

СИНТЕЗ РЕГИСТРОВОЙ ЛОГИКИ НА PLD

Соловьев В.В., Самаль Д.И

Наибольшие трудности при проектировании цифровых систем роботики и автоматики вызывает синтез последовательностных схем, содержащих элементы памяти в виде триггеров, защелок, регистров и др. В ряде случаев применение формальных методов синтеза, например, теории автоматов, не всегда возможно или приводит к неэффективным решениям. Поэтому последовательностные схемы часто проектируются на основе опыта и интуиции разработчиков. Но, поскольку большинство схемотехников оперируют категориями элементов малой и средней степени интеграции, то и последовательностные схемы строятся на их основе.

В данной работе предлагается последовательностные схемы строить на новой элементарной базе - **Programmable Logic Devices (PLDs)**. При этом не изменяется способ мышления проектировщика: он разрабатывает произвольную последовательностную схему из вентилях, защелок и триггеров, а затем эта схема "вкладывается" в заданную структуру **PLD**.

Современные **PLD** характеризуются низкой стоимостью, высоким быстродействием, значительными функциональными возможностями (одно **PLD** может заменить несколько десятков корпусов традиционной "жесткой" логики), многократностью перепрограммирования, низкой потребляемой мощностью и др. При использовании современных средств автоматизированного проектирования разработка на основе **PLD** даже достаточно сложных проектов может составлять всего несколько часов. Однако широкому практическому использованию **PLD** препятствуют ограничения в формах представления цифрового устройства: комбинационная схема или конечный автомат.

Произвольная совокупность соединенных между собой вентилях и триггеров называется регистровой логикой. В работе рассматривается алгоритм синтеза регистровой логики на **PLD**, который заключается в последовательностном формировании групп элементов регистровой логики, реализуемой на одной **PLD**. Для улучшения полученного решения используется итерационный алгоритм взаимного обмена элементов из разных групп. Главным критерием оптимизации является минимизация числа **PLD**, вспомогательным - минимизация числа внешних связей между **PLD**. Представляется также программная реализация метода и ее связь с известными пакетами автоматизированного проектирования (**CUPL** и **P-CAD**).