

Описание проекта
**«Разработка прикладной приложения для построения 3D моделей и чертежей
экструзионных головок»**

Разработчик: ст. гр. Ам-1 Ширяев П.С.

Руководитель: старший преподаватель Голубев А. Н.

Учреждение образования «Витебский государственный технологический университет»

Введение

На начальном этапе разработки была поставлена задача: обеспечить возможность автоматизированного проектирования экструзионных формующих головок (матриц) и шнеков. Для решения поставленной задачи были выбраны следующие инструменты: среда программирования Delphi для разработки приложения; система автоматизированного проектирования Компас-3D для разработки параметрических моделей и чертежей.

На данный момент программа позволяет работать с двумя видами головок, взаимодействует с 3D моделями и по итогу выдает техническую документацию в виде ассоциативного чертежа. Для работы программы необходимо наличие на компьютере установленной версии Компас-3D не старше V17.1.

Окончательный вариант приложения должен облегчить работу пользователю, ускорить выполнения операций и получения расчетов, графиков и технической документации.

Принцип работы прикладного приложения

Первоначально были построены параметрические модели и ассоциативные чертежи с них. Примеры показаны на рисунках 1 и 2.

Все параметры, необходимые для построения, заданы в виде внешних переменных, каждой переменной было задано имя и первоначальные параметры, как показано в красном поле на рисунке 1.

В текущей версии приложения отработан модуль взаимодействия с 3D- моделями, работа с чертежами находится на стадии разработки (при изменении геометрических параметров головки, в особенности в большую сторону, чертеж может перестраиваться некорректно).

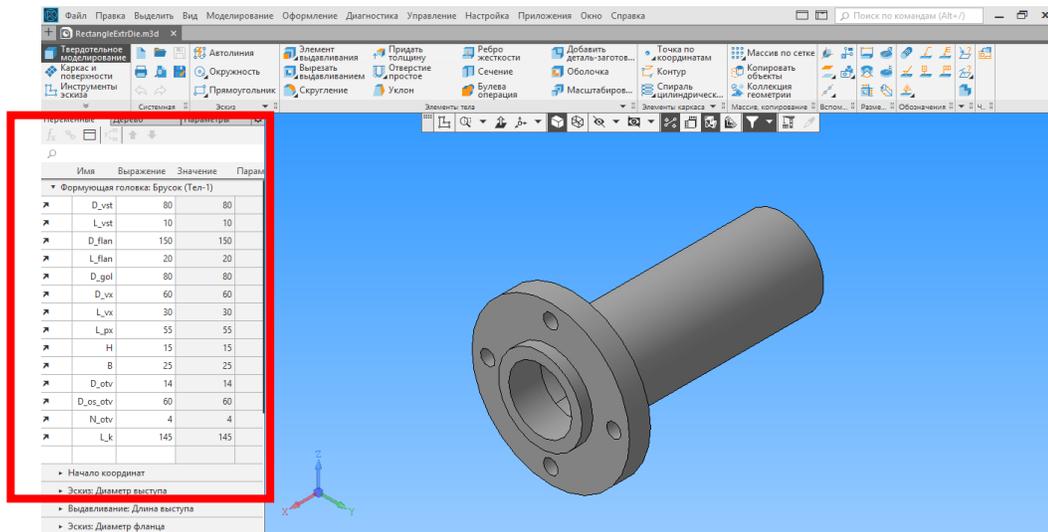


Рисунок 1 – Параметрическая модель в Компас 3D

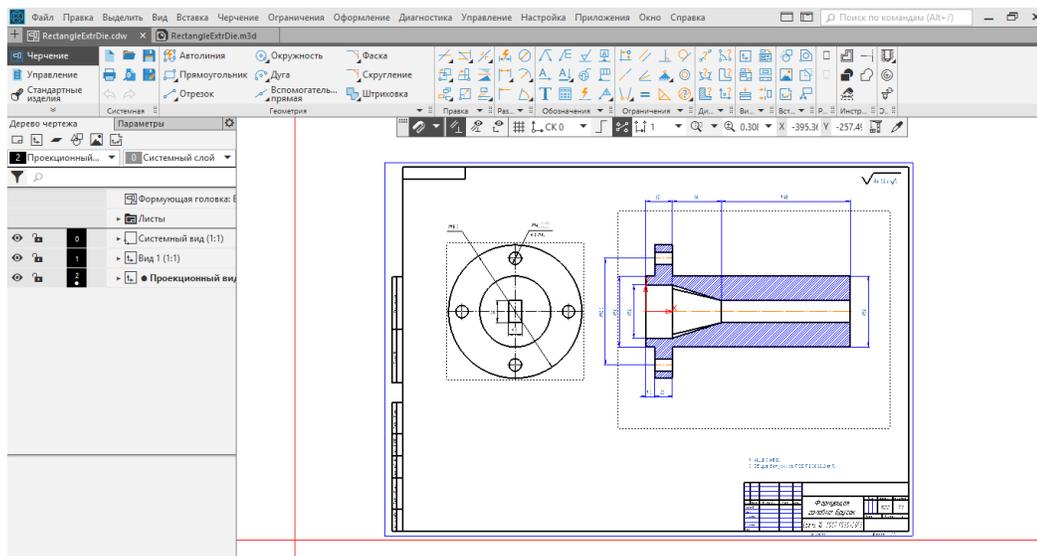


Рисунок 2 – Ассоциативный чертеж в Компас 3D

Разработанное прикладное приложение управляет 3D-моделями и чертежами, обеспечивая их открытие, закрытие и адекватное перестроение. Ниже приведен пример кода, обеспечивающего открытие модели в Компас 3D (рисунок 3). Перестроение моделей осуществляется через доступ к внешним переменным с помощью API, фрагмент кода показан на рисунке 4.

```

function GetKompas: boolean;
begin
  try
    Kompas := KompasObject(GetActiveOleObject(ka)); // если уже запущен
    Result := True;
  except
    try
      Kompas := KompasObject(CreateOleObject(ka)); // если не запущен
      Result := True;
    except
      Result := False;
      exit;
    end;
  end;
  // получаем ссылку на интерфейс окна
  KompasHandle := Kompas.KsGetHWindow;
  // делаем окно видимым
  Kompas.Visible := True;
end;

function OpenKompasModel(FileName: string): boolean;
begin
  Result := False;
  try
    // создаем 3D-документ
    Document3D_1 := ksDocument3D(Kompas.Document3D());
    // открываем модель, которая лежит в одной папке с exe
    Document3D_1.Open(GetCurrentDir + '\ ' + FileName + '.m3d', False);
    RecaleModel;
    Result := True;
  except
    Result := False;
  end;
end;

```

Рисунок 3 – Пример кода открытия модели в Delphi

```

Procedure RecaleModel;
begin
  //получаем ссылку на активный документ
  Document3D_1 := ksDocument3D(Kompas.ActiveDocument3D);
  Part := ksPart(Document3D_1.GetPart(pTop_Part));
  if Part = nil then
    Kompas.KsMessage('Пусто');

  //Получаем массив переменных модели
  VariableCollection := ksVariableCollection(Part.VariableCollection());
  //Изменяем значение первой переменной
  SetVariableValue ('D_flan', D_flan);
  //Изменяем значение второй переменной
  SetVariableValue ('H', H);
  //Изменяем значение 3й переменной
  SetVariableValue ('B', B);
  //Изменяем значение 4й переменной
  SetVariableValue ('D_otv', D_otv);
  //Изменяем значение 5й переменной
  SetVariableValue ('D_vst', D_vst);
  //Изменяем значение 6й переменной
  SetVariableValue ('L_vst', L_vst);
  //Изменяем значение 7й переменной
  SetVariableValue ('L_flan', L_flan);
  //Изменяем значение 8й переменной
  SetVariableValue ('D_gol', D_gol);
  //Изменяем значение 9й переменной
  SetVariableValue ('D_vx', D_vx);
  //Изменяем значение 10й переменной
  SetVariableValue ('L_vx', L_vx);
  //Изменяем значение 11й переменной
  SetVariableValue ('D_os_otv', D_os_otv);
  //Изменяем значение 12й переменной
  SetVariableValue ('D_os_vst', D_os_vst);

```

Рисунок 4 – Пример кода перестроения модели в Delphi

В приложение включена обработка всех возможных ситуаций, при которых модель будет перестроена некорректно, для каждой такой ситуации предусмотрен вид сообщений об некорректном вводе, в это сообщение входит пояснение, как ту или иную ошибку решить (фрагмент когда показан на рисунке 5).

Если перейти по первой кнопке попадаем в окно для работы с первой головкой (рисунок 7). Первоначально программа позволяет ввести вручную все геометрические параметры, которые необходимы пользователю. После при помощи кнопки «Присвоение параметров», параметры можно принять, программа проверит возможность выполнения операции и выведет сообщение с подтверждением, что все введено верно, как показано на рисунке 8, или выдаст ошибки и укажет, какой из участков не может быть построен и почему, как показано на рисунке 9.

Если все переменные введены верно, то кнопка «Запуск модели в КОМПАС» станет активной и пользователь сможет запустить модель с заданными параметрами. В ходе запуска приложение будет оповещать о своих действиях, например, о запуске компаса (рисунок 10) или о запуске модели (рисунок 11). Так же с запуском модели станет активна кнопка «Запуск чертежа в КОМПАС». Для того, чтобы снова произвести какие-либо изменения, нужно снова повторить тот же порядок выполнения.

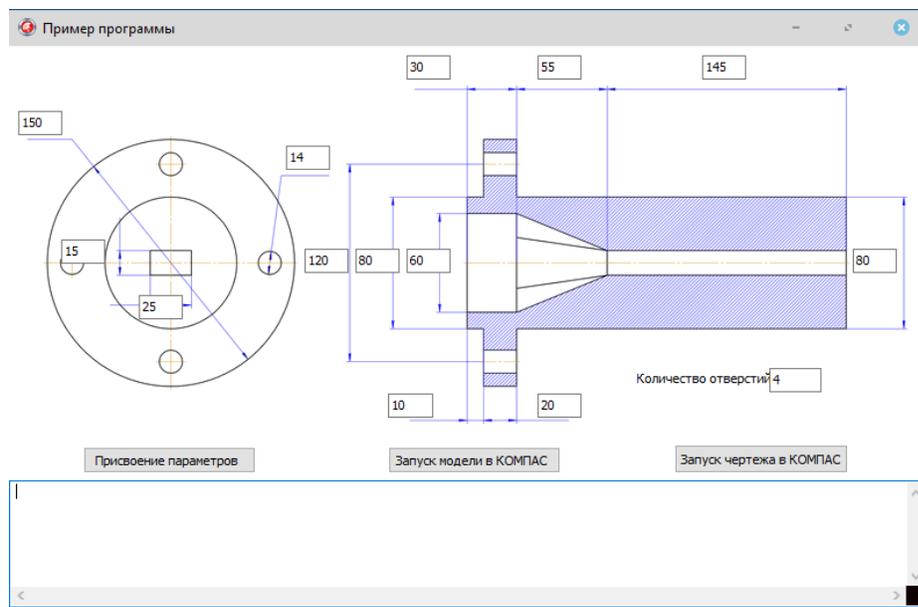


Рисунок 7 – Окно для работы с головкой с прямоугольным каналом



Рисунок 8 – Сообщение при корректном вводе

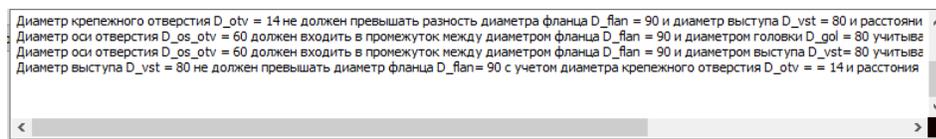


Рисунок 9 – Ошибки, выдаваемые при некорректном вводе

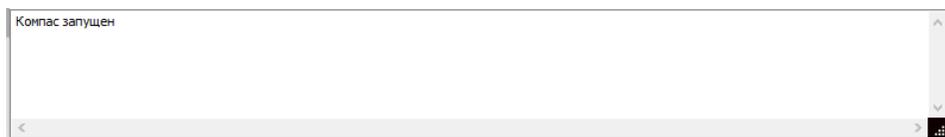


Рисунок 10 – Сообщение о запуске Компас 3D

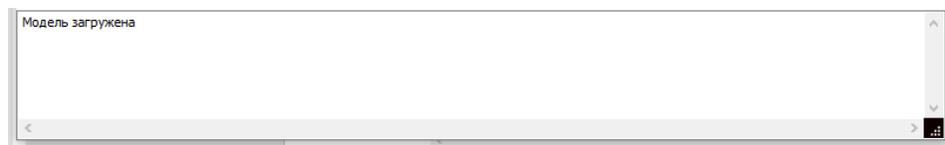


Рисунок 11 – Сообщение о запуске модели в Компас 3D

Рабочее поле для головки с круглым каналом (рисунок 12) идентично полю для головки с прямоугольным каналом, отличие состоит только в задаваемых параметрах.

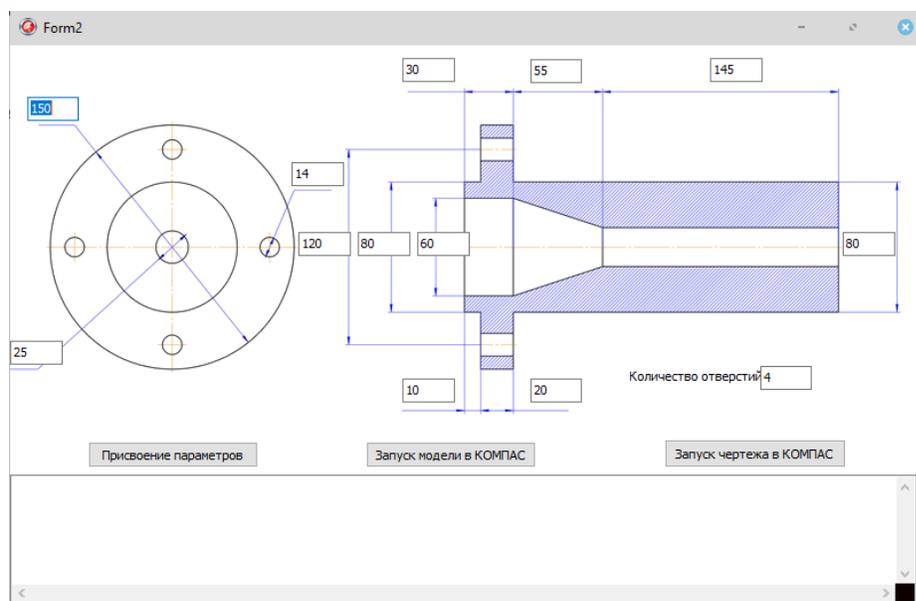


Рисунок 12 – Окно для работы с головкой с круглым каналом

Заключение

В текущей версии реализовано построение матриц прямоугольного и круглого сечения. Приложение позволяет снизить трудоемкость проектирования специализированного экструзионного оборудования. В проект входят параметрические модели экструзионных матриц, ассоциативные чертежи, а также непосредственно прикладное приложение, осуществляющее управление моделями и чертежами.

В дальнейшем будет доступна возможность расчетов в зависимости выбранного материала, также вывод расчетных данных и зависимостей в виде графиков и диаграмм. Расчетная часть будет подключена к базе данных материалов, которая на данный момент практически готова, откуда будут заимствоваться коэффициенты, необходимые в расчетах, также будет подключен модуль с построением и работой с экструзионными шнеками.