_TEST [4-0], а ZG_TEST - десятичная версия ZG_TEST [4-0].

При выполнении самодиагностики для гироскопа, диапазон полной шкалы должен быть установлен на ± 250 dps.

$$\begin{aligned} \begin{cases} FT [Xg] &= 25 * 131 * 1.046^{(XG_TEST-1)} \\ FT [Xg] &= 0 \end{cases} & \begin{aligned} &\text{if } XG_TEST \neq 0 \\ &\text{if } XG_TEST = 0 \end{aligned} \\ \begin{cases} FT [Yg] &= -25 * 131 * 1.046^{(YG_TEST-1)} \\ FT [Yg] &= 0 \end{cases} & \begin{aligned} &\text{if } YG_TEST \neq 0 \\ &\text{if } YG_TEST = 0 \end{aligned} \\ \begin{cases} FT [Zg] &= 25 * 131 * 1.046^{(ZG_TEST-1)} \\ FT [Zg] &= 0 \end{cases} & \begin{aligned} &\text{if } ZG_TEST \neq 0 \\ &\text{if } ZG_TEST = 0 \end{aligned} \end{aligned}$$

2. Самотестирование аппаратного акселерометра: относительный метод

Самотестирование акселерометра позволяет пользователям тестировать механические и электрические части акселерометра. Код для самопроверки включен в программное обеспечение MotionApps, предоставляемое InvenSense. Обратитесь к следующему разделу (озаглавленному Получение значения заводской настройки акселерометра (FT)), если не используете программное обеспечение MotionApps.

Когда активируется самопроверка, бортовая электроника активирует соответствующий датчик. Это приведение в действие имитирует внешнюю силу. Сработавший датчик, в свою очередь, выдаст соответствующий выходной сигнал. Выходной сигнал используется для наблюдения за реакцией самопроверки.

Ответ самотестирования (STR) определяется следующим образом:

SelfTest Response

= *Accelerometer Output with Self-Test Enabled*

– *Accelerometer Output with Self-Test Disabled*

Этот ответ самодиагностики используется для определения того, прошла ли деталь самопроверку или нет, путем обнаружения изменения заводской настройки ответа самопроверки следующим образом:

$$\text{Change from Factory Trim of the Self-Test Response}(\%) = \frac{(STR - FT)}{FT}$$

where,

FT = Factory trim value of selftest response, available via MotionApps software

Это изменение от заводской настройки отклика самопроверки должно быть в пределах, указанных в документе со спецификацией изделия MPU-6000 / MPU-6050, чтобы деталь прошла самотестирование. В противном случае считается, что деталь не прошла самопроверку.

Получение значения заводской настройки акселерометра (FT)

Если программное обеспечение InvenSense MotionApps не используется, необходимо выполнить процедуру, описанную ниже, чтобы получить заводскую настройку отклика самопроверки (FT), упомянутого выше. Для конкретных регистров, упомянутых ниже, пожалуйста, обратитесь к регистрам 13-16.

Заводская подстройка отклика самопроверки (FT) рассчитывается, как показано ниже. FT [Xa], FT [Ya] и FT [Za] относятся к заводским значениям (FT) для осей акселерометра X, Y и Z соответственно. В приведенных ниже уравнениях заводские значения обрезки для акселерата должны быть в десятичном формате, и они определяются путем объединения битов самотестирования верхнего акселерометра (биты 4-2) с битами самопроверки нижнего акселерометра (биты 1-0).

При выполнении самотестирования акселерометра диапазон полной шкалы должен быть установлен на $\pm 8g$.

$$\begin{cases} FT[Xa] = 4096 * 0.34 * \frac{0.92^{(\frac{XA_TEST-1}{2^5-2})}}{0.34} & \text{if } XA_TEST \neq 0. \\ FT[Xa] = 0 & \text{if } XA_TEST = 0. \end{cases}$$

$$\begin{cases} FT[Ya] = 4096 * 0.34 * \frac{0.92^{(\frac{YA_TEST-1}{2^5-2})}}{0.34} & \text{if } YA_TEST \neq 0. \\ FT[Ya] = 0 & \text{if } YA_TEST = 0. \end{cases}$$

$$\begin{cases} FT[Za] = 4096 * 0.34 * \frac{0.92^{(\frac{ZA_TEST-1}{2^5-2})}}{0.34} & \text{if } ZA_TEST \neq 0. \\ FT[Za] = 0 & \text{if } ZA_TEST = 0. \end{cases}$$

Параметры:

XA_TEST 5-битное значение без знака. FT [Xa] определяется с использованием этого значения, как описано выше.

XG_TEST 5-битное значение без знака. FT [Xg] определяется с использованием этого значения, как описано выше.

YA_TEST 5-битное значение без знака. FT [Ya] определяется с использованием этого значения, как описано выше.

YG_TEST 5-битное значение без знака. FT [Yg] определяется с использованием этого значения, как описано выше.

ZA_TEST 5-битное значение без знака. FT [Za] определяется с использованием этого значения, как описано выше.

ZG_TEST 5-битное значение без знака. FT [Zg] определяется с использованием этого значения, как описано выше.

4.2 Регистр 25 - делитель частоты дискретизации**SMPRT_DIV**

Тип: чтение / запись

Register (Hex)	Register (Decimal)	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
19	25	SMPLRT_DIV[7:0]							

Описание:

Этот регистр определяет делитель от выходной скорости гироскопа, используемой для генерации частоты дискретизации для MPU-60X0.

Выход регистра датчика, выход FIFO и выборка DMP основаны на частоте дискретизации.

Частота дискретизации генерируется путем деления скорости вывода гироскопа на SMPLRT_DIV:

Частота дискретизации = Выходная скорость гироскопа / (1 + SMPLRT_DIV), где выходная частота гироскопа = 8 кГц, когда DLPF отключен (DLPF_CFG = 0 или 7), и 1 кГц, когда DLPF включен (см. Регистр 26).

Примечание. Выходная частота акселерометра составляет 1 кГц. Это означает, что для частоты дискретизации, превышающей 1 кГц, одна и та же выборка акселерометра может выводиться в регистры FIFO, DMP и датчика более одного раза.

Схему прохождения сигналов гироскопа и акселерометра см. В разделе 8 документа со спецификацией изделия MPU-6000 / MPU-6050.

Параметры:

SMPLRT_DIV 8-битное значение без знака. Частота дискретизации определяется путем деления выходной скорости гироскопа на это значение.

4.3 Регистр 26 - Конфигурация**CONFIG**

Тип: чтение / запись

Register (Hex)	Register (Decimal)	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
1A	26	-	-	EXT_SYNC_SET[2:0]			DLPF_CFG[2:0]		

Описание:

Этот регистр конфигурирует выборку выводов внешней синхронизации кадров (FSYNC) и настройку цифрового фильтра низких частот (DLPF) для гироскопов и акселерометров.

Внешний сигнал, подключенный к выводу FSYNC, может быть отображен путем настройки EXT_SYNC_SET.

Изменения сигнала на выводе FSYNC фиксируются, так что короткие импульсы могут быть захвачены. Зафиксированный сигнал FSYNC будет дискретизироваться с частотой дискретизации, как определено в регистре 25. После выборки защелка вернется в текущее состояние сигнала FSYNC.

Выбранное значение будет сообщаться вместо младшего значащего бита в регистре данных датчика, определяемом значением EXT_SYNC_SET, в соответствии со следующей таблицей.

EXT_SYNC_SET	FSYNC Bit Location
0	Input disabled
1	TEMP_OUT_L[0]
2	GYRO_XOUT_L[0]
3	GYRO_YOUT_L[0]
4	GYRO_ZOUT_L[0]
5	ACCEL_XOUT_L[0]
6	ACCEL_YOUT_L[0]
7	ACCEL_ZOUT_L[0]

DLPF настраивается с помощью DLPF_CFG. Акселерометр и гироскоп фильтруются в соответствии со значением DLPF_CFG, как показано в таблице ниже.

DLPF_CFG	Accelerometer (F _s = 1kHz)		Gyroscope		
	Bandwidth (Hz)	Delay (ms)	Bandwidth (Hz)	Delay (ms)	Fs (kHz)
0	260	0	256	0.98	8
1	184	2.0	188	1.9	1
2	94	3.0	98	2.8	1
3	44	4.9	42	4.8	1
4	21	8.5	20	8.3	1
5	10	13.8	10	13.4	1
6	5	19.0	5	18.6	1
7	RESERVED		RESERVED		8

Бит 7 и бит 6 зарезервированы.

Параметры:

EXT_SYNC_SET 3-битное значение без знака. Настраивает выборку контактов FSYNC.

DLPF_CFG 3-битное значение без знака. Конфигурирует настройку DLPF.

4.4 Регистр 27 - Конфигурация гироскопа

GYRO_CONFIG

Тип: чтение / запись

Register (Hex)	Register (Decimal)	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
1B	27	XG_ST	YG_ST	ZG_ST	FS_SEL[1:0]		-	-	-

Описание:

Этот регистр используется для запуска самодиагностики гироскопа и настройки полной шкалы гироскопа.

Самодиагностика гироскопа позволяет пользователям тестировать механические и электрические части гироскопа. Самопроверка для каждой оси гироскопа может быть активирована путем управления битами XG_ST, YG_ST и ZG_ST этого регистра. Самотестирование для каждой оси может выполняться независимо или одновременно.

Когда активируется самопроверка, бортовая электроника активирует соответствующий датчик. Это срабатывание будет перемещать контрольные массы датчика на расстояние, эквивалентное предварительно определенной силе Кориолиса. Это контрольное смещение массы приводит к изменению выходного сигнала датчика, что отражается в выходном сигнале. Выходной сигнал используется для наблюдения за реакцией самопроверки.

Ответ самотестирования определяется следующим образом:

Ответ самотестирования = Выход датчика с включенным самотестированием - Выход датчика без включенного самотестирования

Пределы самопроверки для каждой оси гироскопа указаны в таблицах электрических характеристик документа со спецификацией изделия MPU-6000 / MPU-6050. Когда значение ответа самопроверки находится в пределах мин / макс пределов спецификации продукта, деталь прошла самотестирование. Когда ответ самопроверки превышает минимальные / максимальные значения, указанные в документе, деталь считается не прошедшей самопроверку.

FS_SEL выбирает диапазон полной шкалы выходов гироскопа в соответствии со следующей таблицей.

FS_SEL	Full Scale Range
0	± 250 °/s
1	± 500 °/s
2	± 1000 °/s
3	± 2000 °/s

Биты со 2 по 0 зарезервированы.

Параметры:

XG_ST Установка этого бита заставляет гироскоп оси X выполнять самопроверку.

YG_ST Установка этого бита заставляет гироскоп оси Y выполнить самопроверку.

ZG_ST Установка этого бита заставляет гироскоп оси Z выполнить самопроверку.

FS_SEL 2-битное значение без знака. Выбирает полную гамму гироскопов.

4.5 Регистр 28 - Конфигурация акселерометра ACCEL_CONFIG

Тип: чтение / запись

Register (Hex)	Register (Decimal)	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
1C	28	XA_ST	YA_ST	ZA_ST	AFS_SEL[1:0]		-		

Описание:

Этот регистр используется для запуска самопроверки акселерометра и настройки полного диапазона акселерометра. Этот регистр также настраивает цифровой фильтр верхних частот (DHPF).

Самотестирование акселерометра позволяет пользователям тестировать механические и электрические части акселерометра. Самопроверка для каждой оси акселерометра может быть активирована путем управления битами XA_ST, YA_ST и ZA_ST этого регистра. Самотестирование для каждой оси может выполняться независимо или одновременно.

Когда активируется самопроверка, бортовая электроника активирует соответствующий датчик. Это приведение в действие имитирует внешнюю силу. Сработавший датчик, в свою очередь, выдаст соответствующий выходной сигнал. Выходной сигнал используется для наблюдения за реакцией самопроверки.

Ответ самотестирования определяется следующим образом:

Ответ самотестирования = Выход датчика с включенным самотестированием - Выход датчика без включенного самотестирования

Пределы самопроверки для каждой оси акселерометра указаны в таблицах электрических характеристик документа MPU-6000 / MPU-6050. Когда значение ответа самопроверки находится в пределах мин / макс пределов спецификации продукта, деталь прошла самотестирование. Когда ответ самопроверки превышает минимальные / максимальные значения, указанные в документе, деталь считается не прошедшей самопроверку.

AFS_SEL выбирает полный диапазон шкалы выходов акселерометра в соответствии со следующей таблицей.

AFS_SEL	Full Scale Range
0	$\pm 2g$
1	$\pm 4g$
2	$\pm 8g$
3	$\pm 16g$

Параметры:

XA_ST При значении 1 акселерометр оси X выполняет самопроверку.

YA_ST При значении 1 акселерометр по оси Y выполняет самотестирование.

ZA_ST При значении 1 акселерометр по оси Z выполняет самопроверку.

AFS_SEL 2-битное значение без знака. Выбирает полный диапазон акселерометров.

4.6 Регистр 35 - FIFO Enable FIFO_EN

Тип: чтение / запись

Register (Hex)	Register (Decimal)	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
23	35	TEMP_FIFO_EN	XG_FIFO_EN	YG_FIFO_EN	ZG_FIFO_EN	ACCEL_FIFO_EN	SLV2_FIFO_EN	SLV1_FIFO_EN	SLV0_FIFO_EN

Описание:

Этот регистр определяет, какие измерения датчика загружаются в буфер FIFO.

Данные, хранящиеся в регистрах данных датчика (регистры 59–96), будут загружены в буфер FIFO, если соответствующий бит датчика FIFO_EN установлен в 1 в этом регистре.

Когда в этом регистре активирован бит FIFO_EN датчика, данные из регистров данных датчика будут загружены в буфер FIFO. Выборка датчиков производится с частотой дискретизации, определенной в Регистре 25. Для получения дополнительной информации о регистрах данных датчиков см. Регистры 59–96.

Когда соответствующий бит FIFO_EN внешнего подчиненного устройства (SLVx_FIFO_EN, где x = 0, 1 или 2) установлен в 1, данные, хранящиеся в его соответствующих регистрах данных (регистры EXT_SENS_DATA, регистры с 73 по 96), будут записаны в буфер FIFO с частотой дискретизации. Ассоциация регистров EXT_SENS_DATA с подчиненными I2C определяется регистрами I2C_SLVx_CTRL (где x = 0, 1 или 2; регистры 39, 42 и 45). Для получения информации о регистрах EXT_SENS_DATA, пожалуйста, обратитесь к регистрам 73-96.

Обратите внимание, что соответствующий бит FIFO_EN (SLV3_FIFO_EN) находится в I2C_MST_CTRL (регистр 36). Также обратите внимание, что Slave 4 ведет себя иначе, чем Slave 0-3. Пожалуйста, обратитесь к регистрам 49-53 для получения дополнительной информации об использовании Slave 4.

Параметры:

TEMP_FIFO_EN При значении 1 этот бит позволяет записывать TEMP_OUT_H и TEMP_OUT_L (регистры 65 и 66) в буфер FIFO.

XG_FIFO_EN При значении 1 этот бит разрешает запись GYRO_XOUT_H и GYRO_XOUT_L (регистры 67 и 68) в буфер FIFO.

YG_FIFO_EN При значении 1 этот бит разрешает запись GYRO_YOUT_H и GYRO_YOUT_L (регистры 69 и 70) в буфер FIFO.

ZG_FIFO_EN Если установлено в 1, этот бит разрешает запись GYRO_ZOUT_H и GYRO_ZOUT_L (регистры 71 и 72) в буфер FIFO.

ACCEL_FIFO_EN При значении 1 этот бит позволяет ACCEL_XOUT_H, ACCEL_XOUT_L, ACCEL_YOUT_H, ACCEL_YOUT_L, ACCEL_ZOUT_H и ACCEL_ZOUT_L (регистры с 59 по 64) записываться в буфер FIFO.

FIFO_EN При значении 1 этот бит разрешает запись регистров EXT_SENS_DATA (регистров с 73 по 96), связанных с подчиненным устройством 2, в буфер FIFO.

SLV1_FIFO_EN При значении 1 этот бит разрешает запись регистров EXT_SENS_DATA (регистров с 73 по 96), связанных с подчиненным устройством 1, в буфер FIFO.

SLV0_FIFO_EN При значении 1 этот бит разрешает запись регистров EXT_SENS_DATA (регистров с 73 по 96), связанных с ведомым 0, в буфер FIFO.

Примечание.

Для получения дополнительной информации относительно связи регистров EXT_SENS_DATA с определенными подчиненными устройствами, пожалуйста, обратитесь к регистрам 73-96.

4.7 Регистр 36 - I2C Master Control I2C_MST_CTRL

Тип: чтение / запись

Register (Hex)	Register (Decimal)	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
24	36	MULT_MST_EN	WAIT_FOR_ES	SLV_3_FIFO_EN	I2C_MST_P_NSR	I2C_MST_CLK[3:0]			

Описание:

Этот регистр конфигурирует вспомогательную шину I2C для управления одним мастером или несколькими мастерами. Кроме того, регистр используется для задержки прерывания готовности данных, а также позволяет записывать данные ведомого 3 в буфер FIFO. Регистр также конфигурирует переход вспомогательного I2C Master от одного ведомого считывающего устройства к другому, а также внутренние тактовые частоты MPU-60X0 8 МГц.

Возможность нескольких мастеров позволяет нескольким мастерам I2C работать на одной шине. В цепях, где требуется возможность многоканального управления, установите MULT_MST_EN на 1. Это увеличит потребляемый ток примерно на 30 мкА.

В цепях, где требуется возможность использования нескольких главных устройств, состояние шины I2C всегда должно контролироваться каждым отдельным ведущим устройством I2C. Прежде чем I2C Master может принять решение об арбитраже шины, он должен сначала подтвердить, что ни один другой I2C Master не имеет арбитража шины. Когда для MULT_MST_EN установлено значение 1, включается логика обнаружения арбитража шины MPU-60X0, что позволяет ему определять, когда шина доступна.

Когда бит WAIT_FOR_ES установлен в 1, прерывание готовности данных будет задерживаться до тех пор, пока данные внешнего датчика с ведомых устройств не будут загружены в регистры EXT_SENS_DATA. Это используется для гарантии того, что как внутренние данные датчика (то есть с гироскопа и акселератора), так и данные внешнего датчика были загружены в их соответствующие регистры данных (то есть данные синхронизированы), когда инициируется прерывание готовности данных.

Когда бит разрешения ведомого 3 FIFO (SLV_3_FIFO_EN) установлен в 1, данные измерений датчика ведомого 3 будут загружаться в буфер FIFO каждый раз. Ассоциация регистра EXT_SENS_DATA с ведомыми устройствами I2C определяется I2C_SLV3_CTRL (регистр 48).

Для получения дополнительной информации о регистрах EXT_SENS_DATA, пожалуйста, обратитесь к регистрам 73-96.

Соответствующие биты FIFO_EN для ведомого 0, ведомого 1 и ведомого 2 можно найти в регистре 35.

Бит I2C_MST_P_NSR конфигурирует переход мастера I2C от одного ведомого чтения к следующему ведомому чтению. Если бит равен 0, между чтени-

ями будет перезапуск. Если бит равен 1, будет остановка с последующим началом следующего чтения. Когда транзакция записи следует за транзакцией чтения, всегда будет использоваться остановка, сопровождаемая началом последовательной записи.

I2C_MST_CLK - это 4-битное значение без знака, которое настраивает делитель на внутренних тактовых частотах 8 МГц MPU-60X0. Он устанавливает тактовую частоту I2C в соответствии со следующей таблицей:

I2C_MST_CLK	I²C Master Clock Speed	8MHz Clock Divider
0	348 kHz	23
1	333 kHz	24
2	320 kHz	25
3	308 kHz	26
4	296 kHz	27
5	286 kHz	28
6	276 kHz	29
7	267 kHz	30
8	258 kHz	31
9	500 kHz	16
10	471 kHz	17
11	444 kHz	18
12	421 kHz	19
13	400 kHz	20
14	381 kHz	21
15	364 kHz	22

Параметры:

MUL_MST_EN Если установлено в 1, этот бит включает возможность работы с несколькими мастерами.

WAIT_FOR_ES При значении 1 этот бит задерживает прерывание готовности данных до тех пор, пока данные внешнего датчика с ведомых устройств не будут загружены в регистры EXT_SENS_DATA.

SLV3_FIFO_EN При значении 1 этот бит разрешает запись регистров EXT_SENS_DATA, связанных с подчиненным устройством 3, в FIFO. Соответствующие биты для ведомых 0-2 можно найти в регистре 35.

I2C_MST_P_NSR Управляет переходом мастера I2C от одного ведомого чтения к следующему ведомому чтению.

Когда этот бит равен 0, между чтениями происходит перезапуск.

Когда этот бит равен 1, происходит остановка и начало маркировки начала следующего чтения.

Когда запись следует за чтением, остановка и запуск всегда выполняются.

I2C_MST_CLK 4-битное значение без знака. Настраивает главный делитель тактовой частоты I2C.

Примечание.

Для получения дополнительной информации относительно связи регистров EXT_SENS_DATA с определенными подчиненными устройствами, пожалуйста, обратитесь к регистрам 73-96.

4.8 Регистры с 37 по 39 - I2C Slave 0 Control I2C_SLV0_ADDR, I2C_SLV0_REG и I2C_SLV0_CTRL

Тип: чтение / запись

Register (Hex)	Register (Decimal)	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
25	37	I2C_SLV0_RW	I2C_SLV0_ADDR[6:0]						
26	38	I2C_SLV0_REG[7:0]							
27	39	I2C_SLV0_EN	I2C_SLV0_BYTE_SW	I2C_SLV0_REG_DIS	I2C_SLV0_GRP	I2C_SLV0_LEN[3:0]			

Описание:

Эти регистры конфигурируют последовательность передачи данных для ведомого 0. Ведомые 1, 2 и 3 также ведут себя аналогично ведомому 0. Однако характеристики ведомого 4 значительно отличаются от характеристик ведомых 0-3. Для получения дополнительной информации о Slave 4, пожалуйста, обратитесь к регистрам 49-53.

Транзакции ведомых данных I2C между MPU-60X0 и ведомым 0 устанавливаются битом I2C_SLV0_RW как операции чтения или записи. Когда этот бит равен 1, передача является операцией чтения. Когда бит равен 0, передача является операцией записи.

I2C_SLV0_ADDR используется для указания адреса ведомого I2C подчиненного устройства 0.

Передача данных начинается с внутреннего регистра в ведомом 0. Этот адрес этого регистра определяется I2C_SLV0_REG.

Количество переданных байтов определяется I2C_SLV0_LEN. Когда передается более 1 байта ($I2C_SLV0_LEN > 1$), данные считываются с (записываются) последовательных адресов, начиная с I2C_SLV0_REG.

В режиме чтения результат чтения помещается в самый низкий доступный регистр EXT_SENS_DATA. Для получения дополнительной информации о распределении результатов чтения, пожалуйста, обратитесь к описанию регистра EXT_SENS_DATA (регистры 73 - 96).

В режиме записи содержимое I2C_SLV0_DO (регистр 99) будет записано на ведомое устройство.

I2C_SLV0_EN включает ведомое 0 для транзакции данных I2C. Транзакция данных выполняется только в том случае, если между активированным подчиненным устройством ($I2C_SLV0_EN = 1$) должно быть передано более нуля байтов ($I2C_SLV0_LEN > 0$).

I2C_SLV0_BYTE_SW настраивает обмен байтов пар слов. Когда включена замена байтов, старший и младший байты пары слов меняются местами. Пожалуйста, обратитесь к I2C_SLV0_GRP для соглашения о соединении пар слов. Когда этот бит сброшен в 0, байты, переданные в и из подчиненного 0, будут записаны в регистры EXT_SENS_DATA в порядке их передачи.

Когда I2C_SLV0_REG_DIS установлен в 1, транзакция будет только читать или записывать данные. Если сбросить до 0, транзакция запишет адрес регистра перед чтением или записью данных. Этот бит должен быть равен 0 при указании адреса регистра в подчиненном устройстве, с которого будет происходить последующая транзакция данных.

I2C_SLV0_GRP определяет порядок группировки пар слов, полученных из регистров. При очистке до 0 байты из адресов регистров 0 и 1, 2 и 3 и т. Д. (Четные, а затем нечетные адреса регистров) соединяются в одно слово. При значении 1 байты из адресов регистров соединяются в пары 1 и 2, 3 и 4 и т. Д. (Нечетные, а затем четные адреса регистров) в пару.

Транзакции данных I2C выполняются с частотой дискретизации, как это определено в регистре 25. Пользователь несет ответственность за обеспечение того, чтобы транзакции данных I2C с каждого включенного ведомого устройства могли быть завершены в течение одного периода с частотой дискретизации.

Скорость подчиненного доступа I2C может быть уменьшена относительно частоты дискретизации. Эта уменьшенная скорость доступа определяется I2C_MST_DLY (регистр 52). Уменьшается ли скорость доступа подчиненного устройства по отношению к частоте выборки, определяется I2C_MST_DELAY_CTRL (регистр 103).

Порядок обработки для рабов фиксирован. Последовательность, используемая для обработки ведомых устройств: подчиненный 0, подчиненный 1, подчиненный 2, подчиненный 3 и подчиненный 4. Если определенный подчиненный отключен, он будет пропущен.

Доступ к каждому ведомому устройству может быть либо с частотой дискретизации, либо с пониженной частотой дискретизации. В случае, когда к некоторым ведомым получают доступ с частотой дискретизации, а к некоторым ведомым обращаются с пониженной скоростью, последовательность доступа к ведомым (от ведомого 0 до ведомого 4) все еще соблюдается. Однако ведомые устройства с пониженной скоростью будут пропущены, если их скорость доступа диктует, что к ним нельзя обращаться во время этого конкретного цикла. Для получения дополнительной информации об уменьшенной скорости доступа, пожалуйста, обратитесь к Регистру 52. Доступ к подчиненному устройству с частотой дискретизации или с уменьшенной скоростью определяется битами разрешения задержки в регистре 103.

Параметры:

I2C_SLV0_RW При значении 1 этот бит конфигурирует передачу данных как операцию чтения. Когда этот бит сброшен в 0, этот бит конфигурирует передачу данных как операцию записи.

I2C_SLV0_ADDR 7-битный адрес I2C ведомого 0.

I2C_SLV0_REG 8-битный адрес регистра ведомого 0, в / с которого начинается передача данных.

I2C_SLV0_EN При значении 1 этот бит включает ведомое 0 для операций передачи данных. При сбросе в 0 этот бит отключает ведомое 0 от операций передачи данных.

I2C_SLV0_BYTE_SW При значении 1 этот бит разрешает обмен байтов. Когда включена замена байтов, старший и младший байты пары слов меня-

ются местами. Пожалуйста, обратитесь к I2C_SLV0_GRP для соглашения о соединении пар слов.

При очистке до 0 байты, передаваемые в и из подчиненного 0, будут записываться в регистры EXT_SENS_DATA в порядке их передачи.

I2C_SLV0_REG_DIS При значении 1 транзакция будет считывать или записывать только данные. Если сбросить до 0, транзакция запишет адрес регистра перед чтением или записью данных.

I2C_SLV0_GRP 1-битное значение, определяющее порядок группировки пар слов, полученных из регистров. При очистке до 0 байты из адресов регистров 0 и 1, 2 и 3 и т. Д. (Четные, а затем нечетные адреса регистров) соединяются в одно слово. При значении 1 байты из адресов регистров соединяются в пары 1 и 2, 3 и 4 и т. Д. (Нечетные, а затем четные адреса регистров) в пару.

I2C_SLV0_LEN 4-битное беззнаковое значение. Определяет количество байтов, переданных в и из ведомого 0. Сброс этого бита до 0 эквивалентен отключению регистра путем записи 0 в I2C_SLV0_EN.

Пример замены байтов

В следующем примере демонстрируется замена байтов для I2C_SLV0_BYTE_SW = 1, I2C_SLV0_GRP = 0, I2C_SLV0_REG = 0x01 и I2C_SLV0_LEN = 0x4:

1. Первый байт, считанный из регистра ведомого 0 0x01, будет сохранен в EXT_SENS_DATA_00. Поскольку I2C_SLV0_GRP = 0, байты из четного, то нечетные адреса регистров будут объединены в пары слов. Поскольку операция чтения началась с нечетного адреса регистра вместо четного адреса, читается только один байт.

2. Второй и третий байты будут поменяны местами, поскольку I2C_SLV0_BYTE_SW = 1 и I2C_SLV0_REG [0] = 1. Данные, считанные из 0x02, будут сохранены в EXT_SENS_DATA_02, а данные, считанные из 0x03, будут сохранены в EXT_SENS_DATA_01.

3. Последний байт, считанный с адреса 0x04, будет сохранен в EXT_SENS_DATA_03. Поскольку в операции чтения остается только один байт, замена байтов не произойдет.

Пример подчиненного доступа

Доступ к ведомому 0 осуществляется с частотой дискретизации, а к ведомому 1 - с половиной частоты дискретизации. Другие рабы отключены. В первом цикле будут доступны и ведомый 0, и ведомый 1. Однако во втором цикле будет доступен только подчиненный 0. В третьем цикле будут доступны и ведомый 0, и ведомый 1. В четвертом цикле будет доступен только Slave 0. Эта картина продолжается.

4.9 Регистры с 40 по 42 - I2C Slave 1 Control

I2C_SLV1_ADDR, I2C_SLV1_REG и I2C_SLV1_CTRL

Тип: чтение / запись

Register (Hex)	Register (Decimal)	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
28	40	I2C_SLV1_RW	I2C_SLV1_ADDR[6:0]						
29	41	I2C_SLV1_REG[7:0]							
2A	42	I2C_SLV1_EN	I2C_SLV1_BYTE_SW	I2C_SLV1_REG_DIS	I2C_SLV1_GRP	I2C_SLV1_LEN[3:0]			

Описание:

Эти регистры описывают последовательность передачи данных для ведомого 1. Их функции соответствуют функциям, описанным для регистров ведомого 0 (регистры с 37 по 39).

4.10 Регистры с 43 по 45 - I2C Slave 2 Control**I2C_SLV2_ADDR, I2C_SLV2_REG и I2C_SLV2_CTRL**

Тип: чтение / запись

Register (Hex)	Register (Decimal)	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
2B	43	I2C_SLV2_RW	I2C_SLV2_ADDR[6:0]						
2C	44	I2C_SLV2_REG[7:0]							
2D	45	I2C_SLV2_EN	I2C_SLV2_BYTE_SW	I2C_SLV2_REG_DIS	I2C_SLV2_GRP	I2C_SLV2_LEN[3:0]			

Описание:

Эти регистры описывают последовательность передачи данных для ведомого 2. Их функции соответствуют функциям, описанным для регистров ведомого 0 (регистры с 37 по 39).

4.11 Регистры с 46 по 48 - I2C Slave 3 Control**I2C_SLV3_ADDR, I2C_SLV3_REG и I2C_SLV3_CTRL**

Тип: чтение / запись

Register (Hex)	Register (Decimal)	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
2E	46	I2C_SLV3_RW	I2C_SLV3_ADDR[6:0]						
2F	47	I2C_SLV3_REG[7:0]							
30	48	I2C_SLV3_EN	I2C_SLV3_BYTE_SW	I2C_SLV3_REG_DIS	I2C_SLV3_GRP	I2C_SLV3_LEN[3:0]			

Описание:

Эти регистры описывают последовательность передачи данных для ведомого 3. Их функции соответствуют функциям, описанным для регистров ведомого 0 (регистры с 37 по 39).

4.12 Регистры с 49 по 53 - I2C Slave 4 Control**I2C_SLV4_ADDR, I2C_SLV4_REG, I2C_SLV4_DO, I2C_SLV4_CTRL и I2C_SLV4_DI**

Тип: чтение / запись

Register (Hex)	Register (Decimal)	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
31	49	I2C_SLV4_RW	I2C_SLV4_ADDR[6:0]						
32	50	I2C_SLV4_REG[7:0]							
33	51	I2C_SLV4_DO[7:0]							
34	52	I2C_SLV4_EN	I2C_SLV4_INT_EN	I2C_SLV4_REG_DIS	I2C_MST_DLY[4:0]				
35	53	I2C_SLV4_DI[7:0]							

Описание:

Эти регистры описывают последовательность передачи данных для подчиненного устройства 4. Характеристики ведомого устройства 4 сильно отличаются от

характеристик ведомого устройства 0-3. Для получения дополнительной информации о характеристиках ведомых 0-3, пожалуйста, обратитесь к регистрам 37-48.

Транзитные данные I2C между MPU-60X0 и Slave 4 задаются как операции чтения или записи битом I2C_SLV4_RW. Когда этот бит равен 1, передача является операцией чтения. Когда бит равен 0, передача является операцией записи.

I2C_SLV4_ADDR используется для указания адреса ведомого I2C подчиненного устройства 4.

Передача данных начинается с внутреннего регистра в подчиненном устройстве 4. Этот адрес регистра определяется I2C_SLV4_REG.

В режиме чтения результат чтения будет доступен в I2C_SLV4_DI. В режиме записи содержимое I2C_SLV4_DO будет записано на ведомое устройство.

Транзакция данных выполняется, только если бит I2C_SLV4_EN установлен в 1. Транзакция данных должна быть включена после того, как ее параметры сконфигурированы в регистрах _ADDR и _REG. Для записи также необходим регистр _DO. I2C_SLV4_EN будет очищен после выполнения транзакции один раз.

Прерывание запускается при завершении транзакции данных подчиненного устройства 4, если прерывание включено. Статус этого прерывания можно наблюдать в Регистре 54.

Когда I2C_SLV4_REG_DIS установлен в 1, транзакция будет считывать или записывать данные вместо записи адреса регистра. Этот бит должен быть равен 0 при указании адреса регистра в подчиненном устройстве, с которого будет происходить последующая транзакция данных.

I2C_MST_DLY настраивает сниженную скорость доступа подчиненных I2C относительно частоты дискретизации. Когда скорость доступа подчиненного устройства уменьшается относительно частоты дискретизации, доступ к подчиненному устройству осуществляется каждый раз.

$$1 / (1 + I2C_MST_DLY) \text{ выборки}$$

Эта базовая частота выборки, в свою очередь, определяется SMPLRT_DIV (регистр 25) и DLPF_CFG (регистр 26). Уменьшается ли скорость доступа подчиненного устройства по отношению к Частоте выборки, определяется I2C_MST_DELAY_CTRL (регистр 103).

Для получения дополнительной информации о частоте дискретизации, пожалуйста, обратитесь к регистру 25.

Ведомые 4 транзакции выполняются после того, как ведомые 0, 1, 2 и 3 транзакции были завершены. Таким образом, максимальная скорость для транзакций подчиненного устройства 4 определяется частотой дискретизации, определенной в регистре 25.

Параметры:

I2C_SLV4_RW При значении 1 этот бит настраивает передачу данных как операцию чтения. Когда этот бит сброшен в 0, этот бит конфигурирует передачу данных как операцию записи.

I2C_SLV4_ADDR 7-битный адрес I2C для подчиненного устройства 4.

I2C_SLV4_REG 8-битный адрес регистра Slave 4, в который начинается передача данных.

I2C_SLV4_DO Этот регистр хранит данные для записи в подчиненное устройство 4. Если I2C_SLV4_RW установлен в 1 (установлен на чтение), этот регистр не имеет никакого эффекта.

I2C_SLV4_EN При значении 1 этот бит включает ведомое устройство 4 для операций передачи данных. Если этот бит сброшен до 0, этот бит отключает ведомое устройство 4 от операций передачи данных.

I2C_SLV4_INT_EN При значении 1 этот бит позволяет генерировать сигнал прерывания после завершения транзакции подчиненного устройства 4. Когда этот бит сброшен до 0, этот бит отключает генерацию сигнала прерывания после завершения транзакции подчиненного устройства 4.

Статус прерывания можно наблюдать в Регистре 54.

I2C_SLV4_REG_DIS При значении 1 транзакция будет считывать или записывать данные. При сбросе до 0 транзакция будет считывать или записывать адрес регистра.

I2C_MST_DLY Конфигурирует сниженную скорость доступа подчиненных устройств относительно частоты выборки.

I2C_SLV4_DI Этот регистр хранит данные, считанные с подчиненного устройства 4.

Это поле заполняется после транзакции чтения.

4.13 Register 54 – I2C Master Status

I2C_MST_STATUS

Type: Read Only

Register (Hex)	Register (Decimal)	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
36	54	PASS_THROUGH	I2C_SLV4_DONE	I2C_LOST_ARB	I2C_SLV4_NACK	I2C_SLV3_NACK	I2C_SLV2_NACK	I2C_SLV1_NACK	I2C_SLV0_NACK

Описание:

Этот регистр показывает состояние генерирующих прерывание сигналов в ведущем устройстве I2C в MPU-60X0. Этот регистр также сообщает статус прерывания FSYNC хост-процессору.

Чтение этого регистра очистит все биты состояния в регистре.

Параметры:

PASS_THROUGH Этот бит отражает состояние прерывания FSYNC от внешнего устройства в MPU-60X0. Это используется для передачи внешнего прерывания через MPU-60X0 в процессор приложения хоста. При значении 1 этот бит будет вызывать прерывание, если FSYNC_INT_EN установлен в INT_PIN_CFG (Регистр 55).

I2C_SLV4_DONE Автоматически устанавливается в 1, когда транзакция Slave 4 завершена. Это инициирует прерывание, если бит I2C_MST_INT_EN в регистре INT_ENABLE (регистр 56) установлен и если бит SLV_4_DONE_INT установлен в регистре I2C_SLV4_CTRL (регистр 52).

I2C_LOST_ARB Этот бит автоматически устанавливается в 1, когда мастер I2C потерял арбитраж вспомогательной шины I2C (состояние ошибки). Это вызывает прерывание, если установлен бит I2C_MST_INT_EN в регистре INT_ENABLE (регистр 56).

I2C_SLV4_NACK Этот бит автоматически устанавливается в 1, когда мастер I2C получает NACK в транзакции с ведомым 4. Это запускает прерывание, если бит I2C_MST_INT_EN в регистре INT_ENABLE (регистр 56) установлен.

I2C_SLV3_NACK Этот бит автоматически устанавливается в 1, когда мастер I2C получает NACK в транзакции с ведомым 3. Это запускает прерывание, если бит I2C_MST_INT_EN в регистре INT_ENABLE (регистр 56) установлен.

I2C_SLV2_NACK Этот бит автоматически устанавливается в 1, когда мастер I2C получает NACK в транзакции с ведомым 2. Это запускает прерывание, если бит I2C_MST_INT_EN в регистре INT_ENABLE (регистр 56) установлен.

I2C_SLV1_NACK Этот бит автоматически устанавливается в 1, когда мастер I2C получает NACK в транзакции с ведомым 1. Это запускает прерывание, если бит I2C_MST_INT_EN в регистре INT_ENABLE (регистр 56) установлен.

I2C_SLV0_NACK Этот бит автоматически устанавливается в 1, когда мастер I2C получает NACK в транзакции с ведомым 0. Это запускает прерывание, если бит I2C_MST_INT_EN в регистре INT_ENABLE (регистр 56) установлен.

4.14 Регистр 55 - Конфигурация включения пин / байпаса INT INT_PIN_CFG

Тип: чтение / запись

Register (Hex)	Register (Decimal)	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
37	55	INT_LEVEL	INT_OPEN	LATCH_INT_EN	INT_RD_CLEAR	FSYNC_INT_LEVEL	FSYNC_INT_EN	I2C_BYPASS_EN	-

Описание:

Этот регистр настраивает поведение сигналов прерывания на выводах INT. Этот регистр также используется для включения контакта FSYNC в качестве прерывания для процессора главного приложения, а также для включения режима байпаса на ведомом устройстве I2C. Этот бит также включает вывод тактовых импульсов.

FSYNC_INT_EN позволяет использовать вывод FSYNC в качестве прерывания для процессора приложения хоста. Переход к активному уровню, указанному в FSYNC_INT_LEVEL, вызовет прерывание. Состояние этого прерывания считывается из бита PASS_THROUGH в регистре основного состояния I2C (регистр 54).

Когда I2C_BYPASS_EN равен 1, а I2C_MST_EN (регистр 106 бит [5]) равен 0, процессор приложения хоста сможет напрямую обращаться к вспомогательной шине I2C MPU-60X0. Когда этот бит равен 0, процессор приложения хоста не сможет напрямую обращаться к вспомогательной шине I2C MPU-60X0 независимо от состояния I2C_MST_EN.

Для получения дополнительной информации о режиме байпаса, пожалуйста, обратитесь к разделу 7.11 и 7.13 документа спецификации продукта MPU-6000 / MPU-6050.

Параметры:

INT_LEVEL Когда этот бит равен 0, логический уровень для контакта INT является активным. Когда этот бит равен 1, логический уровень для контакта INT является активным низким.

INT_OPEN Когда этот бит равен 0, вывод INT конфигурируется как двухтактный. Когда этот бит равен 1, вывод INT конфигурируется как открытый сток.

LATCH_INT_EN Когда этот бит равен 0, вывод INT излучает импульс длительностью 50 мкс. Когда этот бит равен 1, вывод INT удерживается на высоком уровне, пока прерывание не будет очищено.

INT_RD_CLEAR Когда этот бит равен 0, биты состояния прерывания очищаются только путем чтения INT_STATUS (Регистр 58) Когда этот бит равен 1, биты состояния прерывания очищаются при любой операции чтения.

FSYNC_INT_LEVEL Когда этот бит равен 0, логический уровень для контакта FSYNC (при использовании в качестве прерывания для хост-процессора) является активным высоким. Когда этот бит равен 1, логический уровень для вывода FSYNC (при использовании в качестве прерывания для хост-процессора) является активным низким.

FSYNC_INT_EN Когда этот бит равен 0, этот бит отключает вывод FSYNC от прерывания хост-процессора. При значении 1 этот бит позволяет использовать вывод FSYNC в качестве прерывания для хост-процессора

I2C_BYPASS_EN Когда этот бит равен 1, а I2C_MST_EN (регистр 106 бит [5]) равен 0, процессор приложения хоста сможет напрямую обращаться к вспомогательной шине I2C MPU-60X0. Когда этот бит равен 0, процессор приложения хоста не сможет напрямую обращаться к вспомогательной шине I2C MPU-60X0 независимо от состояния I2C_MST_EN (регистр 106 бит [5]).

4.15 Регистр 56 - Прерывание разрешено INT_ENABLE

Тип: чтение / запись

Register (Hex)	Register (Decimal)	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
38	56		-		FIFO_OFLOW_EN	I2C_MST_INT_EN	-	-	DATA_RDY_EN

Описание:

Этот регистр позволяет генерировать прерывания источниками прерываний.

Для получения информации о статусе прерывания для каждого источника генерации прерывания, пожалуйста, обратитесь к Регистру 58. Дополнительную информацию относительно генерации основного прерывания I2C можно найти в Регистре 54.

Биты 2 и 1 зарезервированы.

Параметры:

FIFO_OFLOW_EN При значении 1 этот бит включает переполнение буфера FIFO для генерации прерывания.

I2C_MST_INT_EN При значении 1 этот бит позволяет любому из источников прерываний мастера I2C генерировать прерывание.

DATA_RDY_EN При значении 1 этот бит включает прерывание готовности данных, которое происходит каждый раз, когда завершается операция записи во все регистры датчика.

4.16 Регистр 58 - Состояние прерывания INT_STATUS

Тип: только чтение

Register (Hex)	Register (Decimal)	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
3A	58	-	-	-	FIFO_OFLOW_INT	I2C_MST_INT	-	-	DATA_RDY_INT

Описание:

Этот регистр показывает состояние прерывания каждого источника генерации прерывания. Каждый бит очищается после чтения регистра.

Информацию о соответствующих битах разрешения прерываний см. В регистре 56.

Список основных прерываний I2C приведен в Регистре 54.

Биты 2 и 1 зарезервированы.

Параметры:

FIFO_OFLOW_INT Этот бит автоматически устанавливается в 1, когда генерируется прерывание переполнения буфера FIFO. Бит сбрасывается в 0 после считывания регистра.

I2C_MST_INT Этот бит автоматически устанавливается в 1, когда генерируется главное прерывание I2C. Список основных прерываний I2C приведен в Регистре 54. Бит сбрасывается в 0 после считывания регистра.

DATA_RDY_INT Этот бит автоматически устанавливается в 1, когда генерируется прерывание готовности данных. Бит сбрасывается в 0 после считывания регистра.

4.17 Регистры с 59 по 64 - измерения акселерометра

ACCEL_XOUT_H, ACCEL_XOUT_L, ACCEL_YOUT_H, ACCEL_YOUT_L, ACCEL_ZOUT_H и ACCEL_ZOUT_L

Тип: только чтение

Register (Hex)	Register (Decimal)	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
3B	59	ACCEL_XOUT[15:8]							
3C	60	ACCEL_XOUT[7:0]							
3D	61	ACCEL_YOUT[15:8]							
3E	62	ACCEL_YOUT[7:0]							
3F	63	ACCEL_ZOUT[15:8]							
40	64	ACCEL_ZOUT[7:0]							

Описание:

Эти регистры хранят самые последние измерения акселерометра.

Измерения акселерометра записываются в эти регистры с частотой дискретизации, определенной в регистре 25.

Регистры измерения акселерометра, наряду с регистрами измерения температуры, регистрами измерения гироскопа и регистрами данных внешнего датчика, состоят из двух наборов регистров: набора внутренних регистров и набора регистров считывания, обращенного к пользователю.

Данные во внутреннем регистре датчиков акселерометра всегда обновляются с частотой дискретизации. Между тем, пользовательский набор регистров чтения дублирует значения данных внутреннего набора регистров всякий раз, когда последовательный интерфейс простаивает. Это гарантирует, что пакетное считывание регистров датчика будет считывать измерения с одного и того же момента выборки. Обратите внимание, что если пакетные чтения не используются, пользователь несет ответственность за обеспечение того, чтобы набор однобайтовых чтений соответствовал одному моменту выборки, проверяя прерывание Data Ready.

Каждое 16-разрядное измерение акселерометра имеет полную шкалу, определенную в ACCEL_FS (регистр 28). Для каждой настройки полной шкалы чувствительность акселерометров на LSB в ACCEL_xOUT показана в таблице ниже.

AFS_SEL	Full Scale Range	LSB Sensitivity
0	$\pm 2g$	16384 LSB/g
1	$\pm 4g$	8192 LSB/g
2	$\pm 8g$	4096 LSB/g
3	$\pm 16g$	2048 LSB/g

Параметры:

ACCEL_XOUT 16-битное значение дополнения до 2.

Хранит самое последнее измерение акселерометра оси X.

ACCEL_YOUT 16-битное значение дополнения до 2.

Сохраняет самое последнее измерение акселерометра оси Y.

ACCEL_ZOUT 16-битное значение дополнения до 2.

Сохраняет самое последнее измерение по оси Z

4.18 Регистры 65 и 66 - измерение температуры

TEMP_OUT_H и TEMP_OUT_L

Тип: только чтение

Register (Hex)	Register (Decimal)	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
41	65	TEMP_OUT[15:8]							
42	66	TEMP_OUT[7:0]							

Описание:

Эти регистры хранят самые последние измерения датчика температуры.

Измерения температуры записываются в эти регистры с частотой дискретизации, определенной в регистре 25.

Эти регистры измерения температуры, наряду с регистрами измерения акселерометра, регистрами измерения гироскопа и регистрами данных внешнего датчика, состоят из двух наборов регистров: набора внутренних регистров и набора регистров считывания для пользователя.

Данные в наборе внутренних регистров датчика температуры всегда обновляются с частотой дискретизации. Между тем, пользовательский набор регистров чтения дублирует значения данных внутреннего набора регистров всякий раз, когда последовательный интерфейс простаивает. Это гарантирует, что пакетное считывание регистров датчика будет считывать измерения с одного и того же момента выборки. Обратите внимание, что если пакетные чтения не используются, пользователь несет ответственность за обеспечение того, чтобы набор однобайтовых чтений соответствовал одному моменту выборки, проверяя прерывание Data Ready.

Масштабный коэффициент и смещение для датчика температуры приведены в таблице «Электрические характеристики» (раздел 6.4 документа «Технические характеристики изделия MPU-6000 / MPU-6050»).

Температура в градусах Цельсия для данного значения регистра может быть вычислена как:

Температура в градусах C = (значение регистра TEMP_OUT в виде количества со знаком) / 340 + 36,53

Обратите внимание, что математика в приведенном выше уравнении в десятичной системе счисления.

Параметры:

TEMP_OUT 16-битное значение со знаком.

Хранит самые последние измерения датчика температуры.

4.19 Регистры с 67 по 72 - Гироскопические измерения

GYRO_XOUT_H, GYRO_XOUT_L, GYRO_YOUT_H, GYRO_YOUT_L, GYRO_ZOUT_H и GYRO_ZOUT_L

Тип: только чтение

Register (Hex)	Register (Decimal)	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
43	67	GYRO_XOUT[15:8]							
44	68	GYRO_XOUT[7:0]							
45	69	GYRO_YOUT[15:8]							
46	70	GYRO_YOUT[7:0]							
47	71	GYRO_ZOUT[15:8]							
48	72	GYRO_ZOUT[7:0]							

Описание:

Эти регистры хранят самые последние измерения гироскопа.

Измерения гироскопа записываются в эти регистры с частотой дискретизации, определенной в регистре 25.

Эти регистры измерения гироскопа, наряду с регистрами измерения акселерометра, регистрами измерения температуры и регистрами данных внешнего датчика, состоят из двух наборов регистров: набора внутренних регистров и набора регистров считывания, обращенного к пользователю.

Данные в наборе внутренних регистров датчиков гироскопа всегда обновляются с частотой дискретизации. Между тем, пользовательский набор регистров чтения дублирует значения данных внутреннего набора регистров всякий раз, когда последовательный интерфейс простаивает. Это гарантирует, что пакетное считывание регистров датчика будет считывать измерения с одного и того же момента выборки. Обратите внимание, что если пакетные чтения не используются, пользователь несет ответственность за обеспечение того, чтобы набор однобайтовых чтений соответствовал одному моменту выборки, проверяя прерывание Data Ready.

Каждое 16-битное измерение гироскопа имеет полную шкалу, определенную в FS_SEL (Регистр 27). Для каждой полной шкалы чувствительность гироскопов на LSB в GYRO_xOUT показана в таблице ниже:

FS_SEL	Full Scale Range	LSB Sensitivity
0	± 250 °/s	131 LSB/°/s
1	± 500 °/s	65.5 LSB/°/s
2	± 1000 °/s	32.8 LSB/°/s
3	± 2000 °/s	16.4 LSB/°/s

Параметры:

GYRO_XOUT 16-битное значение дополнения до 2.

Сохраняет самое последнее измерение оси X-гироскопа.

GYRO_YOUT 16-битное значение дополнения до 2.

Хранит самое последнее измерение гироскопа оси Y.

GYRO_ZOUT 16-битное значение дополнения до 2.

Сохраняет самое последнее измерение оси Z-гироскопа.

4.20 Регистры с 73 по 96 - данные внешнего датчика**EXT_SENS_DATA_00 - EXT_SENS_DATA_23**

Тип: только чтение

Register (Hex)	Register (Decimal)	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
49	73	EXT_SENS_DATA_00[7:0]							
4A	74	EXT_SENS_DATA_01[7:0]							
4B	75	EXT_SENS_DATA_02[7:0]							
4C	76	EXT_SENS_DATA_03[7:0]							
4D	77	EXT_SENS_DATA_04[7:0]							
4E	78	EXT_SENS_DATA_05[7:0]							
4F	79	EXT_SENS_DATA_06[7:0]							
50	80	EXT_SENS_DATA_07[7:0]							
51	81	EXT_SENS_DATA_08[7:0]							
52	82	EXT_SENS_DATA_09[7:0]							
53	83	EXT_SENS_DATA_10[7:0]							
54	84	EXT_SENS_DATA_11[7:0]							
55	85	EXT_SENS_DATA_12[7:0]							
56	86	EXT_SENS_DATA_13[7:0]							
57	87	EXT_SENS_DATA_14[7:0]							
58	88	EXT_SENS_DATA_15[7:0]							
59	89	EXT_SENS_DATA_16[7:0]							
5A	90	EXT_SENS_DATA_17[7:0]							
5B	91	EXT_SENS_DATA_18[7:0]							
5C	92	EXT_SENS_DATA_19[7:0]							
5D	93	EXT_SENS_DATA_20[7:0]							
5E	94	EXT_SENS_DATA_21[7:0]							
5F	95	EXT_SENS_DATA_22[7:0]							
60	96	EXT_SENS_DATA_23[7:0]							

Описание:

Эти регистры хранят данные, считанные с внешних датчиков ведомым 0, 1, 2 и 3 на вспомогательном интерфейсе I2C. Данные, считанные ведомым 4, сохраняются в I2C_SLV4_DI (регистр 53).

Данные внешнего датчика записываются в эти регистры с частотой дискретизации, определенной в регистре 25. Эту скорость доступа можно уменьшить с помощью регистров включения ведомой задержки (регистр 103).

Регистры данных внешнего датчика вместе с регистрами измерения гироскопа, регистрами измерения акселерометра и регистрами измерения температуры состоят из двух наборов регистров: набора внутренних регистров и набора регистров считывания, обращенного к пользователю.

Данные в наборе внутренних регистров внешних датчиков всегда обновляются с частотой дискретизации (или сниженной скоростью доступа) всякий раз, когда последовательный интерфейс простаивает. Это гарантирует, что пакетное считыва-

ние регистров датчика будет считывать измерения с одного и того же момента выборки. Обратите внимание, что если пакетные чтения не используются, пользователь несет ответственность за обеспечение того, чтобы набор однобайтовых чтений соответствовал одному моменту выборки, проверяя прерывание Data Ready.

Данные помещаются в эти регистры данных внешнего датчика в соответствии с I2C_SLV0_CTRL, I2C_SLV1_CTRL, I2C_SLV2_CTRL и I2C_SLV3_CTRL (регистры 39, 42, 45 и 48). Когда считывается больше нуля байтов ($I2C_SLVx_LEN > 0$) от включенного ведомого ($I2C_SLVx_EN = 1$), ведомое считывается со скоростью выборки (как определено в регистре 25) или с задержкой (если указано в регистре 52 и 103). Во время каждого цикла выборки считывания ведомого устройства выполняются в порядке номера ведомого. Если все ведомые устройства доступны для чтения с более чем нулевым байтом, порядок должен быть ведомым 0, за которым следуют ведомый 1, ведомый 2 и ведомый 3.

Каждое включенное ведомое устройство будет иметь регистры EXT_SENS_DATA, связанные с ним по количеству прочитанных байтов ($I2C_SLVx_LEN$) в порядке номера подчиненного устройства, начиная с EXT_SENS_DATA_00. Обратите внимание, что это означает, что включение или отключение ведомого может изменить регистры, связанные с ведомыми подчиненными. Кроме того, если в результате такого изменения с внешних датчиков будет считано меньше общих байтов, то данные, оставшиеся в регистрах, которые больше не имеют связанного ведомого устройства (т.е. регистры с большими номерами), останутся в этих ранее выделенных регистрах, если только сброс.

Если сумма длин чтения всех транзакций SLVx превышает количество доступных регистров EXT_SENS_DATA, лишние байты будут отброшены. Существует 24 регистра EXT_SENS_DATA, и, следовательно, общая длина чтения между всеми ведомыми устройствами не может превышать 24, иначе некоторые байты будут потеряны.

Примечание: поведение Slave 4 отличается от поведения Slave 0-3. Для получения дополнительной информации о характеристиках подчиненного устройства 4, пожалуйста, обратитесь к регистрам 49-53.

Пример:

Предположим, что ведомому 0 разрешено чтение с 4 байтами ($I2C_SLV0_EN = 1$ и $I2C_SLV0_LEN = 4$), а ведомому 1 разрешено чтение с 2 байтами ($I2C_SLV1_EN = 1$ и $I2C_SLV1_LEN = 2$). В такой ситуации EXT_SENS_DATA_00 - _03 будет связан с ведомым 0, а EXT_SENS_DATA_04 и 05 будет связан с ведомым 1.

Если подчиненное устройство 2 также включено, регистры, начиная с EXT_SENS_DATA_06, будут назначены подчиненному устройству 2.

Если подчиненное устройство 2 отключено, а подчиненное устройство 3 разрешено в той же ситуации, то вместо этого ведомому 3 будут назначены регистры, начиная с EXT_SENS_DATA_06.

Распределение регистра для динамического отключения или обычного отключения

Если ведомое устройство отключено в любое время, пространство, первоначально выделенное ведомому в регистре EXT_SENS_DATA, останется связанным с этим ведомым устройством. Это сделано для того, чтобы избежать динамической настройки распределения регистров.

Распределение регистров EXT_SENS_DATA пересчитывается только тогда, когда (1) все подчиненные устройства отключены или (2) установлен бит I2C_MST_RST (регистр 106).

Это выше также верно, если один из рабов получает NACKed и перестает функционировать.

4.21 Регистр 99 - I2C Slave 0 Data Out I2C_SLV0_DO

Тип: чтение / запись

Register (Hex)	Register (Decimal)	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
63	99	I2C_SLV0_DO[7:0]							

Описание:

Этот регистр содержит выходные данные, записанные в ведомое устройство 0, когда ведомое устройство 0 установлено в режим записи.

Для получения дополнительной информации относительно управления ведомым 0, пожалуйста, обратитесь к регистрам 37-39.

Параметры:

I2C_SLV0_DO 8-битное значение без знака, которое записывается в подчиненное устройство 0, когда подчиненное устройство 0 установлено в режим записи.

4.22 Регистр 100 - выход I2C Slave 1 I2C_SLV1_DO

Тип: чтение / запись

Register (Hex)	Register (Decimal)	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
64	100	I2C_SLV1_DO[7:0]							

Описание:

Этот регистр содержит выходные данные, записанные в подчиненное устройство 1, когда подчиненное устройство 1 установлено в режим записи.

Для получения дополнительной информации относительно управления Slave 1, пожалуйста, обратитесь к регистрам 40-42.

Параметры:

I2C_SLV1_DO 8-битное значение без знака, которое записывается в подчиненное устройство 1, когда подчиненное устройство 1 установлено в режим записи.

4.23 Регистр 101 - I2C Slave 2 Data Out I2C_SLV2_DO

Тип: чтение / запись

Register (Hex)	Register (Decimal)	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
65	101	I2C_SLV2_DO[7:0]							

Описание:

Этот регистр содержит выходные данные, записанные в подчиненное устройство 2, когда подчиненное устройство 2 переведено в режим записи.

Для получения дополнительной информации относительно управления Slave 2, пожалуйста, обратитесь к регистрам 43-45.

Параметры:

I2C_SLV2_DO 8-битное значение без знака, которое записывается в подчиненное устройство 2, когда подчиненное устройство 2 установлено в режим записи.

4.24 Регистр 102 - I2C Slave 3 Data Out

I2C_SLV3_DO

Тип: чтение / запись

Register (Hex)	Register (Decimal)	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
66	102	I2C_SLV3_DO[7:0]							

Описание:

Этот регистр содержит выходные данные, записанные в подчиненное устройство 3, когда подчиненное устройство 3 установлено в режим записи.

Для получения дополнительной информации относительно управления подчиненным устройством 3, пожалуйста, обратитесь к регистрам 46-48.

Параметры:

I2C_SLV3_DO 8-битное значение без знака, которое записывается в подчиненное устройство 3, когда подчиненное устройство 3 установлено в режим записи.

4.25 Регистр 103 - I2C Master Delay Control

I2C_MST_DELAY_CTRL

Тип: чтение / запись

Register (Hex)	Register (Decimal)	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
67	103	DELAY_ES_SHADOW	-	-	I2C_SLV4_DLY_EN	I2C_SLV3_DLY_EN	I2C_SLV2_DLY_EN	I2C_SLV1_DLY_EN	I2C_SLV0_DLY_EN

Описание:

Этот регистр используется для указания времени затенения данных внешнего датчика. Регистр также используется для уменьшения скорости доступа подчиненных устройств относительно частоты дискретизации.

Когда DELAY_ES_SHADOW установлен в 1, теневое копирование данных внешнего датчика задерживается до тех пор, пока все данные не будут получены.

Когда I2C_SLV4_DLY_EN, I2C_SLV3_DLY_EN, I2C_SLV2_DLY_EN, I2C_SLV1_DLY_EN и I2C_SLV0_DLY_EN включены, скорость доступа для соответствующих подчиненных устройств уменьшается.

Когда скорость доступа подчиненного устройства уменьшается относительно частоты дискретизации, доступ к подчиненному устройству осуществляется каждый раз.

$$1 / (1 + I2C_MST_DLY) \text{ выборки.}$$

Эта базовая частота выборки, в свою очередь, определяется SMPLRT_DIV (регистр 25) и DLPF_CFG (регистр 26).

Для получения дополнительной информации о I2C_MST_DLY, пожалуйста, обратитесь к регистру 52.

Для получения дополнительной информации о частоте дискретизации, пожалуйста, обратитесь к регистру 25.

Биты 6 и 5 зарезервированы.

Параметры:

DELAY_ES_SHADOW Если установлено, задерживает теневое копирование данных внешнего датчика, пока все данные не будут получены.

I2C_SLV4_DLY_EN Если включено, к подчиненному устройству 4 будет доступ только с пониженной скоростью.

I2C_SLV3_DLY_EN Если включено, к подчиненному устройству 3 будет доступ только с пониженной скоростью.

I2C_SLV2_DLY_EN При включении, ведомый 2 будет доступен только с пониженной скоростью.

I2C_SLV1_DLY_EN Если включено, доступ к ведомому 1 возможен только с пониженной скоростью.

I2C_SLV0_DLY_EN При включении, ведомый 0 будет доступен только с пониженной скоростью.

4.26 Регистр 104 - сброс пути сигнала

SIGNAL_PATH_RESET

Тип: только запись

Register (Hex)	Register (Decimal)	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
68	104	-	-	-	-	-	GYRO_RESET	ACCEL_RESET	TEMP_RESET

Описание:

Этот регистр используется для сброса аналоговых и цифровых трактов сигналов гироскопа, акселерометра и датчиков температуры.

Сброс вернет путь прохождения сигнала аналого-цифровых преобразователей и фильтров к их конфигурации включения питания.

Примечание. Этот регистр не очищает регистры датчика. Сброс также инициализирует последовательный интерфейс.

Биты с 7 по 3 зарезервированы.

Параметры:

GYRO_RESET При значении 1 этот бит сбрасывает пути аналогового и цифрового сигнала гироскопа.

ACCEL_RESET При значении 1 этот бит сбрасывает пути аналогового и цифрового сигнала акселерометра.

TEMP_RESET При значении 1 этот бит сбрасывает пути аналогового и цифрового сигналов датчика температуры.

4.27 Регистр 106 - Контроль пользователя

USER_CTRL

Тип: чтение / запись

Register (Hex)	Register (Decimal)	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
6A	106	-	FIFO_EN	I2C_MST_EN	I2C_IF_DIS	-	FIFO_RESET	I2C_MST_RESET	SIG_COND_RESET

Описание:

Этот регистр позволяет пользователю включать и отключать буфер FIFO, основной режим I2C и основной интерфейс I2C. Буфер FIFO, I2C Master, пути сигналов датчиков и регистры датчиков также могут быть сброшены с помощью этого регистра.

Когда I2C_MST_EN установлен в 1, основной режим I2C включен. В этом режиме MPU-60X0 действует как ведущее устройство I2C для ведомых устройств внешнего датчика на вспомогательной шине I2C. Когда этот бит сбрасывается в 0, вспомогательные линии шины I2C (AUX_DA и AUX_CL) логически управляются первичной шиной I2C (SDA и SCL). Это предварительное условие для включения режима байпаса. Для получения дополнительной информации о режиме байпаса, пожалуйста, обратитесь к регистру 55.

MPU-6000: основной интерфейс SPI будет включен вместо отключенного основного интерфейса I2C, когда I2C_IF_DIS установлен в 1.

MPU-6050: всегда записывать 0 в I2C_IF_DIS.

Когда биты сброса (FIFO_RESET, I2C_MST_RESET и SIG_COND_RESET) установлены в 1, эти биты сброса иницииируют сброс, а затем сбрасываются в 0.

Биты 7 и 3 зарезервированы.

Параметры:

FIFO_EN При значении 1 этот бит разрешает операции FIFO. Когда этот бит сброшен в 0, буфер FIFO отключен. Буфер FIFO нельзя записывать или читать, пока он отключен.

Состояние буфера FIFO не изменяется, если MPU-60X0 в циклическом режиме.

I2C_MST_EN При значении 1 этот бит включает основной режим I2C. Когда этот бит сбрасывается в 0, вспомогательные линии шины I2C (AUX_DA и AUX_CL) логически управляются первичной шиной I2C (SDA и SCL).

I2C_IF_DIS MPU-6000: при значении 1 этот бит отключает основной интерфейс I2C и вместо этого включает интерфейс SPI.

MPU-6050: всегда записывать этот бит как ноль.

FIFO_RESET Этот бит сбрасывает буфер FIFO, когда установлен в 1, тогда как FIFO_EN равен 0. Этот бит автоматически сбрасывается в 0 после запуска сброса.

I2C_MST_RESET Этот бит сбрасывает мастер I2C, когда он установлен в 1, тогда как I2C_MST_EN равен 0. Этот бит автоматически сбрасывается в 0 после запуска сброса.

SIG_COND_RESET При значении 1 этот бит сбрасывает пути прохождения сигналов для всех датчиков (гироскопов, акселерометров и датчиков температуры). Эта операция также очистит регистры датчика. Этот бит автоматически сбрасывается в 0 после запуска сброса.

При сбросе только пути прохождения сигнала (а не регистров датчика) используйте регистр 104, SIGNAL_PATH_RESET.

4.28 Регистр 107 - Управление питанием 1

PWR_MGMT_1

Тип: чтение / запись

Register (Hex)	Register (Decimal)	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
6B	107	DEVICE_RESET	SLEEP	CYCLE	-	TEMP_DIS	CLKSEL[2:0]		

Описание:

Этот регистр позволяет пользователю настроить режим питания и источник синхронизации. Он также предоставляет бит для сброса всего устройства и бит для отключения датчика температуры.

Установив SLEEP в 1, MPU-60X0 можно перевести в режим ожидания с низким энергопотреблением. Когда CYCLE установлен в 1, а SLEEP отключен, MPU-60X0 будет переведен в режим цикла. В циклическом режиме устройство циклически переключается между режимом ожидания и пробуждением, чтобы взять одну выборку данных из акселерометра со скоростью, определенной LP_WAKE_CTRL (регистр 108). Чтобы настроить частоту пробуждения, используйте LP_WAKE_CTRL в регистре управления питанием 2 (регистр 108).

В качестве источника тактового сигнала MPU-60X0 можно выбрать внутренний генератор 8 МГц, часы на основе гироскопа или внешние источники. Если в качестве источника синхронизации выбран внутренний генератор 8 МГц или внешний источник, MPU-60X0 может работать в режимах малой мощности с отключенными гироскопами.

При включении питания тактовый источник MPU-60X0 по умолчанию использует внутренний генератор. Однако ***настоятельно рекомендуется, чтобы устройство было настроено на использование одного из гироскопов (или внешнего источника синхронизации) в качестве эталона синхронизации для повышения стабильности.*** Источник часов может быть выбран в соответствии со следующей таблицей.

CLKSEL	Clock Source
0	Internal 8MHz oscillator
1	PLL with X axis gyroscope reference
2	PLL with Y axis gyroscope reference
3	PLL with Z axis gyroscope reference
4	PLL with external 32.768kHz reference
5	PLL with external 19.2MHz reference
6	Reserved
7	Stops the clock and keeps the timing generator in reset

Для получения дополнительной информации о тактовом источнике MPU-60X0, пожалуйста, обратитесь к документу спецификации продукта MPU-6000 / MPU-6050.

Бит 4 зарезервирован.

Параметры:

DEVICE_RESET При значении 1 этот бит сбрасывает все внутренние регистры к значениям по умолчанию.

Бит автоматически сбрасывается в 0 после сброса.

Значения по умолчанию для каждого регистра можно найти в разделе 3.

SLEEP При значении 1 этот бит переводит MPU-60X0 в режим ожидания.

CYCLE Когда этот бит установлен в 1 и SLEEP отключен, MPU-60X0 будет циклически переключаться между режимом ожидания и пробуждением, чтобы взять одну выборку данных от активных датчиков со скоростью, определенной LP_WAKE_CTRL (регистр 108).

TEMP_DIS При значении 1 этот бит отключает датчик температуры.

CLKSEL 3-битное значение без знака. Определяет источник синхронизации устройства.

Примечание: При использовании интерфейса SPI пользователь должен использовать **DEVICE_RESET** (регистр 107), а также **SIGNAL_PATH_RESET** (регистр 104), чтобы гарантировать, что сброс выполняется правильно. Используемая последовательность должна быть:

1. Установите **DEVICE_RESET** = 1 (зарегистрируйте **PWR_MGMT_1**)
2. Подождите 100 мс
3. Установите **GYRO_RESET** = **ACCEL_RESET** = **TEMP_RESET** = 1 (зарегистрируйте **SIGNAL_PATH_RESET**)
4. Подождите 100 мс

4.29 Регистр 108 - Управление питанием 2

PWR_MGMT_2

Тип: чтение / запись

Register (Hex)	Register (Decimal)	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
6C	108	LP_WAKE_CTRL[1:0]		STBY_XA	STBY_YA	STBY_ZA	STBY_XG	STBY_YG	STBY_ZG

Описание:

Этот регистр позволяет пользователю настраивать частоту пробуждений в режиме акселерометра только с низким энергопотреблением. Этот регистр также позволяет пользователю переводить отдельные оси акселерометра и гироскопа в режим ожидания.

MPU-60X0 можно перевести в режим акселерометра только с низким энергопотреблением, выполнив следующие действия:

- (i) Установите бит **CYCLE** на 1
- (ii) Установите бит **SLEEP** в 0
- (iii) Установите бит **TEMP_DIS** в 1
- (iv) Установите биты **STBY_XG**, **STBY_YG**, **STBY_ZG** в 1

Все вышеперечисленные биты можно найти в регистре управления питанием 1 (регистр 107).

В этом режиме устройство отключит все устройства, за исключением основного интерфейса I2C, включив только акселерометр с фиксированными интервалами, чтобы выполнить одно измерение. Частота пробуждений может быть настроена с помощью **LP_WAKE_CTRL**, как показано ниже.

LP_WAKE_CTRL	Wake-up Frequency
0	1.25 Hz
1	5 Hz
2	20 Hz
3	40 Hz

Для получения дополнительной информации о режимах питания MPU-6050, пожалуйста, обратитесь к Регистру 107.

Пользователь может перевести отдельные оси акселерометра и гироскопа в режим ожидания с помощью этого регистра. Если устройство использует ось гироскопа в ка-

Register (Hex)	Register (Decimal)	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
74	116	FIFO DATA[7:0]							

Описание:

Этот регистр используется для чтения и записи данных из буфера FIFO.

Данные записываются в FIFO в порядке регистрационного номера (от самого низкого до самого высокого). Если все флаги включения FIFO (см. Ниже) включены и все регистры данных внешнего датчика (регистры с 73 по 96) связаны с ведомым устройством, содержимое регистров с 59 по 96 будет записано в порядке с частотой дискретизации.

Содержимое регистров данных датчика (регистры 59–96) записываются в буфер FIFO, когда их соответствующие флаги включения FIFO установлены в 1 в FIFO_EN (регистр 35). Дополнительный флаг для регистров данных датчика, связанных с ведомым устройством I2C, можно найти в I2C_MST_CTRL (регистр 36).

Если буфер FIFO переполнен, бит состояния FIFO_OFLOW_INT автоматически устанавливается на 1. Этот бит находится в INT_STATUS (регистр 58). Если буфер FIFO переполнен, самые старые данные будут потеряны, а новые данные будут записаны в FIFO.

Если буфер FIFO пуст, чтение этого регистра вернет последний байт, который был ранее считан из FIFO, до тех пор, пока не будут доступны новые данные. Пользователь должен проверить FIFO_COUNT, чтобы убедиться, что буфер FIFO не читается, когда пустой.

Параметры:

FIFO_DATA 8-битные данные, передаваемые в буфер FIFO и из него.

4.32 Регистр 117 - Кто я

WHO_AM_I

Тип: только чтение

Register (Hex)	Register (Decimal)	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
75	117	-	WHO_AM_I[6:1]						-

Описание:

Этот регистр используется для проверки личности устройства. Содержимое WHO_AM_I - это старшие 6 бит 7-битного адреса I2C MPU-60X0. Младший значащий бит адреса I2C MPU-60X0 определяется значением контакта AD0. Значение контакта AD0 не отражается в этом регистре.

Значение по умолчанию для регистра - 0x68.

Биты 0 и 7 зарезервированы. (Жестко запрограммирован на 0)

Параметры:

WHO_AM_I Содержит 6-битный адрес I2C MPU-60X0.

Значение Power-On-Reset для Бита 6: Бит 1 составляет 110 100.

Эта информация, предоставленная InvenSense, считается точной и надежной. Тем не менее, InvenSense не несет никакой ответственности за его использование или за любые нарушения патентов или других прав третьих лиц, которые могут возникнуть в результате его использования. Технические характеристики могут быть изменены без предварительного уведомления. InvenSense оставляет за собой право вносить изменения в этот продукт, включая его схемы и программное обеспечение, с целью улучшения его конструкции и / или производительности без предварительного уведомления. InvenSense не дает никаких гарантий, ни явных, ни подразумеваемых, в отношении информации и спецификаций, содержащихся в этом документе. InvenSense не несет никакой ответственности за какие-либо претензии или убытки, возникающие в связи с информацией, содержащейся в этом документе, или с использованием продуктов и услуг, подробно описанных в нем. Это включает в себя, но не ограничивается ими, претензии или убытки, основанные на нарушении патентов, авторских прав, масок и / или других прав интеллектуальной собственности.

Определенная интеллектуальная собственность, принадлежащая InvenSense и описанная в этом документе, защищена патентом. Никакая лицензия не предоставляется косвенно или иным образом в соответствии с каким-либо патентом или патентными правами InvenSense. Эта публикация заменяет и заменяет всю ранее предоставленную информацию. Торговые марки, которые являются зарегистрированными торговыми марками, являются собственностью соответствующих компаний. Датчики InvenSense не должны использоваться или продаваться при разработке, хранении, производстве или использовании любого обычного оружия или оружия массового поражения или для любого другого оружия или приложений, угрожающих жизни, а также в любых других критически важных для жизни приложениях, таких как медицинское оборудование, транспортировка, аэрокосмические и ядерные приборы, подводное оборудование, оборудование для электростанций, оборудование для предотвращения бедствий и предупреждения преступности.

InvenSense® является зарегистрированным товарным знаком InvenSense, Inc. MPU™, MPU-6000™, MPU-6050™, MPU-60X0™, Digital Motion Processor™, DMP™, Motion Processing Unit™, MotionFusion™ и MotionApps™ являются товарными знаками InvenSense, Inc ,