



Описание программы для контроллера АМХ.

Получение информации о температуре и  
влажности.



## Содержание

История изменений.....	2
Введение .....	3
Общее описание .....	3
Использование.....	4
Пример использования.....	4
Запросы на получение данных.....	6
Сообщения об ошибках .....	8
Подключение датчиков температуры и влажности.....	8

### История изменений

Дата	Комментарии
Февраль 2008	Разработка модуля и тестирование
Март 2009	Подготовка документации
Январь 2011	Использование стандартного протокола ModBus для опроса датчиков
Май 2011	Добавлен код ошибки (№ 251) при ошибке опроса датчика (стр. 8)
Октябрь 2011	Добавлена команда для изменения интервала опроса датчиков.

## **Введение**

В данном техническом руководстве содержится описание программы для контроллера АМХ. Целью работы программы является получение данных о температуре и влажности с датчиков, подключенных к контроллеру АМХ.

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.

Программа предназначена для опроса состояния датчиков ТН-2 (Evolution) по протоколу ModBus. Программа не требует от программиста знания протокола ModBus.

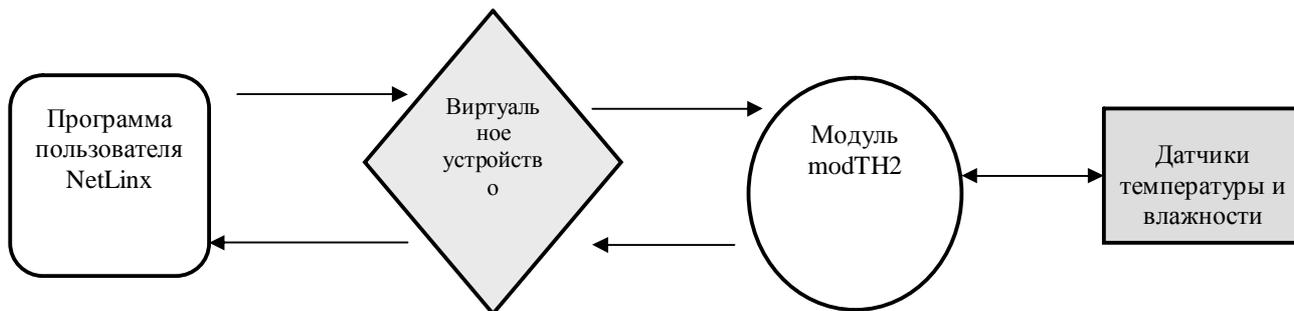
Для подключения модуля необходимы начальные знания в программировании контроллеров АМХ.

Обратите внимание на другие модули компании РеалДом для контроллеров АМХ:

1. [АМХ. Время восхода и заката солнца](#) – бесплатно.
2. [АМХ. Прогноз погоды](#) – 5000 р.
3. [АМХ. Курсы валют](#) – 3000 р.

## **Общее описание**

Модуль modТН2 обеспечивает получение данных с установленных датчиков температуры и влажности, подключенных к 232/485 порту контроллера АМХ.



Модуль modТН2 является буфером между командами, поступающими из программы пользователя и датчиками температуры, работающими по протоколу ModBus.

Максимальное количество датчиков, подключаемых к одному порту контроллера АМХ – 255.

Модуль поддерживает возможность получения температуры и влажности с каждого установленного датчика как периодически, так и в любое время по запросу пользователя (программы пользователя).

Настройка параметров порта 232/485 производится автоматически модулем modTH2.

## **Использование**

Для использования modTH2.tko модуля, программист должен выполнить следующие шаги:

1. Определить адрес виртуального устройства для обмена данными с модулем modTH2. Адреса таких устройств в NetLinx начинаются с 33001.
2. Определить адрес сенсорной панели, при нажатии на клавиши которой будет выдаваться запрос на получение информации с датчиков (**не является обязательным**).
3. Определить порт процессора AMX (232/485) к которому подключены датчики температуры.
4. Объявить модуль modTH2.tko в программе NetLinx при помощи оператора DEFINE\_MODULE.
5. Перечислить адреса установленных датчиков температуры в отдельном массиве типа INTEGER.
6. Задать интервал опроса датчиков, перечисленных в массиве.

## **Пример использования.**

```
PROGRAM_NAME='123'  
DEFINE_DEVICE  
dvTH = 5001:1:0           // 232/485 порт на процессоре AMX, к котрому подключены  
датчики температуры и влажности  
vdvTH = 33001:1:0        // Виртуальное устройство для связи с модулем обработки  
информации от датчиков  
dvPANEL = 10001:1:0      // Панель управления  
  
DEFINE_CONSTANT  
#INCLUDE 'incTH'         // Описание структуры для значений датчика  
  
DEFINE_VARIABLE  
TH_ANSWER_Type TH_ANSWER // Структура, содержащая значения температуры и  
влажности для текущего датчика  
INTEGER TH_LIST[] = {179, 201, 172, 168, 165} // Массив адресов датчиков температуры  
INTEGER TH_DELAY = 10    // Интервал опроса датчиков в секундах  
  
DEFINE_START  
DEFINE_MODULE 'modTH2' modTEMP(dvTH, vdvTH, TH_LIST, TH_DELAY) // Модуль  
работы с датчиками температуры  
  
DEFINE_EVENT  
BUTTON_EVENT[dvPANEL,1] { // При нажатии на клавишу с номером канала '1' - опросить  
значения датчика  
  PUSH: {  
    SEND_COMMAND vdvTH, "TH?=179" // Опросить датчик с номером 179
```

```

}
}
DATA_EVENT [vdvTH] // Обработка событий от датчиков температуры
{
  STRING: {
    STACK_VAR CHAR tmpCOMMAND[TH_MAXLEN_STR]
    STACK_VAR CHAR tmpPARAM[TH_MAXLEN_STR]

    tmpPARAM = DATA.TEXT
    tmpCOMMAND = REMOVE_STRING (tmpPARAM,'TH=',1) // Разбираем
    полученную строку
    SWITCH (tmpCOMMAND) {
      CASE 'TH=': { // Если строка начинается с 'TH='
        STRING_TO_VARIABLE(TH_ANSWER,tmpPARAM,1) // Преобразуем
        строку в переменную
        SEND_STRING
        0,"TH.ADDR=',ITOA(TH_ANSWER.ADDR),'TH.ERROR=',ITOA(TH_ANSWER.ERROR),'TH.
        T=',FTOA(TH_ANSWER.T),'TH.H=',FTOA(TH_ANSWER.H)"
        }
        DEFAULT: { // Любая другая строка - ошибка
          SEND_STRING 0,"ERROR STR FROM modTH=',tmpCOMMAND,tmpPARAM"
        }
      }
    }
  }
}
DEFINE_PROGRAM

```

## Запросы на получение данных.

Получение данных о температуре и влажности возможно двумя способами:

1. Периодически с указанным интервалом.
2. В любое время по запросу.

Для периодического получения данных, необходимо указать интервал опроса датчиков (в секунду) и передать это значение модулю modTH2 (в примере выше это переменная [TH\\_DELAY](#)).

**DEFINE\_MODULE 'modTH2' modTEMP(dvTH, vdvTH, TH\_LIST, TH\_DELAY) // Модуль работы с датчиками температуры**

```
dvTH // 232/485 порт на процессоре АМХ, к котрому подключены датчики температуры и влажности
v dvTH // Виртуальное устройство для связи с модулем обработки информации от датчиков
INTEGER TH_LIST[] // Массив адресов датчиков температуры
INTEGER TH_DELAY // Интервал опроса датчиков в секундах
```

Модуль будет опрашивать датчики, адреса которых перечислены в массиве TH\_LIST с интервалом значения переменной TH\_DELAY.

Например, если в массиве перечислены адреса пяти датчиков температуры, а интервал опроса равен 60 секундам (одной минуте), то каждый датчик температуры будет опрашиваться раз в 5 минут. Если интервал опроса поставить равным 30 секундам, то каждый датчик температуры будет опрашиваться с интервалом в  $30 * 5 = 150$  секунд.

Информация о температуре и влажности будет находиться в структуре типа с типом [TH\\_ANSWER\\_Type](#) (см. пример выше).

```
DEFINE_CONSTANT
TH_MAXLEN_STR = 255
DEFINE_TYPE
STRUCTURE TH_ANSWER_Type {
    INTEGER ADDR // Адрес устройства
    INTEGER ERROR // Код ошибки (0 – ошибки нет, 4 – такого датчика нет).
    FLOAT T // Температура
    FLOAT H // Влажность
}
```

Для получения значения температуры и влажности в любое время по запросу, необходимо отправить команду виртуальному устройству vdvTH ([см. пример выше](#)).

**SEND\_COMMAND vdvTH, " TH?=<номер датчика>"** (Например `send_command vdvTH, "TH?=172"`)

После получения информации от датчика будет сгенерировано событие **DATA\_EVENT (STRING)** ([см. пример выше](#)).

Информация о температуре и влажности будет находиться в массиве структур типа **TH\_ANSWER\_Type** ([см. пример выше](#)).

Для изменения интервала опроса датчиков, необходимо отправить команду:

**SEND\_COMMAND** vdvTH," TH\_DELAY=<время опроса в секундах>" (Например `send_command vdvTH," TH_DELAY=10"`) (Установить время интервала последовательного опроса датчиков равным 10 секунд).

## **Сообщения об ошибках**

Сообщения об ошибках находятся в структуре с типом `TH_ANSWER_Type`, переменная `ERROR`.

Значение этой переменной должно быть равно 0 (нулю) – это значит, что устройство и сеть работает нормально.

Если значение переменной равно 251, то это означает, что при опросе этого датчика он не вернул значения о температуре и влажности (обрыв цепи, датчик с таким номером не существует).

Другие значения служат в диагностических целях и зарезервированы для дальнейшего использования (ввод поправочных коэффициентов при измерении значений температуры и влажности в зависимости от места установки датчика / параметров окружающей среды).

## **Подключение датчиков температуры и влажности**

Датчики TH-2 (Evolution) подключаются к 232/485 порту контроллера АМХ.

Настройка параметров порта происходит автоматически и не требует вмешательства пользователя.

К одному порту может быть подключено до 255 датчиков. Схема подключения датчиков может быть разной - звезда, последовательное соединение, комбинированная.

Для питания датчиков необходимо постоянное напряжение 12 вольт. Потребление датчика не превышает 5мА.

Для подсоединения используется кабель типа «витая пара».



**Рисунок 1. Вид датчика без корпуса.**

К разъему с номером № 1 подключается «плюс» от источника питания.

К разъему с номером № 2 подключается сигнал А 485 порта.

К разъему с номером № 3 подключается сигнал Б 485 порта.

К разъему с номером № 4 подключается «минус» от источника питания.

Схема подключения к RS-485 контроллера AMX (Рисунок из документации на контроллер AMX - NI-X100.HardwareReferenceGuide).

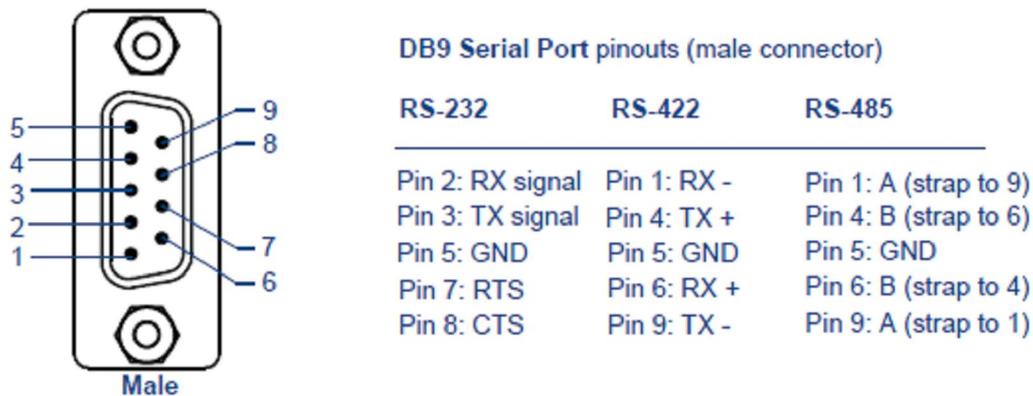


Рисунок 2. Схема подключения RS-232/422/485 к контроллеру AMX.