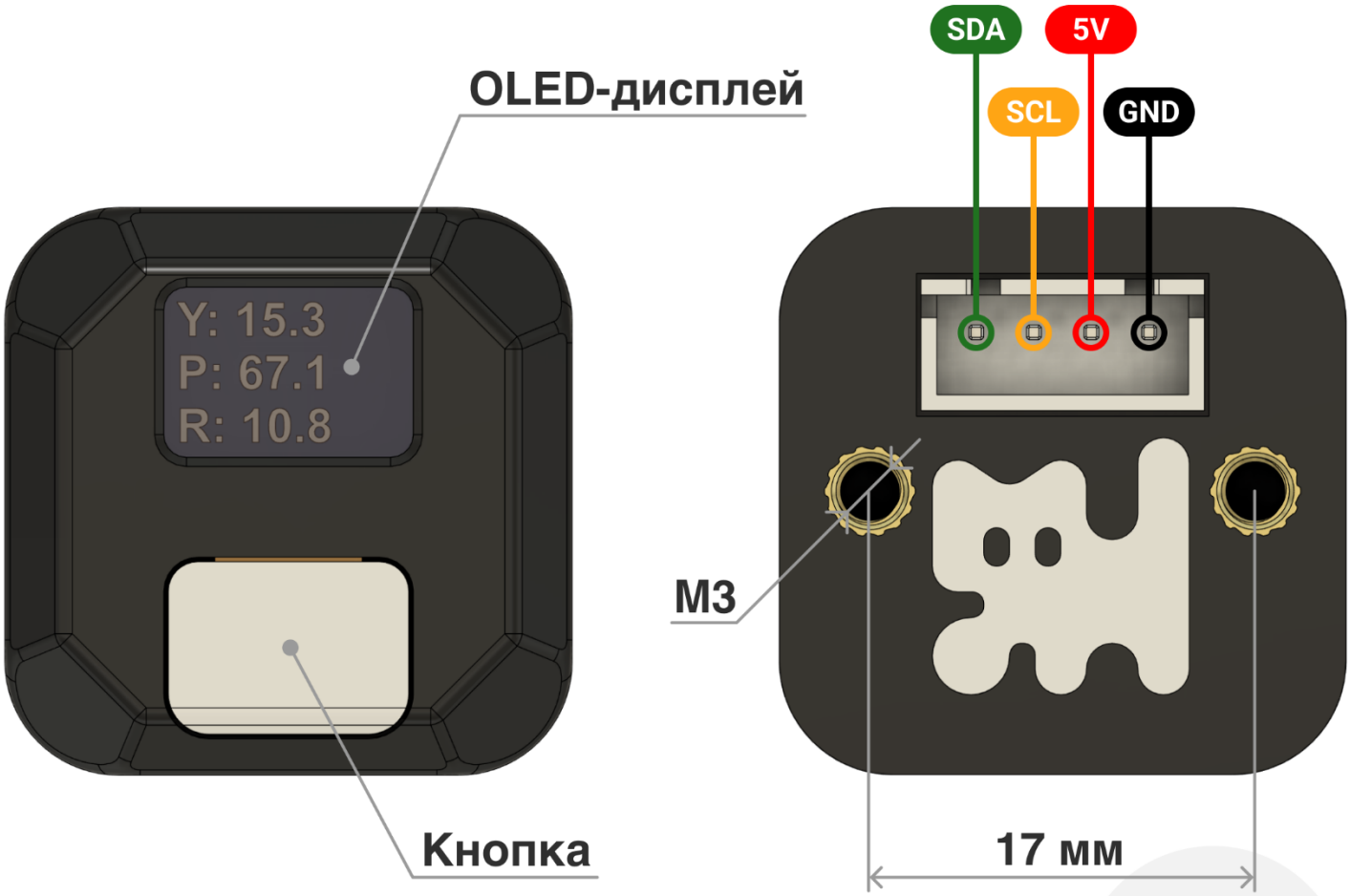


Датчик движения Баригадам — это интеллектуальный I²C-модуль движения для робототехники, обучения и встраиваемых проектов, где важны стабильные данные ориентации и простая интеграция. Внутри 6-осевое измерительное ядро работает совместно с выделенным встроенным контроллером и интегрированным OLED-дисплеем. Датчик измеряет ускорение и угловую скорость, рассчитывает yaw, pitch и roll внутри модуля и выдаёт как обработанные углы, так и сырые значения датчика через небольшую карту регистров I²C. Подключение простое: всего четыре провода (GND, VCC, SCL, SDA) к вашему контроллеру — Arduino, ESP32, платы на базе STM, Raspberry Pi и многим другим.

Модуль работает как готовый к использованию интеллектуальный датчик: хост-контроллеру не нужно запускать IMU-фильтры, вычислять смещения или работать с низкоуровневыми регистрами motion-чипа. Датчик движения постоянно обновляет внутреннюю оценку ориентации и в стандартном режиме показывает текущие значения yaw, pitch и roll на OLED-экране. Адрес I²C и режим работы настраиваются и сохраняются внутри устройства. Для начинающих прошивка настроена так, что простое чтение первых 6 байт по I²C возвращает yaw, pitch и roll в компактном знаковом 16-битном формате.

Разработано ***Баригадам***

© 2026, ООО «БАРИГАДАМ». Все права защищены.

Содержание

1. Обзор 2
2. Быстрый старт 2
   2.1. Что понадобится 2
   2.2. Питание и логические уровни 2
   2.3. Типичное подключение 3
   2.4. Настройки по умолчанию (из коробки)
3. Интерфейс I²C 4
   3.1. Адрес 4
   3.2. Протокол доступа к регистрам 4
4. Карта регистров 5
5. Формат вывода углов 6
   5.1. Значения Yaw / Pitch / Roll 6
   5.2. Сырые значения акселерометра и гироскопа 6
6. Режимы работы 6
   6.1. Сводка 6
   6.2. Подробности 7
7. Команды: адрес, режим, сброс нуля и калибровка 7
   7.1. Изменение адреса I²C 7
   7.2. Изменение режима 8
8. Чтение данных в коде 9
   8.1. Стандартное чтение по регистрам 9
   8.2. Простое чтение первых N байт 9
9. Поведение кнопки и OLED-дисплея 10
10. Рекомендации по использованию 10
11. Библиотека Arduino: Barigadam_MotionSensor 10
12. Важная информация и уведомление 13

© 2026, ООО «БАРИГАДАМ». Все права защищены.

1. Обзор

Датчик движения Баригадам — это интеллектуальный модуль движения со следующими возможностями:

- ✓ Обработанный вывод yaw, pitch и roll внутри датчика
- ✓ Сырые данные акселерометра: AX, AY, AZ
- ✓ Сырые данные гироскопа: GX, GY, GZ
- ✓ Встроенный OLED-дисплей для текущих значений и служебных сообщений
- ✓ Физическая кнопка для сброса нуля, вывода информации об адресе/режиме и калибровки
- ✓ Интерфейс I²C с небольшой картой регистров (22 регистра)
- ✓ Настраиваемый адрес I²C и режим работы, сохраняемые во flash-памяти и сохраняющиеся после отключения питания

Он разработан так, чтобы быть **простым в использовании даже для начинающих**. Вам потребуется только:

- Микроконтроллер с I²C (Arduino, STM32, ESP32, Raspberry Pi и т. д.)
- 4 провода: GND, VCC, SCL, SDA

2. Быстрый старт

2.1. Что понадобится

- Модуль датчика движения Баригадам
- Любая плата контроллера с I²C:
  - Ардуино Уно / Мега / Нано
  - ESP32 / ESP8266
  - Платы STM32
  - Raspberry Pi и т. д.
- 4 соединительных провода (мама-папа или папа-папа, в зависимости от вашей платы)

2.2. Питание и логические уровни

- **Напряжение питания: рекомендуется 5 В.**
- **Линии I²C (SDA, SCL):** логический уровень соответствует напряжению питания датчика.
  • При питании 5 В → логика I²C 5 В.
  • При питании 3,3 В → логика I²C 3,3 В.
- **Скорость I²C: работает как на 100 кГц, так и на 400 кГц.**
- **Встроена защита от переполюсовки, предотвращающая повреждение при неправильном подключении питания.**

© 2026, ООО «БАРИГАДАМ». Все права защищены.

2.3. Типичное подключение

Типичный 4-контактный разъём (точный порядок контактов смотрите на маркировке вашей платы):

Пин датчика	Подключение к плате микроконтроллера
GND	GND
VCC	5V или 3.3V, если поддерживается вашей версией
SCL	Пин I²C SCL
SDA	Пин I²C SDA

Примеры:

- **Ардуино Уно**
  - SDA → A4
  - SCL → A5
- **Ардуино Мега**
  - SDA → D20
  - SCL → D21
- **ESP32**
  - Любые GPIO, совместимые с I²C и настроенные в программном обеспечении.

2.4. Настройки по умолчанию (из коробки)

- **7-битный адрес I²C: 0x40**
- **Режим: 1 (обычный режим отображения с текущими значениями yaw, pitch и roll на OLED)**
- **Формат вывода углов: int16 = градусы × 10**
- **Частота опроса встроенного измерительного чипа: 416 Гц для акселерометра / 416 Гц для гироскопа в текущем семействе прошивок**
- **Размер карты регистров: 22 байта (0x00–0x15)**

Сразу после включения питания вы можете:

1. Просканировать шину I²C → устройство должно обнаружиться по адресу 0x40.
2. Считать первые 6 байт → получите yaw, pitch и roll.
3. При необходимости считать дополнительные байты для получения сырых данных акселерометра, гироскопа, режима, статуса, ID устройства и версии прошивки.

© 2026, ООО «БАРИГАДАМ». Все права защищены.

3. Интерфейс I²C

3.1. Адрес

- Датчик использует 7-битные адреса I²C.
- Заводской адрес по умолчанию: 0x40
- Адрес можно изменить на любое значение из диапазона 0x40–0x60 (33 возможных 7-битных адреса).

В Arduino Wire всегда используется 7-битный адрес напрямую (например, `Wire.beginTransmission(0x40);`).

3.2. Протокол доступа к регистрам

Датчик работает как небольшое I²C-устройство на основе регистров.

Чтобы считать регистры:

1. **Установите указатель регистра**
   - Начните запись по адресу устройства
   - Отправьте 1 байт: адрес регистра (0x00–0x15)
   - Завершите передачу (или выполните повторный старт)
2. **Считайте данные**
   - Начните чтение с того же адреса устройства
   - Считайте 1 или несколько байт
   - Внутренний указатель регистра автоматически увеличивается после каждого байта
   - После последнего регистра (0x15) указатель переходит обратно к 0x00

Это позволяет считать все измерительные данные за один приём.

Пример шаблона (словесно):

- Запись: [DeviceAddr + W] [0x00] → устанавливает указатель на первый регистр
- Чтение 6 байт: [YAW_L, YAW_H, PITCH_L, PITCH_H, ROLL_L, ROLL_H]

Важно: обычная запись в регистры используется для установки указателя регистра. Однобайтные командные записи используются для адреса, режима, сброса нуля и калибровки, как описано в разделе 7.

© 2026, ООО «БАРИГАДАМ». Все права защищены.

4. Карта регистров

Датчик движения Баригадам предоставляет 22 регистра (0x00–0x15):

Адрес	Имя	Тип	Описание
0x00	YAW_L	int16 LSB	Угол yaw, младший байт
0x01	YAW_H	int16 MSB	Угол yaw, старший байт
0x02	PITCH_L	int16 LSB	Угол pitch, младший байт
0x03	PITCH_H	int16 MSB	Угол pitch, старший байт
0x04	ROLL_L	int16 LSB	Угол roll, младший байт
0x05	ROLL_H	int16 MSB	Угол roll, старший байт
0x06	AX_L	int16 LSB	Акселерометр X, младший байт
0x07	AX_H	int16 MSB	Акселерометр X, старший байт
0x08	AY_L	int16 LSB	Акселерометр Y, младший байт
0x09	AY_H	int16 MSB	Акселерометр Y, старший байт
0x0A	AZ_L	int16 LSB	Акселерометр Z, младший байт
0x0B	AZ_H	int16 MSB	Акселерометр Z, старший байт
0x0C	GX_L	int16 LSB	Гироскоп X, младший байт
0x0D	GX_H	int16 MSB	Гироскоп X, старший байт
0x0E	GY_L	int16 LSB	Гироскоп Y, младший байт
0x0F	GY_H	int16 MSB	Гироскоп Y, старший байт
0x10	GZ_L	int16 LSB	Гироскоп Z, младший байт
0x11	GZ_H	int16 MSB	Гироскоп Z, старший байт
0x12	MODE	uint8_t	Текущий режим работы
0x13	STATUS	uint8_t	Статус / байт сигнатуры, 0xAC в текущей прошивке
0x14	DEV_ID	uint8_t	ID устройства, 0xBA для семейства смарт-датчиков Баригадам
0x15	FW	uint8_t	Версия прошивки, 0x01 в этом семействе прошивок

- Все значения int16 имеют формат little-endian: value = LOW + (HIGH << 8).
- Yaw, pitch и roll являются знаковыми значениями в градусах × 10.

© 2026, ООО «БАРИГАДАМ». Все права защищены.

- Сырые значения акселерометра и гироскопа являются знаковыми LSB-значениями из измерительного ядра движения.
- STATUS, DEV_ID и FW можно использовать для проверки наличия датчика и совместимости с вашей библиотекой.

5. Формат вывода углов

5.1. Значения Yaw / Pitch / Roll

Yaw, pitch и roll возвращаются как знаковые 16-битные целые числа. Единица измерения — градусы × 10.

Пример значения регистра	Что означает
0	0.0°
900	90.0°
-450	-45.0°
1800	180.0°

Для перевода значения в градусы: float angle_deg = raw_value / 10.0f;

5.2. Сырые значения акселерометра и гироскопа

Модуль также предоставляет сырые значения акселерометра и гироскопа. Они полезны для продвинутых пользователей, которые хотят реализовать собственные фильтры, диагностику или алгоритмы обработки движения на хост-контроллере.

- AX, AY, AZ — сырые показания акселерометра.
- GX, GY, GZ — сырые показания гироскопа.
- Для большинства робототехнических задач рекомендуется использовать обработанные значения yaw / pitch / roll.

6. Режимы работы

Датчик движения Баригадам поддерживает режимы отображения и взаимодействия. Текущий режим хранится внутри датчика и может быть считан из регистра MODE (0x12).

6.1. Сводка

Режим	Имя	Поведение OLED	Кнопка	Типичное применение
0	Экран выключен	OLED выключен	Отключена	Работа без вывода на дисплей и без визуальной индикации
1	Обычный	Показывает текущие yaw / pitch / roll	Включена	Рекомендуемый режим по умолчанию для большинства пользователей

© 2026, ООО «БАРИГАДАМ». Все права защищены.

2	Кнопка отключена	Показывает текущие yaw / pitch / roll	Отключена	Когда случайные нажатия кнопки должны игнорироваться

6.2. Подробности

Режим 0 — экран выключен

- OLED-дисплей выключен.
- Датчик продолжает обновлять измерительные регистры I²C.
- Используется, когда нужно уменьшить визуальное отвлечение или полностью отключить индикацию на устройстве.

Режим 1 — обычный режим

- Стандартный рекомендуемый режим.
- OLED-дисплей показывает текущие значения yaw, pitch и roll.
- Физическая кнопка используется для сброса нуля, вывода информации об адресе/режиме и калибровки.

Режим 2 — кнопка отключена

- OLED-дисплей продолжает показывать текущие значения.
- Действия кнопки игнорируются.
- Полезно для соревновательных роботов, где случайное нажатие кнопки не должно изменять состояние датчика.

7. Изменение адреса I²C и режима

Датчик движения Баригадам позволяет изменять адрес I²C, менять режим работы, сбрасывать текущие углы в ноль и запускать калибровку гироскопа. Адрес и режим сохраняются во внутренней flash-памяти и остаются после отключения питания. Сброс нуля является рабочим смещением и при необходимости должен выполняться повторно после включения питания.

7.1. Изменение адреса I²C

- Текущий/адрес по умолчанию: A_old (например, 0x40).
- Желаемый новый адрес: A_new (в диапазоне 0x40–0x60).

Последовательность:

1. Подайте питание на датчик.
2. Выполните I²C-запись на адрес A_old
   с ровно одним байтом данных: A_new.

© 2026, ООО «БАРИГАДАМ». Все права защищены.

Если команда выполнена успешно:

- Датчик сохраняет A_new во flash-память,
- если дисплей включён, OLED показывает сообщение об изменении адреса,
- начиная со следующего обмена (и при всех последующих включениях), датчик будет использовать адрес A_new.

Если получен недопустимый байт, датчик отклоняет его и показывает короткое сообщение на дисплее.

7.2. Изменение режима

- Командные байты режима: 0x20–0x22.
- Номер режима = командный байт − 0x20.

Командный байт	Новый режим
0x20	0
0x21	1
0x22	2

Последовательность:

1. Подайте питание на датчик.
2. Запишите один байт на текущий адрес I²C: 0x20 + mode.

Если команда выполнена успешно:

- Датчик сохраняет новый режим во flash-память,
- датчик продолжает работу в этом режиме и запоминает его для следующего включения.

Если байт не относится к допустимым командным байтам или адресам:

- 0x20–0x22 для режимов
- 0x23–0x25 для сброса и калибровки
- 0x40–0x60 для адреса I²C

датчик отклоняет его и показывает короткое сообщение на дисплее. Изменения не сохраняются.

Если записанный байт совпадает с текущим сохранённым адресом/режимом: ничего не меняется, запись во flash-память не выполняется, датчик просто продолжает нормальную работу.

© 2026, ООО «БАРИГАДАМ». Все права защищены.

7.3. Команды сброса и калибровки

Командный байт	Действие	Описание
0x23	Сброс текущих значений yaw / pitch / roll в ноль	Сохраняет текущую ориентацию как рабочую нулевую точку. Полная калибровка гироскопа не выполняется.
0x24	Быстрая калибровка гироскопа	Запускает короткую калибровку гироскопа, обычно N = 100 выборок, затем устанавливает текущую ориентацию в ноль.
0x25	Полная калибровка гироскопа	Запускает более длительную калибровку гироскопа, обычно N = 500 выборок, затем устанавливает текущую ориентацию в ноль.

Важные примечания по калибровке:

- Во время калибровки держите датчик полностью неподвижным.
- Используйте 0x24 для быстрой калибровки, когда время настройки ограничено.
- Используйте 0x25 для наиболее стабильного результата перед точным движением или точными поворотами.

8. Чтение данных в коде

В библиотеке или проекте данные с датчика можно считывать двумя способами.

8.1. Стандартное чтение по регистрам

Это классический шаблон доступа к регистрам I²C, совместимый с будущими изменениями прошивки.

1. **Выберите начальный регистр (например, 0x00 для YAW_L).**
2. **Считайте N байт.**
3. **Восстановите знаковые 16-битные значения из младшего и старшего байтов.**

Для yaw, pitch и roll считайте 6 байт:

```
YAW_L, YAW_H, PITCH_L, PITCH_H, ROLL_L, ROLL_H
```

Затем восстановите значения:

```
int16_t yaw10   = YAW_L   | (YAW_H   << 8);
int16_t pitch10 = PITCH_L | (PITCH_H << 8);
int16_t roll10  = ROLL_L  | (ROLL_H  << 8);
float yaw_deg   = yaw10   / 10.0f;
float pitch_deg = pitch10 / 10.0f;
float roll_deg  = roll10  / 10.0f;
```


© 2026, ООО «БАРИГАДАМ». Все права защищены.

8.2. Простое чтение первых N байт

Для удобства прошивка устроена так, что если указатель регистра не выбран, чтение по I²C начинается с регистра 0x00. Поэтому можно просто считать:

- 6 байт: yaw, pitch, roll
- 18 байт: yaw, pitch, roll + сырые данные акселерометра + сырые данные гироскопа
- 22 байта: полный набор регистров, включая MODE, STATUS, DEV_ID и FW

9. Поведение кнопки и OLED-дисплея

Физическая кнопка обеспечивает простые служебные действия без написания кода на хост-контроллере.

Действие кнопки	**Поведение**
Одно короткое нажатие	Сбрасывает текущие значения yaw, pitch и roll в ноль
Двойное короткое нажатие	Показывает текущий адрес I²C и режим на OLED
Долгое нажатие	Запускает полную калибровку гироскопа, затем устанавливает текущую ориентацию в ноль

В режиме 2 действия кнопки отключены. В режиме 0 OLED выключен, но вывод данных по I²C продолжается.

10. Рекомендации по использованию

- Для лучших результатов держите датчик неподвижным во время запуска и калибровки.
- Перед задачами, требующими точных поворотов, выполните полную калибровку.
- Выполните сброс нуля после установки робота в стартовое положение.
- Не устанавливайте датчик на гибкие или вибрирующие части робота.
- Считывайте значения yaw / pitch / roll достаточно часто для вашего контура управления, но избегайте лишнего I²C-трафика, если роботу не нужна высокая частота обновления.
- Для соревновательных роботов режим 2 может предотвратить случайную перекалибровку или сброс нуля из-за нажатия кнопки.

© 2026, ООО «БАРИГАДАМ». Все права защищены.

11. Библиотека Arduino: Barigadam_MotionSensor

Библиотека **Barigadam_MotionSensor** является необязательной вспомогательной библиотекой для
Arduino-совместимых плат. Она написана для Arduino API
(Arduino.h, Wire.h, Serial) и работает с:

- Классическими платами Ардуино (Уно, Мега, Нано и т. д.)
- Платами STM32 / ESP32, использующими ядро Arduino

Библиотека оборачивает протокол I²C, описанный в этом руководстве, в простые функции, поэтому вы можете считывать yaw, pitch и roll без ручной работы с регистрами.

Чтобы использовать библиотеку, подключите заголовочный файл в скетче:

```
#include <Wire.h>
#include <Barigadam_MotionSensor.h>
```

Перед использованием функций убедитесь, что в setup() вызваны Wire.begin() и Serial.begin(...).

Barigadam_readYPR_All (uint8_t addr, int16_t &yaw, int16_t &pitch, int16_t &roll)
Низкоуровневая вспомогательная функция, которая считывает первые 6 байт из регистров датчика и возвращает все три 16-битных значения углов:

- yaw — вращение вокруг вертикальной оси
- pitch — наклон вперёд / назад
- roll — наклон влево / вправо

Значения возвращаются как знаковые 16-битные целые числа в градусах × 10. Эта функция устанавливает указатель регистра на 0x00, затем считывает 6 байт, поэтому следует стандартному шаблону чтения по регистрам, описанному в разделе 8.1. Она полезна, когда нужно считать все три угла за одну транзакцию.

int GetYPRi(uint8_t addr, uint8_t axis)
Считывает yaw, pitch или roll как целое значение в градусах × 10.

- axis = 1 → yaw
- axis = 2 → pitch
- axis = 3 → roll
- Любое другое значение → возвращает 0

Например:

- 900 означает 90.0°
- -450 означает -45.0°
- 0 означает 0.0°

float GetYPR(uint8_t addr, uint8_t axis)

© 2026, ООО «БАРИГАДАМ». Все права защищены.

Считывает yaw, pitch или roll как значение с плавающей точкой в градусах.

- axis = 1 → yaw
- axis = 2 → pitch
- axis = 3 → roll
- Любое другое значение → возвращает 0.0

uint8_t readRegister (uint8_t addr, uint8_t reg)
Универсальная вспомогательная функция для чтения одного 8-битного регистра:

- Устанавливает внутренний указатель регистра на reg
- Считывает один байт и возвращает его

Используйте эту функцию, если нужен прямой доступ к MODE, STATUS, DEV_ID, FW или любому другому регистру, описанному в карте регистров.

bool sendCommand(uint8_t addr, uint8_t commandByte)
Низкоуровневая вспомогательная функция, которая отправляет один командный байт напрямую в датчик. Используйте эту функцию, когда требуется отправить точный командный байт, например:

- 0x23 — сброс нуля
- 0x24 — быстрая калибровка гироскопа
- 0x25 — полная калибровка гироскопа.

Функция возвращает true, если I²C-транзакция выполнена успешно, и false при ошибке I²C.

static void sendToSensor(uint8_t currentAddr, uint16_t dataToSend)
Вспомогательная функция для изменения **адреса I²C** или **режима** (см. раздел 7):

- Если dataToSend находится в диапазоне 0…2
  → он интерпретируется как номер режима.
  Функция автоматически добавляет 0x20 и отправляет соответствующий командный байт (0x20…0x22).
- Если dataToSend находится в диапазоне 0x40…0x60
  → он интерпретируется как новый адрес I²C.
- Если dataToSend равен 0x23 → команда сброса нуля.
- Если dataToSend равен 0x24 → команда быстрой калибровки гироскопа.
- Если dataToSend равен 0x25 → команда полной калибровки гироскопа.

Функция выводит диагностические сообщения в Serial и многократно отправляет выбранный байт на currentAddr.

Типичный порядок использования:

1. Загрузите скетч, в котором sendToSensor(...) вызывается в setup().
2. Откройте Serial Monitor.

© 2026, ООО «БАРИГАДАМ». Все права защищены.

3. Перезапустите питание датчика, чтобы он получил команду во время окна загрузки.

Если параметры недопустимы (адрес вне диапазона, совпадает с текущим и т. д.), функция выводит сообщение об ошибке и ничего не делает.

void I2Cscan()
Простой сканер шины I²C для отладки:

- Сканирует все 7-битные адреса от 0x01 до 0x7E
- Выводит все обнаруженные устройства в Serial Monitor
- Повторяет сканирование каждые 5 секунд в бесконечном цикле

Это полезно, чтобы подтвердить, что датчик движения Баригадам присутствует по ожидаемому адресу (0x40 по умолчанию или любой пользовательский адрес в диапазоне 0x40–0x60).

Эти функции можно свободно сочетать в ваших скетчах Arduino:
например, сначала использовать I2Cscan() для поиска устройства, sendToSensor() для настройки адреса и режима, а затем GetYPR() для получения значений yaw, pitch и roll.

12. Важная информация и уведомление

Информация в этом документе отражает текущие знания и понимание компании Баригадам на дату публикации. Она частично основана на данных, полученных от производителей компонентов и других третьих сторон. Баригадам не гарантирует, что такая информация от третьих сторон является полной, точной или актуальной. Мы постоянно работаем над улучшением нашей документации, однако можем не проводить разрушающие испытания, полный химический анализ или другие исчерпывающие проверки для каждой партии поступающих материалов.

Часть информации считается конфиденциальной для Баригадам и её поставщиков; поэтому некоторые идентификаторы (например, подробные химические данные или внутренние номера деталей) могут не раскрываться.

© 2026, ООО «БАРИГАДАМ». Все права защищены.

Уведомление

Баригадам оставляет за собой право в любое время и без предварительного уведомления изменять конструкцию, функции или технические характеристики любого продукта, описанного в данном документе, а также прекращать выпуск любого продукта. Перед размещением заказа или проектированием новых систем с использованием продукции Баригадам клиентам следует всегда получать у Баригадам самую актуальную информацию о продукте и документацию.

Баригадам не несёт ответственности за использование каких-либо схем, примеров или рекомендаций, содержащихся в данном документе, а также за проектирование или работу любых изделий клиента, в состав которых входят компоненты Баригадам. Продажа или использование продукции Баригадам не предоставляет никакой лицензии — ни явной, ни подразумеваемой — на какие-либо патенты, авторские права, права на топологии микросхем или иные права интеллектуальной собственности. Баригадам также не гарантирует, что описанные здесь схемы или конструкции свободны от претензий третьих лиц в отношении интеллектуальной собственности.

Продукция Баригадам предоставляется «как есть» без каких-либо явных или подразумеваемых гарантий, включая, помимо прочего, гарантии товарной пригодности, пригодности для конкретной цели или отсутствия нарушений прав третьих лиц. Баригадам не несёт ответственности за какие-либо прямые, косвенные, случайные, специальные или последующие убытки, возникающие в результате использования либо невозможности использования любой продукции Баригадам или документации.

Применения, критически важные для безопасности

ПРОДУКЦИЯ БАРИГАДАМ НЕ РАЗРАБАТЫВАЕТСЯ, НЕ ТЕСТИРУЕТСЯ И НЕ ПРЕДНАЗНАЧЕНА ДЛЯ ИСПОЛЬЗОВАНИЯ В ПРИЛОЖЕНИЯХ, ГДЕ ОТКАЗ ИЛИ НЕИСПРАВНОСТЬ МОГУТ РАЗУМНО ОЖИДАЕМО ПРИВЕСТИ К ТРАВМАМ, СМЕРТИ ИЛИ СЕРЬЁЗНОМУ УЩЕРБУ ИМУЩЕСТВУ. СЮДА ОТНОСЯТСЯ, В ЧАСТНОСТИ, СИСТЕМЫ ЖИЗНЕОБЕСПЕЧЕНИЯ, ИМПЛАНТИРУЕМЫЕ МЕДИЦИНСКИЕ УСТРОЙСТВА, АВАРИЙНОЕ ИЛИ КРИТИЧЕСКИ ВАЖНОЕ ДЛЯ БЕЗОПАСНОСТИ ОБОРУДОВАНИЕ, А ТАКЖЕ ЛЮБЫЕ ДРУГИЕ ПРИЛОЖЕНИЯ ТИПА «FAIL-SAFE».

ЛЮБОЕ ТАКОЕ ИСПОЛЬЗОВАНИЕ ПРОДУКЦИИ БАРИГАДАМ ОСУЩЕСТВЛЯЕТСЯ ИСКЛЮЧИТЕЛЬНО НА РИСК КЛИЕНТА, И КЛИЕНТ НЕСЁТ ЕДИНОЛИЧНУЮ ОТВЕТСТВЕННОСТЬ ЗА ВСЕ НЕОБХОДИМЫЕ МЕРЫ БЕЗОПАСНОСТИ, РЕЗЕРВИРОВАНИЯ И СОБЛЮДЕНИЯ ТРЕБОВАНИЙ.

© 2026, ООО «БАРИГАДАМ». Все права защищены.