



Описание программы для контроллера АМХ.

Получение прогноза погоды для выбранного
города.

Содержание

История изменений.....	2
Введение	3
Общее описание	3
Использование.....	3
Пример использования.....	4
Запросы на получение прогноза погоды.....	6
Сообщения об ошибках	7

История изменений

Дата	Комментарии
Февраль 2006	Разработка модуля и тестирование
Март 2006	Подготовка документации, создание серийного образца
Апрель 2006	Добавлена функция разбора строки, передаваемой от модуля в структуру пользователя

Введение

В данном техническом руководстве содержится описание программы для контроллера АМХ. Целью работы программы является получение прогноза погоды на четыре дня вперед..

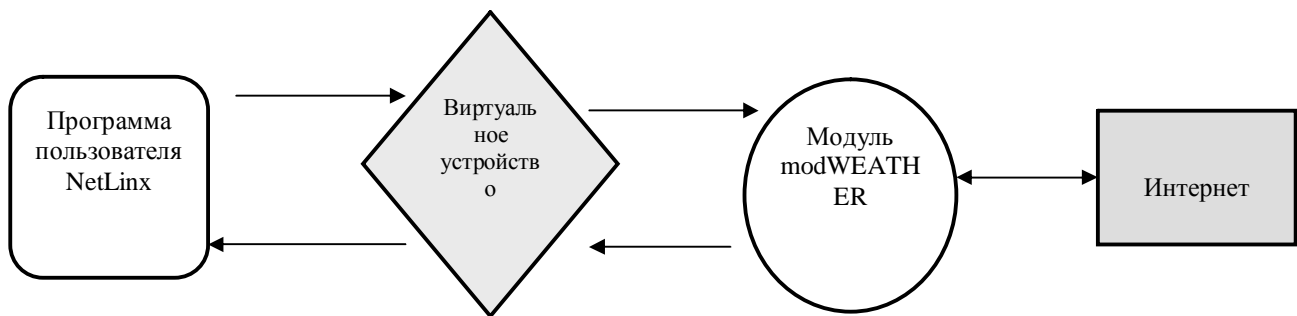
Программа для контроллера АМХ может работать как самостоятельно, так и в составе общего программного обеспечения. С этой целью, исполняемая часть оформлена в виде отдельного модуля легко адаптируемого в общее программное обеспечение.

Для подключения модуля необходимы начальные знания в программировании контроллеров АМХ.

Для обеспечения нормальной работы, локальная сеть составной частью которой является процессор АМХ, должна быть настроена инсталлятором на беспрепятственное прохождения запросов с сервера АМХ в Интернет.

Общее описание

Модуль modWEATHER обеспечивает получение данных о прогнозе погоды для города Москва (можно установить любой другой город мира) на текущую дату (дату, установленную в контроллере АМХ).



Модуль modWEATHER является буфером между командами, поступающими из программы пользователя NetLinx для получения прогноза погоды и строками данных, описывающих полученный прогноз погоды, получаемый по сети интернет.

Интерфейс для получения курсов валют доступен пользователю только в виде модуля modWEATHER.tko.

Модуль поддерживает возможность получения прогноза погоды как в определенное время, так и по специальному запросу.

Использование

Для использования modWEATHER.tko модуля, программист должен выполнить следующие шаги:

1. Адрес клиента NetLinx для связи с сервером "Прогноз погоды". Адреса таких устройств в NetLinx имеют вид 0:xxx:0.
2. Определить адрес виртуального устройства для обмена данными с модулем modWEATHER. Адреса таких устройств в NetLinx начинаются с 33001.
3. Определить адрес сенсорной панели, при нажатии на клавиши которой будет выдаваться запрос на получение прогнозов погоды (**не является обязательным**).
4. Объявить модуль modWEATHER.tko в программе NetLinx при помощи оператора DEFINE_MODULE.

Пример использования.

DEFINE_DEVICE

```
dvIPWeather      = 0:4:0      // Адрес клиента NetLinx для связи с сервером "Прогноз
погоды"
vdvWeather       = 33002:1:0  // Виртуальное устройства для обмена данными с модулем
modWEATHER
dvPANEL          = 10001:1:0  // Адрес сенсорной панели
```

DEFINE_CONSTANT

```
GETWeather = 2 // Код клавиши при нажатии на которую нужно получить прогноз
погоды
```

DEFINE_TYPE

```
STRUCTURE typeWEATHER {
  CHAR wPOINT[20] // Город
  CHAR wDATE[20]  // Дата прогноза погоды
  CHAR wCOVER[3]  // Облачность в процентах
  CHAR wPRESS[3]  // Давление в мм.рт.столба
  CHAR wTEMP[3]   // Температура в градусах по Цельсию
  CHAR wHUM[3]    // Относительная влажность в процентах
  CHAR wWDIR[5]   // Направление ветра
  CHAR wWVEL[3]   // Скорость ветра
  CHAR wFALLS[1]  // Тип осадков (0 - отсутствие осадков, 1 - дождь, 2 - дождь со снегом,
3 - снег)
}
```

DEFINE_VARIABLE

```
typeWEATHER curWEATHER[4] // Прогноз погоды
CHAR WeatherTIME[8] = '16:30:00' // Время ежедневного получения прогноза погоды.
НАПРИМЕР в 16:30:00.
INTEGER tmpI // Переменная цикла
```

DEFINE_START

```

DEFINE_MODULE 'modWEATHER' WEATHER (dvIPWeather, vdvWeather, WeatherTIME)

DEFINE_EVENT
// Получение прогноза погоды при нажатии соответствующей кнопки на панели управления
BUTTON_EVENT[dvPanel,GETWeather] { // На панели нажата кнопка для получения
прогноза погоды
    PUSH: {
        SEND_COMMAND vdvWeather,"GET Weather" // Пошлaем команду модулю на
получение прогноза погоды
    }
}
// Получение прогноза погоды ежедневно в установленное время (переменная WeatherTIME)
DATA_EVENT[vdvWeather] { // Ждем ответа от модуля
    STRING: { // Получена строка
        SELECT {
            ACTIVE (LEFT_STRING(DATA.TEXT,8) == 'Weather='){ // Получен прогноз погоды
                GET_BUFFER_STRING(DATA.TEXT,8) // Убираем текст в начале
                ('Weather=')
                STRING_TO_VARIABLE(curWEATHER,DATA.TEXT,1) // Оставшуюся строку
преобразуем в структуру
// НАПРИМЕР. Выводим информацию в окно "Diagnostics" NetLinx Studio (только для примера)
                SEND_COMMAND 0,"SITY=',curWEATHER[1].wPOINT"
                FOR(tmpI=1;tmpI<=4;tmpI++) {
                    SEND_COMMAND 0,"DATE=',curWEATHER[tmpI].wDATE"
                    SEND_COMMAND 0,"COVER=',curWEATHER[tmpI].wCOVER"
                    SEND_COMMAND 0,"PRESS=',curWEATHER[tmpI].wPRESS"
                    SEND_COMMAND 0,"TEMP=',curWEATHER[tmpI].wTEMP"
                    SEND_COMMAND 0,"HUM=',curWEATHER[tmpI].wHUM"
                    SEND_COMMAND 0,"VEL=',curWEATHER[tmpI].wWVEL"
                    SEND_COMMAND 0,"FALLS=',curWEATHER[tmpI].wFALLS"
                }
            }
        }
    }
}

```

Запросы на получение прогноза погоды.

Получение прогноза погоды осуществляется двумя способами:

1. Ежедневно в установленное время.
2. В любое время по запросу.

Для получения прогноза погоды в установленное время, необходимо указать нужное время в описании модуля modWEATHER в строке

DEFINE_MODULE 'modWEATHER' WEATHER (dvIPWeather, vdvWeather, WeatherTIME).

В качестве третьего параметра указать время в формате NetLinx. "hh:mm:ss" (24 часовой формат).

Если этот параметр указан правильно, то в это время модуль будет генерировать событие

DATA_EVENT (STRING) ([см. пример выше](#)).

Информация о полученных курсах валют будет находиться в структуре типа с типом **typeWEATHER** ([см. пример выше](#)).

curWEATHER[1].wPOINT	- Город
curWEATHER[1..4].wDATE	- Дата прогноза погоды
curWEATHER[1..4].wCOVER	- Облачность в процентах
curWEATHER[1..4].wPRESS	- Давление в мм.рт.столба
curWEATHER[1..4].wTEMP	- Температура в градусах по Цельсию
curWEATHER[1..4].wHUM	- Относительная влажность в процентах
curWEATHER[1..4].wWVEL	- Направление ветра
curWEATHER[1..4].wFALLS	- Тип осадков (0 - отсутствие осадков, 1 - дождь, 2 - дождь со снегом, 3 - снег)

Обратите внимание:

1. Все данные уже представлены в текстовом виде.
2. Информация представлена в виде массива структур. Размерность массива равно 4 (по количеству дней на которые представлен прогноз погоды).

Для получения курсов валют в любое время по запросу, необходимо отправить команду виртуальному устройству vdvWeather ([см. пример выше](#)).

SEND_COMMAND vdvWeather, "GET Weather"

После получения прогноза погоды будет сгенерировано событие **DATA_EVENT (STRING)** ([см. пример выше](#)). Прогноз погоды будет представлен на 4 дня вперед.

Информация о полученном прогнозе погоды будет находиться в массиве структур типа **typeWEATHER** ([см. пример выше](#)).

Обратите внимание: все данные уже представлены в текстовом виде.

Сообщения об ошибках

Сообщения об ошибках выводятся командой SEND_STRING на устройство 0 (диагностика).

Модуль modWEATHER.tko анализирует состояние подключения к сети TCP/IP стандартными средствами NetLinx. Сообщения об ошибках тоже стандартны:

Номер ошибки	Описание
2	General failure (out of memory)
4	Unknown host
6	Connection refused
7	Connection timed out
8	Unknown connection error
14	Local port already used
16	Too many open sockets
17	Local Port Not Open

Таблица 1 – Сообщения об ошибках