

Модуль счета импульсов. Принцип работы и протокол передачи данных по Lora-WAN.

Модуль счета импульсов (далее просто модуль) предназначен для считывания данных с импульсных входов, хранения считанных данных, и передачи накопленных данных на верхний уровень с использованием технологии LoraWAN.

1. Принцип функционирования.

С момента начала функционирования в модуле действует расписание LoraWAN.

Расписание LoraWan.

1. При запуске модуль находится в незарегистрированном режиме.

2. В незарегистрированном режиме начинает действовать расписание регистрации.

3. Расписание регистрации может быть настроено на периодичность от 1 до 3 раз в сутки:

- при периодичности 1 модуль будет пытаться зарегистрироваться один раз в день в интервале времени 00:00 до 12:00;

- при периодичности 2 модуль будет пытаться зарегистрироваться два раза в день в интервалах времени [00:00 до 06:00, 12:00 до 18:00];

- при периодичности 3 модуль будет пытаться зарегистрироваться три раза в день в интервалах времени [00:00 до 04:00, 08:00 до 12:00, 16:00 до 20:00].

4. Если регистрация прошла успешно, начинает действовать расписание передачи данных.

5. Расписание передачи данных может быть настроено на периодичность [1, 2, 3, 4, 6] раз в сутки:

- при периодичности 1 модуль будет пытаться передать данные один раз в день в интервале времени 00:00 до 12:00;

- при периодичности 2 модуль будет пытаться передать данные два раза в день в интервалах времени [00:00 до 06:00, 12:00 до 18:00];

- при периодичности 3 модуль будет пытаться передать данные три раза в день в интервалах времени [00:00 до 04:00, 08:00 до 12:00, 16:00 до 20:00].

- при периодичности 4 модуль будет пытаться передать данные четыре раза в день в интервалах времени [00:00 до 03:00, 06:00 до 09:00, 12:00 до 15:00, 18:00 до 21:00].

- при периодичности 6 модуль будет пытаться передать данные шесть раз в день в интервалах времени [00:00 до 02:00, 04:00 до 06:00, 08:00 до 10:00, 12:00 до 14:00, 16:00 до 18:00, 20:00 до 22:00].

6. Если модуль 5 раз подряд не смог передать данные, то происходит разрегистрация модуля и переход к шагу 2.

В заводских настройках по умолчанию расписание регистрации настроено на периодичность 1, а расписание передачи данных настроено на периодичность 2.

2. Данные которые хранятся в модуле.

2.1. Текущие данные.

Регистры импульсов 1 всего и по четырем тарифам — целое число в диапазоне от 0 до 4294967295 соответствующее количеству импульсов насчитанному импульсным входом 1. Состоят из регистра содержащего суммарное значение, и четырех регистров содержащих значения накопленные по тарифам. Если в модуле не инициализированы часы, то накопление тарифных значений не производится.

Регистры импульсов 2 всего и по четырем тарифам — целое число в диапазоне от 0 до 4294967295 соответствующее количеству импульсов насчитанному импульсным входом 2. Состоят из регистра содержащего суммарное значение, и четырех регистров содержащих значения накопленные по тарифам. Если в модуле не инициализированы часы, то накопление тарифных значений не производится.

Время — содержит текущую дату и время модуля.

Регистр флагов состояния, содержит такие параметры как:

- Часы не инициализированы, признак того что текущая дата и время модуля недостоверны (внимание! При установленном флаге модуль перестает сохранять суточные и часовые архивы, а так-же насчитывать тарифные значения регистров);

- Регистры импульсов восстановлены. Данный флаг указывает на то что из за пониженного напряжения питания значения регистров импульсов стали недостоверными и были восстановлены последние достоверные данные из суточного архива;

- Устройство не зарегистрировано. Данный флаг указывает на то что устройство не зарегистрировано в сети LoraWAN;

- GPIO Ch1 out. Канал GPIO 1 настроен как выход;
- GPIO Ch2 out. Канал GPIO 2 настроен как выход;
- GPIO Ch3 out. Канал GPIO 3 настроен как выход;
- GPIO ChV out. Канал GPIO V настроен как выход;
- GPIO Ch1. В канале GPIO 1 логическая 1;
- GPIO Ch2. В канале GPIO 2 логическая 1;
- GPIO Ch3. В канале GPIO 3 логическая 1;
- GPIO ChV. В канале GPIO V логическая 1.

2.2. Архив событий.

Данный архив содержит дату и время возникновения последнего события для каждого типа событий.

В модуле предусмотрены следующие типы событий:

- Перезапуск устройства. Содержит дату, время и причину последнего перезапуска устройства;

- Установка часов. Содержит дату и время последнего события корректировки времени;

- Установка начальных показаний. Содержит дату и время последней установки начальных показаний в модуле.

2.3. Архивные значения.

Модуль позволяет вести суточные и часовые архивы.

2.3.1. Суточный архив.

Глубина хранения суточных архивов находится в пределе 361 — 365 записей.

Запись суточного архива формируется в 00:00:00 каждого дня, либо при установке часов если часы были переведены через сутки.

Если часы не инициализированы, то формирование суточных записей не происходит.

При установке начальных показаний суточный архив стирается.

В каждой записи хранятся следующие данные:

- Дата формирования записи в формате (год,месяц,день);
- Регистр импульсов 1 на момент формирования записи;
- Регистр импульсов 1 тариф 1 на момент формирования записи;
- Регистр импульсов 1 тариф 2 на момент формирования записи;
- Регистр импульсов 1 тариф 3 на момент формирования записи;
- Регистр импульсов 1 тариф 4 на момент формирования записи;
- Регистр импульсов 2 на момент формирования записи;
- Регистр импульсов 2 тариф 1 на момент формирования записи;
- Регистр импульсов 2 тариф 2 на момент формирования записи;
- Регистр импульсов 2 тариф 3 на момент формирования записи;
- Регистр импульсов 2 тариф 4 на момент формирования записи;
- Регистр флагов состояния устройства на момент формирования записи.

2.3.2. Часовой архив

Глубина хранения часового архива 4 месяца минус 32 часа.

Запись часового архива происходит каждый час, либо при установке часов если часы были переведены через час.

Если часы не инициализированы, то формирование часового архива не происходит.

При установке начальных показаний часовой архив стирается.

В каждой записи хранятся следующие данные:

- Год, месяц, день, час к которым относятся данные содержащиеся в записи;
- Приращение регистра импульсов 1 в течении часа;
- Приращение регистра импульсов 2 в течении часа;
- Статус устройства на конец часа.

2.4. Конфигурация устройства.

2.4.1. Неизменяемая конфигурация устройства.

Данная конфигурация записывается производителем модуля и не может быть изменена пользователем.

Неизменяемая конфигурация устройства содержит следующие параметры:

- Идентификатор устройства DevEUI 8б. Имеет следующий формат 0xADvvYYMMnnnnnnnn, где AD-тип устройства, vv-версия прошивки, YYMM-год и месяц выпуска устройства, nnnnnnnn-сквозной номер в диапазоне [0x00000000,...,0xffffffff];
- Идентификатор приложения AppEUI 8б, по умолчанию 0x2222222222222222;
- Ключ приложения AppKey 16б, по умолчанию 0x33333333333333333333333333333333;
- Уход часов за сутки — содержит калибровочный коэффициент для коррекции часов;
- Электронная пломба — блокирует возможность записи неизменяемой конфигурации, а в случае если модуль не поддерживает GPIO блокирует возможность изменять GPIO. Электронная пломба устанавливается производителем при выпуске с производства.

2.4.2. Изменяемая конфигурация устройства.

Данная конфигурация содержит параметры которые пользователь может изменить.

Изменяемая конфигурация устройства содержит следующие параметры:

- Переход на летнее зимнее время (отключен, включен, автоматически);
- Частота попыток регистрации в сети (1,2,3). Количество попыток регистрации в сети Lora WAN в сутки;
- Частота попыток передачи данных (1,2,3,4,6). Количество передач данных по LoraWan в сутки;
- Максимум передач в день всего (1..15). Ограничение на количество передач по LoraWAN в сутки. В заводских настройках - 10;
- Максимум возможных запросов по LoraWAN (0..15). Ограничение на количество запросов по LoraWAN с верхнего уровня в сутки. В заводских настройках - 3.
- Параметры передаваемые модулем по умолчанию;
- Конфигурация передач с подтверждением в протоколе LoraWAN. Содержит статическую и случайную составляющие (диапазон [0,..,15]). Модуль формирует счетчик по формуле (статическая составляющая + случайное число в диапазоне от 0 до случайная составляющая). Если счетчик равен 0, то при передаче данных по LoraWAN используется пакет с подтверждением приема, иначе — без подтверждения. При каждой успешной передаче данных счетчик уменьшается на единицу. Если статическая и случайная составляющие равны 15, то счетчик не уменьшается (т.е. все передачи будут без подтверждения. !!! не рекомендуется использовать данный режим т.к. модуль никогда не узнает связан ли он с сервером приложений!!!). Если статическая и случайная составляющие равны 0, то все передачи будут с подтверждением (данный режим используется по умолчанию). Также следует учитывать, что при запросе верхним уровнем дополнительных данных, модуль всегда отвечает с подтверждением.
- Конфигурация импульсного входа 1. Позволяет задать тип измеряемых импульсов, минимальную длительность при которой импульс будет зарегистрирован и количество импульсов в единице младшего разряда регистра импульсов 1;
- Конфигурация импульсного входа 2. Позволяет задать тип измеряемых импульсов, минимальную длительность при которой импульс будет зарегистрирован и количество импульсов в единице младшего разряда регистра импульсов 2;
- Конфигурация GPIO. Задаёт для каждого канала GPIO является ли он входом или выходом. Если канал является выходом то дополнительно задается уровень на этом канале;

- Расписание GPIO. С помощью расписания GPIO можно задать сценарии по которым будут формироваться сигналы GPIO, а так же запуск внеочередной передачи по LoRaWAN при обнаружении заданного сигнала;

- Тарифное расписание. Содержит тарифы рабочих и выходных дней для каждого импульсного входа (минимальный интервал переключения – пол часа), фиксированные праздничные дни (в заводских настройках - белорусские праздники). Выходные дни (суббота, воскресенье) - рассчитываются устройством самостоятельно.

Параметры передаваемые модулем по умолчанию - позволяет задать список параметров из которых будет формироваться пакет LoRaWAN передаваемый по расписанию.

Пакет должен содержать не более 48 байт.

В состав пакета могут входить следующие параметры:

- Маска передаваемых параметров (2б передается всегда);
- Текущее время модуля (6б);
- Дата на начало суток (3б);
- Регистр импульсов 1 (4б);
- Регистр импульсов 2 (4б);
- Регистр импульсов 1 на начало суток (4б);
- Регистр импульсов 2 на начало суток (4б);
- Регистр статуса устройства (2б);
- Регистр статуса устройства на начало суток (2б);
- Регистр импульсов 1 и регистр импульсов 2 по тарифу 1 (8б);
- Регистр импульсов 1 и регистр импульсов 2 по тарифу 2 (8б);
- Регистр импульсов 1 и регистр импульсов 2 по тарифу 3 (8б);
- Регистр импульсов 1 и регистр импульсов 2 по тарифу 4 (8б);

При выпуске с производства модуль настроен на передачу следующих параметров:

- Текущее время модуля (6б);
- Регистр импульсов 1 (4б);
- Регистр импульсов 2 (4б);
- Регистр статуса устройства (2б).

3. Каналы GPIO

В модуле есть 3 канала GPIO которыми пользователь может управлять: Ch1, Ch2, Ch3. Так же есть один виртуальный канал (без физической клеммы): ChV.

Каналам GPIO можно задать функцию которую они будут выполнять: вход, выход.

Если канал является выходом, то ему можно задать логический уровень в котором он будет находиться.

Управлять каналами GPIO можно: с помощью интерфейса USB программой «LoraUtilPulseCounter», с помощью протокола LoraWAN, с помощью расписания GPIO.

При использовании каналов GPIO пользователь сам несет ответственность за потребление устройства и срок службы батареи.

3.1. Расписание GPIO

С помощью расписания GPIO можно задать сценарии по которым будут формироваться сигналы GPIO, а так же запуск внеочередной передачи по LoraWAN при обнаружении заданного сигнала.

Расписание GPIO состоит из 5 «задач GPIO» и 4 «сторожей GPIO».

3.1.1. Задача GPIO

Задача GPIO позволяет в заданный момент времени выставить нужные сигналы на заданных каналах GPIO.

Настройка момента запуска включает в себя: множество месяцев {1,...,12} в которые будет срабатывать задача (если не один месяц не выбран, то задача выключена), переменная задающая день срабатывания {любой, через заданный интервал, фиксированный}, переменная задающая час срабатывания {любой, через заданный интервал, фиксированный}, переменная задающая минуту срабатывания {любой, через заданный интервал, фиксированный}, переменная задающая секунду срабатывания {любой, через заданный интервал, фиксированный}.

Настройка каналов GPIO включает в себя выбор каналов которые нужно сделать выходами, и уровни сигналов которые следует выставить выбранным каналам.

3.1.2. Сторож GPIO

Сторож GPIO позволяет отслеживать состояние выбранных каналов GPIO и если это состояние удерживается заданное время то выполнить заданные действия.

Действия могут состоять из установки каналов GPIO и принудительной передачи по LoraWAN.

3.1.3. Алгоритм работы расписания GPIO

Модуль каждую секунду проверяет критерии срабатывания задач GPIO и сторожей GPIO. Если проверка удовлетворяет критерию срабатывания, то выполняется действие заданное в сработавшей задаче или стороже. Критерии срабатывания проверяются сначала для задач GPIO (от младшей задачи к старшей). Затем проверяются критерии срабатывания для сторожей GPIO (от младшего сторожа к старшему).

Критерием срабатывания задачи GPIO является наступление заданного момента запуска задачи. Т.е. задача GPIO может запуститься только в соответствии с заданным расписанием времени запуска.

Критерием срабатывания сторожа GPIO является обнаружение заданного состояния выбранных каналов GPIO и удержание этого состояния в течение заданного времени регистрации события. Если время регистрации события равно нулю, то сторож работает сразу при обнаружении нужного состояния. Признаком обнаружения заданного состояния

является сравнение отслеживаемых каналов в текущий момент времени с предыдущим. Если в предыдущий момент времени отслеживаемые каналы находились не в заданном состоянии, а текущий момент времени отслеживаемые каналы находятся в заданном состоянии то сторож считает что заданное состояние наступило. Предыдущее состояние сохраняется каждую секунду после обработки всех задач и сторожей.

3.1.4. Примеры конфигураций расписания GPIO

Для удобства записи далее будут использоваться следующие обозначения.

Задача GPIO – M[1,..12,*]:D[*,%d,d]:h[*,%d,d]:m[*,%d,d]:s[*,%d,d]:p[V321,1101]

где:

- M[1,..12,*] - выбор месяцев в которые будет срабатывать задача, *-любой месяц;
- D[*,%d,d] – выбор дней в которые будет срабатывать задача, *-любой день, %d – d в день кратный d, d – в день равный d.
- h[*,%d,d] – выбор часов в которые запустится задача, параметры как и выше.
- m[*,%d,d] - выбор минут в которые запустится задача, параметры как и выше.
- s[*,%d,d] - выбор секунд в которые запустится задача, параметры как и выше.
- p[V321,1101] – V321 – каналы которые следует сделать выходами, 1101 – уровни которые следует задать соответствующим каналам. Например p[31,01] – каналы 3 и 1 сделать выходами, каналу 3 задать уровень 1, каналу 1 задать уровень 0.

Сторож GPIO – T[0..32767]:reg[pin,lvl]:setPin[pin,lvl]:LoraWan

где:

- T[0..32767] – задает время удержания заданного состояния.
- reg[pin,lvl] – задает отслеживаемое событие, pin – каналы GPIO за которыми мы следим (например V2), lvl – ожидаемые состояния выбранных каналов (например 10).
- setPin[pin,lvl] – установить выбранные каналы при срабатывании сторожа, pin – каналы которые следует сделать выходами, lvl – уровни которые следует задать соответствующим каналам.
- LoraWan - выполнить передачу LoraWAN при срабатывании сторожа. Отсутствует если не надо.

Пример 1.

Задача.

В летние и весенние месяцы с 6:00 до 21:00 канал Ch1 устанавливать в 0, в остальное время в 1.

В осенние и зимние месяцы с 9:00 до 18:00 канал Ch1 устанавливать в 0, в остальное время в 1.

Задача GPIO1 - M[3-8]:D[*]:h[6]:m[0]:s[0]:p[1,0]

// в месяцы 3-8, любой день, 6 часов, 0 минут, 0 секунд, Ch1 установить в 0

Задача GPIO2 - M[3-8]:D[*]:h[21]:m[0]:s[0]:p[1,1]

// в месяцы 3-8, любой день, 21 час, 0 минут, 0 секунд, Ch1 установить в 1

Задача GPIO3 – M[1,2,9-12]:D[*]:h[9]:m[0]:s[0]:p[1,0]

// в месяцы 1,2,9-12, любой день, 9 часов, 0 минут, 0 секунд, Ch1 установить в 0

Задача GPIO4 - M[1,2,9-12]:D[*]:h[18]:m[0]:s[0]:p[1,1]

// в месяцы 1,2,9-12, любой день, 18 часов, 0 минут, 0 секунд, Ch1 установить в 1

Пример 2.

Задача.

Если на канале Ch3 обнаружен низкий уровень в течении 3 секунд, то установить Ch1 в высокий уровень на время 5 секунд по истечении которых выполнить передачу по LoraWAN.

Сторож GPIO1 – T[3]:reg[3,0]:setPin[1,1]

Сторож GPIO2 – T[5]:reg[1,1]:setPin[1,0]:LoraWAN

Пример 3.

Задача.

Сделать подсчет времени когда на канале Ch3 низкий уровень. Время необходимо сохранять посуточно.

Задача GPIO1 - M[1-12]:D[*]:h[*]:m[*]:s[*]:p[V,1]

// в каждую секунду, ChV установить в 1

Задача GPIO2 – M[1-12]:D[*]:h[*]:m[*]:s[%2]:p[V,0]

// в каждую вторую секунду, ChV установить в 0

// при выполнении этих задач на виртуальном канале получится миандр с периодом 2с

Сторож GPIO1 – T[0]:reg[V3,00]:setPin[1,0]

// Если ChV и Ch3 в 0, то установить Ch1 в 0

Сторож GPIO2 – T[0]:reg[V3,10]:setPin[1,1]

// Если ChV в 1, а Ch3 в 0, то установить Ch1 в 1

// теперь если Ch3 в 0 то миандр с периодом 2с транслируется на канал Ch1

Нам осталось завести Ch1 на один из импульсных входов нашего модуля, после чего в регистре импульсов мы увидим время в течение которого Ch3 в низком уровне. Вес младшего значения в регистре импульсов – 2с. Т.к. вся логика работы регистров импульсов сохраняется, то значения регистра на начало суток попадут в суточный архив.

4. Протокол передачи данных USB-UART.

Для конфигурации модуля используется разъем USB. Протокол передачи данных позволяет прочитать и настроить все данные хранящиеся в модуле за исключением неизменяемой конфигурации.

Для параметризации модуля используется ПО «LoraUtilPulseCounter» которое предоставляется производителем устройства.

5. Протокол передачи данных по LoraWAN.

Модуль передает данные в периоды времени заданные расписанием. Верхний уровень может послать дополнительный запрос после приема данных от модуля.

Максимальный размер передаваемых данных ограничен 48 байтами.

В случае если модулю необходимо передать MAC команду протокола LoraWAN и при этом не хватает места для передаваемых данных, будет произведена посылка пустого пакета с командами MAC на порт LoraWAN 0.

Модуль может передавать данные с подтверждением и без см.п.2.4.2. Если модуль отправляет данные по запросу с верхнего уровня, то такая передача всегда идет с подтверждением.

Протокол данных предусматривает наличие пяти команд: чтение, запись, стереть, сумма.

Команда «чтение» использует порт LoraWAN 1.

Команда «запись» использует порт LoraWAN 2.

Команда «стереть» использует порт LoraWAN 3.

Команда «сумма» использует порт LoraWAN 4.

При использовании многобайтных параметров, младший байт передается первым.

5.1. Форматы команд.

5.1.2. Команда «чтение».

От модуля к верхнему уровню пакет имеет следующий вид:

Маска передаваемых параметров 2б	Параметр1	Параметр2	...	ПараметрN
----------------------------------	-----------	-----------	-----	-----------

Маска передаваемых параметров определяет параметры переданные в пакете.

Параметры в пакете следуют от младшего указанного в маске к старшему.

От верхнего уровня к модулю (запрос дополнительных параметров) пакет имеет следующий вид:

Маска запрашиваемых параметров 2б	Дополнительные данные
-----------------------------------	-----------------------

Маска запрашиваемых параметров определяет параметры запрашиваемые у модуля.

Дополнительные данные присутствуют при запросе даты на начало суток, регистров импульсов 1 на начало суток, регистров импульсов 2 на начало суток, регистра статуса устройства на начало суток, часовых архивов, суммы часовых архивов, конфигурации задач, конфигурации сторожей, тарифов, фиксированных праздников, выходных дней.

В этом случае в дополнительных данных передаются:

- для суточных архивов, дата запрашиваемой записи в формате uint8[3]{год-2000, месяц, день};

- для часовых архивов, дата и час запрашиваемой записи в формате uint8[4]{год-2000, месяц, день, час};

- для суммы часовых архивов, дата и часы в формате uint8[5]{год-2000, месяц, день, час с которого считать сумму, час до которого считать сумму включительно};
- для конфигурации задач, uint8 msk — маска запрашиваемых элементов, младший бит — младший элемент;
- для конфигурации сторожей, uint8 msk - маска запрашиваемых элементов, младший бит — младший элемент;
- для тарифов, uint8 tar - определяет тип тарифного расписания. Bit0 — 0: рабочие дни, 1: выходные дни. Bit1 — 0: тарифное расписание для импульсного входа 1, 1 — тарифное расписание для импульсного входа 2;
- для фиксированных праздников, uint16 msk — маска запрашиваемых месяцев, bit0 — январь, .. , bit11 — декабрь;
- для выходных дней, дата запрашиваемого элемента в формате uint8[2]{год-2000, месяц}.

5.1.3. Команда «запись».

От верхнего уровня к модулю пакет имеет следующий вид:

Маска записываемых параметров 2б	Параметр1	Параметр2	...	ПараметрN
----------------------------------	-----------	-----------	-----	-----------

Маска записываемых параметров определяет параметры для записи переданные в пакете.

Параметры в пакете следуют от младшего указанного в маске к старшему.

5.1.4. Команда «стереть».

От верхнего уровня к модулю пакет имеет следующий вид:

Маска стираемых параметров 2б

Маска стираемых параметров определяет параметры которые нужно стереть.

Команды для стирания регистров импульсов действуют на все блоки регистров.

5.1.5. Команда «сумма».

От верхнего уровня к модулю пакет имеет следующий вид:

Маска параметров для суммы 2б	Параметр1	Параметр2	...	ПараметрN
-------------------------------	-----------	-----------	-----	-----------

Маска параметров для суммы определяет параметры для суммирования переданные в пакете.

Доступны следующие параметры для суммы:

- текущая дата-время, в этом случае параметр должен содержать значение в секундах которое следует добавить к текущему времени модуля. Формат параметра — int32;
- регистры импульсов 1 всего и по четырем тарифам, в этом случае параметр должен содержать значение, которое следует добавить к текущему значению регистра. Формат параметра — int32;
- регистры импульсов 2 всего и по четырем тарифам, в этом случае параметр должен содержать значение, которое следует добавить к текущему значению регистра. Формат параметра — int32.

Параметры в пакете следуют от младшего указанного в маске к старшему.

5.2. Форматы данных используемые в протоколе.

Дата

```
Date{
    uint8 year;
    uint8 month;
    uint8 day;
}
```

year — год, указывается как смещение к 2000г.
month — месяц в диапазоне 1..12.
day — день месяца в диапазоне 1..31.

Время

```
Time{
    uint8 hour;
    uint8 min;
    uint8 sec;
}
```

hour — часы в диапазоне 0..23.
min — минуты в диапазоне 0..59.
sec — секунды в диапазоне 0..59.

Дата и время

```
DateTime{
    Date date;
    Time time;
}
```

date — дата.
time — время.

Запись часового архива

```
HourSrez{
    uint8 hour;
    uint16 srezI1;
    uint16 srezI2;
    uint16 statusDev;
}
```

hour — час к которой относится запись.
srezI1 — приращение регистра импульсов 1.
srezI2 — приращение регистра импульсов 2.
statusDev — статус устройства на конец часа.

Сумма срезов из часового архива

```
SumHourSrez{
    uint8 hourStart;
    uint8 hourEnd;
    uint32 sumSrezI1;
    uint32 sumSrezI2;
    uint16 orStatusDev;
}
```

hourStart, hourEnd — часы с какого по какой надо просуммировать записи.

sumSrezI1 — содержит сумму срезов для регистра импульсов 1.

sumSrezI2 — содержит сумму срезов для регистра импульсов 2.

orStatusDev — содержит все события зафиксированные в суммируемых срезах.

Статус устройства.

```
StatusDev{
    uint16 status;
}
```

status — битовое поле размерности 16 бит.

№ бита	Расшифровка	Значение 0	Значение 1
bit0	Часы не инициализированы	Нет	Да
bit1	Регистры импульсов восстановлены	Нет	Да
bit2	Устройство не зарегистрировано	Нет	Да
bit3-bit7	Зарезервировано	X	X
bit8	Значение канала Ch1	0	1
bit9	Значение канала Ch2	0	1
bit10	Значение канала Ch3	0	1
bit11	Значение канала ChV	0	1
bit12	Функция канала Ch1	Вход	Выход
bit13	Функция канала Ch2	Вход	Выход
bit14	Функция канала Ch3	Вход	Выход
bit15	Функция канала ChV	Вход	Выход

Сезоны

```
Season{
    DateTime toWinter;
    DateTime toSummer;
    uint8 seasonEnable;
}
```

toWinter — дата время перехода на зимний сезон.

toSummer — дата время перехода на летний сезон.

SeasonEnable — {

0-переход сезонов отключен;

1-переход сезонов включен, время перехода задано в toWinter и toSummer
2-переход сезонов включен, время перехода toWinter и toSummer
рассчитывается автоматически.

}.
}

Параметры LoraWAN.

ConfigLoraWAN{

uint8 loraJoinPeriod;
uint8 loraSendPeriod;
uint16 loraDefPrmSend;
uint8 loraMaxSendPerDay;

}

loraJoinPeriod — периодичность попыток регистрации в сети{

0-один раз в день.
1-два раза в день.
2-три раза в день.

}.
}

loraSendPeriod — расписание отправки данных по сети LoraWAN{

0-один раз в день.
1-два раза в день.
2-три раза в день.
3-четыре раза в день.
4-шесть раз в день.

}.
}

loraDefPrmSend — маска задающая параметры отправляемые модулем, см.п. «Маска параметров».

loraMaxSendPerDay — задает предельное количество пакетов, которое модуль может выдать в день{

bit0..bit3 — максимальное количество передач всего [1..15].

bit4..bit7 — максимальное количество передач по запросу с верхнего уровня [0..15].

}.
}

Конфигурация импульсного входа

```
pulseCfg{
  uint8 impCfg;
  uint16 impToRegCnt;
}
```

impCfg - bit7 — 1-считаем 1, 0-считаем 0. bit6-bit0 — минимальное время импульса при котором он будет зарегистрирован [mS] (защита от дребезга).

impToRegCnt — количество импульсов в единице младшего разряда регистра [1..65535].

Причина перезапуска устройства.

```
uint8 resetCause;
resetCause — причина перезапуска{
  0x40-System Reset.
  0x20-Watchdog Reset.
  0x10-External Reset.
  0x04-Brown Out 33 Detector Reset.
  0x02-Brown Out 12 Detector Reset.
  0x01-Power-On Reset.
}.
```

Конфигурация GPIO.

```
uint8 gpioCfg
```

bit7-bit4 — задают функцию каналов ChV-Ch1 соответственно. 1-выход, 0-вход.

bit3-bit0 — для каналов являющихся выходами задают уровень сигнала. 1-высокий, 0-низкий.

Конфигурация времени срабатывания задачи GPIO

```
StrTimeTask{
  uint16 monthFlag;
  uint8 day;
  uint8 hour;
  uint8 min;
  uint8 sec;
}
```

monthFlag — bit15 - 1: не срабатывать при сбое часов, bit11..bit0 - 1:месяц выбран, 0:месяц отключен. Младший месяц начинается с младшего бита. Если все месяцы отключены то задача неактивна.

day - 0x80 — любой день, 0x80+x(7b) — день кратный x, 0x00+x(7b) — день равный x.

hour - 0x80 — любой час, 0x80+x(7b) — час кратный x, 0x00+x(7b) — час равный x.

min - 0x80 — любая минута, 0x80+x(7b) — минута кратная x, 0x00+x(7b) — минута равная x.

sec - 0x80 — любая секунда, 0x80+x(7b) — секунда кратная x, 0x00+x(7b) — секунда равная x.

Конфигурация задачи GPIO

```
StrTaskGpio{
```

```
StrTimeTask timeTask;
uint8 gpioCfg;
}
```

timeTask — задает время запуска задачи.

gpioCfg - задание на установку каналов GPIO. bit7-bit4 — каналы которые надо сделать выходами, bit3..bit0 — уровни которые надо выставить выбранным каналам.

Конфигурация сторожа GPIO

```
StrSpyGpio{
    uint16 enAndPeriod;
    uint8 regEvent;
    uint8 workPin;
    uint8 workLora;
}
```

enAndPeriod — bit15 — 1-включен, 0-выключен. Bit14..bit0 — период регистрации события в секундах.

regEvent - bit7-bit4 — каналы за которыми надо следить. bit3-bit0 — отслеживаемые значения каналов.

workPin - задание на установку каналов GPIO. bit7-bit4 — каналы которые надо сделать выходами, bit3..bit0 — уровни которые надо выставить выбранным каналам.

workLora — 1 - выполнить передачу LoraWAN, 0 — не выполнять.

Конфигурация счетчика пакетов без подтверждения

```
uint8 ucfGenCfg;
```

ucfGenCfg содержит два числа. Bit3..bit0 – содержит статическую составляющую ucfStatic, а bit7..bit4 – содержит случайную составляющую ucfRandom. Счетчик пакетов без подтверждения формируется по формуле (ucfStatic+random(ucfRandom)). См.п. 2.4.2.

Тарифы

```
StrTarifs{
    uint8 tar;
    uint8 tariffs[24];
}
```

tar — определяет тип тарифного расписания. Bit0 — 0: рабочие дни, 1: выходные дни. Bit1 — 0: тарифное расписание для импульсного входа 1, 1 — тарифное расписание для импульсного входа 2.

tarifs — каждый элемент массива определяет тарифы для соответствующего часа. Младшие 4 бита это тарифы действующие в первом получасе (младший бит — первый тариф), старшие 4 бита это тарифы действующие в втором получасе (младший бит — первый тариф).

Фиксированные праздники

```
StaticHoliday{
    uint16 monthMsk;
    uint32 staticHoliday[];
}
```

monthMsk — маска для выбора месяцев. Bit0 — январь, .. , bit11 — декабрь.

StaticHoliday[] - массив масок определяющих выходные дни, размер массива определяется переменной monthMsk. Каждый элемент побитно определяет выходные дни в соответствующем месяце. Bit0 — первый день месяца, .. , bit30 — 31 день месяца.

Выходные дни

```
WeekDay{  
    uint8 year;  
    uint8 month  
    uint32 weekDay;  
}
```

year — год, указывается как смещение к 2000г.

month — месяц в диапазоне 1..12.

weekDay - побитно определяет выходные дни в соответствующем месяце. Bit0 — первый день месяца, .. , bit30 — 31 день месяца.

5.3. Маска параметров.

Маска параметров используется для определения параметров передаваемых в пакетах LoRaWAN.

Размерность маски параметров — 2 байта.

Старшие 4 бита определяет группу параметров.

Младшие 12 бит побитно выбирает параметры из группы. Значение бита равное единице означает что параметр выбран, значение равное нулю означает что параметр не выбран.

Маска		Название параметра	Чтение	Запись	Стереть	Сумма	Размер параметра и тип
Bit[15..12]	Bit[11..0]						
Основная группа параметров							
0x0	0x001	Текущие дата и время	Да	Да	Нет	Да	DateTime — 6б для команд чтение, запись, стереть. int32 для команды сумма
0x0	0x002	Дата	Да	Нет	Нет	Нет	Date - 3б
0x0	0x004	Регистр импульсов 1	Да	Да	Да	Да	uint32 - для команд чтение, запись. int32 для команды сумма
0x0	0x008	Регистр импульсов 2	Да	Да	Да	Да	uint32 - для команд чтение, запись. int32 для команды сумма
0x0	0x010	Регистр импульсов 1 на начало суток	Да	Нет	Нет	Нет	uint32
0x0	0x020	Регистр импульсов 2 на начало суток	Да	Нет	Нет	Нет	uint32
0x0	0x040	Статус устройства	Да	Нет	Нет	Нет	StatusDev — 2б.
0x0	0x080	Статус устройства на начало суток	Да	Нет	Нет	Нет	StatusDev — 2б.
0x0	0x100	Регистры импульсов тариф1	Да	Да	Да	Да	{ uint32 regI1T1; uint32 regI2T1; } - для команд чтение, запись. {

							int32 addRegI1T1; int32 addRegI2T1; } - для команды сумма
0x0	0x200	Регистры импульсов тариф2	Да	Да	Да	Да	{ uint32 regI1T2; uint32 regI2T2; } - для команд чтение, запись. { int32 addRegI1T2; int32 addRegI2T2; } - для команды сумма
0x0	0x400	Регистры импульсов тариф3	Да	Да	Да	Да	{ uint32 regI1T3; uint32 regI2T3; } - для команд чтение, запись. { int32 addRegI1T3; int32 addRegI2T3; } - для команды сумма
0x0	0x800	Регистры импульсов тариф4	Да	Да	Да	Да	{ uint32 regI1T4; uint32 regI2T4; } - для команд чтение, запись. { int32 addRegI1T4; int32 addRegI2T4; } - для команды сумма
Суточные и часовые архивы							
0x1	0x001	Дата архива	Да	Нет	Нет	Нет	Date - 36
0x1	0x002	Регистры импульсов на начало суток	Да	Нет	Нет	Нет	{ uint32 regI1; uint32 regI2; }
0x1	0x004	Регистры импульсов на начало суток по тарифу 1	Да	Нет	Нет	Нет	{ uint32 regI1T1; uint32 regI2T1; }
0x1	0x008	Регистры импульсов на начало суток по тарифу 2	Да	Нет	Нет	Нет	{ uint32 regI1T2; uint32 regI2T2; }

0x1	0x010	Регистры импульсов на начало суток по тарифу 3	Да	Нет	Нет	Нет	{ uint32 regI1T3; uint32 regI2T3; }
0x1	0x020	Регистры импульсов на начало суток по тарифу 4	Да	Нет	Нет	Нет	{ uint32 regI1T4; uint32 regI2T4; }
0x1	0x040	Статус устройства на начало суток	Да	Нет	Нет	Нет	StatusDev — 2б.
0x1	0x080	Часовой срез	Да	Нет	Нет	Нет	HourSrez — 7б.
0x1	0x100	Сумма часовых срезов	Да	Нет	Нет	Нет	SumHourSrez — 12б.
Тарификация							
0xD	0x001	Тарифы	Да	Да	Нет	Нет	StrTarifs — 25б.
0xD	0x002	Фиксированные праздники	Да	Да	Нет	Нет	StaticHoliday — 2+4*(кол-во месяцев)
0xD	0x004	Выходные дни	Да	Нет	Нет	Нет	WeekDay — 6б.
Группа параметров «Архив событий»							
0xE	0x001	Перезапуск устройства	Да	Нет	Да	Нет	{ DateTime-6б, resetCause-1б }-7б
0xE	0x002	Установка времени	Да	Нет	Да	Нет	DateTime-6б
0xE	0x004	Установка начальных показаний	Да	Нет	Да	Нет	DateTime-6б
Группа параметров «Конфигурация устройства»							
0xF	0x001	DevEUI	Да	Нет	Нет	Нет	8б
0xF	0x002	AppEUI	Да	Нет	Нет	Нет	8б
0xF	0x004	AppKey	Да	Нет	Нет	Нет	16б
0xF	0x008	Коррекция времени	Да	Нет	Нет	Нет	int16
0xF	0x010	Сезоны	Да	Да	Нет	Нет	Season — 13б.
0xF	0x020	Параметры LoraWAN	Да	Да	Нет	Нет	ConfigLoraWAN — 5б.
0xF	0x040	Конфигурация счетчика пакетов без подтверждения	Да	Да	Нет	Нет	ucfGenCfg

0xF	0x080	Конфигурация импульсных входов	Да	Да	Нет	Нет	{ pulseCfg – вход1 pulseCfg – вход2 }-6б
0xF	0x100	Установка GPIO	Да	Да	Нет	Нет	gpioCfg
0xF	0x200	Конфигурация задачи GPIO	Да	Да	Нет	Нет	Msk 1b – побитно определяет номера заданий. StrTaskGpio[] - задания GPIO
0xF	0x400	Конфигурация сторожей GPIO	Да	Да	Нет	Нет	Msk 1b – побитно определяет номера сторожей. StrSpyGpio[] - сторожа GPIO