

МОДИФИКАЦИЯ ДРАЙВЕРА ПРОТОКОЛА МЭК 60870-5-104

Филиал «Протвино» университета «Дубна»
Кафедра информационных технологий

В данной работе модифицирован драйвер протокола под управлением ПО «Монитор РВ», добавлена возможность подключения устройств-клиентов к серверу по конкретному порту.

«Монитор Реального Времени» — это управляющая программа центральной приемопередающей станции (ЦПС) «СИСТЕЛЬ», предназначенной для работы в составе автоматизированных систем диспетчерского управления (АСДУ) в качестве устройства сбора данных по последовательным каналам связи, каналам IP с использованием стандартных сетевых интерфейсов. Программа «Монитор» управляет работой ЦПС «СИСТЕЛЬ» и обеспечивает ее функциональность в соответствии с «Отраслевыми техническими требованиями к типовому комплексу центральной приемопередающей станции (ЦПС) для системного оператора» в частности:

- прием данных, вводимых по каналам связи от различных устройств;
- обработка данных в соответствии с конкретными телемеханическими протоколами;
- запись полученных данных в оперативную базу данных в памяти компьютера;
- синхронизация времени подключенных устройств телемеханики.

Функционально программу «Монитор» можно разделить на следующие составляющие:

- Модуль обслуживания устройств приема/передачи информации осуществляет запуск процессов, управляющих работой каналов ввода/вывода; обеспечивает унифицированный доступ к устройствам разных типов;
- Модуль обслуживания протоколов осуществляет запуск системных каналов, обработчиков протоколов информации и управляет их работой; унифицирует работу с разными типами протоколов;
- База оперативных данных представляет собой динамический массив в памяти компьютера, в котором содержится актуальная телемеханическая информация;
- Модуль обслуживания АРМ Телемеханика обеспечивает доступ к базе данных «Монитора» по запросам АРМ Телемеханика.

Используется язык программирования C++, среда разработки *Microsoft Visual Studio 6.0*. В качестве дополнительного программного обеспечения использовался пакет *Microsoft Office Visio 2007*. Работа с ПО «Монитор» проводилась под управлением таких операционных систем, как:

- *Windows Server 2003*
- *Windows XP*

Данная работа связана с модулем обслуживания устройств. Модуль *MwServer104Device* функционирует в качестве сервера в терминологии протокола *TCP/IP*. Он предоставляет «Монитору» набор стандартных функций работы с устройством. При загрузке параметров, которые находятся в таблицах баз данных, происходит настройка конфигурации устройства, осуществляется регистрация подключаемого устройства, сканирование таблиц каналов на предмет создания подключения и другие функции.

Главной задачей модуля является осуществление подключения к клиенту и поиск свободного канала. Схема работы модуля *MwServer104Device* представлена на рис. 1. При входе в главную функцию производится попытка создания подключения к клиенту по конкретному порту. При обнаружении такого клиента производится сканирование внутренних таблиц на наличие подключения к данному клиенту. Если подключения нет, то необходимые характеристики заносятся в таблицу каналов — в противном случае происходит повтор подключения.

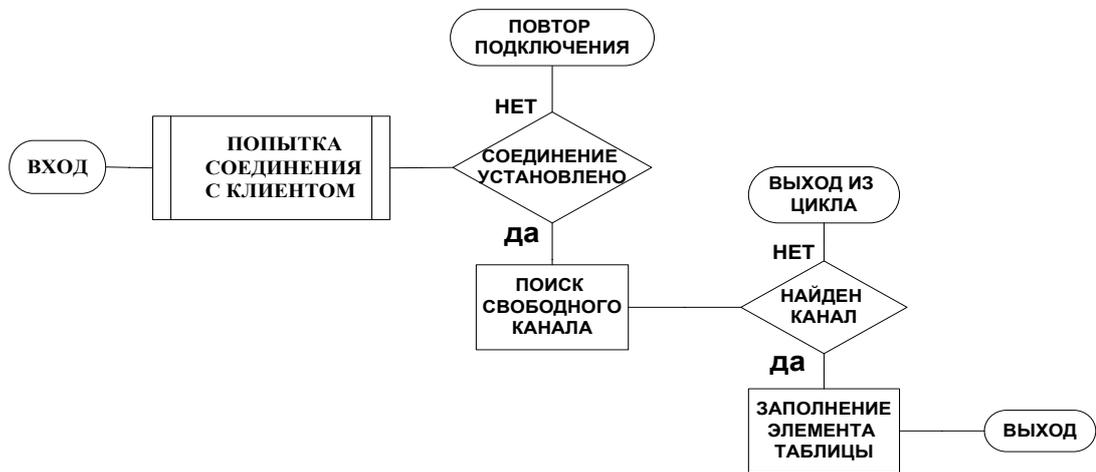


Рис. 1 Схема работы модуля *MwServer104Device*

По старой схеме работы подключение устройств производилось по одному порту для всех клиентов. На сервере был установлен конкретный порт и при подключении клиента по порту, отличному от установленного, связь не осуществлялась. Поставленная задача заключалась в создании для каждого канала структуры с индивидуальными параметрами, вследствие чего появилась возможность прослушивания клиентов по разным портам (рис. 2).

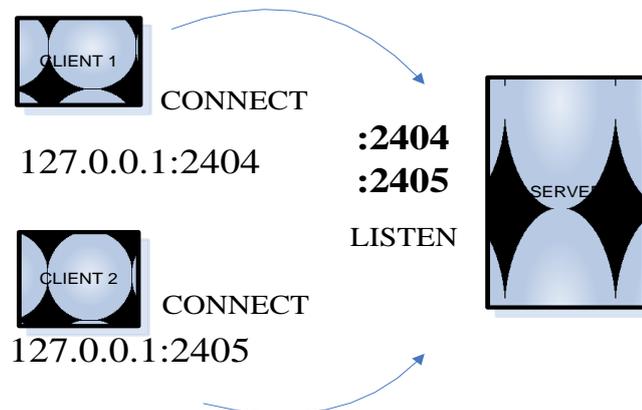


Рис. 2. Схема реализации поставленной задачи

Для решения поставленной задачи было внесено структурное решение. С каждым каналом этого подключаемого устройства связан список. Его элементами являются структуры, которые содержат все необходимые параметры. Данный список содержит не только информацию об *ip*-адресе клиента, но также номер порта и другие необходимые параметры (рис. 3).

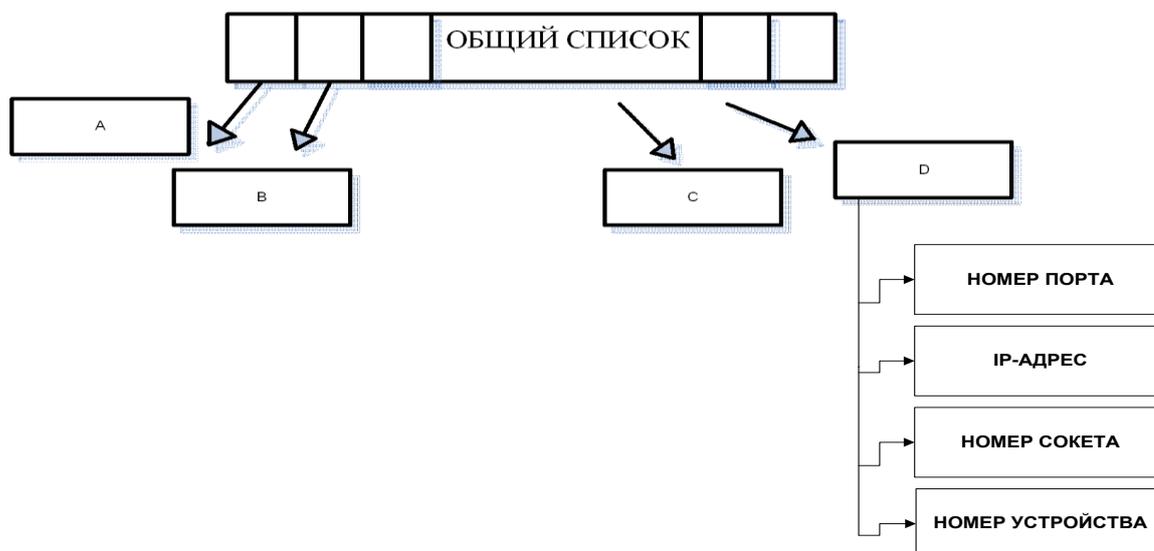


Рис. 3 Схема внесенной модификации

Программный код из рабочего класса выглядит следующим образом:

```

pMyStruct->pTcpLink = pLink104;
pMyStruct->Addr = Addr;
pMyStruct->Numbe = NumNul;
pMyStruct->StructPort = atoi(Port);
pMyStruct->IsConnect = FALSE;
pSockAddr->sin_family = AF_INET;
pSockAddr->sin_port = htons(pMyStruct->StructPort);
pSockAddr->sin_addr.s_addr = htonl(INADDR_ANY);
pMyStruct->pSockAddrServ = (sockaddr_in*) pSockAddr;
pMyStruct->ClientSocket = INVALID_SOCKET;

```

```
List104.push_back(pMyStruct);
```

Модифицированная программа была протестирована на стенде, состоящем из 20 клиентов и 1 сервера. При помощи программы *Memory Validator* осуществлялась проверка на наличие утечек памяти и обнаружения ошибок в программном коде. Были проанализированы полученные результаты.

В программный модуль *MwServer104Device* была добавлена возможность прослушивания каждого клиента по своему порту. Таким образом, были сняты ограничения, по которым подключение клиента к серверу осуществлялось только по одному установленному порту.

Библиографический список

1. Документация «Монитор реального времени системы сбора и первичной обработки телеинформации. Руководство системного программиста» — 149 с.
2. Шилдт Г. Самоучитель C++ / Г. Шилдт. — СПб. : BHV, 1997. — 512 с.
3. Лафоре Р. Объектно-ориентированное программирование в C++ / Р. Лафоре. — СПб : Питер, 2006. — 928 с.