Общее описание:

Имеется два устройства.

УСТРОЙСТВО 2 (стационарное):

Устройство 2 состоит из:  
а) ESP8266

б) bluetooth-модуль (HC-08 или другой ?)

в) модуль памяти М95320

г) микроконтроллер STM32  
д) кварц на 9.216МГц  
е) ШИМ стабилизатор

ж) обвязка

Большую часть времени устройство находится в режиме сна. Устройство должно просыпаться при подачи питания на расходомер. Питание на расходомер подается контролируется герконовым замыкателем.

Устройство считает импульсы с расходомера и суммирует их. Если между импульсами была пауза больше 10 секунд, тогда эти импульсы передаются на стационарное устройство и затем удаляются из памяти устройства 1.

Если подключиться к стационарному устройству не удалось, то повторное подключение происходит через 30 секунд циклично.

Если при неудачном подключении к стационарному устройству подаются ещё импульсы, то они не суммируются к тем импульсам, которые еще не переданы, должны передаваться отдельным пакетом, т.е. импульсы должны передаваться пакетами (отдельным пакетом считается набор импульсов, у которых пауза между импульсами более 10сек).

Вместе с пакетом импульсов передаются след. данные:

а) серийный номер устройства (задаётся в прошивке)

б) версия прошивки (задаётся в прошивке)

Должна быть возможность будить устройство через Bluetooth, чтобы получать пинги.

Используем powerDown ножку внешнего прерывания мы подвязываем на ТХ

Обновление прошивки по проводу:

Стационар:

Память  
контроллер  
HC-08

Esp8266

1) все наливы должны передаваться в импульсах

2) ошибки должны логироваться на сервер

3) питание должно быть от пальчиковых батареек, c пингами должен приходить остаток заряда батареек

4) сейчас если сервер не доступен, то все наливы будут будут объединяться в один налив, надо это исправить чтобы каждый налив был записан отдельно (и сохранялся даже если пропадет питание). Возможно если серевер не доступен длительное время надо перезагрузить устройство

5) web-интерфейс

6) питание с ESP должно отключаться физически, а не зажиматься ножка reset (возможно это спасет от ошибок ESP и сэкономить заряд батареек)

Решение будет таким: на питание расходомера будет завязан геркон, который будет замыкаться, когда появится мазут. Когда мазут отсутствует, то питание пропадает.

Время пингов будем сокращать, возможно до нескольких часов. Сам расходомер будет активен не более 1мин в сутки. Если понадобится, то будет использовано 20 батареек (не желательно, но возможно). Нужно добиться автономной работы 1 год.

Доступ в интернет происходит путем подключения к роутеру через WI-FI (имя сети SSID и пароль для подключения лучше вынести в отдельные переменные, чтобы я их мог менять самостоятельно)

При включении контроллера (когда на него подается питание), он должен подключиться к веб серверу и получить текущую дату и время (timestamp).

После получения настроек времени на контроллере запускаются часы.

Каждые 4 часа, контроллер должен запрашивать дату и время заново (таким образом синхронизируется время)

Если подключиться к WI-FI роутеру не удалось, то повторное подключение должно происходить через 1 минуту (до тех пор пока не подключиться)

Если получить настройки даты и времени не удалось (не доступен веб-сервер), то повторная попытка должна происходить через 30 секунд

После получения текущих настроек даты и времени контроллер уходит в спящий режим.

Каждые 5 минут контроллер должен посылать на сервер своё состояние (пинг) в формате JSON (текущее время, состояние расходомера, состояние самого контроллера, версию прошивки и всю служебную информацию, которую можно передать).

Ошибки логирутся и отсылаются на сервер в формате JSON.

Контроллер должен входить в спящий режим после отправки данных о наливе на сервер или же через 30 неактивности.

Выход из спящего режима осуществляется при активности расходомера.

При активности расходомера в память контроллера записывается начальное время активности расходомера и данные с расходомера. Если промежуток времени между активностью расходомера составляет меньше 15 секунд, то это считается одним циклом активности (одним наливом).

После окончания цикла активности расходомера (после налива), данные считываются из памяти и отсылаются на сервер в формате JSON.

После **успешной отправки** данных на сервер эти данные удаляются из памяти контроллера.

Если по каким-то причинам не удалось отправить данные на сервер (нет подключения к интернету или сервер не доступен). То повторная отправка данных происходит через 20 секунд.

Если начался цикл налива, но в контроллере НЕ установлены настройки даты и времени, то в память записывается значение времени налива равное нулю и эти данные НЕ отсылаются на сервер.

Если в памяти есть данные с неопределенной датой, то каждые 5 секунд запускается цикл, который увеличивает значение времени данных на 5 и проверяет наличие настроек даты и времени в контроллере и если в контроллере появились настройки даты и времени, тогда от этого времени отнимается время записанное в данных (в секундах) и в данные записывается это время. Таким образом узнаётся время когда цикл налива начался. После этого данные отсылаются на сервер и удаляются из памяти контроллера.