

**АВИАМАШИНОСТРОЕНИЕ  
И ТРАНСПОРТ СИБИРИ**

Сборник статей  
III Всероссийской научно-практической  
конференции

**(Иркутск, 11–12 апреля, 2013 г.)**



# АВИАМАШИНОСТРОЕНИЕ И ТРАНСПОРТ СИБИРИ

Сборник статей  
III Всероссийской научно-практической конференции  
(Иркутск, 11–12 апреля, 2013 г.)



ИЗДАТЕЛЬСТВО  
Иркутского государственного технического университета  
2013



Печатается по решению редакционно-издательского совета

**Авиационная техника и технологии** : сб. статей III Всерос. науч.-практ. конф. (Иркутск, 11–12 апреля, 2013 г.) – Иркутск: Изд-во ИрГТУ, 2013. – 442 с.

Представлены статьи, посвященные актуальным вопросам развития авиационно-транспортных технологий и транспортного комплекса городов и регионов РФ. Приводятся обобщения результатов научных исследований, обмена практическим опытом в интересах развития перспективных конструкций и технологий авиа- и машиностроения, повышения качества транспортного обслуживания регионов РФ.

**Редакционная коллегия:**

Ответственный редактор – Зедгенизов А.В. – канд. техн. наук, доцент;

Ахатов Рашид Хадиатович – канд. техн. наук, доцент, директор Института авиационно-транспортных технологий и транспорта;

Грудинин Владимир Гарриевич – ст. преподаватель кафедры конструирования и стандартизации в машиностроении

**СОДЕРЖАНИЕ:**

**Авиационная техника и технологии**

<b>А.И. Демидов, И.О. Бобарика.</b> Численное моделирование аэродинамики несущих элементов летательных аппаратов потоком несжимаемой жидкости при малых числах Маха.	8
<b>А.С. Говорков, И.В. Фокин.</b> Применение инновационных технологий и интерактивных методов обучения при изучении курса "технология производства самолетов"	14
<b>А.С. Говорков, А.Э. Рябцева.</b> Методика проведения кинематического анализа с учетом податливости тел в NX MotionFlexiblebody на примере узла интерцептора самолета МС-21	18
<b>А.И. Исаев, Нго Куанг Туен.</b> Гидродинамическое исследование формирования вихревых течений камеры сгорания	24
<b>В.А. Жуков.</b> Совершенствование эксплуатации жидкостных систем охлаждения транспортных ДВС.	28
<b>С.И. Феоктистов, С.В. Белых, М.М. Погарцева.</b> Учет влияния фрезерования на форму готовых деталей летательных аппаратов из прессованных профилей, получаемых методами изгиба.	36
<b>А.А. Перевалов, С.В. Белых.</b> Применение MSC.MARC для моделирования процесса изгиба прессованных профилей несимметричного сечения.	45
<b>С.В. Белых, А.В. Станкевич, В.А. Мишагин.</b> Особенности автоматизированного изготовления длинномерных деталей летательных аппаратов из прессованных профилей.	53
<b>Р.Ф. Крупский, А.В. Станкевич, А.А. Кривенок, С.В. Белых.</b> Опыт использования обтяжного пресса для формообразования деталей летательного аппарата из профилей.	61
<b>А.В. Черняев, М.И. Шевченко.</b> Анализ видов, методов и задач подтверждения соответствия программно-математического обеспечения изделий авиакосмической техники.	70
<b>А.И. Селиверстов, И.В. Шевченко.</b> Влияния нелинейности диссипативных сил на динамические характеристики ракеты – носителя.	77
<b>Ю.А. Остяков, М.И. Шевченко.</b> Взаимосвязь надежности и экономической эффективности изделий машиностроения.	82
<b>Т.А. Назарова, Т.А. Школьникова.</b> Влияние отделочно-зачистных технологий на увеличение ресурса работы машины.	90
<b>М.В. Лаврентьева.</b> Автоматизация типовых процессов проектирования электронного макета изделия.	96
<b>К.А. Однокурцев, П.А. Лукин.</b> Средства автоматизированного монтажа сборочной оснастки в самолетостроении.	102
<b>Ю.Ф. Огнев, О.Ш. Бердиев, Ю.П. Денисенко.</b> Новый метод контроля каркасных авиационных панелей из ПКМ.	109

## Автоматизация типовых процессов проектирования электронного макета изделия

М.В. Лаврентьева

Национальный исследовательский Иркутский государственный технический университет (НИ ИрГТУ), 664074, г. Иркутск, ул. Лермонтова, 83.

В статье приведен краткий обзор методики автоматизированного проектирования с использованием программы NXOPEN на языке C++, представлены преимущества данной методики, показан пример сравнения двух методов с управляющей программой и непосредственно в среде NX.

Ключевые слова: автоматизированное проектирование; самолетостроение; язык программирования C++;

**Лаврентьева Мария Вячеславовна**, аспирантка кафедры СМ и ЭАТ, тел.: 89021757833 e-mail: mira.amazon@gmail.com

В современном мире автоматизация занимает приоритетное место в списке задач, которые необходимо решить в области машиностроения. Автоматизация производственных процессов – способ организации производства, при котором функции управления и контроля, обычно возлагавшиеся на человека, перекладываются на автоматические системы и устройства [1].

В последнее время все больше внимания уделяется автоматизации задач связанных с мыслительной деятельностью. В сфере самолетостроения это работа конструктора и технолога. Конечно, полностью автоматизировать проектирование или разработку технологического процесса возможно только в отдельных случаях для относительно простых объектов, однако частично это стало возможным практически для любого вида работ.

Самолетостроение является одной из наиболее нуждающихся в этом областей, так как большой объем работ связанных с проектированием CAD моделей, определением метода базирования и выбором базирующих элементов, является типовым, но достаточно затратным по времени. В этом случае автоматизация работы конструктора и технолога позволит решить основные задачи:

- снизить трудоемкость процессов проектирования и разработки технологической документации;
- увеличить производительность труда;
- исключить вероятность ошибок, вызванных «человеческим» фактором, при выполнении типовых операций;
- минимизировать временные затраты на работу с типовыми элементами.

Это относится к проектированию и монтажу сборочных приспособлений, поскольку они в своей конструкции имеют большое число типовых элементов.

Первостепенной задачей является разработка специализированной программы способной проектировать типовые электронные макеты, запрашивая у конструктора, значения, определенные в техническом задании или полученные в результате прочностных расчетов. Это значительно сократит затраты на создание сборочного приспособления, так как предполагает возможность разработки универсального программного продукта, который не потребует подгонять под конкретное изделие.

Для достижения оптимального результата удобно использовать тот же программный продукт, в котором производятся проектные работы – NX: SiemensPLMSoftware, а точнее один из его программных модулей NX/Open API (ApplicationProgramInterface) располагающий достаточно развитыми возможностями, которые позволяют:

- получить доступ к объектам геометрической модели для программных приложений разработчиков или программ отдельных пользователей;
- дает возможность программным способом создавать геометрические модели, получать информацию об объектах, формировать сборки, генерировать чертежную документацию и т.д.

Практически все функциональные возможности NX, доступные пользователю при интерактивном взаимодействии с системой, реализуемы посредством функций API. Однако утверждать обратное – ошибка, так как существует ряд объектов (к примеру, CustomObjects – Объекты пользователя), формирование которых возможно только программным способом [2].

Для создания тела программы используются программные комплексы, в основе которых заложены работы языками программирования C, C++ и объектно-ориентированные системы управления базами данных.

Программа способна реализовать все общие этапы построения – выбор рабочей системы координат, рабочей плоскости, стартовой точки построения, размеров, помимо этого способна запоминать предыдущие параметры и использовать их для создания ответной модели (болт-отверстие), циклически повторять и/или копировать построение какого-либо элемента.

Для реализации данного метода, необходимо решить две последовательные задачи:

- инженерную, включающую в себя контроль инженерного проектирования входящих типовых элементов, их классификацию, а так же анализ последовательности построения конструктивного электронного макета, определение основных критериев базирования и самих сборочных баз;
- программистскую, включающую разработку непосредственно программного кода на языке C++ и адаптация его к управляющей программе NXOPEN.

Если вторая задача и выглядит однозначной, то приступить к ней возможно лишь по результатам решения первой задачи. Необходимо выделить типовые элементы и разбить их по классам, далее определить последовательность разработки и перечень типовых операций построения этих элементов.

Ниже приведем и проанализируем простейший пример, демонстрирующий работу в программе обычным способом, т.е. когда конструктор самостоятельно задает цепочку построения и альтернативный метод, когда это делает программа NXOPEN.

На рисунке 1 показаны операции при помощи, которых конструктор может создавать модель.

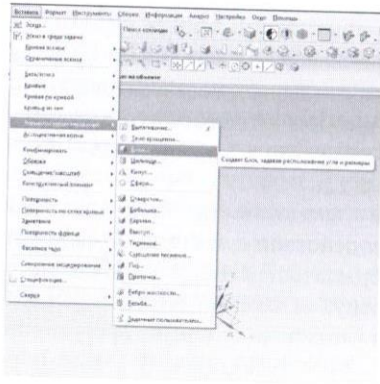


Рисунок 1- Выбор элемента блок 1.

На рисунке 2 показан первый блок со стартовой точкой (0;0;0) длина, ширина и высота которого, равны 100мм.

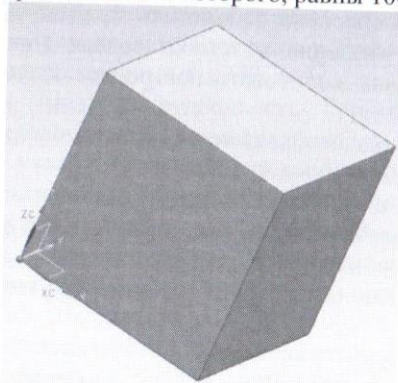


Рисунок 2 - Построение элемента блок 1.

Рисунок 3 иллюстрирует второй блок со стартовой точкой (0;50;100) длина, ширина и высота которого, равны 50мм.

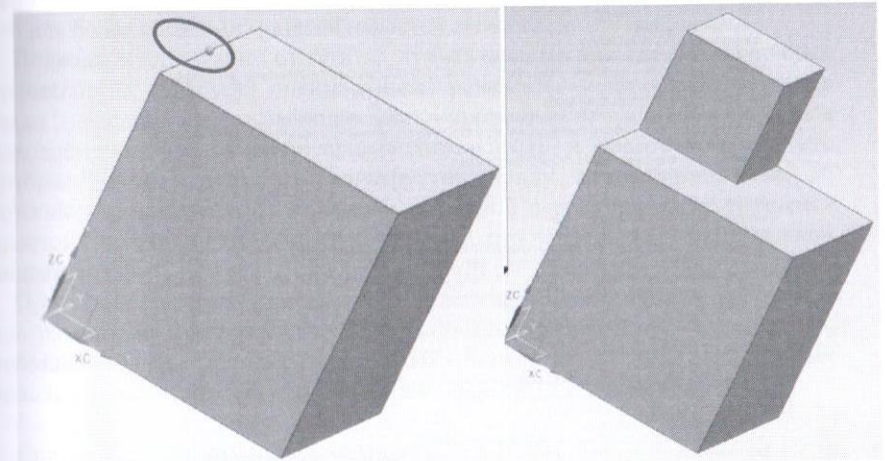


Рисунок 3 - а – выбор стартовой точки для построения элемента блок 2; б – построение элемента блок 2.

На рисунке 4 показан третий блок стартовая точка, которого совпадает со стартовой точкой первого блока (0;0;0) длина, ширина и высота, равны 50мм, так же представлена операция вычитания блока 3 из блока 1.

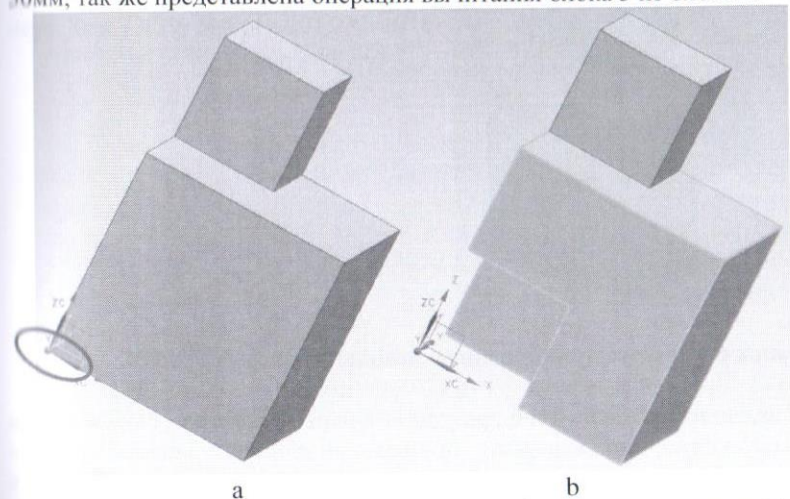


Рисунок 4 - а – выбор стартовой точки для построения элемента блок 3; б – построение элемента блок 3 с вычитанием.

На рисунке 5 приведено тело программы построения блоков на языке C++ в среде разработки Microsoft Visual Studio 2010.

```
// Глобальная область
// Первый блок
std::vector<Body *> targetBodies1(1);
Body *nullBody(NULL);
targetBodies1[0] = nullBody;
Features::BlockFeatureBuilder *blockFeatureBuilder1;
blockFeatureBuilder1 = workPart->Features()->CreateBlockFeatureBuilder(nullFeatures_Feature);
blockFeatureBuilder1->BooleanOption()->SetTargetBodies(targetBodies1);
Point3d originPoint1(0.0, 0.0, 0.0);
blockFeatureBuilder1->SetOriginAndLengths(originPoint1, "100", "100", "100");
blockFeatureBuilder1->SetBooleanOperationAndTarget(Features::Feature::BooleanTypeCreate, nullBody);
Features::Feature *feature1;
feature1 = blockFeatureBuilder1->CommitFeature();
blockFeatureBuilder1->Destroy();

// Второй блок
Features::BlockFeatureBuilder *b12;
Point3d originPoint2(0, 50, 100);
b12 = workPart->Features()->CreateBlockFeatureBuilder(nullFeatures_Feature);
b12->SetOriginAndLengths(originPoint2, "50", "50", "50");
Features::Feature *feature2;
feature2 = b12->CommitFeature();
b12->Destroy();

// Третий блок
Features::BlockFeatureBuilder *b13;
Point3d originPoint3(0, 0, 0);
b13 = workPart->Features()->CreateBlockFeatureBuilder(nullFeatures_Feature);
b13->SetOriginAndLengths(originPoint3, "50", "50", "50");
std::vector<Body *> targetBodies11(1);
targetBodies11[0] = nullBody;
b13->BooleanOption()->SetTargetBodies(targetBodies11);
b13->BooleanOption()->SetType(GeometricUtilities::BooleanTypeSubtract);
Features::Feature *feature3;
feature3 = b13->CommitFeature();
b13->Destroy();
```

Рисунок 5 - Построение блоков в программе NX, используя NXOPEN.

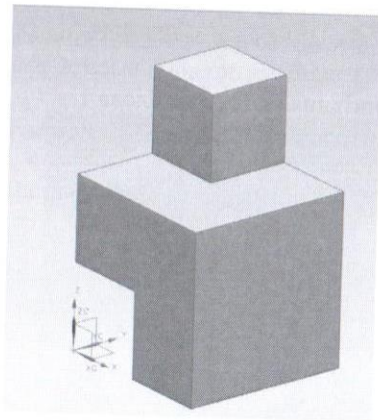


Рисунок 6 - Результат построения с использованием NXOPEN.

Как видно из рисунков 4 и 6, результат в первом и втором случае один и тот же. Конечно, приведенный пример построения блоков является наипростейшим, но и в этом случае видно, что количество действий в обоих методах несоизмеримо, построении в NX проходит в 6 операций, при использовании модуля NXOPEN это умещается в сравнительно

небольшой программе. К тому же построение программы это разовая операция, в то время как построение каждой новой модели непосредственно в программе NX, потребует повторения этих операций, что для более сложных моделей является существенным минусом.

Подводя итог нужно отметить, что автоматизация проектировочных процессов является оптимальным решением поставленных выше задач. Упростив и унифицировав процесс проектирования, мы одновременно увеличиваем производительность и исключаем ошибки, которые может допустить конструктор ввиду невнимательности, и, конечно, следует обратить внимание, что такой метод, при всей кажущейся простоте, потребует от конструктора высокого уровня профессиональной подготовки.

Применение автоматизированного проектирования и умение работать с программными продуктами требуется каждому инженеру. Предприятия, работающие без применения САПР или лишь с их частичным использованием, вполне могут оказаться неконкурентоспособными.

#### Список использованной литературы:

- 1 Норенков И.П. Автоматизированное проектирование: учеб. пособие по информатике для студентов технического университета. М.: МГТУ им. Н. Э. Баумана, 2000. - 188с.
- 2 Ахатов Р. Х., Русов Б. В. Проектирование изделия с помощью его параметрической информационной модели, построенной на основе экспертных знаний // Известия Самарского научного центра Российской академии наук - Самара: Самарский научный центр Российской академии наук, Том 13(39) №4 (4), 2011. С. 1018-1022.

**АВИАМАШИНОСТРОЕНИЕ  
И ТРАНСПОРТ СИБИРИ**

Сборник статей  
III Всероссийской научно-практической конференции

**(Иркутск, 11–12 апреля, 2013 г.)**

Печатается с оригиналов, подготовленных авторами

Подписано в печать 05.04.2012. Формат 60 x 90 / 16.  
Бумага офсетная. Печать цифровая. Усл. печ. л. 28,0.  
Тираж 100 экз. Зак. 13с.

Лицензия ИД № 06506 от 26.12.2001  
Иркутский государственный технический университет  
664074, г. Иркутск, ул. Лермонтова, 83