

Как использовать лисп, опубликованный на форуме

✚ dwg.ru/pub/9

Вы нашли полезный для себя код на Лиспе и хотите его использовать. Можно выделить 3 шага:

- сохранение программного кода на своем компьютере
- загрузка файла
- запуск приложения

Примечание: AutoLISP программы могут быть запущены только на полной версии Автокада, они не работают под AutoCAD LT. (исключая случаи загрузки дополнительных приложений типа LT Extender и ему подобных, которые специально разработаны для запуска lisp и (или) arx приложений в среде AutoCAD LT.)

Сохранение программного кода на своем компьютере

С помощью курсора выделяем этот код и копируем его в буфер обмена, нажав правую кнопку мыши и выбрав пункт **Копировать** (или **Ctrl+C**).

Запускаем "Блокнот" и вставляем наш код с помощью меню **Правка >> Вставить** (или **Ctrl+V**):

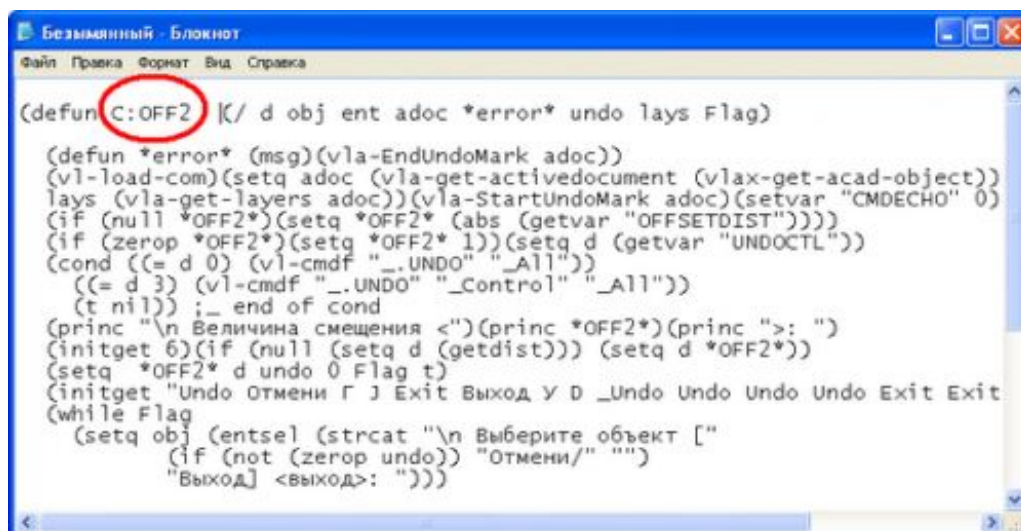
Код:

```

(defun C:OFF2 (/ d obj ent adoc *error* undo lays Flag)
  (defun *error* (msg) (vla-EndUndoMark adoc))
  (vl-load-com) (setq adoc (vla-get-activedocument (vlax-get-acad-object)))
  lays (vla-get-layers adoc) (vla-StartUndoMark adoc) (setvar "CMDECHO" 0)
  (if (null *OFF2*) (setq *OFF2* (abs (getvar "OFFSETDIST"))))
  (if (zerop *OFF2*) (setq *OFF2* 1)) (setq d (getvar "UNDOCTL"))
  (cond ((= d 0) (vl-cmdf "_UNDO" "_All"))
        ((= d 3) (vl-cmdf "_UNDO" "_Control" "_All"))
        (t nil)) ;_ end of cond
  (princ "\n Величина смещения <") (princ *OFF2*) (princ ">: ")
  (initget 6) (if (null (setq d (getdist))) (setq d *OFF2*))
  (setq *OFF2* d undo 0 Flag t)
  (initget "Undo Отмени Г J Exit Выход У D _Undo Undo Undo Undo Exit Exit Exit
Exit")
  (while Flag
    (setq obj (entsel (strcat "\n Выберите объект ["
      (if (not (zerop undo)) "Отмени/" "")
      "Выход] <выход>: ")))
    (cond ((= obj "Undo") (if (not (zerop undo)) (progn
      (vl-cmdf "_UNDO" "_B") (setq undo (1- undo)) (alert "Нечего больше
отменять"))))
      ((= obj "Exit") (setq Flag nil))
      ((null obj) (if (= (getvar "ERRNO") 52) (setq Flag nil) (princ " *** Мимо ***"))))
      (t (setq ent (vlax-ename->vla-object (car obj)))
        (cond ((= (vla-get-lock (vla-item lays (vla-get-layer ent))) :vlax-true)
          (alert "На заблокированном слое!"))
          ((vlax-method-applicable-p ent 'Offset) (vl-cmdf "_UNDO" "_M") (setq undo (1+
undo))
            (vla-offset ent d) (vla-offset ent (- 0 d)))
            (t (alert "Не удастся создать объект, подобный данному")))))
        (initget "Undo Отмени Г J Exit Выход У D _Undo Undo Undo Undo Exit Exit Exit
Exit"))
    (vla-EndUndoMark adoc) (princ))
  (princ "\nНаберите в командной строке OFF2"))

```

После того как вставили код в блокнот, мы должны иметь что-то наподобие этого:



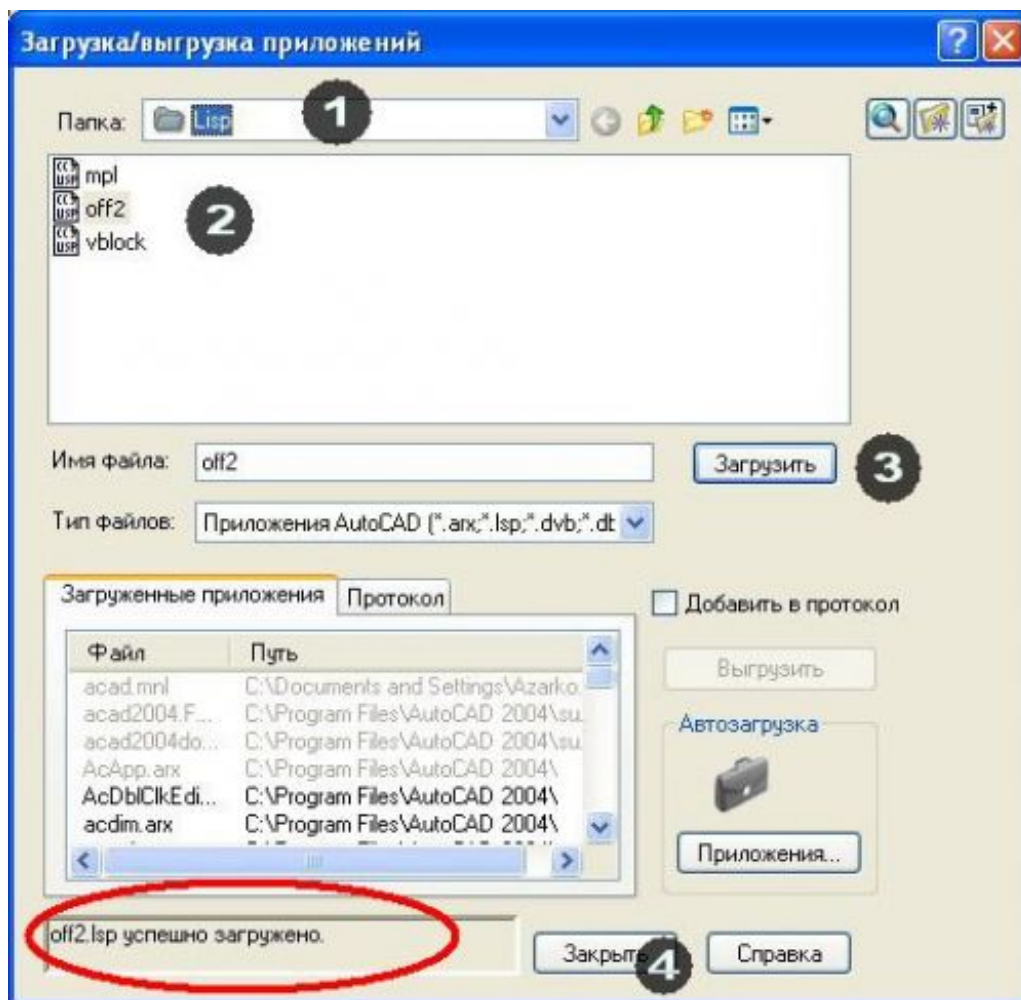
Теперь мы должны сохранить файл. Файл можно назвать, как вам нравится, но расширение у него должно быть **.LSP**. Правильнее (но не обязательно) будет назвать файл так же, как названа команда в найденном Вами коде. Команде всегда предшествует префикс **C:** после слова **defun**. В примере выше видно, что команду назвали **OFF2**, соответственно файл называем **off2.lsp**. Таким образом, мы можем избежать беспорядка, т.к. будем знать, что в файле **off2.lsp** находится команда **OFF2**.

Иногда в коде представлено несколько функций. Необходимо найти "главную". Такая функция либо имеет префикс **C:**, либо сам автор кода уточняет, что "использовать как <...>". В случае нескольких "главных" именование файла не может иметь никаких рекомендаций.

Загрузка lisp файла

Далее, запускаем Автокад и выбираем **Сервис >> Автолисп >> Загрузить (Tools >> AutoLISP >> Load)** или набираем в командной строке **_appload**.

Мы должны увидеть диалоговое окно как представлено ниже:



Используйте следующую последовательность, чтобы загрузить **off2.lsp**:

1. Перейдите к папке, куда Вы сохранили lisp файл.
2. Выберите файл, который Вы хотите загрузить из списка.
3. Нажмите кнопку **Загрузить**.
(Если все прошло удачно, Вы должны увидеть надпись "off2.lsp успешно загружено")

4. Нажмите кнопку **Заккрыть** для завершения команды.

Команда или функция ?

Для начала необходимо определиться, что у нас: команда или функция?

Немного теории.

В описании практически любого кода есть такая строка, как

```
(defun ...
```

и потом после нее чего-то там написано. Так вот то, что там написано и есть имя функции.

Если первыми символами являются **C:**, то функция готова к вызову из командной строки. Если такого нет, то вызывать функцию надо, заключая ее имя в круглые скобки.

Например, есть функция

```
(defun c:test ()<Что-то делаем 1>
```

Ее вызывать (при условии, что она подгружена) с командной строки можно напрямую:

Команда : test

И будут выполнены действия <Что-то делаем 1>

А вот такую:

```
(defun test ()<Чего-то делаем 2>
```

Можно будет вызывать только так:

Команда : (test)

И будут выполнены действия <Чего-то делаем 2>.

С точки зрения AutoCAD'a это 2 разных модуля.

Запуск lisp программы

Как только lisp файл загружен, Вы можете вызвать команду из командной строки. Команда вызывается по ее имени. В нашем примере необходимо набрать в командной строке **OFF2**.

Запомните, имя команды содержит префикс **C:** после слова **defun**. Если Вы видите в коде:

```
(defun c:<имя_команды>
```

то в командной строке необходимо набрать <имя_команды> без префикса **C:**.

Tips: Вы можете также загрузить lisp файл путем перетаскивания (**drag-and-drop**) иконки файла в графическое окно открытого документа Автокада.

AutoCAD ; error: no function definition:

Если Ваша функция или команда не работает или завершает работу с сообщением как в примере ниже, то Вам может помочь простой способ.

```
Команда: (vlax-get-acad-object)  
; error: no function definition: VLAX-GET-ACAD-OBJECT
```

Наберите в командной строке (vl-load-com) или (предпочтительнее) добавьте ее в начало lisp файла.

```
Команда: (vl-load-com)
```

Теперь ваш lisp должен работать без ошибок.

Что такое это (vl-load-com) ?

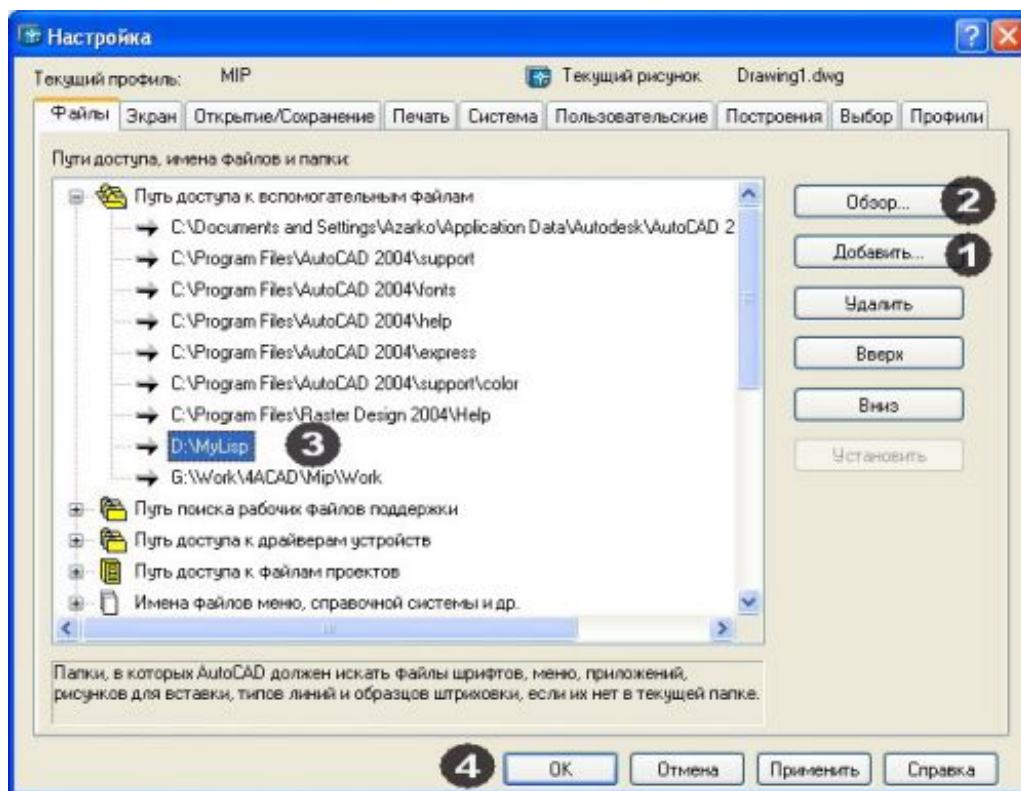
Данная функция загружает большое количество функций, входящих в расширение AutoLisp, позволяющее работать с объектами, свойствами, методами и событиями ActiveX. Признаком необходимости загрузки расширения AutoLisp является наличие в тексте Вашей программы функций, начинающихся с префиксов **vla-** (например **vla-get-activedocument**), **vlax-** (например **vlax-get-acad-object**), **vl-** (например **vl-catch-all-apply**) и **vlr-** (например **vlr-pers**).

В любом случае никогда не помешает добавить **(vl-load-com)** в начало файла, чтобы избежать подобных ошибок.

Организация библиотеки лисп файлов.

Что делать, если набралось достаточное количество файлов? Каждый раз набирать **_upload** для загрузки того или иного приложения? Наиболее эффективным будет организация своей библиотеки. Для начала условимся, что все наши файлы будем складывать в одну папку. Например, **D:\MyLisp**. Пропишем эту папку в путях доступа Автокада. Для этого выбираем **Сервис >> Настройка (Tools >> Options)** или набираем в командной строке **_options**. Переходим на закладку **Файлы (Files)** и раскрываем список **Путь доступа к вспомогательным файлам (Support File Search Path)**.

Мы должны увидеть диалоговое окно, как представлено ниже:



Используйте следующую последовательность, чтобы прописать папку в путях доступа:

1. Нажмите кнопку **Добавить**.
2. Нажмите кнопку **Обзор** и выберите целевую папку (в нашем случае , **D:\MyLisp**)
3. Путь к папке должен появиться в диалоговом окне.
4. Нажмите кнопку **ОК** для завершения команды.

Теперь наша папка прописана в путях доступа Автокада. Что это нам дает? Мы можем загружать наши файлы не указывая конкретного пути, а только имя файла.

Мы можем набрать в командной строке (**load "<имя файла>"**) для загрузки необходимого кода, например (**load "off2"**). Заметьте, что добавлять расширение **.lsp** не обязательно, хотя и не запрещается.

И, наконец, самое главное - это позволит нам создать панели с кнопками для загрузки той или иной программы. Кнопка будет иметь следующее содержание:

```
^C^C(if (null C:<команда>) (load "<файл>"));
<команда>;
```

В нашем случае:

```
^C^C(if (null C:OFF2) (load
"off2"));OFF2;
```

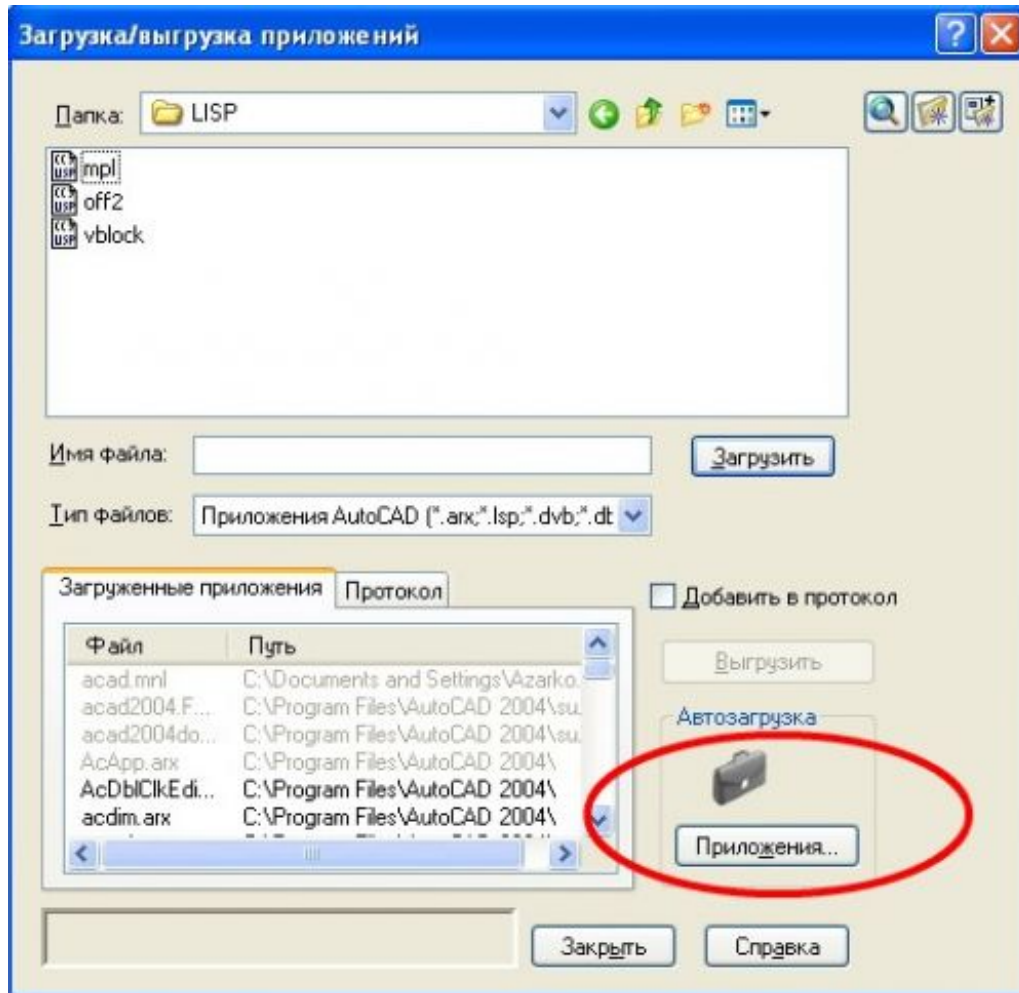
Вместо текста <команда> мы должны будем подставлять имя нужной нам команды, а вместо текста **<файл>** имя того файла, в котором эта команда сохранена. Напоминаю, что файл должен иметь расширение **.lsp**. Более подробно по организации меню можно почитать еще здесь <http://www.kulichki.com/cadhlp/pdmnu.htm>

Автозагрузка файлов.

Если есть необходимость загружать свои lsp файлы при открытии чертежа, то нужно добавить необходимые файлы в список автозагрузки.

Запускаем Автокад и выбираем **Сервис >> Автолисп >> Загрузить (Tools >> AutoLISP >> Load)** или набираем в командной строке **_appload**.

В открывшемся диалоговом окне жмем на кнопку **Приложения** рядом с изображением портфеля



В следующем диалоговом окне жмем на кнопку **Добавить** и указываем на необходимые файлы.

Они должны появиться в списке приложений. Теперь перечисленные файлы будут загружаться каждый раз при открытии рисунка. Важно понимать, что (по крайней мере в полной версии AutoCAD) загрузка приложений, помещенных в **Автозагрузку (Startup Suite)**, как правило, выполняется во все профили и сессии AutoCAD'a. Кроме того, есть еще один момент. Если существует 2 или более файлов, в которых описаны функции с одинаковыми именами, выполняться будет та, которая загрузилась последней. Поскольку предсказать, в какой последовательности будет выполняться загрузка приложений, невозможно, подобных ситуаций следует избегать.

Автор статьи: Владимир Азарко aka VVA

