**1 Анализ предметной области**

База данных – это организованная структура, предназначенная для хранения информации. В современных базах данных хранятся не только данные, но и информация.

Это утверждение легко пояснить, если, например, рассмотреть базу данных крупногой справочной системы. В ней есть все необходимые сведения о транзисторах, об их параметрах, размерах, описании, формфакторе и т.д. Доступ к этой базе данных имеется у достаточно большого количества пользователей системы, но среди них вряд ли найдется такое лицо, которое имеет доступ ко всей базе полностью и при этом способно единолично вносить в нее произвольные изменения. Кроме данных, база содержит методы и средства, позволяющие каждому из пользователей оперировать только с теми данными, которые входят в его компетенцию. В результате взаимодействия данных, содержащихся в базе, с методами, доступными конкретным пользователям, образуется информация, которую они потребляют и на основании которой в пределах собственной компетенции производят ввод и редактирование данных.

С понятием базы данных тесно связано понятие системы управления базой данных. Это комплекс программных средств, предназначенных для создания структуры новой базы, наполнение ее содержимым, редактирование содержимого и визуализации информации. Под визуализацией информации базы понимается отбор отображаемых данных в соответствии с заданным критерием, их упорядочение, оформление и последующая выдача на устройства вывода или передачи по каналам связи.

В мире существует множество систем управления базами данных. Несмотря на то, что они могут по-разному работать с разными объектами и предоставляют пользователю различные функции и средства, большинство СУБД опираются на единый устоявшийся комплекс основных понятий.

Температурный режим транзистора – важнейший критерий работоспособности детали в конкретной схеме, его производительности, увеличении объемов потребления энергии, ухудшения свойств проводимости.

Рабочие нагрузки каждого прибора независимо от вида схемы определяются его личным коэффициентом подавления шума с учетом конечных нагрузок,. Минимальный размер корпуса транзистора устанавливается нормативными актами и мировыми стандартами.

В БД может храниться миллионы записей. В любое время можно найти запись, которая необходима в данный момент. Результатом поиска информации в приведенной БД могут быть названия, суммы, количество, даты. В базах данных можно проводить сортировку информации и вывод её на печать, удаление старой и вставка новой информации, просматривать БД целиком или по частям. С числами в таблицах можно проводить обычные математические операции. Параметры и названия деталей можно упорядочить по алфавиту.

Программное обеспечение для управления и поддержки работоспособности БД называют системой управления базами данных (СУБД). СУБД осуществляют ввод, проверку, систематизацию, поиск и обработку данных, распечатку их в виде отчётов.

Предметом исследования является проектирование базы данных по хранению наименований транзисторов и их параметров.

Практическая значимость работы заключается в поиске типичных недостатков применяемого в справочниках способа расчетов конечной информации и разработке рекомендаций по их устранению. Среди множества СУБД наиболее часто используются пакеты программ dBASE разных версий, FoxBase +, FoxPro, Fox Soft Ware, Clipper, совместимые с dBASE по системе команд и файлам. БД, созданная в одной СУБД, может использоваться в другой совместимой с ней СУБД, имеющей формат файлов dBASE (\*.dbf). Однако есть иные СУБД, например PARADOX и RBase, несовместимые с dBASE. Кроме СУБД для DOS, существуют СУБД, работающие в среде Windows, например Access, MS Works и др. В основе БД лежит представление данных в виде таблиц. Основными понятиями в СУБД являются поля и записи. В полях содержатся данные. Поле характеризуется длиной. Совокупность всех полей в строке называется записью. труктуру простейшей базы данных можно рассматривать как прямоугольную таблицу, состоящую из вертикальных столбцов и горизонтальных строк. Вертикальные столбцы принято называть полями, а горизонтальные строки — записями. Единицей хранимой информации является горизонтальная строка-запись, которая хранит информацию, например, об одном сотруднике фирмы. Каждая запись представляет собой совокупность полей. Числовые данные делятся на 2 вида: целые и вещественные. Длина числового поля должна быть достаточной, чтобы поместились знак числа, целая часть, точка (десятичная) и дробная часть.

**2 Анализ разработок в данной предметной области**

В данном курсовом проекте рассматривается создание базы данных справочника транзисторов и программы для ее использования. Программа рассчитана на предоставление пользователям информации о транзисторах по выбранному критерию, а также на манипулирование данными в базе.

С ростом численности микротехники стремительно усиливается и асортимент, что неизбежно приводит к необходимости струстурировать огромное колличество данных. В этих условия для качественного ведения и сбора данных необходимо вкладывать немалые средства и время. Один из устоявшихся и зарекомендовавших себя с лучшей стороны источник информации – это современные справочники по электротехники.

Целью автоматизации является оказание помощи пользователю в определении полного набора характеристик современных транзисторов, повышение эффективности поиска и предоставлении только той информации, которая действительно нужна.

Перед проектированием базы данных было проведено исследование разработок в данной области. Вот лишь наиболее ценный:

1. Справочник транзисторов (http://kazus.ru/guide/transistors/). База данных содержит основную информацию обо всех транзисторах общего назначения, как современных, так и старых выпусков, зарубежных и отечественных. В базе данных собраны сведения о месте расположения действующих магазинов продажи техники, полное наименование и закупочная стоимость, приведены качественные характеристики и описание назначения для использования в схемах.

База данных «Транзисторы» позволяет проводить сегментацию по формфактору, стоковым параметрам колектора и эмитера. Благодаря нескольким этапам на стадиях формирования запроса выборки в конечный вывод попадает лишь та информация, которая действительно востребована.

Определившись с выбором источника данных переходим к стадиям проектирования базы:

1. **этап. Постановка проблемы**

На этом этапе формируется задание по созданию БД. В нем подробно описывается состав базы, назначение и цели ее создания, а также перечисляется, какие виды работ предполагается осуществлять в этой базе данных (отбор, дополнение, изменение данных, печать или вывод отчета и т.д.).

1. **этап. Анализ объекта**

На этом этапе необходимо рассмотреть, из каких объектов может состоять будущая БД, каковы свойства этих объектов. После разбиения БД на отдельные объекты необходимо рассмотреть свойства каждого из этих объектов, установить, какими параметрами описывается каждый объект. Все эти сведения можно располагать в виде отдельных записей и таблиц. Далее необходимо рассмотреть тип данных каждой отдельной единицы записи (текстовый, числовой и т.д.). Сведения о типах данных также следует занести в составляемую таблицу.

1. **этап. Синтез модели**

На этом этапе по проведенному выше анализу необходимо выбрать

определенную модель БД. Далее рассматриваются достоинства и недостатки каждой модели, сопоставить их с требованиями и задачами вашей БД и выбрать ту модель, которая сможет максимально обеспечить реализацию поставленной задачи.

После выбора модели необходимо нарисовать ее схему с указанием связей между таблицами или узлами.

1. **этап. Способы представления информации, программный инструментарий**

После создания модели необходимо, в зависимости от выбранного программного продукта, определить форму представления информации. В большинстве СУБД данные можно хранить в двух видах:

- С использованием форм;

- Без использования форм.

Форма – созданный пользователем графический интерфейс для ввода данных в базу.

**3 Постановка задачи на проектирование**

Разработать базу данных по транзисторам для создания приложения справочного типа. Получаемая пользователем конечная информация будет представлена следующими полями:

* Имя транзистора
* Стандартные рабочие параметры содержащие:
* Коэфициент усиления h21
* Ток колектор-база, мкА
* Напряжение колектор-база, В
* Накопление заряда на колекторе, пФ
* Сопротивление на базе, Ом
* Ток эмитера
* Напряжение закрытия (насыщения), В
* Накопление заряда на эмиторе, пФ
* Температура
* Частоты, Мгц
* Коэфициенты шума, дБ
* Рисунок корпуса
* Описание назначения
* Предельные рабочие параметры содержащие:
* Ток колектора, макс мА
* Температура, общ
* Напряжение эмитер-база, В
* Напряжение колектор-база, В
* Температура, предел

Программа должна обладать возможностью создания новой записи, обновления уже существующей, удаления записи, поиском по содержимому.

**4 Общее описание системы**

**4.1 Описание структурной схемы**

**4.1.1 Описание технического обеспечения**

При разработке программы был использован компьютер MLCOMM-11M (Рисунок 1) со следующим набором аппаратных средств:

* Процессор Intel® Pentium® M 1,6 ГГц
* Память DDR 266/200 SO-DIMM до 2 Гб
* Резистивный сенсорный экран

Для использования программы требуется драйвер MySQL connector (см. ссылку 5).

Рисунок Компьютер MLCOMM-11M

**4.1.2 Описание программного обеспечения**

В основе программы заложена среда языковых запросов Oracle MySQL. Преимуществом данного продукта является:

* легкость
* простота установки
* надежность
* легко перенести на другой хостинг

Продукт распространяется как под GNU General Public License, так и под собственной коммерческой лицензией. Помимо этого, разработчики создают функциональность по заказу лицензионных пользователей. Именно благодаря такому заказу почти в самых ранних версиях появился механизм репликации.

MySQL является решением для малых и средних приложений. Входит в состав серверов WAMP, AppServ, LAMP и в портативные сборки серверовДенвер, XAMPP, VertrigoServ. Обычно MySQL используется в качестве сервера, к которому обращаются локальные или удалённые клиенты, однако в дистрибутив входит библиотека внутреннего сервера, позволяющая включать MySQL в автономные программы.

Среда разработки Visual Studio 8 C# .Net как одна из передовых технологий в сфере разработок прикладного програмного обеспечения.

**4.1.3 Описание лингвистического обеспечения**

Лингвистическое обеспечение включает в себя:

* Диалог управления – первоначальное окно с меню.
* Окно работы с содержимым таблиц – включает в себя просмотр содержимого таблицы, редактирования строк, создания новых записей, просмотр рисунка корпуса.
* Окно просмотра структуры таблицы – ознакомление с существующими полями таблицы, их типами и размерностью.
* Окно поиска – содержит в себе область просмотра результатов поиска и область для составления логики поисковых запросов к таблице.
* Область просмотра графического материала для сопоставления с записями в таблице.

**4.1.4 Описание методического обеспечения**

Целью справочника является обеспечение достоверной информации конечного пользователя. Помимо этого в программе преследовалось создание дружелюбного пользователю интерфейса с учетом текущих потребностей. Программа позволяет использовать два режима просмотра, стандартный и расширенный. Оба включают в себя просмотр содержимого таблицы, просмотр структуры данных таблицы, а также адаптивный поиск по всем параметрам полей. Администрирование позволяет выполнить операции удаления и создания записей, а также их редактирование.

Главное окно программы обеспечивает удобный переход к таким возможностям как просмотр информации о разработчике, адреса для скачивания исходников программы, просмотра информации общего назначения о подключении к базе, используемого режима кодировки. Окно входа для администрирования содержит поля для ввода логина и пароля, подтверждением правильности введеннх данных является зеленый лэйбл возле поля ввода.

Переход к просмотру содержимого таблиц осуществляется путем щелчка на области вне плитки или на жатием на плитку просмотра.

**4.2 Описание функциональной схемы**

Работа программы представляет статическую структуру проектируемой системы. Но этих знаний не достаточно для понимания и оценки её работы. Для это существует функциональная модель, которая описывает элементы и события, которыми они (элементы управления) связаны. Для проектируемой системы модель состояний предоставлена в приложении Б.

Пользователь, перейдя с главного окна программы имеет возможность просмотреть таблицу, перейти на вкладку просмотра параметров. Если происходит регистрация, пользователь получает права администратора. Возможен переход к поиску нужного транзистора. Переход осуществляется кликом на вкладках верхней области программы.

По клику на каждую строку таблицы выводятся вся информация, относящиеся к данному прибору. Сам прибор имеет поля в нижней части для редактирования. Из каждого сотояния реализована возможность перехода к началу – момент запуска программы под обычным пользователем.

Администратор может просматривать все таблицы, делать в них изменения, переходя к соответсвующим областям каждого окна.

**4.3 Описание даталогической схемы**

Данные (Таблица 1):

Таблица Данные

Первичным ключем таблицы является поле ‘name’, так как недопустимо нахождение двух транзисторов с одинаковым имянем. Остальные поля определены под хранение параметров и могут содержать пустое значение (null).

Для логики обращений к базе был написан отдельный класс с методами, отвечающими за отправку\прием данных от базы.

Для вывода всего содержимого таблицы в область просмотра был написан метод SelectAll():

\_command**.**CommandText **=** "Select \* from transistor"**;**

Составленная команда упаковывается в обертку метода Reader():

\_reader **=** \_command**.**ExecuteReader**();**

Reader представляет инструмент общения между сервером и клиентом по заранее выделеному каналу связи:

\_command **=** \_connection**.**CreateCommand**();**

Поиск использует авторскую технологию адаптивного составления запроса QwinCor.

Метод SelectLink() принимает на обработку массив вида ключ-значение:

KeyValuePair**<**string**,** KeyValuePair**<**string**,** string**>>**

, где для логики (OR, AND) строка имеет следующее содержание:

keyValuePair.Key || keyValuePair.Value, ключ содержит оператор логики, а параметр Value является резервной копией для препятствия возникновению ошибок. Второй вариант передачи параметров:

//для ПОЛЕ(Key) - МАСКА(Value.Key) - ЗНАЧЕНИЕ(Value.Value)

//keyValuePair.Key || keyValuePair.Value.Key || keyValuePair.Value.Value

, где keyValuePair.Key содержит поле, по которому осуществляется поиск, keyValuePair.Value.Key содержит маску для команды LINK, всего использовано три шаблона:

* ‘%выражение’ – совпадает со строками, заканчивающимися выражением.
* ‘выражение%’ – совпадает со строками, начинающимися выражением.
* ‘%выражение%’ – совпадает строка, содержащая выражение (в начале\конце\середине).

Шаблоном в общем виде является строка:

Select \* from transistor where pole like `maska+value+maska` AND ...

Результаты передаются в таблицу программы для дальнейшей обработки пользователем.

Добавление, удаление и изменение параметров транзистора осуществляется с помощью запросов INSERT…, DELETE… и UPDATE соответственно. За составление данных команд отвечают методы UpdateStr(), InsertStr(),DeleteStr().

**5 Пример работы в программе**

Шаги, необходимые для коректного запуска программы:

* Включаем сервер, в данном тестовом примере используется локальный сервер 127.0.0.1\Apache\
* Включаем базу данных на сервере посредством phpMyAdmin
* Запускаем программу справочник

Рисунок Меню выбора

Перед пользователем главное окно – меню выбора (Рисунок 2).

Далее по порядку, вход администратора (Рисунок 3) открывает поля для внесения изменений в таблицу.

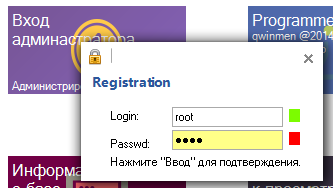
Следующая палень – контактная информация о разработчике (Рисунок 4).

Рисунок Вход администратора

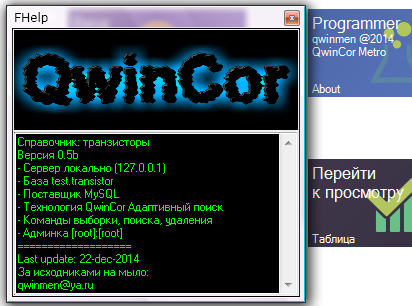


Рисунок Контактная информация

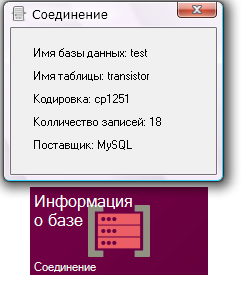
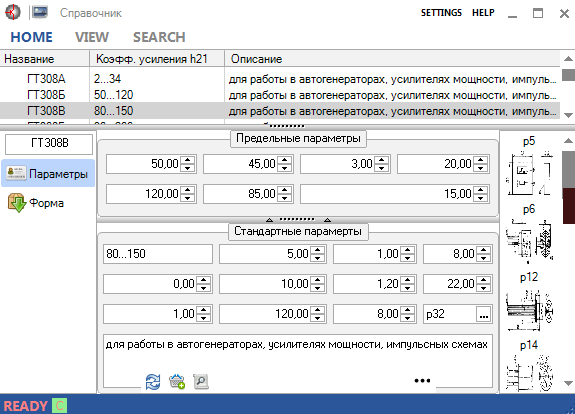
Следующая плитка содержит обобщеную справку о базе (Рисунок 5).

Рисунок 6 Окно редактора

Рисунок 5 Информация о базе данных

Последний пункт и наиболее важный – отобрадение таблицы (рисунок 6).

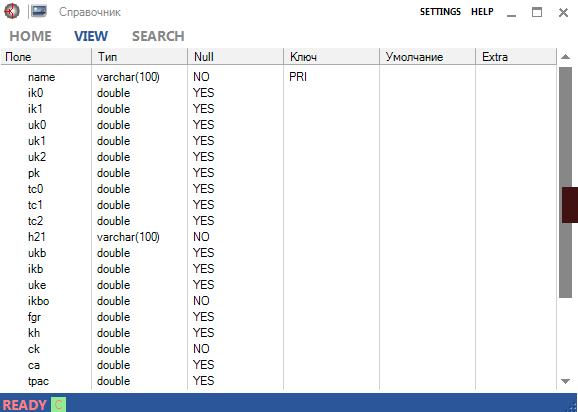
Перейдя на вкладку Просмотр, пользователь сможет узнать подробней, какие поля есть в таблице на сервере, какие им назначены типы данных, а также из размер (Рисунок 7).Последнее окно – Поиск, содержит две области, верхняя в виде таблицы для вывода результатов поиска, и вторая область – составление логики запроса. Благодаря применению технологии адаптивного составления запроса пользователь, даже несведующий в базах данных, сможет с легкостью найти нужный ему транзистор или группу транзисторов. Чтобы произвести поиск, необходимо кликнуть один раз на кнопку «Поиск», после этого действия рядом появится окно, в котором задаются критерии поиска. Если необходимо сделать расширенный выбор, то нажав на значек пазла к текущему окну присоединится новое, уже с выбором логческого оператора (OR или AND).

Рисунок 7 Просмотр полей таблицы

Для начала поиска используется двойной клик по кнопке «Поиск», а за сброс поля с уже сформированным запросом отвечает кнопка «Clear» (Рисунок 8).

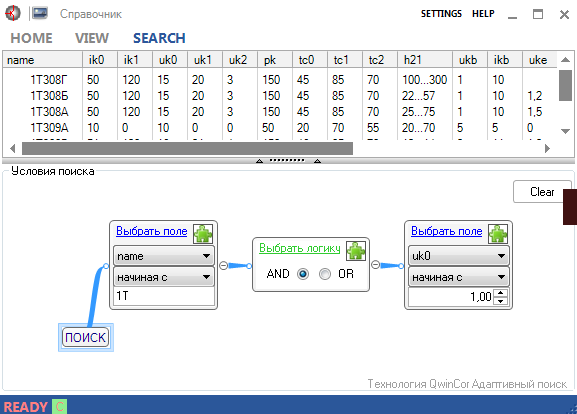
В итоге таблица содержит результаты всех совпадений запроса на поиск данных.

Рисунок 8 Поиск

**Заключение**

В итоге получился справочник, который полностью оправдывает использование в нем технологий MySQL для хранения информации. Простота в использовании поможет пользователю без специальных технических знаний свободно ориентироваться в программе и максимально быстро выполнять все поставленные задачи. Благодаря этому стало возможным повысить эффективность работы. А администрирование стало гораздо комфортнее по сравнению с обычним редактированием таблиц через веб интерфейс сервера.

**Список использованных источников**

1. Справочник по биполярным транзисторам, URL: http://www.qrz.ru/reference/kozak/bipol/bih01.shtml (дата обращения 24.12.2014).
2. Анализ предметной области и разработка структуры базы данных, URL: http://www.fbunion.ru/library/buh/bazy (дата обращения 23.12.2014).
3. Базы данных. Общие понятия, URL: http://egeinf.gym5cheb.ru/p35aa1.html (дата обращения 22.12.2014).
4. Раскрытие тайн SQL, Э. Оппель, 2007г. СПБ-Питер.
5. Драйвер MySQL connector, URL: http://dev.mysql.com/downloads/connector/net (дата обращения 21.12.2014).
6. Библия C#, М. Фленов, 2013г. СПБ-Питер.

|  |
| --- |
| ***Программное обеспечение*** |
| * СУБД MySQL * PHPMyAdmin * Visual Studio 8 C# |
| ***Техническое обеспечение*** |
| MLCOMM-11M со следующим набором аппаратных средств:   * Процессор Intel® Pentium® M 1,6 ГГц * Память DDR 266/200 SO-DIMM до 2 Гб * Резистивный сенсорный экран |
| ***Лингвистическое*** |
| * Диалог управления – первоначальное окно с меню. * Окно работы с содержимым таблиц – включает в себя просмотр содержимого таблицы, редактирования строк, создания новых записей, просмотр рисунка корпуса. * Окно просмотра структуры таблицы – ознакомление с существующими полями таблицы, их типами и размерностью. * Окно поиска – содержит в себе область просмотра результатов поиска и область для составления логики поисковых запросов к таблице. * Область просмотра графического материала для сопоставления с записями в таблице. |
| ***Методическое обеспечение*** |
| * Руководство пользователя |

**Приложение А** – структурная схема

**Приложение Б** – функциональная схема



**Приложение В** – даталогическая схема

Нотация IDEF1X