

# Реализация алгоритма вычисления CRC-16 средствами Java2 для передачи данных по протоколу MODBUS режим RTU

Б.Б. Жмайлов

Представлена реализация расчета контрольной суммы CRC-16 для кадра протокола MODBUS режим RTU средствами Java 2. Выполнен анализ, имеющихся библиотек Java позволяющих осуществлять обмен данными по протоколу MODBUS. УДК 681.3.06

## 1. Введение

Протокол MODBUS обычно используется для осуществления взаимодействия между компьютером и всевозможными подчиненными устройствами. В качестве которых могут выступать электронные или электро- механические датчики, электрические счетчики, тепловычислители, имеющие в своем составе программируемый контроллер. Физически взаимосвязь может быть реализована через проводные телефонные линии связи, сети TCP/IP, GSM и радиоканалы связи, используя модемы, GSM-модемы, сетевые адаптеры соответствующих технологий (Ethernet, Token Ring и др.) или прямое подключение через стандартные интерфейсы RS-232, RS-485.

Соединение осуществляется, используя технологию главный-подчиненный, при которой только одно устройство (главный) может инициировать передачу (сделать запрос). Другие устройства (подчиненные) передают запрашиваемые главным устройством данные, или производят запрашиваемые действия.

## *Реализация алгоритма вычисления CRC-16 средствами Java2 для передачи данных по протоколу MODBUS режим RTU*

Главный может адресоваться к индивидуальному подчиненному или может инициировать широкую передачу сообщения на все подчиненные устройства. Подчиненное устройство возвращает сообщение в ответ на запрос, адресуемый именно ему. Ответы не возвращаются при широковещательном запросе от главного.

Передача данных по протоколу MODBUS может вестись в двух режимах: ASCII и RTU, которые имеют множество различий. В частности в режиме ASCII начало и конец кадра определяются с помощью специальных символов, в режиме RTU — определяется интервалом тишины между кадрами определенной длительности. Контрольная сумма кадра режима ASCII рассчитывается по алгоритму CRC 32, RTU — CRC-16.

## **2. Постановка задачи**

В настоящее время является актуальной задача удаленного сбора информации с датчиков учета тепла, горячей и холодной воды которые установлены как в жилом фонде, так и на производстве, которые используют для обмена протокол MODBUS.

В настоящий момент для реализации обмена по данному протоколу используются в основном OPC–сервера, и фирмы производители выпускают соответствующее программное обеспечение (ПО) для своих датчиков.

Технология связывания и внедрения объектов для систем промышленной автоматизации OPC (OLE for Process Control) предназначена для обеспечения универсального механизма обмена данными между датчиками, исполнительными механизмами, контроллерами, устройствами связи с объектом и системами представления технологической информации, оперативного диспетчерского управления, а также системами управления базами данных и предложена фирмой Microsoft. Это в свою очередь обуславливает ряд проблем связанных с кросс-платформенностью ПО и надежностью работы как прикладного, так и системного ПО и т.д.

В настоящее время, ведутся работы по созданию универсальной системы, с открыты кодом для удаленного сбора информации, на базе технологий Java 2 и СУБД MySQL. Одна из задач, решавшаяся в ходе разработки такой системы — это расчет контрольной суммы по алгоритму CRC-16 для кадра протокола MODBUS режим RTU.

### **3. Результаты эксперимента и их обсуждение**

Для организации взаимодействия через COM порт используется библиотека commAPI, которая не входит в стандартный пакет разработчика SDK, но вместе с тем любой желающий может абсолютно бесплатно скачать данную библиотеку с сайта фирмы Sun. Однако средствами этой библиотеки можно реализовать обмен только по протоколам канального уровня PPP, который так же используется во множестве устройств. Поэтому я вынужден был искать библиотеки других фирм, которые позволяют выполнять обмен по протоколу MODBUS. В результате довольно длительных поисков было найдено две бесплатных библиотеки jamod и jacksum, и одна коммерческая fieldtalk, австралийской фирмы FOCUS.

В результате дальнейшей работы было установлено, что хотя в документации и написано, что библиотека jamod поддерживает режим RTU, в исходных кодах соответствующих классов все методы оказались «заглушены». Библиотека jacksum почему-то на всех контрольных примерах выдавала не правильные результаты, как потом выяснилось в ней используется несколько иной подход.

Следующим этапом стало мое знакомство с коммерческой библиотекой fieldtalk. Данное ПО дают бесплатно попробовать в течении месяца, да и стоимость его относительно невысока, порядка 100 долларов. В результате длительной переписки с технической службой фирмы FOCUS выяснилось, что в приборах отечественных производителей, в частности фирмы ТЕПЛОКОМ, существуют незначительные отклонения от стандарта MODBUS, которые не поддерживает библиотека fieldtalk, поэтому я был вынужден от нее отказаться.

После всего этого было принято решение самостоятельно написать библиотеку класса для реализации вычисления контрольной суммы по алгоритму CRC 16, в паспорте прибора приведен код на языке Си.

```
WORD Crc16(BYTE *Data, ULONG size)

{ union {BYTE b[2]; unsigned short w;}

Sum; char shift_cnt; BYTE *ptrByte;
```

## Реализация алгоритма вычисления CRC-16 средствами Java2 для передачи данных по протоколу MODBUS режим RTU

```
ULONG byte_cnt = size;

for(; byte_cnt>0; byte_cnt--)

{
    Sum.w=(unsignedshort) ((Sum.w/256U)*256U+
        ((Sum.w%256U)^(*ptrByte++)));

    for(shift_cnt=0; shift_cnt<8; shift_cnt++)

        /*обработка байта */

        if (( Sum . w &0 x 1)==1)

            Sum.w=(unsigned short) ((Sum.w>>1)^0xa001U);

        else

            Sum.w>>=1;

    }

}

return Sum.w;

}
```

Однако простое приведение к синтаксису языка Java не дало ни каких результатов. В первую очередь возникла проблема с беззнаковыми типами — в Java их нет. Дело в том, что итоговый результат размером 2 байта (тип `short`) вставляется в кадр протокола младшим байтом вперед и при переполнении воспринимается как отрицательное число. Безусловно, что можно было данный алгоритм реализовать на Ассемблере, однако в этом случае потерялась бы кросс-платформенность. Проблема беззнакового числа была решена следующим образом, итоговая сумма объявляется как переменная тип `int` (4 байта) а в конце расчетов выполняется битовое умножение на число `0x0000ffffh`. Однако и после этого вычисления на контрольном примере не дали правильных результатов.

## Реализация алгоритма вычисления CRC-16 средствами Java2 для передачи данных по протоколу MODBUS режим RTU

В результате, на одном французском сайте <http://membres.lycos.fr/titicoco101/page6.html#ancre96826> был найден алгоритм расчета с примером на двух бинарных числах с промежуточными результатами после каждого шага, на основании которого был скорректирован исходный текст на языке Java. Так же был добавлен метод который «парсит» итоговую сумму и переставляет два младших байта с двумя старшими байтами местами. В результате алгоритм расчета CRC-16 на Java выглядит следующим образом:

```
TextPan tp;

    int Sum=(int)0xffff;

    byte [] arr;

    public crc16(byte[] dataBuffer) {

arr = dataBuffer;

for (int i=0; i<arr.length; i++ ){

    Sum = (Sum^arr[i]);

    System.out.println(Sum);

    for (int j=0; j<8; j++){

        if((Sum & 0x1) == 1)

        {

            Sum >>>=1;

            Sum = (Sum^0xA001);

        }

        else
```



*Реализация алгоритма вычисления CRC-16 средствами Java2 для передачи данных по протоколу MODBUS режим RTU*

Realization of algorithm of calculation CRC-16 by means Java2 for data transmission under report MODBUS mode RTU.

Realization of calculation of control sum CRC-16 for the frame of report MODBUS mode RTU by means Java 2 is submitted. The analysis is executed, available libraries Java allowing to carry out data exchange under report MODBUS.

Жмайлов Борис Борисович (р.1971), доцент кафедры «Информатика» ДГТУ, кандидат технических наук (1996). Окончил технологический факультет ДГТУ (1993).

Основные научные интересы: системный анализ, управление и обработка информации, разработка информационных систем средствами Java2 на базе СУБД MySQL.

Автор 30 научных публикаций.