Габидулин В.М.

САПР от А до Я

Адаптация AutoCAD под стандарты предприятия



ДМК Пресс
Издательство

УДК УДК 62-1:004.9AutoCAD
ББК ББК 30.2с515
Г12

Г12 Габидулин В. Н.
Адаптация AutoCAD под стандарты предприятия. – М.: ДМК Пресс, 2012. – 210 с.: ил.

ISBN 978-5-94074-852-6

Освоив основные приемы работы в системе AutoCAD и выполнив первый реальный проект, проектировщик, как правило, не достигает идеала при оформлении работы. Например, типы линий и штриховок в стандартной поставке системы не удовлетворяют стандартам, принятым на предприятии. В данной книге подробно, с множеством примеров описывается алгоритм создания форм и типов линий, что пригодится представителям самых разных профессий: геодезистам, проектировщикам систем энергоснабжения, строителям и др.

Умение пользоваться блоками, блоками с атрибутами и динамическими блоками – признак высокой культуры работы в AutoCAD, поэтому часть книги посвящена блокам. Один раз созданный динамический блок позволит надолго разрешить проблему создания однотипных элементов, а использование атрибутов – создавать автоматизированные таблицы и спецификации.

Наконец, изучив материал данной книги, читатель научится настраивать пользовательский интерфейс, создавать собственные команды и модифицировать существующие.

Книга предназначена для пользователей, имеющих базовые навыки работы в AutoCAD. Излагаемый материал будет полезен читателям вне зависимости от того, с какой версией системы они работают.

УДК 62-1:004.9AutoCAD
ББК 30.2с515

Все права защищены. Любая часть этой книги не может быть воспроизведена в какой бы то ни было форме и какими бы то ни было средствами без письменного разрешения владельцев авторских прав.

Материал, изложенный в данной книге, многократно проверен. Но поскольку вероятность технических ошибок все равно существует, издательство не может гарантировать абсолютную точность и правильность приводимых сведений. В связи с этим издательство не несет ответственности за возможные ошибки, связанные с использованием книги.

ISBN 978-5-94074-852-6

© Габидулин В. Н., 2012
© Оформление, издание, ДМК Пресс, 2012

Содержание

Введение 7

Для кого предназначена данная книга 8

Принятые термины и сокращения 9

ГЛАВА 1
Создание форм 11

1.1. Создание простых форм 12

1.1.1. Компиляция 16

1.1.2. Загрузка 18

1.1.3. Вставка формы 18

1.2. Создание сложных форм 23

1.3. Создание форм при помощи Express Tools 29

1.4. О шрифтах 32

ГЛАВА 2
Типы линий 35

2.1. Создание простых типов линий 36

2.2. Создание составных типов линий 40

2.3. Составные типы линий с использованием форм 47

2.4. Использование типов линий 51

2.5. Масштаб типа линии 53

2.6. Особенности использования типов линий для различных объектов 58

2.7. Перенос типов линий 60

ГЛАВА 3
Штриховки 62

3.1. Работа со стандартными штриховками 63

3.2. Создание пользовательских штриховок 66

3.3. Расположение файлов образцов пользовательских штриховок 71

ГЛАВА 4
Поля 73

4.1. Дата и время (Date & Time) 75

4.2. Документ (Document) 77

4.3. Объекты (Objects) 79

ГЛАВА 5
Таблицы 85

5.1. Создание стиля таблиц 87

5.2. Создание и редактирование таблиц 90

5.3. Вставка таблиц Excel 98

5.4. Создание таблицы из свойств объектов 100

ГЛАВА 6
Блоки и внешние ссылки 104

6.1. Простые статические блоки 105

6.1.1. Создание определения блока 105

6.1.2. Вставка блока 108

6.1.3. Создание библиотеки блоков 110

6.1.4. Центр управления 111

6.1.5. Переопределение вхождения блока 113

6.2. Блоки с атрибутами 116
6.2.1. Создание блока с атрибутами 117
6.2.2. Редактирование атрибутов....120
6.3. Внешние ссылки....123
6.3.1. Работа с внешними ссылками 126

ГЛАВА 7
Параметризация 130

7.1. Геометрические зависимости....131
7.2. Размерные зависимости 137

ГЛАВА 8
Динамические блоки..... 142

8.1. Знакомство с редактором блоков 143
8.2. Линейный параметр 147
8.3. Параметр видимости....152
8.4. Отражение 156
8.5. Создание динамических блоков с использованием параметрических зависимостей....157
8.5.1. Таблица блоков 161

ГЛАВА 9
Адаптация рабочего пространства 169

9.1. Создание собственной инструментальной палитры....172
9.2. Модификация панелей инструментов и палитр....176
9.2.1. Создание выпадающего меню 178

9.2.2. Размещение команд
на инструментальную палитру 179
9.3. Создание собственной панели инструментов 180
9.4. Создание новых команд 183
9.5. Создание пользовательских кнопок.....186
9.6. Создание рабочих пространств 188
9.7. О чем не сказано в этой главе 192

ГЛАВА 10
Шаблоны и стандарты 196
10.1. Стандартные шаблоны 197
10.2. Описание файла-прототипа.....198
10.3. Создание шаблона 203
10.4. Создание файла стандартов.....205
Заключение 208
Список литературы 208

Формы в AutoCAD – это объекты, занимающие очень незначительное место в памяти компьютера, и их использование напоминает использование блоков. Если в проекте часто используется геометрически простой элемент, то использование форм очень эффективно. Именно при помощи форм создаются шрифты, т. е. каждая буква алфавита представляет собой форму (на самом деле компилированную форму, но об этом чуть позже).

Английское название формы – SHAPE. В литературе по AutoCAD используются два термина: «формы» и «фигуры», которые в данном контексте являются синонимами. Автор рекомендует использование термина ФОРМА (SHAPE) во избежание путаницы в терминологии. Дело в том, что существует примитив системы ФИГУРА (SOLID), который не имеет никакого отношения к излагаемому материалу.

Порядок создания формы следующий:

1. Написание текста формы в текстовом редакторе и сохранение этого текстового файла в формате SHP.
2. Компиляция SHP-файла. В результате получается файл в формате SHX, представляющий собой объектный код.
3. Загрузка компилированной формы в чертеж.
4. Вставка формы.

Пункты 2–4 – это формальные шаги, для которых существуют соответствующие команды. Наиболее трудоемкий и требующий понимания – пункт 1, к описанию которого мы сейчас и перейдем.

1.1. Создание простых форм

Текст формы пишется в любом текстовом редакторе. Как правило, используется Блокнот, входящий в набор программ «Стандартные» системы Windows. Блокнот можно вызвать и находясь в системе AutoCAD. Для этого необходимо набрать в **КС:(ДВ)** слово NOTEPAD. Если вы работаете в русской версии системы, то добавьте нижнее подчеркивание перед словом, вот так: _NOTEPAD. В версиях AutoCAD 2012 и более поздних достаточно набрать первые три буквы NOT (можно без подчеркивания; оставшиеся буквы система допишет сама) и нажать клавишу ┘. Кнопки команда не имеет.

Для написания текста формы используются как десятичные, так и шестнадцатеричные цифры. Список соответствия шестнадцатеричных и десятичных цифр представлен в табл. 1.1.

Таблица 1.1. Соответствие цифр

Шестнадцатеричные	0	1	2	3	4	5	6	7	8	9
Десятичные	0	1	2	3	4	5	6	7	8	9
Шестнадцатеричные	A	B	C	D	E	F				
Десятичные	10	11	12	13	14	15				

Сам текст формы выглядит следующим образом:

```
*номер_формы,число_байтов,имя_формы
байт1,байт2,байт3,…,0
```

* – обязательный символ, с которого начинается описание формы.

Номер_формы – уникальный номер формы в пределах файла. Может принимать значения от 1 до 258. Значения 256, 257, 258 использовать не рекомендуется. (При создании шрифтов номер формы должен совпадать с ASCII-кодом соответствующего символа.)

Число_байтов – количество байтов описания формы, т. е. количество байтов во второй строке, включая последний, имеющий значение 0. Последний байт 0 является окончанием определения формы, и его присутствие обязательно. (В справочной системе и в специальной литературе вместо слова «байт» зачастую используют слово «код». В данном контексте это синонимы).

Имя_формы – произвольное имя, написанное с использованием английской раскладки клавиатуры ТОЛЬКО ПРОПИСНЫМИ БУКВАМИ.

При создании исходного текста формы можно использовать символ ; (точка с запятой). Если поставить этот символ в начале строки, любой следующий за ним текст воспринимается как комментарий.

Что представляют собой байты во второй строке?

Каждый байт описывает какую-либо часть формы. Все байты в совокупности определяют единую геометрическую фигуру.

Байт состоит из трех шестнадцатеричных чисел.

Число1 – это обязательно символ 0. Символ 0 указывает программе, что все остальные символы – шестнадцатеричные числа.

Число2 – длина. Длина не должна превышать 15 единиц (F – в шестнадцатеричной системе).

Число3 – направление из начальной точки. Значения от 0 до F.

Сама форма рисуется в поле 4 × 4 (рис. 1.1) по определенным правилам.

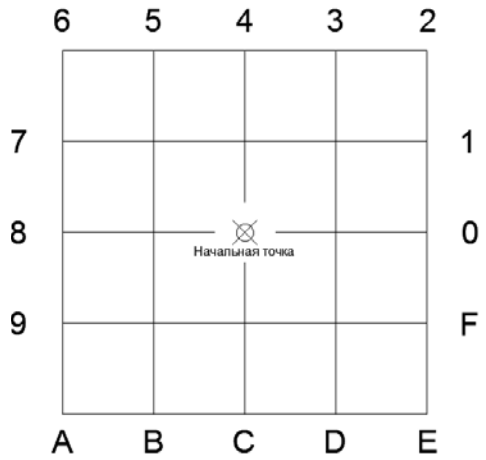


Рис. 1.1. Поле создания формы

Правила создания формы:

- чертить можно только в шестнадцати направлениях;
- чертить можно только с использованием отрезков;
- длина каждого отрезка – не более 15 единиц (F);
- условное «перо» нельзя отрывать от бумаги.

Для того чтобы освоить создание формы, необходимо нарисовать ее вручную, карандашом. Лучшим шаблоном послужит лист бумаги в клеточку. Нарисуем форму, похожую на русскую букву «П» (рис. 1.2).

Последовательность движения реального карандаша по реальному листу бумаге отмечена на рисунке. «Расскажем» о своем намерении системе AutoCAD. Пишем вторую строку.

Первый отрезок – это первый байт. Он начинается в начальной точке; его длина – 2 единицы. Если мы посмотрим из начальной точки по стрелке, то увидим цифру 4. Следовательно, первый байт имеет вид 024.

Второй отрезок – второй байт. Он начинается в конце первого отрезка и имеет длину 2 единицы. Если мы посмотрим из начальной точки по стрелке, то увидим цифру 0. Следовательно, второй байт имеет вид 020.

Третий отрезок – третий байт. Он начинается в конце второго отрезка и имеет длину 2 единицы. Если мы посмотрим из начальной точки по стрелке, то увидим цифру C. Следовательно, третий байт имеет вид 02C.

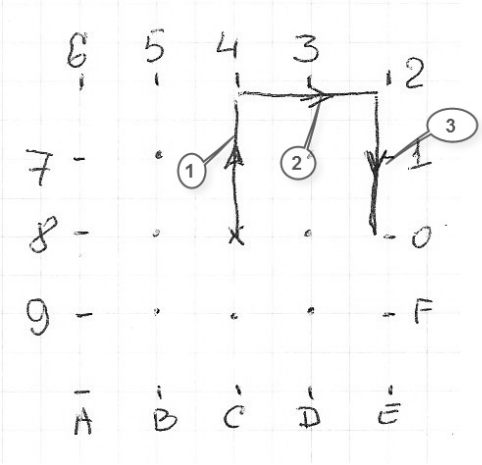


Рис. 1.2. Создание формы П

Основное для понимания выделено курсивом, а именно – *направление всегда задается из начальной точки.*

Сформируем окончательно вторую строку:

```
024,020,02C,0
```

Теперь пишем первую строку:

```
*1,4,PE1
```

Еще раз о правилах:

* – не обсуждается, это обязательный элемент.

1 – номер нашей фигуры. Мы могли бы присвоить ей какой-либо другой. Главное – не присваивать следующим формам тот же номер.

4 – число байтов во второй строке, включая 0.

PE1 – имя фигуры. Помните: оно указывается только в английской раскладке клавиатуры и только заглавными буквами.

Полный текст нашей конкретной формы под названием PE1 выглядит так:

```
*1,4,PE1
024,020,02C,0
```

Наберем этот текст в Блокноте. При наборе текста, несмотря на его простоту, можно допустить ошибки. Основная из них – наличие лишних пробелов. Предупредить эту ошибку просто: выделите набранный текст комбинацией клавиш **Ctrl**+**A** и удалите лишние пробелы (рис. 1.3).

Созданный файл необходимо сохранить, присвоив ему имя PE1, т. е. имя файла должно совпадать с именем фигуры. Блокнот

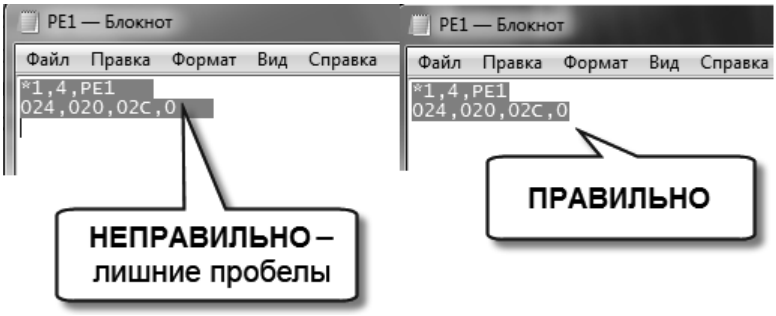


Рис. 1.3. Текст формы

предлагает по умолчанию расширение txt. Его надо вручную заменить на shp:

Файл → Сохранить как → введите имя файла PE1.shp (рис. 1.4).

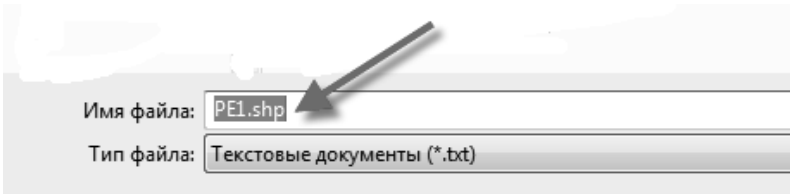


Рис. 1.4. Создание файла shp

Сохраненный файл получает «фирменный» значок, показанный на рис. 1.5.



Рис. 1.5. Обозначение файла формы

Формы необходимо располагать в созданных вами папках. Главное – знать месторасположение папки, в которой находится форма, к примеру, C:\Users\[пользователь]\Desktop\Формы\PE1. Эта информация пригодится в дальнейшем.

На прилагаемом диске формы расположены в папке Формы.

Остальные операции с формами осуществляются непосредственно в системе AutoCAD.

1.1.1. Компиляция

Откроем AutoCAD. Наберем в **КС:(ДВ)** команду КОМПИЛ (COMPILE). Активизируется диалоговое окно **Выбор файла формы или шрифта** (Select Shape or Font File). Выделим нужный нам файл

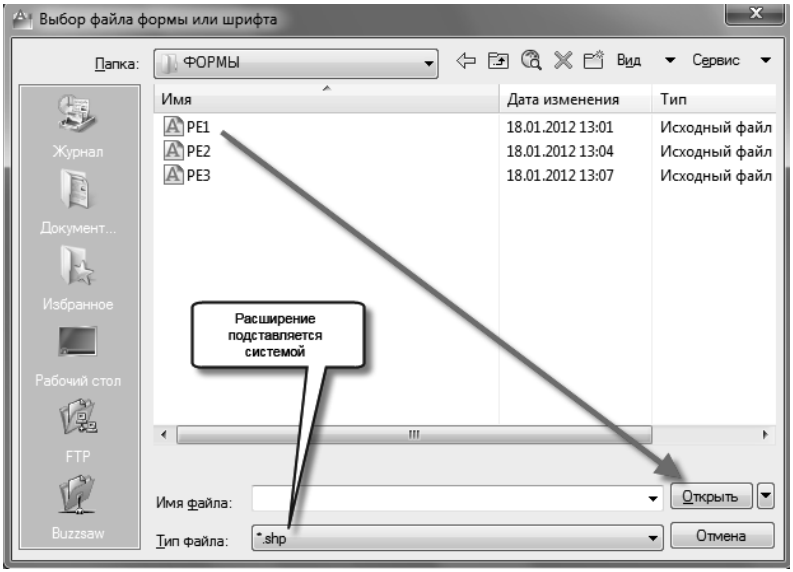


Рис. 1.6. Компиляция

```
Компилируется файл описания форм или шрифта
Компиляция завершена.
Файл вывода C:\Users\пользователь\Desktop\ФОРМЫ\PE1.shx содержит 45 байт.
Команда:
```

Рис. 1.7. Сообщение о компиляции

с расширением. shp и щелкнем по кнопке **Открыть** (Open). Программа автоматически скомпилирует этот файл в новый и создаст в той же папке файл с тем же именем, с расширением .shx (рис. 1.6).

В **КС:** будет выведено сообщение, представленное на рис. 1.7. Обратите внимание на размер созданного файла: в нашем случае – 45 байт.

Созданный системой файл имеет другой «фирменный» значок, представленный на рис. 1.8.



Рис. 1.8. Обозначение файла компилированной формы

При наличии синтактических ошибок в исходном файле формы компиляция не произойдет, на что укажет соответствующая диагностика в **КС:**. Будьте внимательны: в окне динамического ввода диагностика не появляется! В этом случае необходимо открыть соответствующий файл .shp, внести коррективы и заново откомпилировать форму.

1.1.2. Загрузка

Созданную компилированную форму необходимо загрузить в файл.

Наберем в **КС:(ДВ)** команду ЗАГРУЗИТЬ (LOAD). Активизируется диалоговое окно **Выбор файла форм** (Select Shape File). В этом окне предлагаются к загрузке только файлы форм с расширением .shx (рис. 1.9). Выделим нужный нам файл с расширением .shx и щелкнем по кнопке **Открыть** (Open).

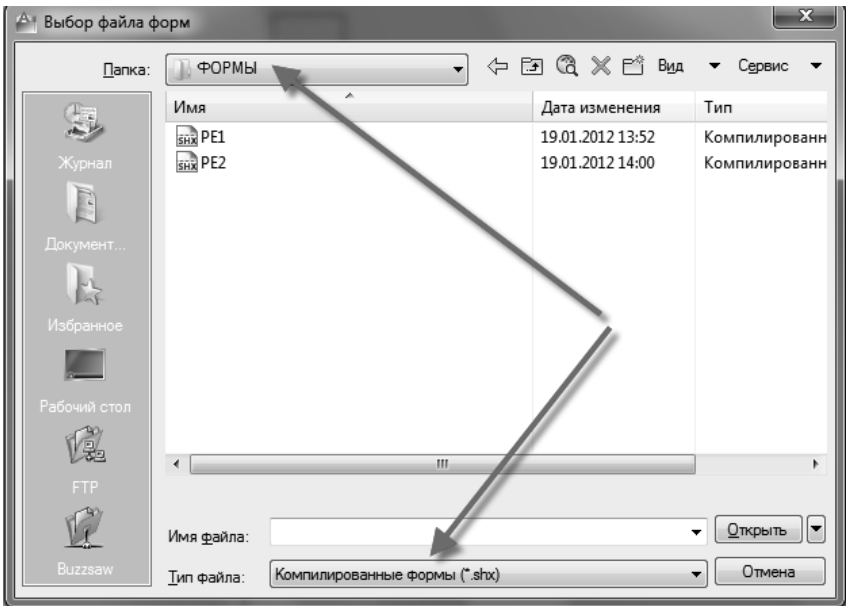


Рис. 1.9. Загрузка компилированной формы

1.1.3. Вставка формы

Наберем в **КС:(ДВ)** команду ФОРМА (SHAPE) и ответим на следующие вопросы **КС:(ДВ)**:

– Имя формы или [?] <PE1>: (Enter shape name or [?]:)

Введите имя фигуры или ? и ┘ для получения списка всех форм.

– Точка вставки: (Specify insertion point:)

Задайте точку вставки.

– Высота <1.0000>: (Specify height<1.0000>:)

Запрос аналогичен запросу команды МАСШТАБ. Если зададите, к примеру, 5, получите форму в пять раз больше.

– Угол поворота <0>: (Specify rotation angle <0>:)

Введите угол поворота.

Результат приведен на рис. 1.10.

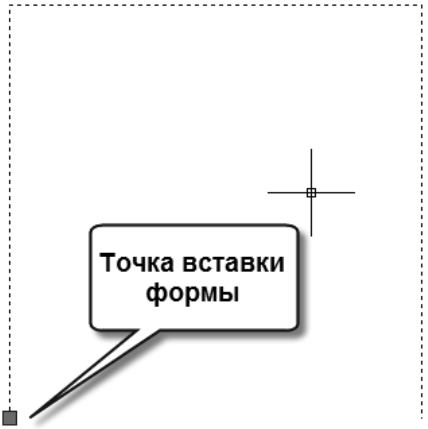


Рис. 1.10. Готовая форма PE1

Для того чтобы научиться создавать формы, необходим некоторый практический навык, который невозможно приобрести, создав одну лишь форму. На рис. 1.11 представлено несколько достаточно простых готовых форм.

Исходные (.shp) и компилированные (.shx) файлы форм находятся на диске в папке Формы. Тексты исходных файлов форм приведены ниже:

PE1.shp

```
*1,4,PE1
024,020,02C,0
```

PE2.shp

```
*2,4,PE2
024,020,04C,0
```

PE3.shp

```
*3,4,PE3
024,020,0FC,0
```

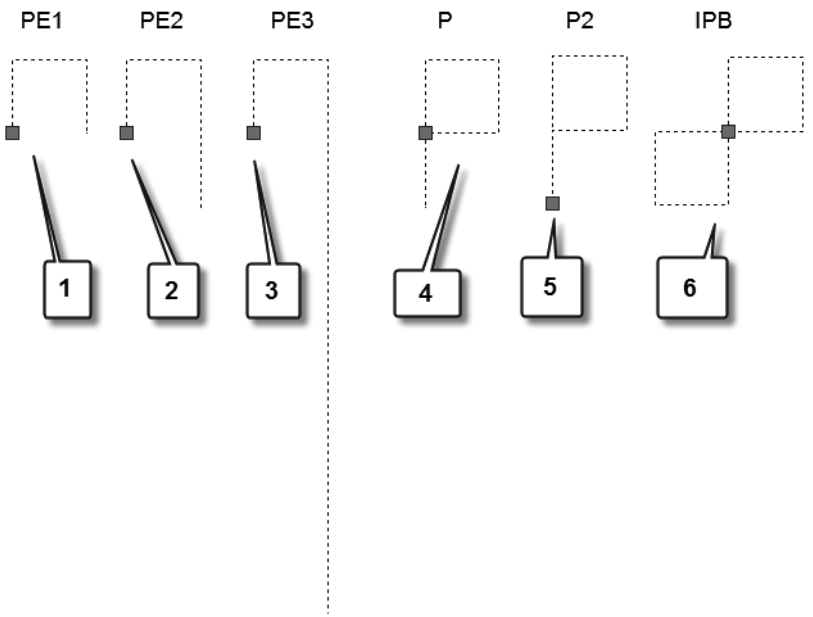


Рис. 1.11. Примеры форм

P.shp

```
*4,6,P
024,020,02C,028,02C,0
```

P2.shp

```
*5,6,P2
044,020,02C,028,02C,0
```

IBP.shp

```
*6,9,IPB
02C,028,024,020,024,020,02C,028,0
```

Обратите внимание на длины векторов в формах PE1, PE2 и PE3. Формы P и P2 отличаются только точкой вставки.

Попробуйте создать формы, похожие на приведенные выше, но с другими точками вставки.

А теперь создайте свои собственные формы.

Несколько практических советов.

Созданный некорректно исходный файл .shp можно отредактировать в Блокноте. При редактировании не забывайте о числе байтов во второй строке и признаке конца формы (байта 0). Если вы

назвали файл формы не так, как описано его название в первой строке, то скомпилировать файл удастся; формально проходит и загрузка, но при попытке вставить форму диагностика будет выглядеть так, как показано на рис. 1.12.

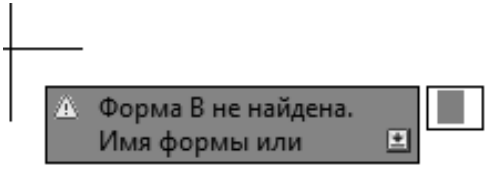


Рис. 1.12. Диагностика ошибки

Для удаления из файла компилированной формы необходимо:

1. Удалить все вхождения формы из файла.
2. Вызвать команду ОЧИСТИТЬ (PURGE), см. рис. 1.13.

На рис 1.13 проиллюстрирован пример форм, которые нельзя удалить из чертежа, потому что вхождения этих форм присутствуют в пространстве модели.

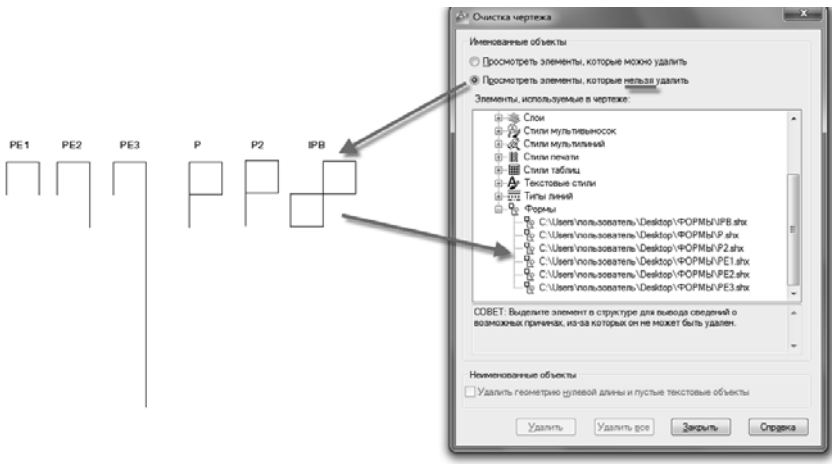


Рис. 1.13. Очистка файла от форм

Иногда команду ОЧИСТИТЬ (PURGE) приходится повторять неоднократно. Если и это не помогает, и в **КС:** печатается диагностика с текстом, приведенным на рис. 1.12, то попробуйте закрыть файл с сохранением, и заново его открыть. Наконец,

радикальное решение – сохранить файл .shp под другим именем, заново его скомпилировать и загрузить.

Зачастую вставляется предыдущая, не устраивающая вас версия формы. В 99 случаях из 100 это происходит потому, что не было изменено название формы – третий (не считая символа *) элемент первой строки.

Мы создали формы, используя только ортогональные направления. Создадим формы с использованием диагональных направлений (рис. 1.14).

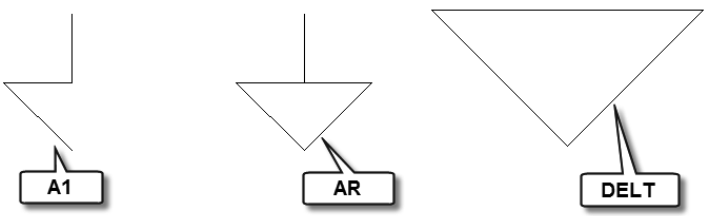


Рис. 1.14. Формы с использованием диагональных направлений

Приведем тесты исходных файлов:

DELT.shp

```
*7,4,DELT
022,048,02E,0
```

A1.shp

```
*8,4,A1
01C,018,01E,0
```

AR.shp

```
*9,6,AR
01C,018,01E,012,018,0
```

В созданных формах длины диагональных отрезков не являются истинными значениями, поскольку конечные точки отрезков выравниваются по узлам воображаемой сетки. По сути, это длины проекций на ближайшую из осей.

Редактировать форму можно при вставке, задавая масштаб, угол поворота. То же можно делать с использованием палитры **Свойства**. В в разделе **Разное** этой палитры можно задать коэффициент

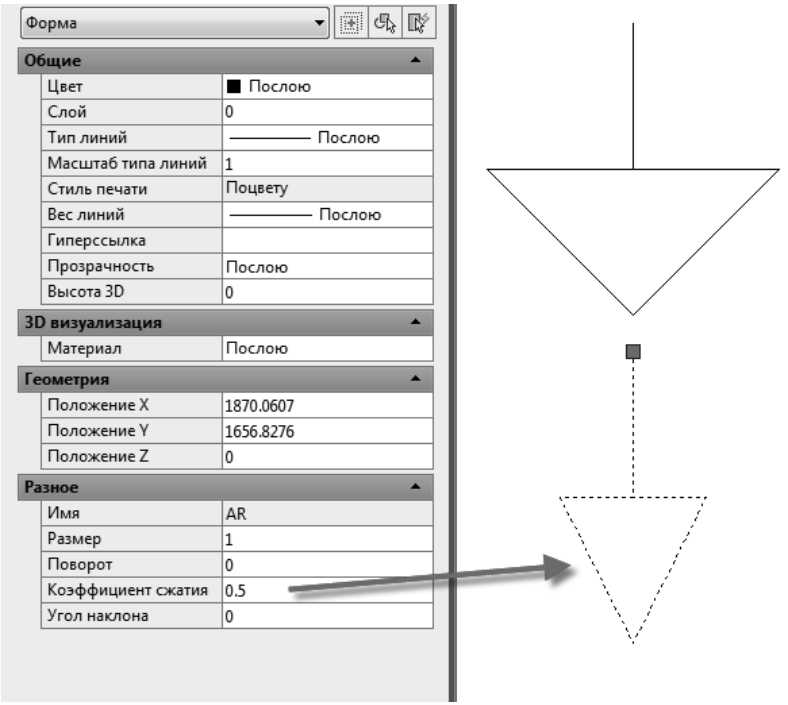


Рис. 1.15. Использование коэффициента сжатия

сжатия для вхождения формы, тем самым расширив геометрию объектов, не создавая новых форм. Использование коэффициента сжатия проиллюстрировано на рис. 1.15.

1.2. Создание сложных форм

Для того чтобы создавать формы с более сложной геометрией, необходимо использовать еще один набор кодов (байтов) в дополнение к кодам предыдущего раздела. Приведем таблицу специальных кодов и ознакомимся с основными правилами их использования.

Освоив работу с этим набором кодов, можно написать даже самый сложный иероглиф. Создание шрифтов – отдельная тема, но используя эти коды даже частично, можно создавать более разнообразные формы.

Рассмотрим более подробно некоторые коды, представленные в табл. 1.2.

Таблица 1.2. Специальные коды

Шестнадцатеричный код	Десятичный код	Описание
000	0	Окончание формы
001	1	Опустить перо
002	2	Поднять перо
003	3	Деление на следующее число
004	4	Умножение на последующее число
005	5	Занесение текущей позиции в стек
006	6	Вставка текущей позиции из стека
007	7	Отрисовка формы, номер которой задан следующим байтом
008	8	Рисование отрезка со смещением, заданным следующими байтами
009	9	Рисование нескольких отрезков с заданными координатами смещений
00A	10	Октантная дуга
00B	11	Дробная дуга
00C	12	Дуга по прогибу
00D	13	Несколько дуг по прогибу
00E	14	Специальный код только для шрифтов

000 – с этим кодом мы уже встречались. Он обязателен для окончания формы. Собственно говоря, при его отсутствии форма не будет скомпилирована.

001/002 – поднимает/опускает перо. При включенном режиме отрисовки (001) рисуются линии, заданные векторами. При отключенном режиме отрисовки (002) перо переходит в точку, заданную этими векторами без проведения линии.

003/004 – делит/умножает длину вектора на значение последующего числа.

Код 3,4 означает, что длины делятся на 4 и размер формы уменьшается в четыре раза. Код 4,4 означает, что длины умножаются на 4 и размер формы увеличивается в четыре раза.

005/006 – загружает в стек / извлекает из стека текущие координаты пера.

007 – строит субформу, номер которой определен следующим байтом.

008 – рисует произвольный отрезок:

8, X-смещение,Y-смещение,

где X,Y – в диапазоне –128 до +128.

8, (–5,2) – рисует вектор, соединяющий текущую точку с точкой, отстоящей от нее на пять единиц влево и на две единицы вверх. После отработки кода 8 восстанавливается режим построения стандартных векторов.

009 – рисует последовательность нестандартных векторов. После него следует любое количество пар значений по X–Y. Последовательность должна закончиться парой (0,0).

00A – октантная дуга. Следующие два байта задают границы октантов. На рис. 1.16 приведен рисунок, позволяющий понять, что такое октанты. Каждый октант содержит 45°. Их нумерация производится против часовой стрелки, начиная с востока. Дуга называется *октантной*, ели она стягивает один или несколько октантов.

Описание дуги имеет вид:

```
00A,радиус,(-),0SC
```

или

```
10,радиус,(-),0SC
```

Значение «радиус» может быть от 1 до 255. Символ «–» показывает, что дуга рисуется по часовой стрелке.

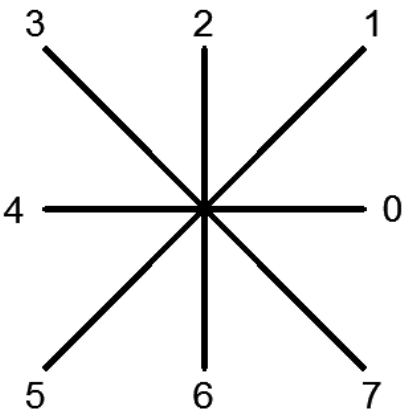


Рис. 1.16. Октанты

S принимает значения от 0 до 7 и указывает начальный октант.

C принимает значения от 0 до 7 и указывает количество охватываемых дугой октантов.

00B – дробная дуга; специальный байт для дуг, начинающихся не на границе октантов.

00C и 00D – для отрисовки дуг, заданных прогибом.

При написании сложных форм можно использовать скобки. Они не воспринимаются при компиляции и нужны исключительно для наглядности. Для полноты картины в табл. 1.2 представлены все коды форм, но в последующих примерах мы будем использовать их лишь частично.

Форма E.shp

```
*58,6,E
024,002,013,001,020,0
```

Рассмотрим вторую строку.

Первый байт 024 рисует отрезок длиной 2 единицы в направлении 4 (естественно, от начальной точки).

Второй байт 002, находясь в достигнутой точке, поднимает условное перо.

Третий байт 013 перемещает поднятое перо на 1 единицу в направлении 3.

Четвертый байт 001 опускает условное перо.

Пятый байт 020 рисует отрезок длиной 2 единицы в направлении 0.

Шестой байт – признак окончания отрисовки фигуры.

В результате компиляции и загрузки данной исходной формы получим форму в виде двух взаимно перпендикулярных непересекающихся отрезков, показанную на рис. 1.17.

Опишем алгоритм создания еще одной формы с использованием дополнительных кодов.

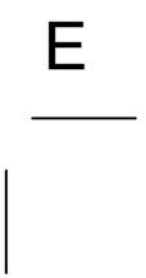


Рис. 1.17. Форма E