

```

REM ' *****
REM ' Система доступа по картам SKD.BAS
REM SUB
REM ' ver 2.0 08.01.17
REM ' *****
REM ' Требуется BASIC DEVICE 3.4.8 или старше
REM ' *****
REM ' Установить переключку 1 Wire
REM ' Подключить кнопку открытия к этому входу
REM ' LED красный
REM ' LED зеленый
REM ' *****
REM ' Требуется ОБРАБОТКА ПРЕКОМПИЛЯТОРОМ BASIC
REM ' *****
REM ' База ключей - одна запись 6 байт
REM ' #(x), #(x+1), #(x+2), #(x+3), #(x+4), #(x+5)
REM ' всего 170 записей (1020) байт
REM ' #(x) = 255 - ключа нет
REM ' #(x) = битовое поле 4 младших бита
REM ' бит 0 - считыватель 1
REM ' бит 1 - считыватель 2
REM ' бит 2 - реле 1 от считывателя 1
REM ' бит 3 - реле 2 от считывателя 1
REM ' бит 4 - реле 1 от считывателя 2
REM ' бит 5 - реле 1 от считывателя 2
REM ' бит 6 -
REM ' бит 7 - 0 - есть ключ в базе
REM ' Если бит установлен , то проверяется код на данных
REM ' считывателях и при совпадении срабатывает данное
REM ' реле
REM ' #(x+1), #(x+2), #(x+3), #(x+4), #(x+5) - КОД КЛЮЧА 5БАЙТ
REM ' *****
REM ' EEPROM addr(1020) Время открытия двери * 0,1 сек
REM ' *****
REM ' EEPROM addr(1021) = 255 - РАЗБЛОКТРОВАНЫ КНОПКИ
REM ' =0 Настройка с кнопок ЗАБЛОКИРОВАНА
REM ' *****
REM ' EEPROM addr(1022) Блокировка считывателя
REM ' 0 = LOCK 1,2,3 -UNLOCK (bits 1,0)
REM ' *****
REM ' EEPROM addr(1023) Блокировка двери
REM ' 0 = LOCK 1,2,3 -UNLOCK (bits 1,0)
REM ' *****
REM ' ПЕРЕМЕННЫЕ
REM ' *****
REM ENDSUB
REM ' *****
REM SUB ' VARIABLE MEMORY
REM ' *****
REM ' СТРУКТУРА 8 байт
REM ' Данные из базы / в базу - 8 байт
REM ' #(960)-#(968)=#(x), #(x+1), ..., #(x+5)
REM ' #(960) - права доступа =#(x)
REM ' #(961), #(962), #(963), #(964), #(965) - ключ
REM ' #(966) - адрес ключа в базе
REM ' #(967) -

REM ' СТРУКТУРА 8 байт
REM ' Данные из считывателя / временные - 8 байт
REM ' #(968)-#(975)=#(y), #(y+1), ..., #(y+5)
REM ' #(968) - права доступа / код = #(y)
REM ' #(969), #(970), #(971), #(972), #(973) - ключ
REM ' #(974) - адрес ключа
REM ' #(975) -
REM ENDSUB
REM ' *****
REM SUB ' VARIABLE MEMORY
REM ' СТРУКТУРА 8 байт команды / запросы от мастера
REM ' #(976) - код запроса команды

```

```

REM '#(977) - статус выполнения команды
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 ' 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 ' МАСТЕР - опрашивает код, обработка данных, удаление
REM ' кода, запись данных, выставление статуса
REM ENDSUB
REM SUB ' ***** RAM*****
REM '*****
REM ' Переменные для программы
REM '#(992) - USE
REM '#(993) - NOT USE
REM '#(994) - NOT USE
REM '#(995) - NOT USE
REM '#(996) - NOT USE
REM '#(997) - NOT USE
REM '#(998) - NOT USE
REM '#(999) - NOT USE
REM '*****
REM ENDSUB
REM '*****
REM SUB ' MASTER COMMAND, REPLY SLAVE
REM '*****
REM ' СТРУКТУРА 8 байт команды / запросы от мастера
REM '#(976) - код запроса команды
REM '#(977) - статус выполнения команды
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 ' # (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 СЛЕД. КОМАНДЫ (ALARM)
REM ' 12-ОТМЕНИТЬ КОМАНДУ 11
REM ' 13- ПЕРЕЗАГРУЗКА
REM ' 14-УДАЛИТЬ КЛЮЧ ПО НОМЕРУ ДАННЫЕ #(978)

```

```

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 (ОБНОВЛЕНЫ ПРАВА)
REM ' 4 - КЛЮЧ УДАЛЕН
REM ' 5 - ВСЕ КЛЮЧИ УДАЛЕНЫ
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 '*****
REM ENDSUB

```

```

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

```

```

REM '*****
REM '      Меню Управления
REM '*****
[MENU]
REM SUB ' ЗАПУСКАТЬ ЛИ ПРОГРАММУ ?
REM ' Если ничего не нажато - выходим
§=GKEY{}:IF §=0 THEN RETURN
REM SUB ' Заблокированы ли кнопки ?
#(975)=RDEE{1021}:IF #(975)=0 THEN RETURN
REM SUB ' ЗАПУСКАЕМ
BEEP 15,2
REM ' ВЫДЕЛИМ ПАМЯТЬ В СТЕКЕ = А Б В Г
REM ' А = НОМЕР СТРОКИ МЕНЮ (0)
REM ' Б = адрес места в еепром
REM ' В = права пользователя
REM ' Г = tmp
§=0:§=0:§=0:§=0
REM ' ждем отпущения
[WAIT_START_MENU]
§=GKEY{}:IF §<>0 THEN GOTO [WAIT_START_MENU]
REM ' СЧИТАЕМ И ПОКАЗЫВАЕМ СКОЛЬКО ЕСТЬ КЛЮЧЕЙ
GOSUB [CALK_KEY]
CLS:LPRINT "KEY in base =",§:PAUSE 1000
REM '*****
REM ' Покажем меню
[SEL_MENU]
CLS:LPRINT "Select MODE"
[PRN_MENU]
IF A=0 THEN AT 64:LPRINT "ADD key"
IF A=1 THEN AT 64:LPRINT "DEL key"
IF A=2 THEN AT 64:LPRINT "VIEW key"

```

```

IF A=3 THEN AT 64:LPRINT "SEL Time open      "
IF A=4 THEN AT 64:LPRINT "DEL ALL key      "
IF A=5 THEN AT 64:LPRINT "Reboot          "
IF A=6 THEN AT 64:LPRINT "EXIT            "
REM 'ждем реакции
§=KEY{10}:BEEP 15,2
CASE §,0,[EXIT_MENU],1,[INC_menu],2,[DEC_menu],4,[OK_MENU],8,[INC_menu],16,[DEC_menu]
GOTO [SEL_MENU]
[INC_menu]
A=A+1:IF A>6 THEN A=0
GOTO [PRN_MENU]
[DEC_menu]
A=A-1:IF A<0 THEN A=6
GOTO [PRN_MENU]
REM '*****
[OK_MENU]
REM ' выбрали какойто пункт, выбираем что делать
B=63:CASE A,0,[ADD_K],1,[DEL_K],2,[VIEW_K],3,[TIME_OPEN]
CASE A,4,[DEL_ALL_K],6,[EXIT_MENU]
§^:CLR:GOTO [START]:REM '***** REBOOT
REM '*****

REM '*****
REM '  Просмотр базы ключей с правами
[VIEW_K]
REM 'Просмотр ключей с перврго
B=0
[VIEW_K_1]
FOR §=B TO 169
REM ' READ DATABSE FIRST bute KEY
#(975)=RDEE{A*6}:IF #(975)<>255 THEN Г=§:EXFOR [VIEW_K_2]
NEXT §
AT 64:LPRINT "NO next key      ":PAUSE 1000:GOTO [SEL_MENU]
[VIEW_K_2]
Г=B*6:RDPE *6,§(960),Г
CLS:LPRINT %3,"#KEY=",B,"Perm=",#(960):AT 64
LPRINT "KEY= ",@(961),@(962),@(963),@(964),@(965)
§=KEY{15}:BEEP 15,2
IF §=0 THEN GOTO [EXIT_MENU]
B=B+1:GOTO [VIEW_K_1]
REM '*****

REM '*****
REM ' Добавление ключа
[ADD_K]
REM ' права по умодчанию - все
PAUSE 20:§=GKEY{}:IF §<>0 THEN GOTO [ADD_K]
CLS:LPRINT "ADD KEY"
GOSUB [FIND_FREE]
B=#(975):IF B=255 THEN GOTO [LIMIT_EE]
AT 7:LPRINT %3,"#",B," R="
[ADD_K_1]
PAUSE 20:§=GKEY{}:IF §<>0 THEN GOTO [ADD_K_1]
AT 14:LPRINT %2,B
AT 64:LPRINT "WAIT KEY      "
z=0
[ADD_K_2]
z=z+1:IF z>100 THEN BEEP 15,2:GOTO [EXIT_MENU]
REM ' ЕСЛИ КНОПКА - РЕДАКТИРУЕМ ПРАВА
§=GKEY{}:IF §=0 THEN GOTO [WT_ADD_KEY]

REM '*****
REM ' EDIT RIGHTS
BEEP 15,2:z=0
[ADD_K_1_EDIT]
REM 'PRINT RIGHTS СЧИТЫВАТЕЛЬ,
REM 'ДВЕРЬ ОТ 1 СЧИТЫВАТЕЛЯ,
REM 'ДВЕРЬ ОТ 2 СЧИТЫВАТЕЛЯ
REM '1 -ПЕРВЫЙ, 2 ВТОРОЙ, 3 - ОБА(Е)

```

```

AT 64:Γ=(B&3):LPRINT %1,"in:",Γ,"out:"
Γ=(B&12)/4:LPRINT %1," A:",Γ
Γ=(B&48)/16:LPRINT %1," B:",Γ
REM 'ждем реакции
Γ=KEY{10}:BEEP 15,2
CASE Γ,16,[SEL_READ],1,[SEL_IN1_OUT],8,[SEL_IN2_OUT],4,[SEL_MENU]
GOTO [ADD_K_1]
[SEL_READ]
Γ=(B&3):Γ=Γ+1:Γ=Γ&3:B=B&60:B=B|Γ
AT 14:LPRINT %2,B:GOTO [ADD_K_1_EDIT]
[SEL_IN1_OUT]
Γ=(B&12):Γ=Γ+4:Γ=Γ&12:B=B&51:B=B|Γ
AT 14:LPRINT %2,B:GOTO [ADD_K_1_EDIT]
[SEL_IN2_OUT]
Γ=(B&48):Γ=Γ+16:Γ=Γ&48:B=B&15:B=B|Γ
AT 14:LPRINT %2,B:GOTO [ADD_K_1_EDIT]
REM '*****

REM '*****
REM ' собственно ожидание считывателя
[WT_ADD_KEY]
REM ' WAIT AND ADD KEYS
#(968)=RFID{1}
IF #(968)=1 THEN GOTO [GO_ADD]
#(968)=RFID{2}
IF #(968)=1 THEN GOTO [GO_ADD]
GOTO [ADD_K_2]
REM '*****

REM '*****
REM ' ключ пойман, добавим
[GO_ADD]
GOSUB [FIND_KEY]
IF #(966)=255 THEN GOTO [GO_ADD_1]
BEEP 5,8:AT 64
LPRINT "THIS KEY PRESENT":PAUSE 1000:GOTO [ADD_K]
[GO_ADD_1]
#(968)=B:§=B*6
WRPE *6,$(968),§:BEEP 15,2
AT 64:LPRINT "DONE"
§=KEY{2}:BEEP 15,2:IF §>0 THEN GOTO [SEL_MENU]
GOTO [ADD_K]
REM '*****
[LIMIT_EE]
AT 64:LPRINT "NO FREE MEMORY ":PAUSE 1000:GOTO [SEL_MENU]
REM '*****

REM '*****
REM ' удаление всех ключей
[DEL_ALL_K]
CLS:LPRINT "DEL ALL KEY "
AT 64:LPRINT "ARE YOU SURE ?"
§=KEY{5}:BEEP 15,2:IF §<>4 THEN GOTO [SEL_MENU]
GOSUB [FORMAT_EEPROM]
AT 64:LPRINT "DONE"
GOTO [SEL_MENU]
REM '*****

REM '*****
REM ' удаление ключа
[DEL_K]
CLS:LPRINT "DEL KEY "
AT 64:LPRINT "WAIT KEY"
z=0
[WT_DEL_KEY]
z=z+1:IF z>100 THEN BEEP 15,2:GOTO [EXIT_MENU]
#(968)=RFID{1}
IF #(968)=1 THEN GOTO [GO_DEL]
#(968)=RFID{2}

```



```

REM '#(984) =03 открыто обе двери
REM '#(984) =40 Ключ не опознан
REM ' читаем , если считыватель разрешен
#(975)=RDEE{1022}:IF #(975)&1=0 THEN GOTO [SKP_1]
#(968)=RFID{1}
IF #(968)=1 THEN #(968)=1:#(992)=10:GOTO [GO_WORK]
[SKP_1]
#(975)=RDEE{1022}:IF #(975)&2=0 THEN GOTO [SKP_2]
#(968)=RFID{2}
IF #(968)=1 THEN #(968)=2:#(992)=20:GOTO [GO_WORK]
[SKP_2]
GOTO [EXIT_ZAMOK]
REM '*****
[GO_WORK]
CLS:LPRINT "KEY= ",@(969),@(970),@(971),@(972)
LPRINT @(973):AT 64:LPRINT "NO OPEN DOOR      "
REM ' запись в базу
#(986)=0:#(987)=#(969):#(988)=#(970)
#(989)=#(971):#(990)=#(972):#(991)=#(973)
GOSUB [FIND_KEY]
IF #(966)>169 THEN BEEP 5,16:GOTO [ERR_KEYS_EXIT]
REM ' ЕСТЬ В БАЗЕ, ТЕСТ ПРАВ
#(986)=#(960)
REM '#(968) - НОМЕР СЧИТЫВАТЕЛЯ
REM '#(960) - ПРАВА из базы
REM ' Допустим ли считыватель ?
#(967)=#(968)&#(960)&3
IF #(967)=0 THEN BEEP 5,8:GOTO [ERR_KEYS_EXIT]
REM ' узнаем что включить
REM ' 1 и 2 дверь
#(967)=#(968)&((#(960)/4)&3)
#(975)=#(968)&((#(960)/16)&3)
IF #(967)+#(975)=0 THEN BEEP 5,8:GOTO [ERR_KEYS_EXIT]
REM ' открытие
REM ' открытие - ПРОВЕРИМ БЛОКИРОВКУ
#(974)=RDEE{1023}:IF #(974)&1=0 THEN GOTO [SKP_OP_1]
IF #(967)=0 THEN GOTO [SKP_OP_1]
GREEN_ON:RED_OFF:#(992)=#(992)+1
OPEN_DOOR_1
[SKP_OP_1]
#(974)=RDEE{1023}:IF #(974)&2=0 THEN GOTO [SKP_OP_2]
IF #(975)=0 THEN GOTO [SKP_OP_2]
GREEN_ON:RED_OFF:#(992)=#(992)+2
OPEN_DOOR_2
[SKP_OP_2]
#(984)=#(992)
IF #(992)&3=0 THEN GOTO [SKP_OP_3]

REM 'открытие - попкикиваем
AT 64:LPRINT "OPEN DOOR      "
REM ' EEPROM addr(1020) Время открытия двери * 0,1 сек
#(967)=RDEE{1020}

FOR §=0 TO #(967)
IF A%5=0 THEN BEEP 15,2
AT 75:LPRINT %3,A
PAUSE 100
GOSUB [PC_COMMAND]
NEXT §

[SKP_OP_3]
CLOSE_DOOR
GREEN_OFF:RED_ON
GOTO [EXIT_ZAMOK]
[ERR_KEYS_EXIT]
#(960)=0:#(984)=#(992)+40:PAUSE 300
[EXIT_ZAMOK]
RETURN
REM '*****

```

```

REM '**** САМ ЗАМОК END
REM '*****

REM '*****
REM '**** функции
REM '*****

REM '*****
REM ' Подсчет количества ключей в базе
REM '*****
REM SUB
[CALK_KEY]
§=0
FOR §=0 TO 169:REM ' READ DATABSE FIRST bute KEY
#(975)=RDEE{A*6}:IF #(975)<>255 THEN B=B+1
NEXT §
RETURN
REM ' количество - в стеке 0 -170
REM '*****

REM '*****
REM ' Найти ключ в базе
REM ' требует #(y+1=969), #(y+2), #(y+3), #(y+4), #(y+5)
REM ' Возвращает ПРАВА = #(960)
REM ' Возвращает #(966) - адрес ключа 255 если нет
REM '*****
REM SUB
[FIND_KEY]
FOR §=0 TO 169
§=A*6
RDPE *6, $(960), §
§=SCOMP{#(961), #(969), 5}
IF #(960)=255 THEN A=0
IF §=1 THEN #(966)=§:EXFOR [KEY_PRESENT]
NEXT §
#(966)=255
[KEY_PRESENT]
RETURN
REM '*****

REM '*****
REM ' Найти пустое место
REM ' возвращает адрес 0-169 или 255 если нет В #(975)
REM '*****
REM SUB
[FIND_FREE]
FOR §=0 TO 169:REM ' READ DATABSE FIRST bute KEY
#(975)=RDEE{A*6}:IF #(975)=255 THEN EXFOR [FREE_OK]
NEXT §
#(975)=255
[FREE_OK]
#(975)=§
RETURN
REM '*****

REM '*****
REM ' Форматирование памяти
REM '*****
REM SUB
[FORMAT_EEPROM]
#(975)=255
FOR §=0 TO 1019
WREE #(975), A
NEXT §
RETURN
REM '*****

```



```

#(975)=255:§=#(966)*6
WREE #(975),§
#(976)=0:#(977)=4:#(978)=0:AT 64
LPRINT "DEL KEY OK      "
GOTO [EXIT_CMD]
REM ENDSUB '*****

REM '*****
REM SUB ' 3-УДАЛИТЬ ВСЕ КЛЮЧИ
[M_DEL_A]
GOSUB [FORMAT_EEPROM]
#(976)=0:#(977)=5:#(978)=0:AT 64
LPRINT "DEL ALL KEY OK  "
GOTO [EXIT_CMD]
REM ENDSUB '*****

REM '*****
REM SUB '4-УСТАНОВИТЬ ВРЕМЯ-ДААННЫЕ #(978);0- АПРОС ВРЕМ.
[M_TIME]
IF #(978)=0 THEN #(978)=RDEE{1020}:GOTO [M_TIME_EXIT]
WREE #(978),1020
[M_TIME_EXIT]
#(976)=0:#(977)=6:AT 64
LPRINT "SET TIME OK    "
GOTO [EXIT_CMD]
REM ENDSUB '*****

REM '*****
REM SUB '5-ЗАПРОС статуса
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
[M_COUNT]
GOSUB [CALC_KEY]
#(978)=§
#(979)=RDEE{1020}
#(980)=RDEE{1021}
#(980)=(!#(980))&1
#(981)=RDEE{1022}
#(981)=(!#(981))&3
#(982)=RDEE{1023}
#(982)=(!#(982))&3
#(983)=0
IF PORT*{3}=1 THEN #(983)=#(983)+1
IF PORT*{4}=1 THEN #(983)=#(983)+2
#(976)=0:#(977)=1:AT 64
LPRINT "SEND STATUS OK  "
GOTO [EXIT_CMD]
REM ENDSUB '*****

REM '*****
REM SUB '14-DEL KEY NUMBER
REM ' ЗАПРОС ДАННЫЕ НОМЕРА #(978)
[M_DEL_N]
IF #(978)=0 THEN GOTO [ERR_NO_KEY]
§=0
FOR §=0 TO 169:REM ' READ DATABASE FIRST bute KEY
REM ' ЦИКЛ ПОИСКА ВАЛИДНЫХ КЛЮЧЕЙ НУЖНОГО НОМЕРА
#(967)=A:REM ' сохраним адрес в базе
#(975)=RDEE{A*6}:IF #(975)=255 THEN GOTO [M_DEL_1]
B=B+1:IF B=#(978) THEN §=:EXFOR [M_DEL_2]
[M_DEL_1]
NEXT §
GOTO [ERR_NO_KEY]

```

```

[M_DEL_2]
REM ' DEL ключ
§=#(967)*6: #(975)=255
WREE #(975), §
#(976)=0: #(977)=1: AT 64
LPRINT "DEL KEY OK      "
GOTO [EXIT_CMD]

REM ENDSUB '*****

REM '*****
REM SUB '6-ПОЛУЧИТЬ КЛЮЧ ПО НОМЕРУ =(1 - КОЛИЧЕСТВО)
REM '      ЗАПРОС ДАННЫЕ НОМЕРА #(978)
REM '      ОТВЕТ ДАННЫЕ КЛЮЧА #(978)-#(983)
[M_GET]
IF #(978)=0 THEN GOTO [ERR_NO_KEY]
§=0
FOR §=0 TO 169: REM ' READ DATABASE FIRST byte KEY
REM ' ЦИКЛ ПОИСКА ВАЛИДНЫХ КЛЮЧЕЙ НУЖНОГО НОМЕРА
#(967)=A: REM ' сохраним адрес в базе
#(975)=RDEE{A*6}: IF #(975)=255 THEN GOTO [M_GET_1]
B=B+1: IF B=#(978) THEN § -: EXFOR [M_GET_2]
[M_GET_1]
NEXT §
GOTO [ERR_NO_KEY]

[M_GET_2]
REM ' читаем ключ
§=#(967)*6: RDPE *6, $(960), §
#(978)=#(960): #(979)=#(961): #(980)=#(962)
#(981)=#(963): #(982)=#(964): #(983)=#(965)
#(976)=0: #(977)=1: AT 64
LPRINT "SEND KEY OK      "
GOTO [EXIT_CMD]
REM ENDSUB '*****

REM '*****
REM SUB '7-ЗАБЛОКИРОВАТЬ/РАЗБЛ.НАСТРОЙКУ С КНОПОК #(978)
REM '      ДАННЫЕ #(978) = 1- LOCK 0 = UNLOCK
[M_LOCK]
#(975)=255
IF #(978)=1 THEN #(975)=0
IF #(978)=0 THEN #(975)=255
WREE #(975), 1021
#(976)=0: #(977)=1: AT 64
LPRINT "LOCK KBD ON/OFF  "
GOTO [EXIT_CMD]
REM ENDSUB '*****

REM '*****
REM SUB '8-ЗАБЛ.СЧИТЫВ. #(978)= 0 -UNLOCK 1,2,3 -LOCK
REM ' EEPROM addr(1022) Блокировка считывателя
REM ' 0 = LOCK 1,2,3 -UNLOCK (bits 1,0)
[M_R_LOCK]
#(978)=(!#(978))&3
WREE #(978), 1022
#(976)=0: #(977)=1: AT 64
LPRINT "LOCK READ ON/OFF"
GOTO [EXIT_CMD]
REM ENDSUB '*****

REM '*****
REM SUB '9-ЗАБЛОКИР. ДВЕРЬ #(978) = 0-UNLOCK 1,2,3 -LOCK
REM ' EEPROM addr(1023) Блокировка двери
REM ' 0 = LOCK 1,2,3 -UNLOCK (bits 1,0)
[M_D_LOCK]
#(978)=(!#(978))&3
WREE #(978), 1023
#(976)=0: #(977)=1: AT 64

```

```

LPRINT "LOCK DOOR ON/OFF"
GOTO [EXIT_CMD]
REM ENDSUB '*****

REM '*****
REM SUB ' 10-ОТКРЫТЬ ДВЕРЬ #(978) = 1,2,3
[M_OPEN]
#(976)=0:#(977)=1
IF #(978)=0 THEN GOTO [EXIT_CMD]
GREEN_ON
IF #(978)&1=1 THEN OPEN_DOOR_1
IF #(978)&2=2 THEN OPEN_DOOR_2
AT 64:LPRINT "OPEN DOOR CMD "
#(967)=RDEE{1020}
FOR §=0 TO #(967)
IF A%5=0 THEN BEEP 15,2
PAUSE 100
NEXT §
CLOSE_DOOR:GREEN_OFF
GOTO [EXIT_CMD]
REM ENDSUB '*****

REM '*****
REM SUB '11-ОТКРЫТЬ ДВЕРИ ДО СЛЕД. КОМАНДЫ (ALARM)
[M_A_OPERN]
#(976)=0:#(977)=1
OPEN_DOOR_1
OPEN_DOOR_2
AT 64:LPRINT "OPEN ALL DOOR "
GOTO [EXIT_CMD]
REM ENDSUB '*****

REM '*****
REM SUB '
[M_RET]
#(976)=0:#(977)=1
CLOSE_DOOR
AT 64:LPRINT "CLOSE DOOR CMD "
GOTO [EXIT_CMD]
REM ENDSUB '*****

REM '*****
REM SUB '13- ПЕРЕЗАГРУЗКА
[M_RST]
AT 64:LPRINT "REBOOT "
#(976)=0:#(977)=1:PAUSE 500
§^:CLR:GOTO [START]:REM '***** REBOOT
REM ENDSUB '*****

REM ' OK_CMD_1
REM ' #(976)=0:#(977)=1:#(978)=0:GOTO [EXIT_CMD]
[ERR_CMD_33]
#(976)=0:#(977)=33:#(978)=0
AT 64:LPRINT "ERROR CMD "
[EXIT_CMD]
RETURN
REM '*****

[START]
REM '*****
REM ' Start SKD
REM '*****

#define OPEN_DOOR_1 EQU PORT 3,1
#define OPEN_DOOR_2 EQU PORT 4,1
#define CLOSE_DOOR EQU PORT 4,0:PORT 3,0

```

```

#define RED_ON EQU PORT 5,1
#define RED_OFF EQU PORT 5,0

#define GREEN_ON EQU PORT 9,1
#define GREEN_OFF EQU PORT 9,0

LINIT 1:BEEP 15,2
LPRINT "SKD V2.0 Start "
RED_ON:GREEN_OFF
REM ' Отчет о количестве ключей
GOSUB [CALK_KEY]
AT 64:LPRINT "KEY in base =",§:PAUSE 1500:BEEP 15,2
CLS:LPRINT "READY":GOTO [MAIN]

REM ' *****
REM '          ГЛАВНЫЙ ЦИКЛ
REM ' *****
[MAIN]
REM ' Вызов функций
GOSUB [MENU]
GOSUB [DISPLAY_STATUS]
GOSUB [ZAMOK_WORK]
GOSUB [PC_COMMAND]
GOSUB [ALARM_OPEN]
GOTO [MAIN]
REM ' *****

REM ' *****
REM '          Подпрограмма состояний
REM ' *****
[DISPLAY_STATUS]
CLS:LPRINT "READY"
RETURN

REM ' *****
REM '          Подпрограмма открытия по кнопке
REM ' *****
[ALARM_OPEN]
REM ' *****
REM ' ***** DEBUG
REM ' GOTO [EXIT_ALARM_OPEN]
REM ' *****
IF PORT{10}=1 THEN GOTO [EXIT_ALARM_OPEN]
OPEN_DOOR_1
OPEN_DOOR_2
GREEN_ON
REM ' запись в базу
#(986)=0:#(987)=0:#(988)=0
#(989)=0:#(990)=0:#(991)=0
#(960)=0:#(984)=83
#(967)=RDEE{1020}
FOR §=0 TO #(967)
IF A%5=0 THEN BEEP 15,2
PAUSE 100
NEXT §
CLOSE_DOOR:GREEN_OFF
[EXIT_ALARM_OPEN]
RETURN

```

