

## РЕАЛИЗАЦИЯ ПЛИС НА ОСНОВЕ МИКРОКОНТРОЛЛЕРА АТМЕГА328

Голов А.С., Лагутик Р.Ю., студенты гр. 458304

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Ковшер Е.И. – ассистент

Представлена реализация программируемой логической интегральной схемы (ПЛИС) на основе микроконтроллера ATmega. Система предназначена для изучения булевых функций и анализа логических схем. Рассматриваются методы задания и обработки булевых функций, а также их интерпретация микроконтроллером. Разработано программное обеспечение для взаимодействия с устройством, позволяющее визуализировать работу логических схем в реальном времени.

В настоящее время в цифровых устройствах широко используются программируемые логические интегральные схемы (ПЛИС), что требует разработки доступных и гибких решений, позволяющих пользователям реализовывать логические функции без использования специализированного оборудования. В условиях активного развития технологий автоматизированного проектирования и цифровых систем, поиск экономичных и удобных способов реализации логики становится особенно актуальным. Одним из возможных подходов является программная эмуляция ПЛИС на базе микроконтроллеров, таких как ATmega. Такой метод дает возможность значительно снизить затраты на разработку логических устройств, повысить гибкость их настройки, а также облегчить интеграцию в более сложные системы.

В данной работе представлена система, позволяющая формировать логические схемы на основе заданных пользователем булевых выражений в виде СДНФ или СКНФ. Эта система обеспечивает динамическое управление выходами устройства в зависимости от входных состояний, поступающих на плату, что в свою очередь дает возможность программной реализации логики на микроконтроллере. Это особенно актуально для образовательных проектов, где важно изучение принципов цифровой логики без необходимости использования дорогостоящего оборудования.

Для реализации устройства используется микроконтроллер ATmega328 на отладочной плате Arduino UNO (см. рисунок 1), который обеспечивает необходимую вычислительную мощность для обработки логических функций. Взаимодействие с пользователем происходит в виде задания булевых функций через COM-порт. Такой процесс делает систему гибкой, удобной для работы с различными логическими структурами и расширяемой для будущих версий.



Рисунок 1 — Отладочная плата Arduino UNO

**Задание булевых функций.** Пользователь должен ввести булеву функцию вида [ТИП]; [ФУНКЦИЯ], где [ТИП] означает выбор между СКНФ и СДНФ функции, а [ФУНКЦИЯ] содержит саму булеву функцию, которую будет исполнять микроконтроллер. Для выбора типа функции пользователю следует ввести команду SDNF или SKNF. Затем через символ ";", программа ожидает ввода функции. Стоит учесть, что все термины друг от друга должны быть изолированы скобками. Операция логического НЕ задается символом "!", операции логического ИЛИ и И — символами "|" и "&" соответственно. Для сброса введенных функций пользователю необходимо ввести команду "SBROS", после чего будет предложено ввести функции заново.

**Выполнение функций.** В данной версии устройства, на вход микроконтроллера могут поступать на вход и выход только четыре сигнала (см. рисунок 2).

После задания функций устройство начинает свою работу. Входные сигналы считываются и результат каждой функции в зависимости от исходных данных поступает на выход ПЛИС.

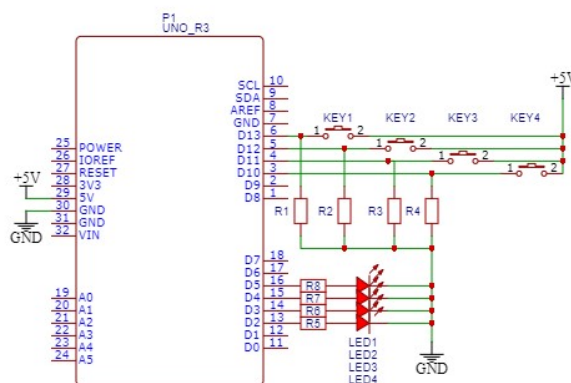


Рисунок 2 — Принципиальная схема устройства ПЛИС

При реализации ПЛИС одной из проблем является создание парсера булевых выражений для их верификации и последующей обработки. Алгоритм преобразования функций из строки в выражение, понятное микроконтроллеру, включает несколько ключевых этапов:

1. Разделение строки на тип и выражение: первоначально строка должна быть разделена на два компонента: тип функции (СДНФ или СКНФ) и сама логическая функция. Проверяется первая часть строки на наличие ключевых слов "СДНФ" или "СКНФ", а оставшаяся часть строки используется как логическое выражение для дальнейшей обработки.

2. Определение структуры: для СДНФ термы разделяются через символ "|", а литералы внутри термов — через символ "&". Для СКНФ термы разделяются через символ "&", а литералы внутри термов — через символ "|".

3. Разбиение на термы: для СДНФ разделение происходит по символу "|", для СКНФ по символу "&".

4. Разбор термов на литералы: для СДНФ: внутри каждого терма литералы разделяются символом "&". Для СКНФ внутри каждого терма литералы разделяются символом "|". Для каждого литерала нужно определить, является ли он инвертированным. Если литерал начинается с символа "!", то это указывает на инверсию.

5. Проверка ошибок: баланс скобок: необходимо проверить, что в выражении правильно расставлены открывающие и закрывающие скобки. Переменные должны быть в пределах допустимых символов ("a" до "d"). При иных символах устройство выдает ошибку. Если встречается пустой терм (например, "(" или "a&&b|"), то это также является ошибкой, так как такой терм не имеет смысла.

В результате разработки была создана первая версия устройства, позволяющая интерпретировать написанные пользователем функции в готовые цифровые логические устройства. Данное решение может быть интегрировано в различные электрические цепи, работающие с напряжением 5В, обеспечивая удобство использования и широкую сферу применения. Использование доступного микроконтроллера делает данную ПЛИС гибкой и универсальной, позволяя применять ее как в образовательных целях, для изучения логических функций и принципов работы цифровых схем, так и в практических задачах, требующих гибкой настройки логики без необходимости использования специализированного оборудования. В дальнейшем планируется расширение возможностей системы, включая поддержку сложных логических выражений, увеличение числа входов и выходов, а также интеграцию с другими микроконтроллерами и цифровыми устройствами.

#### Список использованных источников:

1. Программная эмуляция ПЛИС на микроконтроллерах / А.С. Иванов, П.В. Смирнов // *Электронные компоненты и системы*, 2020. — С. 45-52.
2. Цифровые логические схемы на базе микроконтроллеров / В.Н. Петров, И.В. Кузнецов.—М.: Технополис, 2018. — 320 с.
3. Оптимизация булевых выражений для встроенных систем / Е.В. Соколов [и др.] // *Вестник инженерных исследований*, 2019. — С. 78-85.