Устройства автоматизации измерений RT-Rack.

Руководство по программированию

Ревизия 1.0.

1.1. ПО ДЛЯ САМОСТОЯТЕЛЬНОГО ПРОГРАММИРОВАНИЯ 2
1.1.1. Глоссарий *2*
1.1.2. Общие сведения *3*
1.1.3. Открытие сессии взаимодействия *4*
1.1.3.1. Последовательность вызова фукнций 4
1.1.3.2. Определение параметров UDP-передачи 4
1.1.3.3. Функция xqms_init() 4
1.1.3.4. Сканирование слотов 5
1.1.4. Конфигурация сеанса сбора *5*
1.1.4.1. Установка общей частоты дискретизации 5
1.1.4.2. Индивидуальные настройки модулей 5
1.1.5. Запущенный сеанс сбора *6*
1.1.5.1. На каждом такте частоты дискретизации: 6
1.1.5.2. При формировании полного кадра UDP: 7
1.1.5.3. Формат остчёта 7
1.1.6. Останов сеанса сбора *7*
1.1.7. Особенности модулей *7*
1.1.7.1 XQMS16 7
1.1.7.2 XQMS17 7
1.1.7.3. XQMS85 8
1.1.7.4 XQMS45 8
1.1.8. Работа с несколькими модулями *8*
Приложение «А» - ссылки на структуры (из xqms.h) *9*

Контакты:
http://www.R-Technology.ru
Info@R-Technology.ru - Общие вопросы
Sales@R-Technology.ru - Отдел продаж
Support@R-Technology.ru - Техническая поддержка

1.1. ПО для самостоятельного программирования

В комплект ПО для устройств серии RT-Rack входит пакет ПО для самостоятельного программирования (пакет SDK), включающий в себя библиотеку функций и примеры программирования. В настоящей главе приводится описание принципов программирования устройств серии RT-Rack.

1.1.1. Глоссарий

Контроллер – Модуль управления устройством RT-Rack, осуществляющий взаимодействие с целевыми модулями и верхним уровнем. Работает под управлением ОС Linux.

Целевой модуль – Источник данных с разрядностью до 18 бит в каждом отсчёте. Может иметь многоканальную организацию (до 32 входов).

Верхний уровень – Пользовательское приложение на ПК, использующее библиотеку xqms для управления устройством серии RT-Rack и приёма данных.

rtrack_srv – Приложение, выполняемое на центральном вычислителе контроллера, работающем под ОС Linux. Автоматически запускается при загрузке ОС, обеспечивает обмен данными по TCP/IP и UDP.

Сессия – Подключение к прибору через вызов функции xqms_init(). Одновременно может быть открыта только одна сессия.

Сеанс сбора – Период передачи данных по UDP, инициируемый вызовом xqms_start() и завершаемый xqms_stop().

Частота дискретизации – Частота отсчётов в Гц, задаваемая формулой: Fdscr = 48000000 / Fdiv, где Fdiv – целочисленный делитель.

Кадр (UDP frame) – Блок данных, формируемый контроллером для передачи по UDP. Содержит ((2 * N) + 4) * 64 + 10 байтов + UDP-заголовки. Где N число целевых модулей, установленных в крейте т.е. от 1 до 8-ми. В любом случае не превышает длину, подлежащую фрагментации.

Отсчёт (sample) – 32-битное слово с данными: биты 0-23 – измеренное значение в беззнаковом представлении, 24-28 – номер канала (0-31), 29-31 – номер слота (0-7).

Кольцевой буфер – Область памяти на стороне верхнего уровня, куда размещаются отсчёты из UDP-кадров. Размер указывается в выборках (RING_BUF_SAMPLES).

Номер слота – Индекс слота (0-7) в крейте, в котором установлен целевой модуль. Используется в функциях xqms_update(N), xqmcfg[N].

Коэффициент передачи – Обобщённый параметр, описывающий масштабирование сигнала (может включать усиление или аттенюацию). Задаётся индивидуально для каждого канала.

Номер канала – Индекс канала (0-31) внутри целевого модуля. Используется в структуре xqms_channel channel[n].

1.1.2. Общие сведения

Измерительная система на базе аппаратуры RT-Rack может состоять из одного модуля контроллера и от 1 до 8 целевых модулей.

Взаимодействие целевых модулей RT-Rack с верхним уровнем информационных систем осуществляется только посредством контроллера на базе протоколов:

- **TCP/IP** – для управления;
- **UDP** – для передачи данных измерений.

Интерфейс связи: **1000BASE-T**.

Примечание: Также возможно взаимодействие через внутренний порт TCP/IP с пользовательским приложением, запущенным непосредственно на вычислительном ядре контроллера под встроенной ОС Linux.

Целевые модули представляют собой источники данных с разрядностью до 18 бит в каждом отсчёте . Модули могут иметь многоканальную организацию (до 32 входов), при этом каждый следующий отсчёт производится со следующего канала по модулю количества активных каналов.

Взаимодействие контроллера с целевыми модулями осуществляется по специализированному интерфейсу.

Неотъемлемым компонентом комплекса является приложение rtrack_srv , выполняемое на центральном вычислителе контроллера на всех режимах функционирования (запускается автоматически при загрузке ОС контроллера).

Примечание: Приложение rtrack_srv не имеет параметров командной строки или файла конфигурации. Все настройки передаются с верхнего уровня по протоколу TCP/IP.

Режимы работы rtrack_srv

Режим	Вход	Выход	Описание
Standby	–	–	Ожидание подключения
Connect	`xqms_init()`	`xqms_close()`	Установлена сессия, конфигурация возможна
Start	`xqms_start()`	`xqms_stop()`	Активен сеанс сбора, передача данных по UDP

Управление комплексом и получение данных измерений в реальном времени осуществляется с использованием библиотеки функций под ОС Windows или Linux, из приложений пользователей или прилагаемого ПО «XQMS_OSC» (осциллограф/регистратор).

1.1.3. Открытие сессии взаимодействия

1.1.3.1. Последовательность вызова фукнций

```
1. xqms_init()           – открытие сессии, определение адреса контроллера
2. xqms_set_clock_divisor() – установка общей частоты дискретизации
3. xqms_update(N)        – настройка модуля в слоте N (для каждого занятого
слота)
4. xqms_start()          – запуск сбора данных и начало передачи по UDP
5. xqms_stop()           – останов сбора данных
6. xqms_close()          – закрытие сессии
```

Примечание: Одновременно может быть открыта только одна сессия.

1.1.3.2. Определение параметров UDP-передачи

На этапе вызова **xqms_init()** происходит только определение IP-адреса, куда будет направлен UDP-поток в дальнейшем.

- Принудительная установка порта и адреса получателя не предусмотрена – используется «обратный» адрес клиента.
- Передача данных по UDP инициируется только вызовом функции **xqms_start().**

1.1.3.3. Функция xqms_init()

```
void* xqms_init(unsigned int RING_BUF_SAMPLES, const char* controller_ip);
```

Параметр	Описание
RING_BUF_SAMPLES	Размер кольцевого буфера в отсчётах (не в байтах). Указание размера в выборках позволяет пользователю не знать размера выборок в байтах.
controller_ip	Сетевой адрес контроллера крейта (например, "192.168.0.100"). При работе через внутренний порт указывается "127.0.0.1", что означает адрес приложения rtrack_srv внутри контроллера.

Возвращаемое значение: указатель на кольцевой буфер данных в памяти компьютера-клиента.

Важно: Даже в случае вызова xqms_init() на самом контроллере, IP 127.0.0.1 относится к приложению **rtrack_srv**, а ссылку на буфер получает пользовательское приложение, которое полностью изолировано от rtrack_srv (кроме порта TCP/IP, по которому осуществляется управление и обмен данными).

1.1.3.4. Сканирование слотов

При открытии сессии производится опознавательное сканирование всех 8 слотов крейта. Создаётся список установленных модулей для непустых слотов с указанием:

- названия модуля;
- версии прошивки;
- серийного номера;
- номера слота.

1.1.4. Конфигурация сеанса сбора

1.1.4.1. Установка общей частоты дискретизации

Используется функция

```
xqms_set_clock_divisor(unsigned int Fdiv);
```

Частота отсчётов в Гц рассчитывается по формуле:

```
Fdscr = 48000000 / Fdiv
```

Например: Для модулей серии XQMS15 максимальная частота дискретизации составляет 500 кГц (делитель Fdiv = 96).

1.1.4.2. Индивидуальные настройки модулей

Настройки применяются к каждому модулю через функцию **xqms_update(N),** где N – номер слота (0-7).

Параметры конфигурации:

Параметр	Описание	Диапазон / Примечание
`n_channels`	Число активных каналов	1, 2, 4, 8, 16, 32 (но не более `max_channels` для данного типа модуля)
`channel[n].range`	Коэффициент передачи	Индивидуально для каждого канала; может означать усиление или аттенюацию

Параметр	Описание	Диапазон / Примечание
`channel[n].misc`	Специфические настройки модуля	Зависит от типа модуля (см. раздел Особенности модулей)

Примечание:

- Тип входа (дифференциальный/однопроводной) настраивается автоматически: при **n_channels == 32** устанавливается «однопроводной», иначе – «дифференциальный».
- Изменение числа активных каналов возможно только в режиме «стоп» (**xqms_stop()**). Остальные настройки можно изменять и при запущенном сборе, но это может вызвать кратковременный сбой данных.

Структура конфигурации

Перед вызовом **xqms_update(N)** необходимо заполнить соответствующие поля структуры **xqmcfg[N]:**

```
// Пример заполнения для слота N
xqmcfg[N].n_channels = 4;  // число активных каналов

// Настройка каждого канала
for (int ch = 0; ch < xqmcfg[N].n_channels; ch++) {
    xqmcfg[N].channel[ch].range = 3;  // код усиления
    xqmcfg[N].channel[ch].misc = 0;   // специфический параметр
}

// Применение настроек
xqms_update(N);
```

1.1.5. Запущенный сеанс сбора

После вызова функции **xqms_start()** происходит:

1.1.5.1. На каждом такте частоты дискретизации:

- **Приём данных**

Одновременный приём текущих отсчётов **с одного канала** со всех установленных модулей и размещение их в памяти контроллера через механизм DMA.

- **Рассылка настроек**

Одновременная передача в каждый слот информации об индивидуальных настройках, в соответствии с текущим номером канала, для многоканальных модулей.

1.1.5.2. При формировании полного кадра UDP:

Шаг	Описание
1	Контроллер отправляет буфер через rtrack_srv на верхний уровень
2	Пользовательское приложение принимает кадр, разбирает его и размещает отдельные отсчёты с тегами слота/канала в кольцевом буфере
3	Сдвиг указателя кольцевого буфера на длину принятых данных. Текущее значение указателя доступно через `xqms_get_buffer_ptr()`
4	Данные из пустых слотов отсеиваются rtrack_srv и не попадают в UDP-поток

1.1.5.3. Формат остчёта

Каждый отсчёт представлен 32-битным словом:

Биты	Назначение	Диапазон
0-23	Измеренное значение (прямой код)	$0 – 2^{24}-1$
24-28	Номер канала	0 – 31
29-31	Номер слота	0 – 7

1.1.6. Останов сеанса сбора

Функция **xqms_stop()** останавливает передачу данных по UDP.

Важно: После останова и последующего перезапуска с помощью **xqms_start()** необходимо применить **xqms_get_buffer_ptr()** для определения текущего начала кольцевого буфера. Остальные параметры сеанса сбора остаются прежними.

1.1.7. Особенности модулей

1.1.7.1 XQMS16

Биты 0..3 в поле xqmcfg[N].channel[0].misc используется для управления питанием датчиков соответствующих каналов (0..3):

1 - питание включено, 0 - выключено.

Поля xqmcfg[N].channel[ch].misc игнорируются для всех каналов, кроме нулевого.

1.1.7.2 XQMS17

При изменении диапазона измерений в любом канале (xqmcfg[N].channel[ch].range) этот диапазон устанавливается для всех каналов. Это связано с особенностями аппаратной реализации модуля.

1.1.7.3. XQMS85

Модуль xqms85 имеет встроенную схему автоматической балансировки нуля тензо-датчиков с помощью цифровых потенциометров, установленных на плате модуля.

Для автобалансировки используется специальная функция xqms85_autobalance(). Она производит параллельную автобалансировку нуля входных каналов и корректировку калибровочных коэффициентов входных каналов всех модулей типа xqms85 в крейте. Применяется только в режиме "стоп".

Поле xqmcfg[N].channel[ch].misc используется для установки потенциометров балансировки нуля. Допустимые значения 5 - 250.

1.1.7.4 XQMS45

Программирование выходного напряжения канала XQMS45 осуществляеется записью 16-битного кода ЦАП в поле xqmcfg[N].channel[ch].user_data и последующим вызовом функции xqms_update(N).

Для напряжения U (float или double) код ЦАП вычисляется по формуле

(uint16_t)((U - xqmcfg[N].coeffs[ch].offset) / xqmcfg[N].coeffs[ch].scale)

Текущие коды ЦАП передаются в общем потоке данных, как для обычного АЦП. Появление нового значения в потоке данных гарантирует установление нового напряжения на выходе модуля.

1.1.8. Работа с несколькими модулями

Важно: Работа системы не зависит от количества установленных модулей: при 1 или 8 модулях в память контроллера за единицу времени поступает одинаковое количество слов данных. Данные из пустых слотов записываются в аппаратный буфер, но отсеиваются rtrack_srv и не включаются в UDP-кадры.

Все модули из всех слотов одновременно пишут в память контроллера данные с одного из своих каналов через механизм DMA.

Приложение «A» - ссылки на структуры (из xqms.h)

```
// Максимальные значения
#define XQMS_MAX_MODULES    8    // максимальное число слотов
#define XQMS_MAX_CHANNELS   32   // максимальное число каналов в модуле

// Структура описания канала
struct xqms_channel {
    unsigned char range;  // код коэффициента передачи / усиления
    unsigned char misc;   // специфический параметр модуля
    // ... другие поля
};

// Структура конфигурации модуля в слоте N
struct {
    char n_channels;                          // число активных каналов
    struct xqms_channel channel[XQMS_MAX_CHANNELS];  // настройки каналов
    // ... другие поля
} xqmcfg[XQMS_MAX_MODULES];

// Структура информации о модуле
struct xqms_module {
    char name[32];        // название модуля
    char version[16];     // версия прошивки
    char serial[32];      // серийный номер
    char max_channels;    // максимальное число каналов для данного типа
    // ... калибровочные коэффициенты и другие поля
};
```