

```

REM '*****
REM ' Система доступа по картам MASTER.BAS
REM SUB ' ver 2.1 08.01.17
REM '*****
REM ' Требуется BASIC 3.4.7 или старше
REM '*****
REM ' SET USER PASSWORD = "" OR "xxxxxxxx"
REM ' SET INT FILESYSTEM = OFF
REM ' SET AUTORUN = ON
REM ' ПОДРОБНОЕ ОПИСАНИЕ В HELP.BAS
REM ENDSUB
REM '*****

REM '*****
REM SUB ' HELP.BAS ver 2.0
REM '*****
REM ' В этом файле
REM ' Описание всех переменных
REM ' Описание взаимодействия с SLAVE
REM ' Описание файловой системы
REM ' Описание взаимодействия с WEB
REM ' Дополнительная информация
REM '*****
REM ENDSUB

REM '*****
REM SUB 'Общий алгоритм работы
REM '*****
REM 'Контроллер постоянно сканирует переменную В
REM ' Если не ноль, выполняе команду и ее обнуляет
REM ' Раз в секунду опрос Slave
REM ' Если было событие, пишет в log.htm
REM ' Раз в секунду опрашивает местный считыватель
REM ' Раз в секунду Текущее время контроллера в переменную
REM ENDSUB

REM '*****
REM SUB 'VARIABLE MEMORY Распределение памяти
REM '*****
REM ' Переменные , которые отображает WEB сервер
REM ' Список адресов SLAVE контроллеров
REM ' Пример подстановки ~VARD0a~ = a
REM ' a - адрес контроллера #1 хранится в еепром #0
REM ' b - адрес контроллера #2 хранится в еепром #2
REM ' c - адрес контроллера #3 хранится в еепром #4
REM ' d - адрес контроллера #4 хранится в еепром #6
REM ' e - адрес контроллера #5 хранится в еепром #8
REM ' f - адрес контроллера #6 хранится в еепром #10
REM ' g - адрес контроллера #7 хранится в еепром #12
REM ' h - адрес контроллера #8 хранится в еепром #14
REM ' если равен нулю - отключен
REM '
REM ' i - Текущий адрес контроллера web server
REM ' j - Текущий номер контроллера web server
REM '
REM ' k - Количество ключей в текущем контроллере
REM ' l - Время открытия двери в текущем контр.
REM ' m - статус блокировки кнопок в текущем контр.
REM ' n - статус блокировки считывателя в текущем контр.
REM ' o - статус блокировки двери в текущем контр.
REM ' p - права доступа.
REM '
REM '*****
REM ' Перменные для использования внутри программы
REM '
REM ' А позиция записи лога A=LOC{ } SEEK A
REM ' допустимо 97 - 32497
REM '
REM ' В Текущий номер контроллера

```

```

REM ' допустимо 1-8
REM '
REM ' C,D,E ПАРАМЕТРЫ web
REM '
REM ' F позиция БАЗЫ ПОЛЬЗОВАТЕЛЕЙ F=LOC{ } SEC F
REM ' допустимо 49 - 11049; 44*250
REM '
REM ' G - НОМЕР ПОЗИЦИИ В БАЗЕ
REM ' H,I,J ВРЕМЕННЫЕ ПЕРМЕННЫЕ
REM ' K - НОМЕР ПОЗИЦИИ В БАЗЕ ЗАПИСИ base_kn.htm
REM '
REM ' z - Текущий адрес контроллера Log File
REM '
REM ' Ключ из считывателя бинарник
REM '#(750) = 0/1
REM '#(751)-#(755) = ключ AB153345CD
REM '
REM ' Для сервиса IP
REM '#(235)- #(244) NAME + остальное
REM 'qrst - IP q=#(244) - t=#(245)
REM 'v - DG = #(246)
REM 'x - MASK =#(247)
REM 'IPCFG #(235) В переменную #(235) помещается структура
REM 'текущих сетевых настроек.
REM 'В первые девять байт #(235)- #(243) имя контроллера,
REM 'как текст (может быть выведен как $(235))
REM '#(244)- #(247) текущий IP, 4 байта = qrst
REM '#(248) - DG последний октет = v
REM '#(249) - MASK последний октет =x
REM '#(250) - reserved (00)
REM '
REM ENDSUB

```

```

REM '*****
REM SUB 'TEXT STRING in memory
REM '*****
REM '#(520) = len
REM '#(521)-#(599) ВРЕМЕННЫЕ ПЕРЕМЕННЫЕ
REM '
REM ' Текстовые сообщения, которые показывает WEB сервер
REM 'Пример подстановки строки текста ~VAR615~
REM '
REM ' статус связи , отдельно для каждого контроллера
REM ' Изменяется программой опроса лога из Slave
REM '#(600) = len
REM '#(601)-#(614) = text "Доступен" or "Не подключен"
REM ' or "Нет связи "
REM '#(615) = len, #(616)-#(629) = text
REM '#(630) = len, #(631)-#(644) = text
REM '#(645) = len, #(646)-#(659) = text
REM '#(660) = len, #(661)-#(674) = text
REM '#(675) = len, #(676)-#(689) = text
REM '#(690) = len, #(691)-#(704) = text
REM '#(705) = len, #(706)-#(719) = text
REM ' Всего 8 шт по 15 байт
REM '
REM ' код ключа в местном считывателе - текст
REM '#(720) = len =10
REM '#(721)-#(734) = AB153345CD
REM '[KEY_BAD]
REM '" - - - - - "
REM '
REM ' Статус состояния выполнения команды WEB сервера
REM '#(735) = len =14
REM '#(736)-#(749) = or
REM '"Готов "'
REM '"Обработка " - занят выполнением команды
REM ' Команда , поступившая в процессе обработки другой
REM ' скорее всего не будет выполнена

```

```

REM '
REM ' ГРАНИЦА 256 БАЙТ = 768 - ПЕРЕМЕННАЯ ДО И ПОСЛЕ
REM ' Статус выполнения последней команды
REM ' #(770) = len =19
REM ' #(771)-#(789) =
REM ' "Удачное завершение "
REM ' "Ошибка команды "
REM ' "Ошибка данных "
REM ' "Не выполнено "
REM ' "Нет места "
REM ' "Ключ не найден "
REM ' "Ключ есть в базе "
REM ' "Нет такого контрол."
REM ' "Контроллер занят "
REM ' "Нет связи с контр. "
REM '
REM ' Результаты поиска пользователя
REM ' пишет значения или обнуляет, если нет
REM ' #(814)- #(824) ключ 11 байт
REM ' #(900)- #(931) имя 32 байта
REM '
REM ' Текущее время контроллера
REM ' #(860)- #(879) Текст 19 байт max текущее время
REM ENDSUB

REM ' *****
REM SUB 'Data from web server
REM ' *****
REM ' sample .htm?lg=&ps=&VDA=123&VSB=1&IS010=121321
REM '
REM ' переменные, которые меняет WEB server
REM ' Текущий контроллер - с которым был последний сеанс
REM ' обмена данных по запуску WEB сервера
REM '
REM ' WEB сервер пердает (меняет переменные)
REM ' C - Новый/старый номер текущего контроллера
REM ' D - код операции , что сделать
REM ' E - данные - в зависимости от операции
REM ' Пример подстановки name="VDE"
REM '
REM ' Структура от WEB сервера поиск/нове/изменение
REM ' ключ
REM ' #(790) = len =10
REM ' #(791)-#(801) = ключ AB153345CD
REM ' Пример подстановки name="IS790"
REM ' Пользователь
REM ' #(825)- #(857) имя 32 байта - имя пользователя
REM '
REM ' Для сервиса усьановки времени
REM ' .htm?VDD=31&IS880=23%3A59&IS888=2017-01-04&~KY~=:
REM ' #(880)- #(887) Текст 8 байт время 1+7
REM ' #(888)- #(898) Текст 11 байт дата 1+10
REM ENDSUB

REM ' *****
REM SUB 'Расшифровка кодов команд от WEB сервера - D
REM ' *****
REM ' Команда выполняется для слейва указанного в C
REM ' C - Текущий номер контроллера WEB ->j
REM ' допустимо 1-8
REM ' D - код команды

REM ' =1 - обновить базу ключей
REM ' =2-выбрать текущий контроллер НОМЕР в C -> j
REM ' =3 - записать время открытия E, в C
REM ' =4 - Выкл Локальное управление в контроллере C
REM ' =5 - Вкл Локальное управление в контроллере C
REM ' =6 - выкл считыватель 1 в контроллере C
REM ' =7 - выкл считыватель 2 в контроллере C

```

```

REM ' =8 - выкл считыватель оба в контроллере С
REM ' =9 - выкл считыватель нет в контроллере С
REM ' =10 - выкл дверь 1 в контроллере С
REM ' =11 - выкл дверь 2 в контроллере С
REM ' =12 - выкл дверь обе в контроллере С
REM ' =13 - выкл дверь - нет в контроллере С
REM ' =14 - открыть дверь 1 в контроллере С
REM ' =15 - открыть дверь 2 в контроллере С
REM ' =16 - открыть дверь обе в контроллере С
REM ' =17 - открыть дверь - нет в контроллере С
REM ' =18 - открыть двери ПОЖАР в контроллере С
REM ' =19 - отменить ПОЖАР в контроллере С
REM ' =20 - ПЕРЕЗАГРУЗКА контроллера С
REM ' =21 - Смена АДРЕСА контроллера С (a-h) на E
REM ' =22 - очистка лог файла [CLEAR_LOG]
REM ' =23 - обновить локальный ключ - не выполняется
REM ' =24 - Добавить ключ
REM '      #(790) = len =10
REM '      #(791)-#(801) = ключ AB153345CD
REM '      E - права доступа ->р
REM ' =25 - Удалить ключ
REM '      #(790)=len=10 #(791)-#(801)=ключ AB153345CD
REM ' =26 - Удалить ключ по номеру E
REM ' =27 - Удалить все ключи
REM ' =28 - добавить пользователя
REM '      #(790)=len=10 #(791)-#(801)=ключ AB153345CD
REM '      #(825)- #(857) имя 32 байта
REM ' =29 - Поиск пользователя
REM '      #(790)=len=10 #(791)-#(801)=ключ AB153345CD
REM '      #(825)- #(857) имя 32 байта
REM ' =30 - очистка базы пользователей
REM ' =31 - получение / установка времени
REM '      показ #(860)- #(879) Текст 19 байт max текущее время
REM '      отправка sendstp.htm?VDD=31&IS880=23%3A59&IS888=2017
REM '      -01-04&~KY~=
REM '      #(880)- #(887) Текст 8 байт время 1+7
REM '      #(888)- #(898) Текст 11 байт дата 1+10
REM ' =32 - удалить пользователя из базы MASTER
REM '      #(790)=len=10 #(791)-#(801)=ключ AB153345CD
REM '      #(825)- #(857) имя 32 байта
REM ' *****
REM ENDSUB

```

```

REM ' *****
REM SUB 'MODBUS Master <-> Slave обмен
REM ' *****
REM ' ***** MODBUS *****
REM ' СТРУКТУРА 8 байт команды / запросы от мастера
REM ' #(976) - код запроса команды Master WR
REM ' #(977) - статус выполнения команды Slave WR
REM ' #(978), #(979), #(980), #(981), #(982), #(983)
REM ' - данные
REM ' МАСТЕР - сначала пишем данные, потом код запроса.
REM ' Слейв-ждет код, обработка данных, удаление входного
REM ' кода, запись данных, выставление статуса
REM ' *****
REM ENDSUB

```

```

REM ' *****
REM SUB ' Лог ФАЙЛ - данные контроллера
REM ' после события SLAVE выставляет данные для опроса
REM ' СТРУКТУРА 8 байт - для ведения логов
REM ' #(984) - код данных
REM ' #(985) - статус выполнения
REM ' #(986), #(987), #(988), #(989), #(990), #(991)
REM ' - данные
REM ' Слейв - выдаем данные, потом код данных.
REM ' #(984) =0 ключ опознан = сумма чисел
REM ' #(984) =10 ключ опознан 1 считыватель

```

```
REM '#(984) =20 ключ опознан 2 считыватель
REM '#(984) =01 открыта 1 дверь
REM '#(984) =02 открыта 2 дверь
REM '#(984) =03 открыто обе двери
REM ' МАСТЕР - опрашивает код, обработка данных, удаление
REM ' кода, запись данных, выставление статуса
REM '
REM ' После обработки мастер #(984) = 0
REM ENDSUB
```

```
REM '*****
REM SUB ' Коды команд MASTER -> SLAVE
REM '*****
REM '      #(976) - код запроса команды
REM ' 0 - НЕТ КОМАНДЫ Слейв ничего не делает
REM ' СЛЕЙВ СБРАСЫВАЕТ В НОЛЬ ПОСЛЕ ОБРАБОТКИ
REM ' 1-ДОБАВИТЬ КЛЮЧ(ОБНОВИТЬ) ДАННЫЕ #(978)-#(983)
REM ' 2-УДАЛИТЬ КЛЮЧ ДАННЫЕ #(978)-#(983)
REM ' 3-УДАЛИТЬ ВСЕ КЛЮЧИ
REM ' 4-УСТАНОВИТЬ ВРЕМЯ - ДАННЫЕ #(978); 0- ЗАПРОС ВРЕМЕНИ
REM ' 5-ЗАПРОС статуса
REM ' КОЛИЧЕСТВО КЛЮЧЕЙ - ДАННЫЕ ОТВЕТ #(978) 0-170
REM ' Время - ОТВЕТ #(979) 1-255
REM ' Блокировка кнопок - ОТВЕТ #(980) 0/1
REM ' Блокировка считывателя - ОТВЕТ #(981) 0/1/2/3
REM ' Блокировка двери - ОТВЕТ #(982) 0/1/2/3
REM ' Состояние дверей - ОТВЕТ #(983) 0/1/2/3
REM ' 6-ПОЛУЧИТЬ КЛЮЧ ПО НОМЕРУ ДИАПАЗОН =(1-КОЛИЧЕСТВО)
REM ' ЗАПРОС ДАННЫЕ НОМЕРА #(978)
REM ' ОТВЕТ ДАННЫЕ КЛЮЧА #(978)-#(983)
REM ' 7-ЗАБЛОКИРОВАТЬ/РАЗБЛ. БАЗУ С КНОПОК #(978)
REM ' ДАННЫЕ #(978) = 1- LOCK 0 = UNLOCK
REM ' 8-ЗАБЛОКИРОВАТЬ СЧИТЫВАТЕЛЬ 0 = UNLOCK 1,2,3 -LOCK
REM ' 9-ЗАБЛОКИРОВАТЬ ДВЕРЬ #(978) = 0-UNLOCK 1,2,3 -LOCK
REM ' 10-ОТКРЫТЬ ДВЕРЬ #(978) = 1,2,3
REM ' 11-ОТКРЫТЬ ДВЕРИ ДО СДЕД. КОМАНДЫ (ALARM)
REM ' 12-ОТМЕНИТЬ КОМАНДУ 11
REM ' 13- ПЕРЕЗАГРУЗКА
REM '*****
REM ENDSUB
```

```
REM '*****
REM SUB ' Коды Ответов SLAVE -> MASTER
REM '*****
REM ' После обработки ответа #(977) = 0 !!!!
REM '*****
REM '      #(977) - статус выполнения команды
REM ' 0 - ДЕЙСТВИЙ НЕ ПРОИЗВОДИЛОСЬ
REM ' Мастер сбрасывает в 0 после проверки статуса
REM ' 1 - 31 КОМАНДА ВЫПОЛНЕНА УСПЕШНО
REM ' 1 - КОМАНДА ВЫПОЛНЕНА УСПЕШНО
REM ' 2 - КЛЮЧ ДОБАВЛЕН
REM ' 3 - КЛЮЧ ПЕРЕПИСАН (ОБНОВЛЕНА ПРАВА)
REM ' 4 - КЛЮЧ УДАЛЕН
REM ' 5 - ВСЕ КЛЮЧИ УДАЛЕНА
REM ' 6 - ВРЕМЯ УСТАНОВЛЕННО / ВЫПОЛНЕНО
REM '
REM ' 32 - 63 ОШИБКА
REM ' 32 - НЕ ВЫПОЛНЕНО
REM ' 33 - КОМАНДА НЕ РАСПОЗНАНА
REM ' 34 - НЕТ МЕСТА
REM ' 35 - НЕТ В БАЗЕ
REM ' 36 - ЕСТЬ В БАЗЕ
REM ' 37 - ОШИБКА ВХОДНЫХ ДАННЫХ
REM ' 38 - нет связи
REM ' 39 - нет такого контроллера
REM ' 40 - пользователь не найден
REM '*****
REM ENDSUB
```

```

REM '*****
REM SUB ' Файловая система WEB SERVER
REM '*****
REM ' adduser.htm Добавление пользователя
REM ' favicon.ico иконка
REM ' index.htm стартовый файл
REM ' logo.gif картинка
REM ' lpass.htm Если авторизация , запрос пароля
REM ' mysite.css стиль сайта
REM ' report.htm простмотр отчетов
REM ' sendadd.htm отправка настройки- имена
REM ' senddev.htm отправка настройки- устройства
REM ' sendrep.htm отправка настройки- отчеты
REM ' sendstp.htm отправка настройки- setup
REM ' senduser.htm отправка настройки- пользователи
REM ' setdev.htm настройка устройств
REM ' setup.htm общие настройки
REM ' setuser.htm настройки пользователей
REM ENDSUB

```

```

REM '*****
REM SUB ' Файловая система - лог файл
REM '*****
REM ' log.htm лог файл событий. Контроллер содаст сам
REM ' Структура файла , все текстом
REM '*****
REM ' Шапка
REM ' "<PRE> Лог событий" + 0x0D,0x0A -17+2
REM ' "Адрес контроллера," - 18
REM ' "Код события," -12
REM ' "Время-Дата," - 11
REM ' "Права доступа," - 14
REM ' "Ключ," -5
REM ' "ФИО пользователя"+ 0x0D,0x0A - 16+2
REM ' итого 97 байт
REM '*****
REM ' ТЕЛО лог файла
REM ' Структура одной записи,если нет -все пробелы в полях
REM ' xxx,yy,12:20:07 15-01-17,63,123456889A,NAME_USER
REM ' xxx - адрес контроллера LEN=3
REM ' yy-событие 1=опознан ключ,открыли;2- не опознан =2
REM ' 12:20:07 15-01-17 дата,время события LEN=17
REM ' 63 - права доступа этого ключа LEN=2
REM ' 123456889A сам ключ LEN=10
REM ' NAME_USER ФИО пользователя LEN=31
REM ' конец строки = 0x0D 0x0A
REM ' итого с разделителями = 72 байт
REM ' ВСЕГО 450 таких записей = 32400 байт
REM '*****
REM ' Подвал
REM ' </PRE>0x0D 0x0A - 6 байт +2 = 8 байт
REM '
REM ' Файл всего = 32505 Для записи в ЛОГ -
REM ' Константа старта = 97 Мах длинна = 32497
REM ENDSUB

```

```

REM '*****
REM SUB ' Файловая система - база контроллера
REM '*****
REM ' base_kn.htm считанная база ключей текущего
REM ' контроллера и пропущенная через соответствие ФИО
REM ' Структура файла, все текстом. Контроллер содаст сам
REM '*****
REM ' Шапка
REM ' "<PRE> База контроллера #1." - 26+2
REM ' "Номер ключа," -12
REM ' "Права доступа," - 14
REM ' "Ключ," 5

```

```

REM ' "ФИО пользователя"+ 0x0D,0x0A - 16+2
REM ' итого 77 байт
REM '*****
REM ' ТЕЛО файла
REM 'xxx,63,123456889A,NAME_USER 0x0D 0x0A
REM 'xxx - НОМЕР КЛЮЧА 1-170          LEN=3
REM '63 - права доступа этого ключа  LEN=2
REM '123456889A сам ключ              LEN=10
REM 'NAME_USER ФИО пользователя      LEN=31
REM ' конец строки = 0x0D 0x0A
REM ' итого с разделителями = 51 байт
REM ' ВСЕГО 170 таких записей = 8670 байт
REM '*****
REM ' Подвал
REM '</PRE>0x0D 0x0A - 6 байт +2 = 8 байт
REM '
REM ' файл всего = 8755 Для записи -
REM ' Константа старта = 77 Мах длинна = 8747

```

REM **ENDSUB**

```

REM '*****
REM SUB ' base.htm файловая система - база пользователей
REM '*****
REM ' Шапка
REM ' "<PRE> База пользователей" + 0x0D,0x0A -24+2
REM ' "Ключ," -5
REM ' "ФИО пользователя"+ 0x0D,0x0A - 16+2
REM ' итого 49 байт
REM '*****
REM ' ТЕЛО лог файла
REM ' Структура одной записи,если нет -все пробелы в полях
REM '123456889A,NAME_USER
REM '123456889A сам ключ              LEN=10
REM 'NAME_USER ФИО пользователя      LEN=31
REM ' конец строки = 0x0D 0x0A
REM ' итого с разделителями = 44 байт
REM ' ВСЕГО 250 таких записей = 11000 байт
REM '*****
REM ' Подвал
REM '</PRE>0x0D 0x0A - 6 байт +2 = 8 байт
REM '
REM ' файл всего = 11057 Для записи в БАЗУ -
REM ' Константа старта = 49 Мах длинна = 11049
REM ENDSUB

```

```

REM '*****
REM SUB ' Подпрограммы и функции
REM '*****

```

```

REM '***** MASTER_SUB.BAS *****
REM ' '[GET_LOCAL_KEY] ' Получить локальный ключ
REM ' Ключ из считывателя -> #(750) = 0/1 нет/да
REM ' #(751)-#(755) = ключ AB153345CD bin
REM ' '[GET_ADDR_BY_NUM]' Получить адрес по номеру
REM ' номер = C; j=C;
REM ' согласно номера i=a,b,c,d,e,f,h,g =адрес контроллера
REM ' Для несуществующего номера i=0
REM ' '[WAIT_READY]' ожидание готовности контроллера
REM ' ДО 1 СЕК ; #(977) = код ответа
REM ' '[GET_STATUS_K] 'Получить статус контроллера
REM ' Input i - Текущий адрес, j - Текущий номер
REM ' OUT k-Количество ключей, l-Время открытия двери.
REM ' m-блок.кнопки, n-блок.считыв., o-блок.двери.
REM ' '[GET_LOG]' Опрос SLAVE, НА ПРЕДМЕТ СОБЫТИЙ
REM ' '[SATRT_GET_LOGS] 'Подпрограмма опроса
REM ' input z - адрес контроллера , B - номер
REM ' '[SET_TEXT_ERROR]
REM ' устанавливает сообщение по коду возврата #(977)

```

```

REM '***** FILE_SUB.BAS *****
REM ' '[CREATE_LOG]
REM ' Создание лог файла, если его нет
REM ' '[CLEAR_LOG]
REM ' Очистка лог файла
REM ' '[FIND_FREE_LOG]
REM ' Указатель "А" на пустое место.
REM ' '[CREATE_SLAVE_BASE]
REM ' Создание/Очистка файла базы ключей SLAVE
REM ' '[CREATE_USER_BASE]
REM ' Создание файла базы пользователей
REM ' '[CLEAR_USER_BASE]
REM ' Очистка файла базы пользователей
REM ' '[FIND_FREE_USER]
REM ' Поиск пустой записи в базе, указатель -> F
REM ' '[FIND_NAME_BY_KEY]' Поиск ФИО ПО КЛЮЧУ
REM ' KEY=@(987),@(988),@(989),@(990),@(991)
REM ' Возвращает #(520) = len, #(521) USER;
REM ' IF NOT #(520) = 0
REM ' '[SAVE_LOGS] 'ЗАПИСЬ СТРОКИ ЛОГ ФАЙЛА В Поз.=A
REM ' Device=z, Kod=#(984), Righ =@(986)
REM ' KEY=@(987),@(988),@(989),@(990),@(991)
REM ' фιο = #(520) = len, #(521) USER ФИО ; A=NEW POZ
REM '*****
REM '
REM '
REM ENDSUB

```

```

0000
NOERR 2
GOTO [START]
REM '** ЕСЛИ ВЫЛЕТЕЛА ПРОГРАММА -> перезагрузка
0002
CLR:CLOSE:REM ' ОЧИСТКА ВСЕГО
BEEP 15,5:PAUSE 200:BEEP 15,5:PAUSE 200:BEEP 15,5
NOERR 2
REM ' DEBUG
STOP
GOTO [START]

```

```

REM '*****
REM SUB ' Создание/Очистка файла базы пользователей
REM '*****
[CREATE_USER_BASE]
REM ' Проверка существования файла
§=DIR{"base.htm"}:IF §=1 THEN GOTO [END_CL_US_BASE]
CREATE 32,11057,"base.htm"
[CLEAR_USER_BASE]
OPENWR "base.htm"
FPRINT "<PRE> База пользователей"
FPRINT "Ключ, ";
FPRINT "ФИО пользователя"

FOR §=0 TO 249
FPRINT "          , ";
FPRINT "          "
NEXT §
FPRINT "</PRE>":F=49:CLOSE
[END_CL_US_BASE]
GOTO [CMD_OK]
REM ENDSUB '*****

```

```

REM '*****
REM SUB ' Поиск пустого места в базе пользователей

```



```

REM ' Возвращает F - ПОЗИЦИЯ ЗАПИСИ 0=НЕТ места
REM ' *****
[FIND_FREE_USER]
OPENRD "base.htm"
FOR F=0 TO 249
§=F*44+49:SEEK §:FREAD@ §
IF §=8224 THEN F=F*44+49:EXFOR [END_FF_USER]
NEXT F
REM ' ИНАЧЕ В НАЧАЛО; §=8224 = два пробела
F=0
[END_FF_USER]
CLOSE
RETURN
REM ' *****

REM ' *****
REM SUB ' Поиск ФИО ПО КЛЮЧУ
REM ' KEY=@(987),@(988),@(989),@(990),@(991)
REM ' Возвращает #(520) = len, #(521) USER;
REM ' G = номер строки
REM ' IF NOT #(520) =0 G = 249
REM ' *****
[FIND_NAME_BY_KEY]
REM ' ПРИГОТОВИМ КЛЮЧ В ТЕКСТЕ = $(570)
VPRINT $(570) @(987),@(988),@(989),@(990),@(991);
[FIND_NAME_BY_KEY_TXT]
DATA $(520), " "
#(520)=0
OPENRD "base.htm"
FOR G=0 TO 249
H=G*44+49:SEEK H:FREAD$ 10,$(555)
IF #(556)=32 THEN GOTO [FIND_NEXT_KEY]
REM ' КЛЮЧ НЕ ПУСТОЙ , СРАВНИМ $(570) И $(555)
I=SCOMP$( $(570) ,$(555) )
IF I=1 THEN EXFOR [READ_NAME]
[FIND_NEXT_KEY]
NEXT G
GOTO [EX_FIND_NAME_BY_KEY]
[READ_NAME]
H=G*44+49+11
SEEK H:FREAD$ 31,$(520):#(520)=31
[EX_FIND_NAME_BY_KEY]
CLOSE
RETURN
REM ' *****

REM ' *****
REM SUB ' ЗАПИСЬ СТРОКИ ЛОГ ФАЙЛА
REM ' *****
REM ' Device=z, Kod=#(984), Righ =@(986)
REM ' KEY=@(987),@(988),@(989),@(990),@(991)
REM ' фИО = #(520) = len, #(521) USER ФИО
[SAVE_LOGS]
IF A>=32497 THEN A=97
IF A<97 THEN A=97
OPENWR "log.htm":SEEK A
SDT $(0):FPRINT %3,z," ";
FPRINT %2,#(984)," ";
FPRINT *$(0)," ";
FPRINT %2,#(986)," ";
FPRINT @(987),@(988),@(989),@(990),@(991)," ";
K=LOC{}:CLOSE

GOSUB [FIND_NAME_BY_KEY]
IF #(520)=31 THEN GOTO [SKIP_EMPTY_USER]
DATA $(520), " "
#(520)=31
[SKIP_EMPTY_USER]

```

```

OPENWR "log.htm":SEEK K
FPRINT$ #(520)
A=LOC{ }:A=A+2:IF A>=32497 THEN A=97
CLOSE
RETURN
REM '*****

REM '*****
REM SUB ' Создание лог файла, если его нет на 512 записей
REM '*****
[CREATE_LOG]
REM ' Проверка существования файла
§=DIR{"log.htm"}:IF §=1 THEN GOTO [FIND_FREE_LOG]
CREATE 32,32505,"log.htm"
[CLEAR_LOG]
OPENWR "log.htm"
FPRINT "<PRE> Лог событий"
FPRINT "Адрес контроллера,";
FPRINT "Код События,";
FPRINT "Время-Дата,";
FPRINT "Права доступа,";
FPRINT "Ключ,";
FPRINT "ФИО пользователя"

FOR §=0 TO 449
FPRINT " , , " ;
FPRINT " , , , " ;
FPRINT " "
NEXT §

FPRINT "</PRE>"
REM ' НАЧАЛЬНАЯ УСТАНОВКА НАЧАЛА ЗАПИСИ
REM ' А позиция записи лога A=LOC{ } SEEK А
REM ' допустимо 97 - 32497
A=97
GOTO [END_INIT_FILE]
REM ' *****
REM ' файл есть ИЩЕМ ПУСТОЕ МЕСТО И ТУДА УКАЗАТЕЛЬ ="А"
[FIND_FREE_LOG]
OPENRD "log.htm"
FOR A=0 TO 449
§=A*72+97:SEEK §:FREAD@ §
IF §=8224 THEN A=A*72+97:EXFOR [END_INIT_FILE]
NEXT A
REM ' ИНАЧЕ В НАЧАЛО; §=8224 = два пробела
A=97
[END_INIT_FILE]
CLOSE:GOSUB [CMD_OK]
RETURN
REM '*****

REM '*****
REM SUB ' Создание/Очистка файла базы ключей SLAVE
REM '*****
[CREATE_SLAVE_BASE]
REM ' Проверка существования файла
§=DIR{"base_kn.htm"}:IF §=1 THEN GOTO [CLEAR_SLAVE_BASE]
REM ' создадим
CREATE 32,8755,"base_kn.htm"
[CLEAR_SLAVE_BASE]
OPENWR "base_kn.htm"
FPRINT "<PRE> База контроллера #0."
FPRINT "Номер ключа,";
FPRINT "Права доступа,";
FPRINT "Ключ,";
FPRINT "ФИО пользователя"

FOR §=0 TO 169
FPRINT " , , , " ;

```

```

FPRINT "
NEXT §
FPRINT "</PRE>":CLOSE
RETURN
REM '*****
REM '*****
REM SUB ' Получить локальный ключ
REM '*****
REM 'Ключ из считывателя
REM '#(750) = 0/1
REM '#(751)-#(755) = ключ AB153345CD
[GET_LOCAL_KEY]
#(750)=RFID{1}
IF #(750)=0 THEN GOTO [KEY_BAD]
VPRINT $(720) @(751),@(752),@(753),@(754),@(755);
RETURN

REM '*****
REM SUB ' Получить адрес по номеру
REM '*****
[GET_ADDR_BY_NUM]
j=C
IF C=1 THEN i=a:GOTO [EXIT_GET_ADDR_BY_NUM]
IF C=2 THEN i=b:GOTO [EXIT_GET_ADDR_BY_NUM]
IF C=3 THEN i=c:GOTO [EXIT_GET_ADDR_BY_NUM]
IF C=4 THEN i=d:GOTO [EXIT_GET_ADDR_BY_NUM]
IF C=5 THEN i=e:GOTO [EXIT_GET_ADDR_BY_NUM]
IF C=6 THEN i=f:GOTO [EXIT_GET_ADDR_BY_NUM]
IF C=7 THEN i=g:GOTO [EXIT_GET_ADDR_BY_NUM]
IF C=8 THEN i=h:GOTO [EXIT_GET_ADDR_BY_NUM]
i=0
[EXIT_GET_ADDR_BY_NUM]
RETURN

REM '***** ADDRESS = i *****
REM SUB ' ожидание готовности контроллера ДО 1 СЕК
[WAIT_READY]
FOR §=0 TO 100
PAUSE 10:§=MDRD{i,#(977)}:IF §=0 THEN GOTO [WAIT_READY_1]
IF #(977)>0 THEN §-=EXFOR [WAIT_READY_2]
[WAIT_READY_1]
NEXT §
#(977)=32
[WAIT_READY_2]
§=#(977):#(977)=0:§=MDWR{i,#(977)}
IF §=0 THEN §-=§=38
#(977)=§
RETURN
REM '*****

REM '*****
REM SUB ' Получить статус контроллера
REM '*****
REM ' i - Текущий адрес контроллера
REM ' j - Текущий номер контроллера
REM '*****
[GET_STATUS_K]
k=0:l=0:m=0:n=0:o=0
REM ' НЕ ЗАПРАШИВАЕМ НЕСУЩЕСТВУЮЩИЕ
IF i=0 THEN GOTO [CMD_ERR_NO_KONTR]
IF i>127 THEN GOTO [CMD_ERR_NO_KONTR]
REM ' начальный сброс команд
#(977)=0:§=MDWR{i,#(977)}
IF §=0 THEN GOTO [CMD_ERR_LINK]
REM ' Дали команду на подсчет количества ключей
#(976)=5:§=MDWR{i,#(976)}
IF §=0 THEN GOTO [CMD_ERR_LINK]
REM ' получить СТАТУС
GOSUB [WAIT_READY]

```

```

IF #(977)>31 THEN GOTO [SET_TEXT_ERROR]
REM ' КОЛИЧЕСТВО КЛЮЧЕЙ - #(978)
§=MDRD{i,#(978)}:IF §=0 THEN GOTO [CMD_ERR_LINK]
§=MDRD{i,#(979)}:IF §=0 THEN GOTO [CMD_ERR_LINK]
§=MDRD{i,#(980)}:IF §=0 THEN GOTO [CMD_ERR_LINK]
§=MDRD{i,#(981)}:IF §=0 THEN GOTO [CMD_ERR_LINK]
§=MDRD{i,#(982)}:IF §=0 THEN GOTO [CMD_ERR_LINK]
k=#(978):l=#(979):m=#(980):n=#(981):o=#(982)
GOSUB [CMD_OK]
RETURN
REM '*****

REM '*****
REM SUB ' заполнить текст ошибки
REM '*****

[SET_TEXT_ERROR]
IF #(977)=1 THEN GOTO [CMD_OK]
IF #(977)=2 THEN GOTO [CMD_OK_ADD]
IF #(977)=3 THEN GOTO [CMD_OK_UPDATE]
IF #(977)=4 THEN GOTO [CMD_OK_DEL]
IF #(977)=5 THEN GOTO [CMD_OK_ALL_DEL]
IF #(977)=6 THEN GOTO [CMD_OK_TIME]

IF #(977)<32 THEN GOTO [CMD_OK]

IF #(977)=33 THEN GOTO [CMD_ERR_CMD]
IF #(977)=34 THEN GOTO [CMD_ERR_FREE]
IF #(977)=35 THEN GOTO [CMD_ERR_KEY]
IF #(977)=36 THEN GOTO [CMD_ERR_KEY_PRESENT]
IF #(977)=37 THEN GOTO [CMD_ERR_DATA]
IF #(977)=38 THEN GOTO [CMD_ERR_LINK]
IF #(977)=39 THEN GOTO [CMD_ERR_NO_KONTR]
IF #(977)=40 THEN GOTO [CMD_ERR_USER]
GOTO [CMD_ERR]
REM ENDSUB '*****

REM '*****
REM SUB ' Опрос контроллеров, НА ПРЕДМЕТ СОБЫТИЙ
REM '*****
REM ' ОПРОС ВСЕХ ПО ОЧЕРЕДИ
[GET_LOG]
z=a:B=1
GOSUB [SATRT_GET_LOGS]
z=b:B=2
GOSUB [SATRT_GET_LOGS]
z=c:B=3
GOSUB [SATRT_GET_LOGS]
z=d:B=4
GOSUB [SATRT_GET_LOGS]
z=e:B=5
GOSUB [SATRT_GET_LOGS]
z=f:B=6
GOSUB [SATRT_GET_LOGS]
z=g:B=7
GOSUB [SATRT_GET_LOGS]
z=h:B=8
GOSUB [SATRT_GET_LOGS]
RETURN

REM '*****
REM SUB ' Подпрограмма опроса по адресу, номеру
REM ' z - адрес контроллера , B - номер
REM '*****

[SATRT_GET_LOGS]
IF z=0 THEN GOTO [SKP_GET_LOGS]
IF z>127 THEN GOTO [SKP_GET_LOGS]
REM ' ЗАПРОС
§=MDRD{z,#(984)}:IF §=0 THEN GOTO [ERR_GET_LOGS]
IF #(984)=0 THEN GOTO [GET_NO_DATA]

```

```

REM ' ПОЛУЧИМ ДАННЫЕ
§=MDRD{z,#(986)}:IF §=0 THEN GOTO [ERR_GET_LOGS]
§=MDRD{z,#(987)}:IF §=0 THEN GOTO [ERR_GET_LOGS]
§=MDRD{z,#(988)}:IF §=0 THEN GOTO [ERR_GET_LOGS]
§=MDRD{z,#(989)}:IF §=0 THEN GOTO [ERR_GET_LOGS]
§=MDRD{z,#(990)}:IF §=0 THEN GOTO [ERR_GET_LOGS]
§=MDRD{z,#(991)}:IF §=0 THEN GOTO [ERR_GET_LOGS]
REM ' НА ДИСПЛЕЙ
CLS:LPRINT "Dev=",z," Kod=",#(984)
AT 64:LPRINT "KEY="
LPRINT @(986),@(987),@(988),@(989),@(990),@(991)

GOSUB [SAVE_LOGS]

REM ' УДАЛИМ ФЛАГ ДАННЫХ, НЕ ПРОВЕРЯЕМ ОШИБКУ
#(984)=0:§=MDWR{16,#(984)}
IF §=0 THEN GOTO [ERR_GET_LOGS]

REM ' НЕ БЫЛО СОБЫТИЙ ИЛИ ВСЕ ХОРОШО - ТЕКСТ
[GET_NO_DATA]
CASE B,1,[K1_OK],2,[K2_OK],3,[K3_OK],4,[K4_OK]
CASE B,5,[K5_OK],6,[K6_OK],7,[K7_OK],8,[K8_OK]
GOTO [END_GET_LOGS]

REM ' КОНТРОЛЛЕР НЕ ОТВЕЧАЕТ - ТЕКСТ
[ERR_GET_LOGS]
CASE B,1,[K1_BAD],2,[K2_BAD],3,[K3_BAD],4,[K4_BAD]
CASE B,5,[K5_BAD],6,[K6_BAD],7,[K7_BAD],8,[K8_BAD]
GOTO [END_GET_LOGS]

REM ' КОНТРОЛЛЕР НЕ ПОДКЛЮЧЕН - ТЕКСТ
[SKP_GET_LOGS]
CASE B,1,[K1_NO],2,[K2_NO],3,[K3_NO],4,[K4_NO]
CASE B,5,[K5_NO],6,[K6_NO],7,[K7_NO],8,[K8_NO]

[END_GET_LOGS]
RETURN
REM ' *****

REM ' *****
REM ' ИНИЦИАЛИЗАЦИЯ текстовых переменных
REM ' *****
[STATUS_READY]
DATA $(735),"Готов" :RETURN
[STATUS_BUSY]
DATA $(735),"Обработка" :RETURN

REM ' *****
[CMD_OK]
DATA $(770),"Удачное завершение" :RETURN
[CMD_OK_ADD]
DATA $(770),"Ключ добавлен" :RETURN
[CMD_OK_UPDATE]
DATA $(770),"Ключ обновлен" :RETURN
[CMD_OK_DEL]
DATA $(770),"Ключ удален" :RETURN
[CMD_OK_ALL_DEL]
DATA $(770),"Все ключи удалены" :RETURN
[CMD_OK_TIME]
DATA $(770),"Время установлено" :RETURN
[CMD_OK_USER]

```

```
DATA $(770), "Пользов. добавлен " :RETURN

[CMD_ERR_CMD]
DATA $(770), "Ошибка команды " :RETURN
[CMD_ERR_DATA]
DATA $(770), "Ошибка данных " :RETURN
[CMD_ERR]
DATA $(770), "Не выполнено " :RETURN
[CMD_ERR_FREE]
DATA $(770), "Нет места " :RETURN
[CMD_ERR_KEY]
DATA $(770), "Ключ не найден " :RETURN
[CMD_ERR_USER]
DATA $(770), "Пользов. не найден " :RETURN
[CMD_ERR_KEY_PRESENT]
DATA $(770), "Ключ есть в базе " :RETURN
[CMD_ERR_USER_PRESENT]
DATA $(770), "Пользов. есть в базе" :RETURN
[CMD_ERR_NO_KONTR]
DATA $(770), "Нет такого контрол." :RETURN
[CMD_ERR_BUSY]
DATA $(770), "Контроллер занят " :RETURN
[CMD_ERR_LINK]
DATA $(770), "Нет связи с контр. " :RETURN
REM '*****'
[KEY_BAD]
DATA $(720), "- - - - -" :RETURN

REM '*****'
[K1_OK]
DATA $(600), "Доступен " :RETURN
[K1_NO]
DATA $(600), "Не подключен" :RETURN
[K1_BAD]
DATA $(600), "Нет связи " :RETURN

[K2_OK]
DATA $(615), "Доступен " :RETURN
[K2_NO]
DATA $(615), "Не подключен" :RETURN
[K2_BAD]
DATA $(615), "Нет связи " :RETURN

[K3_OK]
DATA $(630), "Доступен " :RETURN
[K3_NO]
DATA $(630), "Не подключен" :RETURN
[K3_BAD]
DATA $(630), "Нет связи " :RETURN

[K4_OK]
DATA $(645), "Доступен " :RETURN
[K4_NO]
DATA $(645), "Не подключен" :RETURN
[K4_BAD]
DATA $(645), "Нет связи " :RETURN

[K5_OK]
DATA $(660), "Доступен " :RETURN
[K5_NO]
DATA $(660), "Не подключен" :RETURN
[K5_BAD]
DATA $(660), "Нет связи " :RETURN

[K6_OK]
DATA $(675), "Доступен " :RETURN
[K6_NO]
DATA $(675), "Не подключен" :RETURN
[K6_BAD]
```

```

DATA $(675), "Нет связи" :RETURN

[К7_ОК]
DATA $(690), "Доступен" :RETURN
[К7_НО]
DATA $(690), "Не подключен":RETURN
[К7_БАД]
DATA $(690), "Нет связи" :RETURN

[К8_ОК]
DATA $(705), "Доступен" :RETURN
[К8_НО]
DATA $(705), "Не подключен":RETURN
[К8_БАД]
DATA $(705), "Нет связи" :RETURN
REM '*****
REM '*****
REM '*****
REM SUB ' Парсер команд WEB сервера
REM '*****

REM '*****
REM ' Подпрограммы Парсера команд WEB сервера
REM '*****

REM '*****
REM SUB ' Послать команду в SLAVE #(978), #(976)
[SEND_1X_COMMAND]
GOSUB [GET_ADDR_BY_NUM]
IF i=0 THEN GOTO [CMD_ERR_NO_KONTR]
IF i>127 THEN GOTO [CMD_ERR_NO_KONTR]
#(977)=0:§=MDWR{i, #(977)}:§-
REM ' отправим
§=MDWR{i, #(978)}:IF §=0 THEN GOTO [CMD_ERR_LINK]
§=MDWR{i, #(976)}:IF §=0 THEN GOTO [CMD_ERR_LINK]
GOSUB [WAIT_READY]
REM ' если ошибка -выход с сообщением
IF #(977)>31 THEN GOTO [SET_TEXT_ERROR]
§=#(977)
GOSUB [GET_STATUS_K]
IF l=0 THEN §=:GOTO [CMD_ERR]
#(977)=§
GOSUB [SET_TEXT_ERROR]
RETURN

REM '*****
REM SUB '=1 - обновить базу ключей КОНТРОЛЛЕРА С
[UPD_BASE_KEY]
GOSUB [GET_ADDR_BY_NUM]
IF i=0 THEN GOTO [CMD_ERR_NO_KONTR]
IF i>127 THEN GOTO [CMD_ERR_NO_KONTR]
REM ' i= adres C= number
REM ' ПРОВЕРИМ ДОСТУПНОСТЬ И ПОЧИСТИМ БАЗУ
j=C:§=MDRD{i, #(976)}:IF §=0 THEN GOTO [CMD_ERR_LINK]
IF #(976)<>0 THEN GOTO [CMD_ERR_BUSY]
GOSUB [CLEAR_SLAVE_BASE]
REM ' Получим STATUS
GOSUB [GET_STATUS_K]
REM ' ЕСЛИ ВРЕМЯ =0 ТО ошибка
IF l=0 THEN GOTO [CMD_ERR_LINK]
REM ' ЕСЛИ КЛЮЧЕЙ НЕТ
IF k=0 THEN GOTO [CMD_ERR_KEY]
REM ' k - Количество ключей в текущем контроллере
IF k>170 THEN GOTO [CMD_ERR_DATA]
OPENWR "base_kn.htm"
REM ' запишем адрес онтроллера
SEEK 24:FPRINT %1, j;
SEEK 77:z=0:REM ' z - СЧЕТЧИК ЦИКЛА

```

```

[CIKLE_READ_KEY]
z=z+1:IF z>k THEN GOTO [EXIT_READ]
REM ' Дали команду 6 на запрос ключа НОМЕРОМ z
#(978)=z:#(976)=6
§=MDWR{i,#(978)}:IF §=0 THEN GOTO [CMD_ERR_LINK]
§=MDWR{i,#(976)}:IF §=0 THEN GOTO [CMD_ERR_LINK]
GOSUB [WAIT_READY]
REM ' если ошибка -выход с сообщением
IF #(977)>31 THEN GOTO [SET_TEXT_ERROR]
REM ' ЕСТЬ КЛЮЧ- ДАННЫЕ КЛЮЧА #(978)-#(983)
§=MDRD{i,#(978)}:IF §=0 THEN GOTO [CMD_ERR_LINK]
§=MDRD{i,#(979)}:IF §=0 THEN GOTO [CMD_ERR_LINK]
§=MDRD{i,#(980)}:IF §=0 THEN GOTO [CMD_ERR_LINK]
§=MDRD{i,#(981)}:IF §=0 THEN GOTO [CMD_ERR_LINK]
§=MDRD{i,#(982)}:IF §=0 THEN GOTO [CMD_ERR_LINK]
§=MDRD{i,#(983)}:IF §=0 THEN GOTO [CMD_ERR_LINK]
REM ' SAVE KEY
FPRINT %3,z,"";
FPRINT %2,#(978),"";
FPRINT @(979),@(980),@(981),@(982),@(983),"";
K=LOC{}:CLOSE
REM ' FIND user
#(987)=#(979):#(988)=#(980):#(989)=#(981)
#(990)=#(982):#(991)=#(983):GOSUB [FIND_NAME_BY_KEY]
OPENWR "base_kn.htm":SEEK K
REM ' SAVE User
#(520)=31:FPRINT$ #(520):I=LOC{}:I=I+2:SEEK I
GOTO [CIKLE_READ_KEY]
REM ' конец цикла
[EXIT_READ]
CLOSE:GOSUB [CMD_OK]
RETURN
REM '*****

REM '*****
REM SUB '=2-выбрать текущий контроллер НОМЕР в С -> j
[SET_DEFAULT_SLAVE]
j=C:GOSUB [GET_ADDR_BY_NUM]
GOSUB [GET_STATUS_K]
RETURN

REM '*****
REM SUB 'D=21 - Смена АДРЕСА контроллера С на Е
[UPD_ADDR_C_E]
j=C
IF C=1 THEN i=E:a=E:WREE a,0:GOTO [GET_STATUS_K]
IF C=2 THEN i=E:b=E:WREE b,2:GOTO [GET_STATUS_K]
IF C=3 THEN i=E:c=E:WREE c,4:GOTO [GET_STATUS_K]
IF C=4 THEN i=E:d=E:WREE d,6:GOTO [GET_STATUS_K]
IF C=5 THEN i=E:e=E:WREE e,8:GOTO [GET_STATUS_K]
IF C=6 THEN i=E:f=E:WREE f,10:GOTO [GET_STATUS_K]
IF C=7 THEN i=E:g=E:WREE g,12:GOTO [GET_STATUS_K]
IF C=8 THEN i=E:h=E:WREE h,14:GOTO [GET_STATUS_K]
C=1:j=1:i=a:p=63
GOSUB [GET_STATUS_K]
RETURN
REM '*****

REM '*****
REM SUB '=24 - Добавить ключ С , #(790)
[ADD_KEY_SLAVE]
REM ' IS790=130070CE8 &VDC=1&VDE=63&VDD=24
REM '#(976)=1 - add
REM '#(976)=2 - del
REM ' Е - права доступа ->p
REM '#(790) = len =10
REM '#(791)-#(801) = ключ AB153345CD
REM ' 1-ДОБАВИТЬ КЛЮЧ(ОБНОВИТЬ) ДАННЫЕ #(978)-#(983)
REM 'D=24 C=1 E=63 &ID790=1122334455

```



```

#(977)=0:§=MDWR{i,#(977)}:§-
IF #(790)<>10 THEN GOTO [CMD_ERR_DATA]
GOSUB [GET_ADDR_BY_NUM]
IF i=0 THEN GOTO [CMD_ERR_NO_KONTR]
IF i>127 THEN GOTO [CMD_ERR_NO_KONTR]

#(979)=VAL{@(790),1}
#(980)=VAL{@(790),3}:#(981)=VAL{@(790),5}
#(982)=VAL{@(790),7}:#(983)=VAL{@(790),9}

§=MDWR{i,#(978)}:IF §=0 THEN GOTO [CMD_ERR_LINK]
§=MDWR{i,#(979)}:IF §=0 THEN GOTO [CMD_ERR_LINK]
§=MDWR{i,#(980)}:IF §=0 THEN GOTO [CMD_ERR_LINK]
§=MDWR{i,#(981)}:IF §=0 THEN GOTO [CMD_ERR_LINK]
§=MDWR{i,#(982)}:IF §=0 THEN GOTO [CMD_ERR_LINK]
§=MDWR{i,#(983)}:IF §=0 THEN GOTO [CMD_ERR_LINK]
REM ' данные загрузили, командуем
§=MDWR{i,#(976)}:IF §=0 THEN GOTO [CMD_ERR_LINK]

GOSUB [WAIT_READY]
REM ' если ошибка -выход с сообщением
IF #(977)>31 THEN GOTO [SET_TEXT_ERROR]

§=#(977)
GOSUB [GET_STATUS_K]
IF l=0 THEN §=:GOTO [CMD_ERR]
#(977)=§
GOSUB [SET_TEXT_ERROR]
RETURN
REM ' *****

REM ' *****
REM SUB '=29 - добавить пользователя в базу MASTER
REM '      #(790)=len=10 #(791)-#(801)=ключ AB153345CD
REM '      #(825)- #(857) имя 32 байта
[ADD_USER_MASTER]
#(977)=0:§=MDWR{i,#(977)}:§-
IF #(790)<>10 THEN GOTO [CMD_ERR_DATA]
IF #(825)<3 THEN GOTO [CMD_ERR_DATA]
IF #(825)>31 THEN GOTO [CMD_ERR_DATA]
REM ' Ищем такой же ключ
#(570)=#(790):#(571)=#(791):#(572)=#(792):#(573)=#(793)
#(574)=#(794):#(575)=#(795):#(576)=#(796):#(577)=#(797)
#(578)=#(798):#(579)=#(799):#(580)=#(800):#(581)=#(801)
GOSUB [FIND_NAME_BY_KEY_TXT]
IF #(520)<>0 THEN GOTO [CMD_ERR_KEY_PRESENT]
REM ' Ищем Такого пользователя
OPENRD "base.htm"
FOR G=0 TO 249
H=G*44+49+11
SEEK H:FREAD$ 31,$(520):#(520)=31
REM ' сравним пользователя
I=SCOMP${$(825),$(520)}
IF I=1 THEN EXFOR [CMD_ERR_USER_PRESENT]
NEXT G
CLOSE
REM ' Ищем пустое место
GOSUB [FIND_FREE_USER]
REM ' Возвращает F - ПОЗИЦИЯ ЗАПИСИ
IF F=0 THEN GOTO [CMD_ERR_FREE]
REM ' Добавим
OPENWR "base.htm"
SEEK F
FPRINT *$(790)," ";
F=LOC{}
FPRINT " ";
SEEK F
FPRINT *$(825);
CLOSE:GOTO [CMD_OK_USER]

```

REM ENDSUB

```
REM '*****
REM SUB '=29 - Поиск пользователя в базе MASTER
REM '      #(790)=len=10 #(791)-#(801)=ключ AB153345CD
REM '      #(825)- #(857) имя 32 байта
REM '      Результаты поиска пользователя
REM '      пишет значения или обнуляет, если нет
REM '      #(814)- #(824) ключ 11 байт
REM '      #(900)- #(931) имя 32 байта
```

[FIND_USER_MASTER]

DATA \$(814), "- - - -"

DATA \$(900), " "

#(977)=0:§=MDWR{i,#(977)}:§-

IF #(790)<>10 THEN GOTO [FIND_USER_MASTER_NAME]

REM ' ПОИСК ПО КЛЮЧУ

#(570)=#(790):#(571)=#(791):#(572)=#(792):#(573)=#(793)

#(574)=#(794):#(575)=#(795):#(576)=#(796):#(577)=#(797)

#(578)=#(798):#(579)=#(799):#(580)=#(800):#(581)=#(801)

GOSUB [FIND_NAME_BY_KEY_TXT]

IF #(520)=0 THEN GOTO [CMD_ERR_USER]

#(814)=#(790):#(815)=#(791):#(816)=#(792):#(817)=#(793)

#(818)=#(794):#(819)=#(795):#(820)=#(796):#(821)=#(797)

#(822)=#(798):#(823)=#(799):#(824)=#(800)

H=520:I=900

FOR §=0 TO 31

#(I)=#(H):H=H+1:I=I+1

NEXT §

GOTO [CMD_OK]

[FIND_USER_MASTER_NAME]

REM ' поиск по имени

REM ' Ищем Такого пользователя

OPENRD "base.htm"

FOR G=0 TO 249

H=G*44+49+11

SEEK H:FREAD\$ 31,\$(520):#(520)=31

REM ' сравним пользователя

I=SCOMP\${\$(825),\$ (520)}

IF I=1 THEN EXFOR [FIND_USER_PRESENT]

NEXT G

CLOSE

GOTO [CMD_ERR_USER]

[FIND_USER_PRESENT]

REM ' read key

H=G*44+49:SEEK H:FREAD\$ 10,\$(814):#(814)=10

CLOSE

IF #(815)=32 THEN GOTO [CMD_ERR_KEY]

H=520:I=900

FOR §=0 TO 31

#(I)=#(H):H=H+1:I=I+1

NEXT §

GOTO [CMD_OK]

REM ENDSUB

```
REM '*****
REM SUB '=32 - удалить пользователя из базы MASTER
REM '      #(790)=len=10 #(791)-#(801)=ключ AB153345CD
REM '      #(825)- #(857) имя 32 бай
```

[DEL_USER_MASTER]

#(977)=0:§=MDWR{i,#(977)}:§-

IF #(790)<>10 THEN GOTO [CMD_ERR_DATA]

IF #(825)<3 THEN GOTO [CMD_ERR_DATA]

IF #(825)>31 THEN GOTO [CMD_ERR_DATA]

```

#(570)=#(790):#(571)=#(791):#(572)=#(792):#(573)=#(793)
#(574)=#(794):#(575)=#(795):#(576)=#(796):#(577)=#(797)
#(578)=#(798):#(579)=#(799):#(580)=#(800):#(581)=#(801)
GOSUB [FIND_NAME_BY_KEY_TXT]
IF #(520)=0 THEN GOTO [CMD_ERR_KEY]
REM ' сравним пользователя
I=SCOMP${$(825),$(520)}
IF I=0 THEN GOTO [CMD_ERR_USER]
REM ' совпал ключ и ФИО - DEL
OPENWR "base.htm"
H=G*44+49:SEEK H:FPRINT "          ", " ;
FPRINT "          "
CLOSE:GOTO [CMD_OK_DEL]
REM ENDSUB

REM '*****
REM SUB '=31 - получение / установка времени
REM ' 880=LEN+23:59
REM ' 888=LEN+2017-01-04
REM '      #(880)- #(887) Текст 8 байт время 1+7
REM '      #(888)- #(898) Текст 11 байт дата 1+10
[SET_TIME_MASTER]
IF #(880)<>5 THEN GOTO [CMD_ERR_DATA]
IF #(888)<>10 THEN GOTO [CMD_ERR_DATA]

#(590)=VAL{$(880),1}:IF #(590)>23 THEN GOTO [CMD_ERR_DATA]
#(591)=VAL{$(880),4}:IF #(591)>59 THEN GOTO [CMD_ERR_DATA]

#(592)=VAL{$(888),3}:IF #(592)>50 THEN GOTO [CMD_ERR_DATA]
IF #(592)<17 THEN GOTO [CMD_ERR_DATA]
#(593)=VAL{$(888),6}:IF #(593)>12 THEN GOTO [CMD_ERR_DATA]
IF #(593)<1 THEN GOTO [CMD_ERR_DATA]
#(594)=VAL{$(888),9}:IF #(593)>31 THEN GOTO [CMD_ERR_DATA]
IF #(594)<1 THEN GOTO [CMD_ERR_DATA]
REM ' set time
YEAR #(592):MONTH #(593):DAY #(594)
HOUR #(590):MIN #(591):SEC 0
REM '      показ #(860)- #(879) Текст 19 байт мах текущее время
SDT $(860)
GOTO [CMD_OK_TIME]
REM ENDSUB

REM ' C - номер контроллера
REM ' D - команда
REM ' E - параметры
REM ' i - Текущий адрес контроллера web server
REM ' j - Текущий номер контроллера web server
REM ' ключ #(790)=len=10 #(791)-#(801)=AB153345CD
REM ' Пользователь #(825)-#(857)имя 32 байта #(825)=len
REM '#(880)- #(887) Текст 8 байт время 1+7 23%3A59
REM '#(888)- #(898) Текст 11 байт дата 1+10 2017-01-04

REM ' *****
[PARSER_WEB_COMMAND]
REM '=1 - обновить базу ключей
IF D=1 THEN GOTO [UPD_BASE_KEY]

REM '=2-выбрать текущий контроллер НОМЕР в C -> j
IF D=2 THEN GOTO [SET_DEFAULT_SLAVE]

REM '=3 - записать время открытия E, в C
REM 'УСТАНОВИТЬ ВРЕМЯ - ДАННЫЕ #(978)
IF D=3 THEN #(978)=E:#(976)=4:GOTO [SND_1X_COMMAND]

```

```

REM '=4-Выкл Локальное управление в контроллере С
REM '=5-Вкл Локальное управление в контроллере С
REM ' ДАННЫЕ #(978) = 1- LOCK 0 = UNLOCK
IF D=4 THEN #(978)=1:#(976)=7:GOTO [SND_1X_COMMAND]
IF D=5 THEN #(978)=0:#(976)=7:GOTO [SND_1X_COMMAND]

REM '=6 - выкл считыватель 1 в контроллере С
REM ' =7 - выкл считыватель 2 в контроллере С
REM ' =8 - выкл считыватель оба в контроллере С
REM ' =9 - выкл считыватель нет в контроллере С
REM ' 8-ЗАБЛОКИР СЧИТЫВАТЕЛЬ 0 = UNLOCK 1,2,3 -LOCK
IF D=6 THEN #(978)=1:#(976)=8:GOTO [SND_1X_COMMAND]
IF D=7 THEN #(978)=2:#(976)=8:GOTO [SND_1X_COMMAND]
IF D=8 THEN #(978)=3:#(976)=8:GOTO [SND_1X_COMMAND]
IF D=9 THEN #(978)=0:#(976)=8:GOTO [SND_1X_COMMAND]

REM ' =10 - выкл дверь 1 в контроллере С
REM ' =11 - выкл дверь 2 в контроллере С
REM ' =12 - выкл дверь обе в контроллере С
REM ' =13 - выкл дверь - нет в контроллере С
REM ' 9-ЗАБЛОКИРОВАТЬ ДВЕРЬ #(978) = 0-UNLOCK 1,2,3 -LOCK
IF D=10 THEN #(978)=1:#(976)=9:GOTO [SND_1X_COMMAND]
IF D=11 THEN #(978)=2:#(976)=9:GOTO [SND_1X_COMMAND]
IF D=12 THEN #(978)=3:#(976)=9:GOTO [SND_1X_COMMAND]
IF D=13 THEN #(978)=0:#(976)=9:GOTO [SND_1X_COMMAND]

REM ' =14 - открыть дверь 1 в контроллере С
REM ' =15 - открыть дверь 2 в контроллере С
REM ' =16 - открыть дверь обе в контроллере С
REM ' =17 - открыть дверь - нет в контроллере С
REM ' 10-ОТКРЫТЬ ДВЕРЬ #(978) = 1,2,3
IF D=14 THEN #(978)=1:#(976)=10:GOTO [SND_1X_COMMAND]
IF D=15 THEN #(978)=2:#(976)=10:GOTO [SND_1X_COMMAND]
IF D=16 THEN #(978)=3:#(976)=10:GOTO [SND_1X_COMMAND]
IF D=17 THEN D=0 GOTO [CMD_OK]

REM ' =18 - открыть двери ПОЖАР в контроллере С
REM ' =19 - отменить ПОЖАР в контроллере С
IF D=18 THEN #(978)=0:#(976)=11:GOTO [SND_1X_COMMAND]
IF D=19 THEN #(978)=0:#(976)=12:GOTO [SND_1X_COMMAND]

REM ' =20 - ПЕРЕЗАГРУЗКА контроллера С
IF D=20 THEN #(978)=0:#(976)=13:GOTO [SND_1X_COMMAND]

REM ' =21 - Смена АДРЕСА контроллера С на Е
IF D=21 THEN GOTO [UPD_ADDR_C_E]

REM ' =22 - очистка лог файла [CLEAR_LOG]
IF D=22 THEN GOTO [CLEAR_LOG]

REM ' =23 - обновить локальный ключ
REM ' И ТАК АВТОМАТОМ ОБНОВЛЯЕТСЯ
IF D=23 THEN GOTO [CMD_OK]

REM ' =24 - Добавить ключ С , #(790)
REM ' Е - права доступа ->р
REM '#(790) = len =10
REM '#(791)-#(801) = ключ AB153345CD
REM ' 1-ДОБАВИТЬ КЛЮЧ(ОБНОВИТЬ) ДАННЫЕ #(978)-#(983)
REM 'D=24 C=1 E=63 &ID790=1122334455
IF D=24 THEN #(976)=1:#(978)=E:p=E:GOTO [ADD_KEY_SLAVE]

REM ' =25 - Удалить ключ С , #(790)
REM ' ДАННЫЕ #(978)-#(983)
IF D=25 THEN #(976)=2:#(978)=255:GOTO [ADD_KEY_SLAVE]

REM ' =26 - Удалить ключ по номеру Е
IF D=26 THEN #(978)=E:#(976)=14:GOTO [SND_1X_COMMAND]

```

```

REM ' =27 - Удалить все ключи
IF D=27 THEN #(978)=0:#(976)=3:GOTO [SND_1X_COMMAND]

REM ' =28 - добавить пользователя в базу MASTER
REM '      #(790)=len=10 #(791)-#(801)=ключ AB153345CD
REM '      #(825)- #(857) имя 32 байта
IF D=28 THEN GOTO [ADD_USER_MASTER]

REM ' =29 - Поиск пользователя в базе MASTER
REM '      #(790)=len=10 #(791)-#(801)=ключ AB153345CD
REM '      #(825)- #(857) имя 32 байта
REM '      Результаты поиска пользователя
REM '      пишет значения или обнуляет, если нет
REM '      #(814)- #(824) ключ 11 байт
REM '      #(900)- #(931) имя 32 байта
IF D=29 THEN GOTO [FIND_USER_MASTER]

REM ' =30 - очистка базы пользователей MASTER
IF D=30 THEN GOTO [CLEAR_USER_BASE]

REM ' =31 - получение / установка времени
REM '      показ #(860)- #(879) Текст 19 байт max текущее время
REM '      отправка sendstp.htm?VDD=31&IS880=23%3A59&IS888=2017
REM '      -01-04&~KY~
REM '      #(880)- #(887) Текст 8 байт время 1+7
REM '      #(888)- #(898) Текст 11 байт дата 1+10
IF D=31 THEN GOTO [SET_TIME_MASTER]

REM ' =32 - удалить пользователя из базы MASTER
REM '      #(790)=len=10 #(791)-#(801)=ключ AB153345CD
REM '      #(825)- #(857) имя 32 байта
IF D=32 THEN GOTO [DEL_USER_MASTER]

GOSUB [CMD_ERR_CMD]
RETURN
REM '*****

REM '*****
REM ' НАЧАЛЬНАЯ ИНИЦИАЛИЗАЦИЯ
REM '*****
[START]
REM ' НАСТРОЙКА ПЕРИФИРИИ
LINIT 1:BEEP 15,2:UART 2,7:LPRINT "Starting..."
REM ' Загрузка адресов контроллеров
a=RDEE{0}:b=RDEE{2}:c=RDEE{4}:d=RDEE{6}
e=RDEE{8}:f=RDEE{10}:g=RDEE{12}:h=RDEE{14}
IF a>=0 THEN IF a<127 THEN GOTO [SKIP_INI_ADDR]
REM ' Загрузка адресов контроллеров по умолчанию
a=16:b=0:c=0:d=0:e=0:f=0:g=0:h=0
WREE a,0:WREE b,2:WREE c,4:WREE d,6
WREE e,8:WREE f,10:WREE g,12:WREE h,14
[SKIP_INI_ADDR]

REM ' СОЗДАДИМ ЛОГ ФАЙЛ , ЕСЛИ НЕТ
GOSUB [CREATE_LOG]

REM ' Понять IP и показать
REM ' IPCFG #(235) В переменную #(235) помещается структура
REM ' текущих сетевых настроек.
REM ' В первые девять байт #(235) - #(243) имя контроллера,
REM ' как текст (может быть выведен как $(235))

IPCFG #(235):LPRINT " IP="
AT 64:LPRINT %3,#(244),".",#(245),".",#(246),".",#(247)
q=#(244):r=#(245):s=#(246):t=#(247):v=#(248):x=#(249)

```

```

REM ' СОЗДАДИМ БАЗЫ, ЕСЛИ НЕТ
GOSUB [CREATE_USER_BASE]
GOSUB [CREATE_SLAVE_BASE]
REM ' Создание текстовых переменных - ключ
GOSUB [KEY_BAD]
GOSUB [STATUS_READY]
GOSUB [CMD_OK]
REM ' Создание текстовых переменных начальное
REM ' КОНТРОЛЛЕР ПОДКЛЮЧЕН/НЕ ПОДКЛЮЧЕН
GOSUB [K1_BAD]
IF a=0 THEN GOSUB [K1_NO]
GOSUB [K2_BAD]
IF b=0 THEN GOSUB [K2_NO]
GOSUB [K3_BAD]
IF c=0 THEN GOSUB [K3_NO]
GOSUB [K4_BAD]
IF d=0 THEN GOSUB [K4_NO]
GOSUB [K5_BAD]
IF e=0 THEN GOSUB [K5_NO]
GOSUB [K6_BAD]
IF f=0 THEN GOSUB [K6_NO]
GOSUB [K7_BAD]
IF g=0 THEN GOSUB [K7_NO]
GOSUB [K8_BAD]
IF h=0 THEN GOSUB [K8_NO]
REM ' НАЧАЛЬНЫЕ УСТАНОВКИ ТЕКУЩЕГО КОНТРОЛЛЕРА
j=1:i=a:p=63
GOSUB [GET_STATUS_K]
DATA $(814), "- - - -"
CLS:LPRINT "READY"
GOTO [MAIN]
REM ' *****
REM ' *****

[MAIN]
REM ' *****
REM ' *****  Главный цикл *****
REM ' *****

REM ' *****
REM ' Получить локальный ключ
GOSUB [GET_LOCAL_KEY]

REM ' *****
REM ' Получить локальное время
SDT $(860)

REM ' *****
REM ' Опрос всех на предмет записи лога
GOSUB [GET_LOG]

REM ' *****
FOR §=0 TO 40
REM ' Проверка наличия команды WEB Сервера
IF D<>0 THEN §-:EXFOR [WEB_COMMAND_PRESENT]
PAUSE 10
NEXT §
AT 0:LPRINT "READY"
STIME $(520):LPRINT *$(520)
AT 64:LPRINT " " :SDATE $(520)
LPRINT *$(520), " "
GOTO [MAIN]

[WEB_COMMAND_PRESENT]
GOSUB [STATUS_BUSY]
GOSUB [PARSER_WEB_COMMAND]
CLOSE:GOSUB [STATUS_READY]

```

```
D=0:GOTO [MAIN]
```

```
REM /*****
```

```
REM /*****
```