IEDDE «CifCAD»

ИСПЭИ (Интегрированная Среда Проектирования Электрических Изделий)  
IEDDE (Integrated Electrical Devices Design Environment)

Contents

[Методика Сбора Требований 2](#_Toc114192096)

[Рабочий Процесс 2](#_Toc114192097)

[«Фиктивные Заказчики» - Тестовые Проекты 2](#_Toc114192098)

[Проект №1: Подсистема питания печатной платы с системного до схемного уровня 3](#_Toc114192099)

[Проект №:2 Простой проект платы от структурной схемы до CAM-файлов 3](#_Toc114192100)

[Проект №3: Многоплатное устройство в корпусе 3](#_Toc114192101)

[Проект №4: Многоуровневая система вплоть до принципиальных схем 3](#_Toc114192102)

[Проект №5: Одноплатное устройство в корпусе с DFMEA от требований до сборочной документации 3](#_Toc114192103)

[Проект №6: Электрика в квартире 3](#_Toc114192104)

# Методика Сбора Требований

## Рабочий Процесс

Методика сбора требований основана на следующем воркфлоу разработки:

1. Описание идеи
2. Формулирование требований
   1. Сбор требований
   2. Анализ требований
3. Проектирование
   1. Синтез структуры
   2. Выбор решений
   3. Выбор компонентов
4. Разработка
   1. Разработка принципиальной схемы
   2. Разработка геометрии и топологии
5. Подготовка производства
   1. Разработка документации для изготовления деталей
   2. Разработка документации для сборки
   3. Разработка документации для функционального тестирования и настройки
6. Отладка
7. Внедрение
   1. Разработка документации для смежных систем
   2. Разработка пользовательской документации

## «Фиктивные Заказчики» - Тестовые Проекты

Собираем требования, делая вид, как будто у нас есть несколько заказчиков, каждый их которых хочет нашу систему для своего проекта, и у каждого из проектов свои особенности.

Прогнать по всем релевантным этапам несколько типов проектов:

* Подсистема питания печатной платы с системного до схемного уровня
* Простой проект платы от структурной схемы до CAM-файлов
* Многоплатное устройство в корпусе
* Многоуровневая система вплоть до принципиальных схем
* Одноплатное устройство в корпусе с DFMEA от требований до сборочной документации
* Электрика в квартире

## План Сбора Требований

* Сначала пройтись по [Описанию Замысла](Описание%20Замысла.docx) и формализовать всё, что там содержится.
* Затем прогнать все тестовые проекты
* Все возникающие требования относить к одной из шести категорий:
  + Идеологические
  + Общесистемные
  + Требования к текстовому редактору
  + Требования к редактору схем
  + Требования к топологическому векторному редактору
  + Требования к системе трёхмерного моделирования

## Термины и Определения

**Продукт** – так везде в этом документе называется разрабатываемая IEDDE CifCAD.

**Инструмент** – доступная пользователю часть **продукта**, использующаяся для решения конкретного набора задач.

**Примитив** – мельчайший мыслимый как единое целое элемент данных, создаваемых с помощью **инструментов продукта**.

# Требования из Общих Соображений

Источник – [Описание Замысла](Описание%20Замысла.docx)

|  |  |  |  |
| --- | --- | --- | --- |
| **Ключ** | **Категория** | **Требование** | **Источник** |
| **HL.CS-001** | Общесистемные | Продукт должен быть предустанавливаемым приложением. | Идея и Цель |
| **HL.CS-002** | Общесистемные | Продукт должен запускаться на ОС Windows версии 10 и выше. | Идея и Цель;  Наброски Требований и Общие Принципы: Системные Требования |
| **HL.CS-003** | Общесистемные | Продукт должен предоставлять инструментарий для проектирования и разработки электрической части изделий на уровне печатной платы и выше. | Идея и Цель |
| **HL.CS-004** | Общесистемные | Продукт должен предоставлять инструментарий для создания трёхмерных моделей простых механических деталей, трёхмерной визуализации электрических частей изделия и полных электромеханических сборок. | Идея и Цель |
| **HL.CS-005** | Общесистемные | Продукт должен предоставлять инструментарий для написания различных текстовых документов, рисования схем и осуществления автоматизированных математических вычислений (на элементарных функций). | Идея и Цель |
| **HL.CS-006** | Общесистемные | В архитектуре продукта должна быть сущность «**Проект** (**Project**)» - набор файлов, создающийся при разработке одного конкретного изделия. Продукт должен предоставлять возможность создания, редактирования, сохранения и чтения проектов. Пользователь продукта должен иметь возможность манипулировать проектами (вместе со всеми его данными) как едиными атомарными сущностями. | Идея и Цель |
| **HL.CS-007** | Общесистемные | Продукт должен предоставлять возможность сделать каждый **атомарный** **элемент** проектных данных интерактивным объектом:   * **Элемент** должен **создаваться** и **определяться** в одном конкретном месте одного конкретного **проектного** **файла**; * Должна быть возможность **использовать** данные **элемента** в любом другом **файле**; * При **изменении** значения **элемента** во всех подстановках данных этого **элемента** во всех **файлах проекта** использованные значения должны автоматически **обновиться**. | Идея и Цель |
| **HL.CS-008** | Общесистемные | **Проект** должен быть древовидным – каждая составная часть проектируемого изделия должна быть отдельной сущностью в архитектуре продукта. | Идея и Цель |
| **HL.CS-009** | Общесистемные | В продукт должны быть встроены следующие инструменты математического моделирования и CAE:   * Моделирование электрических цепей; * Моделирование плотности токов на печатной плате; * Моделирование тепловых процессов в электрических изделиях – на проводах, проводниках платы и компонентах с учётом современных методов теплоотвода и охлаждения; * Расчёт паразитных параметров конструкций приборов (проводов, элементов топологии печатных плат). | Идея и Цель |
| **HL.CS-010** | Редактор схем | В продукте должен быть инструмент «**Редактор схем** (**Diagram Editor**)». Его назначение – создание, редактирование, сохранение и просмотр схем, таких как:   * Различные иерархические и организационные диаграммы; * Графы общего вида; * Принципиальные электрические схемы; * Функциональные и структурные схемы, блок-схемы; * Схемы соединений и подключений, в том числе электрические. | Наброски Требований и Общие Принципы:  Универсальность Инструментария |
| **HL.CS-011** | Редактор схем | **Редактор схем** ad hoc должен использоваться также для создания УГО электрических компонентов. | Наброски Требований и Общие Принципы:  Вложенность Проектов |
| **HL.CS-012** | Общесистемные | Любое разрабатываемое в продукте электрическое изделие (в частности, электрическое устройство, печатная плата или электрический компонент) должно представлять собой отдельный **проект**, который можно подключить в другой **надпроект** изделия, использующее данный **проект** как составную часть. | Наброски Требований и Общие Принципы:  Вложенность Проектов |
| **HL.CS-013** | Общесистемные | В продукте не должно быть неизменяемых ограничений на то, какого типа **надпроекты** и **подпроекты** могут сопрягаться (иными словами, например, ограничений на то, чтобы использовать в проекте сложного распределённого устройства отдельную микросхему без её включения в состав **проекта** печатной платы, устройства, модуля стойки и т. д.). | Наброски Требований и Общие Принципы:  Вложенность Проектов |
| **HL.CS-014** | Общесистемные | Каждый **проект** должен иметь возможность включать в себя данные и документы разного типа, разного назначения и относящиеся к разным стадиям проектирования и разработки по желанию пользователя. | Наброски Требований и Общие Принципы:  Вложенность Проектов |
| **HL.CS-015** | Общесистемные | При подключении **подпроекта** в **проект** должны добавляться данные всех аспектов **подпроекта** (данные и документы разного типа, разного назначения и относящиеся к разным стадиям проектирования и разработки, см. требование **HL.CS-014**) по желанию пользователя. | Наброски Требований и Общие Принципы:  Вложенность Проектов |
| **HL.CS-016** | Общесистемные | Продукт должен предоставлять возможность создавать, редактировать, сохранять и читать **проекты** с любой степенью вложенности **подпроектов**. | Наброски Требований и Общие Принципы:  Вложенность Проектов |
| **HL.CS-017** | Общесистемные | Продукт должен предоставлять возможность удобным для пользователя образом создавать, изменять и переиспользовать **параметры**, **требования**, **термины**, **определения**, **отрывки** и **обозначения** в любом файле проекта. Эта возможность должная использовать реализацию требования **HL.CS-007**. | Наброски Требований и Общие Принципы: Параметрическое Конструирование и Когерентность |
| **HL.CS-018** | Текстовый редактор | В продукте должен быть инструмент «**Текстовый редактор** (**Text Editor**)». Его назначение – создание, редактирование, сохранение и просмотр документов, содержащих форматированные текст и записи вычислений. | Наброски Требований и Общие Принципы: Параметрическое Конструирование и Когерентность |
| **HL.CS-019** | Общесистемные | Продукт должен иметь механизм, позволяющий пользователю удобным образом по любому **использованию** интерактивного **атомарного элемента** (реализация требования **HL.CS-007**) перейти (сфокусировать экран) к **созданию** и **определению** этого **элемента.** | Наброски Требований и Общие Принципы: Параметрическое Конструирование и Когерентность |
| **HL.CS-020** | Общесистемные | Каждыйинтерактивный **атомарный элемент** (реализация требования **HL.CS-007**) должен иметь уникальный идентификатор («**имя**» («**name**»)), определяемый пользователем. | Наброски Требований и Общие Принципы: Параметрическое Конструирование и Когерентность |
| **HL.CS-021** | Общесистемные | Продукт должен иметь механизм, позволяющий пользователю удобным образом по любому **использованию** интерактивного **атомарного элемента** (реализация требования **HL.CS-007**) изменить **имя** этого **элемента.** | Наброски Требований и Общие Принципы: Параметрическое Конструирование и Когерентность |
| **HL.CS-022** | Текстовый редактор | **Текстовый редактор** должен иметь примитив «**Интерактивное вычисление**», запись вычисления, результат которого вычисляется автоматически и является **атомарным элементом**. При просмотре **интерактивное вычисление** должно отображаться в удобочитаемом виде, принятом в научной и технической литературе. | Наброски Требований и Общие Принципы: Параметрическое Конструирование и Когерентность |
| **HL.CS-023** | Общесистемные | Продукт должен предоставлять пользователю возможность менять:   * Содержимое; * Отображаемый вид; * Набор вновь **создаваемых** или **определяемых атомарных элементов**;   в любом файле **проекта** в зависимости от значения какого-либо **атомарного элемента** в любом из файлов **проекта**. | Наброски Требований и Общие Принципы: Параметрическое Конструирование и Когерентность |
| **HL.CS-024** | Текстовый редактор | Продукт должен предоставлять возможность использовать в **интерактивном вычислении** числовое значение **параметра** или **требования**, которые являются реализацией требования **HL.CS-017**. | Наброски Требований и Общие Принципы: Параметрическое Конструирование и Когерентность |
| **HL.CS-025** | Редактор схем | **Редактор схем** должен иметь примитив «**Соединение**» - звездообразная ломаная линия, предназначенный, в том числе, для представления одной эквипотенциальной точки в электрических цепях. | Наброски Требований и Общие Принципы: Параметрическое Конструирование и Когерентность |
| **HL.CS-026** | Общесистемные | Продукт должен предоставлять пользователю возможность удобным образом создавать **классы** – группы **атомарных элементов** одного типа, имеющие заданное пользователем **имя**. | Наброски Требований и Общие Принципы: Параметрическое Конструирование и Когерентность |

|  |  |  |  |
| --- | --- | --- | --- |
| **HL.CS-027** | Редактор схем | **Редактор схем** должен иметь специальный примитив «**Метка класса**», который пользователь может располагать на отрисованных **соединениях**. **Метка класса** должна иметь параметр «**Имя**», в котором можно по ссылке использовать данные **атомарного элемента**. Должна быть возможность отобразить или скрыть **имя метки класса**. Также **редактор схем** должен предоставлять пользователю возможность автоматически сгенерировать **классы** цепей (эквипотенциальных точек), которые включают в себя все цепи, отмеченные **метками** с одинаковым **именем** (по всему **проекту**, по **подпроектам**, по документу или по части документа), и задать этому **классу** такое же, как у **метки имя**. | Наброски Требований и Общие Принципы: Параметрическое Конструирование и Когерентность |
| **HL.CS-028** | Редактор схем | Должна быть возможность использовать в качестве имени **метки класса** данные **атомарных элементов** любого типа. | Наброски Требований и Общие Принципы: Параметрическое Конструирование и Когерентность |
| **HL.CS-029** | Топологический векторный редактор | В продукте должен быть инструмент «**Топологический векторный редактор** (**Topological Vector Editor**)». Его назначение – создание, редактирование, сохранение и просмотр векторных изображений с множеством **слоёв**. Интерфейс **векторного редактора** должен быть наиболее удобен для создания чертежей, файлов представления топологии печатных плат. | Наброски Требований и Общие Принципы: Параметрическое Конструирование и Когерентность |
| **HL.CS-030** | Топологический векторный редактор | Для **векторного редактора** (по всему **проекту**, по **подпроектам**, по документу или по части документа) должна быть возможность задать **правила топологии** – утверждения, состоящие из выражений от параметров примитивов в документах (главным образом в файлах топологии печатных плат) **проекта** и данных **атомарных элементах** на арифметических и логических операторах и операторов сравнения. | Наброски Требований и Общие Принципы: Параметрическое Конструирование и Когерентность |
| **HL.CS-031** | Общесистемные | В рамках каждого **проекта** должна быть возможность создать сущность «**Проектное решение**». Пользователю должен быть предоставлен удобный способ выбрать одну из нескольких заданных им альтернатив и в любой момент поменять выбор. От выбранного пользователем **решения** должно зависеть отображение части файлов **проекта**, части содержимого **проектных** файлов и значения части **атомарных элементов**. **Проектное решение** является частью реализации требования **HL.CS-023**. | Наброски Требований и Общие Принципы: Интерактивность |
| **HL.CS-032** | Общесистемные | В продукте везде, где это возможно, должна быть реализована **автоподстановка** – продукт должен предлагать пользователю добавить в подходящее место в документе данные определённого типа всегда, когда продукт способен предсказать необходимость добавления таких данных | Наброски Требований и Общие Принципы: Интерактивность |
| **HL.CS-033** | Текстовый редактор | В текстовом редакторе должна быть реализована **автоподстановка** значений **определённых** в **проекте** **атомарных элементов** или их полей, если они имеют текстовый тип. Такая **автоподстановка** является частью реализации требования **HL.CS-032**. | Наброски Требований и Общие Принципы: Интерактивность |
| **HL.CS-034** | Общесистемные | В продукте везде, где это возможно, должны быть реализованы инструменты класса «**Генератор**». Такого рода инструменты позволяют пользователю по одному простому действию, совершённому в одном документе, автоматически сгенерировать как можно более полные, наполненные контекстной информацией, и связанные данные (содержимое, примитивы) в другом документе. | Наброски Требований и Общие Принципы: Интерактивность |
| **HL.CS-035** | Общесистемные | Продукт должен предоставлять пользователю возможность кастомизировать интерфейс, позволяя сделать любую пользовательскую функцию более быстродоступной, чем по умолчанию. | Наброски Требований и Общие Принципы: Настраиваемость и Расширяемость |
| **HL.CS-036** | Общесистемные | Продукт должен предоставлять пользователю возможность создать собственные программные инструменты для работы с **проектными** данными и интегрировать их в продукт так, чтобы с точки зрения UX/UI они не отличались от нативных инструментов. | Наброски Требований и Общие Принципы: Настраиваемость и Расширяемость |
| **HL.CS-087** | Идеологические | Продукт должен содержать **валидаторы** – инструменты, запускаемые вручную или автоматически, которые позволяют оценить валидность **проектных** данных, когерентность данных между разными частями проекта и между проектом и централизованным хранилищем библиотечных данных и выполнение требований. | Наброски Требований и Общие Принципы: Непрерывная Валидация |
| **HL.CS-088** | Идеологические | Принцип максимизации ошибки: в случае, если **валидатор** даёт отрицательно оценивает качество **проекта**, продукт должен максимально заметным образом сигнализировать об этом пользователям. | Наброски Требований и Общие Принципы: Непрерывная Валидация |
| **HL.CS-099** | Идеологические | В продукте должна быть разновидность **валидаторов** –«**Верификаторы**», которые проверяют соответствие данных **проекта** совокупности определённых пользователем правил. | Наброски Требований и Общие Принципы: Непрерывная Валидация |
| **HL.CS-100** | Идеологические | В продукте должна быть разновидность **валидаторов** –«**Синхронизаторы**», которые проверяют когерентность данных между частями распределённого хранилища данных – разными частями **проекта**, и между локальными данными и централизованным хранилищем. | Наброски Требований и Общие Принципы: Непрерывная Валидация |
| **HL.CS-037** | Общесистемные | Продукт должен предлагать пользователю готовый набор кастомизаций пользовательского интерфейса, наиболее подходящий для каждого из распространённых типов задач в индустрии проектирования электрических изделий. | Наброски Требований и Общие Принципы: Встроенная Кастомизация |
| **HL.CS-038** | Идеологические | В разработке UX/UI необходимо придерживаться принципа интуитивной понятности. | Наброски Требований и Общие Принципы: Интуитивная Понятность |
| **HL.CS-039** | Общесистемные | **ВТОРОСТЕПЕННОЕ ТРЕБОВАНИЕ:**  Продукт должен запускаться на ОС семейства Linux. | Наброски Требований и Общие Принципы: Системные Требования |
| **HL.CS-040** | Общесистемные | **ВТОРОСТЕПЕННОЕ ТРЕБОВАНИЕ:**  Продукт должен предоставлять инструменты для отслеживания ситуации на различных рынках (как минимум, электронных компонентов и расходных материалов) в реальном времени. | Наброски Требований и Общие Принципы: Второстепенные Аспекты Функциональности |
| **HL.CS-041** | Общесистемные | **ВТОРОСТЕПЕННОЕ ТРЕБОВАНИЕ:**  Продукт должен предоставлять полный функционал современной распределённой **системы контроля версий** для **проектов**. | Наброски Требований и Общие Принципы: Второстепенные Аспекты Функциональности |
| **HL.CS-042** | Общесистемные | **ВТОРОСТЕПЕННОЕ ТРЕБОВАНИЕ:**  Продукт должен предоставлять полный функционал современной **PDM** системы для **проектов**. | Наброски Требований и Общие Принципы: Второстепенные Аспекты Функциональности |
| **HL.CS-043** | Общесистемные | **ВТОРОСТЕПЕННОЕ ТРЕБОВАНИЕ:**  Продукт должен предоставлять удобный функционал для командной работы. | Наброски Требований и Общие Принципы: Второстепенные Аспекты Функциональности |
| **HL.CS-044** | Общесистемные | Продукт должен предоставлять пользователю возможность хранить все данные одного **проекта централизованно** – на одном физическом носителе в одном каталоге. | Наброски Требований и Общие Принципы: Второстепенные Аспекты Функциональности |
| **HL.CS-045** | Общесистемные | Продукт должен предоставлять пользователю возможность **зафиксировать** в **централизованном хранилище проектных** данных (уникальных для данного **проекта**) текущую версию проектных данных (**ревизию**). **Централизованное** хранилище должно сохранять все **ревизии проекта** (если пользователь явно не настроил **хранилище** иным образом) и удобным образом переключаться между ними (загружать в рабочее пространство продукта). | Наброски Требований и Общие Принципы: Второстепенные Аспекты Функциональности |
| **HL.CS-046** | Общесистемные | Каждый **подпроект**, включённый в **проект**, не должен передавать свои данные в **ревизии** **проекта**, так как является отдельно хранимой сущностью. Продукт должен затруднять внесение изменений в **подпроекты** при работе с **проектом**, сигнализируя пользователю, что он правит другую хранимую сущность. | Наброски Требований и Общие Принципы: Второстепенные Аспекты Функциональности |
| **HL.CS-047** | Общесистемные | Продукт должен предоставлять пользователям настраивать **права доступа** к каждому файлу **проектных** данных. Должна быть возможность предоставить **доступ** к каждому отдельному файлу каждому отдельному пользователю только на **чтение**, на **чтение и запись**, и вообще **лишить** пользователя **доступа** к файлу. | Наброски Требований и Общие Принципы: Второстепенные Аспекты Функциональности |
| **HL.CS-048** | Общесистемные | Проект должен иметь сущность «**Переменная**», реализующую требование **HL.CS-007**. **Переменная** – это любой набор данных определённого **типа**, имеющий заданный **идентификатор** определённого **типа**. **Переменные** могут **создаваться** и **определяться** пользователем в любом файле **проекта**. | Отдельные Моменты и Фичи: Переменные |
| **HL.CS-049** | Общесистемные | Любой набор **проектных** данных должен являться **переменной**. | Отдельные Моменты и Фичи: Переменные |
| **HL.CS-050** | Общесистемные | Формат **идентификатора** и **тип** поля данных **переменной** должен определять пользователь. Должны быть доступны следующие **типы** данных:   * Число * Текст * Число с единицей измерения (единицы измерения и префиксы-множители должны быть предопределены в продукте) * Ссылка на **переменную** * Упорядоченная комбинация этих **типов**   **Тип** и содержимое поля данных должны определяться пользователем. | Отдельные Моменты и Фичи: Переменные |
| **HL.CS-051** | Общесистемные | Любой **проект** не должен содержать **переменные** с одинаковыми **идентификаторами**. | Отдельные Моменты и Фичи: Переменные |
| **HL.CS-052** | Общесистемные | Продукт должен предоставлять пользователю возможность **объявить** **переменную** (зарезервировать для неё **идентификатор**), **инициализировать переменную** (задать поле данных) и **определить переменную** (одновременно **объявить** и **инициализировать**)в любом месте **проекта** (исключая включённые существовавшие ранее **подпроекты**). | Отдельные Моменты и Фичи: Переменные |
| **HL.CS-053** | Общесистемные | Продукт должен предоставлять пользователю возможность использовать данные любой **переменной проекта** в любом файле **проекта** какое угодно количество раз. | Отдельные Моменты и Фичи: Переменные |
| **HL.CS-054** | Система трёхмерного моделирования | В продукте должен быть инструмент «**Система трёхмерного моделирования** (**3-Dimensional Modeling System**)». Его назначение – создание, редактирование, сохранение и просмотр 3-мерных моделей твёрдых тел с, опционально, гнущимися элементами, а также гибких продолговатых тел (проводов). Интерфейс и возможности **системы трёхмерного моделирования** не обязаны полностью соответствовать полнофункциональному современному MCAD. **Система** должна быть заточена на задачи проектирования электрических изделий, в которой твёрдые тела скорее просматриваются, а не проектируются. | Отдельные Моменты и Фичи: Подсистемы |
| **HL.CS-055** | Общесистемные | В продукте должна быть реализована «**Система компьютерной алгебры** (**Computer Algebra System**)» - программный компонент, позволяющий производить математические вычисления и визуализировать их результаты. Набор доступных типов вычислений должен быть достаточным для решения задач проектирования и разработки электрических изделий. | Отдельные Моменты и Фичи: Подсистемы |
| **HL.CS-056** | Общесистемные | В продукте должна быть реализована «**Система моделирования динамических систем** (**Dynamic System Simulation System**)» - программный компонент, позволяющий производить численное решение систем обыкновенных дифференциальных уравнений, заданных в виде блок-схем, а также визуализировать результаты. Набор доступных элементов блок-схем должен быть достаточным для задач проектирования и разработки электрических изделий. | Отдельные Моменты и Фичи: Подсистемы |
| **HL.CS-057** | Общесистемные | В продукте должна быть реализована «**Система конечноэлементного моделирования физических полей** (**Physical Fields Finite Element Simulation System**)» - программный компонент, позволяющий производить численное решение краевой задачи для трёхмерных и двухмерных скалярных и векторных полей, представляющих релевантные для задач проектирования электрических изделий физические поля (электромагнитное, тепловое поля, поля токов и механических напряжений), а также визуализировать эти решения. | Отдельные Моменты и Фичи: Подсистемы |
| **HL.CS-058** | Общесистемные | В продукте должны быть сущности типа «**Редактор**» - объединения инструментов, имеющие каждый свою отдельную рабочую область и позволяющие пользователю создавать, редактировать, сохранять и просматривать каждый свои типы документов. Доступ к функциям разных **редакторов** должен быть организован на основе единообразного UX/UI. **Текстовый редактор**, **редактор схем**, **топологический векторный редактор** и **система трёхмерного моделирования** должны быть **редакторами**. | Отдельные Моменты и Фичи: Подсистемы |
| **HL.CS-059** | Общесистемные | Каждый **редактор** должен уметь выполнять автоматизированные вычисления, являющиеся реализацией требования **HL.CS-005**. | Отдельные Моменты и Фичи: Редакторы |
| **HL.CS-060** | Общесистемные | Каждый **редактор** должен предоставлять пользователю возможность создать себе **профиль** – определённую настройку пользовательского интерфейса, в которой ряд функций, возможностей и инструментов **редактора** сделана более быстродоступной. **Профиль** реализует требование **HL.CS-035**. | Отдельные Моменты и Фичи: Редакторы |
| **HL.CS-061** | Текстовый редактор | **Текстовый редактор** должен предоставлять пользователю возможность использовать следующие варианты разбиения и ограничения рабочей области (физического размера документа):   * Ограничение сверху и слева (как в системах наподобие Confluence) * Ограничение сверху и справа (как в системах наподобие Confluence для арабо-израильских локализаций) * Ограничение по бокам (бесконечная лента) * Разбивка по всем 4-м сторонам (печатные листы) | Отдельные Моменты и Фичи: Редакторы |
| **HL.CS-062** | Редактор схем | **Редактор схем** должен предоставлять пользователю возможность использовать следующие варианты разбиения и ограничения рабочей области (физического размера документа):   * Разбивка по всем 4-м сторонам (печатные листы) | Отдельные Моменты и Фичи: Редакторы |
| **HL.CS-063** | Топологический векторный редактор | **Топологический векторный редактор** должен предоставлять пользователю возможность использовать следующие варианты разбиения и ограничения рабочей области (физического размера документа):   * Разбивка по всем 4-м сторонам (печатные листы) | Отдельные Моменты и Фичи: Редакторы |
| **HL.CS-064** | Общесистемные | В продукте должны быть следующие типы («**общие виды**») пользовательского интерфейса:   * **Панели инструментов общего вида** * **Скриптовый язык запросов** (**ВТОРОСТЕПЕННОЕ ТРЕБОВАНИЕ**) * **Низкоуровневое API** | Отдельные Моменты и Фичи: Пользовательский Интерфейс |
| **HL.CS-065** | Общесистемные | В продукте должны быть следующие типы («**быстрого доступа**») пользовательского интерфейса:   * **Панели инструментов быстрого доступа** * **Контекстные меню** * **Горячие клавиши** | Отдельные Моменты и Фичи: Пользовательский Интерфейс |
| **HL.CS-066** | Общесистемные | Продукт должен предоставлять пользователю перенести каждый из элементов пользовательского интерфейса **общего вида** в любую категорию пользовательского интерфейса **быстрого доступа**. Этот механизм должен использоваться для создания **профилей** **редакторов**. | Отдельные Моменты и Фичи: Пользовательский Интерфейс |
| **HL.CS-067** | Текстовый редактор | **Текстовый редактор** должен иметь следующие предустановленные **профили**:   * **Текстовый документ** – для написания и вёрстки текстов * **Электронная таблица** – для создания интерактивных таблиц (наподобие Excel) * **Интерактивный файл расчётов** – для написания сложных расчётов в удобочитаемом виде и представления их результатов в визуальной форме | Отдельные Моменты и Фичи: Предопределённые Профили Редакторов |
| **HL.CS-068** | Редактор схем | **Редактор схем** должен иметь следующие предустановленные **профили**:   * **Граф** – минималистичный набор инструментов для рисования диаграмм в форме графов * **Эскиз** – для создания эскизов пользовательского интерфейса приборов и простой векторной графики не в масштабе * **Структурная схема** – для рисования блок-схем, Э1, и в целом схем типа 1 * **Функциональная схема** – для рисования схем Э2 и в целом схем типа 2, более подробных, чем блок-схема, но менее детализированных, чем принципиальные * **Принципиальная схема** – для рисования полностью детализированных принципиальных схем Э3 и в целом схем типа 3 * **Схема соединений** – для рисования схем Э4 и в целом схем типа 4 * **Схема жгута** – для рисования схем жгутов и кабелей с распайками разъёмов и указанием параметров проводов * **Условно-графическое обозначение** – для рисования УГО | Отдельные Моменты и Фичи: Предопределённые Профили Редакторов |
| **HL.CS-069** | Топологический векторный редактор | **Топологический векторный редактор** должен иметь следующие предустановленные **профили**:   * **Рисунок** – для простой векторной графики в масштабе * **Печатная плата** – для проектирования печатных плат * **Посадочное место** – для проектирования посадочных мест для электронных компонентов * (**ВТОРОСТЕПЕННОЕ ТРЕБОВАНИЕ**) **Геометрическое представление жгута** – для проектирования элементов жгутов, для которых важно точное геометрическое расположение * **Чертёж** – для разработки технических чертежей, в том числе в масштабе | Отдельные Моменты и Фичи: Предопределённые Профили Редакторов |
| **HL.CS-070** | Система трёхмерного моделирования | **Система трёхмерного моделирования** должна иметь следующие предустановленные **профили**:   * **Сборка** – для отображения и взаимного перемещения множества тел * **Трассировка проводов** – для прокладывания трасс гибких продолговатых тел по поверхностям твёрдых тел (трассировки проводов) * **Деталь** – для разработки 3D-моделей твёрдых тел, в том числе с гнущимися элементами | Отдельные Моменты и Фичи: Предопределённые Профили Редакторов |
| **HL.CS-071** | Текстовый редактор | **Текстовый редактор** должен иметь примитив «**Текстовое поле**» - кусок форматированного текста определённого размера, который можно перемещать по документу и перемешивать с себе подобными, а также другими примитивами **текстового редактора**. **Текстовое поле** может содержать математические формулы в удобочитаемом виде, принятом в научной и технической литературе. | Отдельные Моменты и Фичи: Базовые Примитивы |
| **HL.CS-072** | Текстовый редактор | **Текстовый редактор** должен иметь примитив «**Блок вычисления**» - форматированная запись математического и / или логического выражения с **оператором сравнения** или **присваивания**, имеющая определённый размер, которую можно перемещать по документу и перемешивать с себе подобными, а также другими примитивами **текстового редактора**. Формула должна иметь удобочитаемый виде, принятом в научной и технической литературе. Формула должна вычисляться автоматически. Результат присваивания по вычисленному значению выражения должен сохраняться в **переменную**. Результат операции **сравнения** должен быть **переменной** булевого типа. | Отдельные Моменты и Фичи: Базовые Примитивы |

|  |  |  |  |
| --- | --- | --- | --- |
| **HL.CS-073** | Текстовый редактор | **Текстовый редактор** должен иметь примитив «**Блок выбора**» - набор контроллов одинакового вида, в котором может быть **установлен** только один из них. **Блок выбора** должен предоставлять пользователю возможность ссылаться на все свои данные – число контроллов, их подписи, названия и то, какой из них **установлен**, а также **установлен** ли хотя бы один. **Установка** контроллов должна осуществляться пользователем. Данные об **установке** контроллов должны являться данными редактируемого файла – изменение в статусе **установки** должно считаться отслеживаемым изменением документа. Блок выбора является реализацией требования **HL.CS-023**. | Отдельные Моменты и Фичи: Базовые Примитивы |
| **HL.CS-074** | Текстовый редактор | **Текстовый редактор** должен иметь примитив «**Электронная таблица**» - двухмерная таблица с возможностью размещения в каждой её **ячейке** **текстового поля**, **блока вычисления** с возможностью использовать данные из этой **таблицы**, а в целом ряду или столбце – блока выбора с сопоставлением каждой **ячейке** своего контролла (не более одного **блока выбора** на **таблицу**). | Отдельные Моменты и Фичи: Базовые Примитивы |
| **HL.CS-075** | Текстовый редактор | **Текстовый редактор** должен иметь примитив «**Рисунок**» - изображение определённого размера, который можно перемещать по документу и перемешивать с себе подобными, а также другими примитивами **текстового редактора**. **Рисунок** может быть автоматически генерируемым представлением файлов, разрабатываемых в других **редакторах** продукта. | Отдельные Моменты и Фичи: Базовые Примитивы |
| **HL.CS-076** | Редактор схем | **Редактор схем** должен иметь примитив «**Узел**» - изображение, созданное из традиционных примитивов векторной графики, имеющее определяемое пользователем положение в координатной сетке файла. Должна быть возможность добавить **узлу** форматированную подпись и зафиксировать её положение относительно **узла**. | Отдельные Моменты и Фичи: Базовые Примитивы |
| **HL.CS-077** | Редактор схем | **Редактор схем** должен иметь примитив «**Точка подключения**» - точка, принадлежащая **узлу**. Должна быть возможность создать для **точки подключения** пиктограмму, созданную из традиционных примитивов векторной графики и зафиксировать её положение относительно **точки подключения**. Должна быть возможность добавить **точке подключения** форматированную подпись и зафиксировать её положение относительно **точки подключения**. Должна быть возможность зафиксировать положение **точки подключения** относительно родительского **узла**. | Отдельные Моменты и Фичи: Базовые Примитивы |
| **HL.CS-078** | Редактор схем | **Редактор схем** должен иметь примитив «**Соединение**» - линия, возможно, ветвящаяся, созданная из традиционных примитивов векторной графики. **Соединение** предназначено для создания визуальной и логической связи между **точками** подключения. Должна быть возможность добавить **соединению** форматированную подпись и зафиксировать её положение относительно **соединения**. **Соединение** реализует требование **HL.CS-025**. | Отдельные Моменты и Фичи: Базовые Примитивы |
| **HL.CS-079** | Редактор схем | **Редактор схем** должен иметь вспомогательный примитив «**Шина**» - линия, возможно, ветвящаяся, созданная из традиционных примитивов векторной графики. **Шина** должна предоставлять возможность заменить собой несколько **соединений** – на одном участке схемы **соединение** может быть оборвано на **шине** и возобновлено отводом от этой **шины** на другом участке схемы. Должна быть возможность «подключить» к **шине** неограниченное количество **соединений**. Для всех **соединений**, использующих данную шину, должна быть возможность однозначно визуально определить, какие конкретно входящие в **шину** отрывки **соединений** относятся к данному **соединению**, а какие нет. | Отдельные Моменты и Фичи: Базовые Примитивы |
| **HL.CS-080** | Редактор схем | **Редактор схем** должен иметь вспомогательный примитив «**Порт**» - пиктограмма, созданная из традиционных примитивов векторной графики. **Порт** должен позволять визуально разорвать одно **соединение**, сохранив его в логической структуре схемы. **Порт** должен иметь точку, на которой можно завершить участок **соединения**. **Соединение** может иметь больше одной компоненты связности, если каждая из них имеет хотя бы одну ветвь, заканчивающуюся на **порту**. Для всех **соединений** в данном файле должна быть возможность однозначно визуально определить, какие конкретно группы **соединений**, использующих **порты**, на самом деле относятся к одному и тому же **соединению**, а какие нет. | Отдельные Моменты и Фичи: Базовые Примитивы |
| **HL.CS-081** | Топологический векторный редактор | **Топологический векторный редактор** должен иметь примитив «**Дорожка**» - прямая либо кривая линия определяемой пользователем постоянной толщины, имеющая определяемое пользователем положение в координатной сетке файла. | Отдельные Моменты и Фичи: Базовые Примитивы |
| **HL.CS-082** | Топологический векторный редактор | **Топологический векторный редактор** должен иметь примитив «**Полигон**» - плоская фигура с определяемой пользователем геометрией. Геометрия определяется как набор вершин, соединённых плоскими или кривыми линиями, являющимися сторонами фигуры. Положение **полигона** в координатной сетке файла должно определяться пользователем. | Отдельные Моменты и Фичи: Базовые Примитивы |
| **HL.CS-083** | Топологический векторный редактор | **Топологический векторный редактор** должен иметь примитив «**Надпись**» - кусок форматированного текста определённого размера, имеющий определяемое пользователем положение в координатной сетке файла. Должна быть возможность автоматически генерировать содержимое **надписи**, ссылаясь на содержимое **атомарных объектов проекта**. | Отдельные Моменты и Фичи: Базовые Примитивы |
| **HL.CS-084** | Система трёхмерного моделирования | **Система трёхмерного моделирования** должна иметь примитив «**Твёрдое тело**», служащее для представления трёхмерных объектов неизменяемой геометрии. Форма и положение в координатной сетке файла **твёрдого тела** определяются пользователем. | Отдельные Моменты и Фичи: Базовые Примитивы |
| **HL.CS-085** | Система трёхмерного моделирования | **Система трёхмерного моделирования** должна иметь примитив «**Упругое тело**», служащее для представления трёхмерных объектов, меняющих форму под воздействием внешней силы, и возвращающихся в исходное состояние при исчезновении этой силы. Изначальная форма, положение в координатной сетке файла и зоны допустимой деформации **упругого тела** определяются пользователем. | Отдельные Моменты и Фичи: Базовые Примитивы |
| **HL.CS-086** | Система трёхмерного моделирования | **Система трёхмерного моделирования** должна иметь примитив «**Тонкостенная пластина**», служащее для представления трёхмерных объектов, у которых одно измерение существенно меньше двух других. Грани **тонкостенной пластины** могут быть закреплены на **твёрдых телах**, и в таком случае могут под воздействием внешней силы изгибаться вокруг оси, параллельной этой грани и оставаться в изменённом состоянии при исчезновении этой силы. Изначальная форма, положение в координатной сетке файла и текущее деформированное состояние **тонкостенной пластины** определяются пользователем. | Отдельные Моменты и Фичи: Базовые Примитивы |
| **HL.CS-098** | Система трёхмерного моделирования | **Система трёхмерного моделирования** должна иметь примитив «**Нитевидное** **тело**», служащее для представления трёхмерных объектов, у которых одно из измерений существенно больше двух остальных. **Нитевидное тело** имеет постоянное поперечное сечение; под воздействием внешней силы изгибается в плоскости этого сечения и остаётся в изменённом состоянии при исчезновении этой силы. Изначальная форма, положение в координатной сетке файла и текущее деформированное состояние **нитевидного тела** определяются пользователем. | Отдельные Моменты и Фичи: Базовые Примитивы |
| **HL.CS-089** | Общесистемные | В продукте должен быть инструмент «**Навигатор**», который предоставляет пользователю наглядный и понятный вид структуры проекта, файлов проекта и позволяет открывать файлы. | Отдельные Моменты и Фичи: Срезы Проектов |
| **HL.CS-090** | Общесистемные | Продукт должен предоставлять пользователю возможность переключаться между **срезами проекта** – наборами **проектных** файлов определённого типа, соответствующими определённому этапу разработки. При включенности определённого **среза** не относящиеся к нему **проектные** документы должны быть скрыты в **навигаторе.** | Отдельные Моменты и Фичи: Срезы Проектов |
| **HL.CS-091** | Общесистемные | Должен быть **срез проекта** «**Замысел**», соответствующий пунктам 1–2 опорного рабочего процесса, и включающий в себя документы следующих типов:   * Спецификация проекта (**текстовый редактор**) * Входные требования (**текстовый редактор**) * Функциональные требования (**текстовый редактор**) * Технические требования (**текстовый редактор**) * Дерево трассировки требований (**редактор схем**) | Отдельные Моменты и Фичи: Срезы Проектов |
| **HL.CS-092** | Общесистемные | Должен быть **срез проекта** «**Проектный облик**», соответствующий пункту 3 опорного рабочего процесса, и включающий в себя документы следующих типов:   * Описание процедуры синтеза архитектуры (**текстовый редактор**) * Пошаговое описание процедуры принятия проектных решений (**текстовый редактор**) * Описание выбора решений (**текстовый редактор**) * Описание выбора компонентов (**текстовый редактор**) * Проверочный расчёт (**текстовый редактор**) * Структурная схема (**редактор схем**) * Функциональная схема (**редактор схем**) * Схема подсистемы (**редактор схем**) * Диаграмма состояний (**редактор схем**) * Эскиз интерфейса (**редактор схем**) | Отдельные Моменты и Фичи: Срезы Проектов |

|  |  |  |  |
| --- | --- | --- | --- |
| **HL.CS-093** | Общесистемные | Должен быть **срез проекта** «**Исходные данные**», соответствующий пункту 4 опорного рабочего процесса, и включающий в себя документы следующих типов:   * Схема соединений (**редактор схем**) * Принципиальная схема (**редактор схем**) * Файл платы (**топологический векторный редактор**) * Схема жгута или кабеля (**редактор схем**) * Файл геометрического представления жгута или кабеля (**топологический векторный редактор**) * Сборка (**система трёхмерного моделирования**) | Отдельные Моменты и Фичи: Срезы Проектов |
| **HL.CS-094** | Общесистемные | Должен быть **срез проекта** «**Подготовка производства: Изготовление**», соответствующий пункту 5a опорного рабочего процесса, и включающий в себя документы следующих типов:   * Чертёж-спецификация платы (**топологический векторный редактор**) * Файлы производства платы (**топологический векторный редактор**) * Чертёж мультизаготовки (**топологический векторный редактор**) * Файлы производства мультизаготовки (**топологический векторный редактор**) * Чертёж жгута или кабеля (**топологический векторный редактор**) * Перечень элементов и расходных материалов жгута или кабеля (**текстовый редактор**) | Отдельные Моменты и Фичи: Срезы Проектов |
| **HL.CS-095** | Общесистемные | Должен быть **срез проекта** «**Подготовка производства: Сборка**», соответствующий пунктам 5b-5c опорного рабочего процесса, и включающий в себя документы следующих типов:   * Перечень элементов платы (**текстовый редактор**) * Чертёж трафарета (**топологический векторный редактор**) * Файл топологии трафарета (**топологический векторный редактор**) * Монтажная схема платы (**топологический векторный редактор**) * Файлы автоматизированной сборки платы (**топологический векторный редактор**) * Чертёж финишных защитных покрытий платы (**топологический векторный редактор**) * Файлы топологии финишных покрытий платы (**топологический векторный редактор**) * Сборочный чертёж узла (**топологический векторный редактор**) * Перечень компонентов и расходных материалов узла (**текстовый редактор**) * Схема трассировки проводных соединений узла (**топологический векторный редактор**) * Описание процедуры приёмочного тестирования изделия (**текстовый редактор**) | Отдельные Моменты и Фичи: Срезы Проектов |
| **HL.CS-096** | Общесистемные | Должен быть **срез проекта** «**Отладка**», соответствующий пункту 6 опорного рабочего процесса, и включающий в себя документы следующих типов:   * Описание процедуры проверки требований к изделию (**текстовый редактор**) * Дефектная ведомость (**текстовый редактор**) | Отдельные Моменты и Фичи: Срезы Проектов |
| **HL.CS-097** | Общесистемные | Должен быть **срез проекта** «**Внедрение**», соответствующий пункту 7 опорного рабочего процесса, и включающий в себя документы следующих типов:   * Техническое описание (**текстовый редактор**) * Руководство по эксплуатации (**текстовый редактор**) * Входные данные для программирования (**текстовый редактор**, **редактор схем**) | Отдельные Моменты и Фичи: Срезы Проектов |
| **HL.CS-101** | Идеологические | Продукт должен предоставлять пользователю осуществить **релиз проекта** – одномоментно выполняющаяся фиксация всех изменений, синхронизация данных и проверка валидности. В случае сбоя одного из этих процессов или в случае провала валидации **релиз** должен считаться незавершённым, и продукт должен явно дать знать об этом пользователю. **Релиз** должен гарантировать пользователю, что все его **проектные** данные актуальны и прошли все необходимые проверки. | Отдельные Моменты и Фичи: Релизы |

# Проект №1: Подсистема Питания Печатной Платы

От системного уровня до уровня платы: этапы 2–4

## Методика Сбора Требований

### Входная Информация для Проектирования

* Структурная схема без подсистемы питания, но с пауэр-портами для каждого блока, с известными напряжениями питания и известными для каждого блока токами потребления
* Отдельные зафиксированные требования к подсистеме питания (максимальный входной ток потребления, раздельность источников питания для каких-либо пар блоков, наличие умного монитора питания или ключа для других блоков, необходимость дублирования источников питания для третьих блоков и т. п.)

### Детализованный Рабочий Процесс:

* Сбор требований
* Анализ требований
* Синтез архитектуры
* Выбор решений
* Выбор компонентов
* Проектировочный расчёт узлов
* Проверочный расчёт узлов
* Проверочный расчёт подсистемы
* Разработка функциональной схемы подсистемы
* Схемное моделирование подсистемы
* Анализ целостности сигналов подсистемы
* Моделирование теплового поля подсистемы

### Предполагаемые Условия и Ветвления в Рабочем Процессе

* Условие – выполнение требований
* Ветвление – варианты архитектуры
* Ветвление – варианты решений
* Ветвление – варианты компонентов
* Условие – проверочный расчёт узлов
* Условие – проверочный расчёт подсистемы
* Условие – моделирование
* Условие – анализ целостности

## Собранные Требования

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Ключ** | **Категория** | **Этап** | **Действия** | **Требования** |
| **HL.PS-001** |  | Входные данные | - | Должна быть разновидность **порта** – «**Силовой порт**»  Имена цепи на пауэр-портах с возможностью использовать имя как числовое значение напряжения питания |
| **HL.PS-002** |  | Информация о потреблении, связанная с каждым блоком |
| **HL.PS-003** |  | «Маяк» - условие, о выполнении которого разработчик должен явно сообщить системе. В противном случае проект нельзя зарелизить |
| **HL.PS-004** |  | Сбор требований | Составляется таблица входных требований к подсистеме питания по двум источникам – данным о требуемом электропитании функциональных блоков схемы (номинальные напряжения, токи потребления, уровни пульсаций), а также разбросанным по разным документам пометкам «на будущее» о различных требуемых особенностях архитектуры подсистемы электропитания (см. [Входная Информация для Проектирования](#_Входная_Информация_для)). | Таблица, автоматически подстраивающая набор столбцов и их заголовки под поля требований |
| **HL.PS-005** |  | Автозаполнение строки в таблице требований данными маяка по ссылке |
| **HL.PS-006** |  | Автогенерация требований – один блок на схеме порождает одно требование с числовыми значениями, напряжению питания и току потребления этого блока |
| **HL.PS-007** |  | Анализ требований | Составляется таблица итоговых требований к подсистеме питания. Требования по разным номинальным напряжениям объединяются в одно с суммарным током потребления. Требования к особенностям архитектуры просто остаются теми же самыми и копируются в таблицу итоговых требований. | Генерация требований для каждого напряжения питания с суммированием всех токов потребления |
| **HL.PS-008** |  | Перегенерация требования по вышестоящему требованию прямо в таблице требований |
| **HL.PS-009** |  | Синтез архитектуры | Исходя из требований к особенностям архитектуры, набора требуемых напряжений, суммарных токов потребления и допустимых пульсаций по каждому напряжению, а также исходя из информации о входном питании платы, придумываются несколько вариантов архитектуры и для каждого из них рисуется простая блок-схема. Составляется таблица сравнения, в которой перечислены придуманные варианты и критерии сравнения. Превью схем вставляются в таблицу.  Далее – разрыв в последовательности шагов проектирования, для каждого из предложенных вариантов производится выбор решений, чтобы получить информацию для сравнения. После получения необходимой для сравнения информации происходит возврат на этап синтеза архитектуры, а полученная информация добавляется в таблицу.  После заполнения таблицы сравнения принимается проектное решение, какой из вариантов будет использоваться. Документы или содержимое документов, относящиеся к другим вариантом, скрываются, чтобы не мешали восприятию информации. | Шаблон таблицы для выбора вариантов архитектуры |
| **HL.PS-010** |  | Предвстроенный блок выбора для таблицы выбора вариантов архитектуры |
| **HL.PS-011** |  | Возможность отобразить превью схемы в ячейке таблицы (для вариантов архитектуры) |
| **HL.PS-012** |  | Команда для таблицы, созданной по шаблону – скрыть все неиспользованные столбцы или строки |
| **HL.PS-013** |  | Зависимость документов или их отдельных частей от выбранного варианта в блоке выбора. Неактуальные для данного выбранного варианты документы или части документов скрываются или затемняются в навигаторе и документах. |
| **HL.PS-014** |  | Возможность автоматизировать выбор. Для этого нужно:  Преобразовывать в количественную форму качественные значения критериев выбора  Преобразовывать условные разбалловки в обычные числа  Проставлять веса критериям выбора |
| **HL.PS-015** |  | Выбор решений | Выбор схемотехнического решения производится для каждого из архитектурных блоков каждого из вариантов архитектуры.  Придумываются либо выбираются из уже использовавшихся и сохранённых в общем хранилище несколько вариантов схемотехнических решений. Для комплексных решений рисуется простая функциональная схема. Составляется таблица сравнения, в которой перечислены придуманные варианты и критерии сравнения. Превью схем вставляются в таблицу.  Далее – разрыв в последовательности шагов проектирования, для каждого из предложенных вариантов производится выбор наиболее существенных компонентов и последующие процедуры анализа (расчёты), чтобы получить информацию для сравнения. После получения необходимой для сравнения информации происходит возврат на этап выбора решений, а полученная информация добавляется в таблицу.  После заполнения таблицы сравнения принимается проектное решение, какой из вариантов будет использоваться. Документы или содержимое документов, относящиеся к другим вариантом, скрываются, чтобы не мешали восприятию информации. В одну из инвариантных частей документации добавляется краткая сводка по выбранному решению. | Автогенерация по структурной схеме функциональной – блоки превращаются либо в зоны, либо в заготовки функциональных блоков. |
| **HL.PS-016** |  | Шаблон таблицы для выбора схемотехнических решений |
| **HL.PS-017** |  | Предвстроенный блок выбора для таблицы выбора вариантов схемотехнических решений |
| **HL.PS-018** |  | Автоподстановка в заготовке функциональной схемы – для каждой заготовки блока должно быть контекстное меню с библиотекой решений, которые могут реализовать данную функцию |
| **HL.PS-019** |  | Подстановка выбранного решения для заготовки функционального блока в зависимости от информации в блоке выбора в таблице выбора решений |
| **HL.PS-020** |  | Возможность встроить в ячейку таблицы выбора значение, рассчитанное по формуле |
| **HL.PS-021** |  | Возможность использовать в формуле данные, связанные с конкретным решением как часть библиотечных данных |
| **HL.PS-022** |  | Выбор компонентов | Выбор наиболее существенных компонентов схемотехнического решения производится для каждого из предложенных решений для архитектурных блоков каждого из вариантов архитектуры.  Находятся либо выбираются из уже использовавшихся и сохранённых в общем хранилище несколько вариантов базовых компонентов – источников питания. Составляется таблица сравнения, в которой перечислены выбранные к сравнению варианты и критерии сравнения.  Далее – разрыв в последовательности шагов проектирования, для каждого из предложенных вариантов производится проектировочный расчёт и выбор наиболее существенных компонентов обвязки (катушек, трансформаторов, фильтрующих конденсаторов, задающих резисторов и конденсаторов), чтобы получить информацию для сравнения. После получения необходимой для сравнения информации происходит возврат на этап выбора компонентов, а полученная информация добавляется в таблицу.  После заполнения таблицы сравнения принимается проектное решение, какой из источников питания будет использоваться. Документы или содержимое документов, относящиеся к другим вариантам, скрываются, чтобы не мешали восприятию информации. В одну из инвариантных частей документации добавляется краткая сводка по выбранному источнику и его компонентам обвязки. | Шаблон таблицы для выбора компонентов для реализации схемотехнического решения |
| **HL.PS-023** |  | Предвстроенный блок выбора для таблицы выбора компонентов для реализации схемотехнического решения |
| **HL.PS-024** |  | Возможность по результату в блоке выбора сгенерировать краткое текстовое резюме компонента – какую-либо картинку, название, параметры. Эти данные извлекаются из библиотеки. |
| **HL.PS-025** |  | Возможность по результату в блоке выбора добавить в функциональную схему информацию про выбранный компонент – парт намбер, параметры. Данные взять из библиотеки. |
| **HL.PS-026** |  | Проектировочный расчёт узлов | Для выбранного источника питания с помощью интерактивных формул производится расчёт требуемых параметров основных компонентов обвязки (катушек, трансформаторов, фильтрующих конденсаторов, задающих резисторов и конденсаторов) и параметров источника питания как узла в целом.  Требуемая группа формул либо подгружается из хранилища как элемент данных хранимого там компонента, либо пишется вручную, а затем сохраняется как элемент данных вновь создаваемого компонента.  Каждая формула дополняется подписью с расшифровкой величин и нумеруется. | Автоматическое создание файла проектировочного расчёта для каждого из выбранных компонентов, если такое требуется. |
| **HL.PS-027** |  | Предложение компонентов обвязки по рассчитанным в проектировочном расчёте параметрам. В режиме автодополнения – система предлагает допустимые компоненты, инженер выбирает. |
| **HL.PS-028** |  | Проверочный расчёт узлов | Для выбранного источника питания с помощью интерактивных формул производится проверочный расчёт параметров узла – вычисляются их реальные значения, получаемые при подстановке параметров выбранных источника питания и компонентов обвязки.  Требуемая группа формул либо подгружается из хранилища как элемент данных хранимого там компонента, либо пишется вручную, а затем сохраняется как элемент данных вновь создаваемого компонента.  Каждая формула дополняется подписью с расшифровкой величин и нумеруется.  Пишется либо подгружается из шаблона для данного компонента последовательность интерактивных расчётов / сравнений параметров узлов, которые получаются при подстановке параметров реальных выбранных компонентов обвязки. Полученные параметры сравниваются с требуемыми, подгруженными из списка параметров выбранной микросхемы (либо вбиваются вручную при первом использовании микросхемы). В документе ставится яркая пометка о том, прошло ли проверочный расчёт рассчитанное решение на данных компонентах. | Автоматическое создание файла проверочного расчёта для каждого из выбранных компонентов с выбранными компонентами обвязки, если такое требуется. |
| **HL.PS-029** |  | Должен быть примитив, проверяющий условия по проверочному расчёту. В случае, если проверка не прошла, об этом нужно сигнализировать инженеру и запретить релиз проекта. |
| **HL.PS-030** |  | Проверочный расчёт подсистемы |  | В файле проверочного расчёта дерева электропитания нужно ссылаться на библиотечные данные компонентов (собственные потребления источников) и данные проекта (указанные в требованиях к подсистеме питания токи потребления по каждому питающемуся блоку; ну или опять же по библиотечным данным) |
| **HL.PS-031** |  |  | У переменной, являющейся представлением компонента, может быть поле типа «формула» - для случая, когда ток потребления нельзя просто описать фиксированным числом. Эти формулы нужно использовать в проверочном расчёте подсистемы и проектировочном расчёте узлов. |
| **HL.PS-032** |  | Схемное моделирование подсистемы |  | Автогенерация по функциональной схеме принципиальной – узлы превращаются в листы схемы, заполненные или нет, либо в зоны на схеме; они дополняются просто одиночными выбранными компонентами (с выбранной обвязкой или нет), либо в полноценный схемный сниппет, либо в нечто промежуточное – наряду с реальными полноценными компонентами и проводниками между ними располагаются заготовки связей и компонентов. При клике на компонент система должна предлагать допустимые на данной позиции компоненты. При клике на соединение система должна предлагать отрисовать его по-настоящему. |
| **HL.PS-033** |  |  | В Э3 каждому компоненту можно назначить модель для моделирования схемы питания |
| **HL.PS-034** |  |  | Должна быть возможность провести статическое моделирование схемы дерева электропитания с расчётом токов потребления, выходных напряжений и потоков мощности по библиотечным моделям, встроенным в компоненты Э3 |
| **HL.PS-035** |  |  | Для Э3 нужно создать соответствующий ей расширенный файл модели, в котором кроме информации о соединениях, компонентах и их параметрах будут также источники и измерители воздействий. |
| **HL.PS-036** |  |  | Можно добавить в схему проверщик условий – чтобы полученное при статическом моделировании питания значение токов потребления (или выходных напряжений источников тока, или мощностей источников) сравнивалось с допустимыми (по библиотечным данным компонентов) или с заданными (по требованиям или каким-то иным проектным данным) |
| **HL.PS-037** |  | Анализ целостности сигналов подсистемы |  | Если заданы требования по допустимым отклонениям напряжений источников питания, или допустимые минимальные или максимальные рабочие напряжения для потребителей, должна быть возможность провести статическое моделирование поля электрических токов на плате с расчётом поля потенциалов |
| **HL.PS-038** |  |  | Для файла платы нужно создать соответствующий ему расширенный файл модели платы, в котором кроме информации о топологии, компонентах и их параметрах будут также источники внешних воздействий. |
| **HL.PS-039** |  |  | Препроцессинг моделирования (включая генерацию расширенного файла модели) поля токов должен по максимуму осуществляться автоматически на основании данных препроцессинга схемного моделирования и принципиальной схемы |
| **HL.PS-040** |  |  | Можно добавить в плату проверщик условий – чтобы полученные при статическом моделировании поля токов и потенциалов значения электрических величин сравнивались с допустимыми (по библиотечным данным компонентов) или с заданными (по требованиям или каким-то иным проектным данным) |
| **HL.PS-041** |  | Моделирование теплового поля подсистемы |  | Если заданы требования по допустимому выделению тепла или допустимому перегреву или температуре в комплексе с данными о внешней среде (теплоотводы, замкнутый объём, корпусные детали, вентиляция) должна быть возможность провести статическое моделирование теплового поля, выделяющегося от источников питания и потребителей |
| **HL.PS-042** |  |  | Для файла платы нужно создать соответствующий ему расширенный файл модели платы, в котором кроме информации о топологии, компонентах и их параметрах будут также данные о тепловых условиях работы и объектах теплового окружения (корпуса, вентиляционные отверстия, теплоотводы) |
| **HL.PS-043** |  |  | Можно добавить в плату проверщик условий – чтобы полученные при статическом моделировании теплового поля значения тепловых величин сравнивались с допустимыми (по библиотечным данным компонентов) или с заданными (по требованиям или каким-то иным проектным данным) |

# Проект №:2 Простой проект платы

От структурной схемы до CAM-файлов

|  |  |  |  |
| --- | --- | --- | --- |
| **Этап** | **Шаг** | **Категория** | **Требования** |
|  |  |  |  |

# Проект №3: Многоплатное устройство в корпусе

|  |  |  |  |
| --- | --- | --- | --- |
| **Этап** | **Шаг** | **Категория** | **Требования** |
|  |  |  |  |

# Проект №4: Многоуровневая система вплоть до принципиальных схем

|  |  |  |  |
| --- | --- | --- | --- |
| **Этап** | **Шаг** | **Категория** | **Требования** |
|  |  |  |  |

# Проект №5: Одноплатное устройство в корпусе с DFMEA

От требований до сборочной документации

|  |  |  |  |
| --- | --- | --- | --- |
| **Этап** | **Шаг** | **Категория** | **Требования** |
|  |  |  |  |

# Проект №6: Электрика в квартире

|  |  |  |  |
| --- | --- | --- | --- |
| **Этап** | **Шаг** | **Категория** | **Требования** |
|  |  |  |  |