

Чтение данных с сервера АСКУЭ с помощью элемента ActiveX fw10.osx

Руководство программиста

СОДЕРЖАНИЕ

СОДЕРЖАНИЕ	2
1. Методы элемента ActiveX fw10.ocx	3
2. Свойства элемента ActiveX fw10.ocx.....	6
3. Ошибки, возвращаемые элементом ActiveX fw10.ocx	6
4. Пример использования элемента ActiveX fw10.ocx	7

1. Методы элемента ActiveX fw10.ocx

Элемент ActiveX fw10.ocx реализует связь с сервером АСКУЭ (конкретно с библиотекой c12.dll, поэтому на сервере должна быть установлена и запущена данная библиотека (см. руководство пользователя по библиотеке c12.dll)).

Для организации обмена данными используются следующие методы:

LONG SetIP(BSTR IP, long Port) – используется, если связь будет осуществляться по UDP/IP. Вызывается перед методом **OpenPort**.

Параметры:

IP – строка с IP-адресом сервера (например “192.168.1.2”);

Port - номер порта (IP-порта) сервера (по умолчанию - 5150);

Возвращаемое значение:

В случае успеха метод возвращает 1.

При неудаче метод возвращает 0.

LONG OpenPort(LONG T, LONG N, LONG V) - открытие порта.

Параметры:

T – тип передачи (0 – по COM-порту; 1 – по UDP/IP);

N – номер COM-порта (IP-порта);

V – скорость передачи данных для COM-порта.*

Возвращаемое значение:

В случае успеха метод возвращает: 0 – открыт COM-порт или 1 – открыт IP-порт.

При неудаче метод возвращает код ошибки.

BSTR GetTime(void) - получить системное время с сервера.

Возвращаемое значение:

В случае успеха метод возвращает время в формате: чч:мм:сс.

При неудаче метод возвращает код ошибки.

FLOAT GetIzm(LONG Num, LONG Day, LONG Mon, LONG Year, LONG Hr, LONG Min) - получить из архива значение расчетного измерения.

Параметры:

Num – номер расчетного измерения;

Day – число месяца;

Mon – месяц;

Year – год;

Hr – час;

Min – минута.

Возвращаемое значение:

В случае успеха метод возвращает значение расчетного измерения.

При неудаче метод возвращает код ошибки.

FLOAT GetDataP(LONG Num, LONG DayB, LONG MonB, LONG YearB, LONG DayE, LONG MonE, LONG YearE, LONG VZ, LONG TD) - получить из архива данные за период по выбранному расчетному измерению за указанные временную зону и тип дня.

Параметры:

Num – номер расчетного измерения;

DayB – число месяца (начало периода);

MonB – месяц (начало периода);

YearB – год (начало периода);

DayE – число месяца (конец периода);

MonE – месяц (конец периода);

YearE – год (конец периода);

VZ – номер временной зоны;

TD – тип дня:

1 – рабочий день;

2 – суббота;

3 – воскресенье;

4 – праздничный день;

0 – тип дня не определен.

Номер временной зоны и тип дня задаются в конфигурации c12.dll.

Возвращаемое значение:

В случае успеха метод возвращает значение расчетного измерения.

При неудаче метод возвращает код ошибки.

FLOAT GetDataZ(LONG Num, LONG Day, LONG Mon, LONG Year, LONG VZ, LONG TD) - получить из архива данные за сутки по выбранному расчетному измерению за указанные временную зону и тип дня.

Параметры:

Num – номер расчетного измерения;

Day – число месяца;

Mon – месяц;

Year – год;

VZ – номер временной зоны;

TD – тип дня.

Возвращаемое значение:

В случае успеха метод возвращает значение расчетного измерения.

При неудаче метод возвращает код ошибки.

LONG GetArcPos(void) - получить текущую позицию в архиве событий.

Возвращаемое значение:

В случае успеха метод возвращает текущую позицию в архиве событий.

При неудаче метод возвращает код ошибки.

BSTR GetArc(LONG P) - получить строку из архива событий.

Параметры:

P – позиция в архиве событий.

Возвращаемое значение:

В случае успеха метод возвращает строку из архива событий.

При неудаче метод возвращает код ошибки.

BSTR GetNameRI(LONG Num) - получить название расчетного измерения.

Параметры:

Num – номер расчетного измерения.

Возвращаемое значение:

В случае успеха метод возвращает название расчетного измерения.

При неудаче метод возвращает строку **!@#%\$**.

LONG SetConnect(LONG N) - получение данных по мгновенным значениям и архивам счетчика.

Параметры:

N – число, которое задает параметры запроса, вычисляется по следующей формуле:

$$N=(O*256+L)*1000+NO,$$

Где

O – опция запроса архивов счетчика:

Бит 0 – флаг запроса архива фаз (1 – запросить, 0 – не запрашивать);

Бит 1 – флаг запроса архива состояния (1 – запросить, 0 – не запрашивать);

Бит 2 – флаг запроса архива корректировок (1 – запросить, 0 – не запрашивать);

Остальные биты должны быть равны 0.

L – связной номер счетчика (должен быть в диапазоне от 0 до 255);

NO – номер объекта на сервере АСКУЭ, который опросит счетчик с заданным связным номером.

LONG ClosePort(LONG T, LONG N) - закрытие порта.

Параметры:

T – тип передачи (0 – по COM-порту; 1 – по UDP/IP);

N – номер COM-порта (IP-порта).

2. Свойства элемента ActiveX fw10.ocx

unsigned char Debug – включение отладочного режима.

1- отладочный режим включен; 0- отладочный режим выключен;

short T_Sec – таймаут (в сек.) ожидания ответа сервера

long Prizm – признак достоверности возвращаемого значения

32 – достоверное значение(код символа пробел)

33 – ручной ввод (код символа !)

35 – возвращенное значение является кодом ошибки (код символа #)

42 – превышение лимитов (код символа *)

63 – недостоверное значение (код символа ?)

94 – нет данных (код символа ^)

Используется совместно с GetIzm, GetDataP, GetDataZ

float MaxP – максимальное значение за выбранный период

Используется совместно с GetDataP, GetDataZ

long HalfHour – номер получаса, соответствующего максимальному значению

Используется совместно с MaxP.

short ExtrDay – день месяца, соответствующего максимальному значению

Используется совместно с MaxP.

short CN – точность округления возвращаемого значения

Наименование типов данных дано в соответствии с синтаксисом языка C/C++.

3. Ошибки, возвращаемые элементом ActiveX fw10.ocx

ActiveX возвращает следующие коды ошибок:

- собственные

0 – ошибка открытия порта

2 – доступ запрещен

3 – закончилось время ожидания

- системные, возвращаемые функцией GetLastError ()

Код	Описание	Имя
10004	A blocking operation was interrupted by a call to WSACancelBlockingCall.	WSAEINTR
10022	An invalid argument was supplied.	WSAEINVAL
10038	An operation was attempted on something that is not a socket.	WSAENOTSOCK
10040	A message sent on a datagram socket was larger than the internal message buffer or some other network limit, or the buffer used to receive a datagram into was smaller than the datagram itself.	WSAEMSGSIZE
10049	The requested address is not valid in its context.	WSAEADDRNOTAVAIL
10054	An existing connection was forcibly closed by the remote host.	WSAECONNRESET
10065	A socket operation was attempted to an unreachable host.	WSAEHOSTUNREACH
10093	Either the application has not called WSASStartup, or WSASStartup failed.	WSANOTINITIALISED

4. Пример использования элемента ActiveX fw10.ocx

Для того, чтобы использовать ActiveX, он должен быть зарегистрирован в системе (командой regsvr32).

После этого, необходимо создать проект в используемой среде программирования и включить в него ActiveX fw10.ocx. Далее будет использоваться среда программирования Microsoft Visual Studio .NET и язык программирования C++. В этой среде создаем проект MFC, основанный на диалоге (dialog-based), и при помощи Wizard добавляем в проект элемент ActiveX fw10(рис .1,2).

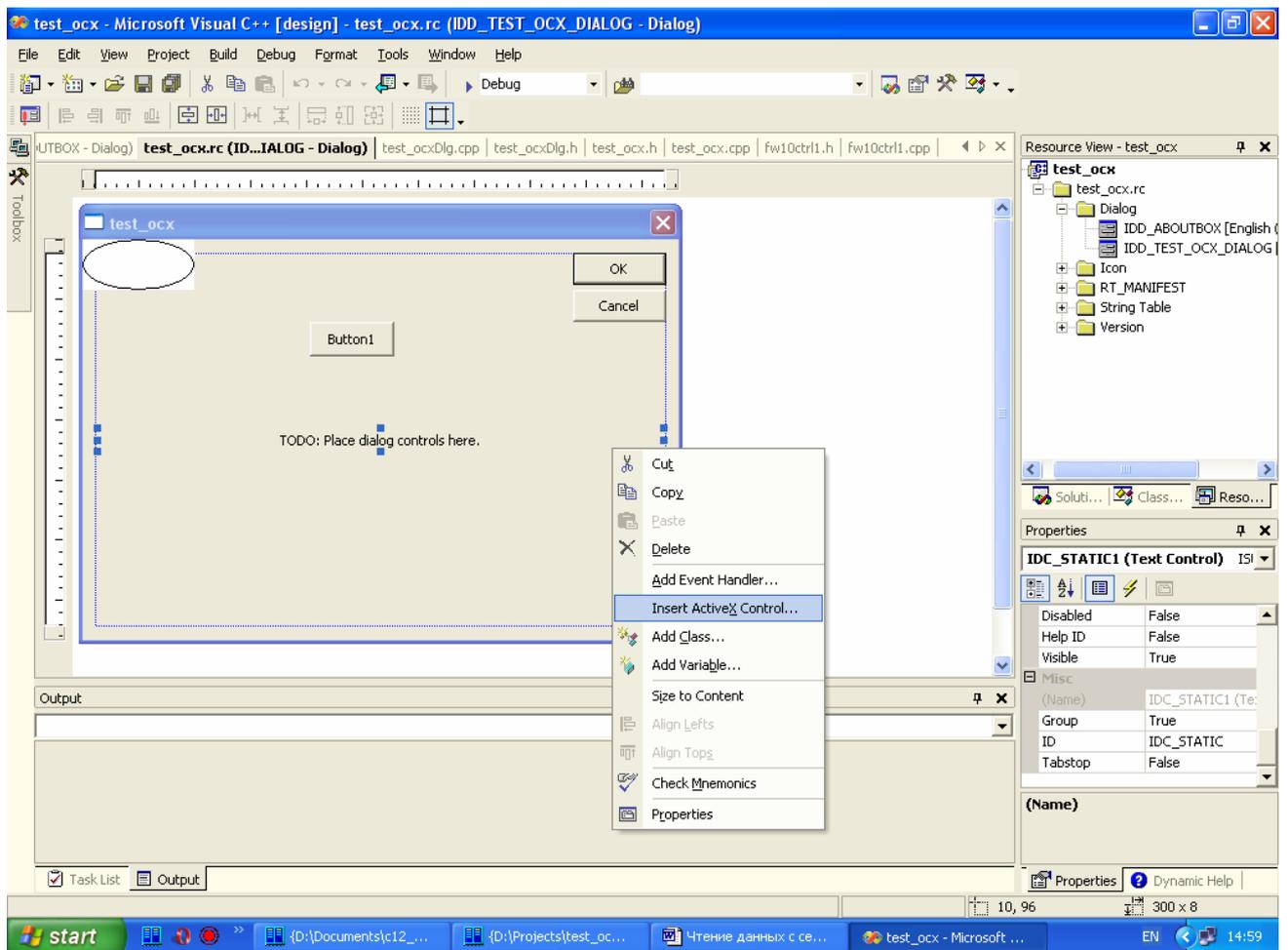


Рис.1. Вызов Мастера (Wizard) для добавления ActiveX

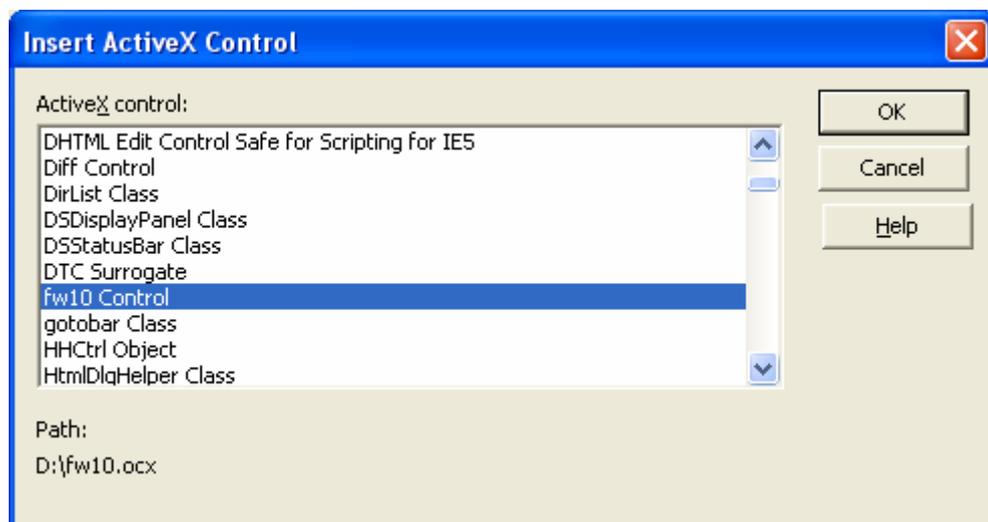


Рис.2. Выбор добавляемого ActiveX

После этих действий элемент добавлен в проект. Для его использования необходимо добавить переменную, с помощью которой будем вызывать методы ActiveX fw10.ocx. (рис. 3,4).

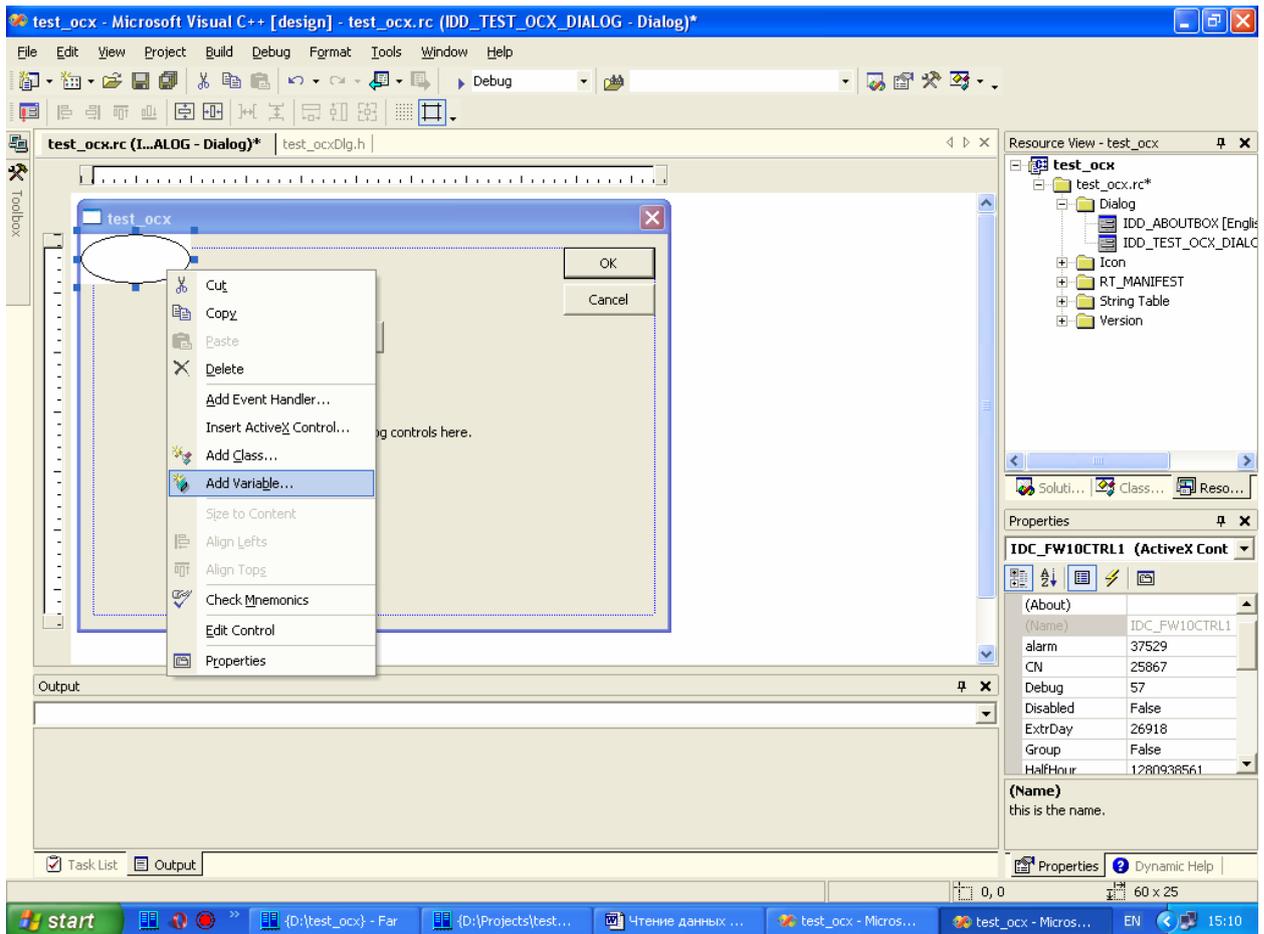


Рис. 3. Вызов Мастера (Wizard) для добавления переменной

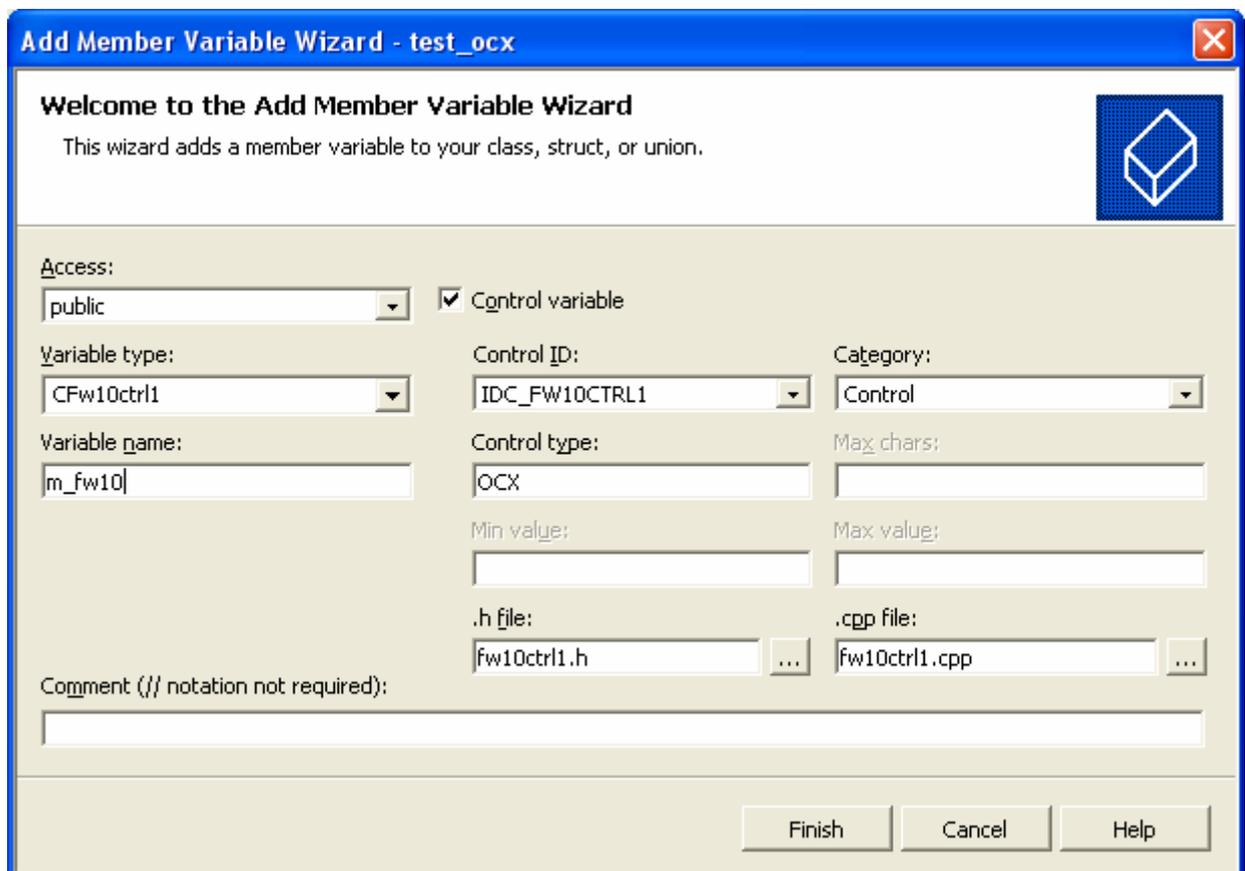


Рис.4. Добавление переменной для обращения к ActiveX

После добавления переменной в проект будет добавлена пара файлов fw10ctrl1.h и fw10ctrl1.cpp, в которых содержится описание методов и свойств ActiveX (представленных в виде класса MFC), а в проект добавится переменная m_fw10 для доступа к классу ActiveX. В дальнейшем обращение к методам и классам осуществляется с помощью операции «.».

Пример работы с ActiveX:

```

m_fw10.SetIP("192.168.1.2",5150); //установка IP-адреса и порта сервера, с
//которым будет связываться приложение
long res=m_fw10.OpenPort(1,1,9600);
if(res!=1)
{
    printf("Ошибка открытия порта\n")
}

long num=(6*256+5)*1000+2; //Открытие порта UDP/IP, при 1 параметре
m_fw10.SetConnect(num); //=1, два последующих не имеют значения
//запрос мгновенных значений и архивов
//состояния и корректировок для счетчика
//со связным номером 5, который подключен
//к объекту сервера с номером 2
//после выполнения запроса сервер
//записывает данные в БД
float data=m_fw10.GetIzm(500,10,12,2006,11,30);
//получение значения из расчетного
//измерения из базы измерений сервера номер
//500 за 10 декабря 2006 года 11:30
long pr=m_fw10.Prizm; //получение признака данных, полученных
//предыдущей командой
float d= m_fw10.GetDataP(200,1,11,2006,30,11,2006,1,1);
//получение значения из расчетного
//измерения из базы измерений сервера номер
//200 за период с 1 по 30 ноября 2006 года по
//временной зоне 1 за рабочие дни
//после успешного выполнения команды
//доступны свойства ActiveX
float max=m_fw10.MaxP; //получение значения максимальной
//мощности за указанный период после
//выполнения команды GetDataP
long half=m_fw10.HalfHour; //получение номер получаса за который было
//максимальное значение после
//выполнения команды GetDataP
short D=m_fw10.ExtrDay; //получение дня месяца за который было
//максимальное значение после
//выполнения команды GetDataP
m_fw10.ClosePort(1,1); //Закрытие порта

```

Для заметок



Республика Беларусь
220141, г. Минск, ул. Ф.Скорины, 54а
Приёмная: тел./факс: (017) 265-82-03
Отдел сбыта: тел. (017) 265-81-87, 265-81-89
Отдел сервиса: тел.: (017) 265 82 09
E-mail: info@strumen.com
<http://www.strumen.com>

Представительства:

г. Брест, тел. (0162) 42-71-06
г. Витебск, тел. (0212) 24-08-43
г. Гомель, тел. (0232) 48-92-03
г. Гродно, тел. (0152) 79-26-70
г. Могилев, тел. (0222) 28-50-47