

Цель

Целью проекта является разработка Android-приложения для чтения и генерации цветовой маркировки резисторов.

Обзор аналогов

- **ElectroDroid** – нет функции генерации цветовой маркировки
- **Резистор калькулятор** – не оптимизировано под сенсорные экраны, нет функции поиска ближайшего стандартного номинала
- **ResistorColorCode** – нет функции поиска ближайшего стандартного номинала
- **Маркировка резисторов** – нет функции генерации цветовой маркировки
- **ResistorCode** (А. Кодачигов) – не оптимизировано под сенсорные экраны

Актуальность темы

Людям, которые занимаются радиоэлектроникой, при ремонте или сборке схем необходимо знать характеристики радиоэлементов. Одним из таких элементов является резистор.

Обычно резисторы маркируются цветовым или числовым кодом. По сравнению с цветовым кодом, числовой легко воспринимается и его расшифровку можно произвести в уме, не используя справочники. Поэтому человеку значительно проще ввести данные цвета в программе и узнать сопротивление резистора, чем самому высчитывать это сопротивление.

Постановка задачи

В программе должны быть реализованы следующие функции:

- Чтение 4-, 5- и 6-полосных цветовых маркировок
- Поиск ближайшего стандартного номинала
- Генерация цветовой маркировки по характеристикам резистора

Основные требования к интерфейсу:

- Быстрое переключение между 4-, 5- и 6-полосными маркировками
- Наличие текстовых подписей в списках выбора цветов
- Простой интуитивно понятный интерфейс

Для корректной работы программы необходимы:

- ОС Android версии 4.1 и выше
- Устройства ввода/вывода: сенсорный дисплей, монитор (с разрешением не менее 800x480).

Схема взаимодействия функциональных блоков

- **Блок загрузки данных.** Данный блок необходим для формирования исходной маркировки, параметров для построения цветовой маркировки.
- **Блок обработки данных.** Данный блок производит разбор цветовой маркировки для получения параметров резистора, выполняет построение маркировки исходя из характеристик резистора, а также находит ближайшие стандартные номиналы.
- **Блок вывода данных.** Отвечает за вывод информации в удобном для человека виде на экран.

Математический аппарат

Математический аппарат представляет собой абстрактный автомат с конечным количеством внутренних состояний. Он изображён на рисунке 2. На схеме: вершины – состояния программы, а дуги – входные сигналы, подаваемые пользователем или генерируемые самой программой.

A1 – начальное состояние программы A2 – обработка цветовой маркировки A3 – окно записи свойств резистора A4 – обработка характеристик резистора A5 – закрытие программы	α1 – выбор цвета полосы маркировки α2 – смена количества полос маркировки α3 – программный сигнал для отображения характеристик резистора α4 – переход в состояние сбора характеристик резистора α5 – построение цветовой маркировки α6 – программный сигнал для отображения цветовой маркировки α7 – переход в начальное состояние программы α8 – сигнал для выхода из программы
--	--

Узкими местом данного математического аппарата является то, что он не проверяет справочные данные на наличие изменений извне.

Схема алгоритмов

- Алгоритм чтения цветовой маркировки
- Алгоритм поиска ближайших стандартных номиналов
- Алгоритм построения цветовой маркировки

Модульная структура программы

1. **ResistorColor** – основной модуль программы, отвечает за работу с интерфейсом программы, за взаимодействие между модулями и за работу с операционной системой.
2. **LabelReader** – модуль чтения цветовой маркировки резисторов, отвечает за считывание цветов полос и определения характеристик резистора.
3. **LabelBuilder** – модуль построения цветовой маркировки, отвечает за построение цветовой маркировки резистора по его характеристикам.
4. **DataBase** – модуль для работы со стандартными номиналами, отвечает за поиск ближайших стандартных номиналов.

Разработка структуры интерфейса

Т. к. данное программное обеспечение разрабатывается под мобильные устройства, у которых чаще всего небольшие экраны, было принято решение распределить основные функции программы на несколько экранов. Также при разработке интерфейса необходимо было предусмотреть возможность поворота экрана устройства.

Заключение

В ходе выполнения курсового проекта было разработано приложение, содержащее в себе такие функции как: чтение цветовой маркировки резистора, генерация цветовой маркировки по характеристикам резистора, поиск ближайших стандартных номиналов. Был разработан пользовательский интерфейс, ориентированный на работу с мобильного устройства.

В качестве одного из направлений дальнейшего развития приложения можно выбрать распространение функционала и на другие радиодетали, которые так же используют цветовой код (конденсаторы, индуктивности, диоды, транзисторы).

Стоит так же обратить внимание и на реализацию такой функции, как считывание цветов полос радиодеталей с помощью камеры с последующим разбором цветовой маркировки. Такое приложение практически не будет иметь аналогов. К сожалению, данная функция не была реализована, т. к. ее разработка требует значительного количества времени и средств.