

**Обновление программного обеспечения  
для «4-х канального микропроцессорного таймера, термостата, часов»  
NM8036/VM8036  
/до версии 1.9/**

Приводится описание обновлённой версии программного обеспечения для **«4-х канального микропроцессорного таймера, термостата, часов»** NM8036, обеспечивающего существенное расширение функциональных возможностей устройства.

Бесплатно скачав эту версию, пользователь сможет улучшить технические характеристики устройства и более успешно применить его в системе «Умный дом» начального уровня в качестве 4-х канального устройства управления бытовыми приборами. Предлагаемое для сборки из набора **NM8036** (или в виде блока **VM8036**) устройство может быть использовано в качестве центральной части системы управления отоплением, охлаждением, вентиляцией, для автоматического полива огорода, для включения-выключения различных электробытовых приборов по программе пользователя.

Устройство также найдёт применение в зоологических кружках, у аквариумистов.

Количество шагов программирования 32.

Обновлена прошивка контроллера и ПО для ПК.

**Версия 1.1a****Введены следующие изменения в микропрограмме контроллера**

1. Активированы дополнительные 4 канала управления (5-8) на свободные выходы микроконтроллера PA4-PA7.
2. Количество подключаемых датчиков увеличено до 8.
3. Добавлено расширенное управление термостатом от персонального компьютера.
4. Введена функция проверки защиты информации при считывании температуры с датчиков (CRC8), что повышает надежность показаний температуры и исключает ложные срабатывания.
5. Активирован режим подключения датчиков по 2-х проводной линии: 1 и 3 выводы датчика, а также вывод «общий» замыкаются вместе. Вывод данных с термодатчика сохраняется, как и прежде.

**Изменения программного обеспечения для ПК**

Программа для ПК сильно видоизменена и представлена в виде основного рабочего окна на следующем рисунке 1.

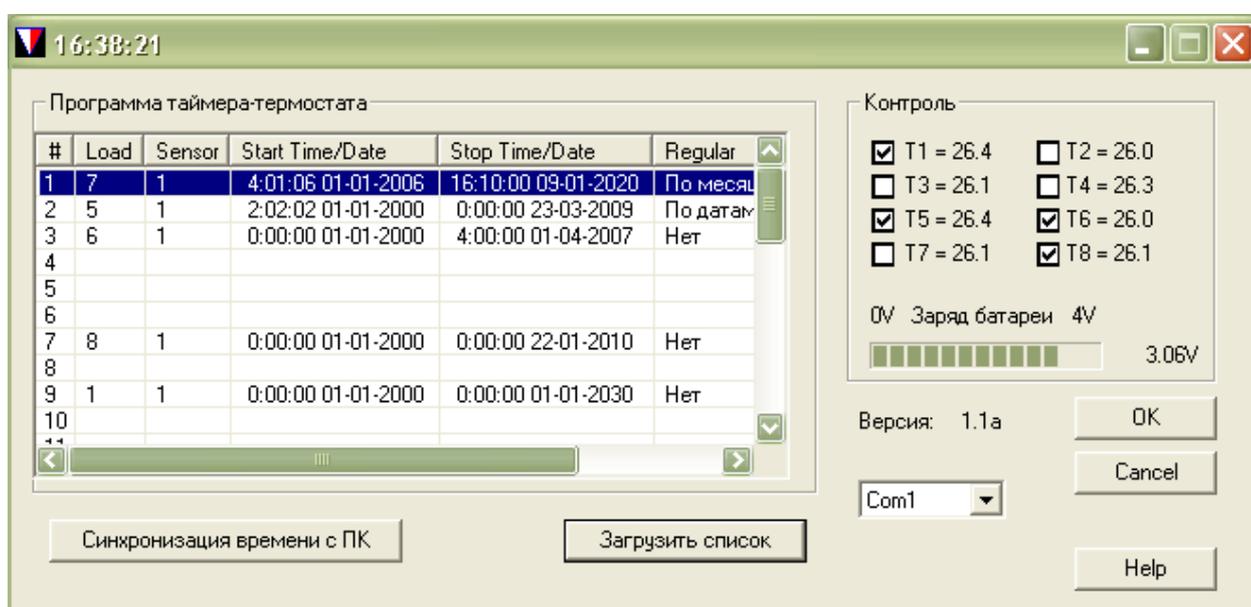


Рисунок 1

Справа осталось поле с температурами и состояниями нагрузок. Слева внизу появилась кнопка синхронизации часов компьютера и часов термостата. Кнопкой «Загрузить список» мы активируем загрузку 32-шаговой программы контроллера в компьютер.

Введена возможность редактирования каждой записи таймера. Запускается редактирование двойным щелчком мыши на соответствующей записи. После чего появляется окно редактирования, представленное на рисунке 2.

Рисунок 2

В данном окне можно активировать/деактивировать выбранный канал. Можно выбирать номер канала управления. Есть календари для выставления дат и времени включения и отключения. Есть поле для выставления периодичности и управление периодичностью (возможность выбора конкретных чисел, дней недели и месяцев). Кроме того, есть выбор типа термостатирования, а также задание температур термостатирования.

При нажатии на кнопку «ОК» появляется окно, предлагающее произвести запись измененной информации в термостат.

По сравнению с версией **ver. 1.0a** в данной версии улучшена стабильность работы термостата, найдены и исправлены мелкие ошибки, что в целом повысило надежность и стабильность функционирования устройства.

## Версия 1.1b

В основном меню добавлен пункт **«Режим экрана»**. Этот пункт отвечает за выбор между режимами **автоматической** смены между температурой и временем либо **статический** вывод или температуры, или времени по выбору пользователя. В статическом режиме выбор показаний выбирается в основном режиме кнопкой «enter».

Если отсутствуют все 4 датчика с 5 по 8, то вывод на экран информации о состоянии нагрузок с 5 по 8 не производится. Однако если нагрузки управляются от таймера и хотя бы одна из нагрузок с 5 по 8 активна, то окно с нагрузками с 5 по 8 отображаться будет.

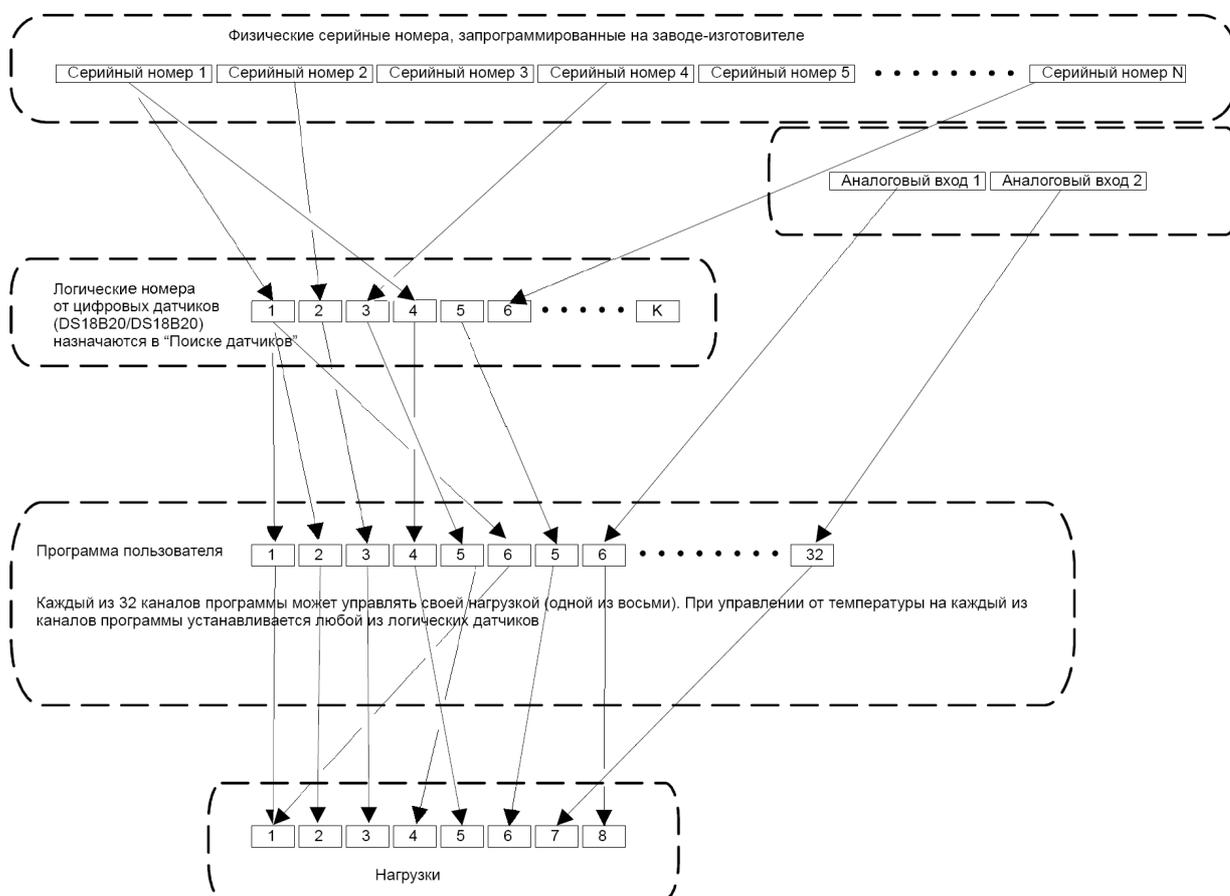
## Версия 1.2a

В данном обновлении серьезно изменен интерфейс установок, а также принципы организации связи датчиков и нагрузок, управляемых от этих датчиков.

Начиная с данного обновления количество датчиков, которые подключаются к устройству увеличено до 32. Каждый датчик получает свой логический номер в «Поиске датчиков». Каждый уже логически пронумерованный датчик может быть назначен на любой из нагрузок, это задается в каждой записи 32-х канальной программы управления. Это позволит, как вариант, управлять одной нагрузкой двумя и более датчиками. Введены в работу 2 аналоговых входа микроконтроллера – 38 и 37: выводы микроконтроллера (соответствуют 1 и 2 логическому входу АЦП).

Выбор от какого входа управлять – аналогового или цифрового осуществляется также в каждой из записей программы управления. Разрядность АЦП 10 бит. Управление, как и раньше подразделяется на нагревание или на охлаждение (а так же просто таймер и будильник как и в предыдущих версиях). Задание порога для АЦП осуществляется в виде двух напряжений гистерезиса. Размах – от 0 до 5В, дискретность шага <math><0.01\text{В}</math>.

Нижеследующая картинка показывает как соотносятся датчики, их логические номера и привязка к нагрузкам



### Описание изменений в меню и в других частях интерфейса

В меню добавился пункт «Монитор АЦП», в данном пункте отображается значение первого и второго каналов АЦП в вольтах.

Алгоритм программирования таймера-термостата.

В данной версии переименован пункт меню «Программа» на «Программа упр.». При входе в данный пункт меню клавишами навигации, как и в предыдущих версиях можно выбирать один из 32 каналов управления и клавишей «ввод» входить в редактирование выбранной записи. Стадии «Старт» и «Стоп» не поменялись, в следующем окне после установки времени остановка вводится теперь номер канала нагрузки, алгоритм управления нагрев/охлаждение (теперь нагрев обозначается вместо кружка стрелкой  $\uparrow$  а охлаждение – стрелкой  $\downarrow$  вместо звездочки), выставляется номер датчика (по регистрации) и тип датчика – аналоговый (номера 1 или 2) стрелками выбирается буква «А» или цифровой «D» (номера от 1 до 32).

Для улучшения стабильности работы при использовании 2-х проводного подсоединения датчиков датчики переводятся в режим меньшей разрядности преобразования (до 10 бит, что укорачивает время преобразования и уменьшает потребляемый ток). Те же датчики, которые подключены по 3 проводной линии (питание-данные-общий) инициализируются в стандартный 12 битный режим. При количестве датчиков более 15 использовать 2-х проводное соединение не рекомендуется, так как паразитного питания может быть уже не достаточно для питания всего массива термо-сенсоров.

#### **Особенности отображения информации на экране**

В данной версии отображение происходит по такому же принципу что и в предыдущей версии, однако, следует внимательно относиться к тому, что температуры отображаются по порядку их регистрации, а не по их привязке к конкретным нагрузкам. Температура датчиков с 9 по 32 не отображается.

Старт программы внешне изменен незначительно: строка «Startup» появляется и отображает инициализацию всех потенциальных 32 термодатчиков, если каких то датчиков нет – то инициализация пропускается, что убыстряет процесс старта.

Во время работы «мониторятся» только зарегистрированные датчики. Это значит, что при циклическом опросе датчиков те датчики которые не зарегистрированы просто пропускаются, что приводит к меньшей инерции измерений. Если все датчики присутствуют на линии (32), то примерно время опроса всех сразу – полминуты. Если, например, подключено 10 датчиков, но зарегистрировано только 2, то происходит опрос только этих двух датчиков (информация по ним будет обновляться примерно раз в 2 секунды).

#### **Изменения в программе для компьютера**

В программе добавлен выбор датчика и его типа. Кроме того в основном окне добавлена кнопка «**посмотреть серийные номера датчиков**». Эта функция не на что не влияет, просто дает возможность пользователю посмотреть так называемую «laser ROM»-память датчиков.

#### **После обновления**

После обновления следует произвести полную ре-инициализацию, так как структура запоминаемых данных претерпела изменения которые не будут совместимы со старым содержимым EEPROM термостата. (клавишей «Меню» при старте)

---

## Версия 1.3а

### Изменения функций АЦП

В данной версии ПО АЦП максимально облагорожен. Теперь АЦП можно «откалибровать» на любые значения отображаемых величин в пределах от  $-3.34E+38$  до  $3.34E+38$  и малых до степеней  $E-38$ ! Это значит, что достаточно ввести пределы соответствия цифрового значения 0 (минимальное значение АЦП) к определенному физически смысловому числу и 1023 (максимальное значение АЦП) так же к определенному числу, то при измерении можно будет наблюдать соответствующую величину, рассчитываемую по линейному закону. Обычно, в документации на датчик можно найти параметры линейной зависимости напряжения на выходе датчика от измеряемого параметра (давление, влажность, температура и т. д.). Из этого соотношения следует выявить зависимость физической величины от выходного напряжения датчика и, подставив в получившуюся формулу 0В и 5В, получим требуемые калибровочные значения. Также можно задать точность вывода информации от 0 до 5 знаков после запятой. Если число имеет значащие разряды далее чем 5 знаков после запятой, то отображение будет производиться в экспоненциальной форме. Так же, если число будет более 7 целочисленных знаков, то оно так же будет отображаться в экспоненциальном виде. Кроме того, есть возможность задать так называемый *суффикс*. Это символ, который будет всегда отображаться в конце числа при выводе пересчитанной величины. Например, для давления это будет 'P', для влажности '%' и т. д. Допускается установить только **1** символ суффикса.

#### Пример применения АЦП №1:

Подключим популярный датчик влажности серии HIH-3610 или HIH4000

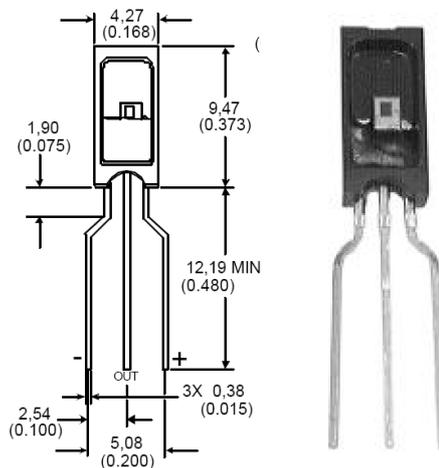


Рис. Внешний вид

Humidity/Moisture Sensors  
Humidity Sensor

HIH-3610 Series

#### FACTORY CALIBRATION

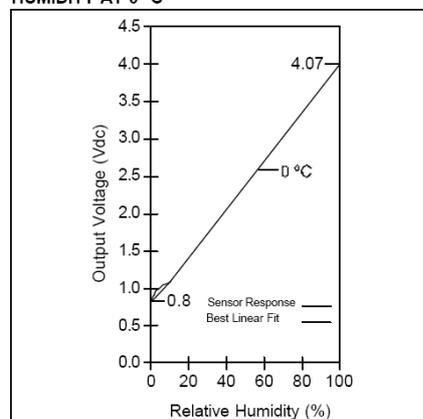
HIH-3610 sensors may be ordered with a calibration and data printout (Table 2). See order guide on back page.

TABLE 2: EXAMPLE DATA PRINTOUT

Model	HIH-3610-001
Channel	92
Wafer	030996M
MRP	337313
Calculated values at 5 V	
$V_{out}$ @ 0% RH	0.958 V
$V_{out}$ @ 75.3% RH	3.268 V
Linear output for 2% RH accuracy @ 25 °C	
Zero offset	0.958 V
Slope	30.680 mV/%RH
RH	$(V_{out}-zero\ offset)/slope$
	$(V_{out}-0.958)/0.0307$
Ratiometric response for 0 to 100% RH	

Рис. Выдержка из документации на датчик HIH3610

FIGURE 2: OUTPUT VOLTAGE VS RELATIVE HUMIDITY AT 0 °C



Электрически подключить данный датчик практически не является проблемой: достаточно подать на «+» 5В, на «-» общий и с «OUT» снять полезный сигнал и подать его на АЦП. А вот как откалибровать его таким образом, чтобы показания соответствовали значению влажности, а не вольтам или чему-нибудь еще – сложнее. Для этого потребуются приложить немного усилий и вспомнить азы математики. Посмотрим часть документации на датчик, в которой описаны калибровочные параметры.

В данной таблице уже присутствует необходимая нам формула, однако нам надо понять откуда она появилась. Для этого выведем ее самостоятельно. Первоначально для данного датчика необходимо написать формулу зависимости напряжения на выходе от влажности. Ориентируясь по таблице, определяем рабочую формулу еще раз и сверяем свои рассуждения с графиком:

Линейный закон диктует то что зависимость выходного напряжения от влажности будет выглядеть как:

$$V_{out} = \frac{V_{out}@75.3RH - V_{out}@0RH}{75,3\% - 0\%} * RH + Zerooffset \quad (1)$$

Эту формулу можно переписать в коротком выражении

$$V_{out} = slope * RH + Zerooffset \quad (2)$$

где slope – угол наклона прямой, равный:

$$slope = \frac{V_{out}@75.3RH - V_{out}@0RH}{75,3\% - 0\%} = 0.0307 \quad (3)$$

а Zerooffset = 0.958 – это аддитивная составляющая, на графике хорошо видно откуда она берется.

Выведем из формулы (2) RH

$$RH = \frac{V_{out} - Zerooffset}{slope} \quad (4)$$

- эта формула и приведена в таблице 2, ошибки нет

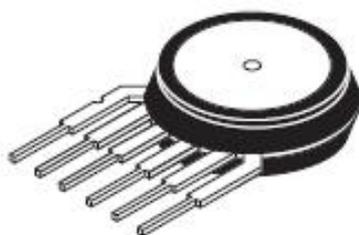
И для того, чтобы получить две виртуальные точки значений RH на 0В и на 5В, достаточно подставить эти напряжения в формулу (4)

$$RH_{0v} = \frac{0 - 0.958}{0.0307} = -31.20 \quad RH_{5v} = \frac{5 - 0.958}{0.0307} = 131.66$$

Эти значения и следует ввести в качестве соответствий минимальному и максимальному значению в установках АЦП. В качестве суффикса методом перебора назначить «%». Точность выставить по желанию, но выше 1 знака после запятой смысла, наверное, нет.

*Пример применения АЦП №2:*

Возьмем датчик давления MPX4115A



**MPX4115A**

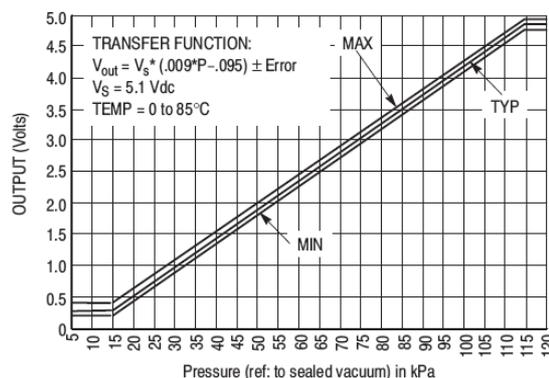


Figure 4. Output versus Absolute Pressure

Возьмем формулу поведения выходного напряжения от давления из документации на данный датчик

$$V_{out} = V_s * (0.009 * P - 0.095) \quad (5)$$

Где  $V_s$  - напряжение питания (в нашем случае 5В взятое из схемы термостата)

Выведем из этой формулы P.

$$P = \frac{\frac{V_{out}}{V_s} + 0.095}{0.009} \quad (6)$$

Теперь подставим граничные напряжения 0В и 5В.

$$P = \frac{\frac{0}{5} + 0.095}{0.009} = 10.5(5) \quad P = \frac{\frac{5}{5} + 0.095}{0.009} = 121.6(6)$$

Эти 2 значения надо ввести в минимум и максимум соответственно. В результате мы будем иметь на дисплее значения давления в кПа. Если требуется перевести эти значения в миллиметры ртутного столба, то следует разделить оба калибровочных значения на  $101325/760 \approx 133,3$ . И получим 79.2 и 913, соответственно. И, если, ввести эти значения в качестве калибровочных коэффициентов, то получим показания в мм рт. ст.

Данный датчик работает в диапазоне давлений от 0 до чуть более 1атм, поэтому измерение атмосферного давления измерения будет производиться у края диапазона датчика. Точность измерения (как следствие разрешающей силы АЦП) будет колебаться в пределах 1 мм рт. ст. Если требуется увеличить точность, то электрическим образом следует сместить и растянуть полезный диапазон в область 0В-5В и соответствующим образом рекалибровать АЦП. Кроме того, любой датчик имеет определенную систематическую погрешность, поэтому при калибровке желательно иметь собственные калибровочные таблицы и по ним рассчитывать свои калибровочные коэффициенты.

#### **Изменения в управлении нагрузками**

Количество управляемых нагрузок увеличено до 12. Список новых нагрузок:

9 нагрузка – РС2

10 нагрузка – РС3

11 нагрузка – PD3

12 нагрузка – PD6

Сигналы с данных выходов можно нагружать током до 20мА. Активный уровень = 5В.

Введена возможность включения импульсного режима управления нагрузками. Этот режим позволяет не постоянно включать нагрузку на время заданное в таймере, а включать её на определенное время потом отключать и снова включать... Диапазон времени, при котором нагрузка находится в режиме активности от 3 до 250 секунд, диапазон времени при котором нагрузка находится в отключенном состоянии от 0 до 250 секунд. Этот режим нужен для управления нагрузками которые требуют кратковременное включение для достижения результата их работы. Чтобы деактивировать импульсный режим достаточно ввести в параметр времени не активности 0с, а во время активности любое число больше 0.

*Примечание: минимальное устанавливаемое время активности 3 секунды, время не активности следует устанавливать не менее 2 секунд.*

#### **Режим отображения**

Режим отображения переработан значительно и теперь позволяет задавать время отображения того или иного параметра, включая время и дату. Есть 7 экранов, которые по желанию можно по очереди наблюдать в основном рабочем режиме:

1. Отображение нагрузок и термодатчиков с 1 по 4.
2. Отображение нагрузок и термодатчиков с 5 по 8.
3. Отображение нагрузок и термодатчиков с 9 по 12.
4. Отображение времени/даты.
5. Отображение значения 2 каналов АЦП в числовом виде.
6. Отображение значения 2 каналов АЦП в виде напряжения.
7. Отображение значения 2 каналов АЦП в виде физической величины с заданной точностью и суффиксом.

Время отображения может варьироваться в пределах от 0 (не отображается) до 30 секунд.

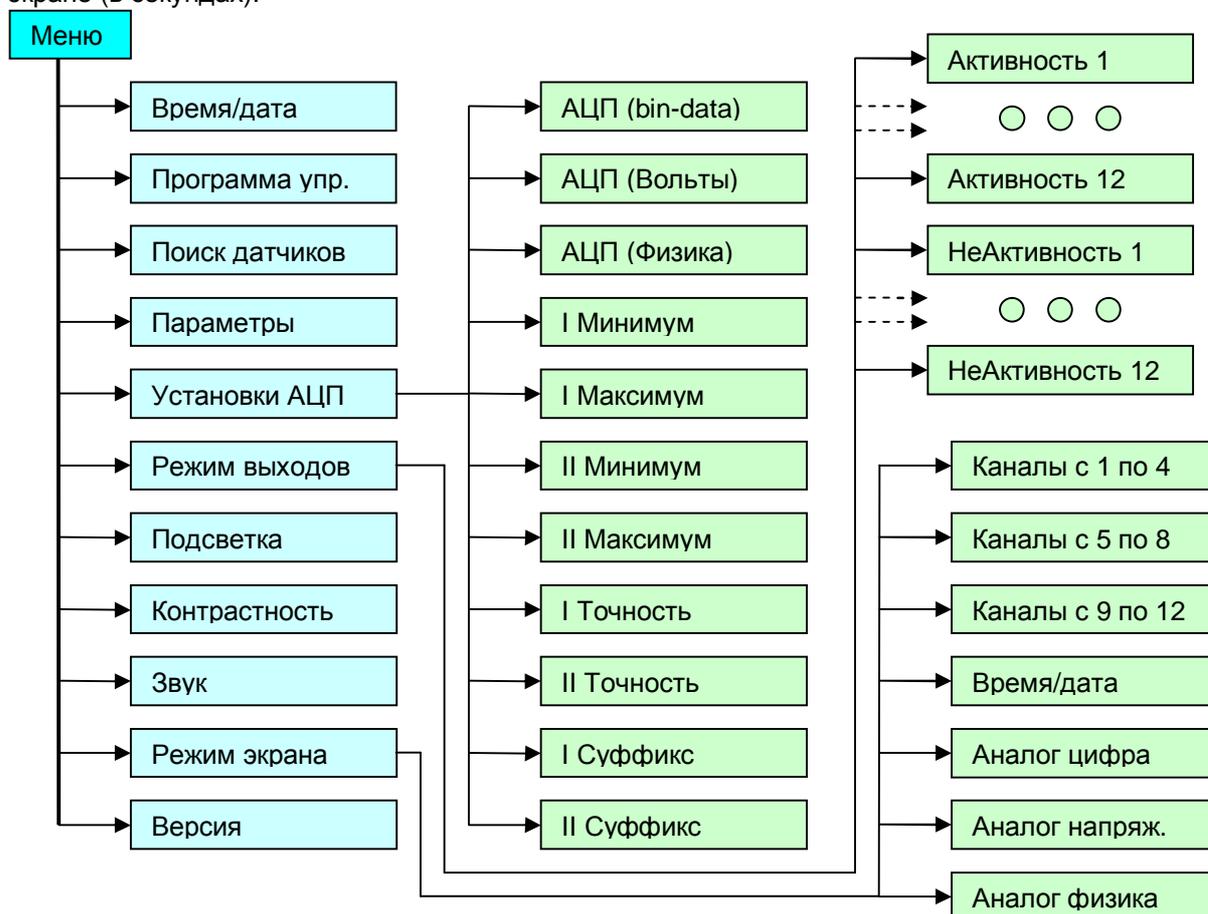
#### **Изменения в меню**

Новая топология меню представлена на нижеследующем рисунке.

Установки АЦП: в первых 1м, 2м и 3м пунктах данного меню можно наблюдать значения АЦП в числовом виде, в виде напряжения и в виде физической величины соответственно (в экспоненциальной форме). В следующем пункте для 1-го канала АЦП задается физическое значение которое соответствует НУЛЮ АЦП. В следующем пункте для 1-го канала АЦП задается физическое значение, которое соответствует МАКСИМУМУ АЦП (1023). Эти значения используются для пересчета текущего цифрового значения в определенное физическое значение.

Следующие 2 пункта выполняют ту же задачу, только для 2 канала. Далее идут пункты, задающее точность и суффикс для обоих каналов.

В главном меню пункт «Режим выходов» разворачивает дополнительное подменю, в котором задается режим работы каждого из 12 выходов. Времена активности и не активности определяют времена (в секундах), на которые будут активированы/деактивированы выходы при условии что все остальные условия, которые должны включать выход выполнены. Крупные изменения коснулись пункта главного меню «Режим экрана», теперь он разворачивает дополнительное подменю, в котором можно задать время соответствующего отображения на экране (в секундах).



#### Изменения в установке программы управления

Теперь можно выбирать нагрузки с 1 по 12. При вводе управления с аналогово-цифрового преобразователя можно задавать физические величины значений, по которым происходит регулировка на выходе. Задание величины осуществляется по градациям АЦП (В примененном микроконтроллере разрешение АЦП составляет 10бит → 1024 градации) и соответственно ШАГ установки будет тем больше (грубее), чем шире диапазон между минимумом и максимумом установленными при калибровке АЦП.

#### Исправленные ошибки

Исправлена ошибка регулировки на охлаждение на нагрузках 5-8.  
Исправлена ошибка отображения датчиков температуры с 5 по 8.  
Исправлена работа с СОМ-портами с номерами более 9.

#### Программа управления для ПК

- В программу управления добавлены кнопки «сохранить» и «открыть», которые позволяют сохранять в файл и восстанавливать из файла программу управления.
- Добавлена кнопка **записи приветствия**, теперь можно записать свои собственные слова приветствия, которое будет вас радовать при каждом включении устройства.
- Введена опция настройки всех параметров, касающихся АЦП, а так же режима отображения «опция отображения».
- Введено диалоговое окно, позволяющее редактировать параметры импульсного режима выходов: включать, отключать или изменять параметры.

## **Версия 1.4a**

В основном изменения в данной версии касаются только исправлению ошибок версии 1.3a.  
Исправлена ошибка:

- с работой датчиков с 9 по 12 (гистерезис)
- при работе на отрицательных температурах

Что еще:

- изменен алгоритм опроса датчиков температуры
- увеличено быстродействие за счет оптимизации кода
- изменена программа для компьютера под работу с версией 1.4a

После программирования версии 1.4a после 1.3a начальная инициализация не требуется

---

## Версия 1.5a

### Изменения:

- в установках режима экрана: при установке всех значений в 0 становится возможным ручное перемещение по «экранам» стрелками «вверх» и «вниз», выбирая, что показывать – время, выходы или значение АЦП и т.д.
- Ускорен алгоритм обработки выходов, исправлены некоторые незначительные ошибки.
- Оцифровка теперь более стабильная, программным образом уменьшено количество шумов при аналогово-цифровом преобразовании как минимум вдвое.

## Версия 1.5b

### Изменения:

- Устранение эффекта «дребезга» контактов кнопок клавиатуры, возникшего при переработке модуля АЦП. При обновлении с версии 1.5a до версии 1.5b начальная инициализация не требуется
- Исправления и дополнения в документации в **приложении 2** (Описание управления термостатом через последовательный интерфейс) для соответствующей версии

## Версия 1.5c

### Изменения:

- Следующий этап борьбы с нежелательным выключением устройства, очередная переработка программного модуля АЦП
- При «пропадании» цифрового датчика из поля видимости – вся деятельность по программам использующих данный датчик прекращается, и возобновляется когда датчик вновь появится
- Устранен эффект скачков вверх/вниз температуры при ее резких изменениях. Это должно положительным образом сказаться при достижении более точной температуры термо-стагирования путем ставшего РЕАЛЬНО возможным уменьшения «ножниц» гистерезиса.

PS: начальная инициализация не требуется. Все программы сохраняются. Программа для ПК не меняется.

## Версия 1.5d

### Изменения:

- Исправлена ошибка работы выходов: когда номер датчика соответствовал номеру нагрузки (по порядковому номеру)...
- Дополнено дополнительными командами от компорта
- Исправлены ошибки в данном документе в **Приложении 2**

PS: начальная инициализация не требуется. Все программы сохраняются. Программа для ПК не меняется.

## Версия 1.6

### Изменения:

- Главная новость: **Представлена альтернативная программа для ПК**, написанная опытным пользователем 8036, *аквариумистом* (и программистом по совместительству). Контактные данные разработчика указаны в самой программе, эти данные можно использовать, чтобы сообщить разработчику об ошибках или высказать свои пожелания по улучшению функциональности софта. Данная программа имеет интуитивно понятный интерфейс, из интересных особенностей данного ПО можно выделить следующее:
  - программа позволяет производить запись значений температур и нагрузок в лог
  - имеет терминальный режим, в котором можно видеть, что на дисплее 8036 в данный конкретный момент и вызывать виртуальные нажатия клавиш
  - *а так же много чего другого интересного*
- Исправлены проблемы с импульсным режимом, точность импульсного режима улучшена точность работы импульсного режима до 1 секунды (вместо 3 в предыдущих версиях)
- Расширен импульсный режим до ~18час/18час (по 65535 секунд на состояние)
- Добавлен флаг обновления содержимого экрана доступен к чтению по последовательному интерфейсу 'd'
- Приветствие теперь хранится в памяти микросхемы часов (DS1307). Таким образом, при вынимании батарейки приветствие будет сбрасываться на «По-умолчанию»
- Программа для ПК изменена в соответствии с новыми особенностями работы новой прошивки, программа не совместима со старыми версиями прошивки
  - Если включен импульсный режим, то в manager активность выхода будет обозначена даже если нагрузка отключена на время Не-активности
- Введен новый режим «Ручное управление». Вызывается из главного меню. После вызова режима отключается обслуживание нагрузок в соответствии с программами. И появляется возможность включать и выключать нагрузки вручную. В данном режиме импульсное управление не имеет действие. При выходе из данного режима происходит активация управления нагрузками по программе, а так же активация импульсного режима если таковой разрешен
- *Улучшена устойчивость 1-wire.*

## Версия 1.7

### Изменения:

- Подсветка при выключенном режиме теперь включена на 10% (вместо ~30%)
  - В режиме переключения экранов исправлена ошибка, и теперь время переключения точно соответствует заданному времени с точностью до 0.1 секунды
  - При управлении в ручном режиме звук включения нагрузок отключен
  - Добавлена команда чтения реального состояния порта в интерфейс связи с ПК: 'z' (чтение нагрузок 'i' теперь обозначает логическую активность канала, не учитывая импульсный режим выхода). А в ручном 'I' режиме работает аналогично 'z'
  - Укорочен звук при нажатии клавиш
  - Скорректированы данные в Приложении 2.
- 

## Версия 1.8

### Исправления:

- Исправлена работа с датчиками DS18S20 (ошибка удвоения температуры)
- Переделан алгоритм будильника с целью улучшения стабильности работы
- Исправлена ошибка интерфейса при вводе чисел
- В ручном управлении нагрузками ошибка исправлена.

### Изменения:

- Ввод «времени старта» меньше «времени остановки» программы в периодическом режиме, теперь не ошибка. Программа будет работать с переходом через полночь. Это позволит сэкономить 1 шаг программы.
  -
- 

## Версия 1.9

### Исправления:

- Улучшена надежность работы с датчиками и в случае «зависания» какого либо датчика и вследствие этого полного информационного разрушения сети. Теперь этот момент отслеживается, и все нагрузки будут отключаться.

### Изменения:

- В программах термостата добавлена новая опция – настройка режима взаимодействия программ. Варианты выбора: «AND» или «OR». Это позволит более гибко запрограммировать систему, установив приоритеты программ.

Если выбран режим OR, то при выполнении всех условий включения канала нагрузки – канал будет активироваться, а при невыполнении условий – данная программа никак не будет влиять на результат влияния предыдущих программ на данную нагрузку. В случае же если выбран режим AND – то при выполнении всех условия активации канал может только остаться активным, если предыдущие программы его уже активировали, а если же программа для данной нагрузки не попадает в установленные условия, то на данном шаге нагрузка не зависимо от выполнения предыдущих программ будет деактивирована.

Аналитически, то, как программа работала в предыдущих версиях можно кратко выразить в следующей упрощенной форме записи:

$$H = (...(((P1 \text{ ИЛИ} P2) \text{ ИЛИ} P3) \text{ ИЛИ} P4)...) )$$

В результате введенной возможности закон управления, например, для программы P3 можно поменять на “AND” (логическое И), и формула приобретет следующий вид:

$$H = (...(((P1 \text{ ИЛИ} P2) \text{ И} P3) \text{ ИЛИ} P4)...) )$$

H – одна из нагрузок

P – программы для управление нагрузкой H

По умолчанию в установке каждой программы будет режим OR (логическое ИЛИ всех «программ») – что значит, что программа будет работать в старом режиме  
Соответственно изменено сообщение с ПК для передачи и приема нового параметра, изменена программа manager.

## Приложение 1 Обновление прошивки микроконтроллера программой **flasher.exe**

Для совершения обновления подключите термостат нуль-модемным кабелем к компьютеру, разархивируйте и запустите программу **flasher.exe**, затем нажмите кнопку «Тест» для проверки соединения. Если последует сообщение о том, что связь есть, то нажмите кнопку «Запись», и в появившемся окне выбора файла найдите новую прошивку для термостата с названием **8036.cod**.

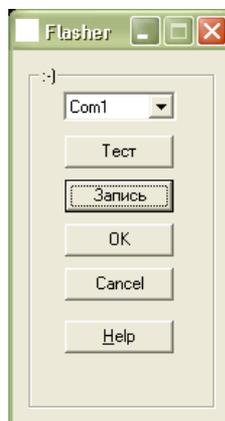


Рисунок 3

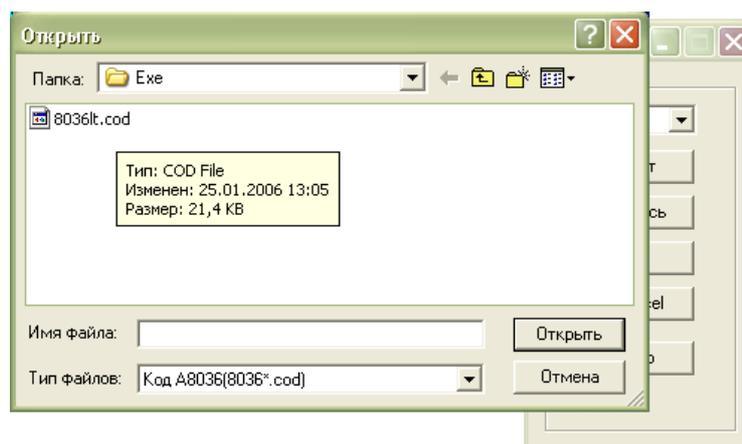


Рисунок 4

После того как начнется процесс программирования, следует нажать и удерживать кнопку «меню» до завершения программирования и не отпускать её до момента перезапуска с новой программой. Это требуется для того, чтобы произвелась инициализация значений переменных в EEPROM в соответствие с новой программой.

### **Внимание!**

Необходимо ввести «правильный» файл при производстве обновления, так как если программе дать не тот файл, то она его запрограммирует в память микроконтроллера без проверки, и устройство перестанет работать. Если все же это произошло, то необходимо предпринять ряд мер чтобы устройство «вернуть к жизни»:

1. Выключить питание;
2. Подсоединить компьютерный кабель;
3. Нажать кнопку «POWER» и удерживая её подать питание на устройство. После этого следует отпустить кнопку;
4. Запустить программу **flasher.exe**;
5. Открыть корректный файл и подождать пока произойдет программирование.

## Приложение 2

## Описание управления термостатом через последовательный интерфейс

## Общие сведения

Скорость связи 9600 бит/с, 2 стоп бита. Зеленым цветом обозначено движение данных от термостата к компьютеру. Синим цветом – обозначено направление передачи данных от компьютера к термостату.

## Описание команд для версии 1.7

## 1. 't' Чтение значений температуры

Команда: 1 байт – ASCII-символ 't'

Ответ: 1 байт количество температурных датчиков (32)  
1 байт – количество термодатчиков (для версии 1.2a и далее =32), далее передаются значения температуры всех 32 датчиков. Формат числа float(16-битное знаковое целое число со значением температуры, умноженное на 100)

## 2. 'l' Чтение логического состояния выходов (не зависимо от импульсного режима)

Команда: 1 байт – ASCII-символ 'l'

Ответ: 2 байтное число, каждый бит которого соответствует состоянию нагрузки. 11 бит соответствует 1 выходу, 10 бит – 2 выходу и т.д. до 0 бита, который соответствует 12 выходу

## 3. 'z' Чтение реального состояния выходов в данный конкретный момент

Команда: 1 байт – ASCII-символ 'z'

Ответ: 2 байтное число, каждый бит которого соответствует состоянию выхода: 11 бит сопоставлен 1 выходу, 10 бит – 2 выходу и т.д. до 0 бита, который соответствует 12 выходу

## 4. 'b' Чтение уровня заряда батареи часов реального времени

Команда: 1 байт – ASCII-символ 'b'

2 байтное слово соответствующее представлению числа от 0 до 1024 что соответствует напряжению соответственно от 0 до 5В

## 5. 'V' Чтение версии ПО микроконтроллера

Команда: 1 байт – ASCII-символ 'V'

Ответ: символ 'V'

1 байт – число, представляющее длину строки текста, который следует после (в конце строки нет флага окончания строки 00h)

## 6. 'T' Запись времени

Команда: 1 байт – ASCII-символ 'T'

Ответ: 1 байт "эхо" символ 'T'

Команда: Далее передаются данные календаря по указанной структуре. Суммарно 8 байт.

Описание структуры:

```
struct struct_clock{
// байт №1
  unsigned char seconds:4; // секунды (4 бита)
  unsigned char ten_seconds:3; // десятки секунд (3 бита)
  unsigned char ch:1; // всегда = 0 (1 бит)
// байт №2
  unsigned char minutes:4; // минуты
  unsigned char ten_minutes:3; // десятки минут
  unsigned char reserved_0:1;
// байт №3
  unsigned char hours:4; // часы
  unsigned char ten_hours:2; // десятки часа
  unsigned char AMPM_24_mode:1; // всегда = 0
  unsigned char reserved_1:1; // зарезервирован
// байт №4
  unsigned char day:3; // День недели (1-7)
  unsigned char reserved_2:5;
// байт №5
  unsigned char date:4; // число (1-31)
  unsigned char ten_date:2; // число (десятки)
  unsigned char reserved_3:2;
// байт №6
  unsigned char month:4; // месяц (1-12)
  unsigned char ten_month:1; // месяц(десятки)
  unsigned char reserved_4:3;
// байт №7
  unsigned char year:4; // год от 0 до 99
  unsigned char ten_year:4; // десятки года
// байт №8
  unsigned char RS:2; // всегда = 0
  unsigned char reserved_5:2;
  unsigned char SQWE:1; // всегда = 1
  unsigned char reserved_6:2;
  unsigned char OUT:1; // всегда = 1
};
```

## 7. 'c' Чтение времени

Команда: 1 байт – ASCII-символ 'c'

Ответ: 1 байт "эхо" символ 'c'

Команда: Далее передаются данные календаря по структуре указанной в п. 5. Суммарно 8 байт.

## 8. 'L' Считывание программы управления в компьютер

Команда: 1 байт – ASCII-символ 'L'

Ответ: 1 байт "эхо" символ 'L'

Ответ: 832 байта данных. Передаются 32 записи, каждая из которых соответствует нижеописанной структуре. Каждая структура занимает 27 байт, поэтому, суммарно  $27 * 32 = 864$  байта

Описание структуры:

```
struct DataToORFromPC{
    unsigned char time_on_h; // время старта час 0-23
    unsigned char time_on_m; // время старта минута 0-59
    unsigned char time_on_s; // время старта секунда 0-59
    unsigned char time_off_h; // время остановки час
    unsigned char time_off_m; // время остановки минута
    unsigned char time_off_s; // время остановки секунда
    unsigned char time_on_day; // число старта 1-31
    unsigned char time_off_day; // число остановки
    unsigned char time_on_month; // месяц старта от 1 до 12
    unsigned char time_off_month; // месяц остановки
    unsigned char time_on_year; // ГОД СТАРТА от 0 до 99 (соответствует от 2000 до 2099)
    unsigned char time_off_year; // ГОД ОСТАНОВКИ
    unsigned char time_load; // номер нагрузки от 0 до 7
    unsigned char time_loadsensor; // датчик, соответствующий данной нагрузке
    unsigned short time_min; // минимум для температуры это от -5500 до +12500 (-55 до 125град)
    unsigned short time_max; // максимум для температуры это от -5500 до +12500 (-55 до 125град)
    unsigned char time_mode ; // режим 0=нагрев, 1=охлаждение, 2=будильник, 3=по таймеру
    unsigned char time_eze; //отрабатывать по дням=1, по дням недели=2, по месяцам=3, без
    // периода=0
    unsigned long time_ezedata; // 32 бита соответствующие выбранной периодичности
    unsigned char time_bud; // служебный (считать и записать в то же состояние)
    unsigned char enable; // разрешение работы данного канала
    unsigned char sensortype; // Бит 0: тип 0 – DS18B20, 1 – аналоговый вход, бит 1:– Закон ИЛИИ
};
```

#### 9. 'W' Запись программы управления в 8036

Команда: 1 байт – ASCII-символ 'W'

Ответ: 1 байт «эхо» - символ 'W'

Команда: В ответ уходит «эхо» - символ 'W'. После отправляется номер редактируемой записи (0-31), далее структура аналогичная в п.8<sup>1</sup>

#### 10. 'D' Считывание серийных номеров зарегистрированных датчиков Dallas

Команда: 1 байт – ASCII-символ 'D'

Ответ: 1 байт «эхо» - символ 'D'

Отправляется 32 поля серийных номеров DS18B20 по 8 байт (суммарно 256 байт)

Если датчик не зарегистрирован на центральном блоке – то получить его серийный номер не получится

#### 11. 'h' Считать стартовые строки приветствия

Команда: 1 байт – ASCII-символ 'h'

Ответ: 1 байт «эхо» - символ 'h'

Принимается 17 байт верхняя строка (16 символов + ноль)

Принимается 17 байт нижняя строка (16 символов + ноль)

#### 12. 'H' Записать стартовые строки приветствия

Команда: 1 байт – ASCII-символ 'H'

Ответ: 1 байт «эхо» - символ 'H'

Команда: Отправляется 17 байт верхняя строка (16 символов + ноль)

Отправляется 17 байт нижняя строка (16 символов + ноль)

#### 13. 'i' Считать различные параметры настройки АЦП и отображения

Команда: 1 байт – ASCII-символ 'i';

Ответ: В ответ уходит «эхо» - символ 'i'

4-байтное число с плавающей точкой (float) нижний предел калибровки АЦП1 (на 0В)

4-байтное число с плавающей точкой (float) верхний предел калибровки АЦП1 (на 5В)

4-байтное число с плавающей точкой (float) нижний предел калибровки АЦП2 (на 0В)

4-байтное число с плавающей точкой (float) верхний предел калибровки АЦП2 (на 5В)

2 байта – символы суффиксов 1 и 2

2 байта – целочисленные значения количества знаков после запятой при отображении значений АЦП.

1 байт – время в секундах отображения с 1 по 4 каналы температуры и выходов

1 байт – время отображения с 5 по 8 каналы температуры и выходов

1 байт – время отображения с 9 по 12 каналы температуры и выходов

1 байт – время отображения часов

1 байт – время отображения значения АЦП в цифровом виде (0-1023)

1 байт – время отображения значения АЦП в виде напряжения

1 байт – время отображения значения АЦП в виде пересчитанной физической величины

#### 14. 'I' Записать различные параметры настройки АЦП и отображения

Команда: 1 байт – ASCII-символ 'I';

Ответ: 1 байт «эхо» - символ 'I'

Команда: 4-байтное число с плавающей точкой (float) нижний предел калибровки АЦП1 (на 0В)

4-байтное число с плавающей точкой (float) верхний предел калибровки АЦП1 (на 5В)

4-байтное число с плавающей точкой (float) нижний предел калибровки АЦП2 (на 0В)

4-байтное число с плавающей точкой (float) верхний предел калибровки АЦП2 (на 5В)

2 байта – символы суффиксов 1 и 2

2 байта – целочисленные однобайтные значения количества знаков после запятой при отображении значений АЦП.

<sup>1</sup> Промежуток между отсылаемыми байтами структуры 5мс.

1 байт – время в секундах отображения с 1 по 4 каналы температуры и выходов  
 1 байт – время отображения с 5 по 8 каналы температуры и выходов  
 1 байт – время отображения с 9 по 12 каналы температуры и выходов  
 1 байт – время отображения часов  
 1 байт – время отображения значения АЦП в цифровом виде (0-1023)  
 1 байт – время отображения значения АЦП в виде напряжения  
 1 байт – время отображения значения АЦП в виде пересчитанной физической величины

**15. 'a' Считать время активности и не-активности выхода (импульсный режим)**

Команда: 1 байт – ASCII-символ 'a'

Ответ: 1 байт «эхо» - символ 'a'

12 значений активности в секундах, каждое в виде 2-байтного числа (итого 24 байт)

12 значений не-активности в секундах, каждое в виде 2-байтного числа (итого 24 байт)

**16. 's' Считывание состояний аналоговых входов**

Команда: 1 байт – ASCII-символ 's'

Ответ: 1 байт «эхо» - символ 's'

4 значения АЦП, каждое в виде 2-байтного беззнакового числа (итого 8 байт)

**17. 'A' Записать время активности и не-активности выхода (импульсный режим)**

Команда: 1 байт – ASCII-символ 'a'

Ответ: 1 байт «эхо» - символ 'a'

Команда: 12 значений активности в секундах, каждое в виде 1-байтного числа (итого 12 байт)

12 значений неактивности в секундах, каждое в виде 1-байтного числа (итого 12 байт)

**18. 'K' Создание эффекта нажатой кнопки**

Команда: 1 байт – ASCII-символ 'K'

Ответ: 1 байт «эхо» - символ 'K'

Команда: 1 байт – код кнопки

```
#define KEY_ENTER      0
#define KEY_ARR_UP    1
#define KEY_ARR_LF    2
#define KEY_ARR_RT    3
#define KEY_ARR_DN    4
#define KEY_MENU      5
#define KEY_PWR       6

#define KEY_NO        0xFF
#define KEY_YES       0xFE
#define KEY_FAST      0x80
```

**19. 'S' Считывание содержимого дисплея**

Команда: 1 байт – ASCII-символ 'S'

Ответ: 1 байт «эхо» - символ 'S'

64 байта – содержимое графической памяти

16 байт – верхняя строка

16 байт – нижняя строка

Итого 96 байт. **Смотреть таблицу символов дисплея!**

**20. 'd' Считывание флага сигнализирующего об изменении содержимого дисплея**

Команда: 1 байт – ASCII-символ 'd'

Ответ: 1 байт «эхо» - символ 'd'

1 байт флаг: 0, если содержимое дисплея не поменялось с момента предыдущего чтения и 1 если содержимое дисплея изменилось

**21. 'E' Обновление ПО**

Команда: 1 байт – ASCII-символ 'E'

Ответ: 1 байт «эхо» - символ 'E'

Далее устройство переходит в особый режим, в котором будет возможна работа загрузчика (Flasher). Выйти из данного режима можно командой 'X'