

ФБГОУ ВПО ФАЖТ

Иркутский государственный университет путей сообщения

экз №

На правах рукописи

**Мухопад Александр Юрьевич**

**АНАЛИЗ И СИНТЕЗ УСТРОЙСТВ УПРАВЛЕНИЯ ПРОБЛЕМНО-ОРИЕНТИРОВАННЫМИ СРЕДСТВАМИ ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ И СЛОЖНЫМИ ТЕХНИЧЕСКИМИ СИСТЕМАМИ**

Специальность 05.13.05 – Элементы и устройства вычислительной техники и систем управления

Диссертация на соискание ученой степени доктора технических наук

Научный консультант доктор  
технических наук, доцент  
Сизых Виктор Николаевич

2015

## ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ.....	4
ГЛАВА 1. УПРАВЛЯЮЩИЕ СПЕЦПРОЦЕССОРЫ СЛОЖНЫХ ТЕХНИЧЕСКИХ СИСТЕМ.....	14
1.1. Концепции развития информационно-управляющих систем реального времени.....	14
1.2. Проектирование управляющих спецпроцессоров.....	19
1.3. Структурные модели информационно-управляющих систем.....	27
1.4. Методология синтеза управляющих спецпроцессоров.....	33
1.5. Структурная организация и классификация управляющих автоматов.....	35
Выводы по главе 1.....	41
ГЛАВА 2. СИНТЕЗ УПРАВЛЯЮЩИХ АВТОМАТОВ.....	42
2.1. Методы структурного синтеза управляющих автоматов.....	42
2.2. Новый метод синтеза управляющих автоматов.....	44
2.3. Структурный синтез управляющих автоматов.....	49
2.4. Анализ структурной организации управляющих автоматов.....	54
2.5. Сравнительный анализ комбинационных схем УА.....	57
2.6. Варианты структурной организации УА нового типа.....	59
2.7. Синтез быстродействующих автоматов.....	62
2.8. Синхронизация управляющих автоматов нового типа.....	70
2.9. Гибридный способ синхронизации управляющих автоматов.....	71
Выводы по главе 2.....	71
ГЛАВА 3. СИНТЕЗ КОМПЛЕКСНЫХ АВТОМАТОВ.....	75
3.1. Структурная организация сложных автоматов.....	75
3.2. Синтез управляющих автоматов с импульсно - кодовой памятью состояний.....	77
3.3. Метод синтеза многорежимных управляющих автоматов.....	83
3.4. Синтез многопрограммных автоматов управления.....	93
3.5. Управляющие автоматы с внутренним автоматом синхронизации.....	99
3.6. Структурно - автоматное программирование.....	100

Выводы по главе 3.....	111
ГЛАВА 4. СИНТЕЗ УПРАВЛЯЮЩИХ АВТОМАТОВ С ДИНАМИЧЕСКИМ КОНТРОЛЕМ ФУНКЦИОНИРОВАНИЯ.....	113
4.1. Методы и средства динамического контроля управляющих автоматов.....	113
4.2. Контроль автоматов управления с избыточными кодами.....	116
4.3. Контроль УА с разделенным кодом схем переходов.....	118
4.4. Метод контроля автоматов с возвратом к исходному коду.....	119
4.5. Динамический контроль сложных управляющих автоматов.....	122
4.6. Контроль управляющих автоматов с использованием соседнего кодирования половинных кодов состояний.....	124
4.7. Методология проектирования самоконтролируемых управляющих автоматов сложных технических систем реального времени .....	140
Выводы по главе 4.....	142
ГЛАВА 5. СПЕЦПРОЦЕССОРЫ И УПРАВЛЯЮЩИЕ АВТОМАТЫ СЛОЖНЫХ ТЕХНИЧЕСКИХ СИСТЕМ.....	144
5.1. Управление спецпроцессором корреляционно- экстремальной навигации.....	144
5.2. Спецпроцессор классификации изображений.....	161
5.3. Спецпроцессор управления нефтеперекачивающей станцией.....	172
5.4. Спецпроцессор криптографической защиты информации.....	180
5.5. Управление реактивным приводом.....	186
5.6. Управление подсистемами транспортной безопасности.....	191
5.7. Управление прибором «Мультимед» .....	201
Выводы по главе 5.....	211
ЗАКЛЮЧЕНИЕ И ОСНОВНЫЕ РЕЗУЛЬТАТЫ.....	212
СПИСОК СОКРАЩЕНИЙ.....	215
СПИСОК ЛИТЕРАТУРЫ.....	217
ПРИЛОЖЕНИЯ.....	244

## ВВЕДЕНИЕ

Для контроля и управления электромеханическими системами, роботами, летательными аппаратами, сложными техническими системами, работающими в экстремальных условиях эксплуатации, разработка специализированных информационно-управляющих систем и специализированных процессоров (СПР) относится к числу важных научных направлений.

Структурная организация СПР отлична от «универсальных» средств обработки информации, т.к. специфика определяется в первую очередь используемыми принципами контроля и управления в реальном времени. Наивысшая адекватность структурной организации СПР специальным алгоритмам чаще всего достигается за счет создания специальных управляющих устройств.

В настоящее время информационные технологии на основе весьма сложных программных средств составляют основу как стационарных, так и подвижных систем. Однако чем сложнее управляющие программы, тем больше риск их повреждения за счет внешнего вредного воздействия (хакерские атаки). Причем программные средства защиты информации также подвержены атакам. Поэтому начали интенсивно развиваться аппаратные методы криптографической защиты информации, и возникла острая необходимость замены ведущих (ответственных) программных средств аппаратными управляющими автоматами. Современный уровень развития интегральной схемотехники позволяет произвести такую замену. Однако методы проектирования сложных управляющих автоматов эквивалентных программным подсистемам не достаточно развиты, т.к. теория синтеза автоматов развивалась применительно к реализации комплекса относительно простых микропрограммных автоматов вычислительной техники.

Диссертационные исследования направлены на создание методики проектирования управляющих автоматов (УА) для спецпроцессоров. При этом не рассматриваются задачи, требующие для своей реализации комплекса параллельно-взаимодействующих процессоров с матричными структурными

связями и физической или программной коммутацией информационных каналов. Такие проблемно-ориентированные системы используются в том случае, когда имеется возможность параллельной реализации вычислительных операций.

В задачах управления мехатронными системами, технологическим оборудованием, пневмо и электроприводами и др. вычислительные операции не являются основными. В большинстве случаев их доля незначительна по сравнению с операциями формирования логических условий и исполнительных команд управления агрегатами. А поэтому возможность распараллеливания весьма ограничена.

В практическом плане такие СПР выполняют функции интерфейсов, таблично-алгоритмических функциональных преобразователей, опознавателей образов, систем управления роботами и механизмами с повышенной энергетической мощностью и специальных подсистем в аппаратной реализации устройств оборонного назначения. Управляющие устройства таких систем существенно отличаются от УА для серийной вычислительной техники.

В диссертационных исследованиях разработаны методы именно структурного (не функционального) синтеза СПР с аппаратно-программными средствами управления с новой организацией управляющих автоматов, предложенных диссертантом.

Задача разработки методики синтеза СПР и УА относится к числу наукоемких направлений исследований в области технической кибернетики.

УА серийной вычислительной техники относятся к классу микропрограммных автоматов (МПА), в которых объединяются в единое управляющее устройство десятки относительно простых микропрограмм с проверкой ограниченного числа ( $q < 6-8$ ) логических условий и числом операторов действия  $N < 30$ .. Методика синтеза МПА достаточно хорошо разработана. Для СПР ситуация противоположна, т.к. необходимо создать УА, реализующий один, но весьма сложный алгоритм как по количеству операторов, так и по количеству логических условий. Тем не менее, эта специфика УА для СПР, как правило, не учитывается. Поэтому задача анализа и синтеза УА для

проблемно-ориентированных средств вычислительной техники и сложных технических систем является актуальной в связи с развитием элементной базы интегральной схемотехники и расширения сфер применения СПР.

**Целью** диссертационных исследований является развитие методов анализа и структурного синтеза устройств управления спецпроцессоров технических систем реального времени.

**Предметом исследования** является структурная организация спецпроцессоров, автоматов управления.

**Объект исследования** - методы анализа, синтеза и проектирования управляющих автоматов.

**Задачи исследования:**

- разработка метода синтеза сложных управляющих автоматов;
- разработка структурной организации управляющих автоматов при большом числе логических условий и состояний;
- разработка структурной организации взаимодействующих автоматов;
- сравнительный анализ разработанных управляющих автоматов с основными структурами автоматов Мура и Мили;
- разработка метода моделирования сложных управляющих автоматов;
- разработка новых методов контроля и повышения надежности сложных управляющих автоматов;
- разработка спецпроцессоров, алгоритмов управления ими для важных народно-хозяйственных применений.

**Методы исследования** основаны на системном анализе, теории дискретных устройств и автоматов, теории кодирования информации, теории контроля и диагностики и методов моделирования систем обработки информации и управления.

**Научная новизна** исследований определяется следующими положениями:

1. Выполнен системный анализ методов и средств синтеза спецпроцессоров и управляющих подсистем за значительный временной период (1960-2015г.г.) и сделан вывод о целесообразности использования пятикомпонентной модели из

функциональной, информационной, логической, адресной и управляющей подсистем для синтеза спецпроцессоров и анализа устройств управления.

2. Предложен новый метод синтеза управляющих автоматов (УА), основанный на преобразовании операторной схемы алгоритма и обеспечивающий введение в структурную организацию мультиплексора со схемой адресации для выбора единственного логического условия из всего множества условий по коду состояния УА. Метод позволяет снизить объем оборудования (по верхней оценке через объем ПЗУ комбинационных схем) в  $2^{q-1}$  раз. Это особенно важно для сложных автоматов с числом входных переменных  $q = 16 \div 24$  и разрядностью кода состояний  $m = 6 \div 8$ .

3. Предложены варианты структурной организации УА и модификация базового метода синтеза, обеспечивающие дополнительное снижение оборудования на реализацию комбинационных схем автоматов  $\approx$  в 2 раза по сравнению с новой структурной организацией УА.

4. Предложен метод синтеза УА с памятью на счетчике, метод декомпозиции, метод структурной организации иерархических и взаимодействующих автоматов на основе новой методики синтеза УА с новой структурной организацией.

5. Предложены оригинальные методы динамического контроля и диагностики автоматов: а) метод представления кода с числом «1» равным количеству трехразрядных групп ( $p$ ) в выходном коде комбинационной схемы переходов - код  $pCn$ . Этот код преобразуется в двоичный непозиционный код (ДНК) состояния автомата для подачи на вход комбинационной схемы. По сравнению с известным кодом  $kCn$ , в котором «1» расположены произвольно, фиксация наличия числа единиц в группах кода  $pCn$  осуществляется на существенно меньшем оборудовании. Метод позволяет контролировать комбинационную схему переходов, как по входу, так и по выходу. В соединении с методом синтеза УА с разделенными комбинационными схемами на младшие и старшие разряды контроль с кодами  $pCn$  реализуется с наименьшими затратами оборудования по сравнению с кодами  $kCn$ ; б) метод реализации дублирующей схемы в виде двух схем сопоставляющих коды предыдущего и последующего состояний

по их половинным частям с добавлением к кодам каждой половины двух разрядов, значения одного из которых доопределяется. Метод используется для автоматов нового типа, предложенных соискателем; в) метод мажоритарного резервирования с использованием резервной схемы с преобразованием выходного кода комбинационной схемы переходов во входной код предыдущего состояния автомата. Метод отличается быстродействием принятия решений при выборе одной из двух резервных схем. Предложена оригинальная реализация с представлением всех дублирующих схем в виде части таблиц переходов с синхронизацией частей по значению «0» или «1» логического условия выбранного мультиплексором. Метод позволяет осуществить не только контроль, но и диагностику управляющего автомата в условиях потока отказов.

6. Предложен метод динамического контроля автоматов с использованием счётчика Грея, делением кодов предыдущего и последующего состояния пополам и специальной процедуры выбора и модификации кодов для состояний, не относящихся к счётчику. Метод отличается тем, что требует минимальных затрат оборудования на средства встроенного контроля и не снижает быстродействия автомата управления.

7. Предложен новый метод моделирования автоматов управления и его обобщение, названный методом структурно-автоматного программирования микроконтроллеров и спецпроцессоров реального времени. Предложенный метод позволяет получать наиболее простые и быстродействующие программы для любых типов микроконтроллеров с единой ведущей программой опроса трех зон ОЗУ по алгоритму функционирования УА.

8. Предложены алгоритмы и структурная организация спецпроцессора с УА нового типа для криптографической защиты информации, отличающиеся тем, что кодирование информации осуществляется аппаратным способом за единицы микросекунд при высоком уровне криптостойкости недостижимом другими методами.

9. Разработаны оригинальные спецпроцессоры для народно-хозяйственного применения в различных отраслях промышленности (управление мехатронными



устройствами, нефтехимическими установками, устройствами навигации подвижных объектов, контроля параметров агрегатов и др.).

10. Предложена новая методология проектирования управляющих автоматов спецпроцессоров сложных технических систем реального времени.

**Соответствие специальности** определяется направлением исследований на «совершенствование и создание принципиально новых элементов и устройств вычислительной техники и систем управления, включая разработку научных основ физических и технических принципов создания указанных элементов и устройств». Диссертационные исследования соответствуют области исследований специальности 05.13.05 по трем из четырех пунктов:

п.1 «Разработка научных основ создания и исследования общих свойств и принципов функционирования элементов, схем и устройств вычислительной техники и систем управления». По п.1 проведены исследования, позволившие разработать методические основы создания моделей спецпроцессоров и управляющих устройств для технологических процессов и сложных технических систем.

п.3 «Разработка принципиально новых методов анализа и синтеза элементов и устройств вычислительной техники и систем управления с целью улучшения их технических характеристик». По п.3 выполнены теоретические исследования, позволившие разработать новую методику синтеза управляющих автоматов и комплекс принципиально новых устройств управления с оригинальной структурной организацией, объем комбинационных схем в которых, в виде ПЗУ (ПЛМ) может быть снижен в сотни и тысячи раз в зависимости от сложности операторных схем алгоритмов управления.

п.4. «Разработка научных подходов, методов, алгоритмов и программ, обеспечивающих надежность, контроль и диагностику функционирования элементов устройств вычислительной техники и систем управления». По п.4 проведены теоретические исследования, позволившие разработать самоконтролируемые управляющие устройства с принципиально новыми методами динамического контроля и диагностики управляющих автоматов.

**На защиту выносятся** следующие положения:

1. Метод синтеза автоматов управления с выбором одного логического условия из полного множества входных переменных, позволяющий снизить сложность комбинационной схемы переходов от десятков раз до нескольких тысяч раз в зависимости от специфики и сложности алгоритмов управления спецпроцессором. Метод позволяет получить различные оригинальные варианты структурной организации автоматов управления, применимых в спецпроцессорах.

2. Метод синтеза автоматов управления со счетчиком в качестве памяти и разделенной комбинационной схемой переходов по значению логического условия. Метод обеспечивает дополнительное снижение оборудования комбинационных схем по сравнению с предложенным вариантом по п.1.

3. Предложен новый метод декомпозиции сложных УА, не требующий реализации независимыми УА декомпозируемых частей и создания ведущего автомата. По предложенному методу создается единый многорежимный УА с обращением к отдельным частям условно декомпозируемой ГСА с указанием кода частей. Предложенный метод позволяет дополнительно обеспечить снижение объема комбинационной схемы переходов.

4. Три самостоятельных метода синтеза самоконтролируемых автоматов, обеспечивающих безопасность функционирования информационно-управляющих систем, позволившие получить оригинальную структурную организацию автоматов с минимальными затратами встраиваемого оборудования.

5. Новый метод структурно-автоматного программирования, отличающийся наличием простой базовой ведущей программы, основанной на считывании информации из таблиц, предварительно рассчитанных по операторным схемам алгоритмов. Базовая программа не меняется при смене алгоритма управления спецпроцессором.

6. Новые структурные организации быстродействующих и высоконадежных спецпроцессоров для различных отраслей промышленности.

7. Новая методология проектирования управляющих автоматов спецпроцессоров сложных технических систем реального времени.

**Практическая значимость** работы состоит в том, что предложенные научные результаты и конструктивные методики синтеза позволяют создавать спецпроцессоры, как эффективные средства повышенной надежности для управления сложными техническими системами мехатроники, нефтехимической отрасли, криптографической защиты информации, навигационных систем подвижных транспортных средств и средств автоматизации промышленных предприятий. Результаты исследований могут найти применение в информационно-измерительных системах широкого профиля, а также в электронной промышленности при проектировании и производстве высоконадежных СБИС, в системах управления распределенными критическими технологиями и для систем контроля и диагностики бортовых автоматизированных комплексов летательных аппаратов.

**Достоверность результатов исследования** подтверждается корректным использованием теоретических методов вышеназванных научных дисциплин, патентной экспертизой, созданием и внедрением СПР в различных сферах промышленности.

**Реализация и внедрение** результатов работы осуществлена в задачах управления технологическими процессами добычи и переработки нефти (ОАО «Томская нефтегазовая компания» г.Томск), в задачах проектирования автоматизации предприятия (ЗАО «Энерпред» г.Иркутск), при разработке устройств управления режимами электроэнергетических систем (ОАО «НТЦ ФСК ЕЭС» - СибНИИЭ г.Новосибирск).

Результаты исследований нашли применение в учебном процессе, в лекционных курсах, курсовом и дипломном проектировании специальностей: мехатроника, управление техническими системами, автоматика и связь на ж.д. транспорте, информационные системы, безопасность информации и др. в ФГБОУ ВПО ИрГУПС, ИРН ИТУ (г.Иркутск), БрГУ (г.Братск), АГТА (Ангарск), ВУНЦ ВВС «ВВА» (г.Воронеж).

**Апробация результатов** исследований проводилась с 1996 по 2015г.г. на:

- конференции «Математическое моделирование систем», Институт динамики и теории систем управления СО РАН, 1996г.
- III и IV международных конференциях «Проблемы механики современных машин», Улан-Удэ, 2006, 2009г.г.
- VII Всероссийской научно-практической конференции аспирантов и молодых ученых «Молодежь и современные информационные технологии», Томск, 2009г.
- Научной международной школе РАН по проблемам машиноведения и технологических процессов, СПб, 2009г.
- VIII международной конференции «Системный анализ в проектировании и управлении», СПб, 2009г.
- XIV Байкальской Всероссийской конференции «Информационные и математические технологии в науке и управлении». – Иркутск: ИСЭМ СО РАН, 2009.
- XXII Международной конференции «Математические методы в технике и технологиях». – Псков, 2009.
- Ежегодной осенней сессии РАЕ с конференцией «Фундаментальные исследования».- Москва, 2010г.
- Международной конференции «Транспорт XXI века».- Екатеринбург, 2011г.
- Научно-практической конференции «Проблемы транспорта Восточной Сибири».- Иркутск: ИрГУПС, 2012г.
- Всероссийской конференции «Актуальные проблемы развития гражданской авиации России». – Иркутск, Москва, 2013.
- XII Всероссийском научном совещании по проблемам управления.- СПб: ИПУ РАН, 2014г.
- II Международной научно-практической конференции «Проблемы и достижения в науке и технике» (г.Омск). Секция №2. Информатика, вычислительная техника и управление (специальность 05.13.00).апрель 2015.
- Ежегодных научно-технических конференциях Иркутского госуниверситета путей сообщения, с 2006- 2015г.г.

**Публикации.** По материалам диссертации опубликовано 59 работ, в том числе 1 монография, 14 работ в журналах из списка ВАК, 4 патента на изобретения, 7 патентов на полезные модели, во Всероссийском фонде зарегистрировано 2 программы, 18 работ в едином авторстве. В работах с соавторами соискателю принадлежит от 45 до 70% результатов. Положения, составляющие научную новизну и выносимые на защиту, получены лично автором.

**Структура и объем работы.** Диссертационная работа состоит из введения, пяти глав, заключения, списка литературы и приложений. Работа содержит 244 страниц, включая текст 167 страниц, 105 рисунков, 50 таблиц, список литературы из 223 наименований, публикации автора. В приложении включены акты о внедрении и краткое описание патентов, листинги программ.

## ГЛАВА 1. УПРАВЛЯЮЩИЕ СПЕЦПРОЦЕССОРЫ СЛОЖНЫХ ТЕХНИЧЕСКИХ СИСТЕМ

В информационно-управляющих системах (ИУС) можно выделить три основные подсистемы: операционную (т.е. процессор), информационную (блоки памяти) и управляющую (взаимодействующие управляющие автоматы). Вычислительные операции в операционном процессоре производятся с аналоговыми величинами или с кодами разрядностью  $\leq 16$ . Причем количество операций вычислительного характера ограничено.

Управляющая подсистема, как правило, задается через граф-схему (ГСА) одного алгоритма или ограниченного комплекса хорошо совместимых алгоритмов управления. Причем эти ГСА в процессе проектирования ИУС многократно корректируются в соответствии с изменением структурной организации ИУС. Поэтому, рассматривая задачи разработки методов анализа и синтеза УА для спецпроцессоров, необходимо их тесно увязывать с проектированием структурной организации ИУС.

### 1.1. Концепции развития информационно-управляющих систем реального времени

В последние десятилетия интенсивно развивается научное направление по созданию ИУС с элементами искусственного интеллекта (нечеткая логика, генетические алгоритмы, нейронные сети). При этом вычислительные операции также не являются основными.

Наиболее развито направление нечетких систем (НС). Для НС необходимо создание базы знаний с нечеткими продукционными правилами и алгоритмов нечеткого вывода. Субъективность выбора функций принадлежности и набора постулируемых правил при невозможности полного учета реальной действительности определило тот факт, что НС подтверждают свою эффективность в задачах управления технологическими процессами главным образом как своего рода дополнение к классическим ПИД регуляторам. Но при

этом необходима последующая настройка по реальному объекту, путем имитационного моделирования и с помощью генетических алгоритмов.

Нейронные сети и нейронные контроллеры для обеспечения эффективности требуют использования слишком больших вычислительных и временных ресурсов, что затрудняет их использование в информационно - управляющих системах реального времени.

Как показывает практика, использование интеллектуальных технологий в реально работающих системах управления технологическими процессами требует объединения в единой структуре дискретных адаптивных регуляторов, нейросистем и подсистем для настройки нейронных сетей на основе генетических алгоритмов с программным управлением [1].

Несмотря на то, что многие авторы относят системы с нечеткой логикой и нейросетями к неалгоритмическим способам вычислений, фактически эти технологии также основаны на использовании алгоритмов. В работе [2] нейронные сети определены как особый тип конечных автоматов, а генетические алгоритмы в принципе не могут быть осуществлены без поддержки сложными автоматами управления или управляющими программами.

Технической базой реализации всех типов ИУС являются интегральные схемы различной степени сложности. Какие либо специальные элементы нейронной техники не разработаны.

В настоящее время выпускаются серии однокристальных микропроцессоров, микроконтроллеров, программируемых логических матриц (ПЛИМ), программируемых логических интегральных схем (ПЛИС), микропроцессорные комплекты с использованием различных технологий ТТЛ, ТТЛШ, МДП, ЭСЛ, И<sup>2</sup>Л, КМОП и др., удовлетворяющих потребности разнообразных средств автоматики и вычислительной техники. Причем ежегодно появляются СБИС с более высоким быстродействием и расширенными сервисными операциями. С целью завоевания рынка производители СБИС придают им свойства «универсальности», расширяя логические и функциональные возможности.

Сфера применения серийных средств вычислительной техники и дискретной автоматики широка и разнообразна, но для ряда практических применений (особенно для техники оборонного комплекса) требуется разработка нестандартного оборудования, объединяющего типовые БИС и СБИС в единый комплекс.

Казалось бы, с появлением СБИС необходимость в интегральных схемах малой и средней интеграции (МИС, СИС) должна уменьшаться. На практике наблюдается противоположная тенденция - с расширением сферы применения микроконтроллеров, БИС памяти и др. увеличивается также потребность в СИС и МИС. Эти элементы совершенно необходимы для создания специализированных интерфейсов, согласователей уровней и других специальных средств для расширения функций микроконтроллеров ИУС. В ряде случаев более высокий уровень надежности ИУС может быть достигнут, если вместо одной СБИС (как правило, это ПЛИС или «универсальный» однокристалльный контроллер) использовать несколько СИС при «оптимальной» декомпозиции алгоритма.

За счет реализации алгоритма по частям или специальной структурной организации подсистем функциональную задачу можно решить на БИС и расширенном комплекте СИС. Такое решение обеспечивает более высокий уровень надежности, т.к. каждый вентиль СБИС потребляет от 0.1 до 4 мкВт мощности, следовательно, при числе вентилях в СБИС более 40млн, рассеиваемая мощность на одном кристалле может составлять несколько ватт и использование таких СБИС невозможно без внешнего охлаждения.

Развитая на сегодня интегральная элементная база средств вычислительной техники и управления дает возможность создавать ИУС, ориентированные на реализацию как основных алгоритмов, так и комплекса сервисных алгоритмов. С учетом включения микропроцессоров и микроконтроллеров, блоков памяти, элементов многофункциональной логики, в целом и сами ИУС как комплекс средств обработки информации с выработкой команд управления, можно отнести к числу сложных технических систем. И потому ИУС как сложная система должна проектироваться по новым методикам, не ориентированных изначально



на известные структурные организации управляющей подсистемы с базовыми микроконтроллерами или параллельным комплексом микропроцессоров.

На каждом этапе развития элементной базы и средств для реализации систем управления и специальной вычислительной техники требуются новые подходы и методы анализа и синтеза систем управления.

Для ИУС реального времени, к которым относятся в первую очередь спецпроцессоры летательных аппаратов, подвижных транспортных средств, технологических комплексов со сверхбыстродействующим темпом обработки информации, специальных оборонных комплексов и др., ориентация идет на аппаратные средства реализации. Аппаратные специализированные ИУС в своем развитии прошли несколько этапов:

1. Схемы из функциональных элементов малой интеграции.
2. Таблично-алгоритмическая структура с использованием элементов средней интеграции и БИС.
3. Специализированные функциональные преобразователи с ограниченным использованием микропроцессоров с управляющей подсистемой на базе СБИС.
4. Многопроцессорные быстродействующие системы со сложными аппаратно-программными системами управления на основе параллельных процессорных подсистем.
5. Комплексы в виде систем взаимосвязанных спецпроцессоров со встроенными средствами контроля и диагностики. Создание таких систем требует решения задач системного анализа и синтеза с целью оптимальной декомпозиции на локально независимые подсистемы, оптимизации процессов обмена информации и организации многоуровневых подсистем управления.

Каждому из этапов 1-5 соответствуют свои теоретические методы синтеза управляющей подсистемы:

1. Прикладная теория функций алгебры логики с минимизацией числа логических операций или вхождения элементов логики в конъюнкции и конкатенации;

2. Минимизация систем булевых функций большой размерности на ПЛМ; Реализация ИУС преимущественно аппаратными методами.

3. Модификация методов анализа и синтеза ИУС, предложенных на предыдущем этапе не только с аппаратной, но и с программной реализацией.

4. Решение задач системного уровня управления, причем программные методы управления превалируют над аппаратными.

5. Системный анализ и синтез СПР с расширением задач 3-4.

Современный этап развития специализированных ИУС реального времени отличается переходом к этапу 3, но на новом технологическом уровне, с использованием более сложных самоконтролируемых управляющих автоматов. В качестве элементов ИУС могут использоваться однокристалльные микроконтроллеры с развитыми функциональными возможностями, а в качестве подсистем управления используются взаимодействующие декомпозированные автоматы на ПЛИС.

В этом случае методы синтеза должны быть ориентированы в первую очередь на оптимизацию структурной организации ИУС со специальными УА.

Прямое решение задачи повышения быстродействия для рассматриваемых ИУС реального времени методом максимального распараллеливания становится не перспективным ввиду сложности организации параллельной процедуры обмена информацией. Повышение быстродействия достигается ценой больших затрат оборудования, а следовательно сопровождается снижением уровня надежности ИУС. Кроме того для управляющих спецпроцессоров алгоритмически существует определенный предел повышения быстродействия таким путем, т.к. имеются ограничения на параллельное формирование команд управления без проверки логических условий. В таких системах, несмотря на распараллеливание, преимущественное использование программных методов управляющей подсистемы может снизить быстродействие ИУС на порядок, по сравнению с аппаратной реализацией СПР.

## 1.2. Проектирование управляющих спецпроцессоров

При создании сложных технических систем (СТС) качественно новым уровнем проектирования является использование информационных технологий (Virtual Product Development – VPD). Например, компания MSC Software предлагает лицензированные программные модули (MSC.MasterKey) для инженерного анализа проектов расчета механических систем (главным образом методом конечных элементов), моделирование их в динамике совместно с работой подсистем управления. Также решаются задачи подготовки технической документации на этапах внедрения в производство [3]. Главное применение VPD нашли в авиационной промышленности, судостроении, теплотехнике и др.

Системы VPD для проектирования средств автоматизации и вычислительной техники также имеются. Традиционно эти системы называют САПР – системы автоматизации проектирования. Наиболее развитые из них - это VPD с использованием программных комплексов VHDL [4]. Обзор и анализ средств САПР для дискретных устройств приведен в работе [5].

Технологии VPD для анализа и моделирования ИУС широко применяются в САПР для ИУС с известной структурной организацией. В работе [6] подробно описано представление на VHDL всех основных базовых блоков ИУС: регистры памяти, дешифраторы, мультиплексоры, ОЗУ и ПЗУ, арифметико-логические устройства. Более того даны рекомендации по реализации на ПЛИС (с описанием на VHDL) комплекса этих средств, т.е. законченного изделия в виде микропроцессора реализуемого комплексом БИС.

Однако речь идет об ИУС с известной (заданной) структурной организацией.

При проектировании специализированных ИУС обычно неизвестно, какие блоки войдут в его структуру, сколько блоков того или иного типа, нужно использовать микроконтроллер или нет, как будет организована передача информации между блоками и управление процессом.

В ряде случаев новая структура ИУС находится эвристически, получается оригинальной и патентуется. Поэтому средства САПР на уровне структурного проектирования со всеми системами VPD могут использоваться для анализа уже спроектированной ИУС, т.е. созданные на сегодня VPD в задачах структурного проектирования могут выполнять функции лишь вспомогательного характера.

Аппаратные средства представляют собой взаимосвязанный набор операционных и управляющих блоков, которому присущи основные свойства технических систем по А.Н. Панченкову [7]:

1. Для систем определено пространство состояний  $E$  и параметрическое пространство  $M$ , в котором задано поведение системы;
2. Число элементов  $E$  не менее двух;
3. Система обладает свойством эмерджентности (целостности), т.е. свойства целого (системы) не сводятся к суммарным свойствам частей (подсистем).

Более того, и сами ИУС также можно отнести к сложным техническим системам не только из-за наличия большого числа элементов и связей между ними, но из-за наличия обратных связей, определяющих «нелинейное» поведение системы во времени.

Сложность дискретных систем некоторые авторы оценивают числом логических вентилях [8]. Каких-либо обобщенных (в какой-то мере объективных) критериев оценки сложности ИУС нет, т.к. ИУС невозможно оценивать без учета проблемной направленности и области применения. Так, например, собственно информационные системы оцениваются в первую очередь объемом хранимой информации с учетом правил нахождения информации в базах данных [9].

Управляющие системы (системы автоматического регулирования) оцениваются степенью аппроксимирующих полиномов передаточных функций и числом контуров обратной связи [10].

Сложность диагностических комплексов проверки качества реализации технологических процессов оценивается количеством алгоритмов и сложностью операторных схем алгоритмов контроля и диагностики [11].

Стремительное развитие вычислительной техники и широкое применение аппарата двоичной логики привело к тому, что язык описания дискретных устройств и методы оптимизации представляются не иначе как применение аппарата булевой алгебры.

Прикладное развитие математической логики для синтеза дискретных систем заложено в трудах В.Н. Шестакова, И.И. Жегалкина, П.С. Порецкого, А.А. Лупанова, М.А. Гаврилова и развито А.А. Ляпуновым, С.В. Яблонским, В.М. Глушковым, Ю.И. Журавлевым, Д.А. Поспеловым, А.Д. Закревским, В.И. Варшавским, В.А. Горбатовым, А.А. Таль, П.П. Пархоменко, Е.С. Согомояном, А.В. Каляевым, И.В. Прангишвили, А.Н. Мелиховым, В.Г. Лазаревым, С.И. Барановым, Э.А. Якубайтисом, Ю.Ф. Мухопадом, Е.И. Пупыревым, А.А. Шалыто, В.В. и Вл.В. Сапожниковыми и др.

Иностранные авторы, внесшие заметный вклад в развитие теории и практики синтеза дискретных устройств это А.Гилл, Э.Мак-Класки, М. Фистер, Э.Мур, Дж.Мили, С.Ли.,С.К. Клини, К.Шеннон, Дж. Хартманис, Р. Стирнс, В.Квайн, С. Петрик, Х.А. Христов, С. Колдуэлл, Д. Ауфенкамп и др.

В работах по теории автоматов ведущих ученых утверждаются положения о том, что любому алгоритму построения минимальной схемы дискретных устройств присуща высокая трудоемкость, т.к. она обусловлена принципиально сверхбольшим числом вариантов при постановке задачи в общем виде.

Кроме алгоритмической сложности, задача проектирования таких ИУС осложняется еще наличием многономенклатурного базиса интегральной схемотехники.

Поэтому практически невозможно пытаться решить задачу структурного синтеза ИУС и определить алгоритмы управления для УА в общем виде. Нужно по возможности четко определить класс решаемых задач (приложений), имеющих совместимое алгоритмическое описание и удачно подобрать структурную модель ИУС, включающую минимальный, но достаточный набор составляющих подсистем (блоков) и их связей.

Известны несколько методик проектирования ИУС, авторы которых считают их не только обобщенными, но даже универсальными.

В работах [12-13] В.А. Чечкин предлагает для проектирования ИУС задать набор функциональных преобразователей информации  $F_1F_2\dots F_m$  каждый из которых реализует некий набор операций  $f_1f_2\dots f_k$ . Тогда для заданного комплекса алгоритмов ГСА1, ГСА2,...ГСАН можно найти вероятность обращения к каждому функциональному преобразователю для каждой операции  $f_1f_2\dots f_k$ . Комплекс функциональных преобразователей с наибольшей частотой обращения проектируется в виде универсального процессора, а все остальные  $F_i \in \{F\}$  составляют основу спецпроцессора. Сама постановка задачи выделения основного ядра ИУС является корректной, однако изначально как раз набор  $F_i \in \{F\}$  неизвестен, а именно это и составляет основу структурной организации ИУС. Автор развивает эту методику с использованием ультра операторов и конечной цели отображения входного множества  $\{X\}$  в выходное  $\{Y\}$ . Понятие ультра оператора основано на понятии ультра отображения. Как видно из [13] в отличие от обычного функционального отображения множеств, здесь подчеркивается информационный аспект, т.е. отображаются не сами множества, а информация об элементах данного множества. В частном случае понятие «ультра» от оператора отображения можно опустить, тогда это будет просто функциональный преобразователь.

Далее, исходя из задачи и функционального алгоритма ИУС, рекомендуется построить сеть Петри с выполнением работ в виде взаимосвязанной последовательности ультраоператоров и определенными дополнительно логическими условиями.

Исследования на сети Петри уже известны – это определение безопасности, живучести, достижимости и наличия тупиков [14,15]. В работе [13] рассматриваются только информационные преобразования на сети Петри без решения вопроса, каким конкретно блокам ИУС нужно «приписывать» функции того или иного ультра оператора и сколько таких блоков в ИУС. Предлагаемый

метод фактически основан на неявном задании структуры ИУС с отображением динамики информационных преобразований сетью Петри.

Аналогичный подход для частично упорядоченных систем, для которых структурная организация определяется В.А. Горбатовым [16] «в виде семантической модели, представленной сетью элементов моделей поведения с взаимно-однозначным соответствием между элементами носителей модели поведения». Это соответствие изначально известно и задано в виде графа или матрицы связности. Элементы моделей поведения в общем случае предполагают отнесение их к классу как полностью аналитически определенных функциональных преобразователей (ФП), так и некоторых устройств отображения множества  $\{X\}$  в  $\{Y\}$ . В качестве практического примера такой системы В.А. Горбатов приводит структуру с двумя вычислителями и восемью информационными массивами. За счет последовательности преобразований матрицы связности и введения в структуру системы коммутатора информации удалось представить такую структуру в виде двух вычислителей, коммутатора с двумя массивами информации и определенной процедурой временного управления.

При этом используются сложные наукоемкие методы преобразования бинарных матриц через отношения упорядоченности (решетки), определяется связность элементов, применяется аппарат дифференцирования булевых функций, частотного разложения и гомоморфизма моделей и др. Синтез по существу сводится к определению количественных показателей ИУС с известной структурной организацией, но несколько расширенным набором функциональных преобразователей (число вычислителей, каналов связей, количество блоков памяти и др.)

Более упрощенный метод определения структуры ИУС связан с непосредственным определением набора функциональных преобразователей (вычислительных операторов) по заданному алгоритму. Этот подход широко используется научной школой В.Б. Смолова [17]. Метод чрезвычайно прост и очевиден для некоторых алгоритмов. Например, для алгоритма Волдера нужно

находить  $\rho = \sqrt{x^2 + y^2}$  и  $\varphi = \arctgy/x$ . Результат достигается на  $n$ -ом шаге через преобразования

$$\begin{aligned}x_{i+1} &= x_{i-1} + y_{i-1}2^{-i} \\y_{i+1} &= y_{i-1} - x_{i-1}2^{-i} \\ \varphi_{i+1} &= \varphi_i - \arctg2^{-i}\end{aligned}\tag{1.1}$$

Как очевидно для непосредственной реализации зависимостей (1.1) в спецпроцессоре потребуется наличие трех сумматоров, двух регистров сдвига и устройства хранения констант  $-\arctg2^{-i}$ .

Для более сложных случаев составляется ярусно-параллельная сеть в виде графа с числом вершин равном числу входных переменных и одной выходной вершиной. Тогда для определения структурной схемы ИУС необходимо задать ограничения (число предельных вершин в ярусе), определить принадлежность каждого ФП к определенному проектировщиком набору блоков, решить задачу оптимизации расписания, т.е. определить необходимые условия для построения операторной схемы алгоритма (ОСА) управления.

При современном подходе построения ИУС со встроенными микропроцессорами согласно [18] задача структурного синтеза ставится еще более упрощенно, т.к. необходимо выполнить всего несколько этапов представленных в виде блок-схемы (рис.1.1.а),

где  $A_1$  –разработка алгоритма;  $A_2$  – выбор типа микропроцессора;

$A_3$  –разработка структуры аппаратных средств и программного обеспечения;

$A_4$  –совместная отладка аппаратных и программных средств в реальном времени;  $A_5$  –передача в эксплуатацию;

$\alpha = 1$ , если ИУС соответствуют техническому заданию.

Для разработки аппаратных средств даются рекомендации [6]: «максимальное использование аппаратных средств упрощает разработку и обеспечивает высокое быстродействие контроллера в целом, но сопровождается, как правило, увеличением стоимости и потребляемой мощности».

Аналогична технология проектирования постпроцессоров для управления гибкими производственными системами [19] и проблемно-ориентированными



микропроцессорными системами управления производством радиоэлектронных компонент [20].

Столь обобщенные рекомендации не позволяют наметить даже общего подхода к синтезу структурной организации ИУС.

Для быстродействующих ИУС аппаратное обеспечение микропроцессорных систем связано с подключением к контроллеру нескольких табличных ФП (названных расширителями функций).

Как видно, во всех вариантах этих методик задается базовая модель или микроконтроллер в качестве операционного устройства. Затем определяется необходимость и возможность аппаратного расширения базовой модели за счет включения табличных ФП, а далее, как обычно, переходят к разработке и формализации алгоритма управления и синтезу УА.

Известна методика синтеза структурной организации ИУС, начинающаяся с анализа общей постановки задачи преобразования информации (рис.1.1.б). Такая методика предложена в работе [21], которая состоит из нескольких этапов:

$A_1$  – представление ИУС в виде единого ФП, реализующего функцию многих переменных.

$A_2$  – алгоритмическое разложение оператора  $A_1$  на комплекс операторов с меньшим числом переменных на входе (декомпозиция).

$A_3$  – реализация ИУС таблично-алгоритмическими структурами.

$A_4$  – моделирование ИУС с целью оценки точности преобразования информации.

$A_5$  – разработка ярусно-параллельной схемы с уровнем разложения  $A_3$ .

$A_6$  – решение задачи оптимального расписания с минимальным объемом памяти ИУС.

$A_7$  – синтез управляющего автомата и переход к функциональному и электрическому проектированию.

Один из методов алгоритмического разложения приведен в работе [21], а результаты таблично-алгоритмической реализации функций четырех переменных

с точностью вычисления определяемой 12 разрядами с фиксированной запятой приведены в работе [22].

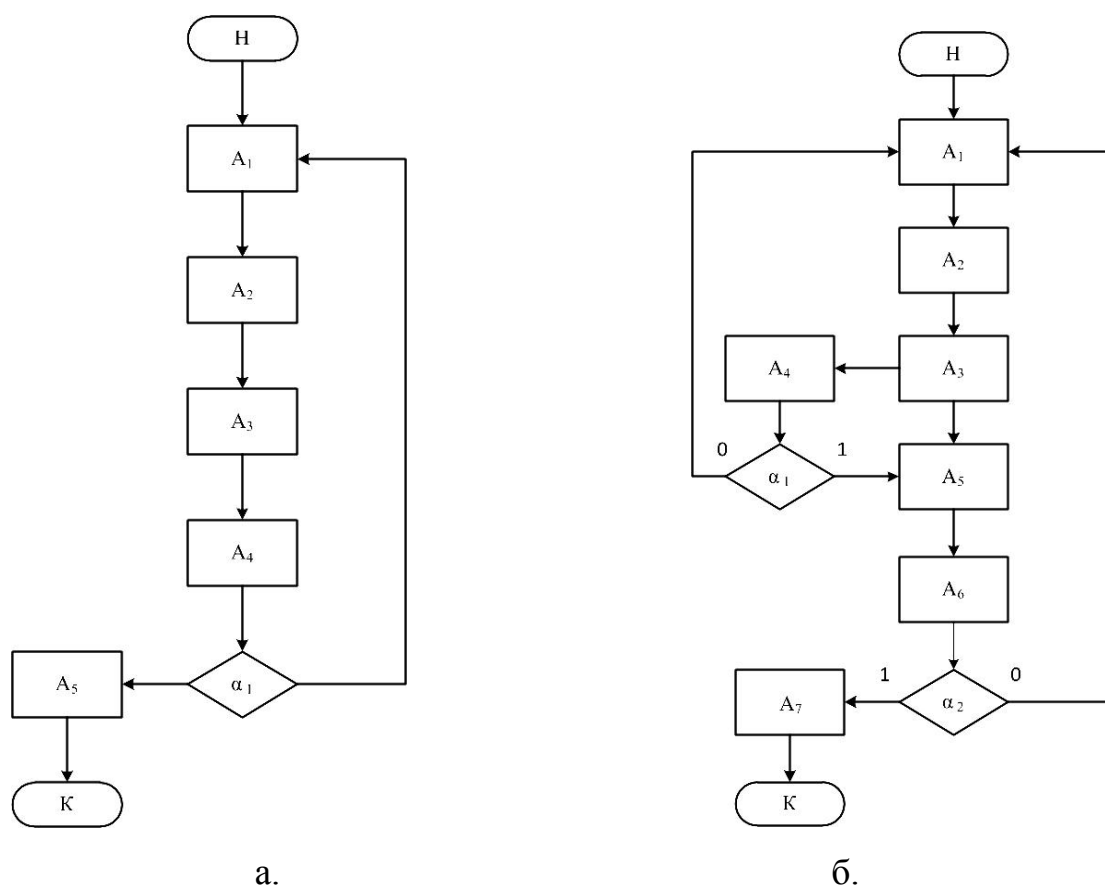


Рисунок 1.1 - Процедура синтеза ИУС:

а – с микропроцессорами, б – с АЦП

Оператор  $A_1$  реализуется как ФП, отображающий множество 48 двоичных переменных в 12 выходных. Удачное алгоритмическое разложение  $A_2$  удалось найти за счет выделения основной переменной и нелинейного квантования двух других (значение крена и тангажа для летательных аппаратов), изменяющихся в ограниченном диапазоне значений и не требующих высокой точности отсчетов при малых значениях величин.

Функция 48 переменных реализуется таблично-алгоритмической структурой на трех ПЗУ от 24-х переменных. Т.е. объем ПЗУ для ФП с  $V = m^{2^n} = 12 \cdot 2^{48}$  снижен до  $W = m^{2^n} = 3 \cdot 12 \cdot 2^{12}$ , т.е. почти в  $(1/3) \cdot 2^{36}$  раз. Методика результативна и нашла применение при проектировании аналого-цифровых ИУС,

однако наиболее эффективно ее применение лишь для задач вычислительного характера с ограниченной точностью.

### 1.3. Структурные модели информационно-управляющих систем

При определении структурной схемы ИУС используются три типа моделей:

- одноблочная модель черного ящика [22-24] – рисунок 1.2,
- двухблочная модель В.М. Глушкова [25] – рисунок 1.3; 1.4,
- пятикомпонентная модель Ю.Ф. Мухопода [26-29] – рисунок 1.5.

В модели черного ящика вся ИУС представляет собой единый блок с  $m$  входами и  $n$  выходами. Несмотря на кажущуюся простоту и наивысшую «укрупненность», эта модель полезна на самом первом этапе анализа всей задачи ИУС и разработки укрупненного алгоритма функционирования ИУС. Разработка детального алгоритма функционирования заканчивается представлением его в операторной форме.

Операторная схема алгоритма (ОСА) может иметь вид граф-схемы алгоритма (ГСА), логической (ЛСА), матричной (МСА), регулярной (РСА) и табличной схем алгоритма (ТСА). Правила перехода от одной формы ОСА к другой определены в работе [26]. Далее будем использовать преимущественно ГСА и ЛСА.

Анализ ИУС с моделью черного ящика позволяет определить допустимую точность обработки информации, а следовательно, и полный набор входных и выходных аргументов ( $m, n$ ).

Структурная модель В.М. Глушкова (рисунок 1.3) представляется в виде двух блоков: ОУ – операционного устройства и УА – управляющего автомата.

ОУ в модели В.М. Глушкова считается заданным или спроектированным по какой-либо известной методике с использованием выбранной элементной базы в виде БИС и СБИС. Причем уровень детализации ОУ должен быть таким, чтобы на его структуре могла быть полностью реализована функциональная задача ИУС и определен полный набор микроопераций управления каждым из блоков ОУ. Тогда задача проектирования ИУС сведется к разработке алгоритма управления

ОУ в форме ОСА и проектировании УА или комплекса УА, выполняющего роль управляющей подсистемы.

Модель В.М. Глушкова не может быть первичной, по которой следует определять структурную схему ИУС, т.к. она предполагает наличие предварительного этапа структурного синтеза ОУ. По существу, модель В.М. Глушкова ориентирована на проведение абстрактного и структурного синтеза УА.

При проектировании аналогово-цифровых устройств управления технологическими процессами [1,30] используется структурная модель, состоящая из пяти блоков (рисунок 1.4). Здесь ОУ и УА имеют такую же интерпретацию, что и в модели В.М. Глушкова.

ИМ – исполнительный механизм (рисунок 1.4), т.е. по существу преобразователь формы представления команд управления; Д– датчики информации.

Для аналого-цифровых систем входные и выходные сигналы Х, У представлены в виде напряжений в заданном диапазоне (например, от -5В до +5В). Команды от УА требуют значительного энергетического усиления, если в качестве исполнительных устройств используются электродвигатели, турбомеханизмы, схваты роботов и др.

В структурной модели В.М. Глушкова команды, как и значения разрядов входной и выходной информации, представлены одинаковыми уровнями «0» и «1», т.к. модель ориентирована на анализ дискретных процессов. Поэтому, не смотря на наличие устройств изменения уровня сигналов, для задач структурного проектирования ИУС, модель рисунка 1.4 - это по существу та же модель В.М. Глушкова.



Рисунок 1.2 - Модель черного ящика

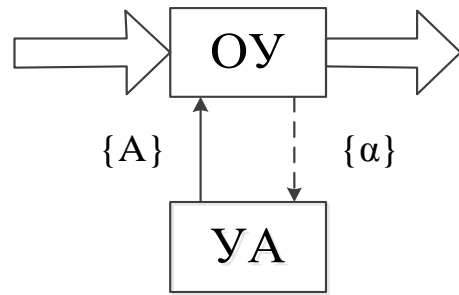


Рисунок 1.3 - Двухблочная модель

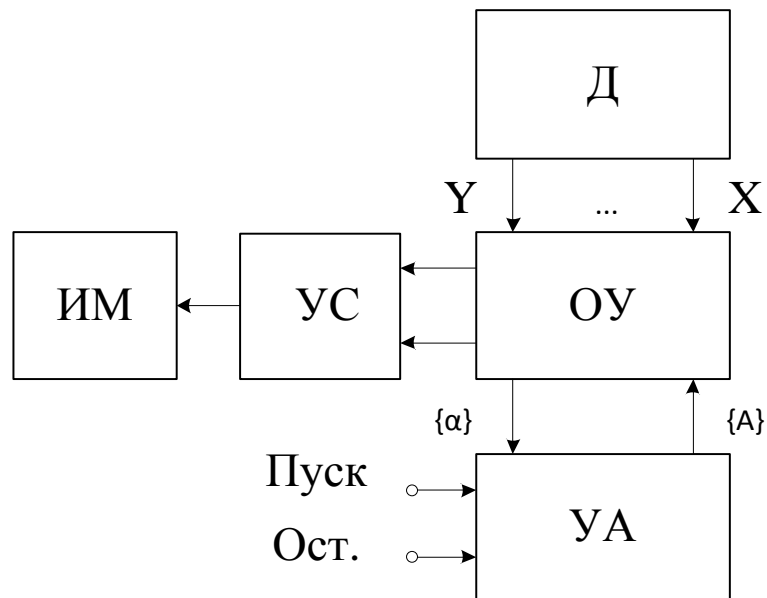


Рисунок 1.4 - Модель аналого-цифровой системы управления

Более детальна и конкретна структурная модель Ю.Ф. Мухопода, которая представляет собой полный пяти вершинный граф (рисунок 1.5), причем вершины графа интерпретируются следующим образом:

- Ф – функциональная подсистема;
- И – информационная подсистема;
- Л – логическая подсистема;
- А – адресная подсистема;
- У – управляющая подсистема ИУС.

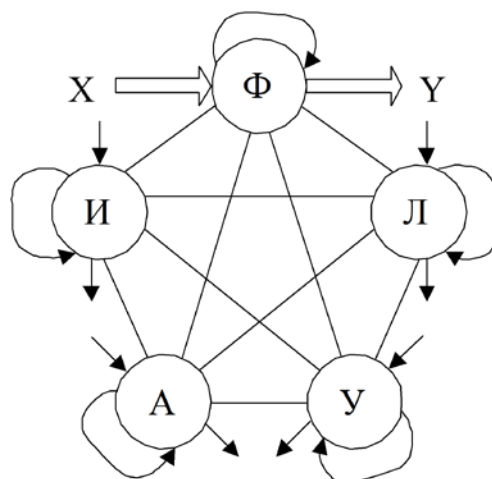


Рисунок 1.5 - Пятиблочная модель спецпроцессора

Речь идет именно о подсистемах, а не блоках ИУС, т.к. каждая из подсистем (Ф, И, Л, А, У) может состоять из набора собственных блоков, объединенных в определенную структуру, и в общем случае со своим микропрограммным автоматом управления. Хотя в частном случае каждая из подсистем может быть представлена одним единым блоком (одной БИС или СБИС) с управлением от единого УА.

Естественно, возникает вопрос, о каком управляющем автомате (УА) может идти речь для управляющей подсистемы (У), если сама эта подсистема выполняет управляющие функции.

Для общего случая речь идет о создании иерархической системы управления в виде УА для каждой из подсистем и верхнего автомата (с программной или микропрограммной организацией) управления. Для частного случая (более простая ИУС с единым УА для всех подсистем) управляющее устройство самого УА есть блок синхронизации.

Связи между вершинами в структурной модели ФИЛАУ представляют собой каналы передачи информации. Они могут быть полными или «усеченными» (т.е. не полными) в зависимости от функционального алгоритма и структуры ИУС.

Сами связи могут быть организованы тремя основными способами:

- связь всех подсистем каждая с каждой по независимым каналам;
- выделяются информационные, адресные и управляющие каналы;
- связи по общей информационной шине.

Определить, какова структура каждой из подсистем (Ф, И, Л, А, У) и каков принцип связи между подсистемами – это и есть задача структурного синтеза.

Очевидно, что пятикомпонентная модель (Ф, И, Л, А, У) обладает существенным преимуществом по сравнению с двумя другими моделями. Эти преимущества определяются тем, что на примере алгоритма управления с нанесенными информационными связями (рисунок 1.6) в работе [26] доказано, что для реализации любого алгоритма необходимо и достаточно именно пяти подсистем Ф, И, Л, А, У.

Действительно, как видно из рисунка 1.6, последовательность состояний 0, 1, 2, ..., 18 определяет действие управляющей подсистемы ИУС. Пунктирные связи между  $A_i \in \{A\}$  определяют передачу данных (адресная подсистема) в ИУС. Аппаратная реализация всех  $A_i$  ( $i = \overline{1,18}$  для примера) определяет Ф подсистему, а реализация множества  $\alpha_i$  ( $i = \overline{1,6}$ ) определяет Л-подсистему. Необходимость в И-подсистеме определяется тем, что потребуется хранение информации при передаче ее от  $A_i$  к  $A_j$  в связи с разным быстродействием функциональных преобразователей, реализующих ту или иную операцию  $A_k \in \{A\}$ . Например, операции пересылки информации от  $A_i$  к  $A_j$  несопоставимы с временем реализации операций умножения и деления.

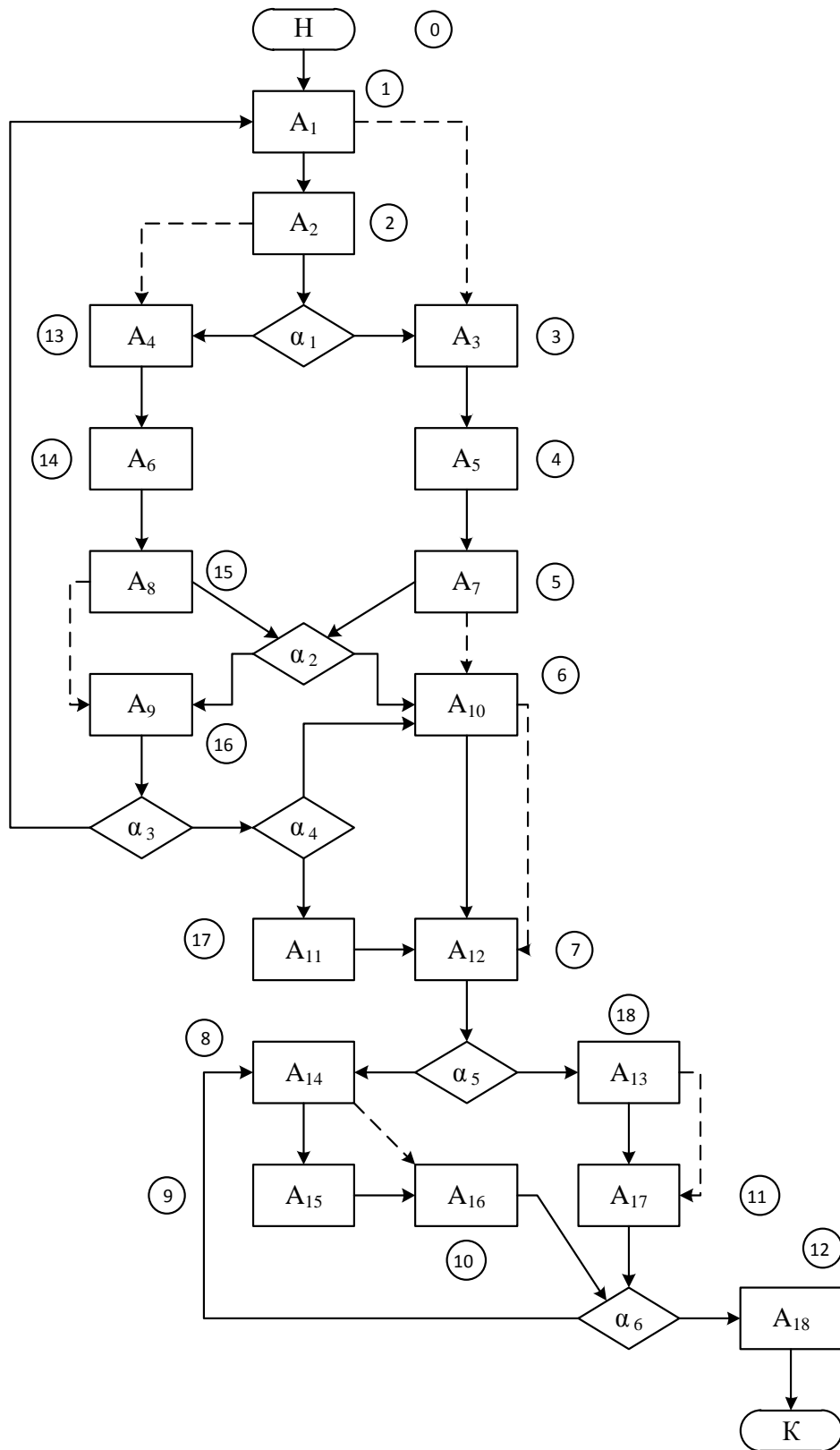


Рисунок 1.6. - ГСА с информационными связями



#### 1.4. Методология синтеза управляющих спецпроцессоров

Основной моделью для структурного проектирования СПР следует считать пятикомпонентную модель ФИЛАУ. Обобщенная структурная организация СПР в виде взаимосвязанных подсистем Ф, И, Л, А, У позволяет определить последовательность перехода к синтезу от одной подсистемы к другой. При заданном функциональном алгоритме ИУС структурный синтез производится в следующей последовательности:

1. При сложности процедур преобразования информации и управления операционными блоками при проектировании СПР для СТС необходимо произвести правильную группировку базовых процедур обработки информации и управления на независимые проблемно-ориентированные задачи. Каждая из таких задач и будет в последующем реализовываться одним или несколькими спецпроцессорами, т.е. ИУС.

2. Через анализ комплекса алгоритмических процедур преобразования информации и управления ИУС определяется базовый набор операций, которые реализуются комплексом блоков Ф подсистемы. Определение состава блоков базируется как на формализованных методиках, так и на эвристическом доопределении. При современной функциональной сложности СБИС и расширенных функциях микроконтроллеров и ПЛИС в ряде случаев функциональная подсистема может быть сведена к одному микроконтроллеру, одной ПЛИС или комплексу ПЛИС с микроконтроллером.

3. Определяется форма представления информации в СПР, объем памяти и структурная организация информационной подсистемы (взаимодействие ОЗУ, ПЗУ, вынесенных регистров памяти и др.). Математический подход этапа 2 базируется на теории информации и методе синтеза коммутационных схем.

4. Определяются способы адресации и методы преобразования адресной информации при функционировании СПР. Решаются задачи минимизации объема

переносимой во времени информации и определения способов передачи информации через минимизацию числа каналов и связей.

5. Определяется полный набор логических условий как внешних датчиков, так и определяемых условий в самой Л-подсистеме, т.е. последовательно определяется структура логической подсистемы (Л). Решается задача использования простых датчиков с накоплением информации во времени или использование интеллектуальных датчиков, преобразующих информацию в процессе измерений.

6. На основе полностью определенной структуры основных операционных подсистем Ф, И, Л, А разрабатывается управляющий алгоритм и осуществляется переход к абстрактному и структурному синтезу У - подсистемы.

Структурная модель ФИЛАУ изначально ориентирована на решение задачи синтеза СПР на основе системного подхода, базирующегося на формально определенных принципах (законах) технической кибернетики и формально определенных этапах системного анализа.

Системный анализ - это комплекс методов (методология), позволяющих решить следующие задачи:

1. проанализировать задачу ИУС и определить (разработать) функциональный алгоритм;
2. осуществить декомпозицию всей ИУС на независимые подсистемы;
3. определить взаимодействие подсистем в пространстве и во времени.

Задачи 2, 3 решаются на основе пятикомпонентной структурной модели с последующим моделированием подсистем Ф, И, Л, А, У.

4. определить критерий эффективности системы. Критерий эффективности не может быть сформулирован в общем виде, т.к. зависит от назначения ИУС, области применения и выдвинутых требований по быстродействию, безопасности и др. Применительно к микроэлектронным ИУС комплексный критерий обычно включает, кроме оценки быстродействия и точности реализации информационных преобразований, степень безотказности и контролепригодности. Очевидно, что формализованное определение критерия

эффективности является самостоятельной задачей при проектировании и не относится к задачам исследований СПР или УА.

5. определить взаимодействие ИУС с внешней средой, которое достигается моделированием всей системы как объекта управления с воздействием возмущающих факторов реальной среды функционирования (электромагнитные шумы, вибрации, температурные воздействия и др.).

Представление ИУС в виде пяти функционально ориентированных подсистем упрощает не только всю процедуру синтеза самих подсистем, но и дает возможность более четкой постановки задачи разработки управляющего алгоритма ИУС с последующим синтезом минимизированного УА.

Для ряда практических приложений пятикомпонентная модель существенно облегчает процедуру проектирования и позволяет получать оригинальные структурные организации преобразователей информации.

### 1.5 Структурная организация и классификация управляющих автоматов.

Управляющие автоматы как дискретные устройства имеют строгое (формальное) определение. По определению [5,16,22,25,26] автомат - это устройство со входным алфавитом  $\{\alpha\}$ , выходным алфавитом  $\{Y\}$ , множеством внутренних состояний  $\{a\}$ , причем переход из состояния  $a(t)$  в  $a(t+1)$  осуществляется по условию  $a(t+1) = F_1(\alpha_1 \alpha_2 \dots \alpha_q x_1 x_2 \dots x_m)$ . Здесь  $\{x\}$ -код состояния  $a(t)$ . В самых ранних [23,31,32] работах структурная организация УА определялась в виде двух основных блоков (рисунок 1.7). Здесь ОУ – операционное устройство, ЛЗ – линия задержки.

Автоматы разделяются на конечные при числе состояний  $N \leq 256$  и бесконечные при  $N \gg 256$ . Если рассматривать регистр памяти с двоичным позиционным кодом (ДПК) как автомат, то при  $m=64$ , значение  $N = 2^{64} \approx 10^{22}$ .

В диссертационной работе исследуются только конечные управляющие автоматы.

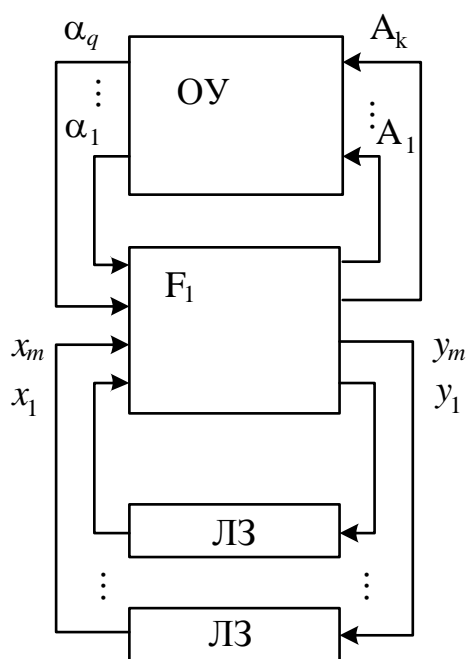


Рисунок 1.7- Управляющий автомат

Переходы из состояния  $a(t)$  в  $a(t+1)$  могут выполняться как детерминированно по функциям  $F_1$ , так и вероятностно, т.е. из одного состояния  $a(t)$  может осуществляться переход в  $p_1 a_1, \dots, p_k a_k$ . При этом сумма вероятностей  $p_1 + \dots + p_k = 1$ . Управляющие автоматы (УА) не относятся к классу вероятностных.

Автоматы называются инициальными, если они имеют всегда одно и то же начальное состояние, в которое осуществляется возврат как в заключительное состояние. В не инициальных автоматах начальное и заключительное состояния могут задаваться извне, но такие автоматы в диссертации не рассматриваются.

В литературе нет специального разделения автоматов на управляющие и микропрограммные. Чаще всего речь идет о микропрограммных автоматах (МПА). Как определено в работе [33] «универсальные микропрограммные автоматы представляют собой ПЗУ (ОЗУ), в котором хранится упорядоченная совокупность кодов микрокоманд. Особенностью универсального МПА является предопределённость его структуры».

Структурный синтез в работе [34] определяется следующим образом: «Структурный синтез автоматов заключается в выборе типов элементарных

автоматов, в составлении функции возбуждения каждого элементарного автомата и функции кодирования состояний и выходов заданного автомата». Несмотря на то, что эта работа относится к числу самых ранних работ (как и работы Глушкова В.М.), точка зрения на структурный синтез УА до сих пор не изменилась [16].

По принципу формирования выходных команд  $A_i \in \{A\}$  автоматы разделяются на автоматы Мура, для которых  $A(t) = F_2(x_1 x_2 \dots x_m)$  и автоматы Мили, для которых  $A(t) = F_2(\alpha_1 \alpha_2 \dots \alpha_q x_1 x_2 \dots x_m)$ .

В диссертационной работе рассматриваются преимущественно автоматы Мура, т.к. они предназначены для управления электро и пневмоприводами и другими устройствами мехатроники, длительность исполнительных команд  $A_i \in \{A\}$ , равная  $T$ , должна быть больше длительности сигналов синхронизации  $\tau$ , обеспечивающих функционирование УА.

Автоматы Мура (рисунок 1.8) с минимальным числом структурно образующих блоков (классическая организация) содержат следующие подсистемы:

1. логическую (Л) в виде регистра памяти для  $\alpha_j \in \{\alpha\}$ ;
2. адресную (А) в виде комбинационной схемы  $F_1$ ;
3. информационную (И) в виде двух парафазно связанных регистров для хранения  $a(t)$  и  $a(t+1)$ . И – подсистема в виде линий задержек, которая использовалась в 1950-х годах, в настоящее время практически не используется;
4. функциональную (Ф) в виде комбинационной схемы  $F_2$  формирования команд  $A_i \in \{A\}$ . Чаще всего комбинационная схема  $F_2$  реализуется как дешифратор кода состояния  $a(t)$  или  $a(t+1)$  и набор схем «ИЛИ», объединяющих те выходы дешифратора, которые соответствуют выбору одной и той же команды  $A_i \in \{A\}$ .
5. управляющую (У) в виде блока синхронизации (БС), который формирует на каждом периоде  $T$  два синхроимпульса  $\tau_1(t) \wedge \tau_2(t) = \emptyset$ .

Автоматы Мура будем подразделять на простые и сложные УА.

Понятие сложности УА имеет три интерпретации:

- количественную;
- структурную;
- функциональную.

Сложность УА в количественном выражении будем определять через значение  $m$  и  $q$ .

Будем считать УА сверхпростыми (СП) при  $m = 3, q \leq 2$ ; простыми (ПА) при  $m = 4, q \leq 5$ ; средней сложности (СА) при  $m = 5, q \leq 8$ , сложные автоматы (АС) при  $m = 6$  и  $q = 11$ , высокой сложности (ВС) при  $m = 7$  и  $q \leq 14$ , особо сложными (ОС) при  $m = 8$  и  $q \leq 17$  и ультрасложными автоматами (УС) при  $m \geq 9, q \leq 21$  [35].

МПА как УА для средств обработки информации и вычислительной техники не отличаются сложностью, т.к. ориентированы на реализацию арифметических операций с учетом небольшого числа логических условий. Сложность МПА проявляется только при их комплексировании для всех 32-64 операций (команд).

Для управляющих спецпроцессоров, как правило, УА относится к СА, ВС и более сложным автоматам. При этом количество совместимых алгоритмов незначительно, т.е. УА для СПР существенно отличаются от МПА для вычислительной техники.

Если не рассматривать этапов конкретного функционального проектирования УА, а определять количественную сложность через объем оборудования ( $V$ ) для реализации  $F_1$  через ПЗУ, то  $V = m2^{m+q}$ . (таблица 1).

Таблица 1. Сравнительный анализ автоматов Мура

Тип	$m$	$q$	$m+q$	$V$
СП	3	2	5	0.1 кбит
ПА	4	5	9	2 кбит
СА	5	8	13	40 кбит
АС	6	11	17	0.8 Мбит
ВС	7	14	21	14 Мбит
ОС	8	17	25	256 Мбит
УС	9	20	29	4 Гбит

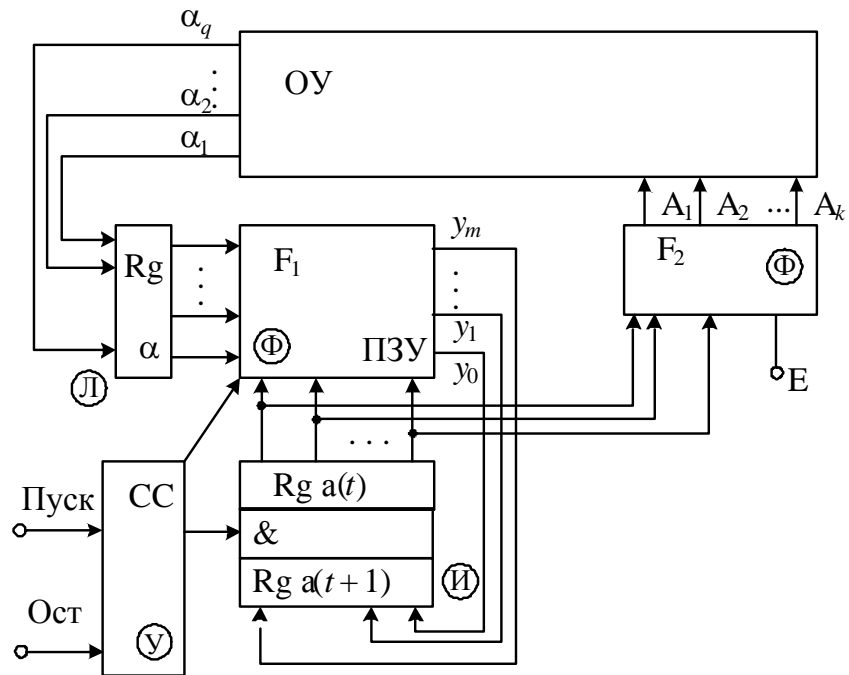


Рисунок 1.8 – Автомат Мура

Как видно, автоматы с классической структурной организацией разнятся по степени сложности в количественном отношении.

Независимо от количественной сложности УА все автоматы с классической структурной организацией относятся к структурно простым УА.

Управляющий автомат отнесем к функционально простым УА, если он реализует только основную функцию формирования  $A_i \in \{A\}$  по значению  $\alpha_j \in \{\alpha\}$  в соответствии с заданной ГСА. Независимо от количественной или структурной сложности такие УА относятся к функционально простым.

В структурно сложных автоматах одна или несколько подсистем ФИЛАУ сложны, т.е. представлены комплексом как минимум в виде двух взаимодействующих блоков. Например, для снижения количественной сложности  $\Phi$  – подсистемы УА комбинационная схема  $F_1$  декомпозируется [32-36], т.е. представляется в виде однонаправленной сети с выходом  $\{Y\}$ , вершинам

которой сопоставляются комбинационные схемы  $F_1^i$ , причем  $i = 1, 2, \dots, k$ . Для каждой  $F_1^i$  разрядность кода  $n_i < m+q$ . Структурно сложный автомат в количественном отношении может оказаться существенно проще структурно-простого УА при тех же значениях  $m$  и  $q$ .

Как правило, структурное усложнение одной из подсистем с целью обеспечения снижения степени количественной сложности приводит одновременно к необходимости модификации, а подчас и усложнению других подсистем и в первую очередь  $У$  – подсистемы УА.

К функционально сложным УА относятся такие автоматы, которые кроме основной функции, реализуют еще какие либо дополнительные функции, например, динамический контроль правильности функционирования самого автомата. УА такого типа называются самоконтролируемыми (СУА).

К функционально сложным относят также УА со специально введенными в ГСА операторами, не относящимися к основным операторам действия  $A_i \in \{A\}$ . Например, вводятся операторы индикации, регистрирующие переходы УА именно на заданную ветвь ГСА, реализующую определенный управляющий процесс.

К функционально сложным УА относятся также УА, сравнивающие время реализации типовой подпрограммы с заданным временем и фиксирующими по этому параметру неисправность внешнего оборудования.

Разрабатываемые в диссертационной работе методы синтеза ориентированы в первую очередь на создание АС, ВС и ОС автоматов. УС автоматы редко используются, т.к. они декомпозируются с переходом к ВС и АС автоматам. Более того, и для ОС автоматов редко используются случаи с предельным числом состояний ( $N = 128$ ).



## Выводы по главе 1

1. Проанализированы тенденции развития и методы проектирования спецпроцессоров.

2. На основе системного анализа показано, что для структурного синтеза спецпроцессоров наиболее эффективной является модель в виде полного пятивершинного графа, вершинам которого сопоставлены функциональная (Ф), информационная (И), логическая (Л), адресная (А) и управляющая (У) подсистемы. Предложено при проектировании структурной организации спецпроцессоров определять состав блоков и связей для каждой подсистемы в определенной последовательности  $\Phi \rightarrow \text{И} \rightarrow \text{Л} \rightarrow \text{А} \rightarrow \text{У}$ .

3. Предложено разделять управляющие автоматы по числу состояний и числу логических условий на сверхпростые (СП), простые (ПА), средние (СА), сложные (АС) и высокой сложности (ВС), а также особо сложные (ОС) и ультрасложные (УС) автоматы. Быстродействие и надежность спецпроцессоров в основном определяется управляющей подсистемой, которая может составлять до 2/3 оборудования.

4. На основе анализа вариантов структурной организации и методов синтеза управляющих автоматов показано, что в структуре управляющих автоматов до 80% оборудования составляет комбинационная схема переходов.

5. Определена актуальность и направление научных исследований в области анализа и синтеза управляющих автоматов спецпроцессоров.

## ГЛАВА 2. СИНТЕЗ УПРАВЛЯЮЩИХ АВТОМАТОВ

### 2.1. Методы структурного синтеза управляющих автоматов

В структуре управляющих спецпроцессоров (СПР) из всех подсистем ФИЛАУ именно управляющая подсистема (У) является наиболее сложной, т.к. (Л) подсистема – это, как правило, набор датчиков двоичной информации и схемы сравнения аналоговых сигналов для определения  $\alpha_j \in \{\alpha\}$ . Подсистема (Ф) - это комплекс функциональных преобразователей информации в виде БИС с логико-временной передачей промежуточной информации. При ограниченном объеме (И) подсистемы, подсистема (А) тривиальна в большинстве СПР.

Базовым методом проектирования управляющей подсистемы является теория конечных автоматов, основы которой заложены Гавриловым А.М., Глушковым В.М., Закревским А.Д., Захаровым В.Н., Горбатовым В.А., Лазаревым В.Г., Пупыревым Е.И., Барановым С.И., Сапожниковыми В.В. и Вл.В., Мухопадом Ю.Ф., Шалыто А.А., Соловьевым В.В. и др.

В большинстве работ по проектированию УА этап структурного проектирования рассматривается фактически как этап функционального синтеза, т.к. решаются задачи определения организации автоматов на различных логических базисах. Это связано с тем, что основная структурная организация УА считалась определённой, а ее изменения связаны лишь с декомпозицией систем булевых функций, реализующих переходы  $a(t) \rightarrow a(t+1)$ . В диссертационной работе этап структурного синтеза УА будет рассматриваться фактически как задача определения базового набора составляющих блоков и связей между ними для реализации УА с минимальными затратами оборудования.

В связи со сложностью А подсистемы УА (в виде системы булевых функций, реализуемых схемой  $F_1$ ) все усилия теории автоматов при синтезе направлены на снижение сложности реализации  $F_1$  за счёт минимизации системы булевых функций или представления единой схемы  $F_1$  комплексом более

простых комбинационных схем (декомпозиция) с меньшим числом логических переменных на входах каждой из ее частей. Имеются отдельные подходы к оценке сложности систем булевых функций, однако они определены лишь для особого класса булевых функций. Общая оценка отсутствует. Поэтому в диссертационной работе эффективность определяется степенью уменьшения объема ПЗУ требующегося для реализации комбинационной схемы адресной (А) подсистемы по сравнению с классическим вариантом УА [36-41].

Неэффективность методов минимизации булевых функций от большого числа  $n$  переменных ( $n = m+q$ ) определяется тем, что система  $m$  функций от  $n$  переменных рассматривается без учета специфики ГСА.

Некоторые способы учета специфики алгоритма УА были предложены Гавриловым М.А., Барановым С.И. и Горбатовым В.А. и Сапожниковыми В.В. и Вл.В. [31,42-44] на этапе декомпозиции и последующей «оптимальной» композиции совместимых подграфов графа переходов УА. Несмотря на наукоемкость предложенных методов синтеза УА, эффект, выраженный в величине снижения затрат оборудования, даже для ПА незначителен, а для СА и ВС автоматов практически отсутствует.

Новый подход с учетом специфики алгоритма при реализации УА с инженерных позиций предложен в работах [45,46]. Если исходный алгоритм для УА на основе последовательной итерации декомпозируется на части, то уменьшается число логических условий ( $q_k$ ), относящихся к каждой из ( $k$ ) частей алгоритма. При этом, естественно, уменьшаются и значения  $m_k$ , относящиеся к каждой части. В структуру УА вводится простой коммутатор логических условий  $\alpha_j$ , который соотносит подмножество  $\alpha_j \in \{\alpha\}$  с соответствующей  $k$ -той частью алгоритма. Метод эффективен, и даже для СА и ВС УА снижает сложность А – подсистемы в 1,5 -1,7 раза. Заметим, что такая эффективность недостижима в других методах декомпозиции. Метод учета специфики УА на уровне алгоритмов, а не графов переходов наиболее эффективен, но требует дальнейшего развития.

Несмотря на достаточно большое число предложенных наукоемких методов

синтеза УА, все они в большей мере были ориентированы на реализацию комплекса несложных ГСА, определяющих набор микропрограмм универсальных вычислительных процессоров. Если рассмотреть ГСА каждой вычислительной операции [47-69], то при аппаратной реализации такой МПА будет соответствовать СП и ПА автомату.

Управляющие устройства относятся к классу ВС, ОС и даже УС автоматов, т.к. число логических условий может быть  $\geq 20$ . Например, при управлении железнодорожным перегонном (не говоря о станции) число переключаемых стрелок и сигналов может быть  $\approx 50$ . Однако существующая теория проектирования УА с ориентацией в основном на ПА и редко СА автоматы т.к. рассматривался комплекс простейших автоматов для создания единого МПА [41,47-63]. Значительно упрощение комбинационных схем УА при полном множестве логических условий удается достичь за счет эквивалентных преобразований ГСА [64]. Теория управляющих автоматов стала развиваться в направлении разработки методов синтеза с учетом особенностей элементной базы. В связи с развитием микроэлектроники и появления разно номенклатурных ПЛМ и ПЛИС (PLA) развивается в основном функциональный синтез [37,65-75].

Методы структурного синтеза управляющих автоматов, относящихся к классу ВС, ОС И УС автоматов практически не развиты.

## 2.2. Новый метод синтеза управляющих автоматов

В диссертационной работе предложен новый подход к решению задачи абстрактного синтеза УА. Во всех известных методиках абстрактного синтеза УА алгоритм в форме операторной схемы (ОСА) считается заданным и не подлежащим принципиальному изменению (кроме совмещения некоторых операторов [14]).

Алгоритм УА может быть задан в виде любой из операторных схем, т.е. в виде граф-схемы (ГСА), матричной схемы (МСА), логической (ЛСА), регулярной (РСА) или табличной схемы (ТСА). В работе [26] даны правила взаимного

преобразования основных видов ОСА. Чаще используется ГСА, хотя при автоматизированных методах анализа ОСА предпочтительнее ЛСА, МСА или ТСА.

В заданной ГСА применяются следующие комбинации операторов действия  $A_i \in \{A\}$  и логических условий  $\alpha_j \in \{\alpha\}$  (рисунок 2.1):

1. логическому условию  $\alpha_j \in \{\alpha\}$  предшествует один - единственный оператор действия  $A_i \in \{A\}$ ;

2. к логическому оператору передается управление от двух и большего числа операторов действия;

3. имеется непрерывная последовательная проверка значений нескольких логических условий  $\alpha_i, \alpha_j, \alpha_k, \dots$  без промежуточных (между двумя соседними  $\alpha_i$  и  $\alpha_j$ ) операторов действия  $A_i \in \{A\}$ ;

4. используется непрерывная последовательность операторов действия с безусловной передачей управления от предыдущего оператора к последующему.

Очевидно, что в первом случае по номеру состояния  $a(t)$ , которое предшествует проверке  $\alpha_j$ , может быть однозначно определен номер логического условия ( $j$ ). Для случая 4 всегда необходимо выбирать безусловный переход  $\alpha_0$ , а для случая 3 одно и то же значение  $a(t)$  будет сопоставлено нескольким логическим условиям.

Случай 3 можно привести к ситуации 1, если ввести дополнительный пустой оператор перед логическим оператором  $\alpha_j$ , тогда оператор  $\alpha_j$  получит управление от единственного предшествующего ему оператора. В данном случае это будет пустой оператор, т.е. такой оператор, которому не сопоставлен ни один из операторов действия заданного множества  $A_i \in \{A\}$ .

Анализ ситуаций 1-4 позволяет сформулировать утверждения 1, 2.

Утверждение 1. Выбор единственного логического условия по коду конкретного состояния  $a(t)$  возможен, если перед логическим оператором поставлен пустой оператор, объединяющий переходы от других операторов к заданному логическому условию.

Утверждение 2. Выбор единственного условия  $\alpha_j \in \{\alpha\}$  достигается в том случае, если между неразделенными логическими операторами  $\alpha_j$  и  $\alpha_i$  поставить пустой оператор (преобразование рисунка 2.1-2 к ситуации рисунка 2.2-б, преобразование условия рисунка 2.1-3 к ситуации рисунка 2.2-а).

Доказательство утверждения 1, 2 очевидно с учетом пунктов 1-4.

Необходимо рассматривать также вариант 5, когда одно и то же  $\alpha_j \in \{\alpha\}$  проверяется в ГСА многократно при разных состояниях  $a(t)$ . Очевидно, что в этом случае один и тот же номер ( $j$ ) будет соответствовать нескольким состояниям  $a(t)$ .

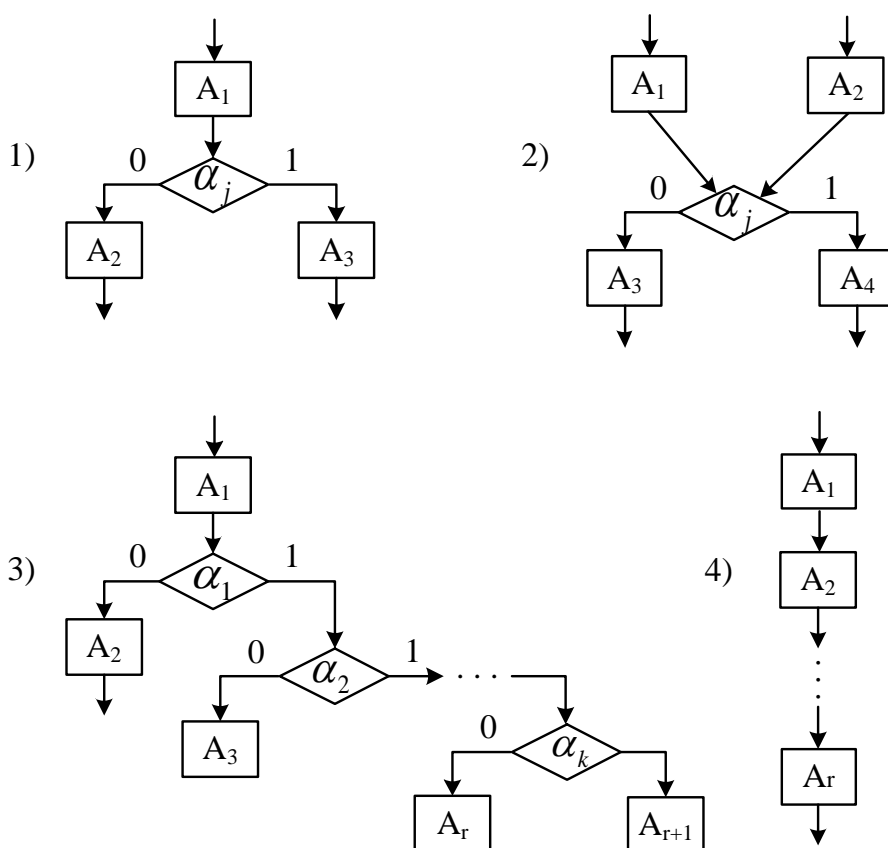
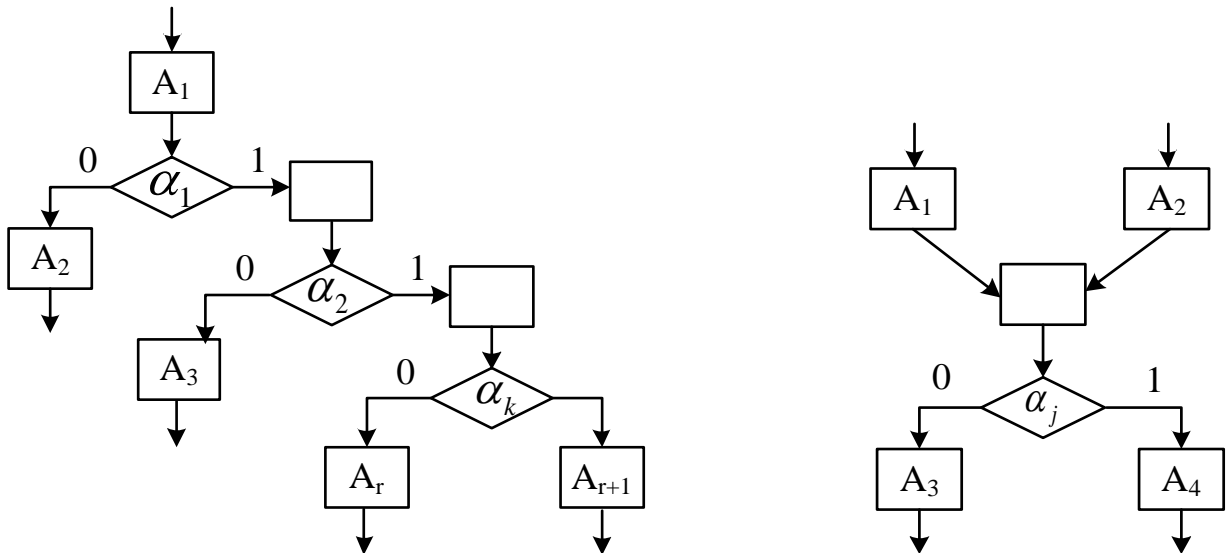


Рисунок 2.1 – Варианты последовательностей операторов действия и логических условий в ГСА



а - между логическими условиями;

б – перед одним логическим условием

Рисунок 2.2 – Способы введения пустых операторов:

Преобразуем ОСА по следующему правилу:

1. После каждого оператора  $\alpha_i \in \{\alpha\}$  вставим пустой оператор  $A_p$ , если в ОСА идет непосредственная передача управления другому оператору  $\alpha_j \in \{\alpha\}$  без промежуточной команды  $A(t)$ .

$$\downarrow^1 A_i \alpha_1 \uparrow^1 \alpha_2 \uparrow^2 A_j \downarrow^2 A_r \rightarrow \uparrow^1 A_i \alpha_1 \uparrow^1 A_0 \alpha_2 \uparrow^2 A_j \downarrow^2 A_r;$$

2. Пустой оператор вставляется также в ОСА в том случае, если к  $\alpha_j \in \{\alpha\}$  идет передача управления от нескольких других операторов.

$$(A_i \wedge A_j) \alpha \uparrow^1 A_r \downarrow^1 A_5 \rightarrow (A_i \wedge A_j) \alpha \uparrow^1 A_2 \downarrow^1 A_5$$

3. Преобразуем ГСА к однородному виду за счет ввода пустых операторов так, чтобы к любому оператору  $A_i \in \{A\}$  передавалось управление не более чем от двух операторов действия (рисунок 2.3).

$$\left. \begin{array}{l} (A_1 \wedge A_2 \wedge A_3)A_4 \rightarrow (A_1 \wedge A_2)A_0 \\ A_3 \end{array} \right\} A_4$$

4. Если осуществляется переход к заключительному (нулевому) состоянию УА от некоторого логического условия, то между выходом этого

логического условия и заключительным состоянием ставится пустой оператор (рисунок 2.4).

5. От отмеченной ГСА осуществляется переход к графу.
6. Составляется таблица соответствия номера (кода) состояния  $a(t)$  номеру ( $j$ ) логического условия  $\alpha_j$  (включая  $\alpha_0$ ).
7. Составляется таблица переходов, в строки которой включаются номера состояний соответствующих формуле:  $a(t+1) = F_1(\alpha_j; a(t))$ .

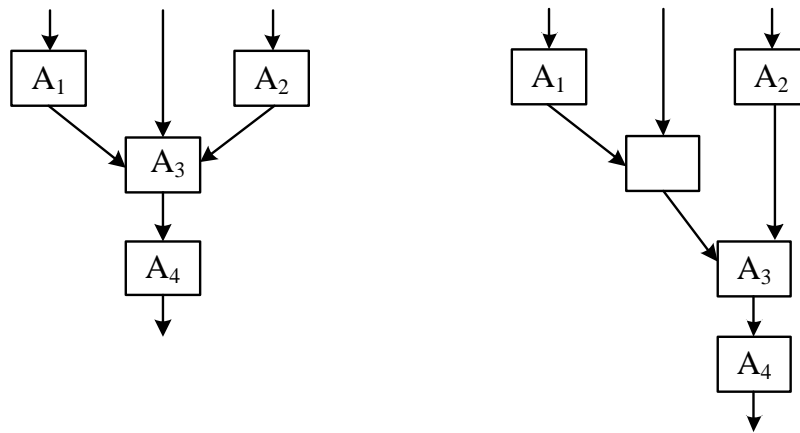


Рисунок 2.3.- Преобразование ГСА (вариант 1)

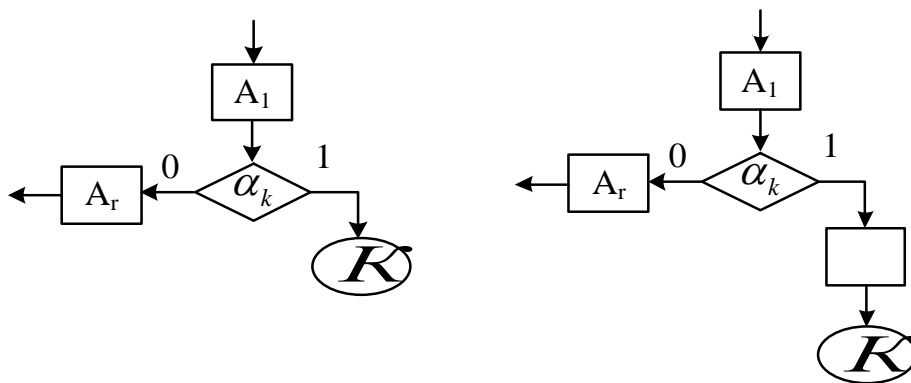


Рисунок 2.4 – Преобразование ГСА (вариант 2)



### 2.3. Структурный синтез управляющих автоматов

Метод структурного синтеза сводится к следующему:

1. Выделяется последовательность состояний автомата на самом длинном пути. Для рисунка 2.8 имеем состояния непрерывных путей: 0-11; 12-14; 15,16; 17-19. Состояния нумеруются по порядку. Всем остальным состояниям графа сопоставляются последующие номера.

2. По графу переходов составляется таблица 2.1 номеров логических условий, соответствующих конкретным состояниям УА.

3. Количество состояний  $N$  в графе переходов определяет разрядность кодов ( $m$ ) состояний из условия  $N \leq 2^m$ . Здесь  $N=23$ ,  $m=5$ .

4. По графу составляется таблица 2.2 переходов  $a(t) = \widetilde{\alpha}_j a(t+1)$ , по которой в структуре УА определяется комбинационная схема  $F_1$ , преобразующая код  $x_1 \dots x_m$  в код  $y_1 y_2 \dots y_m$  с учетом  $\alpha_j \in \{\alpha\}$ .

5. В структурную схему УА вводится мультиплексор, к информационным входам которого подсоединяются сигналы всех логических условий, включая безусловный переход  $\alpha_0$ .

6. Адрес  $z_1 z_2 \dots z_r$  соответствующий номеру ( $j$ ) логического условия определяется через комбинационную схему  $F_3 (y_1 y_2 \dots y_m)$ .

7. В структуре УА предусматривается дешифратор (ДС) для получения унитарного кода  $a(t)$  по двоичному коду  $\{Y\}$ , а также набор схем «ИЛИ», объединяющий те выходы ДС, которым соответствует выбор одной и той же команды управления  $A_i \in \{A\}$ . Набор схем «ИЛИ» составляет комбинационную схему  $F_2$  для формирования команд  $A_1 A_2 \dots A_k$ .

Введение пустых операторов не меняет ни последовательности, ни логического взаимодействия  $A_i \in \{A\}$  в ГСА. Исходная и преобразованная ГСА будут функционально тождественны. Управляющий автомат, синтезированный по модифицированной ГСА, будет менее быстродействующим при реализации тех ветвей ГСА, в которые вставлены пустые операторы.

Таблица 2.1. Выбор логических условий

$\alpha_j$	a(t)		$\alpha_j$	a(t)
1	1		5	5
2	2		6	6
3	3		7	12
4	4		8	15
0	0, 7-11		0	13,14,19-22

Таблица 2.2. Таблица переходов УА

№	a(t)	$\alpha$	a(t+1)	№	a(t)	$\alpha$	a(t+1)
1	0	$\alpha_0$	1	16	10	$\alpha_0$	11
2	1	$\alpha_1$	2	17	11	$\alpha_0$	1
3	1	$\overline{\alpha_1}$	12	18	12	$\overline{\alpha_7}$	11
4	2	$\overline{\alpha_2}$	3	19	12	$\alpha_7$	13
5	2	$\alpha_2$	15	20	13	$\alpha_0$	14
6	3	$\overline{\alpha_3}$	4	21	14	$\alpha_0$	10
7	3	$\alpha_3$	17	22	15	$\alpha_8$	16
8	4	$\overline{\alpha_4}$	5	23	15	$\overline{\alpha_8}$	19
9	4	$\alpha_4$	20	24	16	$\alpha_0$	14
10	5	$\overline{\alpha_5}$	6	25	19	$\alpha_0$	9
11	5	$\alpha_5$	21	26	17	$\alpha_0$	18
12	6	$\alpha_6$	7	27	20	$\alpha_0$	18
13	6	$\overline{\alpha_6}$	22	28	18	$\alpha_0$	19
14	7	$\alpha_0$	8	29	21	$\alpha_0$	8
15	8	$\alpha_0$	9	30	22	$\alpha_0$	0

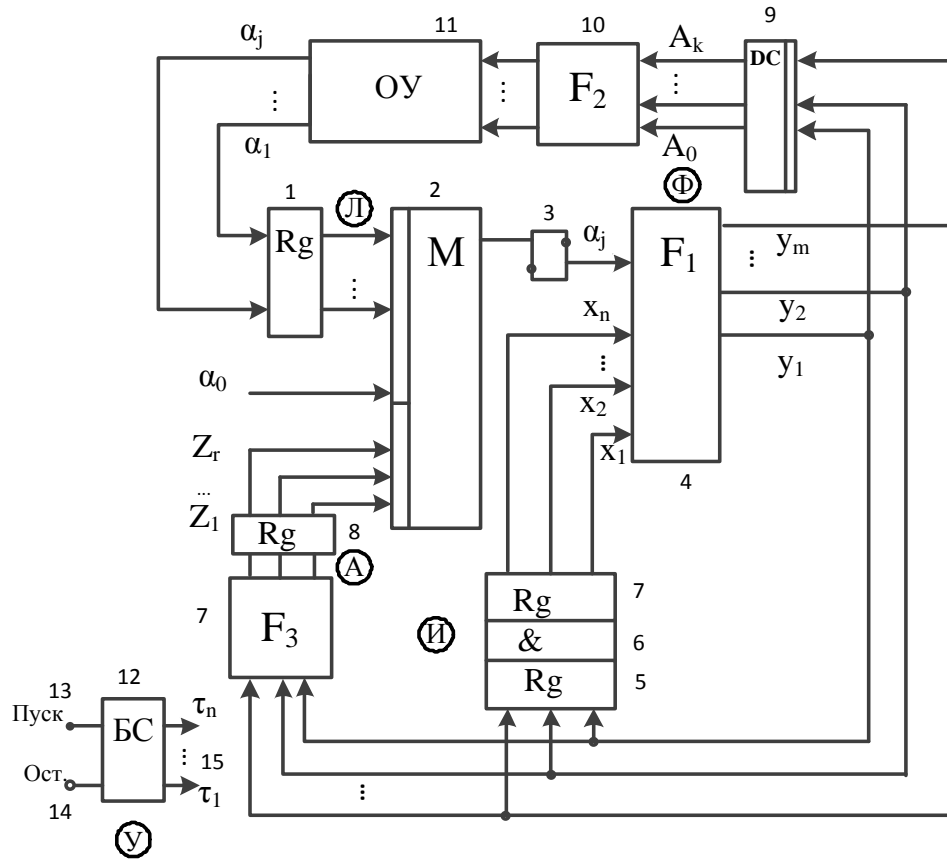


Рисунок 2.5 – Структурная схема управляющего автомата с выбором логического условия по коду состояний

Структурная схема УА (рисунок 2.5) созданная по предлагаемой методике, является оригинальной и запатентована [76].

Преобразуем ГСА (рисунок 2.6) для интерфейсной схемы интеллектуального датчика, которую авторы работы [55] рекомендуют реализовать как УА. Модифицированная ГСА представлена на рисунке 2.7.

По предложенной методике синтеза на основе модифицированной ГСА получена оригинальная структурная организация УА с мультиплексором (М) для выбора  $\alpha_j \in \{\alpha\}$  по коду состояния  $a(t)$ . В схеме адресации  $F_3$  мультиплексора М для определения адресного кода номера логического условия (j) в виде  $Z_1 Z_2 \dots Z_r$  используется код  $Y_1 Y_2 \dots Y_m$ .

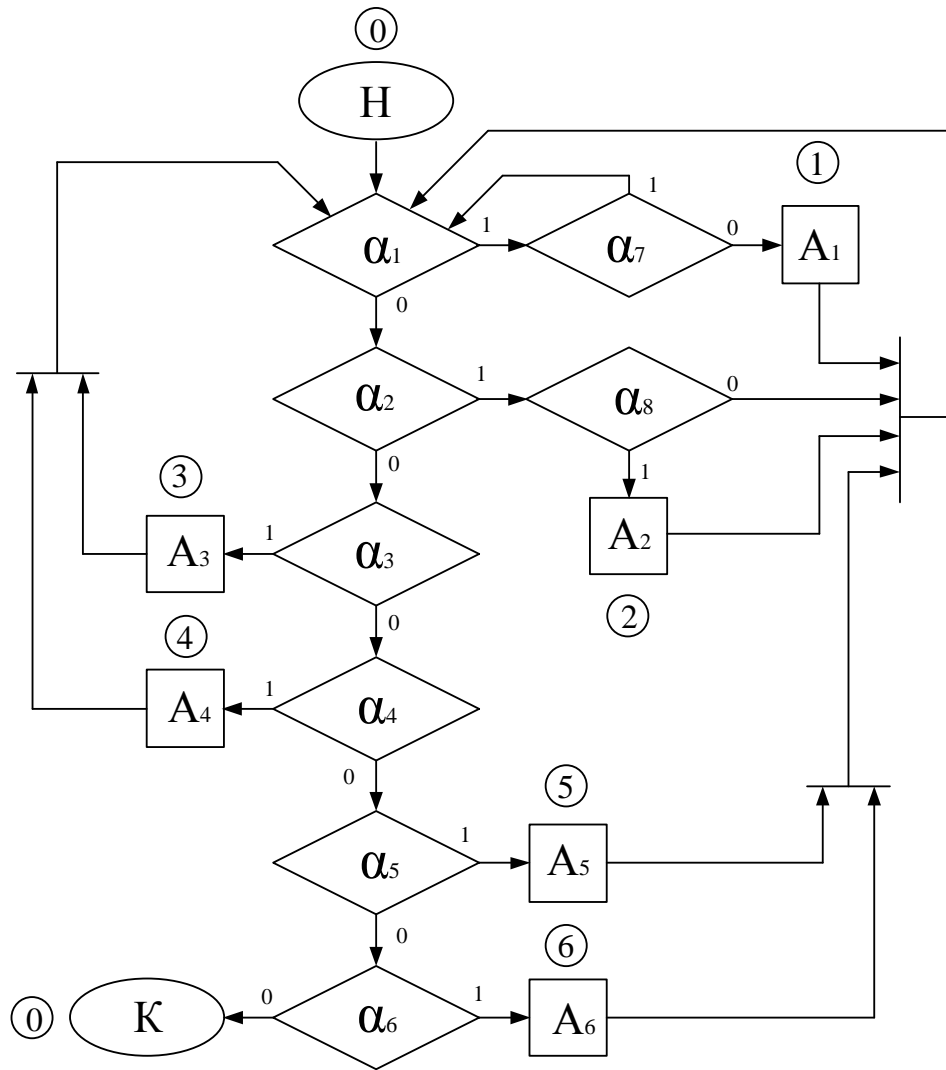


Рисунок 2.6 - Алгоритм управления интерфейсом

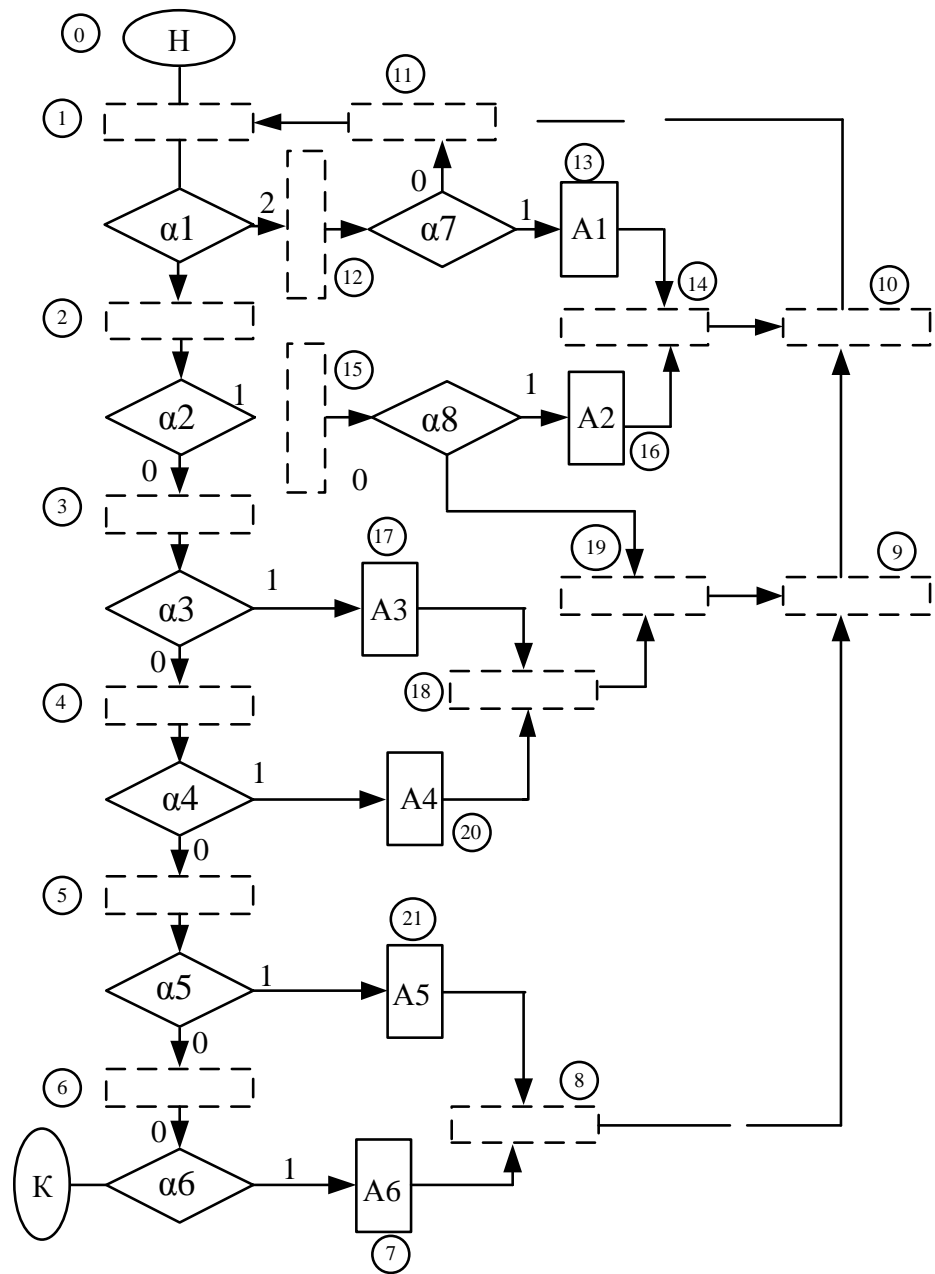


Рисунок 2.7 - Модифицированная ГСА с правилом разметки Мура

При реализации классического УА количество состояний будет меньше ( $m=3$ ), но при  $q=8$  объем ПЗУ составит  $V = m_p 2^{m+q} = 4 \cdot 2^{11} = 2^{14} = 16 \text{кб}$ , а для УА нового типа при  $m=5$  также берется реальное  $m=8$ , тогда  $V = 8 \cdot 2^6 = 2^9 = 0.5 \text{кб}$ . Снижение объема ПЗУ составит  $Q = V/W = 32$ .

#### 2.4. Анализ структурной организации управляющих автоматов

Анализ ГСА (рисунок 2.7), и полученного графа переходов (рисунок 2.8) позволяет сделать важный вывод: переходы в новом графе УА осуществляются либо безусловно, либо при проверке одного логического условия  $\alpha_j \in \{\alpha\}$ . Причем номер этого логического условия ( $j$ ) полностью определяется кодом  $a(t)$ , т.е.  $j = F_3(a(t))$  или  $j = F_3(x_1 x_2 \dots x_m)$ .

В новом УА объем ПЗУ для  $\Phi$  - подсистемы оценивается по формуле  $W = m 2^{m+1}$ . Конкретные значения  $V$ ,  $W$  и  $Q = V/W$  для всех типов автоматов по классификации приведены в табл.2.3. Величины  $V$  и  $W$  выражены в битах. Как видно из анализа (столбцы 7 и 9 таблицы 2.3), объем ПЗУ в новой структуре УА даже для ПА снижается в 16 раз, для СА более, чем в 100, а для ВС ~ в тысячу раз. При этом в структуру УА вводится мультиплексор для выбора одного  $\alpha_j \in \{\alpha\}$  по адресу  $a(t+1)$  код  $z_1 z_2 \dots z_r$ , который определяется функцией  $F_3$  через код  $u_1 u_2 \dots u_m$ .

Эта оценка является верхней, т.к. произведена по объему требуемой памяти ПЗУ. Сложность вновь вводимых блоков составляет не более  $W$ . Оценка сложности вводимых блоков приближительна, т.к. произведена не аналитически, а на основе многих практических реализаций УА. Аналитическое сравнение невозможно в связи с отсутствием методик оценки сложности реализации систем булевых функций УА для ГСА любого типа. Для нового типа автоматов с выбором логического условия аналитическое описание имеет вид:

$$a(t+1) = F_1(a(t), \alpha_j); \quad A(t) = F_2(a(t)); \quad J = F_3(a(t)) \quad (2.1)$$

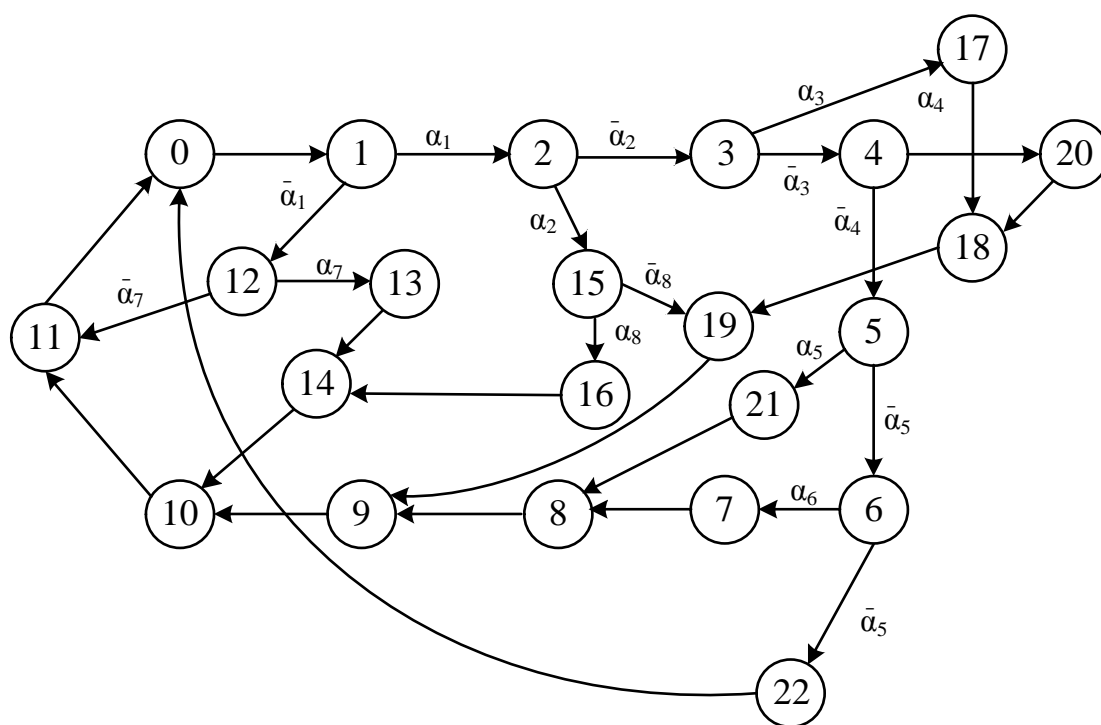


Рисунок 2.8 – Граф переходов

Таблица 2.3. Сравнительный анализ сложности автоматов

Тип	m	q	m+q	m+1	m <sub>p</sub>	V(бит)	W(бит)	Q(раз)
СП	3	2	5	4	4	100	48	2
ПА	4	5	9	5	4	8000	128	60
СА	5	8	13	6	8	65000	512	128
АС	6	11	17	7	8	1М	1024	1000
ВС	7	14	21	8	8	16 М	2048	$8 \cdot 10^3$
ОС	8	17	25	9	8	270 М	4096	$67 \cdot 10^3$
УС	9	20	29	10	8+4	34 Г	$260 \cdot 10^3$	$130 \cdot 10^3$

В отличие от классической структуры автомата Мура с импульсами синхронизации  $\tau_1(t) \wedge \tau_2(t) = \emptyset$ , в автомате нового типа в  $Y$  – подсистеме (блок синхронизации) должны формироваться четыре импульса:

$$\tau_1(t) \tau_2(t) \tau_3(t) \wedge \tau_4(t) = \emptyset.$$

При этом по соответствующим сигналам производятся следующие действия:

Сигнал  $\tau_1$

1. Отключается исполнение команды  $A_j$ .

Опрашивается  $F_1$  (1) и производится запись в  $Rg$  (2),  $Rg$  (5).

Сигнал  $\tau_2$

2. Опрашивается  $Rg$  (2),  $Rg$  (5).

Сигнал  $\tau_3$

3. Опрос СВ (4) и запись  $\alpha_j$  в RS (8).

Сигнал  $\tau_4$

4. Перепись результата схем  $F_1(4)$  на  $Rg(5)$  и через  $F_3$  на  $Rg8$ . Включение  $A_j$  и повторение первого пункта по установке исходного состояния УА.

Как видно из рисунка 2.5, в УА хотя и присутствуют все подсистемы  $\Phi$ , И, Л, А, У, но, согласно уравнениям (2.1), нет необходимости в полных связях. Действительно, У - подсистема УА имеет связь (прямую) со всеми подсистемами, но нет обратных связей от остальных подсистем к У - подсистеме, как и нет связей  $I \rightarrow \Phi$ ,  $I \rightarrow Л$ ,  $\Phi \rightarrow Л$  и др. Новый автомат не является автоматом Мура в полной степени, т.к. он не повторяет его структурную организацию, а только формирует выходные команды  $A(t)$  с длительностью импульсов, равных периоду следования  $T$  при завершившемся переходе из  $a(t)$  в  $a(t+1)$ .

В автомате нового типа реализуется правило формирования  $A(t)$  Мура.

Объем оборудования для реализации мультиплексора и схемы  $F_2$  в сумме  $\approx W_p$  для нового УА. Поэтому общий выигрыш равен  $Q = 2^{q-2}$ .

На сегодня нет более эффективной организации УА (в смысле затрат оборудования на  $\Phi$  - подсистему), чем предложенная структурная схема рисунка 2.5.



## 2.5. Сравнительный анализ комбинационных схем УА

Оценка сложности реализации с.б.ф. на элементах известного логического базиса основана на ряде ограничений [40]. Произведем оценку сложности реализации с.б.ф. в базисе И, ИЛИ, НЕ без учета возможной минимизации по непосредственному представлению в ДНФ. При реализации комбинационной схемы  $F_1$  на ПЗУ объем ПЗУ равен  $V = m2^{m+q}$ .

При переходе к оценке сложности реализации с.б.ф. затраты элементов зависят от сложности конъюнкций  $(m+q)$  переменных и числа конъюнкций в ДНФ определяющих каждую функцию  $y_j \in \{y\}$ .

В автоматах Мура используются две комбинационные схемы:

$F_1$  – реализующая с.б.ф. переходов  $a(t) \rightarrow a(t+1)$  с  $(m+q)$  переменными на входе.

$F_2$  – формирующая команды управления  $A_1 \dots A_k$  через объединение по «ИЛИ» выходов дешифратора ДС, которые определяют одну и ту же команду  $A_i \in \{A\}$ .

В новом УА требуется схема адресации  $F_3$  мультиплексора (М).

В каждой конъюнкции схемы  $F_1$   $(m+q)$  переменных при реализации на логических элементах используется  $k = ((m+q) \bmod r + d_1)$  схем «И», если число входов одного логического элемента не более  $(r)$ . Если  $(m+q) \bmod r$  имеет остаток, то к  $(m+q) \bmod r$  добавляется 1. Здесь  $d_1$  – необходимое количество элементов «И» в ярусах (кроме первого входного слоя) для реализации «дерева», образующего конъюнкцию  $(m+q)$  переменных (рис.2.9).

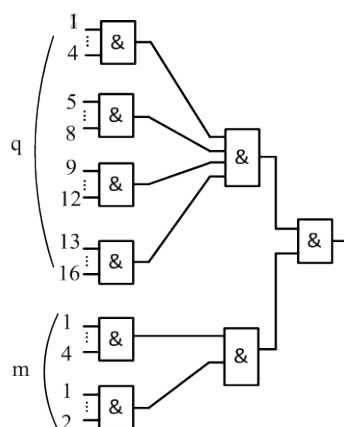


Рис. 2.9 Схема реализации конъюнкции  $(m+q)$  переменных & - логический элемент «И»

Без учета специфики ГСА примем предположение, что для реализации любой  $y_j \in \{y\}$  потребуется ДНФ с одним и тем же (средним) числом конъюнкции  $P$ .

Как видно из таблицы 2.2 для значение  $y_i = 1$  отмечается в половине общего числа строк. Следовательно, число конъюнкций для каждой функции  $y_j \in \{y\} = 2^{m-1}$ , если в наиболее сложном УА данного класса число строк равно  $2^m$ .

При реализации  $F_1$  на ПЛМ или ПЛИС потребуется также  $(m+q)$  элементов «НЕ» [7,12]. Тогда полное число логических элементов для комбинационной схемы  $F_1$ , реализующей переходы в УА Мура можно определить по формуле 2.2.

$$V = (m+q) + m (((m+q) \bmod r + r/2 + r / 4) 2^{m-1}) \quad (2.2)$$

Для УА нового типа за счет наличия мультиплексора (М) и схемы адресации  $F_3$  для комбинационной схемы выбирается только одно  $\alpha_j \in \{\alpha\}$ , поэтому вместо  $(m+q)$  переменных используется  $(m+1)$  переменная.

По новой методике синтеза в ГСА вводятся пустые операторы, поэтому число состояний УА увеличивается, но уменьшается число переходов. При этом число связей в графах (рис.2.8) остается приблизительно одинаковым. Следовательно, и число конъюнкций для каждой функции  $y_j \in \{y\}$  будет тем же. Тогда сложность схемы  $F_1$  можно оценить по формуле 2.3:

$$W = (m+1) + m (((m+1) \bmod r + r / 4) 2^{m-1}) \quad (2.3)$$

В формуле 2.3 отсутствует слагаемая  $r/2$ , т.к. число ярусов в «дереве» реализации комбинационной схемы уменьшается при переходе от  $(m+q)$  к  $(m+1)$ . В автоматах нового типа кроме  $F_1$  имеется еще схема  $F_3$  и мультиплексор М. Однако сложность  $F_3$  более чем в 2.5 раза ниже  $F_1$  за счет уменьшения на один разряд, как числа входов, так и числа выходов  $F_3$ . Практические оценки для СА, ВС и ОС автоматов позволяют считать, что

$$W(\Sigma) = W(F_1) + W(M) + W(F_3) \approx 1.5 W(F_1)$$

Оценка сложности реализации УА нового типа по сравнению с УА Мура при реализации комбинационной схемы через с.б.ф. приведена в таблице 2.4.

Таблица 2.4. Сравнительный анализ автоматов по системе булевых функций

№	УА	m	m+1	m-1	q	m+q	$V_c$	$W_c$	$Q_c$	$W_c(\Sigma)$	$Q_c(\Sigma)$
1	СП	3	4	2	2	5	41	40	1	64	0,68
2	ПА	4	5	3	5	9	137	69	2	156	1,32
3	СА	5	6	4	8	13	413	166	2,5	249	1,65
4	АС	6	7	5	11	17	1553	583	2,6	874,5	1,77
5	ВС	7	8	6	14	21	4053	1352	3,0	2028	2
6	ОС	8	9	7	17	25	10265	4105	2,5	4100	1,66
7	УС	9	10	8	20	29	25373	9226	2,7	13839	1,8

В таблице 2.4  $V_c$ ,  $W_c$ ,  $Q_c$  – объем  $\Phi$  подсистемы и их сравнение для с.б.ф., соответственно  $W_c(\Sigma)$  и  $Q_c(\Sigma)$  – суммарный объем и сравнение для автоматов нового типа.

Таким образом, реализация управляющих автоматов нового типа с перестраиваемой программой позволяет сократить объем ПЗУ с электрическим стиранием информации в десятки, сотни и тысячи раз в зависимости от класса автоматов. Для автоматов с фиксированной программой, которые реализуются комбинационными схемами через систему булевых функций, количество логических элементов для УА нового типа уменьшается почти в 2-3 раза для высоко сложных автоматов. Основные затраты логических элементов в новом УА определяются комбинационными схемами и мультиплексором. Затраты на другие блоки, обеспечивающие работу УА, незначительны и практически такие же, как и в автомате Мура.

## 2.6. Варианты структурной организации управляющих автоматов нового типа

Вариант 1. Для автоматов нового типа с правилом формирования команд управления Мура время реализации последовательности операторов на самом длинном пути ГСА увеличится в  $K$  раз.  $K = (N_1 + N_2) / N_1$ . Здесь  $N_1$  – число операторов действия на самом длинном пути,  $N_2$  – число введенных пустых операторов.



В структуре УА (рисунок 2.11) выходы схем  $F_1^0$  и  $F_1^1$  объединяются схемами «ИЛИ».

При функциональном проектировании на реальной элементной базе общий объем комбинационных схем  $F_1^0$  и  $F_1^1$  принципиально может оказаться меньше объема  $W$ . Однако реально это уменьшение незначительное, поэтому при оценке вариантов структурной организации будем считать, что  $W = F_1^0 + F_1^1$ .

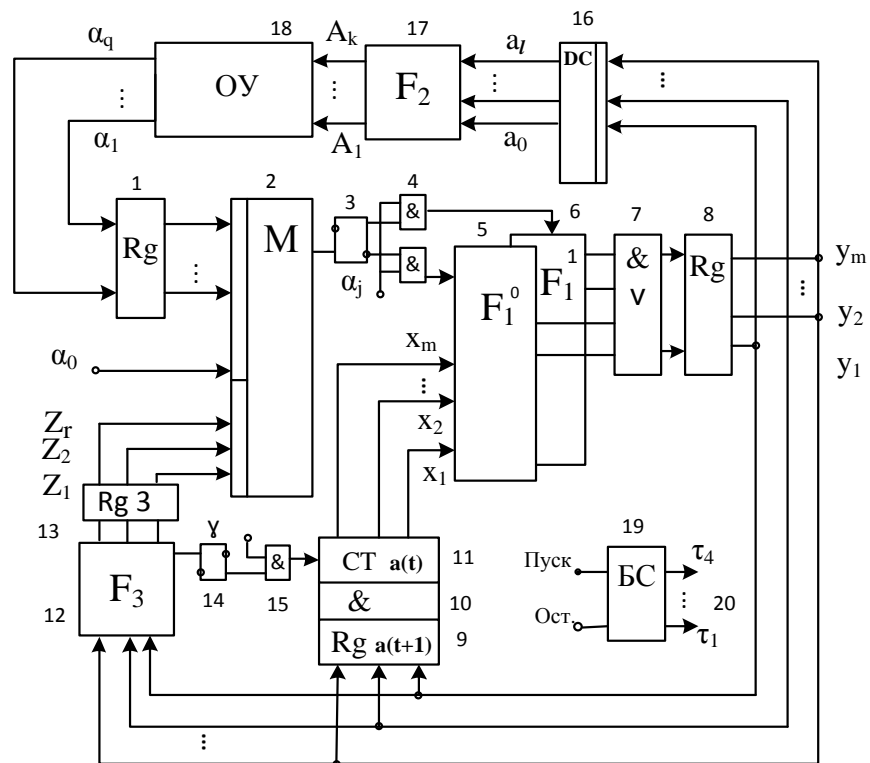


Рисунок 2.11 – Управляющий автомат с разделенной комбинационной схемой переходов по  $\alpha_j$  и  $\bar{\alpha}_j$

Вариант 3. Синтез управляющих автоматов с коммутационной схемой. Существует достаточно большой класс алгоритмов управления, для которого разрядность кода состояний ( $m$ ) больше требуемой разрядности адреса представления номера  $j$  мультиплексора в структуре УА нового типа. Причем число логических условий  $\alpha_0 \alpha_1 \alpha_2 \dots \alpha_n$  меньше общего числа  $2^r$ . Здесь  $r$  – разрядность представления кода  $\{\alpha\}$ . В этом случае нет необходимости использовать полный мультиплексор, т.к. число логических условий  $q < 2^r$ , следовательно, в мультиплексоре могут быть использованы информационные

входы соответствующие только реальным адресным комбинациям. Поэтому вместо полного мультиплексора и схемы его адресации может быть использована коммутационная схема. Рассмотрим ГСА (рисунок 2.7) для которой полное число информационных входов  $2^r = 16$  из которых для данной ГСА будет использовано только 9. Почти половина (7 из 16) входов не используется. Следовательно, 7 соответствующих входных комбинаций в мультиплексоре будут лишними. Структурная схема УА с коммутатором представлена на рисунке 2.12.

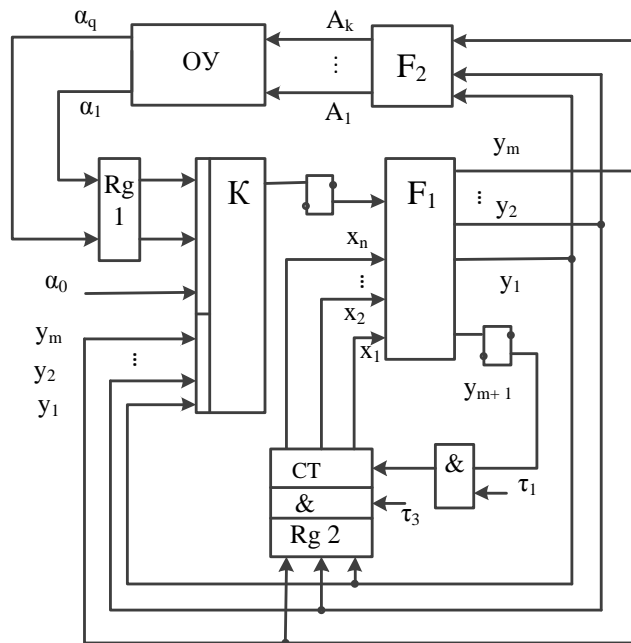


Рисунок 2.12 – Структурная схема УА с коммутационной схемой.

Вариант 4. Существо предложенного метода синтеза сводится к тому, что альтернативное вычисление кода  $a(t+1)$  производится не по полному коду конкатенации  $(\alpha_j x_1 x_2 \dots x_m)$ , а по частям единого кода, т.е. через  $(\alpha_j x_1 x_2 \dots x_p)$  и  $(\alpha_j x_{p+1} \dots x_m)$ . По первой части с учетом  $\alpha_j$  определяются младшие разряды кода  $a(t+1)$ , т.е.  $(y_1 \dots y_p)$ , а по второй – старшие разряды, т.е.  $(y_{p+1} \dots y_m)$ . Поскольку проведено разделение переменных кода  $\{x\}$  на две части, то однозначное соответствие младших  $\{y\}_p$  разрядов по коду  $\alpha \{x\}_p$  не может быть достигнуто, т.к. возникают противоречия без учета старших разрядов  $\{x\}_m$ . Для снятия этих противоречий предложено ввести в каждую часть  $\{x\}_p$  и  $\{x\}_m$  кроме

$\alpha_j$  еще по одному дополнительному разряду  $\delta_1$  и  $\delta_2$ , значение которых определяется в соответствии с кодами строк таблицы переходов  $\alpha_j a(t) \rightarrow a(t+1)$ . Таким образом общая комбинационная схема  $F_1$ , определяющая код состояния  $a(t+1) = F_1(\alpha_j x_1 x_2 \dots x_m)$ , делится на две подсхемы для младших  $a(t+1)_p = F_1^1(\delta_1 \alpha_j x_1 x_2 \dots x_p)$  и старших разрядов  $a(t+1)_m = F_1^2(\delta_2 \alpha_j x_{p+1} \dots x_m)$ . Причем части кодов  $(x_1 x_2 \dots x_p)$  и  $(x_{p+1} \dots x_m)$  могут «перекрываться» одним разрядом  $p$ , если значение  $m$  нечетно. Предложенный метод реализации УА возможен только для автоматов нового типа и не применим в классических УА Мура. Структурная схема такого автомата представлена на рисунке 2.13.

### 2.7. Синтез быстродействующих автоматов

Повышение быстродействия УА может быть осуществлено за счёт использования параллельной памяти [75] для кодов  $a(t)$ ,  $a(t+1)$  (рисунок 2.14).

Однако в ряде случаев и этих мер оказывается недостаточно для сверхбыстродействующих систем, применяемых для обработки радиосигналов в сверхскоростной связи, радиотехнических цифровых фильтрах и др. В таких

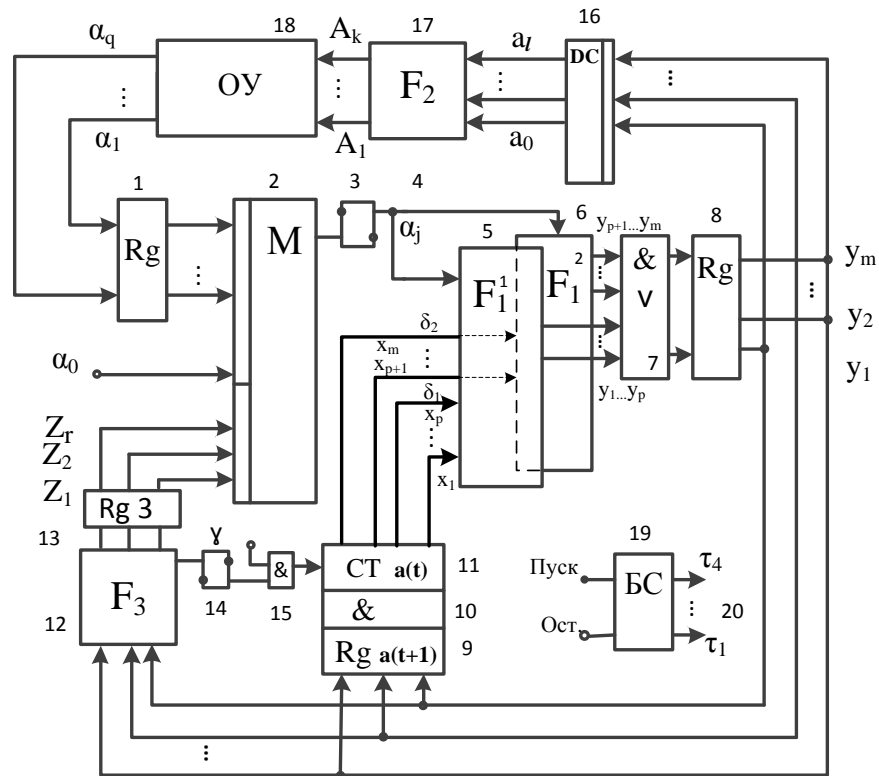


Рисунок 2.13 – Структурная схема УА с разделением КС по разрядам

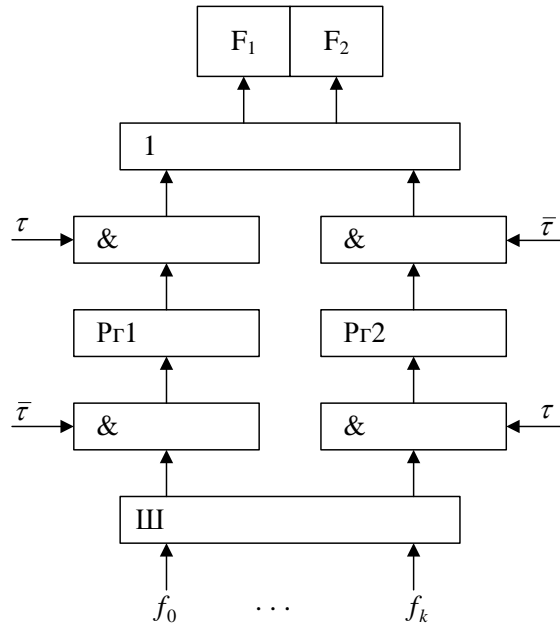


Рисунок 2.14 - Параллельная память автоматов

системах требуется, чтобы микрокоманда начала исполняться сразу в начале перехода  $a(t) \rightarrow a(t+1)$ , т.е. по переднему фронту импульса. В этом случае автоматы реализуются со структурной организацией Мили [26,63] (рисунок 2.15). Автоматы Мили реализуют аналитические зависимости вида:

$$a(t+1)=F_1(a(t), \alpha_1 \alpha_2 \dots \alpha_q); \quad A(t)=F_2(a(t)), \alpha_1 \alpha_2 \dots \alpha_q). \quad (2.4)$$

Заметим, что для автоматов Мили в соответствии с уравнениями (2.4) как для схемы  $F_1$ , так и для схемы  $F_2$  подается полная конкатенация переменных

$$\{\alpha\}\{x\} - \alpha_1 \alpha_2 \dots \alpha_q x_1 x_2 \dots x_m .$$



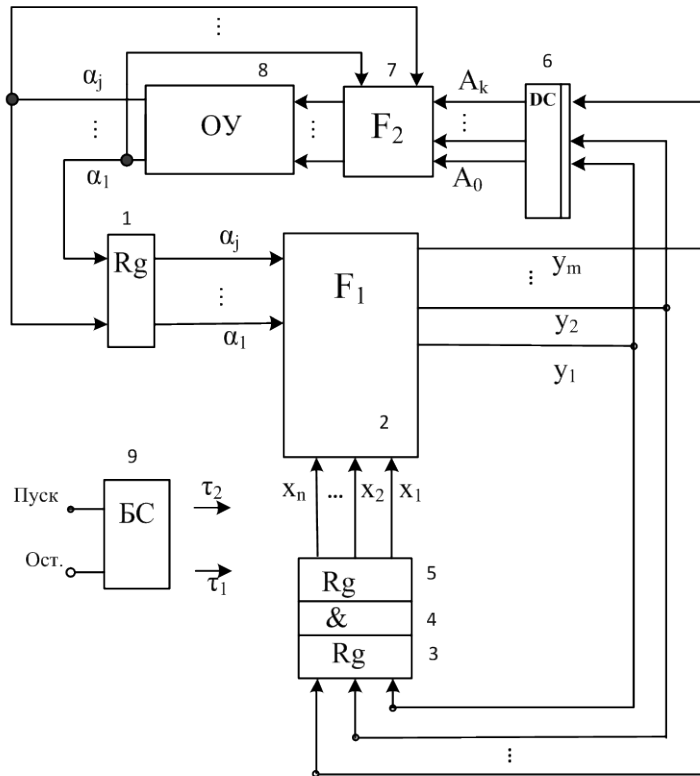


Рисунок 2.15 - Автомат Мили

В соответствии с известными правилами разметки в графе переходов УА Мили число состояний меньше, чем в графе переходов УА Мура (рисунок 2.16а, б). Соответствующие графы переходов УА Мура и Мили представлены на рис.2.17 а, б.

В УА Мили меньшее число состояний, но формирование выходных сигналов  $A_i \in \{A\}$  происходит не после перехода из состояния  $a(t)$  в состояние  $a(t+1)$ , а в начале перехода (в момент фронта сигнала синхронизации) из состояния  $a(t)$  в состояние  $a(t+1)$ . А поскольку переход возможен (при наличии условия  $\alpha_j \in \{\alpha\}$ ) по одному из двух путей в соответствии со значением  $\alpha_j$  и  $\bar{\alpha}_j$ , то команды управления формируются не независимо для  $\alpha_j$  и  $\bar{\alpha}_j$  при переходе в одно общее состояние  $a(t+1)$ . Причем длительность команд  $A_i \in \{A\} = \tau$  (в УА Мура длительность команд равна  $T$ ).

Применим предложенную в диссертации методику синтеза, т.е. точно также вставим пустые операторы между любыми  $\alpha_j$  и  $\alpha_j \in \{\alpha\}$  не разделенными в ГСА операторами действия  $A_i \in \{A\}$ .

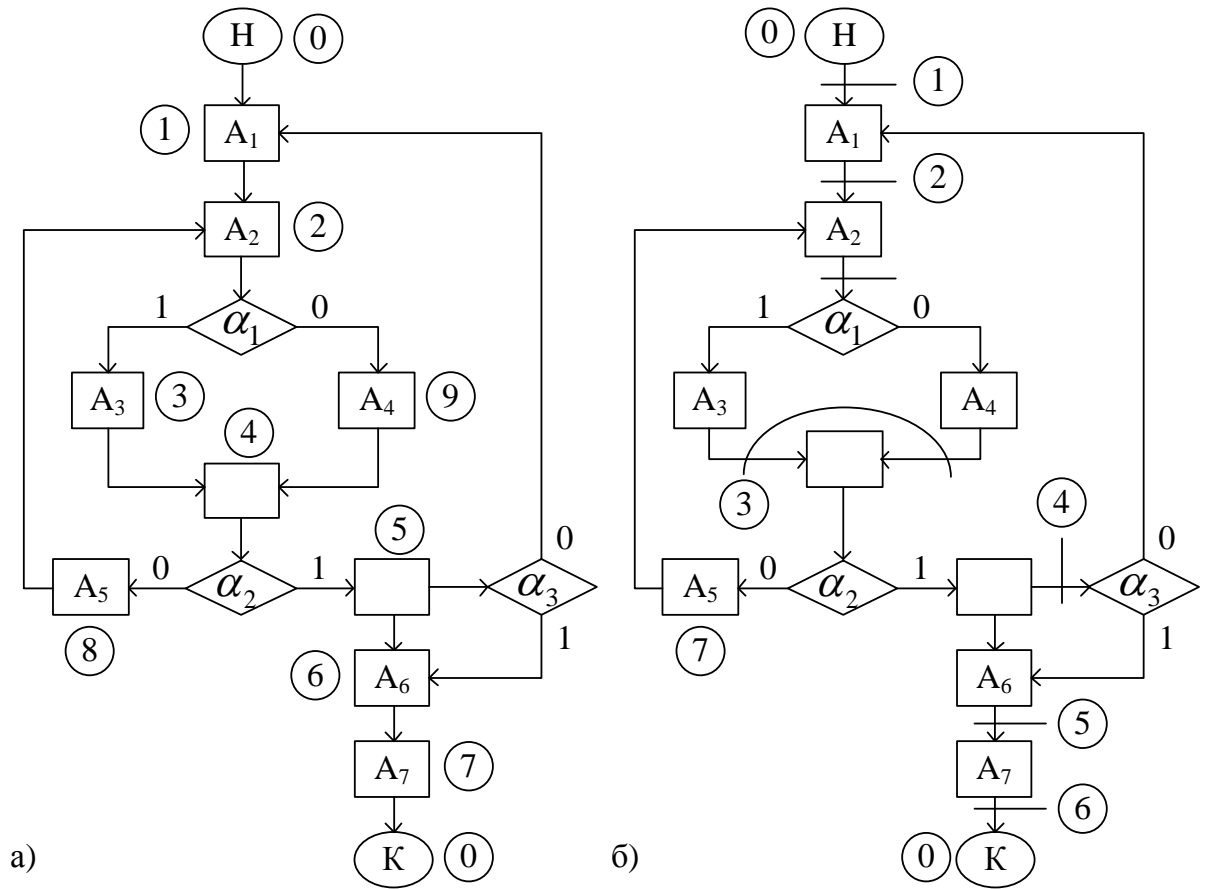
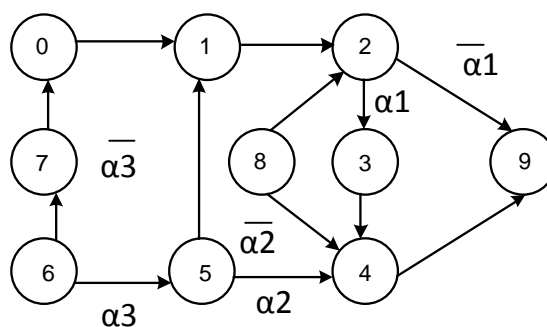
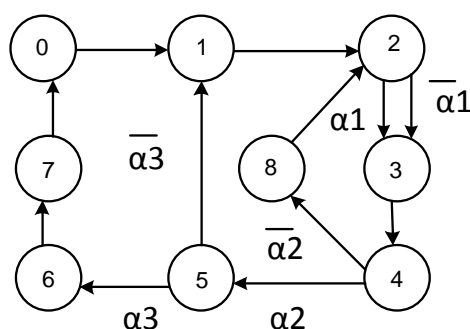


Рисунок 2.16 – Введение пустых операторов в ГСА:  
 а - для УА Мура, б - для УА Мили



а)



б)

Рисунок 2.17 – Графы переходов:

а) – для УА Мура, б) - для УА Мили

На рисунке 2.18 представлена модифицированная ГСА рисунка 2.7 с разметкой состояний по правилу Мили. Отметим, что в отличие от автоматов с правилом Мура, в модифицированной ГСА нет необходимости вводить пустой оператор, если к  $\alpha_j \in \{\alpha\}$  передается управление от двух и более операторов. Однако если ввести пустой оператор, то уменьшится количество входов к последующим состояниям, получившим управление от двух независимых операторов. Несмотря на увеличение количества состояний в графе переходов УА Мили, общее число связей в ГСА не увеличилось, однако для УА Мили ГСА стала более однородной. Поэтому и для автоматов с правилом формирования команд по переходам Мили также сохраним правило введения пустого оператора, если к  $\alpha_j \in \{\alpha\}$  осуществляется передача управления от нескольких операторов.

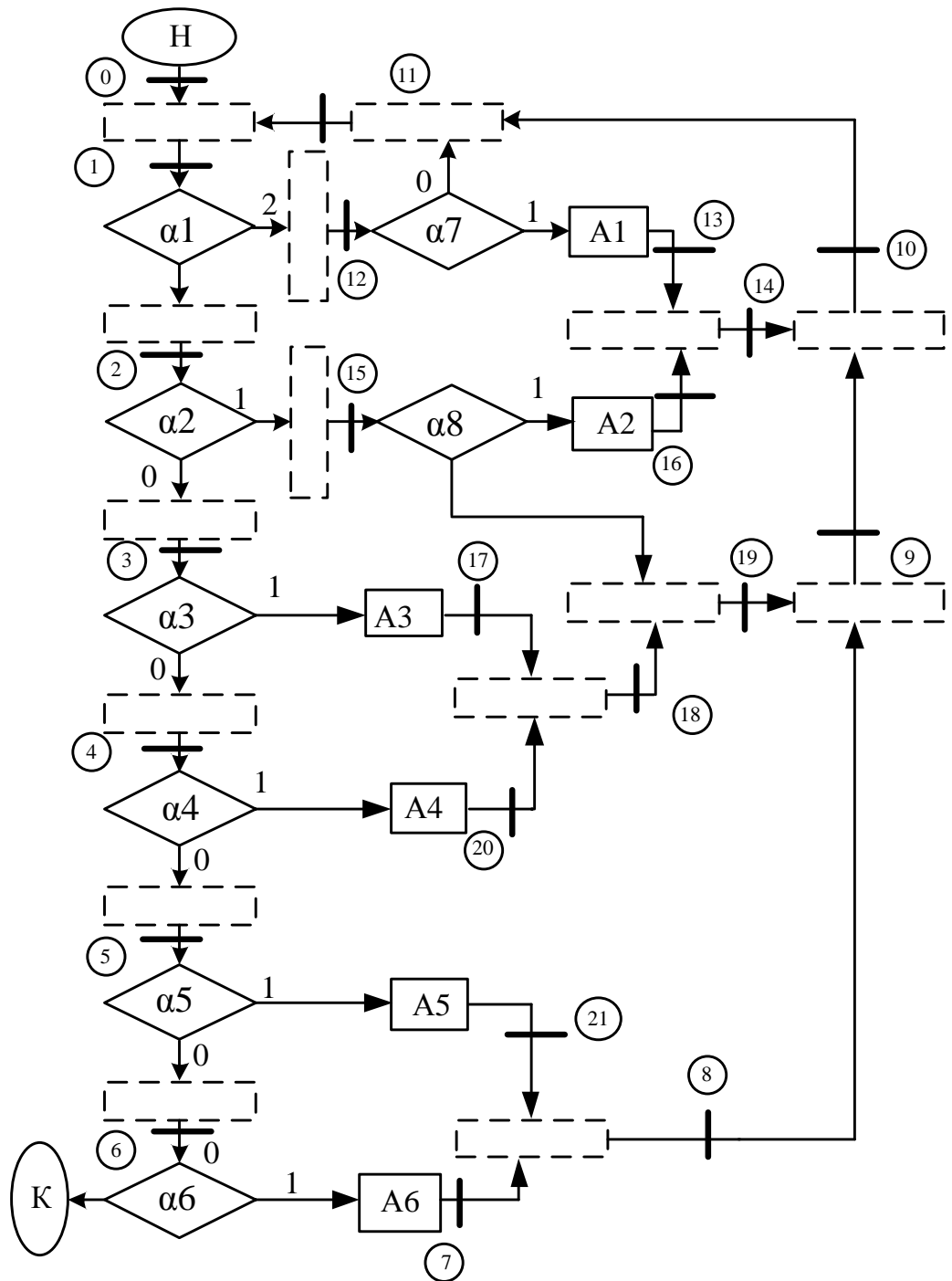


Рисунок 2.18 – Модифицированная ГСА с правилом разметки Мили

УА с мультиплексором позволяют существенно упростить совместную реализацию схем  $F_1$  и  $F_2$ . Если каждому состоянию  $a(t)$  соответствует только одно  $\alpha_j(t)$  из всего множества  $\{\alpha\}$  или безусловный переход  $\alpha_0$ , то функциональную схему  $F_1$  и  $F_2$  можно представить в виде рисунка 2.19.

Сигналы  $\alpha_j$  ( $j=1, \dots, p$ ) состояний  $a(t)$  разделяются схемами «И», которые соответствуют переходам  $a(t) \rightarrow a(t+1)$ . Переходы в одно состояние  $a_j(t+1)$  объединяются в первом блоке схем ИЛИ1 (V1) и через дешифратор CD1 образуют выходы  $y_1 y_2 \dots y_m$ , а через второй блок схем ИЛИ2 (V2) формируются выходные сигналы на основании графа переходов и выходных функций. В схеме рисунка 2.18  $\alpha_j$  и  $\overline{\alpha_j}(t) \in \{\alpha\}$  подаются одновременно на схемы «И» всех тех выходов  $a_1, a_2, \dots, a_p$ , которые образуют переходы без  $\alpha_0$ , но срабатывают лишь те, к которым будет подан импульс  $a_i(t)$  по сигналу синхронизации  $\tau$  для дешифратора DC.

Конкретное подключение к выходам  $a_i(t)$  и к входам блоков схем ИЛИ1 (V1), ИЛИ2 (V2) очевидно из анализа графа переходов (рисунок 2.20). Количество пар схем «И» при  $a(t)$  определяется не общим числом всех состояний автомата, а числом тех состояний в графе переходов, выходы которых разделяются на два по условиям  $\alpha_j \in \{\alpha\}$ . Это число всегда равно величине  $q$ .

Такая структурная организация блока  $F_1$  и  $F_2$  позволяет дополнительно уменьшать объем оборудования комбинационных схем автомата Мили с мультиплексором [73].

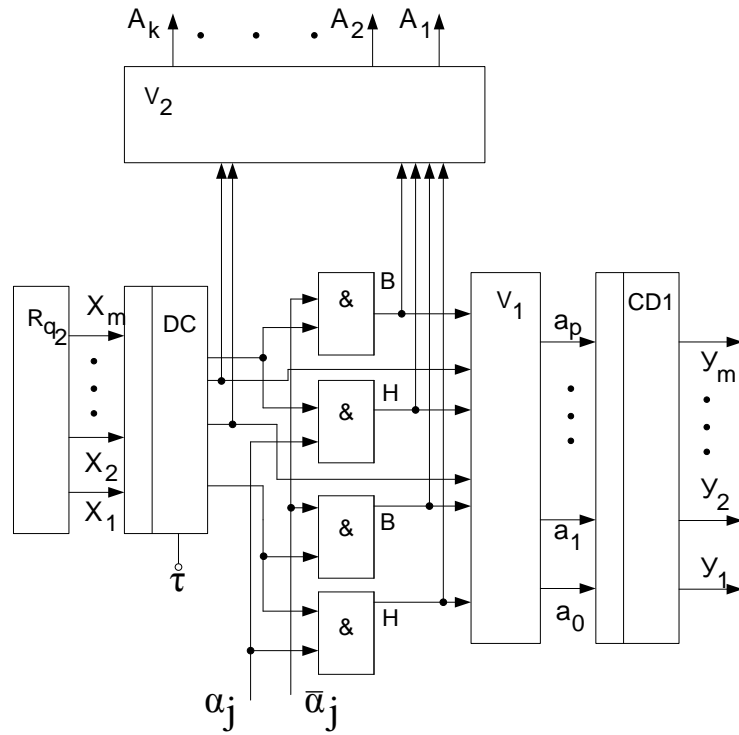


Рисунок 2.19 - Комбинационная схема автомата Мили с мультиплексором

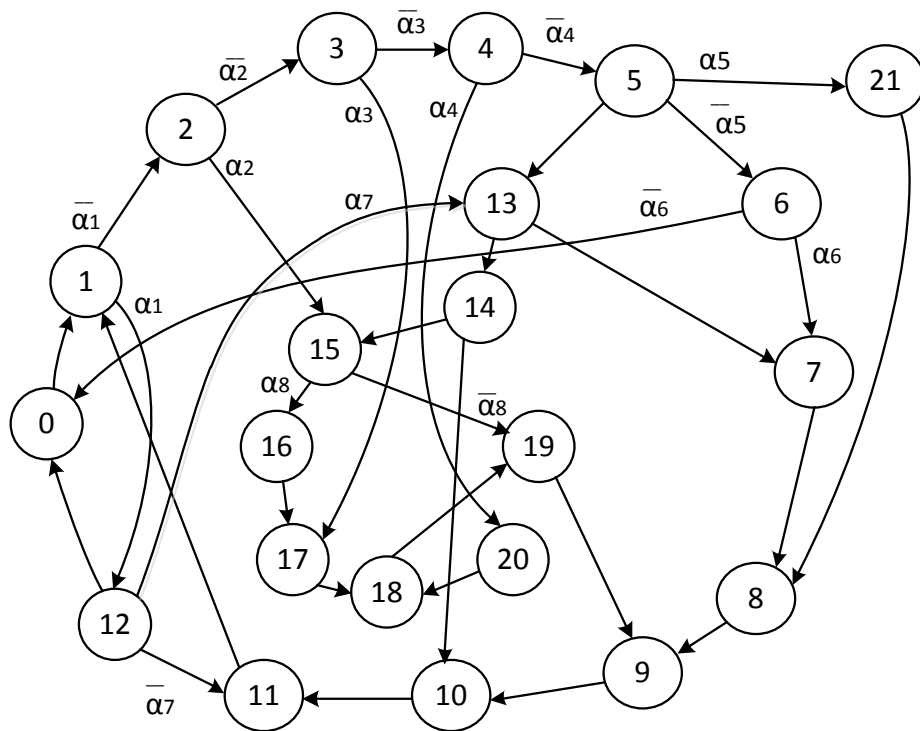


Рисунок 2.20 - Граф переходов УА Мили с мультиплексором

## 2.8. Синхронизация управляющих автоматов нового типа

Цель разметки ГСА для УА Мили состояла в том, чтобы обеспечить формирование  $A_i \in \{A\}$  по переднему фронту импульса синхронизации. Этого можно добиться и в том случае, если разметку ГСА провести по правилу УА Мура, т.е. отметить связи после каждого  $A_i \in \{A\}$ , и не сводить к одному состоянию переходы  $a(t+1)$  от операторов  $\alpha_j A_i$  и  $\bar{\alpha}_j A_r$ . Тогда число состояний для автоматов нового типа как в УА Мили, так и УА Мура окажутся равными. УА Мили и УА Мура потребуют лишь разных способов синхронизации при включении  $A_i \in \{A\}$ . При этом функции формирования переходов ( $F_1$ ) и функции формирования выходных команд  $A_i$  у них будут одинаковыми.

При конструктивном (инженерном) подходе к проектированию, переход к перестраиваемой синхронизации за счет введения дополнительных состояний в УА с правилом формирования сигналов на переходе для нового типа УА совершенно оправдан, если учесть следующие факторы:

- в классических автоматах Мили снижение числа состояний одновременно определяло и снижение затрат на комбинационные схемы, особенно для СП и ПА автоматов. Одновременно решалась сложная наукоемкая задача объединения состояний УА. Тогда объем оборудования снижался в 1,2-1,3 раза, т.к. объем оборудования существенно зависел от числа состояний.

Для нового типа УА даже для сверхпростых автоматов ( $q=5$ ) за счет введения мультиплексора сложность комбинационной схемы снижается  $\approx$  в 16 раз ( $Q = 2^{q-1}$ ), если оценивать ее через объем ПЗУ. Такая результативность для ПА автоматов не достижима ни при каком способе снижения числа состояний в классических автоматах. Для более сложных УА эффективность значительна (таблица 2.3). Поэтому для УА нового типа сама постановка наукоемкой задачи объединения состояний с целью сокращения их числа может быть только теоретически интересной, но конструктивно не целесообразна.

## 2.9. Гибридный способ синхронизации управляющих автоматов

В ряде случаев время выполнения некоторых операторов действия  $A_i \in \{A\}$  может существенно превышать период  $T$ . Такая ситуация возникает при необходимости обращения к подпрограммам. В этом случае кроме таблицы выбора  $\alpha_j \in \{\alpha\}$  по коду  $a(t)$  необходимо сформировать таблицу определения тех  $a_i(t)$ , при которых необходим переход к отключению генератора тактовых импульсов и перехода в асинхронный режим, т.е. включение режима ожидания окончания оператора действия  $A_j$ , соответствующего подпрограмме. В самом УА изменится только блок синхронизации. В остальном структурная организация ФИЛАУ для УА нового типа не изменится.

Рассмотрим алгоритм управления расчетом верности передачи информации в сети ЭВМ [36]. На рисунке 2.21 подпрограммы  $\Pi_i$  помечены двойным контуром. После модификации получим ГСА рисунок 2.22 и граф переходов рисунка 2.23. В состояниях 3,6,9-11 необходимо осуществлять переключение на асинхронный режим исполнения  $A_i \in \{A\}$ . УА с подпрограммами за счет специальной организации блока синхронизации является гибридным, т.к. может работать как синхронный, так и асинхронный автомат.

Приведенный в главе 2 список литературы по анализу и синтезу дискретных автоматов можно назвать минимально необходимым. Преимущественное обращение к ранним работам определяется тем, что соискатель не затрагивает особый пласт изданий по вопросам общей теории автоматов, языков и вычислений, т.к. они не имеют конструктивной направленности. Детальная библиография по этому вопросу могла бы составить несколько сот публикаций. В работах Соловьева В.В. [5] и Сапожниковых В.В. и Вл.В. [8] приведены полные систематизированные списки литературы по вопросам синтеза дискретных автоматов. Публикации после 1990-х годов посвящены преимущественно «приспособлению» известных структур УА к интегральной элементной базе различного уровня интеграции включая СБИС, ПЛМ и ПЛИС.



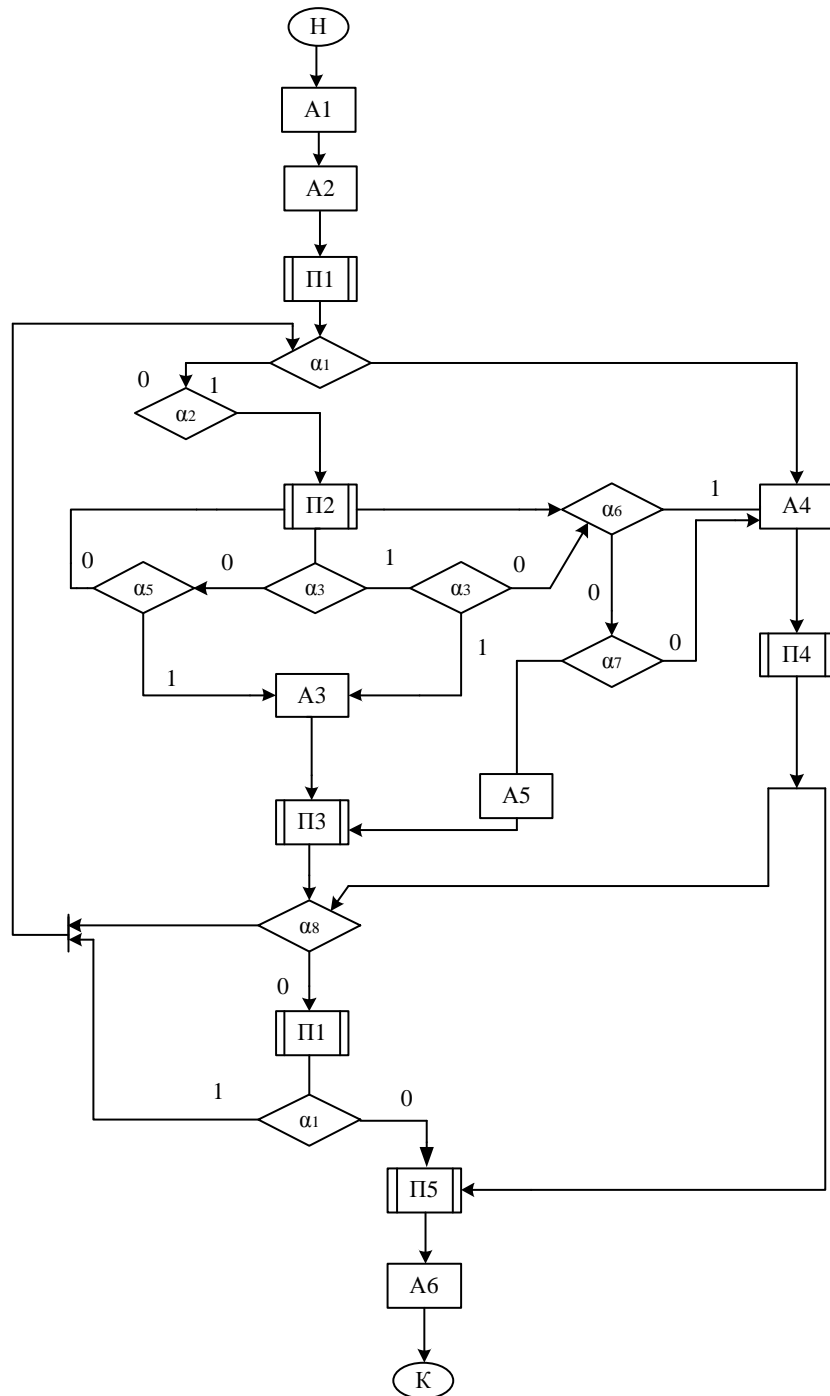


Рисунок 2.21 - ГСА управления расчетом верности передачи информации в сети ЭВМ

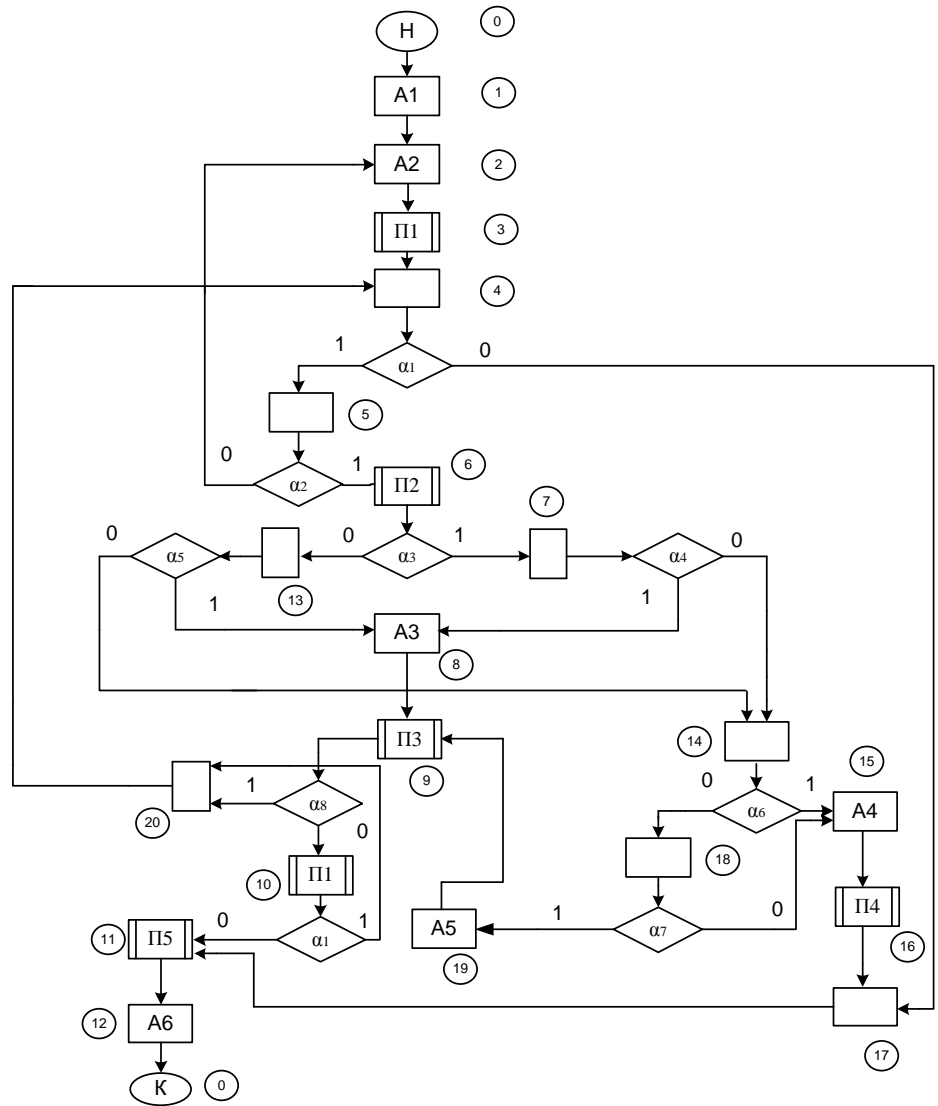


Рисунок 2.22 – Модифицированная ГСА управления расчетом верности передачи информации в сети ЭВМ

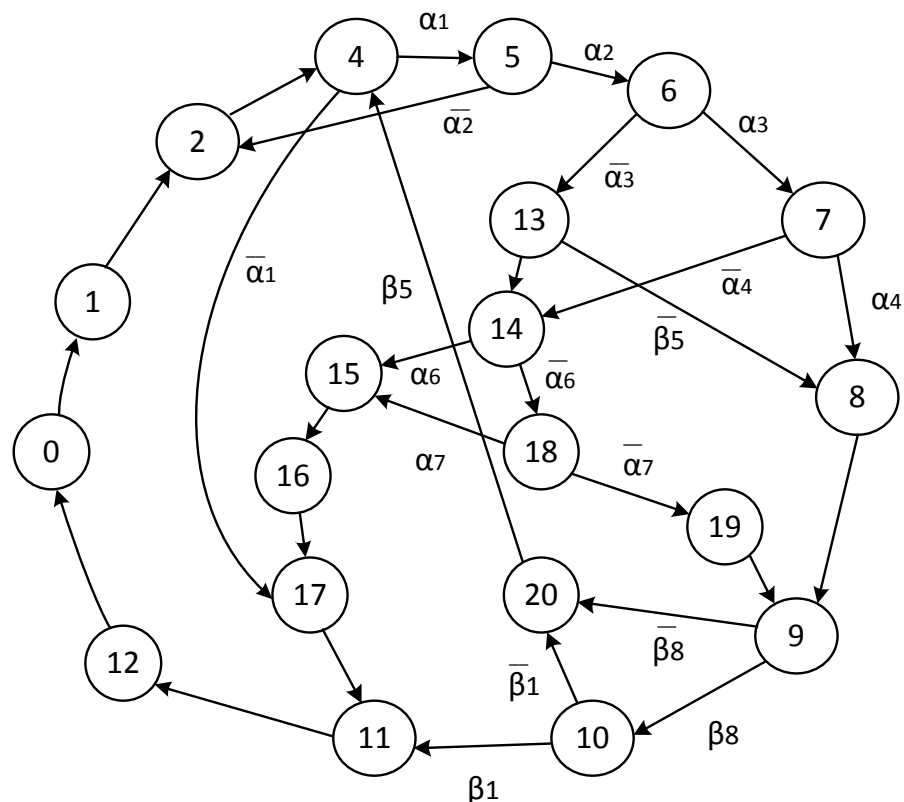


Рисунок 2.23 – Граф переходов управления расчетом верности передачи информации в сети ЭВМ

### Выводы по главе 2

1. Предложена новая методика синтеза управляющих автоматов для сложных технических систем и технологических процессов.
2. На основе новой методики синтеза предложены несколько вариантов структурной организации управляющих автоматов нового типа, отличающихся пониженным в  $2^{q-1}$  раз объемом ПЗУ для реализации комбинационных схем переходов.
3. Показана применимость и высокая эффективность новой методики, как для автоматов Мура, так и для синтеза автоматов Мили и их комбинаций.
4. Варианты структурной организации автоматов нового типа оригинальны, на них получено три патента.

По материалам главы 2 автором опубликованы работы [5,6,8,27,30,33,38,42,45,51,71] и получены патенты [62,65].

## ГЛАВА 3. СИНТЕЗ КОМПЛЕКСНЫХ АВТОМАТОВ

### 3.1. Структурная организация сложных автоматов

При решении задач синтеза УА с  $m \geq 5$  и  $q \geq 12$  ( $m+q > 17$ ) для минимизации систем булевых функций используются эвристические методы, т.к. точное решение задачи требует полных переборов [77,78]. Для того, чтобы уйти от таких задач переходят к замене одного автомата комплексом более простых взаимодействующих автоматов.

Комплексный автомат может рассматриваться в следующих аспектах:

1. Организация автоматов с выделением одного ведущего и нескольких ведомых автоматов (иерархическая структура). Действительно, во всех ЭВМ используется двухуровневая система из устройств программного и микропрограммного управления с перестройкой по коду микрооперации [48].

2. Единый автомат, но выполняющий функции нескольких автоматов с перестройкой на заданные функции УА (i) по коду (i). Эта задача требует решения вопросов выбора наилучшего способа объединения (композиции) автоматов УА(i),  $i = 1, 2, \dots, n$  [79].

3. Организация системы автоматов. Задача связана с решением оптимальной декомпозиции автоматов. Декомпозиция УА рассматривалась в самых ранних работах [32,37,39,44-81], исследования продолжаются в различных приложениях [82-85].

4. Организация единого УА за счет включения в его структуру более простых автоматов. Такое направление также исследуется на протяжении всего развития теории и инженерной практики конструирования УА. В частности, в работе [86] УА представляется с памятью в виде счетчика, коэффициент пересчета которого многократно перестраивается в процессе работы.

5. Организация недетерминированных, ассоциативных и параллельных автоматов [52,87-90].

6. Организация однородной сети автоматов [15,91-93], клеточные автоматы [95] и УА на основе нейронных сетей [16, 94].

Последние 2 направления в диссертации не рассматриваются.

В ряде работ использовались частные, но оригинальные структурные организации комплекса автоматов. В работе [96] после каждого оператора действия  $A_i$  проверялось логическое условие, что потребовало включения в структуру УА комплекса независимых счетчиков. В работе [22] при реализации УА на ПЛМ в виде ферритовых или диодных схем использовалась двухуровневая организация с памятью на регистрах с одной сдвигаемой «1».

В работах [82,97-99] УА реализуется через декомпозицию на два взаимодействующих автомата, в котором один с классической схемой Мура реализует логические переходы в УА, а второй (Мили) осуществляет последовательно во времени линейную развертку подмножества микроопераций в текущей команде  $A(t)$ . Причем  $A(t)$  считается линейной, если подмножество микрокоманд состоит не менее чем из трех последовательно выполняемых микрокоманд  $C_i, C_j, C_k \in \{C\}$  без логических переходов между ними.

Организация комплексных автоматов базируется на использовании двух оптимизационных задач:

- композиции графов переходов и
- декомпозиции графов переходов.

Как видно, оригинальность этих решений определяется следующими аспектами:

- конкретной спецификой ГСА;
- использованием процедуры декомпозиции;
- использованием унитарного кодирования состояний.

Методики синтеза комплексных автоматов разрознены и ориентированы лишь на частные постановки.

Вычисление  $a(t+1)$  с учетом единственного  $\alpha_j \in \{\alpha\}$  позволяет упростить структурную организацию комплексного УА за счет модификации новой методики синтеза для всех трех постановок.

### 3.2. Синтез управляющих автоматов с импульсно-кодовой памятью состояний

Упрощение структурной организации УА за счет использования регистров сдвига с одной единицей или комплекса счетчиков использовалось в СП и ПА автоматах. Для более сложных УА введение счетчика не снижает объем комбинационных схем. Поэтому для ВС автоматов осуществляется декомпозиция УА с учетом всего множества входных переменных  $\{\alpha\}\{X\}$ .

При использовании новой методики абстрактного синтеза УА комбинационная схема существенно упрощается для любых типов ГСА, если применить память состояний автомата в виде параллельной пары регистр-счетчик. Такой тип памяти УА можно назвать импульсно-кодовым, т.к. изменение состояния  $a(t)$  производится двумя способами:

- передача вычисленного параллельного кода  $a(t+1)$  на регистр памяти;
- формирование импульса изменения состояний счетчика для подачи на счетный вход. При использовании универсального счетчика (суммирующе-вычитающего) одновременно формируется сигнал выбора суммирующего или вычитающего входа.

Такая память автомата является сложной, т.к. вместо второго регистра памяти для кода  $a(t)$  используется линейный автомат – универсальный счетчик с параллельными входами приема кода  $a(t)$ .

Сохраним предложенную методику абстрактного синтеза УА с использованием одного логического условия  $\alpha_j \in \{\alpha\}$ , выбираемого по коду состояния  $a(t)$ .

Для комплексных автоматов следующие пункты методики синтеза являются обязательными:

1. Если на графе переходов УА имеются петли, то введем пустую вершину перед той вершиной, к которой осуществляется возврат по логическому условию (рисунок 3.1)

2. На графе переходов выделяются непрерывные пути с последовательным изменением состояний, в которых для каждого соседних состояний выполняется условие  $a(t+1) = \tilde{a}_j a(t) + 1$ .

3. В выделенном пути последовательных переходов заменим  $\bar{\alpha}_j$  на  $\beta_j = \bar{\alpha}_j$ , если переход от  $a(t)$  к  $a(t+1)$  осуществляется по условию  $\bar{\alpha}_j$  (рисунок 3.2). Длинному пути сопоставляется суммирующий счетчик. Поскольку при использовании суммирующего счетчика  $a(t+1) = (\alpha_0 + \alpha_i + \beta_j)a(t)$  и при этом номер логического условия не учитывается (т.к. он нужен только для адресации мультиплексора), то условие определения необходимости прибавления «1» к содержимому счетчика ( $\gamma$ ) может вычисляться через простую дизъюнкцию кодов состояния автомата.

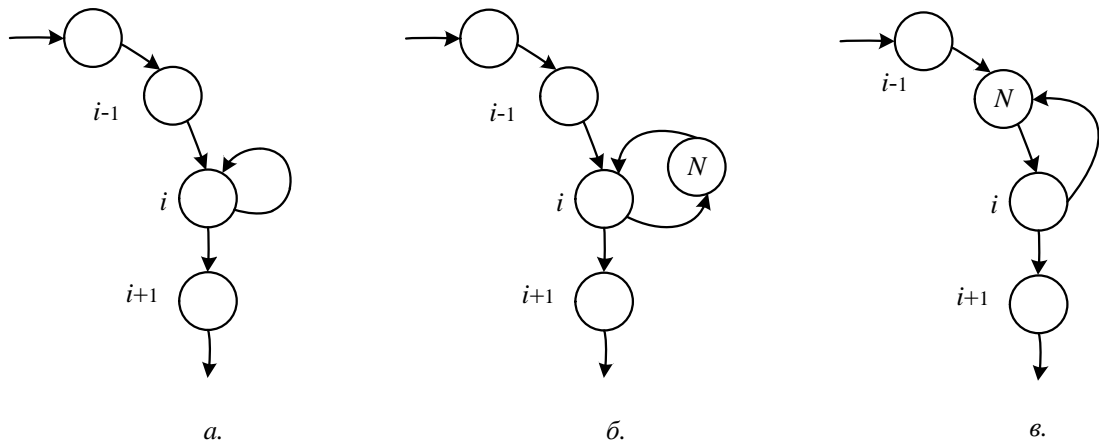


Рисунок 3.1- Способ ликвидации петель на графе переходов:

а – участок с петлей, б – применяемый способ, в – предложенный способ.

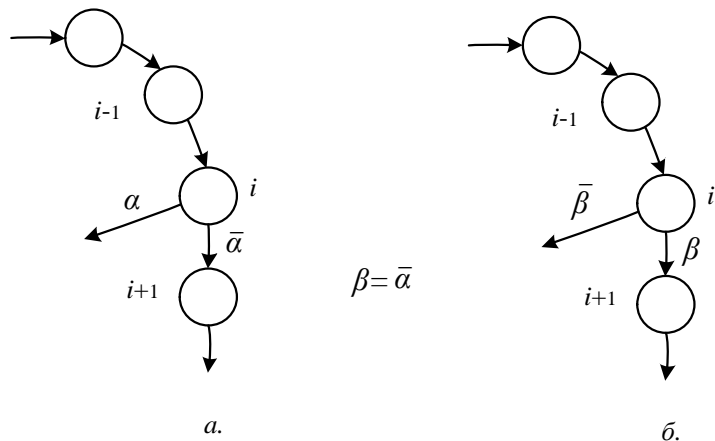


Рисунок 3.2- Способ замены  $\bar{\alpha}$  на  $\beta$  на графе переходов.

4. Если переход к исходному состоянию  $a(t) = 0$  осуществляется не от оператора действия, а от логического условия, то ставится промежуточный пустой оператор.





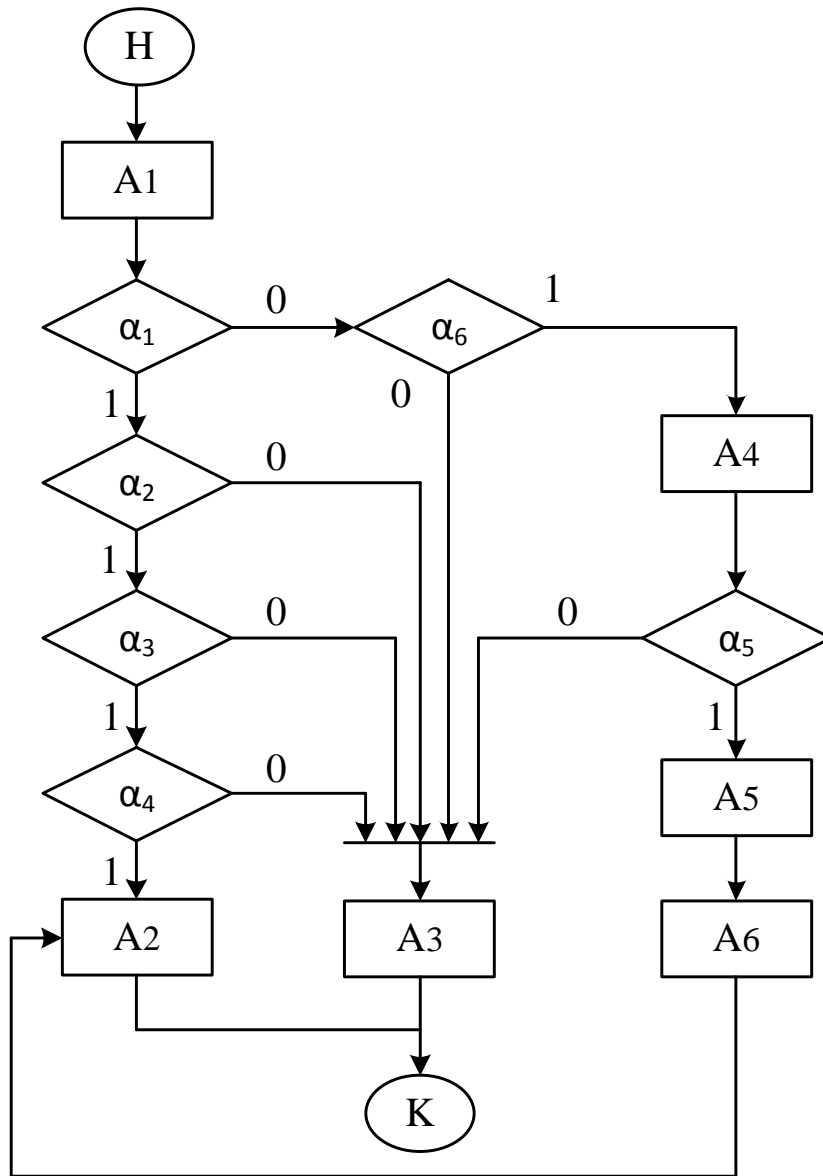


Рисунок 3.4 - Алгоритм управления интерфейсом

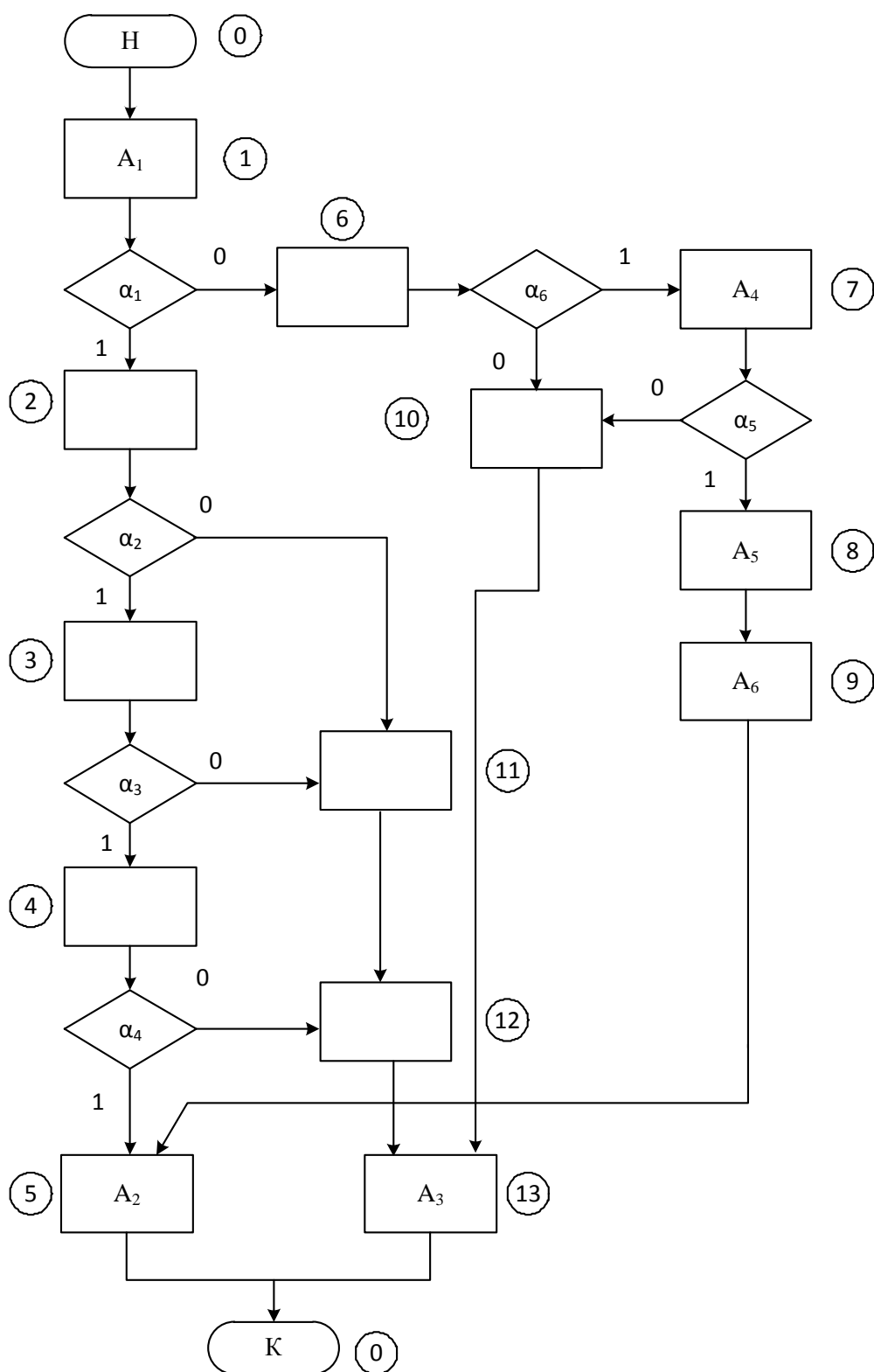


Рисунок 3.5 - Преобразованный алгоритм управления интерфейсом

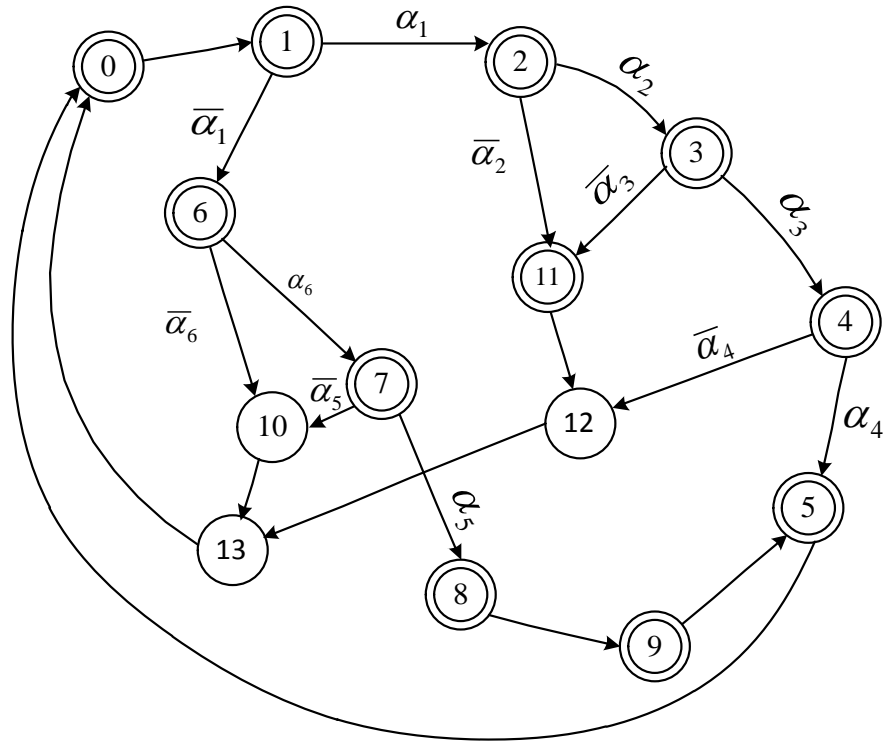


Рисунок 3.6 - Граф переходов алгоритма управления интерфейсом

Таблица 3.1. Таблица переходов

№	a(t)	$\alpha$	a(t+1)
1	1	$\bar{\alpha}_1$	6
2	2	$\bar{\alpha}_2$	11
3	3	$\bar{\alpha}_3$	11
4	4	$\bar{\alpha}_4$	12
5	5	$\alpha_0$	0
6	6	$\bar{\alpha}_6$	10
7	7	$\bar{\alpha}_5$	10
8	9	$\alpha_0$	5
9	10	$\alpha_0$	13
10	13	$\alpha_0$	0

### 3.3. Метод синтеза многорежимных управляющих автоматов

Известные методы декомпозиции автоматов с классической структурной организацией связаны с представлением единой комбинационной схемы переходов, реализующей систему булевых функций  $F_1$  в виде параллельно-последовательной сети нескольких комбинационных схем. Эти комбинационные схемы в сети имеют меньшее число переменных по сравнению с конкатенацией кода состояний и полного множества логических условий. Обычно число разрядов соответствующих переменных удается снизить не более чем в 1,5-2 раза. При этом получается сеть с количеством ярусов равным 2-4.

Для автоматов нового типа, несмотря на беспрецедентное снижение сложности, наличие трех комбинационных схем ( $F_1$ - $F_3$ ) и сложного мультиплексора (М) при  $q > 16$  также появляется необходимость декомпозиции ГСА.

На первом этапе декомпозиции производится деление ГСА на две или три части. Общие основные затраты оборудования УА определяются как  $R(i) = F_1(i) + F_3(i) + D(i)$ . Здесь  $D(i)$  – дополнительное оборудование для  $i$ -го шага декомпозиции. Для  $i=0$ ,  $D(i) = 0$ . Декомпозиция прекращается, если  $R(i) \geq R(i-1)$ . В традиционных подходах после декомпозиции независимо реализуются УА для каждой части.

В этой главе предлагается подход, основанный на создании единого УА, в котором последовательно реализуются операторы  $A_i \in \{A\}$  с логическими условиями  $\alpha_j \in \{\alpha\}$  каждой части в строгом соответствии с переходами в исходном алгоритме. Это достигается формированием кодов номеров частей ГСА для каждого перехода. В таком УА требуется коммутатор для подачи подмножеств  $\{\alpha\}_k$  на информационные входы мультиплексора. Здесь  $k$  – номер декомпозированной части. При этом функции блоков  $F_2$  и  $F_3$  необходимо перестраивать по коду  $P_k$ .

Рассмотрим ГСА управления спецпроцессором контроля промышленных БИС ПЗУ с ультрафиолетовым стиранием информации [98] (рис. 3.7 и 3.8). На рис.3.7 обозначено: СППЗУ –перепрограммируемое ПЗУ; ФВД –формирователь

временных диаграмм [99]; БПР –блок принятия решений; ФЛС –формирователь логических сигналов; БИ- блок индикации, БУ – буферное устройство, ГТИ – генератор тактовых импульсов.

Для УА Мура потребовалась декомпозиция на 5 частей (рисунок 3.9.)

После преобразования ГСА получен модифицированный алгоритм рисунок 3.10, для которого  $m=6$ ,  $q=13$  (включая безусловный переход).

Декомпозируем ГСА на три части с кодовыми номерами  $P_1=01$ ,  $P_2 =10$ ,  $P_3 =11$ . Как видно из ГСА рисунка 3.11, для каждой части  $m=4$ ,  $q<8$ . Отдельные логические условия повторяются в разных частях. В таблице переходов 3.3 строки 2,4,8,14,17,18,21 также можно отнести к счетчику, тогда из 51 переходов в ГСА (рисунок 3.10) только 17 реализуются в схеме  $F_1$ . Причем для рассматриваемой ГСА как разрядность счетчика, так и разрядность выходов  $y_1 y_2 \dots y_m$  равна 4. В этом случае потребуется дешифратор (DC) не на 64, а всего лишь на 16 выходов. Соответственно в 2 раза снижается сложность мультиплексора, т.к. вместо 16 информационных входов потребуется всего 8. Схемы  $F_3$  и  $F_2$  в получившейся структуре УА более сложны за счет перестройки, но в целом УА упрощается за счет снижения сложности мультиплексора, дешифратора и др.

Для  $i=0$ ,  $m=6$ . Однако при реализации  $F_1$ ,  $F_3$  на ПЗУ или ПЛМ необходимо взять реальное  $m_p=8$ . Тогда  $W(0) = 1\text{кбит}$ ,  $F_3(0) = r2^m = 64\text{бит}$ ;  $R(0) = W(0) + F_3(0) + M(0)$ . Через  $W(i)$ ,  $F_3(i)$ ,  $M(i)$  обозначены затраты оборудования соответствующих схем. Для первого шага декомпозиции ( $i=1$ ) получим таблицу 3.2.

Таблица 3.2. Параметры частей УА

P	N	q	m
1	16	7	4
2	15	7	4
3	13	5	4

Общие затраты  $R(i) = 3W(i) + 3F_3(i) + M/2 + D(i) = 0.384 + 0.192 + M/2 + D(i) = 576\text{бит} + M/2 + D(i)$ . Дополнительное оборудование  $D(i) \approx M/2$ . Поэтому

сравнение УА можно проводить только по  $W(i)$ . Поскольку  $W(0) \approx 1$ кбит, а  $W(1) = 0.576$ кбит декомпозиция на три части целесообразна.

Как адрес (j) для выбора  $\alpha_j \in \{\alpha\}$ , так и переходы  $P_i(t) = \alpha_j P_{i-1}(t-1)$  необходимо формировать в комбинационной схеме  $F_3$  по таблицам 3.3, 3.4. Тогда вместо трех счетчиков на 4 разряда каждый (для рассматриваемого примера) достаточно использовать один 4-х разрядный счетчик. При этом для схемы  $F_3$  потребуется предусмотреть параллельную двухразрядную память кодов  $P_i(t)$  и  $P_{i-1}(t-1)$ . Выходные функции  $F_1$  в виде кода  $y_1 y_2 \dots y_m$  также станут 4-х разрядными, тогда как без декомпозиции  $m=6$  (рисунок 3.11). Следовательно, комбинационная схема  $F_1$  становится предельно простой. Тогда УА может быть реализован на одной ПЛИС [100,101].

В предложенной структурной организации нет необходимости реализовать УА через три независимых более простых автомата, а конструировать единый автомат с импульсно-кодовым представлением памяти и единой схемой переходов  $a(t+1) = F_1(\alpha_j x_1 \dots x_m)$ .

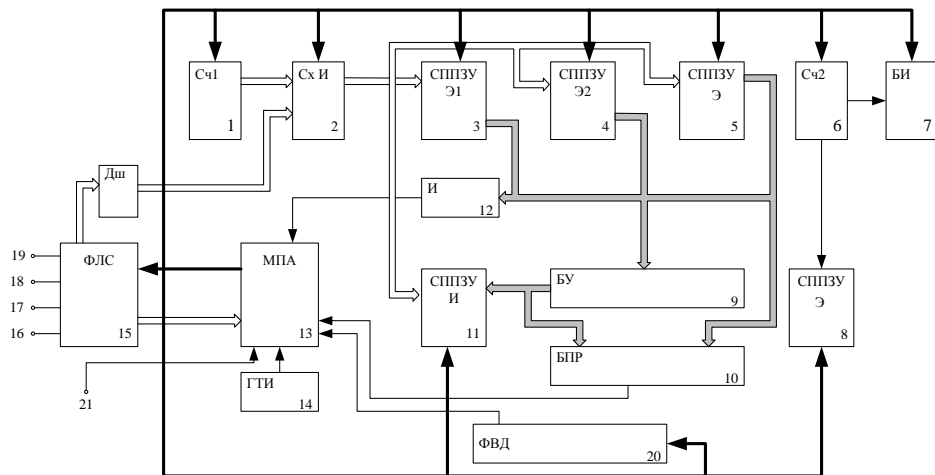


Рисунок 3.7 - Структурная схема системы контроля ПЗУ

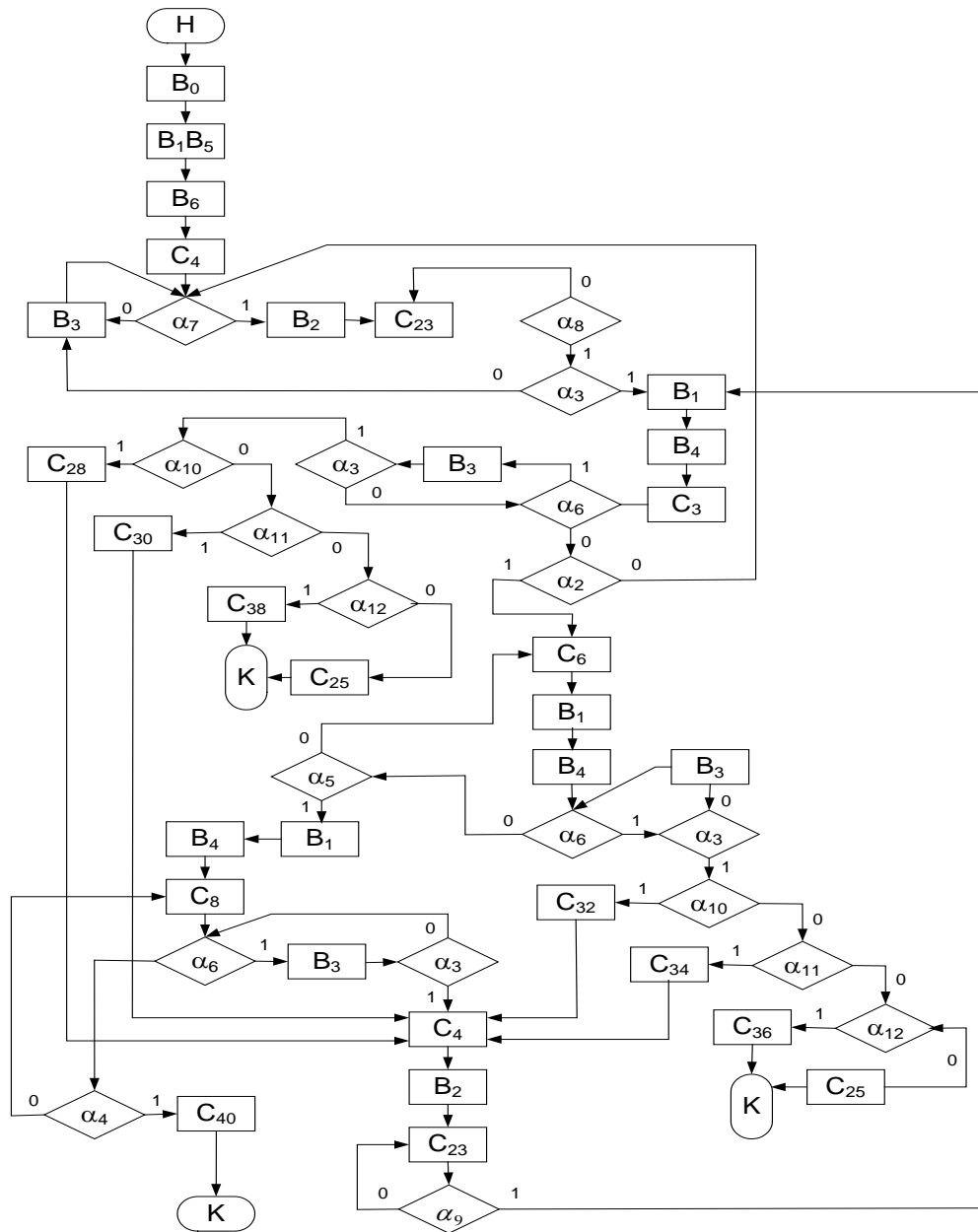


Рисунок 3.8 - Алгоритм управления спецпроцессором контроля БИС ПЗУ

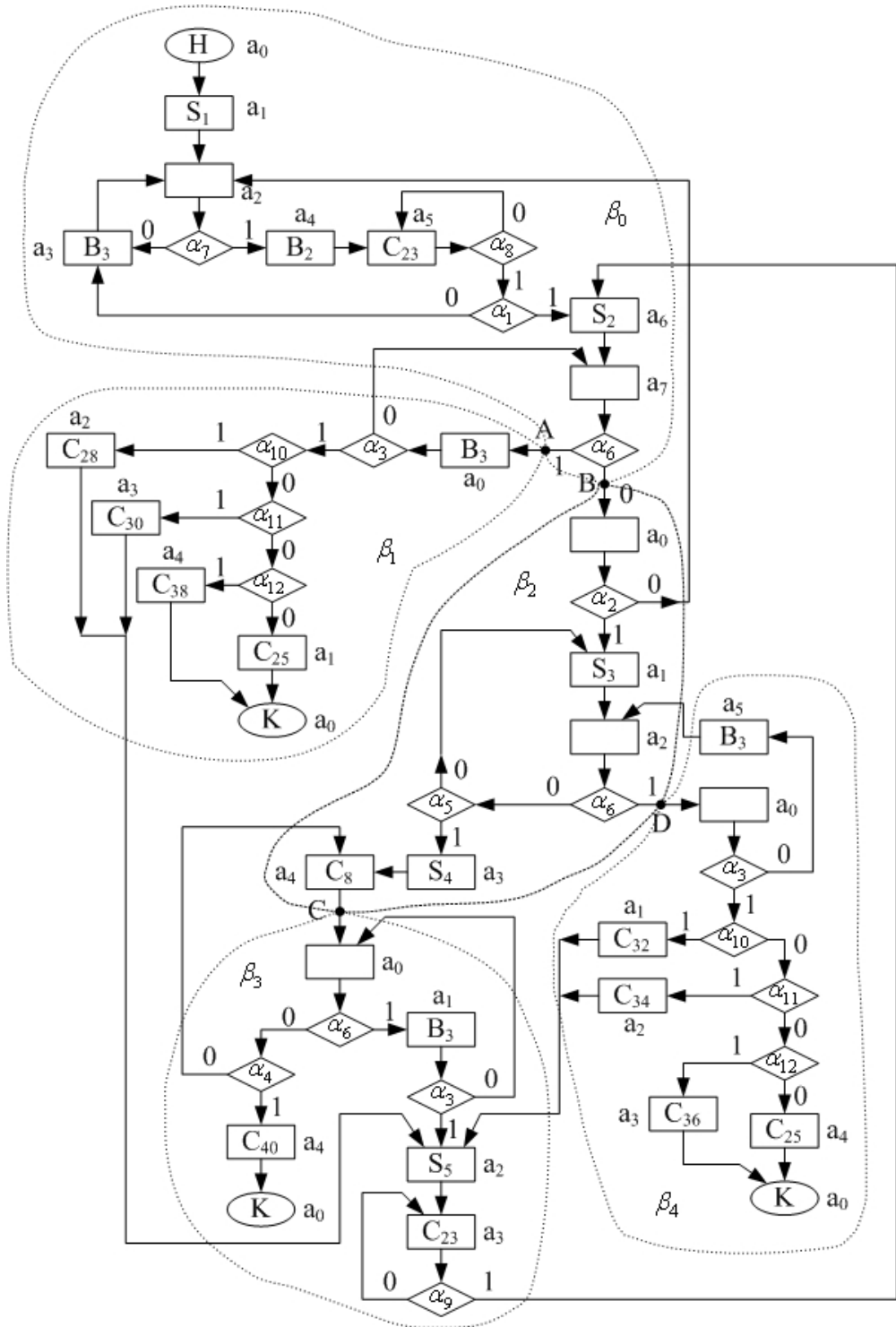


Рисунок 3.9 - Декомпозированный алгоритм управления системой контроля

для классического УА



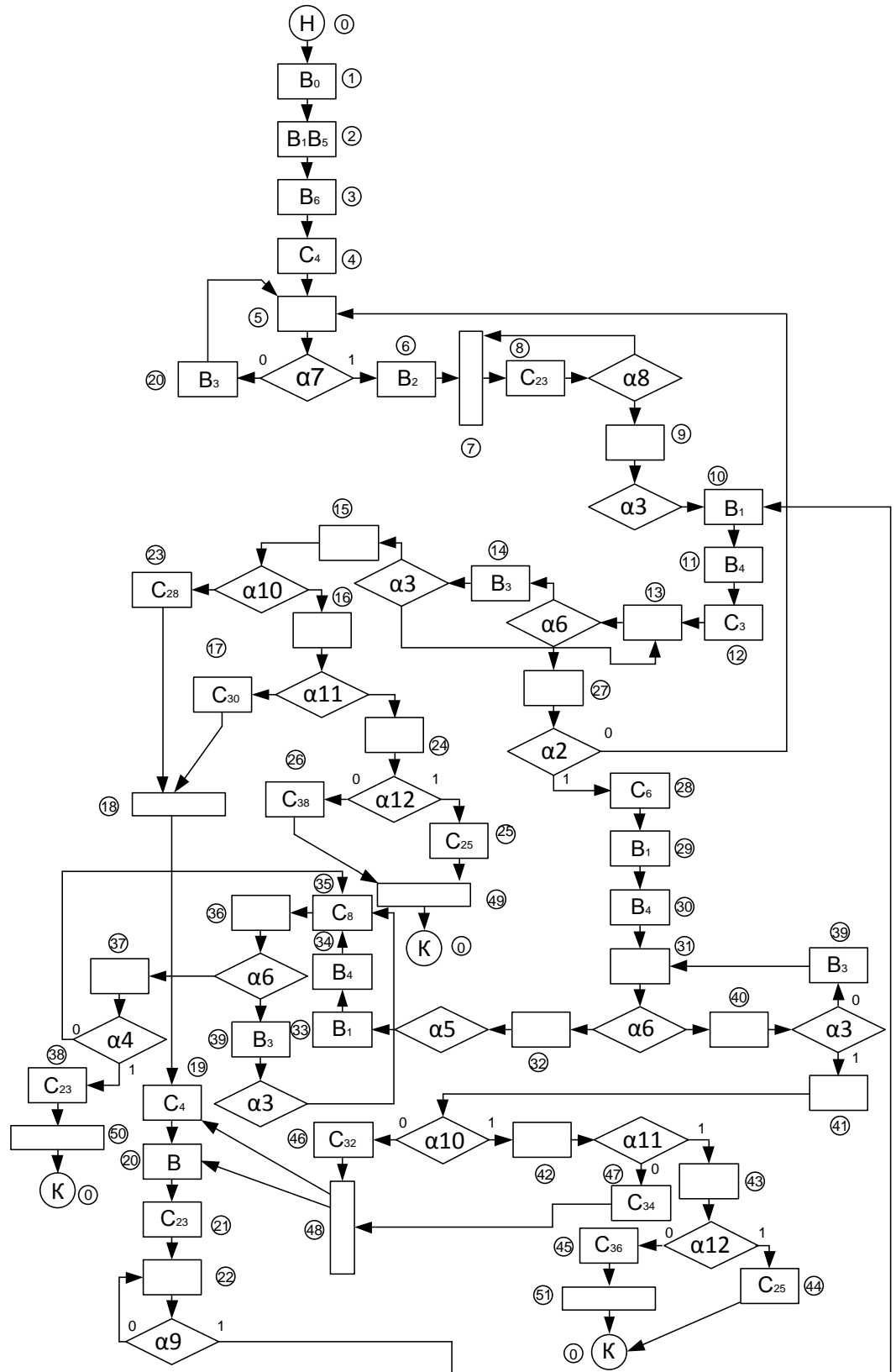


Рисунок 3.10 - Модифицированный алгоритм управления системой контроля

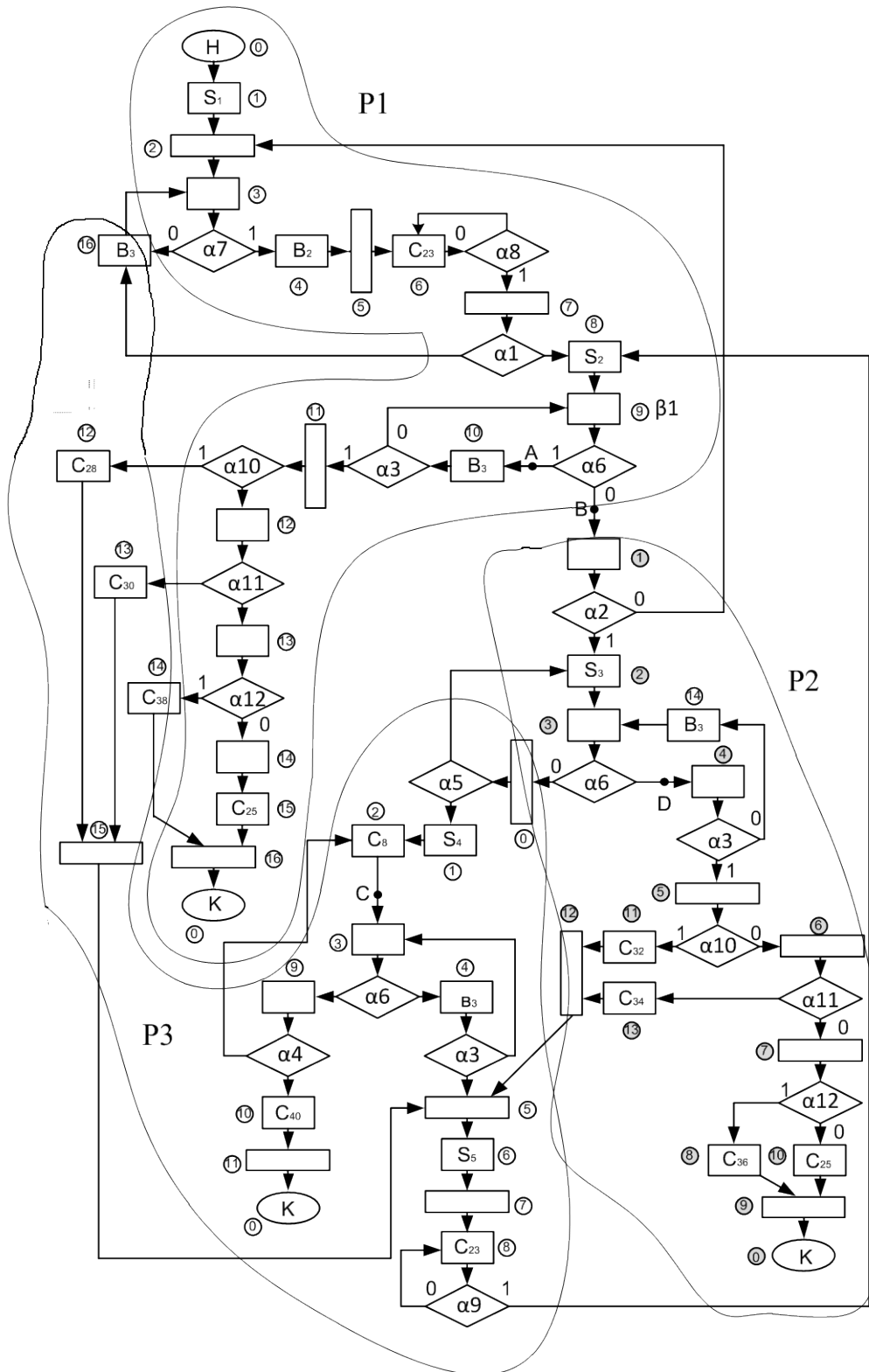


Рисунок 3.11 - Декомпозированный алгоритм управления системой контроля

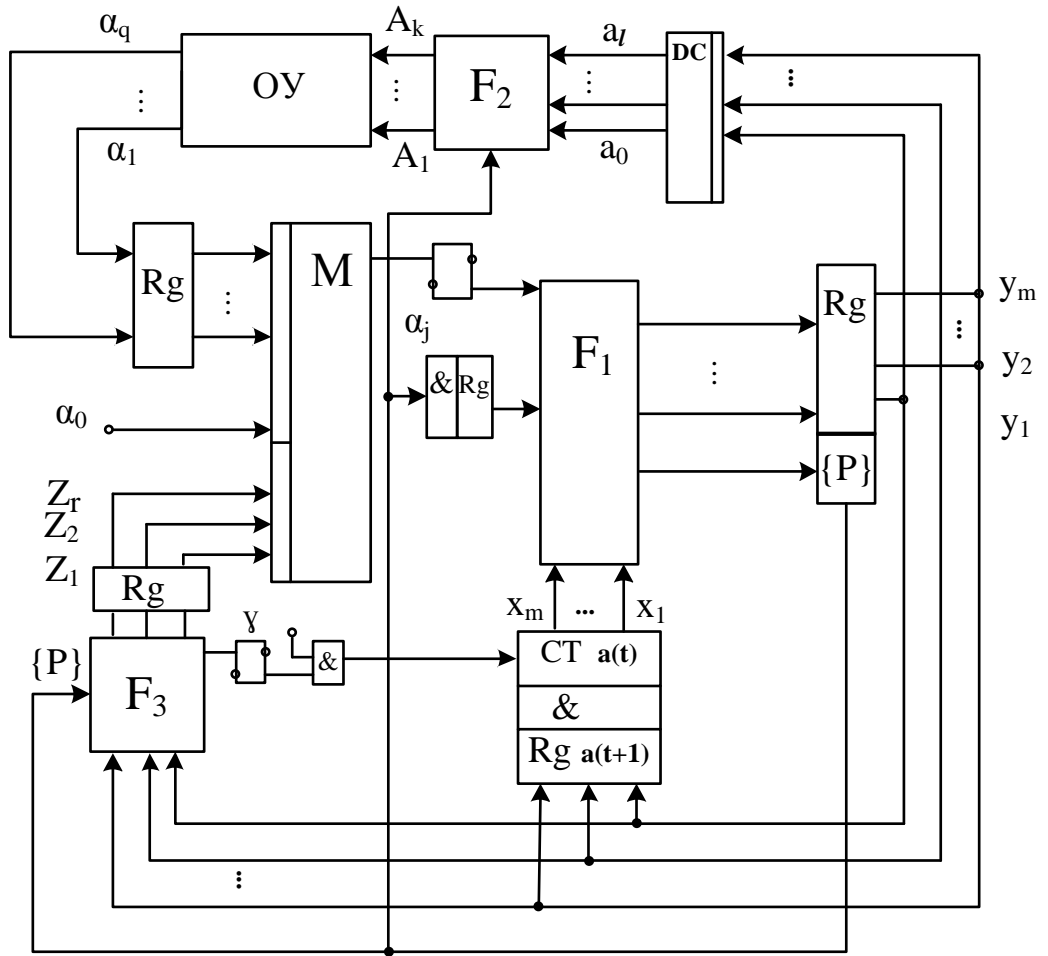


Рисунок 3.12 - Декомпозиционно-композиционный УА

Таблица 3.3. Функции переходов

№	$a(t)$	$\alpha$	$a(t + 1)$	СТ	$P_i(t)$	$P_{i+1}(t+1)$
1	3	$\overline{\alpha_3}$	15		1	1
2	15	$\alpha_0$	2		1	1
3	6	$\overline{\alpha_8}$	5	+	1	1
4	7	$\overline{\alpha_1}$	15		1	1
5	10	$\overline{\alpha_3}$	9	+	1	1
6	13	$\alpha_9$	7		3	1
7	12	$\alpha_{11}$	13	+	1	3
8	13	$\alpha_{12}$	15		3	3
9	15	$\alpha_0$	0		3	3
10	14	$\alpha_0$	0		3	3
11	4	$\alpha_6$	9		3	3
12	9	$\overline{\alpha_4}$	3		3	3
13	7	$\overline{\alpha_6}$	6	+	3	3
14	13	$\overline{\alpha_9}$	12	+	2	2
15	12	$\alpha_9$	7		2	2
16	1	$\overline{\alpha_2}$	3		2	1
17	8	$\alpha_0$	0		2	2
18	9	$\alpha_0$	0		2	2
19	6	$\overline{\alpha_{11}}$	14		2	2
20	3	$\alpha_6$	1		2	3
21	15	$\alpha_0$	3		2	2
22	9	$\overline{\alpha_4}$	3		3	3
23	1	$\overline{\alpha_5}$	2	+	3	2
24	14	$\alpha_0$	11		2	2
25	11	$\alpha_{10}$	11		1	3

Таблица 3.4. Функции выбора частей УА

№	$P_i$	$a(t)$	$\alpha$	$a(t+1)$	$P_{i+1}$
1	00	15	$\overline{\alpha_7}$	2	00
2	00	6	$\overline{\alpha_8}$	5	00
3	00	9	$\overline{\alpha_6}$	1	01
4	01	4	$\overline{\alpha_3}$	3	01
5	01	6	$\overline{\alpha_{11}}$	14	01
6	01	7	$\overline{\alpha_{12}}$	9	01
7	01	8	$\alpha_0$	0	01
8	01	7	$\alpha_{11}$	8	01
9	01	9	$\alpha_0$	0	01
10	01	13	$\alpha_9$	7	00
11	00	11	$\alpha_{10}$	11	10
12	00	12	$\alpha_1$	12	10
13	10	11	$\alpha_0$	13	10
14	10	12	$\alpha_0$	13	10
15	10	13	$\alpha_0$	8	10
16	10	8	$\alpha_0$	13	01
17	10	7	$\overline{\alpha_6}$	6	10
18	10	5	$\overline{\alpha_3}$	4	10
19	10	4	$\overline{\alpha_6}$	9	10
20	10	10	$\alpha_0$	0	10
21	00	10	$\overline{\alpha_3}$	9	00
22	10	9	$\overline{\alpha_4}$	3	10
23	01	12	$\alpha_9$	7	00
24	01	1	$\overline{\alpha_2}$	3	00

Для рассматриваемой ГСА декомпозиция на 4 -5 частей не дает эффекта, т.е. в использовании итерации ( $i=2$ ) нет необходимости. Очевидно, что для ОС и УС автоматов при декомпозиции может быть достигнут больший эффект.

Такая структурная организация декомпозиционно-композиционного УА не возможна для УА Мура. Новая организация УА является структурно-сложной, но одновременно достигается снижение затрат оборудования. Предложенный принципиально новый подход к синтезу сложных автоматов позволяет создавать эффективные и надежные управляющие подсистемы спецпроцессоров для технических систем и технологических процессов реального времени.

### 3.4. Синтез многопрограммных автоматов управления

Для многопрограммных автоматов входное множество логических сигналов представляет собой объединение подмножеств сигналов всех УА( $i$ ). Для реализации комбинационных схем УА( $i$ ) при классической реализации выделяются зоны памяти ПЗУ, соответствующие коду УА( $i$ ). Одновременно необходимо из входного множества сигналов выделять подмножество сигналов УА( $i$ ).

#### 3.4.1. Структурная организация многопрограммных автоматов

Для автоматов нового типа на информационные входы мультиплексора необходимо подать множество сигналов отдельно для каждой программы. Следовательно по коду ( $i$ ) блок адресации должен работать с соответствующим подмножеством. Структурная организация такого автомата представлена на рисунке 3.13. Структурная организация УА базируется на использовании двух коммутаторов: коммутатор КМ1 по коду микропрограммы ( $i$ ) подключает нужные  $\alpha_j \in \{\alpha\}$  к информационным входам мультиплексора, а коммутатор КМ2 подключает соответствующие зоны памяти ОЗУ или ПЗУ реализующие переходы. Одновременно необходима также перестраиваемая схема  $F_2$  для подключения соответствующих  $A_i$  по коду ( $i$ ). Структурная организация коммутаторов не

имеет особенностей применительно к автоматам нового типа и приведена во многих публикациях [41-43].

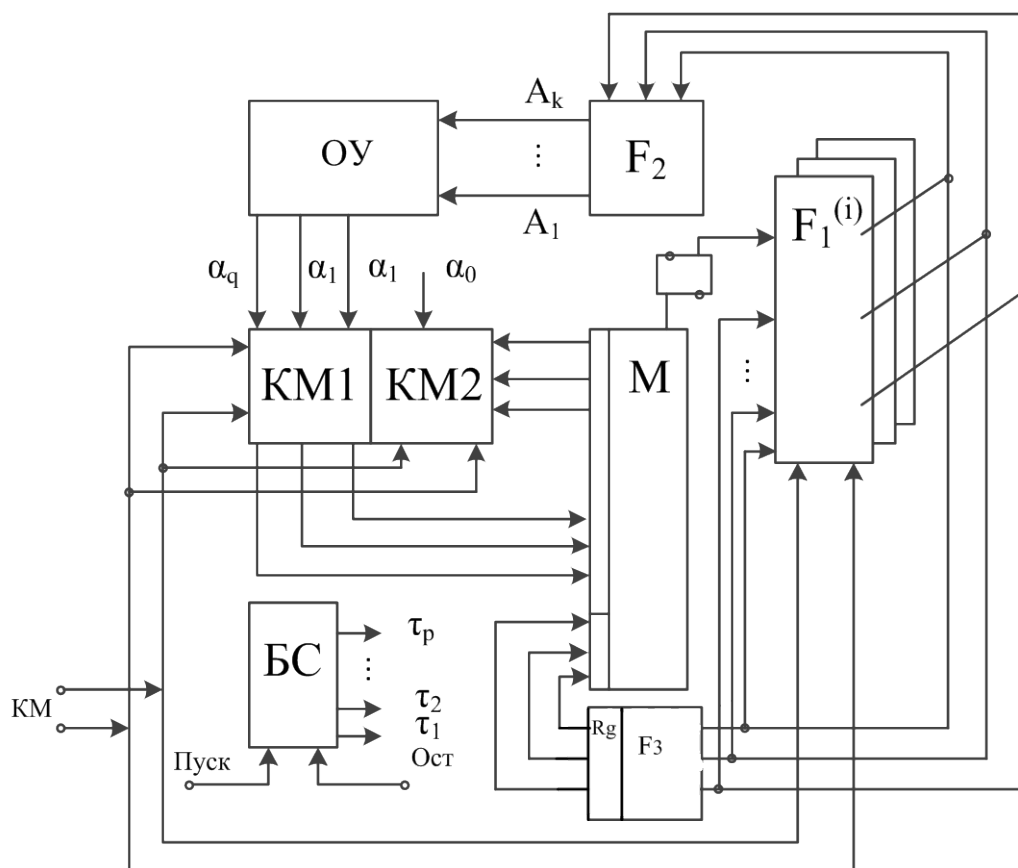


Рисунок 3.13 - Структурная схема многопрограммного автомата

### 3.4.2. Структурная организация взаимодействующих автоматов

Организация системы взаимодействующих автоматов нетривиальна в том случае, когда из одного автомата осуществляется обращение к другому автомату, реализующему некоторую подпрограмму. После завершения подпрограммы идет возврат к продолжению работы основного автомата. Такая процедура в программировании известна с момента появления ЭВМ первых поколений. Для взаимодействующих автоматов метод аппаратной организации такой системы предложен Плотниковым А.В. [79].

Существо метода сводится к следующему:

1. автоматы подпрограмм  $z_1, z_2, \dots$  заканчиваются оператором действия, обеспечивающим возврат к основному автомату  $z_0$  (как и в программировании). Дальнейшая процедура в автоматном исполнении имеет отличия;
2. все автоматы  $z_0, z_1, z_2, \dots$  рассматриваются как единый автомат с общей нумерацией состояний, т.е. если в ведущем автомате  $z_0$  состояния  $a_1, a_2, \dots, a_k$ , то в  $z_1 - a_{k+1}, a_{k+2}, \dots, a_{k+n}$ , в  $z_2$  соответственно  $a_{k+n+1}, a_{k+n+2}, \dots, a_{k+n+r}$ ;
3. вводится регистр памяти состояний и стековая память с односторонней связью (стек первого типа);
4. кроме кодов состояний  $a(t), a(t+1)$  вводится код возврата  $a_r$ ;
5. вводится регистр микрокоманд (РМК) на три поля:  $z$  - поле кода подпрограмм,  $k$  - поле кода  $a(t)$ ,  $y$  - поле кода микроопераций или кода  $a(t+1)$  при ориентации на автоматы Мура;
6. организуется взаимодействие через ведущий автомат Мура, тогда как автоматы  $z_0, z_1, z_2, \dots$  - это автоматы Мили.

Особенность метода синтеза по работе [79] заключается в том, что он синтезирует УА Мили после представления их графами переходов. Этот метод имеет следующие недостатки:

- сам синтез ориентирован на классические автоматы с большим числом логических условий. Причем комплекс из непрерывных (неразделенных операторами действия) логических условий (рисунок 3.14) заменяется системой отдельно обозначенных сложных логических условий, т.е. вводятся условия  $\beta_1, \beta_2, \dots, \beta_k$ . Условие  $\beta_i$  может быть равно  $\alpha_j$  или  $\overline{\alpha_j}$  или соответствовать значению конкатенаций логических условий.

Для реализации этого метода потребуется применение специального блока определения значения каждого  $\beta_i$  с фиксацией его на отдельном триггере. Тогда в каждом из УА, которые обозначим символами  $z_0, z_1, \dots, z_p$ , потребуется адресное устройство для нахождения  $\beta_i$  по полному множеству логических условий  $\{\alpha\}$ , соответствующих отдельным ГСА (рисунок 3.15, 3.16, 3.17). Либо необходимо увеличить значение  $q$ . Первый вариант решения усложняет структурную



организацию УА, а для второго варианта требуется существенное увеличение объема ПЗУ или ПЛМ для реализации автоматов. Кроме того, затрудняется анализ графов переходов и сопоставление его с исходной ГСА.

Сохранив основную идею работы [79], введем следующие изменения:

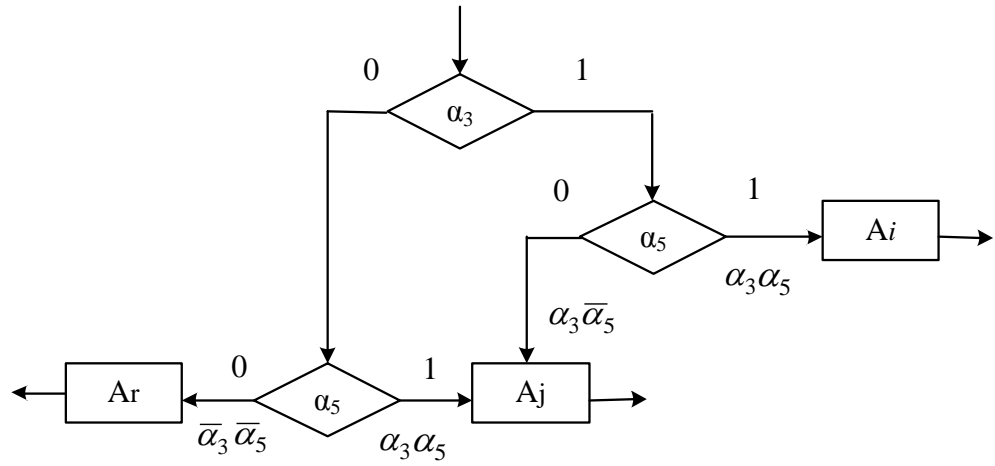
1. модифицируем объединенную ГСА с единой сквозной нумерацией операторов перед получением графа переходов по предложенному ранее методу;
2. на вход синтезируемого УА нового типа нет необходимости вводить коммутатор, обеспечивающий подключение соответствующих логических условий к основному и подчиненным автоматам, т.к. используется единая нумерация состояний автомата. А поэтому одно и тоже  $\alpha_j \in \{\alpha\}$  может выбираться при разных состояниях единого комплекса взаимодействующих автоматов.

Действительно одни и те же логические условия  $\alpha_i \in \{\alpha\}$  могут использоваться как в алгоритме ведущего автомата, так и в алгоритмах подпрограмм  $z_1, z_2 \dots z_p$ . Но в таком случае конкретные условия  $\alpha_j$  будут приписаны к разным состояниям.

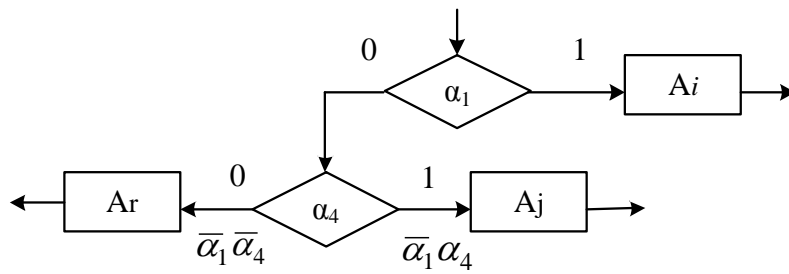
Общая комбинационная схема в единой ГСА даже для такого тривиального примера определится через 27 состояний при 9 логических условиях (с учетом безусловного перехода).

В рассматриваемом примере для комбинационной схемы единого автомата потребуется ПЗУ объемом  $V = m2^{m+q} = 5 \cdot 2^{15}$ . Реальный объем ПЗУ  $W = 8 \cdot 2^{15} = 2^{18} = 256 \text{кб}$ . В предлагаемой реализации  $W = 8 \cdot 2^7 = 1 \text{кб}$ , т.е. система взаимодействующих, соподчиненных автоматов по предлагаемой методике потребует ПЗУ с объемом в 256 раз меньшим (для рассматриваемого примера), чем при классической реализации автоматами Мура и Мили.

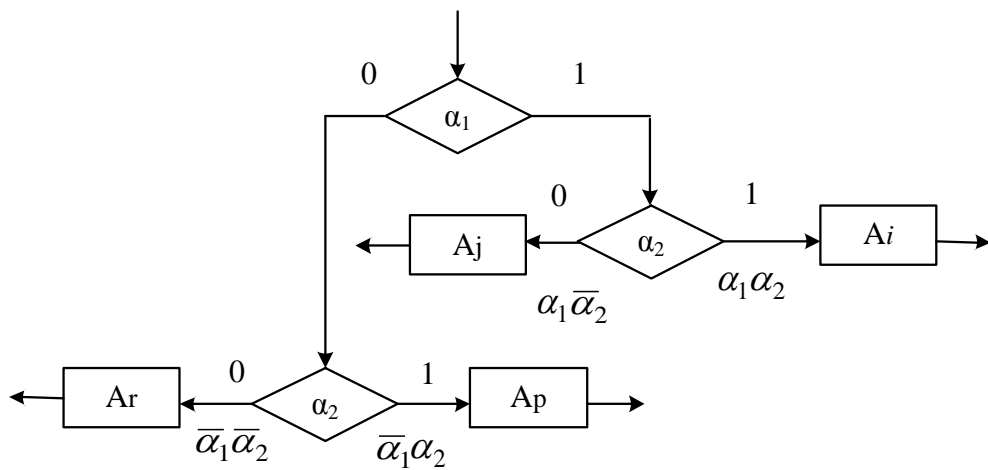
Предлагаемый метод синтеза взаимодействующих автоматов позволяет синтезировать УА, которые будут существенно проще УА по работе [79].



a)



б)



в)

Рисунок 3.14 – Варианты реализации сложных переходов

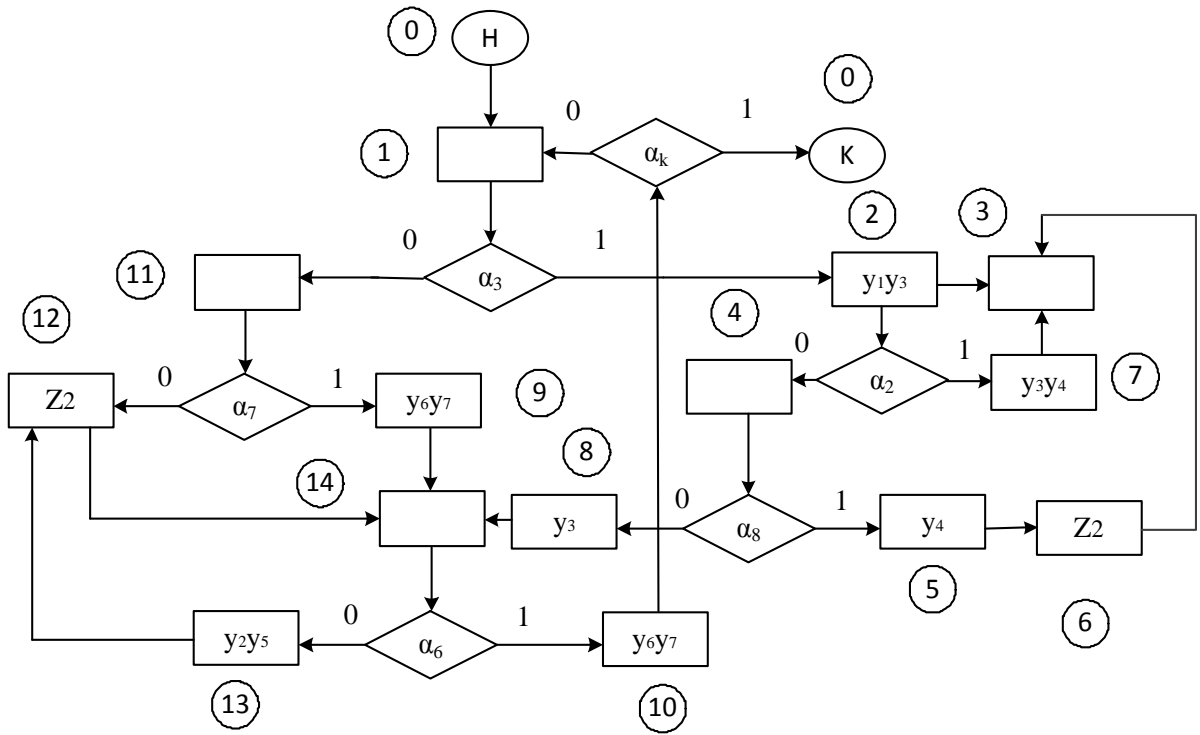


Рисунок 3.15 - Алгоритм ведущего автомата

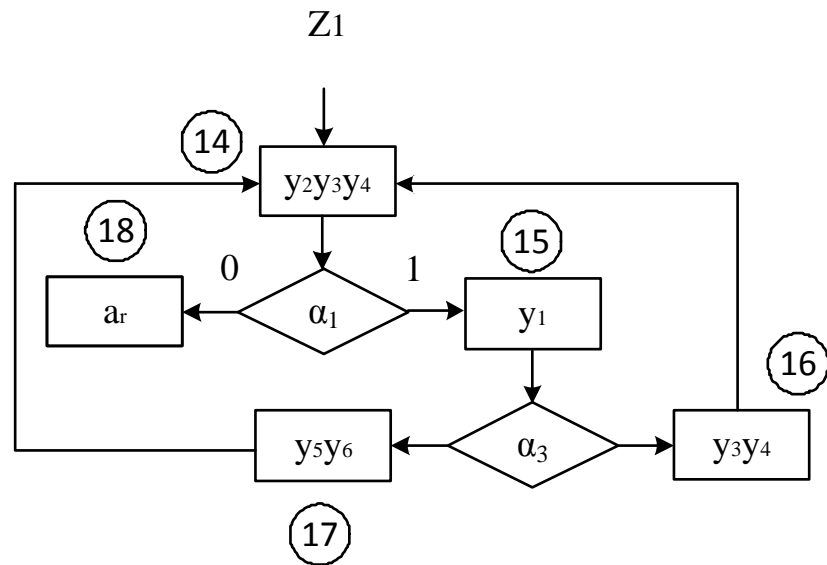


Рисунок 3.16 - Алгоритм первого ведомого автомата Z1

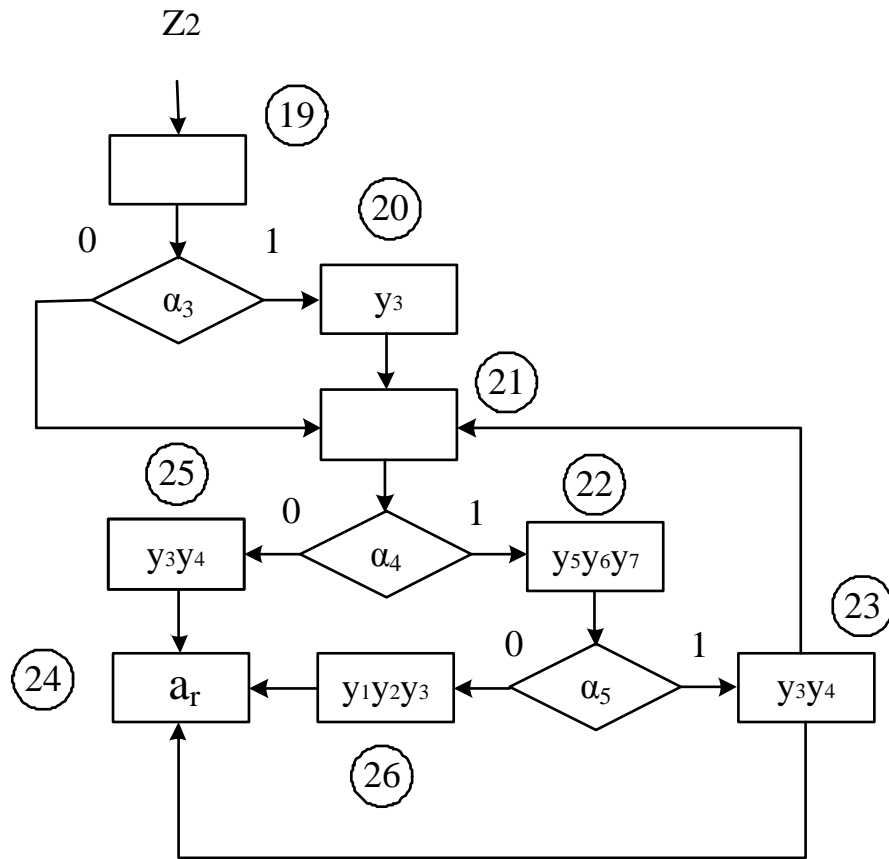


Рисунок 3.17 - Алгоритм второго ведомого автомата Z2

### 3.5. Управляющие автоматы с внутренним автоматом синхронизации.

На практике используются УА, выполняющие не только непосредственные функции выдачи команд управления  $A_1 A_2 \dots A_k$  по сигналам синхронизации

$$\tau_1(t) \wedge \tau_2(t) \wedge \tau_3(t) \wedge \tau_4(t) = \emptyset, \text{ но и ряд других служебных функций.}$$

Примером такой организации является УА с оригинальной декомпозицией [82] на два автомата:

- первый автомат, реализующий функции переходов между простыми и составными (сложными) операторами действия;
- второй автомат, реализующий сложные операторы как последовательность операторов действия  $A_i \in \{A\}$ , не разделенных логическими условиями. Второй автомат по существу представляет собой счетчик с дешифратором.

Другим примером является УА с самоконтролем. Для самоконтроля вводятся специальные контролирующие блоки, но обеспечение функционирования УА как в основном режиме, так и в режиме контроля и диагностики требует наличия специального автомата синхронизации с проверкой дополнительных логических условий. Речь идет о системе взаимодействующих автоматов, которые условно называются «автоматы в автомате» (рисунок 3.18). В этом случае первый автомат реализуется по предложенной методике с правилом функционирования Мура. Автомат синхронизации реализуется с правилом функционирования автоматов Мили, т.к. от него требуется более высокий уровень быстродействия, чем от основного автомата.

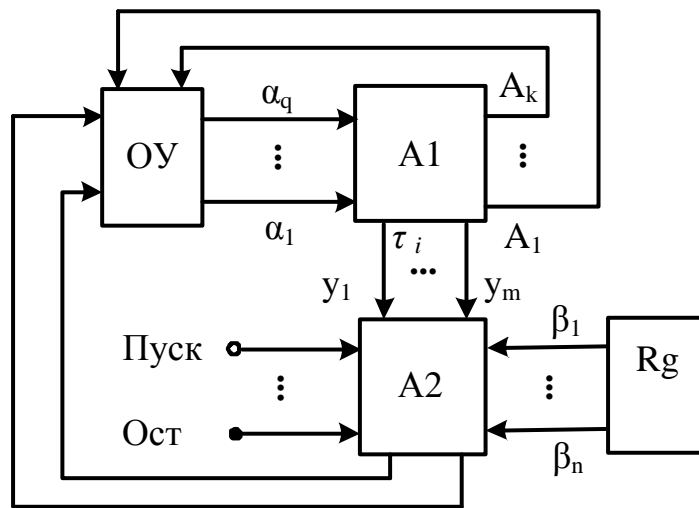


Рисунок 3.18 - Система автоматов

### 3.6. Структурно-автоматное программирование.

Области применения теории автоматов не ограничиваются узкой направленностью и специализацией. Например, один из крупнейших мировых производителей авиационной, космической и военной техники – американская корпорация «Боинг» занимается системами стабилизации самолетов с использованием теории автоматов.

На железнодорожном транспорте система для управления электровозами ВЛ80р строится по принципу действия конечных автоматов, что позволяет

реализовать режим рекуперации энергии (превращение механической энергии движущихся на спусках электропоездов в электрическую энергию) при высоких скоростях движения [102]. Чаще всего такие системы управления основаны на применении управляющих микроконтроллеров со специализированным программным обеспечением.

### 3.6.1. Методы моделирования управляющих автоматов

В процессе проектирования УА нового типа важнейшее значение приобретает проверка правильности функционирования, т.к., не смотря на меньшие затраты оборудования, такие УА относятся к структурно сложным автоматам. В известной научной литературе рассматривались два подхода:

- проверка результатов проектирования на соответствие некоторым формальным соотношениям, свойственным именно данной ГСА;
- проверка правильности функционирования УА методом моделирования динамики УА.

Первый подход развивался с середины 1970-х годов, но привел в итоге к появлению очень сложных методик и алгоритмов, требующих при практической реализации на ЭВМ класса Минск программы ~ с 600 командами [103]. Смена класса ЭВМ на современные контроллеры не приведет к снижению сложности такой программы [105].

Второй подход при реализации автоматных моделей в классе автоматов Мура и Мили также весьма сложен, при  $m \geq 5$ ,  $q \geq 16$  [104-108].

В известных методах моделирования последовательность действий в программном коде определяется сложностью граф-схемы алгоритма. Действительно, чем больше условий  $\{\alpha\}$  и состояний  $a(t)$ , тем сложнее программная модель, т.к. модель основана на реализации систем булевых функций. Для сложных автоматов системы булевых функций зависят от 20 -30 переменных. Поэтому такой подход на сегодня можно считать тупиковым, т.к. методики моделирования автоматов на языках высокого уровня приводят к получению программных «монстров».

В последнее время основой программирования по сложным управляющим ГСА является переход к программам с явно выделенными состояниями (UML технологии и др. [108-110]). Стремление придать стилю (методу) программирования некую неограниченную общность приводит к тому, что как первый, так и второй метод трудно осваиваются в практическом применении.

В связи с широким использованием теории автоматов появились идеи применения автоматного подхода в программировании [111]. Возникла новая парадигма программирования — автоматное программирование, согласно которому, программа или фрагмент программы представляется как модель формального автомата. Процесс исполнения ГСА при программировании в этой технике практически точно воспроизводит методы составления формальных автоматов (таких как машина Тьюринга, автомат Маркова и др.). Автоматное программирование — это метод программирования, основанный на явном выделении состояний и применении методов синтеза автоматов для описания поведения программ. Соответствующий метод проектирования программ назван методом автоматного программирования. Как отдельный компонент программы, так и программа в целом может быть реализована как автомат. Это позволяет проектировщикам работать на высоком уровне обобщения и разрабатывать эффективные программы. Метод профессора Шалыто А.А. хорошо проработан как в теоретическом, так и в инженерном плане [112-116].

Методы автоматного программирования можно разделить на 3 класса:

- процедурное программирование;
- объектно-ориентированное программирование;
- структурное программирование.

### 3.6.2. Программная реализация управляющих автоматов

В диссертационной работе предлагается развитие метода автоматного программирования, который назван структурно-автоматным. Структурно-автоматное программирование — это новый метод программирования,

основанный на структурной схеме автоматов нового типа с правилом функционирования Мура с памятью на счетчике.

Новизна предлагаемого подхода заключается в том, что программирование проводится на основе абстрактного синтеза УА.

Перед началом программирования составляется детальный алгоритм управления реальным операционным устройством или технологическим процессом, формализованная запись которого производится в виде операторной схемы алгоритма (ОСА).

Сущность нового метода структурного программирования сводится к следующему. Считается изначально заданным алгоритм, подлежащий переводу в программный код, где операторы действия ГСА трактуются теперь как вычислительные операторы  $A(t)$ , которые могут быть процедурами или по-прежнему управляющими операторами действия. Параллельные варианты ОСА не рассматриваются, т.к. они сводятся к иерархическим параллельно-зависимым или взаимодействующим автоматам. Программный код УА определяется через последовательность следующих действий:

1. Производится абстрактный синтез УА, т.е. по заданной (модифицированной) ОСА строится граф переходов. На графе переходов выделяется самый длинный путь от  $a_0$  до  $a_k$ . Причем если есть переход вида  $a_i(t) \rightarrow \overline{\alpha_j} a_{i+1}(t+1)$ , то значение  $\overline{\alpha_j}$  заменяется на  $\beta_j$ , где  $\beta_j = \overline{\alpha_j}$ . Тогда все переходы на «траектории счетчика» будут реализовываться по условиям  $\alpha_0$  или  $\alpha_i, \beta_j$ , принимающими значения «1». Условие  $\gamma = 1$  при необходимости прибавления «1» к содержимому счетчика можно определить как логическую сумму кодов состояний  $a(t) \in \{a\}$  на самом длинном пути. Эта процедура повторяется и для других непрерывных участков графа переходов.

2. Выписываются 3 таблицы:

- соответствие номеров логических условий  $\alpha_j$  состояниям  $a(t)$
- соответствие  $A(t)$  состояниям  $a(t)$
- соответствие кодов  $\alpha$   $x_1 \dots x_m$  коду  $y_1 \dots y_m$  для переходов вне счетчика.



При этом номер  $\alpha$  в конкатенации  $\alpha x_1x_2x_3x_4\dots x_m$  не указывается (только его значение), т.к. он выбирается из множества  $0,1,\dots,q$  по состоянию  $a(t+1)$ .

3. Если на графе переходов имеются петли, то они ликвидируются за счет введения пустой вершины графа перед вершиной с петлей.

4. Из преобразованного графа переходов формируется условие  $\gamma = 1$ , определяющее условие перехода как  $a(t) = a(t) + 1$ .

5. Используется структурная организация УА с памятью  $a(t)$  на счетчике.

Таким образом, специфика конкретной программы управления определяется тремя таблицами:

1) таблицей адресации (таблица 3.5), которая будет выполнять функцию определения адреса мультиплексора;

2) таблицей переходов, которая будет выполнять функцию комбинационной схемы  $F_1$  (таблица 3.6), т.е. определять  $a(t+1) \rightarrow \alpha(t+1)$ ;

3) таблицей  $F_2$ , отвечающей за формирование выходного действия (таблица 3.7).

На языках программирования самый простой способ реализации таблиц – представление их массивом данных.

Управление в автомате можно представить в виде отдельных взаимодействующих функций или классов в соответствующих парадигмах процедурного и объектно-ориентированного программирования. Фрагменты программного кода представлены на листингах 1,2 в приложениях.

На основе анализа существующих методов автоматного программирования в контексте двух парадигм — процедурного и объектно-ориентированного программирования можно сделать вывод о том, что предложен новый подход к проблеме автоматного программирования. Предлагаемое программное представление автоматов является простым, поскольку разделяет любой процесс, относящийся к большинству систем с управляющими автоматами, на две составляющие:

- универсальную программную реализацию управления процессом обращения к трем таблицам, представленных в виде массивов;

- специализированную часть, определяющую содержание таблиц по результатам абстрактного синтеза.

Особенность такого представления заключается в необходимости нахождении таблиц для мультиплексора и комбинационных схем, описывающих логику автомата.

В других известных вариантах программирования при вводе неразделенных логических условий пришлось бы вводить множественные операторы ветвления, что привело бы к утере удобного для понимания кода. Кроме того, при числе логических условий в ГСА равном  $q$ , в структуре программного кода должны быть обеспечены  $2^q$  переходов. Это и приводит к существенному усложнению программы и полной потере возможности визуального восприятия и анализа правильности ее реализации.

В предлагаемом структурном программировании нет этих сложностей, т.к. переходы осуществляются по значению одного (единственного) логического условия, которое выбирается программным мультиплексором из полного множества входных логических условий на каждом шаге автоматного программирования. Реализация программного мультиплексора не представляет каких либо затруднений, а адресация мультиплексора осуществляется на основе таблично-алгоритмического метода.

В результате моделирования УА был создан новый шаблон в автоматном программировании, основанный на структурной модели автоматов нового типа. Отличительная особенность созданного программного кода состоит в том, что он может быть использован для любой ГСА. Причем независимо от сложности ГСА, в динамике всегда реализуется одна и та же последовательность действий:

- 1) выбор  $\alpha_j \in \{\alpha\}$ ,  $j = \overline{1; q}$ ;
- 2) считывание значения функции  $F_1$  (т.е. кода  $y_1 y_2 \dots y_m$ ) по конкатенации  $\alpha_j x_1 x_2 \dots x_m$  из таблицы,  $m$  – код состояния  $a(t)$  автомата;
- 3) изменение кода состояний (через счетчик или таблицу  $F_1$ );
- 4) включение соответствующей исполнительной команды  $A_i \in \{A\}$  по таблице  $F_2$ ;
- 5) проверка условия на завершение команды  $A_i \in \{A\}$ ;

- 6) считывание значений  $\alpha_j \in \{\alpha\}$  по результату исполнительной команды  $A_i \in \{A\}$ ;
- 7) передача обновленного логического кортежа  $\alpha_1 \alpha_2 \dots \alpha_q$  на информационный вход мультиплексора;
- 8) проверка на необходимость завершения программы, иначе возврат к п.1.

Алгоритм ведущей программы, который определяет опрос таблиц и обеспечивает реализацию функционирования программной модели во времени, представлен на рисунке 3.19 с расшифровкой операторов в таблице 3.8.

Специфика автоматного подхода к программированию по предложенному методу определяется необходимостью предварительного абстрактного синтеза УА нового типа. Однако процедура синтеза полностью определена и может быть так же автоматизирована.

Предлагаемый метод структурного программирования позволяет существенно упростить создание программ как на языках высокого, так и низкого уровня.

Новый метод автоматного программирования, назван структурно - автоматным. Структурно -автоматное программирование основано на использовании результатов абстрактного синтеза автоматов с выбором одного условия из полного множества входных переменных (условий). Программный код нового типа не меняется при изменении алгоритма, подлежащего программированию. В этом случае необходимо изменять только содержимое трех секций ОЗУ. Общий объем памяти ОЗУ при этом много меньше объема, требующегося для программного кода автоматов с классической структурной организацией Мура (Мили).

Таблица 3.5. Таблица переходов

№	a(t)	$\alpha$	a(t+1)
1	3	$\overline{\beta_3}$	17
2	1	$\overline{\alpha_1}$	12
3	14	$\alpha_0$	10
4	12	$\overline{\alpha_1}$	11
5	2	$\overline{\beta_2}$	15
6	16	$\alpha_0$	14
7	15	$\overline{\alpha_1}$	19
8	19	$\alpha_0$	9
9	4	$\overline{\beta_4}$	20
10	20	$\alpha_0$	18
11	18	$\alpha_0$	19
12	6	$\overline{\alpha_1}$	22
13	6	$\alpha_0$	0
14	10	$\alpha_0$	0

Таблица 3.6. Кодирование переходов

№	a(t)	N(t)	$\alpha_j$	$\alpha$	a(t+1)	N(t+1)
1	0	00000	0	1	1	00001
2	1	00001	1	0	1	00001
3	1	00001	1	1	2	00010
4	2	00010	0	1	3	00011
5	3	00011	0	1	4	00100
6	4	00100	2	0	5	00101
7	4	00100	2	1	12	01100
8	5	00101	3	0	13	01101
9	5	00101	3	1	6	00110
10	6	00110	5	0	6	00110
11	6	00110	5	1	7	00111
12	7	00111	0	1	8	01000
13	8	01000	0	1	9	01001
14	9	01001	7	0	15	01111
15	9	01001	7	1	10	01010
16	10	01010	8	0	9	01001
17	10	01010	8	1	11	01011
18	11	01011	0	1	0	00000
19	12	01100	4	0	14	01110
20	12	01100	4	1	6	00110
21	13	01101	0	1	4	00100
22	14	01110	6	0	14	01110
23	14	01110	6	1	7	00111
24	15	01111	9	0	14	01110
25	15	01111	9	1	11	01011

Таблица 3.7 Выбор логических условий

$\alpha (\beta)$	a(t)
$\alpha_1$	1
$\beta_2$	2
$\beta_3$	3
$\beta_4$	4
$\beta_4$	5
$\alpha_6$	6
$\alpha_7$	12

Таблица 3.8 Семантика операторов действия

$R_i$	Содержание
1	Служебн. (инд.)
2	«0» $\rightarrow$ СТ, Rg 1-4, Тр 1,2
3	$\langle Rg1,3 \rangle \rightarrow M$ ; $\langle M \rangle \rightarrow$ Тр 1
4	$\langle F_1 \rangle \rightarrow$ Rg 2
5	$\langle F_3 \rangle \rightarrow$ Rg 4; $\langle Rg2 \rangle \rightarrow$ Rg3
6	$\langle CT \rangle = \langle CT \rangle + 1$
7	$\langle Rg2 \rangle \rightarrow$ DC
8	Пуск $A_i \in \{A\}$
9	Пустой оператор
10	$\langle OY \rangle \rightarrow$ Rg 1
11	Пустой оператор
12	$\langle Rg3 \rangle \rightarrow$ СТ

$$\alpha_1 = \gamma = 1 \rightarrow \langle CT \rangle = \langle CT \rangle + 1$$

$$\alpha_2 = 1 \text{ окончание } A_i \in \{A\}$$

$$\alpha_3 = 1 \text{ признак возврата (e) к } a_0$$

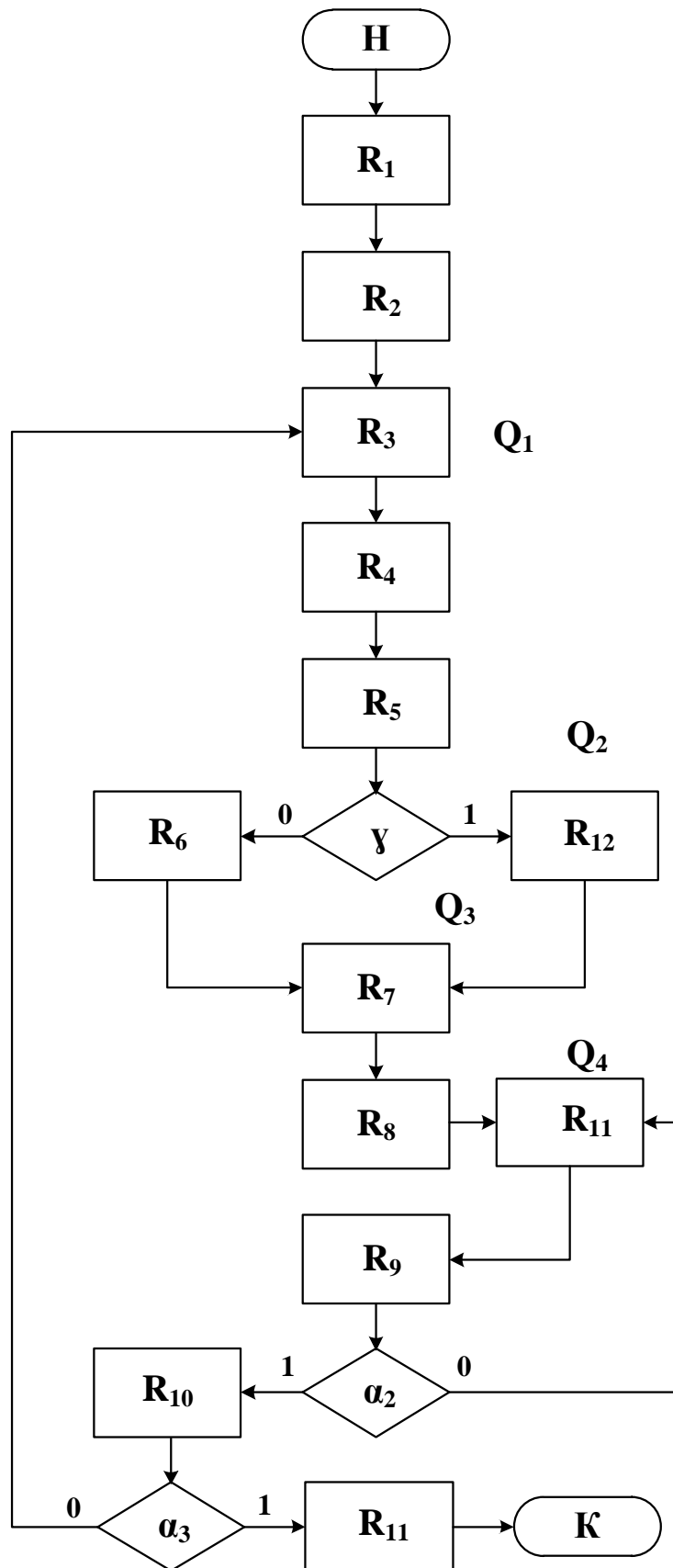


Рисунок 3.19 - Алгоритм управления реализацией процедуры структурно-автоматного программирования микроконтроллеров.

## Выводы по главе 3

1. Новая методика синтеза управляющих автоматов дополнена тремя положениями, определяющими применимость в структуре управляющих автоматов любого типа счетчика в качестве памяти автомата. Такая структурная организация эффективна не только для автоматов преимущественно последовательного типа с незначительным числом логических переменных, но и для сложных автоматов с большим числом состояний и большим числом логических переменных. За счет предложенного развития основной методики включение счетчика, метод позволяет снизить сложность комбинационной схемы переходов в 2-3 раза в уже спроектированном по новой методике автомате.
2. Синтез иерархических автоматов в виде одного ведущего автомата верхнего уровня и нескольких независимых автоматов нижнего уровня предложено осуществлять через создание единого автомата нижнего уровня с общей памятью и с логическим выбором комбинационных схем соответствующих независимым автоматам нижнего уровня. При такой структурной организации нет необходимости во входном коммутаторе логических условий, т.к. при наличии мультиплексора и перестраиваемой схемы адресации не нарушается логика взаимодействия автоматов, но снижаются затраты оборудования при реализации всего комплекса иерархических автоматов.
3. Для взаимодействующих автоматов, не имеющих явно выделенного автомата верхнего уровня управления, предложено комплекс автоматов заменить единым сложным автоматом с числом состояний, равным сумме всех состояний отдельных взаимодействующих автоматов. Эффективность реализации также определяется как  $Q = 2^{q-1}$ , при этом нет необходимости включать в структуру автомата коммутаторы для определения полного множества комбинаций групп непрерывной последовательности логических условий, как это используется в известных методах синтеза.



4. Реализация сложных управляющих автоматов через декомпозицию на несколько независимых автоматов предлагается осуществлять через переход к синтезу единого многорежимного автомата. Методика синтеза многорежимного автомата основана на смене кода частей в условно декомпозируемой граф - схеме автомата. Метод позволяет ограничивать уровень итерационного декомпозирования по критерию минимальных затрат оборудования на реализацию многорежимного автомата. Предложенный метод эффективен и реализуем только для управляющих автоматов нового типа.
5. Для моделирования на ЭВМ предложено развитие метода автоматного программирования на базе новой структурной организации управляющих автоматов. Новый метод (который назван структурно-автоматным программированием) позволяет повысить быстродействие управляющих программ для микроконтроллеров  $\approx$  на порядок или при заданном уровне быстродействия перейти к реализации управляющей программы на микроконтроллере более упрощенного типа с пониженной частотой синхронизации, который отличается более низким уровнем энергопотребления и более высоким уровнем безотказности.

По структурно - автоматному программированию зарегистрировано две программы во Всероссийском фонде алгоритмов и программ. Отличительной особенностью структурно - автоматного программирования является простота и малый объем программного кода, т.к. ведущая программа по заданному алгоритму составляется один раз и не меняется при смене алгоритма. Замена алгоритма требует лишь смены содержимого трех зон памяти для выбранного типа микроконтроллера.

Таким образом, показано, что новая методика синтеза дает возможность существенно снизить затраты оборудования при реализации комплекса автоматов.

По материалам главы 3 автором опубликованы работы [10,11,28,29,32,36,39,40,43,44,46,50,57,71] и получены патенты [67,68,70].

## ГЛАВА 4. СИНТЕЗ УПРАВЛЯЮЩИХ АВТОМАТОВ С ДИНАМИЧЕСКИМ КОНТРОЛЕМ ФУНКЦИОНИРОВАНИЯ

### 4.1. Методы и средства динамического контроля управляющих автоматов

Во всех ИУС для управления критическими технологиями и ответственными процессами недопустима выдача неправильных команд, создающих опасную ситуацию. Для таких информационно-управляющих систем используются УА со встроенными средствами контроля, способными выдать сигнал о неисправности в динамическом режиме, т.е. перед выдачей очередной команды управления.

Управляющие автоматы с динамическим контролем правильности функционирования относят к классу самоконтролируемых управляющих автоматов (СУА). Проектирование схем СУА неотделимо от вопросов проектирования надежных УА, отличающихся минимальными затратами логических элементов.

В 1960-85 г.г. появились статьи и монографии по контролю и диагностике ЭВМ, в которых частично ставилась задача контроля устройств управления главным образом в режиме тестовой (предстартовой) проверки МПА [117-134]. Динамический контроль конечных автоматов развивался самостоятельно в двух не строго делимых направлениях:

- встраивание средств контроля в уже спроектированный МПА;
- проектирование таких МПА, которые обладают свойством самоконтроля.

По встроенным средствам контроля автоматов одними из первых появились работы М.А. Гаврилова, Ю.Л. Сагаловича, В.Г. Лазарева и А. Д. Закревского, В.В. и Вл.В. Сапожниковых. В работе [121] предложено минимизацию систем булевых функций производить не только для основных разрядов кода переходов, но и для дополнительных, образующих контрольные биты, соответствующие коду Хемминга [117]. Метод обеспечивает высокий уровень контролеспособности при

~ 30% увеличении объема  $F_1$ . Однако он не нашел применения в связи с необходимостью встраивать в структуру МПА сложный декодер кода Хэмминга.

Применение корректирующих кодов для контроля комбинационных схем систематизировано в монографиях [135-139].

Второй подход предложен М.А. Гавриловым для синтеза  $d$ -безотказных автоматов, правильно выполняющих функции при числе отказов  $t \leq d$ . Для реализации метода необходимо:

- представить состояния МПА векторами (словами) минимально избыточного корректирующего кода;
- расстояние между отдельными векторами должно соответствовать кодовому расстоянию по Хеммингу не менее  $2d+1$ .

Метод М.А. Гаврилова развит в работе [43]. Однако метод не нашел широкого применения в связи с необходимостью введения значительной избыточности и без того сложных СА и ВС автоматов.

Большинство методов динамического контроля связано с использованием специальных кодов для представления  $a(t)$  и  $a(t+1)$  вместо двоично-позиционного (ДПК) или двоичного соседнего кода (код Грея). Применяются различные методы кодирования для обеспечения самоконтроля УА.

В работах [122-125] предложено использовать код с фиксированным числом «1», обозначенный как  $kC_n$ , т.е.  $k$  единиц в  $n$ -разрядном коде. Для кодирования используются все возможные комбинации с произвольным расположением  $k$  единиц. Однако комбинационные схемы определения наличия именно  $k$  единиц в  $n$ -разрядном коде оказываются непростыми. Поэтому этот метод имеет несколько модификаций, в частности за счет разделения  $m$ -разрядного кода  $a(t)$  на две части и представления каждой из них кодом  $kC_n$  [123]. В работе [8] развиты методы синтеза схем проверки наличия 2-х единиц в пятиразрядном коде или 3-х единиц в 7-разрядном коде и др.

Метод контроля  $kC_n$  является наиболее простым и быстродействующим, но объем ПЗУ вместо  $V = m2^{m+q}$  становится равным  $V = 2m2^{2m+q}$ .

Предлагается также при кодировании  $a(t)$  в виде ДПК дописать инверсный код, соответствующий числу единиц в ДПК (коды Бергера). Этот метод основан на выполнении операции подсчета числа «1» в ДПК через счетчик и сравнении с инверсным кодом [136,137].

Другим самостоятельным подходом контроля является метод В.Н. Балакина и В.В. Барашенкова [140,142], который требует введения в структуру МПА счетчиков для определения числа операторов перед проверкой логических условий, а так же введения в граф переходов специальных контролирующих состояний и др. Метод эффективен, т.к. проверяет не только правильность переходов  $a(t) \rightarrow a(t+1)$ , но и обнаруживает неверную генерацию  $A_i$  вместо  $A_j$ . Однако такие средства контроля весьма сложны, хотя перспективны для контроля процессов с малым числом логических условий ( $\alpha_j$ ) и участками ГСА с большим числом последовательностей операторов действия ( $A_i \in \{A\}$ ).

Результативным подходом является метод А.Н. Буинова [131]. В его работе предложено построить объединенную матричную модель сигналов, представленных множеством допустимых состояний (матрица  $A$ ) и множеством запрещенных состояний для каждого допустимого состояния  $a(t)$  – матрица  $B$ . Сравнение входных сигналов при достигнутом состоянии  $a(t+1)$  с матрицами  $A$  и  $B$  позволяет принять решение о правильности функционирования УА. Метод А.Н. Буинова требует слишком больших аппаратных и временных затрат и не нашел применения для динамического контроля, однако может быть перспективен для тестового контроля УА, который в данной работе не рассматривается.

При динамическом контроле применим и общеизвестный метод полного дублирования УА. Вариантом эффективного контроля является самодвойственное дублирование, обеспечивающее в среднем  $\sim 67\%$  избыточности при  $\sim 99\%$  покрытия неисправностей [143]. Однако этот метод требует дальнейшего исследования, т.к. одновременно необходима и двойная временная избыточность.

В работах [144-156] рассмотрены другие принципиально различные методы построения СУА.

Следует отметить особую роль по созданию методов синтеза СУА д.т.н. В.В. и Вл.В. Сапожниковых. Результаты многолетней деятельности этой научной школы отражены в фундаментальной публикации [8].

Вариантов реализации СУА на основе упрощения или совместного использования рассмотренных методов очень много, однако до сих пор отсутствуют публикации в виде системного обобщения и оценки эффективности методов контроля УА.

Новые методы контроля и структурная организация СУА нового типа, приведенные в этой главе, отражают результаты диссертационных исследований автора.

#### 4.2. Контроль автоматов управления с избыточными кодами

Рассмотрим модификацию кода  $kC_n$ , заключающуюся в следующем: весь код разрядности  $n$  делится на группы по три разряда  $a, b, c$ . Если в  $n$ -разрядном коде число таких групп равно  $p$ , то в таком коде  $kC_n$  будет ровно  $p$  единиц. Такой код обозначен как  $pC_n$ . В этом коде устройство поиска наличия единиц очень простое, т.к. для каждой группы  $abc$  ошибка фиксируется при условии  $\overline{ERR} = a\bar{b}\bar{c} + \bar{a}b\bar{c} + \bar{a}\bar{b}c$ . Опрос всех  $p$  групп может реализовываться как последовательно, так и параллельно. Выходная разрядность схемы переходов в коде  $pC_n$  определяется выражением  $n = 1.5m$ , где  $m$  – разрядность первоначального двоично-позиционного кода (ДПК)  $a(t+1)$ .

Если код  $pC_n$  использовать как на выходе, так и на входе схемы  $F_1$  СУА, то объем ПЗУ для автоматов нового типа определится как  $W = 1.5m2^{1.5m+1}$ .

В предложенной соискателем оригинальной структурной организации СУА [157] с кодом  $pC_n$  введен дополнительно шифратор для перевода кода  $pC_n$  в двоичный непозиционный код (ДНК) разрядности  $(m+1)$ . Правило преобразования группы разрядов  $(abc)$  кода  $pC_n$  в код ДНК задается таблицей 4.1. ДНК всех  $p$  групп передается на вход комбинационных схем  $F_1$  и  $F_3$ . В таком СУА объем ПЗУ для  $F_1 = (m+1)2^{1.5m+2}$ . Структурная схема СУА приведена

на рисунке 4.1. СУА контролируется не только по выходу  $F_1$  за счет кода  $pC_n$ , но и по входу  $F_1$ , т.к. в коде ДНК в каждом двух разрядах не может быть комбинаций «11».

Оригинальная организация СУА такого типа обладает более высокой степенью самоконтролируемости т.к. контролируется как вход, так и выход  $F_1$ . При этом используются более простые встроенные средства контроля по сравнению с известными методами на основе кодов  $kC_n$ .

Таблица 4.1. Двоичный непозиционный код

$pC_n$	ДНК
c b a	$x_2 x_1$
0 0 1	0 0
0 1 0	0 1
1 0 0	1 0

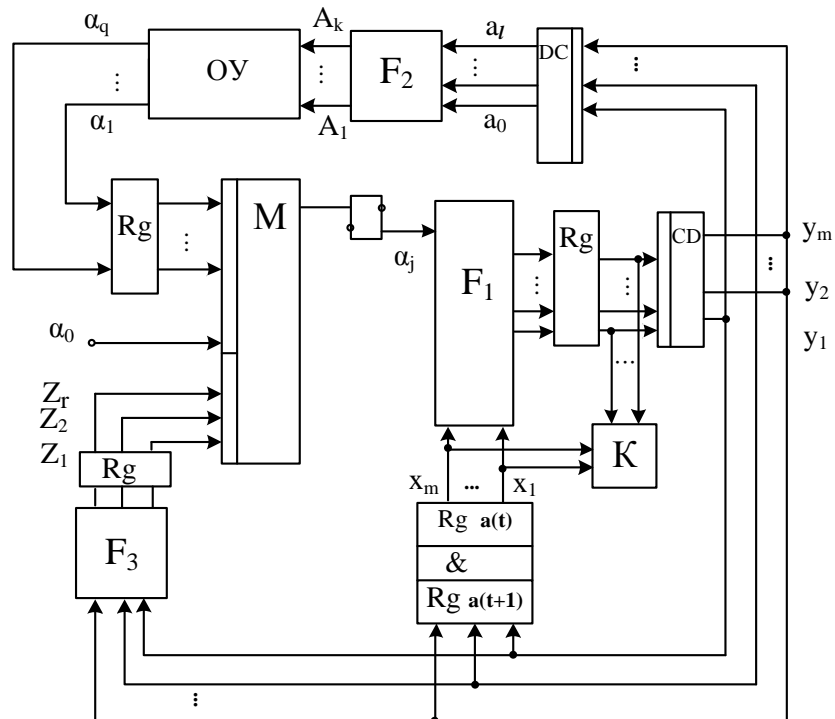


Рисунок 4.1 – Структурная схема СУА

### 4.3. Контроль УА с разделенным кодом схем переходов

Организация УА с разделенными вычислениями младших и старших разрядов кода  $y_1 y_2 \dots y_m$  дает возможность осуществить контроль правильности формирования двух половин  $y_1 \dots y_p$  и  $y_p \dots y_m$  кода  $a(t)$  предельно простыми средствами. Для сложных автоматов (АС)  $m=6$ . По предложенному методу разделения кода  $a(t)$  на 2 части необходимы две комбинационные схемы  $F_1(1)$  и  $F_1(2)$  с разрядностью на входе  $r=5$ , а на выходе  $m/2=3$ . Тогда контрольный код  $pCn$  будет формироваться для трехразрядных выходных кодов  $F_1(1)$  и  $F_1(2)$ , т.е. будет представлен двумя группами по 3 разряда. Количество всех комбинаций при трех разрядах двоичного кода равно 8. Для представления всех возможных комбинаций в коде ( $pCn$ ) двумя группами по три разряда в группе  $abc$  потребуется всего 6 разрядов. Действительно, код  $pCn$  обеспечивает представление девяти различных комбинаций. Итак при общей разрядности  $m=6$  вместо одной комбинационной схемы со входным кодом  $a_j$   $x_1 x_2 \dots x_6$  и выходным  $y_1 y_2 \dots y_6$  потребуются две схемы  $F_1(1)$  и  $F_1(2)$  со входным кодом в 5 разрядов и выходным  $n$ -разрядным кодом  $pCn$  в 6 разрядов.

Согласно методу контроля УА с кодами  $pCn$  с использованием метода разделения  $F_1$  на  $F_1(1)$  и  $F_1(2)$  схемная реализация средств контроля предельно упрощается. Упрощение достигается не только за счет снижения числа групп  $abc$  до двух, но и за счет упрощения средств обратного преобразования кода  $pCn$  в ДНК. При этом возможно последовательное подключение средств контроля как с  $F_1(1)$ , так и с  $F_1(2)$ . При высоких требованиях к быстродействию средства контроля дублируются. Представленный метод контроля УА с минимальными затратами контролирующего оборудования возможен только для УА нового типа и невозможен для классических УА Мура.

#### 4.4. Метод контроля автоматов с возвратом к исходному коду

Несмотря на двойные затраты оборудования, в известных методах дублирования фиксируется лишь сам факт неисправности, но какая из двух схем (базовая или дублирующая) неисправна - определить невозможно. Поэтому используется метод мажоритарного резервирования с трехкратным или даже с пятикратным дублированием для особо ответственных применений. При трехкратном резервировании УА считается исправным, если у двух из трех схем совпадают результаты, а при пятикратном резервировании необходимо зафиксировать три из пяти совпадающих результата. Для принятия решений требуется слишком большие затраты оборудования и большое время для проверки, поэтому такие варианты в системах реального времени промышленного назначения не используются.

На рисунке 4.2 представлен вариант дублирования, предложенный диссертантом, в котором кроме двух дублирующих схем ( $F_1^1$ ) и ( $F_1^2$ ) вводится еще третья схема обратного преобразования ( $F_1^3$ ) выходного кода  $y_1 y_2 \dots y_m$  с учетом  $\alpha_j(t)$  к исходному коду  $x_1 x_2 \dots x_m$ . Тогда проверка правильности функционирования сводится к следующим действиям:

- 1)  $\langle F_1^1 \rangle \rightarrow Rg7$
- 2)  $\langle F_1^3 \rangle \rightarrow Rg9$
- 3)  $\langle CT \rangle \oplus \langle Rg9 \rangle = R$
- 4) Если  $R=0$ , то ( $\langle Rg7 \rangle \rightarrow DC(18)$ ; иди 10)
- 5)  $\langle F_1^2 \rangle \rightarrow Rg7$
- 6)  $\langle F_1^3 \rangle \rightarrow Rg9$
- 7)  $\langle CT \rangle \oplus \langle Rg9 \rangle = R$
- 8) Если  $R=0$ , то ( $\langle Rg7 \rangle \rightarrow DC(18)$ ; иди 10)
- 9) УА неисправен
- 10) УА исправен

В предложенном методе затраты оборудования такие же, как в методе обычного троирования. Однако такой способ трехкратного резервирования, в



отличие от известного способа, не требует проверки всех комбинаций (два из трех), т.к. только одним сравнением достигается установление факта исправности комбинационной схемы  $F_1^1$ . Если же она окажется неисправной, повторяется проверка и для  $F_1^2$ . Выигрыш в быстродействии очевиден.

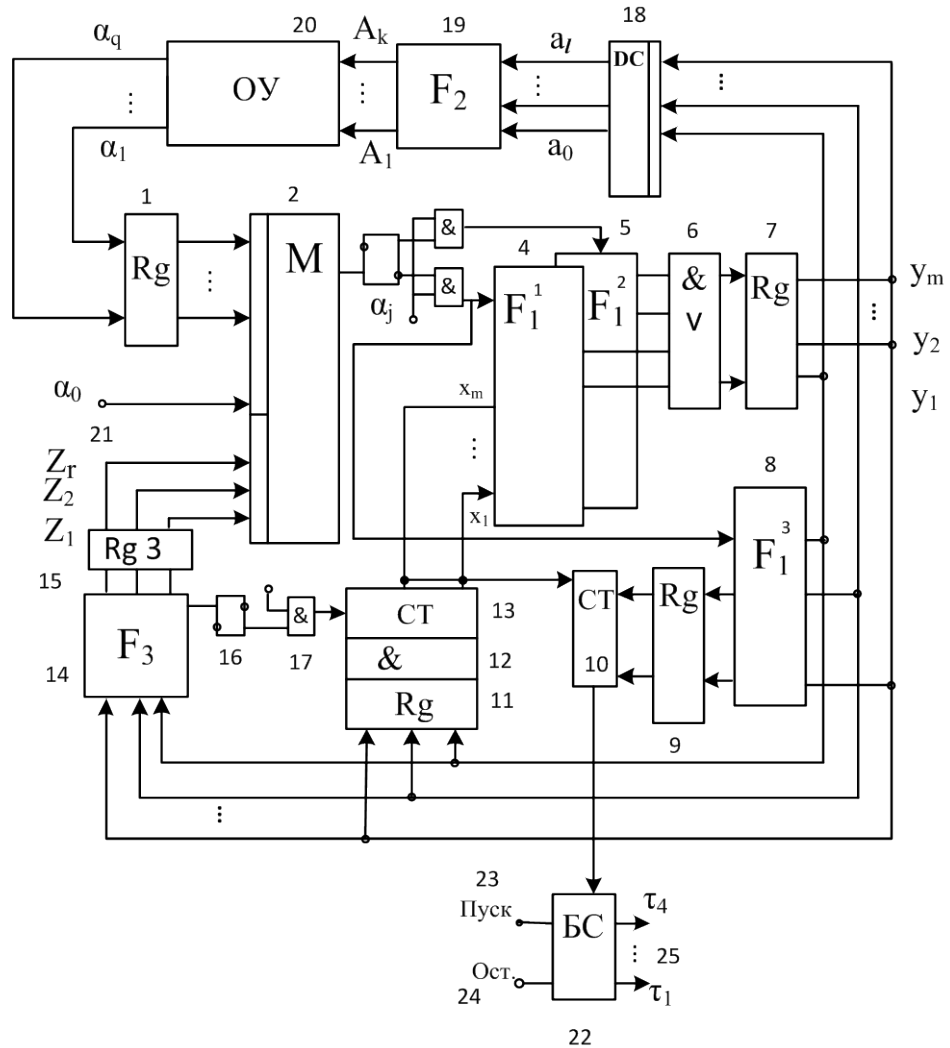


Рисунок 4.2 - Автомат с контролем через возвращение к исходному состоянию

В диссертации новый метод мажоритарного резервирования со схемой возврата к исходному коду  $a(t)$  предложено совместить с методом разделения комбинационной схемы  $F_1$  на две зоны  $F_1^1$  и  $F_1^0$ , т.к. выход из строя  $F_1$  происходит за счет неверного функционирования переходов, относящихся к первой или второй половине (из-за множественных надстрочных и подстрочных знаков для  $F_1$  введем новые обозначения: первые полусхемы прямого преобразования назовем  $F_1^1(1)A$  и  $F_1^1(0)A$ , вторые (дублирующие) полусхемы

прямого преобразования -  $F_1^2(1)B$  и  $F_1^2(0)B$ ). Выход из строя обеих половин одновременно маловероятен. Одновременно задублируем схему обратного преобразования  $F_1^3$ , причем каждую также разделим на схемы, синхронизируемые по значению  $\alpha_j$  или  $\bar{\alpha}_j$ . (полусхемы обратного преобразования -  $F_1^3(1)C$ ,  $F_1^3(0)C$  и  $F_1^3(1)D$ ,  $F_1^3(0)D$ ). Структурная организация полного комплекса мажоритарного резервирования с использованием разделенных схем представлена на рисунке 4.3, где обозначено: БРИ – блок распределения информации,  $C_i$  – сигналы синхронизации.

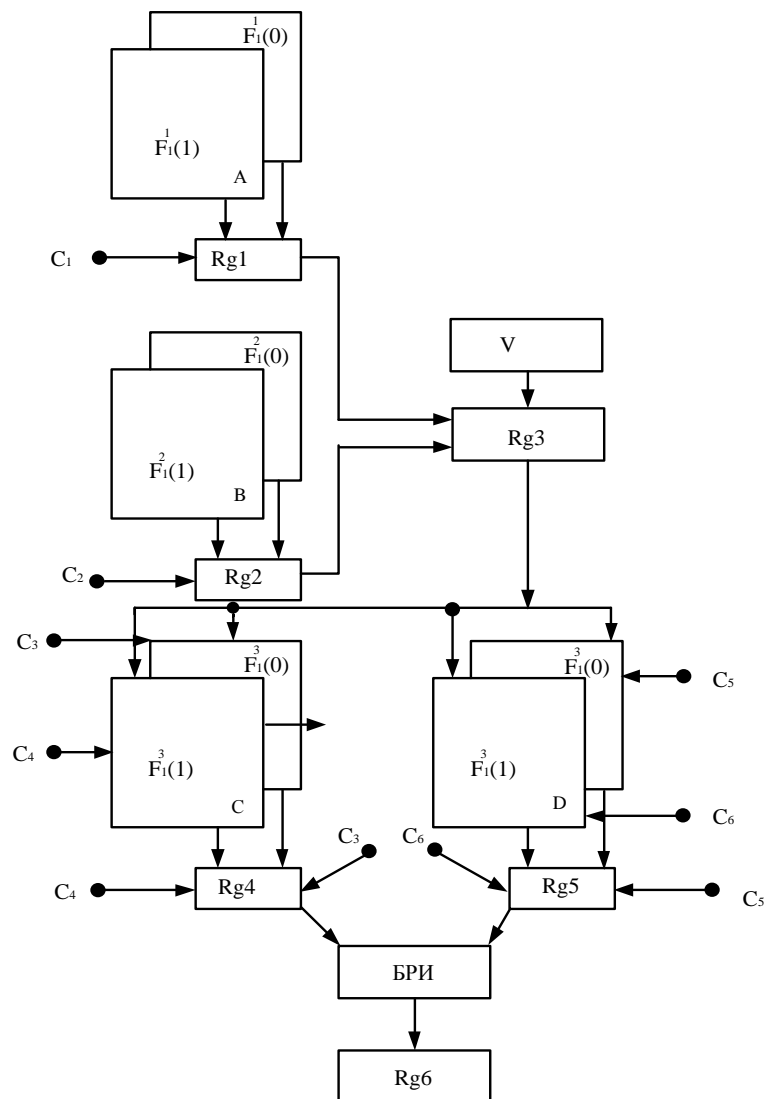


Рисунок 4.3 – Автомат с мажоритарным резервированием

Управление такой сложной структурной организацией комплекса резервируемых схем осуществляется в следующей последовательности:

1. Производится проверка исправности полусхем  $F_1^1A$  и  $F_1^2B$  за счет их сравнения. В случае неисправности конструируется исправный вариант  $F_1(N)$  за счет «перекрестного» объединения соответствующих схем  $F_1^1(1)A$  и  $F_1^2(0)B$  или  $F_1^1(0)A$  и  $F_1^2(1)B$ .

2. Сравнивается  $F_1(N)$  с  $F_1^1A$ . Если  $F_1^1A$  исправна, то для дальнейшей работы используется  $F_1^1A$ , иначе повторяется пункт 2, но для схемы  $F_1^2B$ .

3. Также как для схем  $F_1^1A$  и  $F_1^2B$  производится проверка исправности схем  $F_1^3C$  и  $F_1^3D$  за счет их сравнения. В случае неисправности конструируется исправный вариант  $F_1^3(N)$  за счет «перекрестного» объединения соответствующих схем  $F_1^3(1)C$  и  $F_1^3(0)D$  или  $F_1^3(0)C$  и  $F_1^3(1)D$ .

4. Производится проверка по правилу п.2 для  $F_1^3$

5. Производится сравнение исправных схем прямого и обратного преобразования.

6. Формируется сигнал неисправности, если ни один из пунктов 1-5 не привел к положительному результату.

#### 4.5. Динамический контроль сложных управляющих автоматов.

Наименьшие затраты оборудования на реализацию комбинационной схемы  $F_1$  требуются в УА с памятью на счетчике. Для сложных автоматов диссертантом предлагается новая методика синтеза и структурная организация СУА. Методика сводится к выполнению следующих этапов:

1. Проводится абстрактный синтез УА нового типа со счетчиком, который используется одновременно и как регистр памяти.

2. Счетчик работает в коде Грея.

3. Для соседних вершин  $a(t)$  и  $a(t+1)$  в графе переходов осуществляется контроль по известному методу для кодов Грея на основании фиксации только одной «1» в сумме кодов, если обе вершины относятся к «траектории» счетчика.

4. Для соседних вершин вне «траектории» счетчика коды  $a(t)$  и  $a(t+1)$  делятся на группы младших  $(1 \dots p)$  и старших  $(p+1 \dots m)$  разрядов. Для младших разрядов выбираются такие коды из таблиц Грея, чтобы выполнялось условие

равенства «1» суммы младших разрядов  $(1 \div p)$  для  $a(t)$  и  $a(t+1)$ . Если это условие выполнено для  $(1 \div p)$ , то для старших разрядов этих выбранных кодов Грея это условие выполняться не будет. Тогда к старшим разрядам состояния  $a(t)$  добавляется разряд  $x_0$ , а к старшим разрядам  $a(t+1)$  добавляется разряд  $y_{m+1}$ . Значения этих разрядов доопределяются так, чтобы сумма по mod2 кода старших разрядов  $\{x\}$  и старших разрядов кода  $\{y\}$  с учетом разрядов  $x_0$  и  $y_{m+1}$  также обеспечивала выполнение условия:  $(x_m \dots x_{p+1})x_0 \oplus y_{m+1}(y_m \dots y_{p+1}) = 1$

5. В ряде случаев (рисунок 4.4а) это условие не может быть выполнено даже для младших разрядов, т.к. коды вершин на длинном пути не могут быть изменены. В таких случаях вводится промежуточная вершина, выполняющая роль согласующей (рисунок 4.4.б). Для вершин  $a_j$  и  $a_{j+1}$  выбираются такие номера, чтобы для младших разрядов  $(1 \div p)$  выполнялись условия:  $a(t) \oplus a(j) = 1$ ;  $a(j) \oplus a(j+1) = 1$ ;  $a(j+1) \oplus a(t+3) = 1$ .

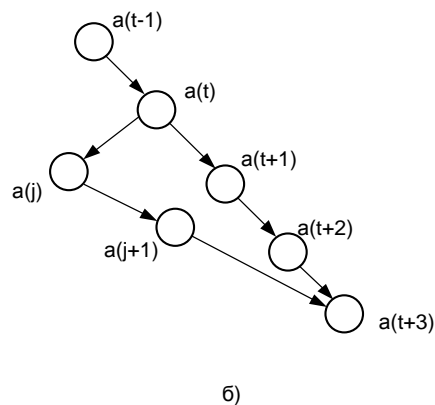
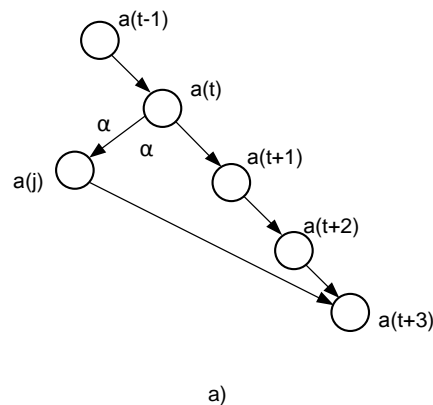


Рисунок 4.4 – Участки графа переходов:

а- заданный, б – с внесенной промежуточной вершиной

б. Сравнение кодов  $a(t)$  и  $a(t+1)$  производится независимо для младших и старших разрядов на соответствующих схемах  $(m/2)$  разрядности. Схемы сравнения не связаны между собой. Результат будет верен, если одна «1» в итоговой сумме будет зафиксирована как для младших, так и для старших разрядов с учетом  $x_0, y_{m+1}$ .

Синтез СУА по предложенной методике имеет ряд особенностей и поэтому рассматривается более подробно.

#### 4.6. Контроль управляющих автоматов с использованием соседнего кодирования половинных кодов состояний

Метод использования неполного кода  $a(t)$  оказывается эффективным не только при дублировании комбинационной схемы переходов, но и в задаче непосредственного контроля УА нового типа.

Выполним абстрактный синтез УА с мультиплексором для выбора  $\alpha_j \in \{\alpha\}$  по коду  $a(t)$  со счетчиком в качестве памяти. По ГСА управления устройством криптографической защиты информации (рисунок 4.5) получена модифицированная ГСА (рисунок 4.6) и граф переходов по предложенной методике (рисунок 4.7). Для непрерывных (длинных) путей в графе переходов для представления состояний используем коды Грея. Тогда  $a(t)$  и  $a(t+1)$  будут отличаться только на одну «1» в одном из разрядов кодов  $X(t)$  и  $Y(t+1)$ . Если в графе переходов УА имеются переходы к предыдущей вершине (за счет ликвидации петель), то кроме признака  $\gamma$ , обозначающего необходимость увеличения состояния счетчика на «1», необходимо формировать и дополнительный признак, свидетельствующий о том, что переход соответствует изменению состояния на (-1). Тогда первый признак  $\gamma$  обозначим как  $\gamma_1$ , а второй как  $\gamma_2$ . Причем  $\gamma_2$  найдется как логическая сумма  $\gamma_1$  со значениями  $\alpha_j a(t)$  всех состояний соответствующих изменению  $a(t+1) = a(t) - 1$ . Для рассматриваемого примера такое условие соответствует переходу  $16 \rightarrow 15$ . Сравнение кодов состояний  $a(t)$  и  $a(t+1)$  будем производить на устройстве половинной  $(m/2)$  разрядности.

Специфика кодов Грея заключается в том, что при количестве разрядов  $p=3$ , код 7 равен коду 8, т.к. их различия определяются в более старших разрядах. Поэтому этот аспект при реализации учитывается формированием особого признака  $\delta$  или использованием значения  $p=4$ .

Очевидно, что результат сравнения (поразрядная сумма по mod 2) будет равен 1 либо для младшей группы, либо для старшей группы разрядов  $(x_m \dots x_r)$ . Обозначим  $(x_1 \dots x_p)$  группу младших разрядов и  $x_r \dots x_m$  группу старших разрядов. Здесь  $r$ - первый разряд в группе старших разрядов  $r = p+1$ , если  $m$  четно (т.е.  $m$  делится на 2) и  $r \neq p+1$ , если  $m$  нечетно. В этом случае разделение  $m$  разрядов может осуществляться пополам с перекрытием одного (общего) разряда в группе  $x_1 \dots x_p$  и  $x_r \dots x_m$ .

Выбор значений  $m, p, s, h$  поясняется в таблице 4.2 и рисунком 4.8, где обозначено:

$m$  – число разрядов кодов состояний УА;

$p$  – количество младших разрядов;

$h$  – количество перекрываемых разрядов;

$s$  – количество старших разрядов.

На рисунке 4.8 добавочный разряд  $x_0$  изображен последним после всех разрядов  $\{x\}$  только для удобства изображения. Точно также для упрощения описания далее младшие разряды будут обозначаться как  $x_p \dots x_1$ , а старшие как  $x_m \dots x_{p+1}$ .

Для непрерывной последовательности  $a(t)$ , относящейся к длинному пути, получим:

$$R_1 = x_1 \dots x_p \oplus y_1 \dots y_p, R_2 = x_2 \dots x_m \oplus y_r \dots y_m \quad (4.1)$$

Очевидно соотношение

$$R = R_1 \overline{R_2} \div \overline{R_1} R_2 \quad (4.2)$$

Здесь  $(+)$  это символ логического сложения, т.е. операция дизъюнкции ( $\vee$ ).



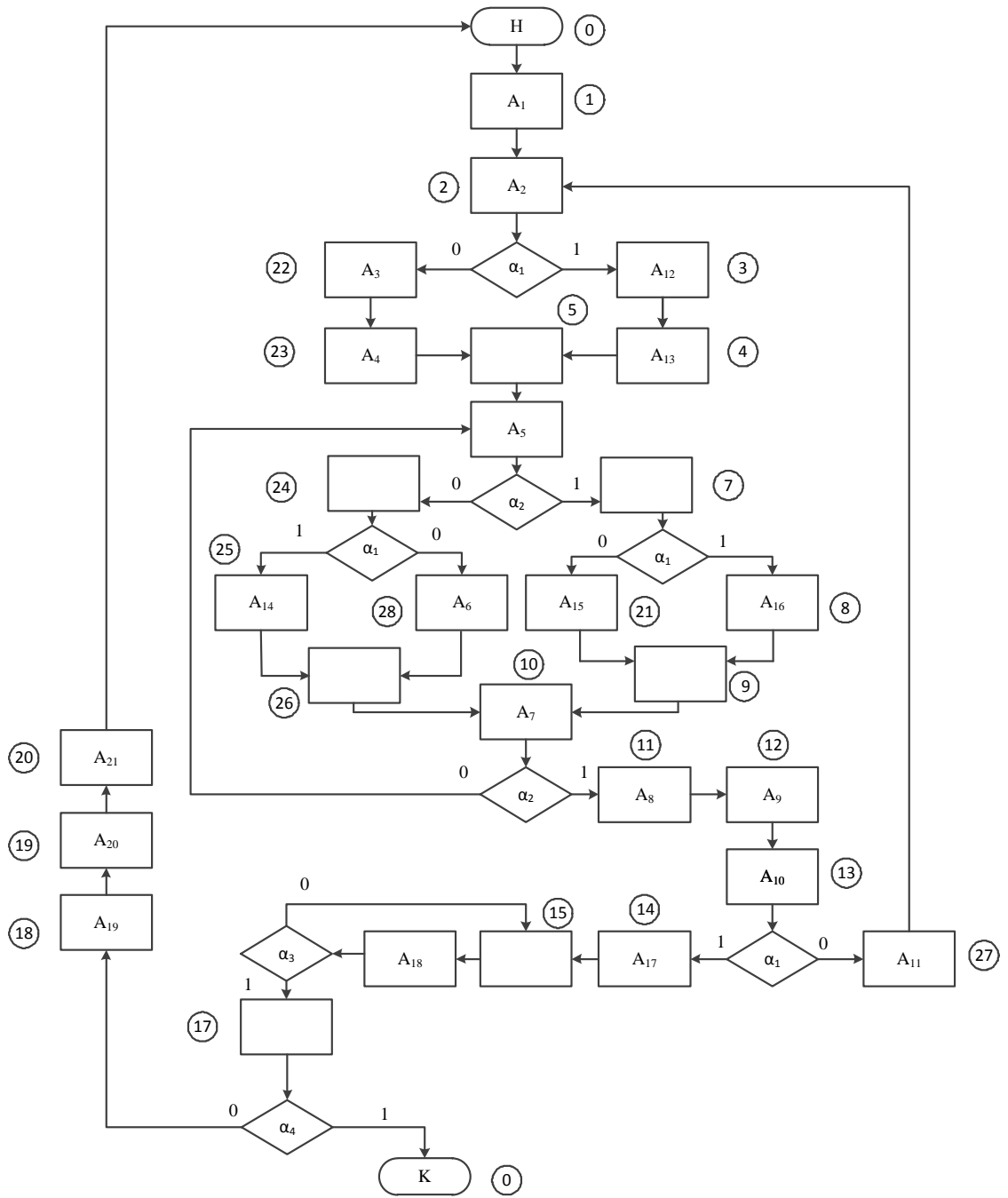


Рисунок 4.6 – Модифицированный алгоритм управления средствами защиты информации



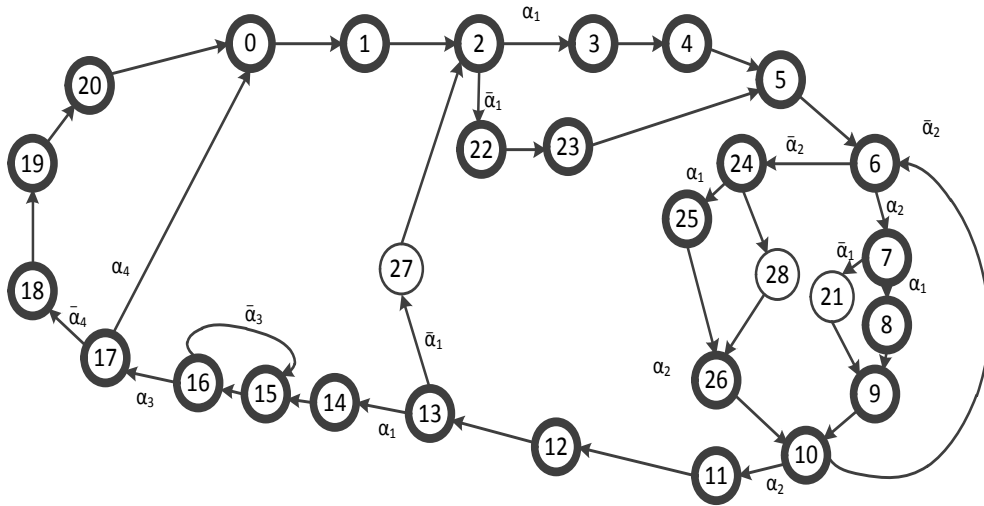


Рисунок 4.7 – Граф переходов

Вернемся к алгоритму управления устройством криптографической защиты информации (рисунок 4.6.). Для вершин  $a(t)$  и  $a(t+1)$ , не относящихся к непрерывному пути, осуществляются следующие действия:

- 1) Выбирается код (из таблицы кодов Грея) такой, чтобы для определенного номера  $a(t)$  и определяемого номера  $a(t+1)$  выполнялось условие

$$R_1 = x_1 \dots x_p \oplus y_1 \dots y_p = 1 \quad (4.3)$$

Не для всяких номеров  $a(t)$  и  $a(t+1)$  ГСА это возможно, т.к. в выборе номеров, относящихся к непрерывному (длинному) пути, нет альтернатив (они уже определены).

- 2) Если такое условие невозможно выполнить для вершин вне длинного пути без существенного увеличения разрядности  $a(t+1)$ , то вводятся промежуточные вершины  $a(t+2)$ ,  $a(t+3)$  с такими номерами, чтобы для каждой пары выполнялось условие (4.3). Для всех пар вершин получим:

$$a(t) \rightarrow a(t+1), \quad a(t+1) \rightarrow a(t+3), \quad a(t+3) \rightarrow a(t+2) \quad (4.4)$$

Для ГСА рисунка 4.6 представлен граф переходов (рисунок 4.9) с новыми введенными вершинами и соответствующими номерами вершин, для выполнения условий (4.3, 4.4) для младших разрядов.

Очевидно, что при выбранных номерах вершин кодов Грея (таблица 4.3.) с выполнением условия равенства «1» суммы по mod2 для кодов младших разрядов

при этом не будет соблюдаться одновременно условие (4.3) для старших разрядов в каждой паре вершин. В этом случае выполняются следующие действия:

- к старшим разрядам  $a(t)$ , т.е.  $x_m \dots x_r$  справа приписывается разряд  $x_0$  (более младший), а к старшим разрядам  $y_m \dots y_r$  приписывается разряд  $y_{m+1}$  слева (более старший). Далее значение разрядов  $x_0$  и  $y_{m+1}$  доопределяются так, чтобы выполнялось условие (4.5)

$$R3 = x_m \dots x_r x_0 \oplus y_{m+1} y_m \dots y_r = 1 \quad (4.5)$$

В таблице 4.4 приведены условия переходов  $a(t) \rightarrow a(t+1)$  и соответствующие им коды  $N(t)$  и  $N(t+1)$ . В таблице 4.5. приведены правила выбора номера логического условия по коду  $a(t)$ . В таблице 4.6 приведено доопределение  $x_0$  и  $y_{m+1}$  для старших разрядов кодов  $N(t)$  и  $N(t+1)$ . Как видно, в этом случае условие (4.3) выполняется.

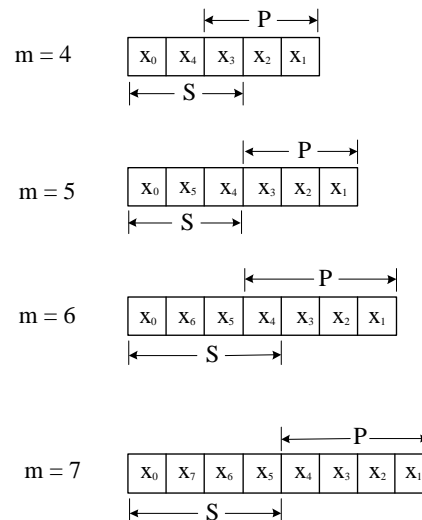


Рисунок 4.8 – Варианты выделения старших и младших разрядов

Таблица 4.2. Распределение разрядов в кодах

m	p	h	s
4	3	1	3
5	3	0	3
6	4	1	4
7	4	0	4



Таблица 4.4 Таблица переходов с кодами состояний

№	$a(t)$	$\alpha$	$a(t+1)$	$N(t)$		$N(t+1)$	
				$x_5x_4$	$x_3x_2x_1$	$y_5y_4$	$y_3y_2y_1$
1	2	$\overline{\alpha_1}$	30	00	011	11	001
2	30	$\alpha_0$	22	11	001	10	001
3	22	$\alpha_0$	5	10	101	00	111
4	6	$\overline{\alpha_2}$	24	00	101	11	100
5	24	$\overline{\alpha_1}$	20	11	100	10	110
6	20	$\alpha_0$	26	10	110	11	111
7	27	$\alpha_0$	10	11	110	01	111
8	7	$\alpha_1$	29	00	100	11	011
9	29	$\alpha_0$	21	11	011	10	111
10	21	$\alpha_0$	9	10	111	01	101
11	10	$\overline{\alpha_2}$	6	01	111	00	101
12	13	$\overline{\alpha_1}$	28	01	011	10	010
13	28	$\alpha_0$	2	11	010	00	011
14	19	$\alpha_0$	0	10	010	00	000
15	17	$\overline{\beta_4}$	0	10	001	00	000

Таблица 4.5. Соответствие логических условий номерам состояний

№	$\alpha$	$a(t)$
1	$\alpha_1$	2, 7, 13, 24
2	$\alpha_2$	6, 10
3	$\alpha_3$	16
4	$\beta_4$	17
5	$\alpha_0$	19-22, 27-30
6	$\gamma$	0-18, 24-26

Таблица 4.6. Доопределение дополнительных состояний

№	$x_5x_4$	$x_0$	$y_{m+1}$	$y_5y_4$	№	$x_5x_4$	$x_0$	$y_{m+1}$	$y_5y_4$
1	00	1	0	11	9	11	0	0	10
2	11	0	0	10	10	10	1	0	01
3	10	0	0	00	11	01	0	0	00
4	00	1	0	11	12	01	1	0	10
5	11	0	0	10	13	11	0	1	00
6	10	1	1	11	14	10	0	0	00
7	11	1	1	01	15	00	0	0	00
8	00	1	0	11					

При реализации СУА по предложенной методике оригинальность достигнута за счет следующих положений:

1. Сравнение группы младших и группы старших разрядов производится отдельно на не связанных между собой группах схем сложения по mod2.
2. Старшие группы разрядов сравниваются двумя способами: а) и б). а).  $R = (x_m \dots x_{p+1}) \oplus (y_m \dots y_{p+1})$ ; б).  $R = (x_m \dots x_{p+1} x_0) \oplus (y_{m+1} y_m \dots y_{p+1})$ .
3. Пункт «а» выполняется всегда независимо от того, вычисляются ли коды переходов через комбинационную схему  $F_1$  или через счетчик.
4. Старшие разряды сравниваются по принципу п.2а, если используется счетчик или по принципу п.2б, если используется схема  $F_1$ .
5. При использовании принципа 2.б должно выполняться условие  $R = R_1 R_2^{\circ}$ . Здесь  $R_2^{\circ}$  это сумма по mod2 старших разрядов по условию 2б.

Реализация условий 1-5 достигается за счет включения в структуру СУА (рисунок 4.10) специальной схемы распределения информации (рисунок 4.11) и схемы принятия решений (рисунок 4.12).

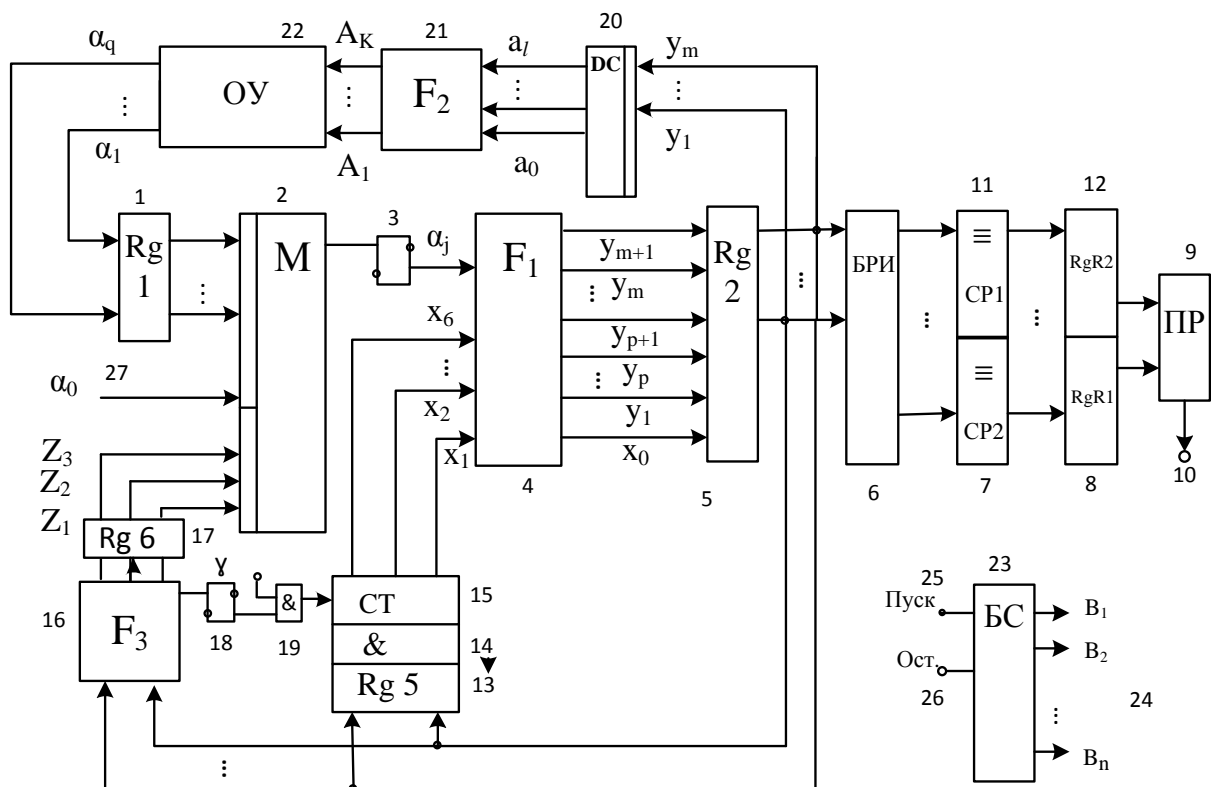


Рисунок 4.10 - Самоконтролируемый автомат с разделением кодов

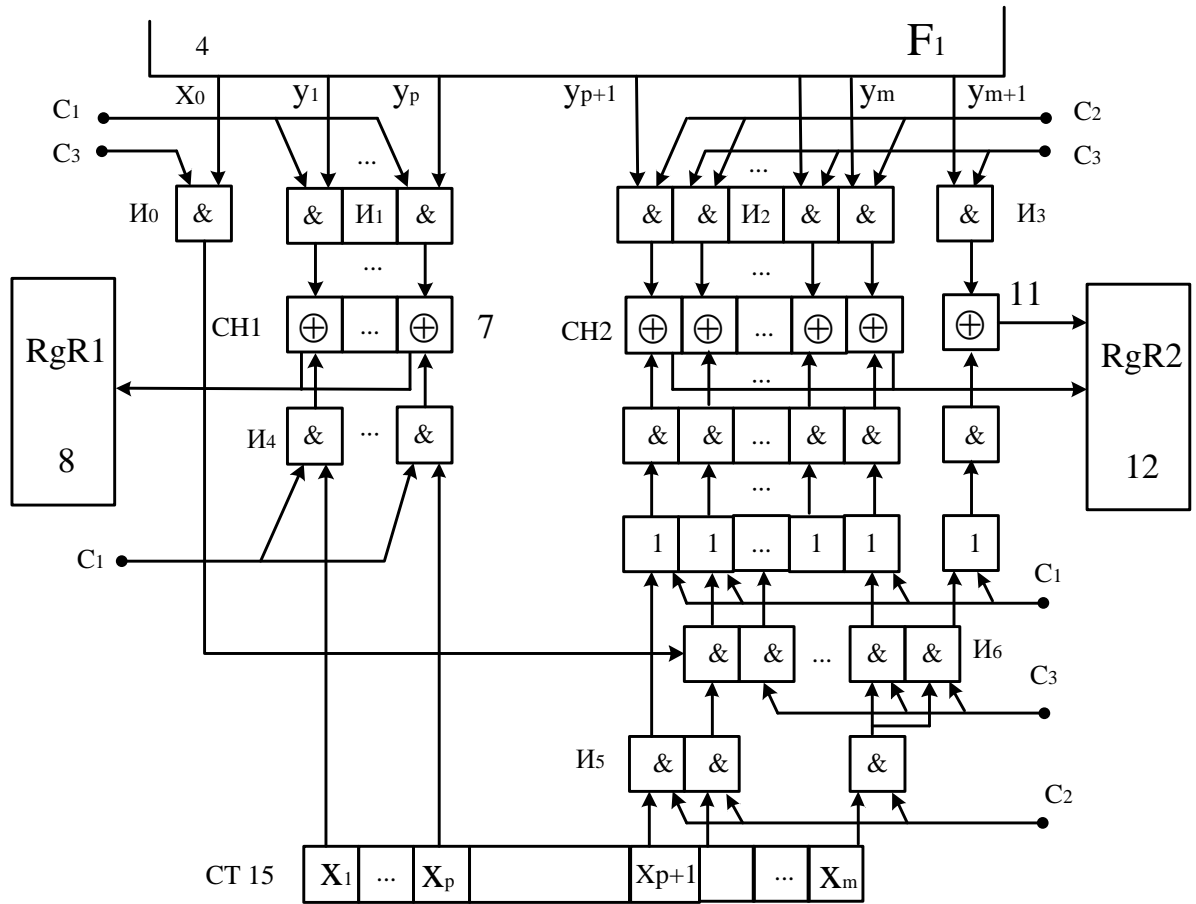


Рисунок 4.11 - Организация схемы распределения информации

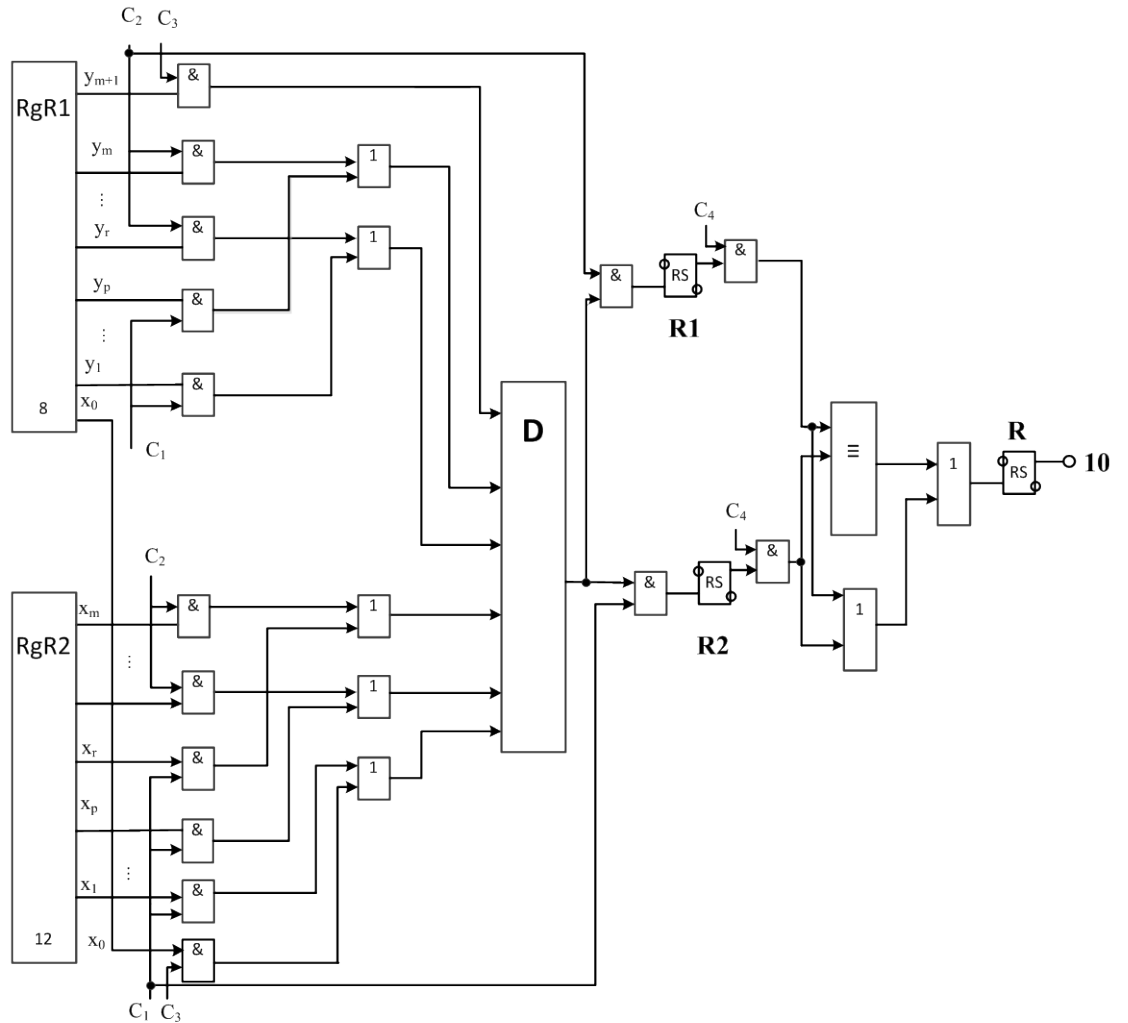


Рисунок 4.12 – Схема принятия решений

В структуре УА количество разрядов младшей группы  $p=3$ . На графе переходов произведена замена  $\alpha_4$  на  $\beta_4 = \overline{\alpha_4}$ . Специфика кодов Грея заключается в том, что при  $p=3$  код 7 равен коду 8, т.к. их различие определяется в более старших разрядах. Поэтому этот аспект при реализации учитывается формированием особого признака ( $\delta$ ) в комбинационной схеме  $F_2$  или используется значение  $p=4$ .

Методика синтеза СУА не зависит от числа разрядов  $m$  кода  $a(t)$ , однако при реализации СУА следует учитывать правила выбора количества старших, младших и перекрываемых разрядов в зависимости от значения  $m$ .

Новый метод контроля по сравнению с известными методами позволяет построить самоконтролируемый автомат с минимальными затратами

оборудования и высоким быстродействием. Структура блоков  $F_1(4)$ ,  $F_2(16)$ ,  $F_3(21)$ , БРИ(6) и ПР (9) не типовая и определяется в соответствии с предложенным методом. Схемы сравнения СР(8) и СР(12) представляют собой набор из  $m/2$  одноразрядных схем не тождественности (mod2). Блок синхронизации БС(23) реализует 2 режима управления:

- управление режимом функционирования самого УА,
- управление режимом самоконтроля.

ГСА управления БС(23), учитывающая оба режима, представлена на рисунке 4.13. Причем при использовании асинхронного режима выполнения операторов действия в ГСА необходимо включить участок алгоритма рисунка 4.14. Расшифровка команд синхронизации  $V_1 \dots V_{14}$  приведена в таблице 4.7.

Последовательность из четырех непересекающихся во времени синхроимпульсов реализуется с помощью обычного генератора импульсов с одним выходом, двухразрядным счетчиком и дешифратором.

Для кодирования состояний использован обычный двоичный позиционный код (ДПК). В таблице 4.8 представлено соответствие команд  $V_i$  выходам дешифратора ДС (20), а выбор  $\alpha_j$  по коду состояний  $a(t)$  в таблице 4.9. В таблице 4.10 представлены переходы УА вне счетчика. В таблице 4.11 представлены правила выбора логических условий.

Блок синхронизации целесообразно реализовать также в виде УА нового типа.

В предлагаемом СУА нет специальных схем, дублирующих вычисление кода  $\{Y\}$ . Отсутствие дублирующих схем – главный фактор снижения затрат на встроенные схемы самоконтроля. Оригинальность решения в организации СУА по предложенной методике определяется использованием специальной процедуры выбора кодов для младших и старших групп разрядов состояния автомата. Причем каждая из групп имеет разрядность  $(m/2)$ .

Предлагаемый метод контроля и его реализация в виде СУА являются новыми, т.к. такой метод и структура СУА не известны по публикациям или патентам.



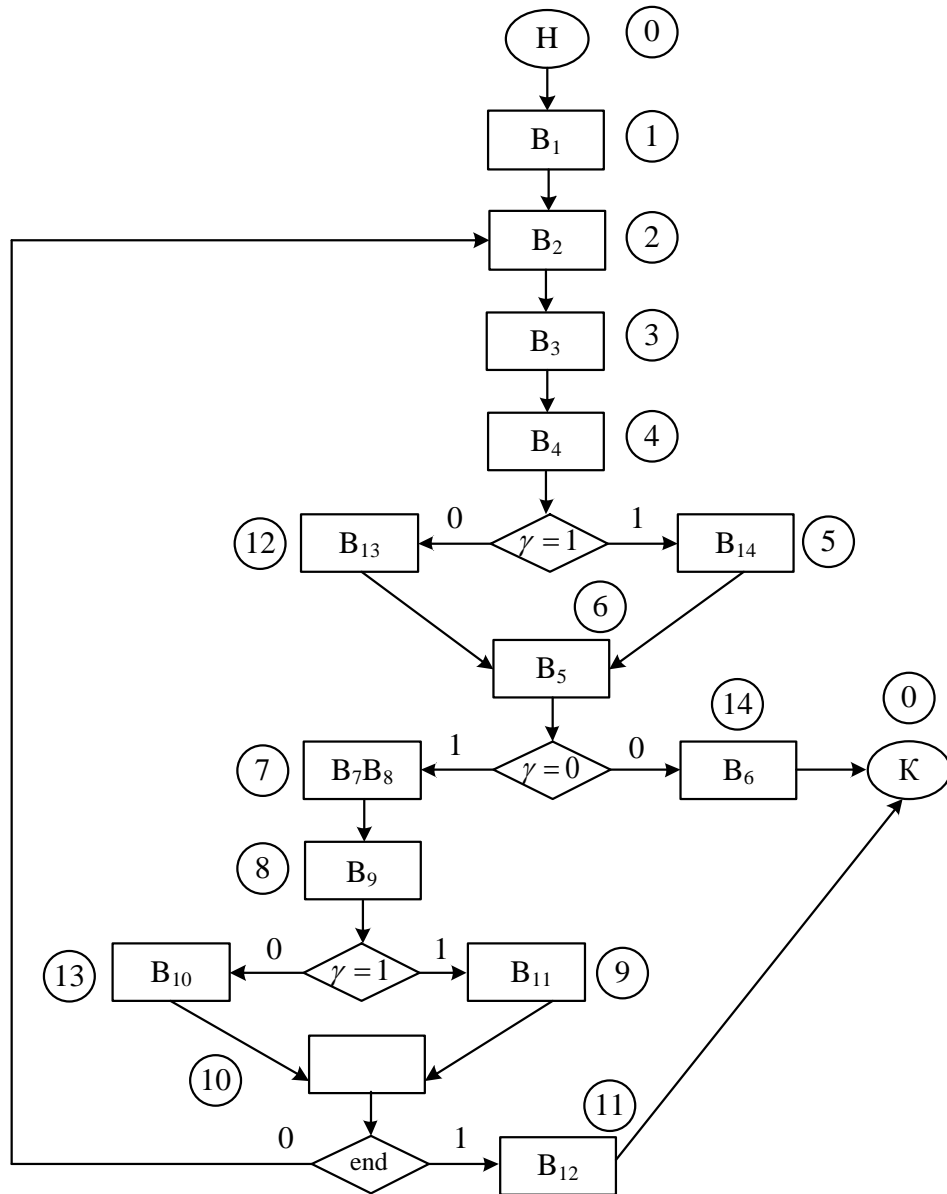


Рисунок 4.13 – Алгоритм управления самоконтролируемым автоматом

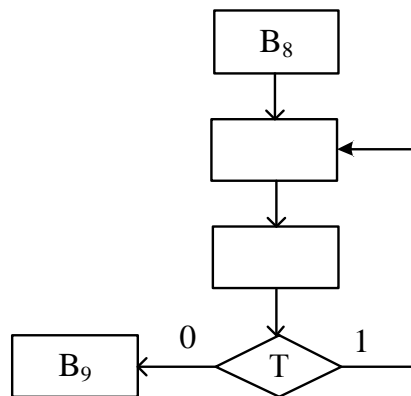


Рисунок 4.14 – Организация операции задержки

Таблица 4.7. Расшифровка команд синхронизации

$B_i$	Содержание
1	Нач. установка: $0 \rightarrow Cч(15), 0 \rightarrow Pг(0) \rightarrow \langle RS(18) \rangle; \langle RS(3) \rangle$
2	Опрос Pг1, Pг6 (17), Запись RS(3)
3	Опрос F <sub>1</sub> (4) Запись Pг2(5)
4	Опрос БР(6) – С1С2 Запись R1, R2
5	Опрос ПР (11)
6	Индикация ошибки
7	Опрос DC(20)
8	Пуск A <sub>i</sub>
9	Запись Pг1, Pг4 (13) Pг (17)
10	$\langle Pг4 (13) \rangle \rightarrow Cх (15)$
11	$Cч (15) = \langle Cч (15) \rangle + 1$
12	Индикация окончания программы
13	Опрос БР (6) –С3 Запись R2
14	Опрос С2

Таблица 4. 8. Соответствие управляющих команд  
выходам дешифратора

$B_i$	Выход DC (20)	$B_i$	Выход DC (20)
1	1	8	7
2	2	9	8
3	3	10	13
4	4	11	9
5	6	12	11
6	14	13	12
7	7	14	5

Таблица 4.9 Соответствие логических условий  
номерам состояний блока синхронизации

№	$\alpha$	$a(t)$
1	$\gamma_1$	4
2	$\delta$	6
3	end	10
4	$\alpha_0$	11-14
5	Сч	0-10
6	$\gamma_2$	8

Таблица 4.10 таблица переходов

№	$a(t)$	$N(t)$	$\alpha$	$a(t + 1)$	$N(t + 1)$ $f_4 f_3 f_2 f_1$
1	4	0100	$\overline{\alpha_1}$	12	1 1 0 0
2	12	1100	$\alpha_0$	6	0 1 1 0
3	6	0110	$\overline{\alpha_2}$	14	1 1 1 0
4	14	1110	$\alpha_0$	0	0 0 0 0
5	8	1000	$\alpha_1$	13	1 1 0 1
6	13	1101	$\alpha_0$	10	1 0 1 0
7	10	1010	$\overline{\alpha_3}$	2	0 0 1 0
8	11	1011	$\overline{\alpha_4}$	0	0 0 0 0

Таблица 4.11. Логические условия блока синхронизации.

Условия	Семантика
$\delta = 1$	ошибка в блоке синхронизации
$\Gamma = 1$	окончание действия $A_j(t)$
$\gamma = 1$	работа со счетчиком Грея
$e = 1$	программа управления выполнена (end)
ERR	ошибка в самоконтролируемом автомате

## МЕТОДОЛОГИЯ ПРОЕКТИРОВАНИЯ САМОКОНТРОЛИРУЕМЫХ УПРАВЛЯЮЩИХ АВТОМАТОВ СЛОЖНЫХ ТЕХНИЧЕСКИХ СИСТЕМ РЕАЛЬНОГО ВРЕМЕНИ

Комплексное использование предложенных методов и средств реализации представляют собой новую методологию синтеза УА принципиально отличающуюся от методологии проектирования на базе автоматов Мура и Мили. Эта методология представлена в виде последовательности этапов как ГСА рис.5.41. Расшифровка операторов действия представлена в таблице 5.27.

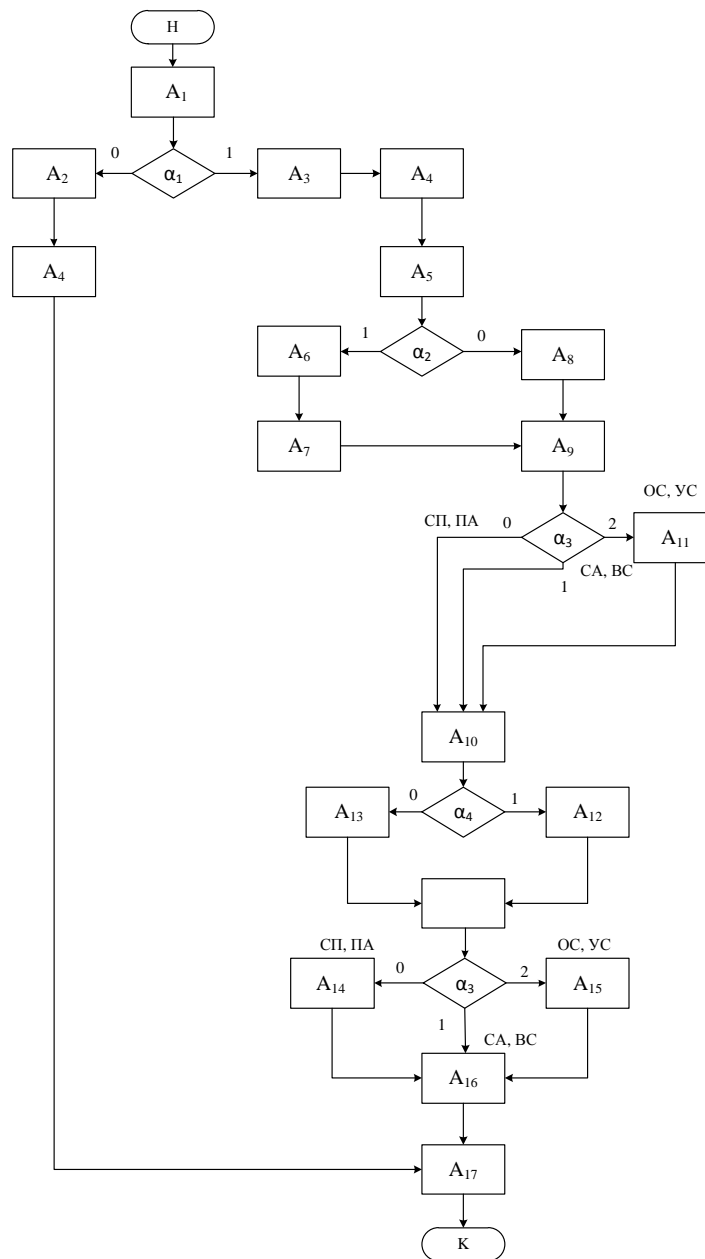


Рисунок 5.41 – Последовательность проектирования УА

Таблица 5.27 Процедуры (этапы) проектирования УА сложных технических систем реального времени.

$A_i$	Семантика
1	Разработка ОСА
2	Проектирование УА Мура
3	Модификация ГСА
4	Переход к графу УА
5	Выделение непрерывных путей с числом состояний N
6	Память на счетчике
7	Дополнительная модификация ГСА
8	Память на регистрах
9	Проектирование УА с мультиплексором
10	Классификация УА
11	Декомпозиция УА
12	Мажоритарное резервирование с $\alpha$ и $\bar{\alpha}$
13	Разделение $F_1$ на старшие и младшие разряды
14	Корректирующие коды
15	Мажоритарное резервирование
16	Код Грея
17	Функционально-электрический синтез
$\alpha_1$	$q > 3$
$\alpha_2$	$N > 7$
$\alpha_3$	0 – УА малой и средней сложности 1 – Сложные УА и высокосложные 2 – Особосложные и ультрасложные УА
$\alpha_4$	0 – нормальные условия эксплуатации 1 – экстремальные условия эксплуатации

Преимущества УА с использованием новой методологии.

- 1) Повышенный уровень надежности за счет уменьшения объема комбинационной схемы переходов в  $2^{q-1}$  раз.
- 2) Эффективное использование счетчика с дополнительным уменьшением комбинационной схемы в 2-3 раза.
- 3) Новые способы повышения надежности УА, эффективно реализуемые только в предложенном УА.
- 4) Упрощение процедуры контроля и снижения встроенного оборудования.
- 5) Упрощение процедуры моделирования УА.
- 6) Возможность эффективной реализации УА как в аппаратном, так и в программном варианте для микроконтроллеров.

- 7) Упрощение процедуры перевода УА на новую ГСА за счет замены трех зон памяти  $F_1, F_2, F_3$ .
- 8) Существенное снижение энергопотребления УА.
- 9) Упрощение структурной организации взаимодействующих автоматов.

#### Выводы по главе 4

1. Для контроля управляющих автоматов предложена модификация  $n$ -разрядного кода с фиксированным числом  $k$  единиц (код  $kCn$ ) с представлением кода последующего состояния автомата в виде  $p$ - трехразрядных групп с одной «1» в группе (код  $pCn$ ). За счет преобразования кода  $pCn$  в специальный непозиционный двоичный код (ДНК) контроль автоматов осуществляется как на входе, так и на выходе комбинационной схемы. При использовании кодов  $pCn$  и ДНК декодирование предельно упрощено по сравнению с кодом  $kCn$ , в котором число разрядов  $n$  увеличивается больше чем в два раза по сравнению с двоичным  $m$ - разрядным позиционным кодом (ДПК). За счет разделения кодов состояний автомата на младшие и старшие разряды количество групп кода  $pCn$  снижается в два раза для автоматов нового типа. В этом случае введенная для контроля избыточность в коде  $pCn$  не приводит к увеличению разрядности кодов состояний автоматов нового типа, т.к. значения младших и старших разрядов вычисляются последовательно и фиксируются схемами контроля половинной разрядности.

2. Предложен новый метод резервирования комбинационных схем переходов, основанный на введении схемы обратного преобразования кода последующего состояния в код предыдущего состояния с проверкой на совпадение кодов вычисленного схемой обратного преобразования и существовавшего кода на входе комбинационной схемы прямого преобразования. Дублируется как основная схема, так и схема обратного преобразования. В такой организации резервирования достигается более высокое быстродействие фиксации исправной схемы при наличии неисправности. При разделении схем прямого и обратного преобразования на две части с опросом частей по значению  $\alpha_j$  и  $\overline{\alpha_j}$  за счет

подбора исправных полусхем достигается возможность самовосстановления автомата при наличии отказов. Тем самым обеспечивается работоспособность управляющего автомата в распределенных системах недоступных для ремонта и в сложных эксплуатационных условиях с повышенным уровнем вероятности выхода из строя при наличии интенсивных помех, предельных климатических воздействий, вибрационных нагрузках и др.

3. Предложена методика синтеза самоконтролируемых управляющих автоматов на основе кода Грея в структурной организации автоматов нового типа с разделением на младшие и старшие разряды. Контроль переходов по счетчику Грея реализуются традиционно, а для переходов вне счетчика вводятся дополнительные вершины в графе переходов, при этом номера соседних вершин выбираются так, чтобы младшие разряды также могли контролироваться по правилу кодов Грея. При этом для кодов старших разрядов вводится дополнительный разряд, значение которого доопределяется для выполнения условий контролируемости кодом Грея. Метод практически не снижает быстродействие управляющего автомат при минимальных затратах встраиваемого оборудования для контроля.

Все методы контроля оригинальны и применимы только к автоматам нового типа. На структурную организацию самоконтролируемых автоматов по 2 и 4 методу получены патенты.

По материалам главы 4 автором опубликованы работы [1-4,9,12,25,31,37,41,42,52,55,71] и получены патенты [59,69].

## ГЛАВА 5. СПЕЦПРОЦЕССОРЫ И УПРАВЛЯЮЩИЕ АВТОМАТЫ СЛОЖНЫХ ТЕХНИЧЕСКИХ СИСТЕМ

В качестве практических применений выбраны задачи, имеющие важное народно-хозяйственное значение, для которых диссертантом разработаны спецпроцессоры, алгоритмы и устройства управления. Спецпроцессоры оригинальны по своей структурной организации, на большинство из них получены патенты.

### 5.1. Управление спецпроцессором корреляционно-экстремальной навигации

Корреляционно-экстремальное сравнение изображений используется для автоматизации управления курсом судна (автолоцман), для управления летательными аппаратами, точного управления исполнительным инструментом при обработке плоских деталей на станках с числовым программным управлением [158-167]. Все эти задачи классифицируются как задачи навигации.

Проектирование спецпроцессоров корреляционно-экстремальной навигации (СКЭН) проводится по модели ФИЛАУ. Кадр неизвестного изображения может фиксироваться радиолокационными, гидро-акустическими или оптическими средствами в зависимости от типа самой навигационной системы. Необходимо осуществить поиск эталонного изображения  $m \times n$  в поле неизвестного изображения  $M \times N$  и зафиксировать координаты участка с максимальной корреляцией с эталоном. Для фотоматриц, регистрирующих неизвестное изображение, размером  $M = N = 64, 256, 1024$ , выбирается  $m = n = 8, 16, 32$  (рисунок 5.1).

В диссертационной работе рассматривается проектирование СКЭН с обработкой бинарных изображений.

Функции всех подсистем ФИЛАУ определены диссертантом в следующей последовательности:  $\Phi \rightarrow \text{И} \rightarrow \text{А} \rightarrow \text{Л} \rightarrow \text{У}$ .

**Функциональная подсистема.** Главной функцией СКЭН является нахождение корреляции между равными участками неизвестного ( $m \times n$ ) и эталонного ( $m \times n$ ) изображения. Чаще всего  $m = n = 8, 16$ .



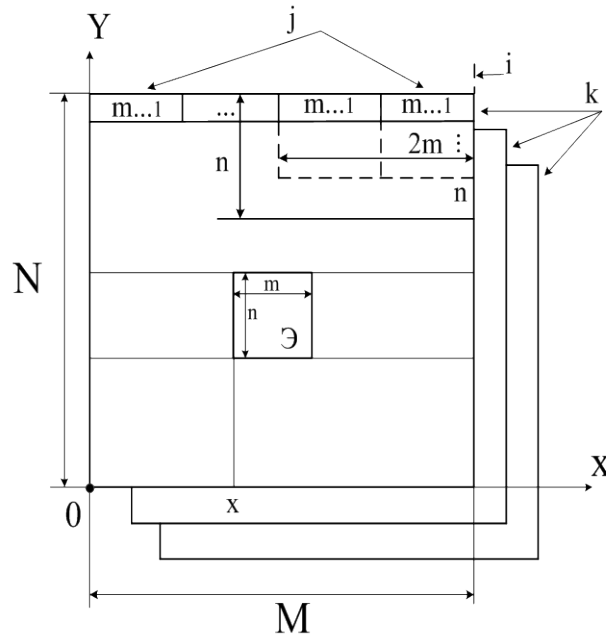


Рисунок 5.1 – Размещение информации в ОЗУ

В диссертационной работе процессор нахождения корреляции между образцом неизвестного изображения (НИО) и образцом эталонного изображения (ОЭИ) представлен укрупненной блок-схемой (рисунок 5.2). НИО записан в магазинной памяти (стек первого типа СТП1) с числом регистров  $n$  разрядностью  $2m$ . Особенностью СТП1 для НИО является то, что стековая организация по координате  $Y$  создана не на запоминающих регистрах, а на регистрах сдвига по координате  $X$ . Заполнение СТП1 будет производиться параллельными словами разрядностью  $2m$  с последующим сдвигом всего НИО по координате  $X$  справа налево. Причем СТП организована в виде двух блоков СТП1(2) и СТП2(4), которые соединяются в одну СТП (СТП1  $\rightarrow$  СТП2) с помощью схем «И» (3). ОЭИ записывается также в СТП4 разрядности  $m$ , но заполнение его производится параллельными словами разрядности  $(n)$ , причем в СТП2 организованы также регистры сдвига по координате  $X$ , но слева направо, с возможностью обратной записи информации в СТП2 (рисунок 5.3). На рисунке 5.3 содержимое СТП2 для дальнейшей обработки переписывается в СТП3.

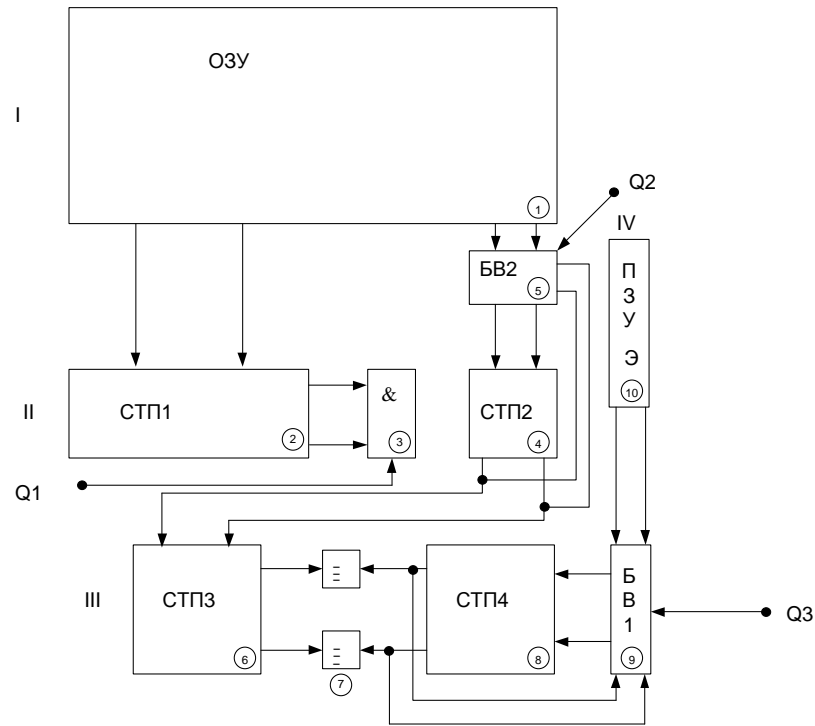


Рисунок 5.2 – Блок – схема функциональной и информационной подсистем спецпроцессора

В принципе функции СТП2 и СТП3 можно совместить. Значения каждого бита «встречных» слов разрядностью  $n$  сравнивается с помощью операции тождественности

$$f_k = a_k(\text{э}) \equiv a_k(\text{н}) \equiv a_k(\text{э})a_k(\text{н}) + \overline{a_k(\text{э})}\overline{a_k(\text{н})}$$

$$k = 0, 1, \dots, n-1.$$

При каждом сравнении столбцов НИО и ОЭИ находится сумма всех  $f_k$ , т.е.

$$F(i) = \sum_{k=0}^{n-1} f_k.$$

Здесь  $(i)$  – номер столбца ( $i = 0, 1, \dots, m$ ). Суммирование осуществляется на аналоговом сумматоре, имеющем входные масштабные преобразователи напряжений  $U(k) = f_k$  так, чтобы сумма всех  $f_k$  (т.е.  $F(i)$ ) не превосходила  $U_{\max}$  для аналогового сумматора в используемой элементной базе. Значение  $U(i) = hF(i)$  сравнивается с пороговым значением  $U_{\text{пор}}(i)$ .

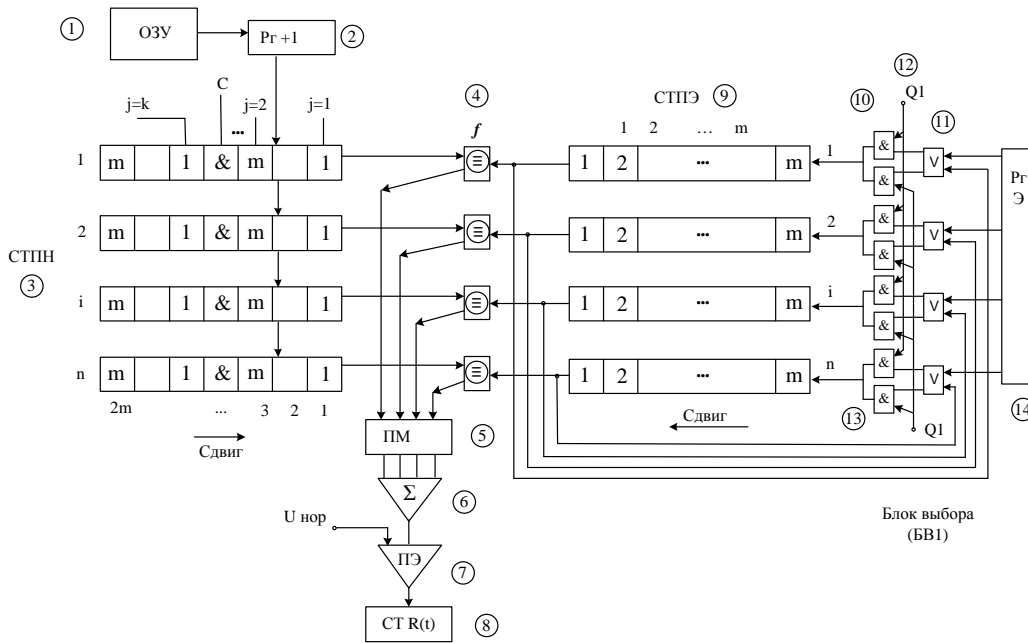


Рисунок 5.3 - Функциональная схема спецпроцессора корреляционно-экстремальной системы навигации

Значение  $U_{пор}(i)$  выбирается таким образом, чтобы корреляция была равна  $0,75 \div 0,8$  для каждого столбца, т.к. коэффициент корреляции равный 1 для всех столбцов не может быть зафиксирован из-за наличия как внешних, так и внутренних помех при фиксации НИО. Результирующее значение степени корреляции по каждому столбцу в форме логической «1» или «0» передается на счетный вход счетчика результатов (СТ R(t)). После полного цикла ( $i=0,1,\dots,m-1$ ) сравнения НИО и ОЭИ в СТ R(t) будет зафиксировано текущее значение суммы R(t). Предложенная организация «процессора» для определения степени корреляции между НИО и ОЭИ позволяет находить значение R(t) с наивысшим быстродействием. Каждое значение R(t) затем запоминается на регистре памяти, т.е. переводится в состояние R(t-1), которое будет сравниваться с новым R(t), полученным на этапе определения степени корреляции того же ОЭИ с новым участком НИО. Столь подробное описание Ф-подсистемы СКЭН дано потому, что именно организация этой подсистемы определяет структуру как (И), так и (А), а следовательно (Л) и (У) подсистем согласно предложенной методике синтеза СПР.

**Информационная подсистема** СКЭН имеет четыре уровня:

I - информационное поле ОЗУ с массивами слов разрядностью  $2m$ ;

II - промежуточная стековая память СТП с параллельной записью информации словами разрядности  $2m$ . СТП1- это  $n$  регистров сдвига разрядностью  $2m$  со сдвигом слева направо;

III - две стековые памяти СТП3 и СТП4, которые составляют основу «процессора» корреляции в структуре Ф-подсистемы;

IV –постоянное запоминающее устройство ПЗУ для хранения нескольких ОЭИ. Возможна смена эталона для нового кадра  $M \times N$ . На рисунке 5.3 все элементы И-подсистемы помечены номерами. СТП второго уровня имеет специальную организацию в виде двух СТП1(2) и СТП2(4), соединенных через блок &(3) по сигналу  $Q1$  в единую структуру, обеспечивающую сдвиг всего массива  $2m \times n$  по горизонтали слева направо. После каждого сдвига на один разряд (смена столбцов) содержимое СТП(2) –(4) переписывается в СТП3-(6) и по цепи обратной связи через блок логического выбора БВ2(5) снова заносится в СТП2-(4). Блоки БВ2(5) и БВ1(9) - это набор из двух «линеек» схем «И», выходы которых поразрядно объединяются схемами ИЛИ, имеющими выходы на СТП2 (4) и СТП4(8). По сигналу  $Q2$  передача осуществляется от ОЗУ(1) на СТП2(4); по сигналу  $\bar{Q}2$  от СТП2(4) через БВ2(5) снова на СТП2(4). Аналогична передача по сигналу  $Q3$  от ПЗУЭ на СТП4(8); а по сигналу  $\bar{Q}3$  от СТП4(8) через БВ1(9) снова на СТП4(8).

Именно такая 4- уровневая организация И-подсистемы при минимальных затратах оборудования позволяет обеспечить «сканирование» ОЭИ по полю неизвестного размера  $M \times N$  со сдвигом на один разряд как по координате  $X$ , так и по координате  $Y$ . Начальное положение ОЭИ отнесено к правому верхнему углу массива ОЗУ – $M \times N$ .

**Адресная подсистема** реализуется с помощью счетчиков, обозначенных на общей схеме СКЭН как СТq (21), СТj(17), СТk(19).

**Логическая подсистема** так же реализуется фиксацией сигналов переполнения (или превышения) заданного количества накоплений значений  $i, j, k, l, p, r$ . При этом обозначено:

$m$  –разрядность строки,  $n$  –количество строк;

$i = 0, 1, 2, \dots, m-1$  – количество сдвигов для определения корреляции ОЭИ и НИО;  
 $j = 0, 1, \dots, \frac{M}{2m} - 1$  – количество чисел разрядностью  $2m$  в строке ФМ;  
 $l$  – количество сдвигов в СТП1(6);  
 $k$  – индекс строки ФМ,  $k = 0, 1, \dots, (N- n)-1$ ;  
 $p$  – индекс адреса ОЗУ;  
 $r$  – счет строки в СТП1(6), ОЗУ;  
 $F_1(j,k,l)$  – комбинационная схема определения начального адреса ОЗУ;  
 $F_2(j, k, l)$  – комбинационная схема определения координаты X,Y;  
 $R(t), R(t-1)$  – результаты корреляции НИО и ОЭИ;  
ПК – преобразователь кодов  $j, k, l$  в координаты X,Y.

Оригинальная структурная организация всего спецпроцессора корреляционно-экстремальной навигации (СКЭН) приведена на рисунке 5.4. Здесь дана новая нумерация всех блоков, не совпадающая с нумерацией рисунка 5.2 (которая была дана с целью пояснения структуры именно И- подсистемы).

Кроме уже известных блоков СТП 1, 2, 3,4; ПЗУ; ОЗУ; регистров  $RgK(20)$ ,  $RgR(t-1)(14)$   $Rgl(30)$ , блока выбора БВ1, 2 (23), (5), блока &(7); счетчиков СТ(13), СТ(3),(17), (19), (25), (26), (27), (28), (29), (21) в спецпроцессоре используется также сумматор  $\Sigma$  (15), преобразователь кодов ПК(8), блок индикации (29); кодирующий блок CD(18) для определения начального адреса считывания ОЗУ, а также фотоматрица ФМ(1) и блок адресации, определяющий не только начальный адрес записи, но и содержащий блок преобразования выходных сигналов ФМ(1) в двоичные слова  $2m$  подлежащие записи в ОЗУ(4). Вместо фотоматрицы ФМ(1) может быть использовано любое другое устройство регистрации изображений, определяемое проблемной ориентацией объекта навигации (корабль, самолет, станок и др.).

Прежде чем определить функции У- подсистемы выделим пять основных процедур (ПР0, 1, ..., 4).

- ПР «0» - перепись информации ФМ в ОЗУ, т.е.  $\langle \text{ФМ} \rangle \rightarrow \text{ОЗУ}(4)$ , рисунок 5.5, таблица 5.1;

- ПР «1» - <ОЗУ> → СТП1 (6-7-8), рисунок 5.6, таблица 5.3;
- ПР «2» - СТП2(6) – СТП3 (9), рисунок 5.7, таблица 5.4;
- ПР «3» - <ПЗУ> → СТП4(24), рисунок 5.8 а, таблица 5.5;
- ПР «4» - корреляция с записью  $R(t)$ , рисунок 5.8 б, таблица 5.6.

Алгоритм управления СКЭН представлен на рисунке 5.10 с расшифровкой операторов  $S_1 \dots S_{19}$  в таблице 5.7 .

Управляющая подсистема (У) представлена в виде ведущего УА с графом переходов рисунок 5.11 и пяти автоматов нижнего уровня (процедуры соответствующие состояниям 1, 2, 4, 6, 8 на графе переходов рисунок 5.10).

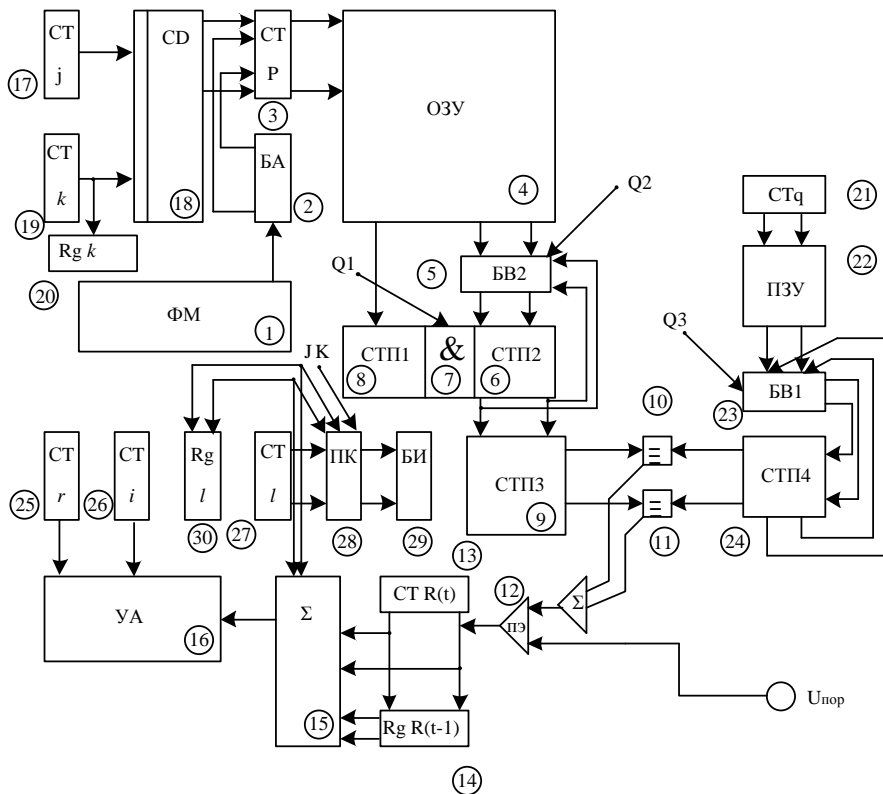


Рисунок 5.4. – Спецпроцессор корреляционно-экстремальной системы навигации

Пр «0»

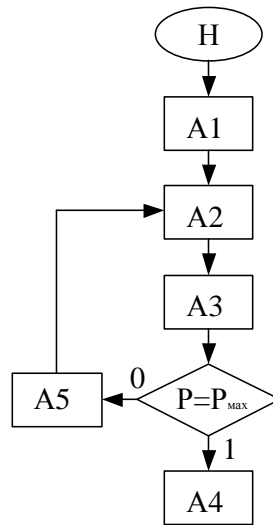


Рисунок 5.5 – Алгоритм «0» процедуры

Таблица 5.1. Операторы «0» процедуры

А	Содержание
1	$P=0$
2	$BA \rightarrow CчP$
3	$\langle \Phi M \rangle \rightarrow OЗУ$
4	Пустой оператор
5	$P = P+1$

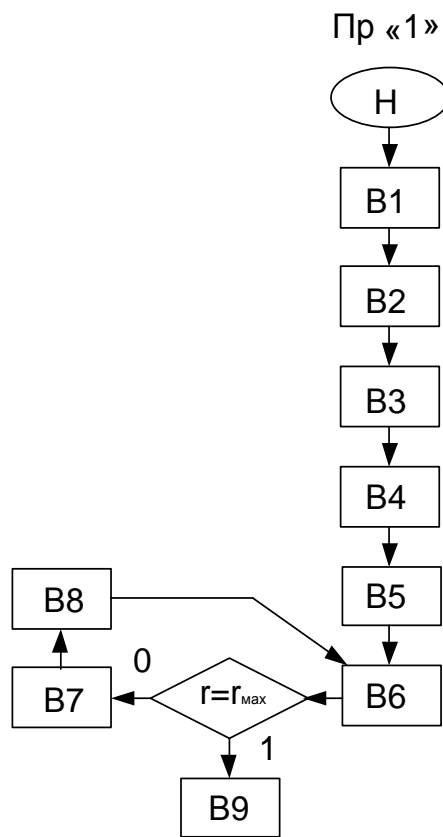


Рисунок 5.6 –Алгоритм «1» процедуры

Таблица 5.2. Операторы «1» процедуры

В	Содержание
1	$k=0$
2	$j=0$
3	Соединение регистров (7)
4	Q1
5	Опрос СП(18) → запись Сч(3)
6	< ОЗУ> →СТП1,2 (6, 7 ,8)
7	$r =r+1$
8	$p=p+1$
9	Пустой оператор



Пр «2»

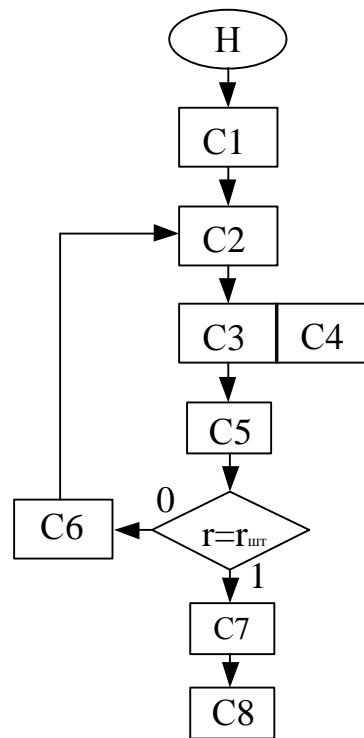


Рисунок 5.7 –Алгоритм «2» процедуры

Таблица 5.3. Операторы «2» процедуры

С	Содержание
1	Q2
2	$r=0$
3	$\langle \text{СТП2} \rangle \rightarrow \text{СТП3}$
4	$\langle \text{СТП2} \rangle \rightarrow \text{СТП2}$
5	Сдвиг СТП2, 3
6	$r = r+1$
7	Q2
8	Пустой оператор

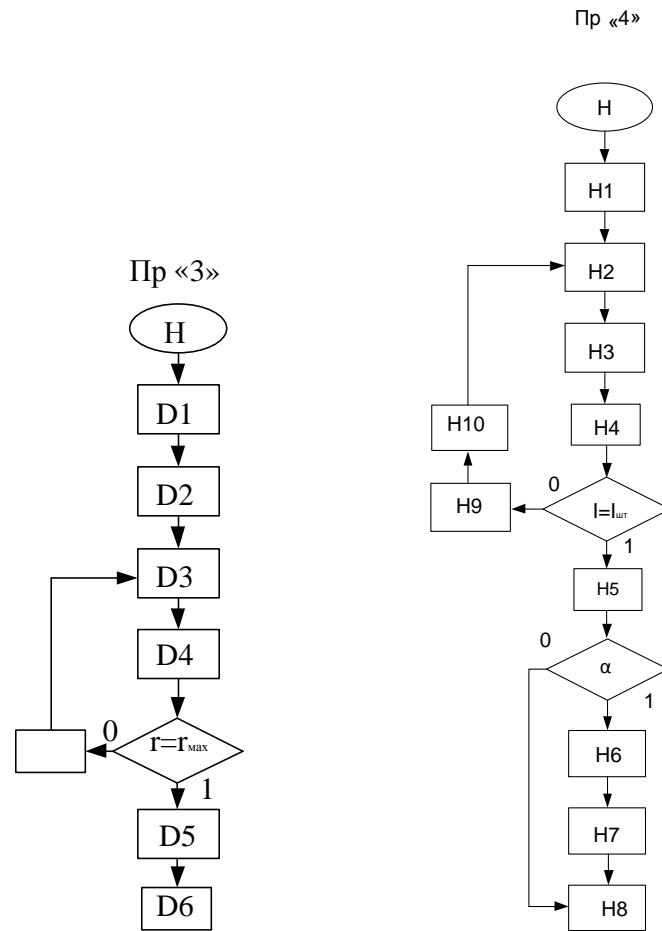


Рисунок 5.8 - Алгоритм а) «3» процедуры, б) «4» процедуры

Таблица 5.4. Операторы «3» процедуры

D	Содержание
1	$q=0$
2	$Q_3 = 1$
3	<ПЗУ> -СТП4 (24)
4	Сдвиг СТП4
5	$\overline{Q_3}$
6	Пустой оператор
7	$r=r+1$

Таблица 5.5. Операторы «4» процедуры

Н	Содержание
1	$\overline{Q_3}$
2	«0»→Сч R(t)
3	i=0
4	Опрос ( $\equiv$ ); Запись в Сч R(t)
5	<СчR(t)> - <PгR(t-1)>
6	R(t) → R(t-1)
7	<Счi> →Pгi <Сч k> →Pгk <Счj> →Pгj
8	Пустой оператор
9	Сдвиг СТПЗ, 4
10	i = i+1

Однако в данном случае как ведущий автомат УА по графу рисунка 5.9, так и «процедурные» УА относятся к классу сверх простых УА. Поэтому всю У-подсистему целесообразно реализовать в виде единого (не иерархического) УА по алгоритму рисунка 5.11 с графом переходов, представленном на рисунке 5.12.

В качестве памяти УА используется суммирующий счетчик, содержимое которого изменяется при состояниях 0-40, 42, 45. Правила выбора  $\alpha_j \in \{\alpha\}$  представлены в таблице 5.7, функции переходов в таблице 5.8.

Тогда для УА при числе состояний  $N = 53$  ( $m=6$ ) и числе логических условий  $q=10$  при классическом автомате Мура потребовалось бы ПЗУ объемом  $V = m2^{m+q} = 6 \cdot 2^{16}$ .  $V_p = 8 \cdot 2^{16} = 2^{19} = 0.5 \text{ Мб}$ .

При реализации УА нового типа  $W = m2^{m+1}$ .  $W_p = 8 \cdot 2^7 = 2^{10} = 1 \text{ кб}$ .

При использовании счетчика  $W_{pc} \sim 0.5 \text{ кб}$ .

Затраты как на схему переходов ( $F_1$ ), так и на схему адресации ( $F_3$ ) мультиплексора  $\approx F_1$ . Общий объем ПЗУ сокращается в  $Q = V_p / W_p = 2^{19} / 2^{10} = 2^9 = 512$  раз.

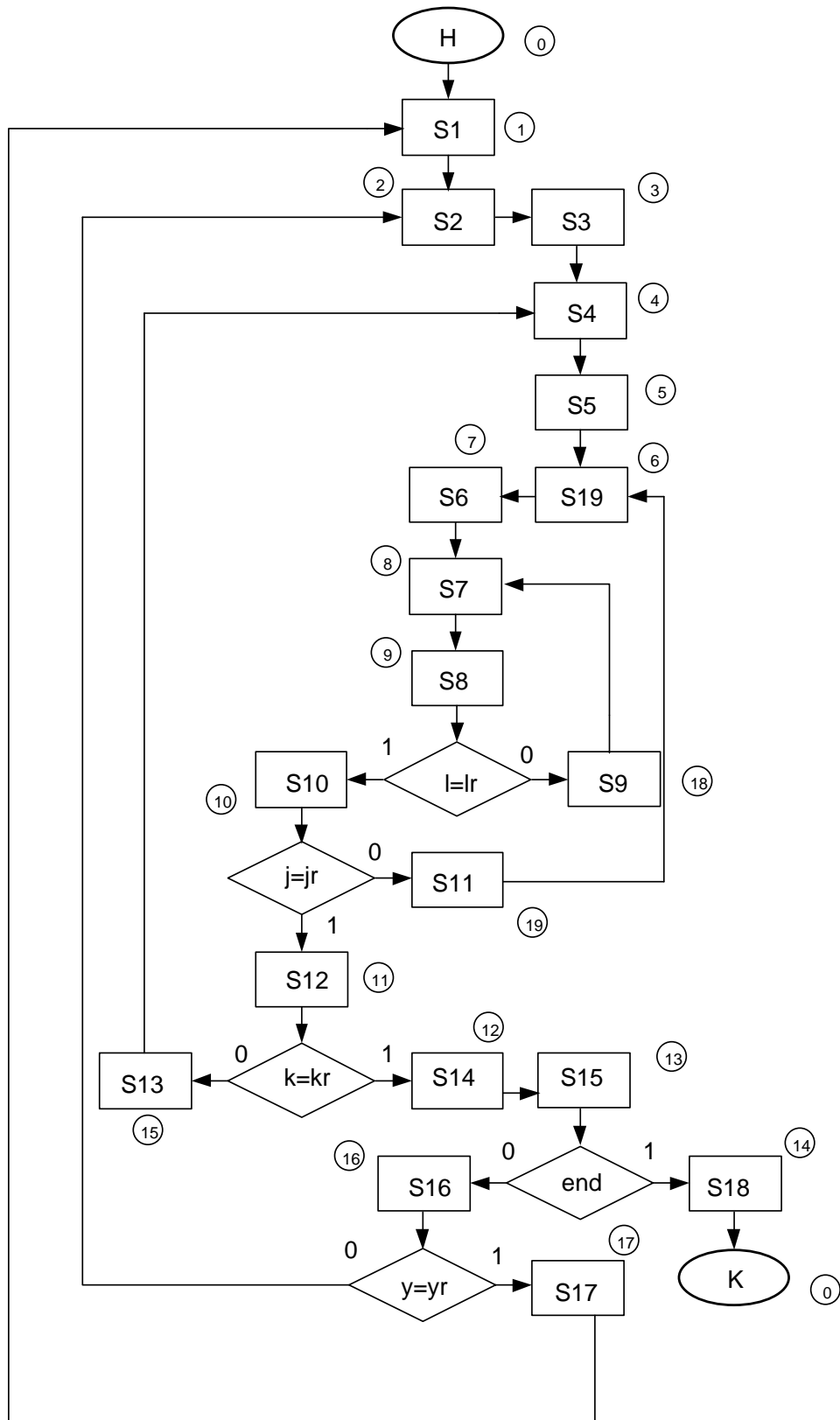


Рисунок 5.9 –Общий алгоритм управления спецпроцессором

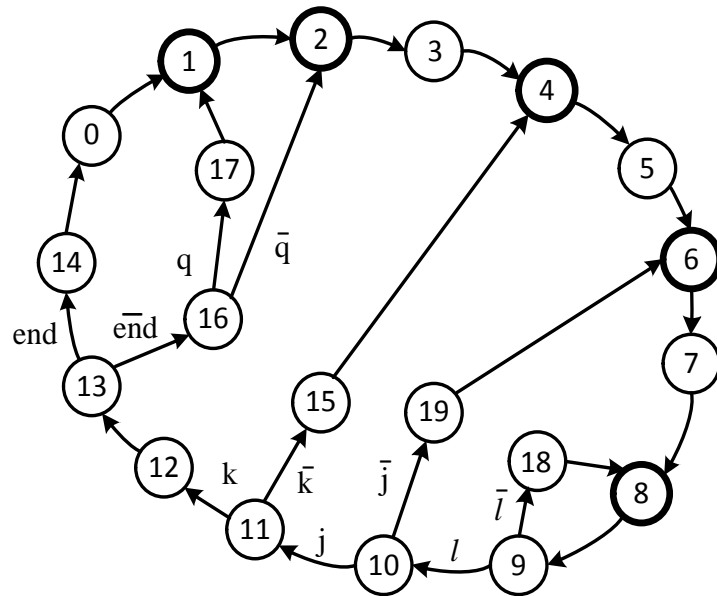


Рисунок 5.10. Граф переходов УА верхнего уровня управления спецпроцессором

Таблица 5.6. Операторы действия (семантика)

S	Содержание	S	Содержание
1	Пр «3»	11	$j = j + 1$
2	ПР «0»	12	пустой
3	Нач.уст.	13	$k = k + 1$
4	ПР «1»	14	Опрос ПК
5	$r = 0$	15	Запись X
6	$l = 0$	16	Пустой
7	$i = 0$	17	$q = q + 1$
8	ПР «4»	18	Пустой
9	Сдвиг СТП 1,2 (6, 7, 8)	19	ПР «2»
10	Пустой	0	Конец

Общий алгоритм управления СКЭН разработан без учета процедуры начальной установки исходного состояния УА. В процедуре используются 5 операторов А1-А5 с проверкой трех логических условий:

A1 – процедура регистрации и записи (НИ) в ОЗУ(3);

$\alpha_1$  -1-процедура завершена;

A2 –установка исходного состояния СП КЭС «0»  $\rightarrow$  PГP(t) -13, PГЗ(t-1) -23, PГX-22, PГY -20, RS-17, RS -28, RS-33, Сч(i) -21, Сч(j) -19, Сч А-25;

A3 –запись N эталона в СчЭ;

A4 - процедура переписи (ЭИ) в ПЗУЭ

$\alpha_2$  -1-перепись закончена;

A5 – (+1 к <СчА >ПЗУЭ);

$\alpha_3$  - <RS28> =1;

A6 - установка «0» RS28;  $\alpha_4 = 1$  есть необходимость менять эталон.

Таблица 5.7. Выбор логических условий

$\alpha$	a(t)	содержание
1	4, 17, 23	$r = r_{\max}$
2	9	$p = p_{\max}$
3	36	$l = l_{\max}$
4	38	$k = k_{\max}$
5	40	end
6	42	$\alpha_q$
7	37	$j = j_{\max}$
8	31	$i = i_{\max}$
9	0-52 кроме $\alpha_0$ a(t) по пунктам 1-8	

Структурная организация корреляционно-экстремального спецпроцессора, полученная по предложенной методике последовательной детализации  $\Phi \rightarrow \text{И} \rightarrow \text{Л} \rightarrow \text{А} \rightarrow \text{У}$  подсистем, не только оригинальна, но и реализуется с управляющей подсистемой высокого быстродействия при минимальных затратах оборудования по схеме автоматов нового типа.

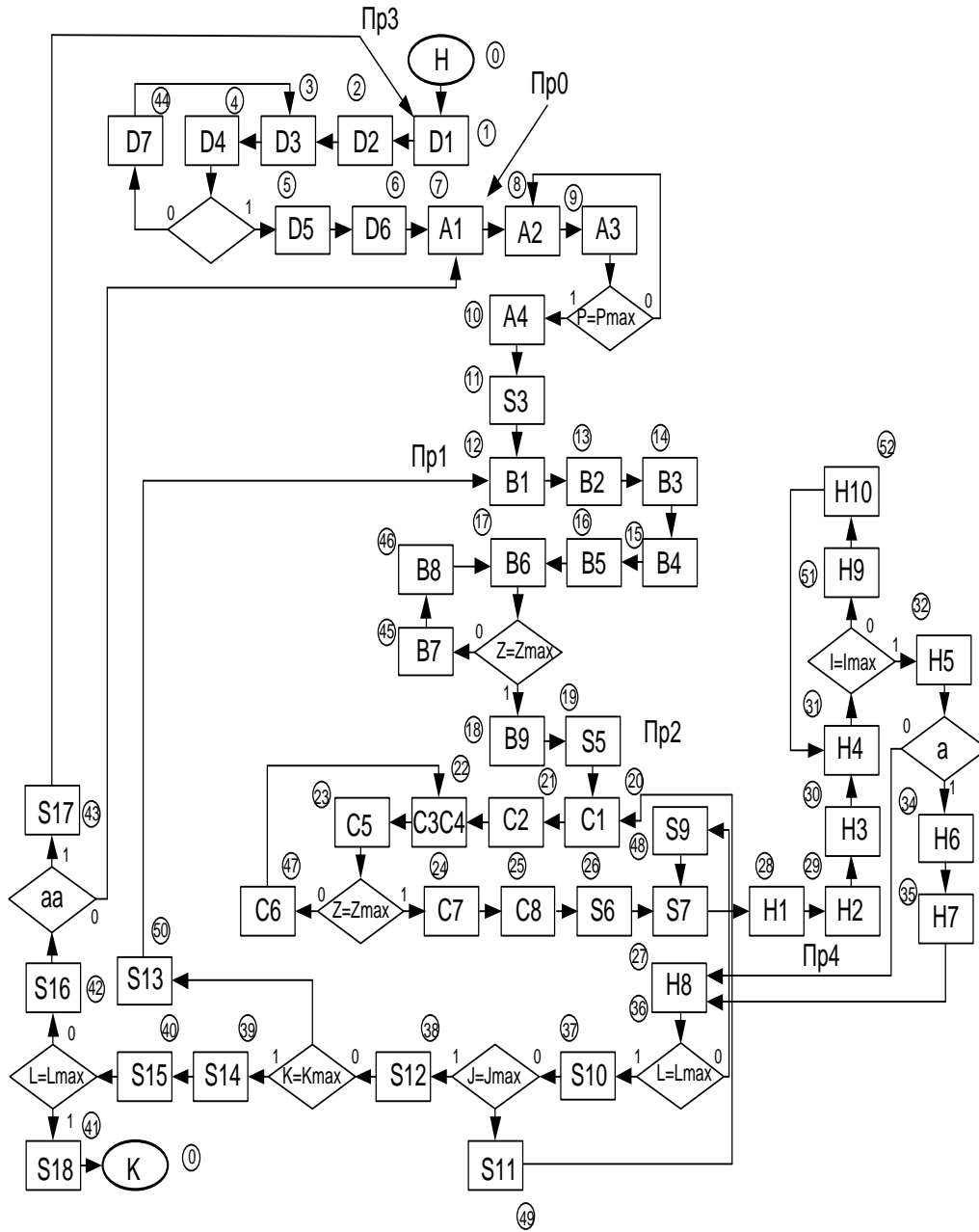


Рисунок 5.11 – Единый алгоритм управления спецпроцессором

Таблица 5.8. Таблица переходов

N	a(t)	$\alpha$	a(t+1)
1	4	$\bar{r}$	44
2	44	$\alpha_0$	3
3	9	$\bar{p}$	8
4	17	$\bar{r}$	45
5	46	$\alpha_0$	17
6	23	$\bar{r}$	47
7	47	$\alpha_0$	22
8	36	$\bar{l}$	48
9	48	$\alpha_0$	27
10	37	$\bar{j}$	49
11	49	$\alpha_0$	20
12	38	$\bar{L}$	50
13	50	$\alpha_0$	12
14	40	$\overline{end}$	42
15	43	$\alpha_0$	1
16	41	$\alpha_0$	0
17	42	$\overline{\alpha_q}$	7



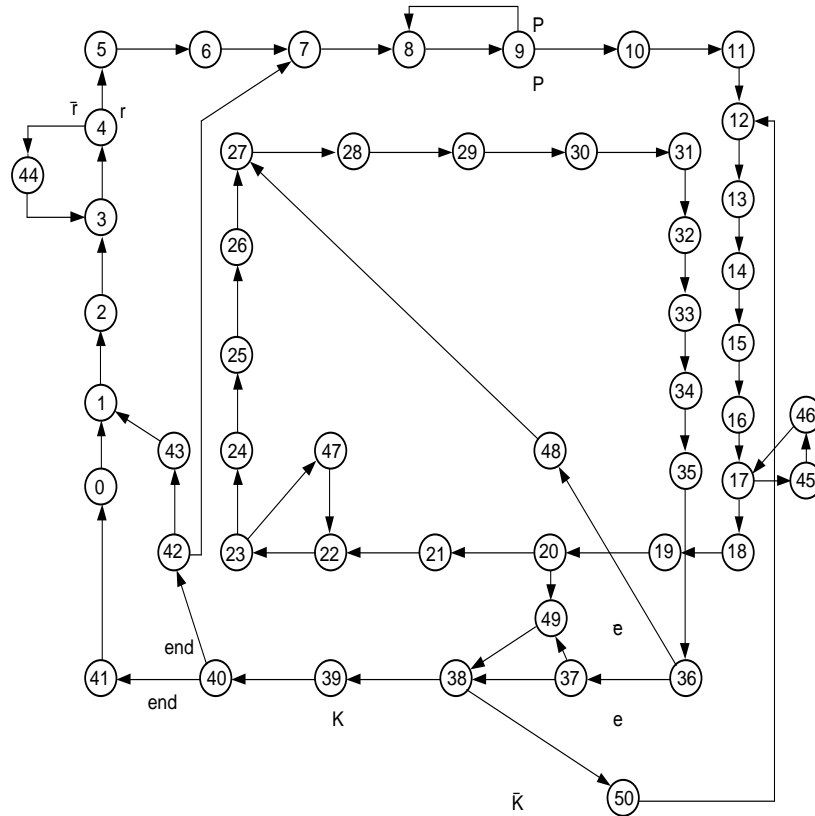


Рисунок 5.12 – Граф переходов управляющего автомата спецпроцессора

## 5.2. Спецпроцессор классификации изображений

Одной из необходимых задач первичной обработки изображений является их классификация, т.е. отнесение к одному из заданных типов (включая неизвестный тип) эталонов [166-169].

Математическая формулировка задачи изложена на основании работы [160]. В диссертации на этой основе разработана структурная схема спецпроцессора и алгоритм управления этим СПР.

В области информационно-измерительной техники для обработки радиолокационных и оптических изображений на первом этапе выделяются однородные участки на подстилающей поверхности, которые могут быть в дальнейшем использованы для опознавания образов. Каждая строка радиолокационного изображения формирователя изображения может быть

представлена дискретной последовательностью цифровых кодов элементов изображения, соответствующих яркости радиолокационного изображения облучаемой поверхности, значение каждого из которых определяется удельной эффективной поверхностью рассеяния  $W$ , соответствующего ему облучаемого участка местности. Поскольку переход от участка местности с одной удельной эффективной поверхностью рассеяния  $W_{k-1}$  к участку местности с другой удельной эффективной поверхностью рассеяния  $W_k$  происходит в случайные моменты времени, то формирователь изображения рассматривается как система со случайной структурой. Присвоим каждому из возможных значений удельной эффективной поверхности рассеяния свой номер. На основе обработки радиолокационного изображения ставим в соответствие каждому цифровому коду элемента изображения номер удельной эффективной поверхности рассеяния. Процесс смены значений удельной эффективной поверхности рассеяния во времени описывается дискретной последовательностью  $\{\theta_k, k = 0, 1, 2 \dots\}$ , которая может принимать одно из возможных значений  $i = \overline{1, M}$ .

Как в прототипе, так и в предлагаемом устройстве используются одни и те же формальные преобразования изображений.

Плотность распределения вероятностей  $W(V)$  цифровых кодов изображения  $V$  описывается гамма-распределением:  $W(V) = \frac{\theta^{N-1} \cdot \beta^{-N} \cdot e^{-\frac{\theta}{\beta}}}{\Gamma(N)}$ , где  $\Gamma(N)$  – гамма-функция,  $\Gamma(N) = (N-1)!$ ,  $N = 1, 2, \dots$ ;  $\beta = N-1W_0$  – величина, определяемая удельной эффективной поверхностью рассеяния местности  $W_0$ ;  $N$  – количество некогерентных накоплений изображения.

Вследствие мультипликативного характера спеклового шума, свойственного радиолокационному изображению, величина  $\theta$  представляется следующим образом:  $\theta = x \cdot n$ , где  $x$  – полезная составляющая, определяемая удельной эффективной поверхностью рассеяния местности  $W_0$ ;  $n$  – составляющая, обусловленная наличием спеклового шума радиолокационного изображения.

Путем логарифмического преобразования величины  $\theta$  ( $D = -\ln\theta$ ) можно добиться его представления в следующем виде  $\ln\theta = nx + \ln n$ , где  $\ln n$  - аддитивная помеховая составляющая, плотность распределения которой можно аппроксимировать гауссовским распределением, имеющим нулевое математическое ожидание и дисперсию  $= 1/N$ .

Полезная составляющая  $\ln x$  принимает конкретное значение, соответствующее величине удельной эффективной поверхности рассеяния.

Решение о принадлежности элемента изображения к участку местности с данной удельной эффективной поверхностью рассеяния принимается на основе решения задачи оптимальной фильтрации для системы со случайной структурой по критерию максимума апостериорной вероятности  $P_k(\theta_k)$ , определяемой следующим образом :

$$P_k(\theta_k) = \frac{e^{-\frac{(\ln V_k - \ln V_{\sigma_0, j})^2 N}{2}} \sum_{i=1}^p \pi(\theta_k / \theta_{k-1} = i) P_{k-1}(\theta_{k-1} = i)}{\sum_{j=1}^p e^{-\frac{(\ln V_k - \ln V_{\sigma_0, j})^2 N}{2}} \sum_{i=1}^p \pi(\theta_k / \theta_{k-1} = i) P_{k-1}(\theta_{k-1} = i)},$$

где  $V_k$  – цифровой код элемента изображения в  $k$ -ый момент времени ( $k = 0, 1, \dots$ );

$V_{\sigma_0, j}$  – цифровой код, соответствующий заданному значению  $j$ -ой удельной эффективной поверхности рассеяния ( $(j = \overline{1, p})$ );

$\pi(\theta_k / \theta_{k-1})$  – переходная вероятность номеров удельной эффективной поверхности рассеяния, определяемая свойствами облучаемого участка местности;

$P_{k-1}(\theta_{k-1})$  - апостериорная вероятность принадлежности элемента изображения к участку местности  $i$ -ой удельной эффективной поверхностью рассеяния в  $(k - 1)$  момент времени;

$N$  - количество некогерентных накоплений радиолокационного изображения.

При облучении подстилающей поверхности радиолокационной станцией бокового обзора с выхода формирователя изображения цифровой код элемента изображения  $V_k = \overline{1,255}$  (при восьмиразрядном цифровом коде), поступает на вход постоянного запоминающего устройства каждого из  $p$  каналов. В постоянном запоминающем устройстве каждого из  $p$  каналов хранятся значения

$$\exp[-\ln\theta_k - \ln\theta_j]^2 N/2]$$

Имеющееся изображение  $\lambda_{l,f}$ , задано на двумерной конечной сетке  $\{(l,f), l=1,\dots,N, f=0,1,\dots\}$  в виде набора однородных участков (ОУ), т.е. участков с постоянным значением средней удельной эффективной поверхностью рассеяния (СУЭПР). Случайные величины  $\{\lambda_k, \theta_k\}$  образуют смешанную марковскую последовательность с известными начальными  $p(\lambda_0, \theta_0)$  и переходными  $p(\lambda_{k+1}, \theta_{k+1}/\lambda_k, \theta_k)$  плотностями вероятности, заданными через условные плотности вероятности и вероятности  $\pi(\theta_{k+1}/\theta_k), P(\theta_0)$

$$P_k(\theta_k = i) = \frac{\sum_{j=1}^M \pi(\theta_k = j / \theta_{k-1} = i) p(\lambda_k/\lambda_{k-1}, \theta_k, \theta_{k-1} = i) P_{k-1}(\theta_{k-1} = i)}{\sum_{\beta=1}^M \sum_{i=1}^M \pi(\theta_k = \beta / \theta_{k-1} = i) p(\lambda_k/\lambda_{k-1}, \theta_k = \beta, \theta_{k-1} = i) P_{k-1}(\theta_{k-1} = i)}$$

$$\text{С начальными условиями: } P_0(\theta_0 = i) = \frac{p(\lambda_0/\theta_0)P(\theta_0=j)}{\sum_{\beta=1}^M p(\lambda_0/\theta_0=\beta)P(\theta_0=\beta)}; \quad j = \overline{1, M}$$

В существующих устройствах содержится  $p$  параллельных каналов, в каждом из которых имеются два независимых ПЗУ, схемы сравнения, схемы экстраполяции и устройства на элементах дискретной логики.

В диссертации разработан спецпроцессор (рисунок 5.13) с оригинальной структурной организацией, отличающийся малым числом ПЗУ по сравнению с известными решениями [167]. Алгоритм управления спецпроцессором (рисунок 5.14) является основой для проектирования управляющего автомата УА(24), в структуру которого вводится мультиплексор (02) для выбора единственного логического условия  $\alpha_j \in \{\alpha_0, \alpha_1 \dots \alpha_7\}$  по коду текущего состояния автомата  $a(t)$ , определяющего адрес мультиплексора. Граф переходов представлен на рисунке 5.15, а результаты абстрактного синтеза УА отражены в таблицах 5.9 – 5.14.

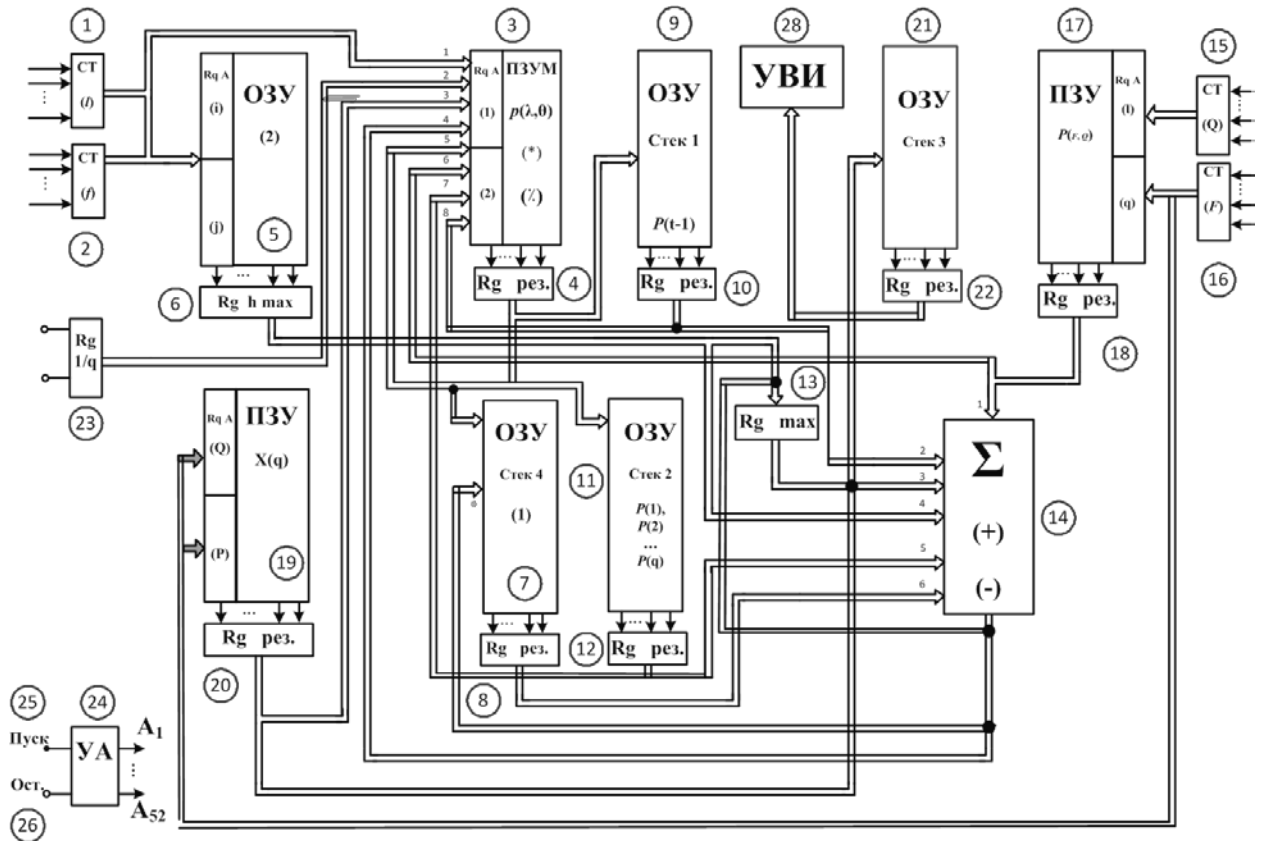


Рисунок 5.13 - Спецпроцессор дешифрирования радиолокационных изображений

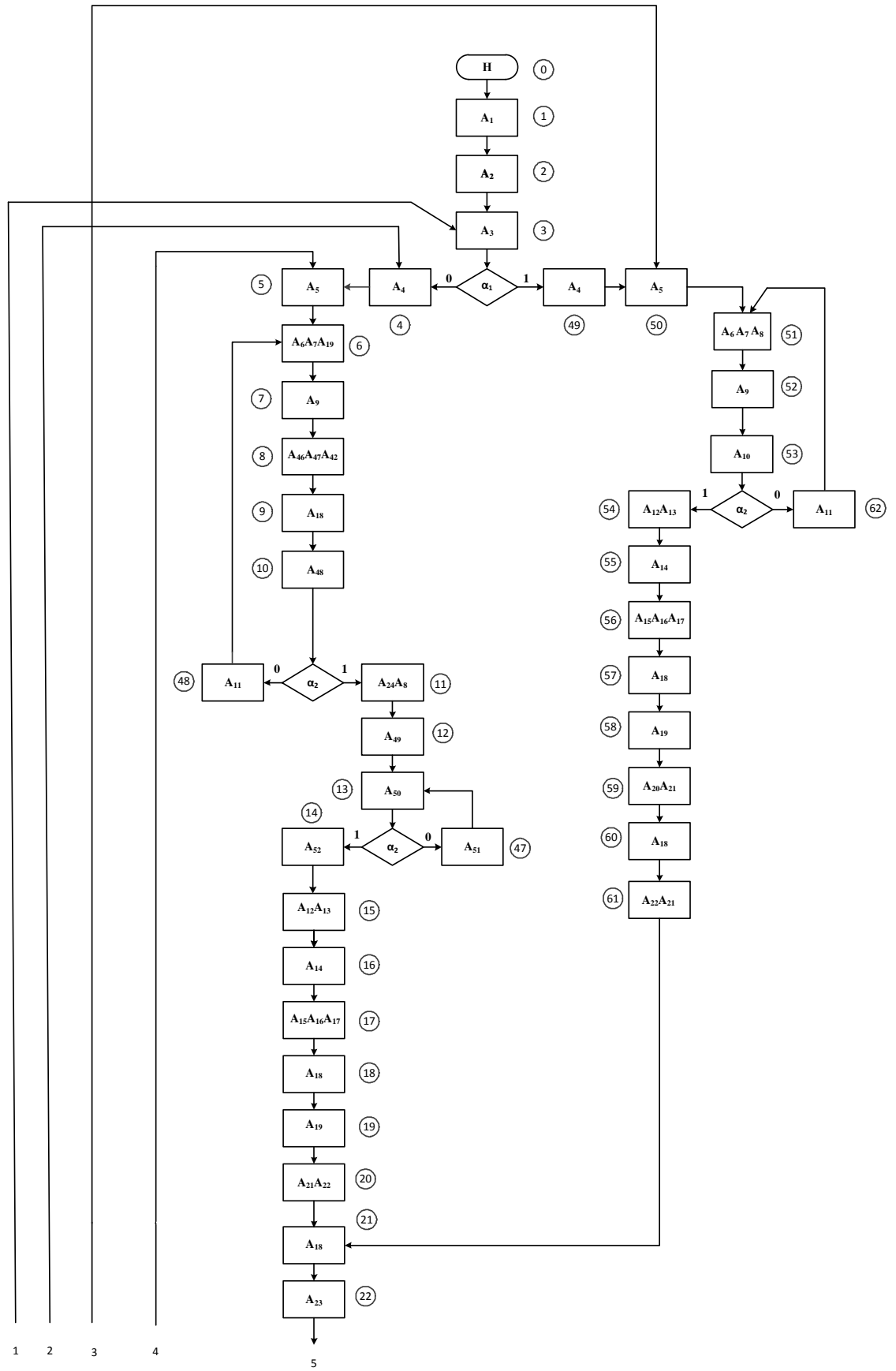


Рисунок 5.14 - Алгоритм управления спецпроцессором

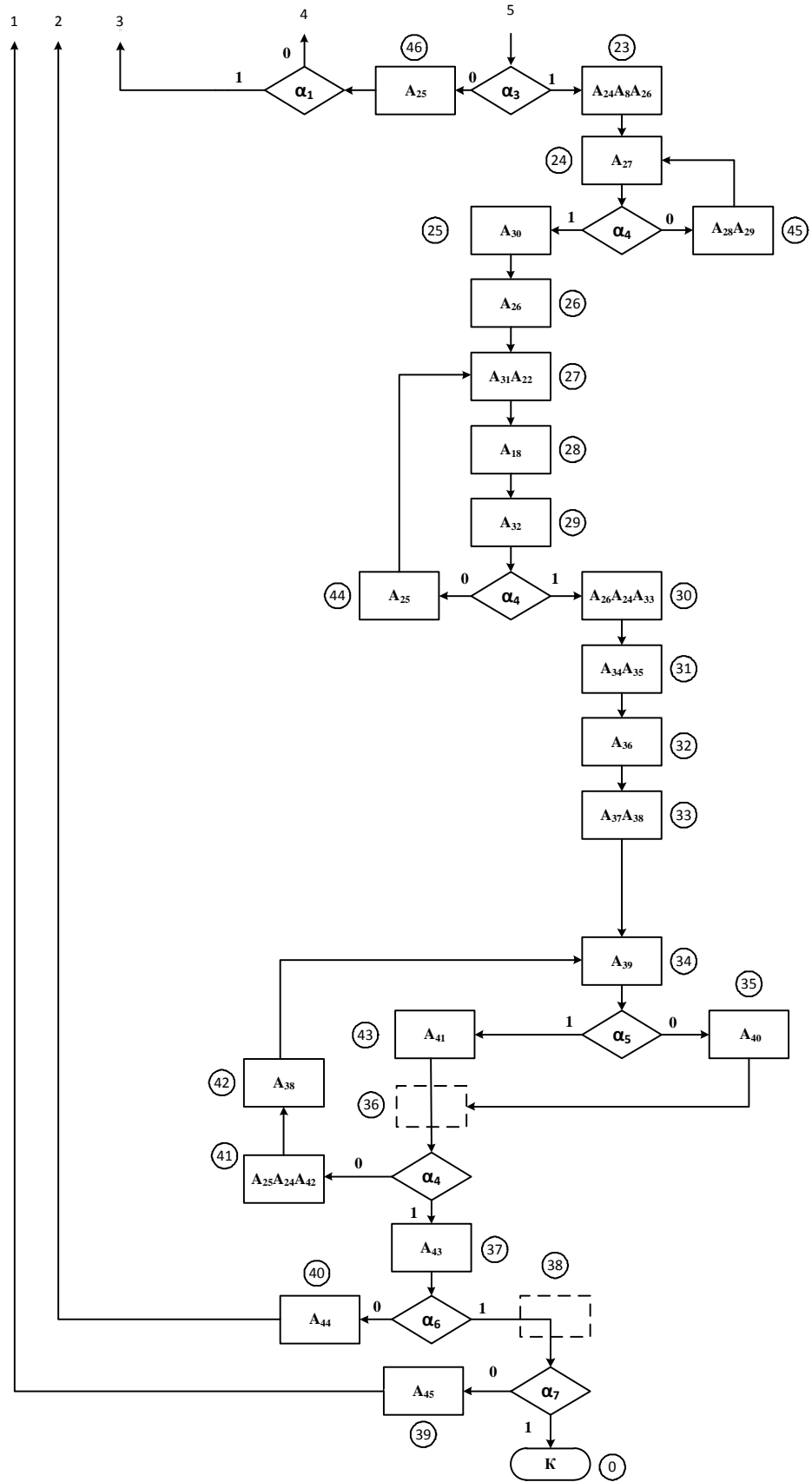


Рисунок 5.14 - продолжение

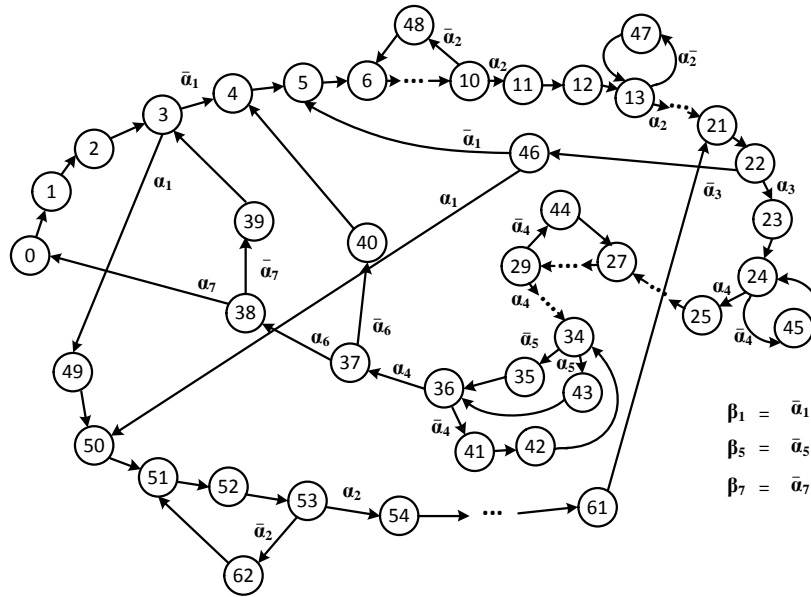


Рисунок 5.15 - Граф переходов управления спецпроцессором

Таблица 5.9. Таблица переходов

№	a(t)	x <sub>6</sub> x <sub>5</sub> x <sub>4</sub> x <sub>3</sub> x <sub>2</sub> x <sub>1</sub>	α	a(t+1)	y <sub>6</sub> y <sub>5</sub> y <sub>4</sub> y <sub>3</sub> x <sub>2</sub> y <sub>1</sub>
1	3	000011	$\bar{\beta}_1$	49	110001
2	10	001010	$\bar{\alpha}_2$	48	110000
3	48	110000	α <sub>0</sub>	6	000110
4	13	001101	$\bar{\alpha}_2$	47	101111
5	47	101111	α <sub>0</sub>	13	001101
6	22	010100	$\bar{\alpha}_3$	46	101110
7	46	101110	$\bar{\alpha}_1$	5	000101
8	46	101110	α <sub>1</sub>	50	110010
9	34	100010	$\bar{\beta}_5$	43	101011
10	43	101011	α <sub>0</sub>	36	100100
11	36	100100	$\bar{\alpha}_4$	41	101001
12	41	101001	α <sub>0</sub>	42	101010
13	42	101010	α <sub>0</sub>	34	100010
14	53	110101	$\bar{\alpha}_2$	62	111110
15	62	111110	α <sub>0</sub>	51	110011
16	61	111101	α <sub>0</sub>	21	010101
17	38	100110	α <sub>7</sub>	0	000000
18	39	100111	α <sub>0</sub>	3	000011
19	40	101000	α <sub>0</sub>	4	000100
20	29	011101	$\bar{\alpha}_4$	44	101100
21	44	101100	α <sub>0</sub>	27	011011
22	45	101101	α <sub>0</sub>	24	011000



23	24	011000	$\overline{\alpha_4}$	45	101101
----	----	--------	-----------------------	----	--------

Таблица 5.10. Выбор логических условий

$\alpha$	$Z_3 Z_2 Z_1$	a(t)
1	001	3, 46
2	010	10, 49,13
3	011	-
4	100	24, 29, 36
5	101	34
6	110	37
7	111	38
0	000	0-2, 4-9, 11,12,14-21, 23,26-28, 30-33, 35,39,40,41,42,44,45,47,48, 49-52,54-61,62

$\gamma \rightarrow \text{Ст} (0 - 38; 49 - 60)$

$\alpha_0 - (0 - 2; 4 - 9; 11, 12, 14 - 21; 23, 25 - 28; 30 - 33; 35, 39, 41, 42, 43, 44, 45, 47, 48, 49 - 52; 54 - 61; 62)$

Таблица 5.11. Логические условия (семантика)

$\alpha_1$	$f = 0$	$\alpha_5$	Знак $\Sigma$
$\alpha_2$	$Q = q$	$\alpha_6$	$f = m$
$\alpha_3$	$P = q$	$\alpha_7$	$l = n$
$\alpha_4$	$h = q$	$\alpha_0$	Безусловный переход

$\gamma = 1$ , если есть необходимость (+1) к <Ст>

$f_{\text{сч}} = \alpha(a_0 + \dots + a_{39} + a_{49} + \dots + a_{61})$

$f_{\alpha} = (\alpha_0 + \alpha_1) (48 + 47 + 46 + 43 + 41 + 42 + 62 + 61)$

Таблица 5.12. Выходные команды

a(t)	A <sub>i</sub>	a(t)	A <sub>i</sub>	a(t)	A <sub>i</sub>	a(t)	A <sub>i</sub>
0	-	17	A <sub>15</sub> A <sub>16</sub> A <sub>17</sub>	34	A <sub>39</sub>	51	A <sub>6</sub> A <sub>7</sub> A <sub>8</sub>
1	A <sub>1</sub>	18	A <sub>18</sub>	35	A <sub>40</sub>	52	A <sub>9</sub>
2	A <sub>2</sub>	19	A <sub>19</sub>	36	-	53	A <sub>10</sub>
3	A <sub>3</sub>	20	A <sub>21</sub> A <sub>22</sub>	37	A <sub>43</sub>	54	A <sub>12</sub> A <sub>13</sub>
4	A <sub>4</sub>	21	A <sub>18</sub>	38	-	55	A <sub>14</sub>
5	A <sub>5</sub>	22	A <sub>23</sub>	39	A <sub>45</sub>	56	A <sub>15</sub> A <sub>16</sub> A <sub>17</sub>
6	A <sub>6</sub> A <sub>7</sub> A <sub>19</sub>	23	A <sub>24</sub> A <sub>8</sub> A <sub>26</sub>	40	A <sub>44</sub>	57	A <sub>18</sub>
7	A <sub>9</sub>	24	A <sub>27</sub>	41	A <sub>25</sub> A <sub>24</sub> A <sub>42</sub>	58	A <sub>19</sub>
8	A <sub>46</sub> A <sub>47</sub> A <sub>42</sub>	25	A <sub>30</sub>	42	A <sub>38</sub>	59	A <sub>20</sub> A <sub>21</sub>
9	A <sub>18</sub>	26	A <sub>26</sub>	43	A <sub>41</sub>	60	A <sub>18</sub>
10	A <sub>48</sub>	27	A <sub>31</sub> A <sub>22</sub>	44	A <sub>25</sub>	61	A <sub>22</sub> A <sub>21</sub>
11	A <sub>24</sub> A <sub>8</sub>	28	A <sub>18</sub>	45	A <sub>28</sub> A <sub>29</sub>	62	A <sub>11</sub>
12	A <sub>49</sub>	29	A <sub>32</sub>	46	A <sub>53</sub>		
13	A <sub>50</sub>	30	A <sub>26</sub> A <sub>24</sub> A <sub>33</sub>	47	A <sub>51</sub>		
14	A <sub>52</sub>	31	A <sub>34</sub> A <sub>35</sub>	48	A <sub>53</sub>		
15	A <sub>12</sub> A <sub>13</sub>	32	A <sub>36</sub>	49	A <sub>4</sub>		
16	A <sub>14</sub>	33	A <sub>37</sub> A <sub>38</sub>	50	A <sub>5</sub>		

Таблица 5.13. Таблица повторяющихся команд.

A <sub>i</sub>	4	5	6	7	12	13	14	18	19	21	22	24
a(t)	4	5	6	6	15	15	16	9	6	20	20	4
	49	50	51	51	54	54	55	57	58	59	59	23

Таблица 5.14. Команды (семантика)

A <sub>1</sub>	Уст. исход. сост. СП	A <sub>28</sub>	$h = h + 1$
A <sub>2</sub>	$l = 0$	A <sub>29</sub>	Сдвиг стека 1
A <sub>3</sub>	$f = 0$	A <sub>30</sub>	Выбор кода (/) в ПЗУМ «1.1»
A <sub>4</sub>	$F = 0$	A <sub>31</sub>	Счит. <стек 1> → Rg А ПЗУМ
A <sub>5</sub>	$Q = 0$	A <sub>32</sub>	Зап. < Rg рез.ПЗУМ> в стек 2
A <sub>6</sub>	$\langle F \rangle \rightarrow \text{Rg А ПЗУ } (P_F, Q) \text{ (1)}$	A <sub>33</sub>	$-\Sigma$
A <sub>7</sub>	$\langle Q \rangle \rightarrow \text{Rg А ПЗУ } (P_F, Q) \text{ (2)}$	A <sub>34</sub>	$\langle f \rangle \rightarrow \text{ОЗУ (5)}$
A <sub>8</sub>	$+\Sigma$	A <sub>35</sub>	$\langle l \rangle \rightarrow \text{ОЗУ (5)}$
A <sub>9</sub>	Счит. ПЗУ $(P_F, Q) \rightarrow \text{Rg рез. ПЗУ } (P_F, Q)$	A <sub>36</sub>	Счит. ОЗУ (5) → Rg h max
A <sub>10</sub>	$\text{Rg рез. ПЗУ } (P_F, Q) \rightarrow \Sigma$	A <sub>37</sub>	$\langle h \text{ max} \rangle \rightarrow \text{Rg МАХ}$
A <sub>11</sub>	$Q = Q + 1$	A <sub>38</sub>	Запись <стек 2> → $\Sigma$
A <sub>12</sub>	$\langle F \rangle \rightarrow \text{Rg А ПЗУ } X(q)$	A <sub>39</sub>	Запись < Rg МАХ > → $\Sigma$
A <sub>13</sub>	Выбор кода $p(\lambda, \theta)$ ПЗУМ «1.0»	A <sub>40</sub>	Запись Rg2 $\Sigma \rightarrow \text{Rg МАХ}$
A <sub>14</sub>	Зап. ПЗУ $X(q) \rightarrow \text{Rg рез. ПЗУ } X(q)$	A <sub>41</sub>	Запись Rg1 $\Sigma \rightarrow \text{Rg МАХ}$
A <sub>15</sub>	$\text{Rg рез. ПЗУ } X(q) \rightarrow \text{Rg А ПЗУМ}$	A <sub>42</sub>	Сдвиг стека 2
A <sub>16</sub>	$\langle l \rangle \rightarrow \text{Rg А ПЗУМ}$	A <sub>43</sub>	Зап. < Rg МАХ > → ОЗУ Стек 3
A <sub>17</sub>	$\langle f \rangle \rightarrow \text{Rg А ПЗУМ}$	A <sub>44</sub>	$f = f + 1$
A <sub>18</sub>	$\langle \text{ПЗУМ} \rangle \rightarrow \text{Rg рез. ПЗУМ}$	A <sub>45</sub>	$l = l + 1$
A <sub>19</sub>	Выбор кода (*) в ПЗУМ «0.1»	A <sub>46</sub>	$\langle \text{Rg рез. ПЗУ } (P_F, Q) \rangle \rightarrow \text{Rg А ПЗУМ}$
A <sub>20</sub>	Запись $\langle l/q \rangle \rightarrow \text{Rg А ПЗУМ}$	A <sub>47</sub>	$\langle \text{Rg рез. ОЗУ Стек 2} \rangle \rightarrow \text{Rg А ПЗУМ}$
A <sub>21</sub>	$\langle \text{Rg рез. ПЗУМ} \rangle \rightarrow \text{Rg А ПЗУМ}$	A <sub>48</sub>	$\langle \text{Rg рез. ПЗУМ} \rangle \rightarrow \text{ОЗУ Стек 4}$
A <sub>22</sub>	$\langle \Sigma \rangle \rightarrow \text{Rg А ПЗУМ}$	A <sub>49</sub>	ОЗУ Стек 4 → Rg рез. ОЗУ Стек 4
A <sub>23</sub>	Запись < Rg рез. ПЗУМ > в стек 1	A <sub>50</sub>	Rg рез. ОЗУ Стек 4 → $\Sigma$
A <sub>24</sub>	Обнуление $\Sigma$	A <sub>51</sub>	Сдвиг стека 4
A <sub>25</sub>	$F = F + 1$	A <sub>52</sub>	$\langle \Sigma \rangle \rightarrow \text{ОЗУ Стек 4}$
A <sub>26</sub>	$h = 0$		
A <sub>27</sub>	$\langle \text{стек 1} \rangle \rightarrow \Sigma$		

где  $l$  – номер строки;

$f$  - номер столбца;

$h$  – промежуточный индекс ( $h=1,2,\dots,q$ );

$F$  – текущий индекс класса внешнего цикла перехода из класса  $F$  в класс  $Q$  ;

$Q$  – текущий индекс класса внутреннего цикла;

$q$  – максимальное число классов;

$m$  – число столбцов в матрице ( $m \times n$ );

$n$  – число строк в матрице ( $m \times n$ );

$P_k, P_{k-1}$  – настоящее и предыдущее значение вероятности  $P$  для класса  $F_k$ .

Преимущества СПР очевидны. Для восьми различных типов участков подстилающей поверхности, в прототипе потребуется 16 ПЗУ, а для 16 необходимо 32 ПЗУ. Тогда как в СПР, независимо от количества участков, всегда используется три ПЗУ.

### 5.3. Спецпроцессор управления нефтеперекачивающей станцией

Оснащение нефтеперекачивающих станций современными микропроцессорными автоматизированными системами существенно снижает нагрузки на трубопроводы при остановках магистральных насосных агрегатов. Система управления построена по многоуровневому принципу. На нижних уровнях основными объектами управления являются нефтеперекачивающие станции. Важнейшими технологическими задачами являются задачи управления мощными насосами и задачи сглаживания волн давления при аварийных и других типах пуска и останова станций [170].

Для управления двумя мощными насосами перекачивающей станции (ПС), в эксплуатируемых на сегодня установках, используется сложная техническая система из информационного процессора и двух восьмиразрядных управляющих контроллеров. В операционных устройствах используются специальные коммутаторы и инверторы на транзисторах IGBT для управления мощными электроприводами. Управление насосами реализуется на микроконтроллере с

использованием операторов, приведенных в таблице 5.15. Этот уровень управления в диссертации не разрабатывался. При переключениях и аварийных остановках для сглаживания волн давления разработан специальный алгоритм.

Оснащение нефтеперекачивающих станций современными микропроцессорными системами существенно снижает нагрузки на трубопроводы при остановках магистральных насосных агрегатов. Скорость роста давления на входе перекачивающей станции (ПС) определяется временем остановки насосных агрегатов и достигает 0.1-0.3 МПа/с. При такой высокой скорости роста давления образуется крутая волна повышенного давления, распространяющаяся со скоростью звука в направлении предшествующей станции. К моменту прихода крутой волны к расположенной выше по потоку нефти насосной станции на большей части длины линейного участка трубопровода давление успевает достигнуть максимального уровня. Сглаживание волны давления достигается за счет частичного сброса нефти в сбросной резервуар через соответствующие клапаны.

Микропроцессорные системы автоматизированного управления имеют трёхуровневую организацию. Верхний уровень системы автоматизации включает в себя АРМ оператора и служит для мониторинга и оперативного управления технологическими процессами. Алгоритмы среднего уровня обеспечивают:

- закрытие задвижек, отсекающих аппаратуру при пожаре, аварийной и длительной загазованности и отключение насосов при достижении аварийного максимального уровня в емкости;
- автоматическое включение насосов откачки при максимальном уровне нефти в емкости и их автоматическое отключение при достижении минимального уровня;
- автоматический ввод резерва насосов откачки и вытяжных вентиляторов.

В диссертации предложен ряд новых процедур управления:

- 1) Автоматический пуск насосов откачки утечек при поступлении сигнала максимального уровня в емкости.

2) При сохранении сигнала максимального уровня в течение заданного времени производится автоматический пуск резервного насоса откачки и автоматическое открытие дополнительной задвижки на линии откачки.

3) При поступлении сигнала минимального уровня в систему управления производится автоматическое отключение насосов и закрытие дополнительной задвижки откачки. Кроме того, с целью исключения негативных причин нерегулируемого сброса нефти и повышения надежности работы предлагаются следующие доработки алгоритмов:

4) Если не произошло включения резервного насоса, команда на закрытие задвижек подключения к нефтепроводу формируется без выдержки времени  $t_2$ .

5) Ввести значение давления «максимальное 1», при достижении которого происходит одновременное включение основного и резервного насосов с открытием дополнительной задвижки линии откачки. Значение «максимальное 1» определяется характеристиками насосов откачки за определенное время. Значение давления, при котором блокируется запуск насосов, задекларировать как «максимальное 2».

6) Если на момент поступления сигнала максимального уровня давление на приеме не превышает значения «максимальное 1», включается основной насос откачки, при сохранении сигнала в течение заданного времени  $t_1$  включается резервный насос и открывается дополнительная задвижка линии откачки.

7) Если при работе двух насосов в течение времени  $t_2$  сигнал максимального уровня сохраняется, формируется команда на закрытие задвижек подключения к нефтепроводу, при этом необходимо предусмотреть возможность изменения значения  $t_2$  с верхнего уровня.

8) Если на момент поступления сигнала максимального уровня и давления на приеме, не превышающем значение «максимальное 1», не включился основной насос откачки, то включается резервный насос. При сохранении сигнала

максимального уровня в течение заданного времени  $t_1$  формируется команда на закрытие задвижек подключения к нефтепроводу.

9) Если на момент поступления сигнала максимального уровня давление на приеме превышает значение «максимальное 1», а включился только один насос откачки, то при сохранении сигнала максимального уровня в течение заданного времени  $t_1$  формируется команда на закрытие задвижек подключения к нефтепроводу. Если не включились оба насоса, команда на закрытие задвижек подключения к нефтепроводу формируется без выдержки времени  $t_2$ .

10) Если на момент поступления сигнала максимального уровня давление на приеме превышает значение «максимальное 2» (при котором пуск насосов заблокирован), команда на закрытие задвижек подключения к нефтепроводу формируется без выдержки времени.

В качестве дополнительной меры повышения надежности вводятся ограничения на время работы, т.к. необходимо формирование команды на закрытие задвижек подключения к нефтепроводу независимо от наличия или отсутствия сигналов максимального уровня, поскольку превышение времени  $t_3$  свидетельствует о не герметичности клапана.

Предлагаемые доработки алгоритмов позволяют повысить надежность рассматриваемых систем автоматики до уровня более технологичных систем с четырехуровневым контролем объемов нефти в резервуарах, позволяющих избежать отключения НПС при отказах.

### 5.3.1. Алгоритм управления технологическими процессами магистральных нефтепроводов.

Для управления двумя мощными насосами перекачивающей станции (ПС), используется сложная техническая система из информационного процессора и двух восьмиразрядных управляющих контроллеров независимых операционных устройств. В операционных устройствах используются специальные коммутаторы и инверторы на транзисторах IGBT для управления мощными электроприводами.

Алгоритм управления операционными устройствами насосов доработан диссертантом, при этом он содержит 29 операторов действия, а автомат управления имеет среднюю сложность ( $m = 5$ ,  $q = 10$ ). Введение пустых операторов не привело к увеличению  $m$  и  $q$ .

Граф переходов автомата (рисунок 5.17), полученный по доработанному диссертантом алгоритму управления ПС (рисунок 5.18), содержит 37 состояний ( $m=6$ ) при числе логических условий  $q = 10$ . Переходы и выбор логических условий представлены в таблицах 5.16 и 5.17. Причем из 52 связей между состояниями менее 1/3 реализуется таблицами, а остальные относятся к счетчику. Для УА Мура  $V = 256\text{кб}$ , для новой структурной организации автомата управления  $W = 2\text{кб}$  и при использовании универсального счетчика  $W = 1\text{кб}$ .

Таблица 5.16. Таблица переходов

№	$a(t)$	$\alpha$	$a(t + 1)$	№	$a(t)$	$\alpha$	$a(t + 1)$
1	2	$\alpha_1$	24	9	16	$\alpha_6$	31
2	24	$\alpha_0$	30	10	31	$\alpha_0$	14
3	30	$\alpha_0$	1	11	18	$\alpha_7$	23
4	3	$\alpha_2$	28	12	19	$\overline{\alpha_8}$	22
5	4	$\alpha_3$	25	13	22	$\alpha_0$	0
6	25	$\alpha_0$	27	14	20	$\overline{\alpha_9}$	19
7	5	$\overline{\alpha_4}$	26	15	21	$\alpha_0$	2
8	9	$\overline{\alpha_5}$	10				



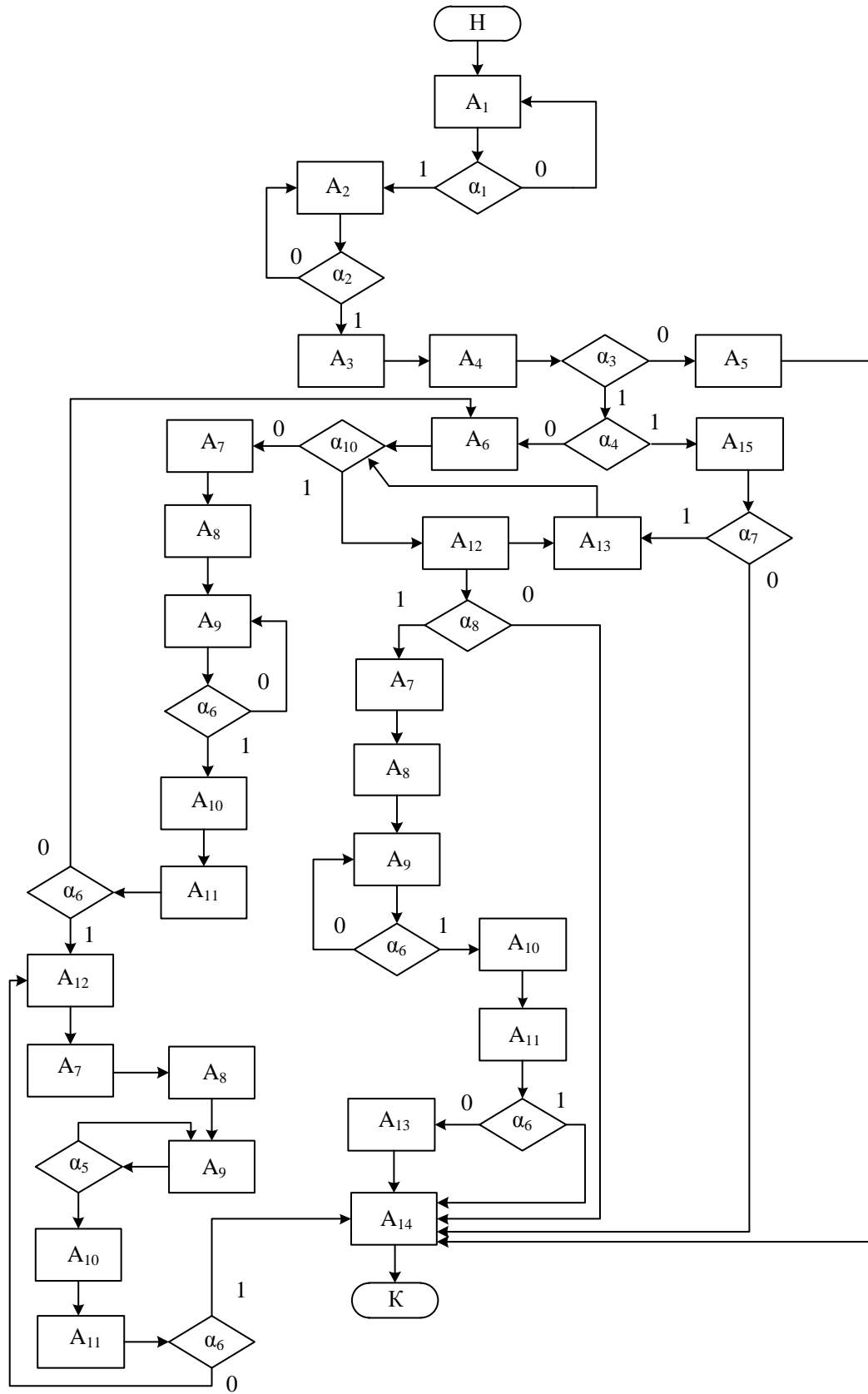


Рисунок 5.17 - Алгоритм сглаживания волн давления.

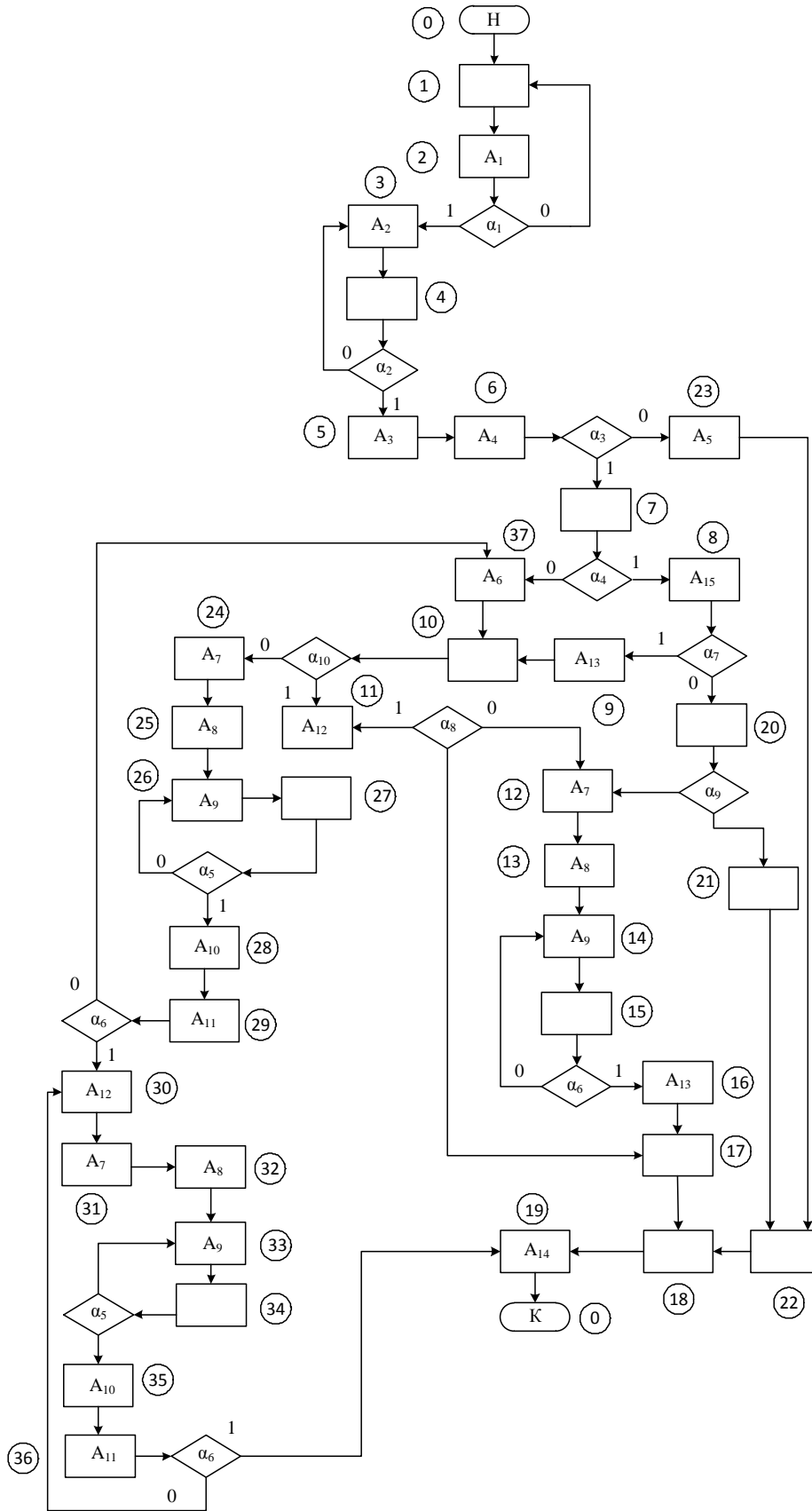
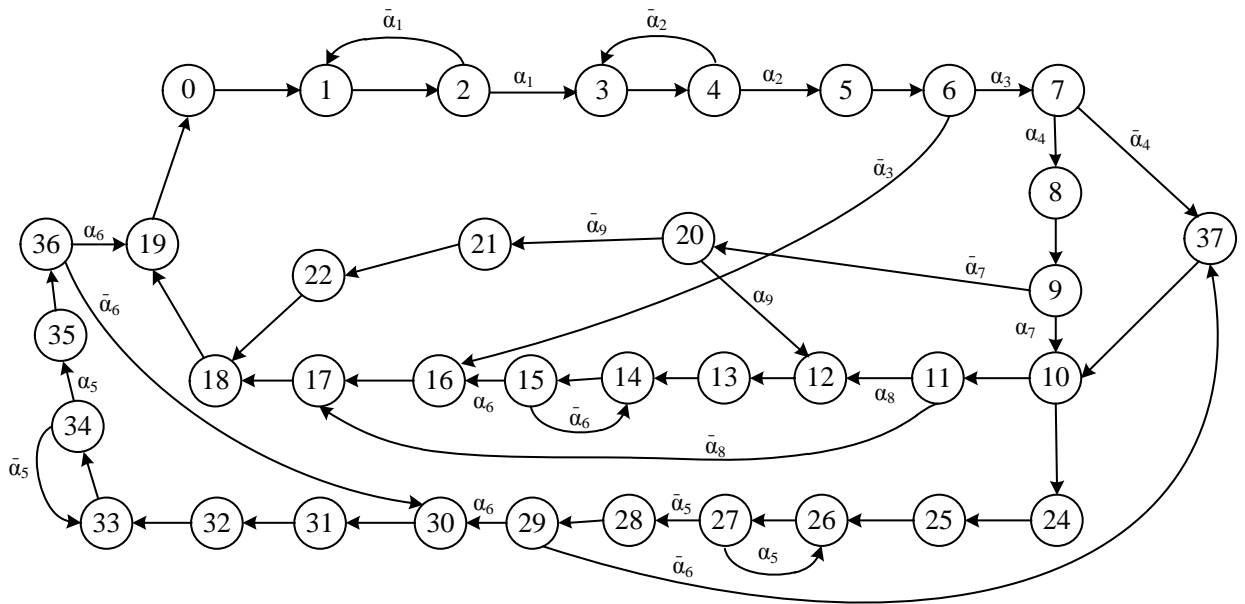


Рисунок 5.18 – Модифицированный алгоритм сглаживания волн давления.



СТ → 0 -19; 20 -21; 24 -35

Рисунок 5.19 – Граф переходов алгоритма сглаживания волн давления.

Таблица 5.17. Выбор логических условий

№	$\alpha$	Состояние	№	$\alpha$	Состояние
1	1	2	7	7	9
2	2	3	8	8	11
3	3	6	9	9	20
4	4	7	10	10	10
5	5	27, 34	11	$\beta_9$	20
6	6	29, 15, 36	12	$\beta_{10}$	10

#### 5.4. Спецпроцессор криптографической защиты информации

Известны программные системы шифрования двоичной информации на основе блочных шифров с использованием методов логического преобразования информации по закрытому ключу с последующим рассеиванием информации [171-178]. Направление использования аппаратных методов в задаче криптографической защиты информации (КЗИ) является новым и не достаточно исследовано.

Преобразование информации в большинстве случаев связано со сложением по модулю два (операция XOR) порции информации с кодом секретного ключа. После операции XOR часть сообщения подвергается перемешиванию, а другая переставляется на место первой (основа метода Фейстеля и др.). Эти процедуры повторяются 32 – 64 раза.

Интенсивное развитие компьютерных средств и информационных технологий значительно повысило уязвимость современных информационных систем и вычислительных средств.

Поэтому проблема обеспечения информационной безопасности, несмотря на разработки большого числа средств защиты, по-прежнему остается актуальной.

Для систем связи и вычислительных систем реального времени особый интерес представляют системы защиты информации с аппаратной реализацией, т.к. они имеют существенное преимущество по быстродействию, при той же криптостойкости, что и системы, реализованные на ЭВМ.

Публикации по аппаратной защите информации малочисленны и в этом направлении отсутствуют обобщающие монографии. Более того, наиболее эффективные аппаратные средства защиты имеют весьма сложную реализацию, основанную на применении параллельно работающих микропроцессорных систем с программным управлением.

Повышение быстродействия систем криптографической защиты информации для систем реального времени, возможно с помощью специализированных структурных решений.

Соискателем предложены три алгоритма защиты информации, реализованных в виде оригинальных устройств [179-181].

В спецпроцессоре используется оригинальный алгоритм перепутывания информации. Суть его в том, что секретный ключ - это комбинация из равного числа нулей и единиц, которые распределены по случайному закону. В регистрах сдвига С и D разрядностью  $n/2$  очередной бит сообщения  $X_i$  записывается по правилу  $C_i = X_i/b_i = 1$ ,  $D_i = X_i/b_i = 0$ . Далее образуется соединение С и D с выполнением операция  $\text{mod}2$ . В каждом следующем шаге производится сдвиг в регистре ключа.

На следующем этапе часть сообщения  $n$  является не только кодом для передачи информации, но и адресом для постоянного запоминающего устройства, где хранятся секретные ключи (патент №82890). Даже при  $n=16$  количество ключей больше 64000 (количество ключей равно  $2^n$ ), что говорит о невозможности взлома информации с помощью простого перебора, т.к. число вариантов перебора будет равным  $2^{16N}$ , где  $N$  - количество ключей всего закодированного текста сообщения.

В патенте №82974 устройство шифрования основано также на разбиении последовательности поступающих данных на группы по  $n$  бит ( $n=3,4,6,8$ ). Отличие заключается в том, что эта группа рассматривается как коэффициент при соответствующем разряде полиадического числа. Несколько групп последовательности из сообщения преобразуется в полиадическое число, количество весовых коэффициентов и конкретное значение весов которого является секретом.

Преобразованное таким образом сообщение  $x(t)$  переводится из двоичного в полиадический код по формуле:

$$R(x) = x_0 + x_1 L_1 + x_2 L_1 L_2 + \dots + L_1 L_2 \dots L_p x_p,$$

где  $x_1, x_2, \dots, x_n$  - численное значение кодов групп  $x_i$  ( $i = 1, 2, \dots, p$ ) преобразованного сообщения, зафиксированного на  $RgP(25)$  и разделенного на  $p$  групп.

Коэффициенты  $L_1, L_2 \dots L_p$  могут быть любыми, в том числе  $L_i = L_j$  но любое  $L_i > 2^m$ , т.к. в  $m$ - разрядах группы записано число  $(2^m - 1)$ .

Пример для четырехразрядных групп приведен в таблице 5.18. Как видно значение первой и четвертой группы взято равным 17. Число, которое может быть получено при выбранных значениях  $L_i$  равно  $R(x)$ .

$R_i$  - коэффициенты полиадической системы счисления, соответствующие каждой группе числа, т.е. 16 - разрядное число, представленное в двоичном позиционном коде, преобразуется в 20- ти разрядный двоичный код полиадического числа.

Таблица 5.18

№	4	3	2	1
$m$	4	4	4	4
$x_i$	5	3	11	1
$L_i$	17	23	21	17
$R_i$	697935	24633	3927	17

$$Q = 697935 + 24633 + 3927 + 17 = 726512;$$

Нетрудно видеть, что произведя обратные преобразования делением на  $R_4$ , затем остатка на  $R_3$  и т.д. до  $R_1$  получим коэффициенты 5,3,11,1 соответствующие значениям  $x_i$ . В примере константа  $a_0$  взята равной «0».

Все блоки и логические элементы предлагаемого изобретения – типовые, не имеют специфических особенностей, кроме блока групповых схем И (26) и блока синхронизации БС (11).

В патенте №82889 устройство по этому способу формирует  $K$  групп по  $n$  бит, где  $n$  – длина обрабатываемого текста. Причем каждая из групп обрабатывается в зависимости от значения  $n$ -бит секретного ключа, с перемешиванием поточного шифра по случайному закону. Поточный шифр суммируют по модулю 2 с порцией сообщения. Устройство содержит регистр

секретного ключа, блок управления, четыре последовательно-параллельных регистра, регистр показателей степени, блок перемешивания бит, логический блок с сумматором по модулю 2, элементы «И», «ИЛИ» и др.

В патенте № 2475838. БИ №5, 2013г. реализован спецпроцессор криптографической защиты информации, основанный на оригинальном способе комплексирования в единой структуре алгоритмов по патентам №82974; 82889; 82890; 63588; 82888 с использованием иерархической системы управления. Структурная схема операционной части спецпроцессора приведена на рисунке 5.19, с расшифровкой микроопераций в таблице 5.19. Алгоритм управления спецпроцессором представлен на рисунке 5.20.

На ГСА рисунка 5.20 множество  $\{\alpha\}$  представлено в виде  $\beta, m, k, n, e$ , где  $\beta$  – значение «1» или «0» на выходе RgCдв3 (10);  $m$  – разрядность информационного слова;  $k$  – количество групп;  $n$  – разрядность выходного слова;  $e$  – признак окончания программы (end).

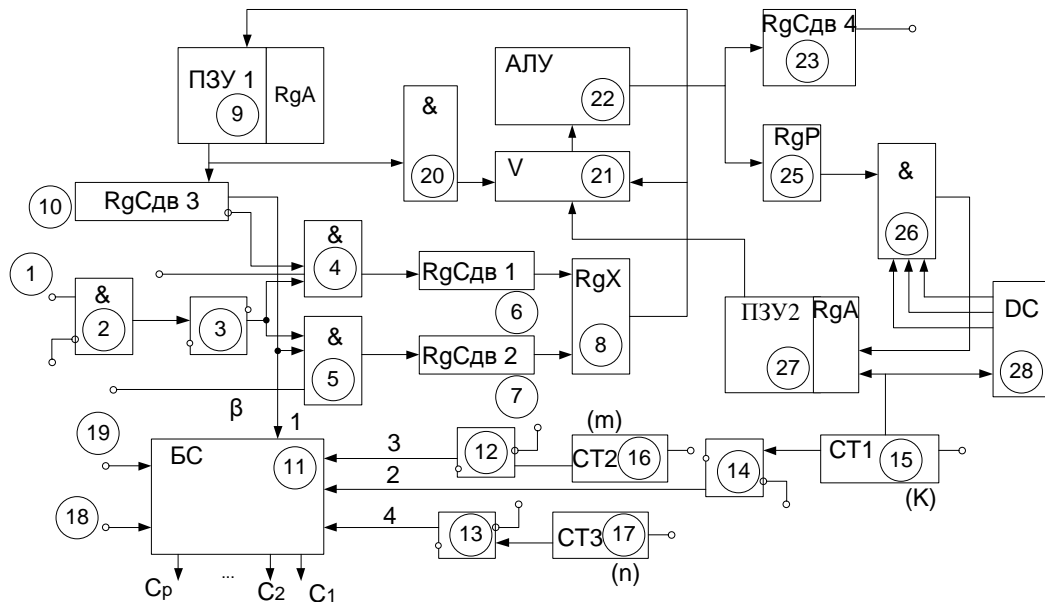


Рисунок 5.19 - Структурная схема устройства криптографической защиты информации

Таблица 5.19 Операторы действия

А	Содержание
1	Уст «0» для Тр (3), СТ2 (16), ПЗУ1(9)
2	Уст «0» Тр (13), Счит. ПЗУ1 (9)
3	Счит. И (2) через И (20) Счит. ПЗУ1 (9) с записью на RgСдв3 (10)
4	Запись RgСдв1 (6) через И (4)
5	Уст «0» Тр (13), +1 СТ2 (16)
6	Уст «0» СТ1 (15), Счит. RgСдв1(6), RgСдв 2(7) с записью RgX (8)
7	Уст «0» Тр (12) Счит. RgX (8), логич. сложение через ИЛИ (21) Запись RgP(25) Запись ПЗУ1 (9)
8	Счит. СТ1 (15), RgP (25) и запись ПЗУ2 (27)
9	Счит. ПЗУ2 (27), арифм. сложение
10	+1 СТ1 (15)
11	Запись RgСдв2 (7) через И (5)
12	Уст «0» СТ3 (17)
13	Уст «0» Тр (14). Счит. рез. суммы. с записью RgСдв4 (23)
14	Сдвиг RgСдв4 (23)
15	+1 СТ3 (17)
16	Пустой оператор

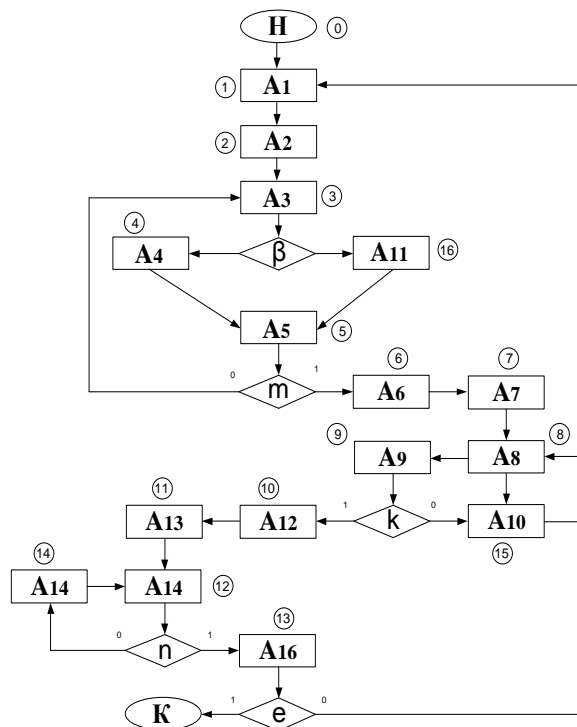


Рисунок 5.20 - Алгоритм управления устройством криптографической защиты информации



Иерархия управления определяется тем, что сам УА для управления синхронизируется во времени так же с помощью автомата, с алгоритмом функционирования представленным на рис.5.20. Благодаря использованию иерархии каждый из уровней управления относительно не сложен. Однако реализация исполнительного уровня управления спецпроцессором в виде УА нижнего уровня по классическим структурам автоматов Мура или Мили окажется весьма не простым.

Действительно, для алгоритма рисунка 5.20 потребуется автомат с памятью на  $N=16$  состояний, с логическими условиями  $\alpha_1\alpha_2\dots\alpha_q$ . При этом число логических условий  $q=4$ . Для  $N=16$  разрядность кода для представления состояний автомата  $a(t)$  и  $a(t+1)$  составит  $m=4$ . Тогда для комбинационной схемы, реализующей переход из предыдущего состояния  $a(t)$  в последующее  $a(t+1)$ , потребуется ПЗУ объемом  $V = m \cdot 2^{m+q} = 1$  кбит.

Использование структурной организации УА с минимальными затратами оборудования на реализацию комбинационных схем дает возможность применить встроенные средства динамического контроля так же на основе оригинальных решений автора.

Таким образом, новый принцип структурной организации аппаратных средств криптографической защиты информации, заключается в следующем:

- комплексирование нескольких алгоритмов защиты информации, что обеспечивает повышенный уровень криптозащиты;
- использование спецпроцессора с микропрограммным принципом управления, что обеспечивает высокое быстродействие полного цикла кодирования и декодирования;
- использование структурных методов повышения надежности работы УА за счет применения оригинальных средств самоконтроля.

На сегодня автору не известны столь быстродействующие средства криптографической защиты информации. Очевидна перспективность применения предложенного метода как в системах с закрытым, так и в системах с открытым ключом шифрования.

### 5.5. Управление реактивным пневмоприводом

Управление средствами мехатроники осуществляется с применением систем автоматического регулирования на основе ПИД регуляторов [182,183]. При высоких требованиях к быстродействию разворота и отсутствии необходимости стабилизации на заданном угле при проектировании используются алгоритмические методы управления робототехникой [183-187].

Важным классом мехатронных систем являются установки с быстрой ориентацией исполнительных механизмов на  $\pm 90^\circ$  в плоскости (x,y) и по вертикали (2) на угол  $\varphi \leq 45^\circ$ .

В качестве таких исполнительных устройств могут быть использованы системы быстрого переноса высоконапорных водяных потоков при ликвидации пожаров на ответственных объектах и др.

При разработке системы управления пневмоприводами в известных конструкциях используются методы теории автоматического управления с применением ПИД регуляторов, однако для быстрого разворота целесообразно использовать реактивные пары. В этом случае при проектировании применяются методы алгоритмического управления робототехникой [187-189].

Принципом управления движением массивных исполнительных механизмов для их разворота на заданный угол  $\theta$  за время в единицах или долях секунды, зависит от конструкции привода. При высоких требованиях к быстродействию исполнения команды разворота необходимо использование двух пар двигателей. Реактивная пара – это два конструктивно связанных двигателя с противоположной направленностью выхлопных сопел, т.е. с противоположным направлением вращения. Реактивная пара может быть реализована с помощью пневмо - гидросистем или на основе «пороховых» зарядов. При этом в случае использования «пороховых» двигателей привод является одноразовым. Один из вариантов реализации реактивной пары в пневмоавтоматике приведен на рис. 5.21.

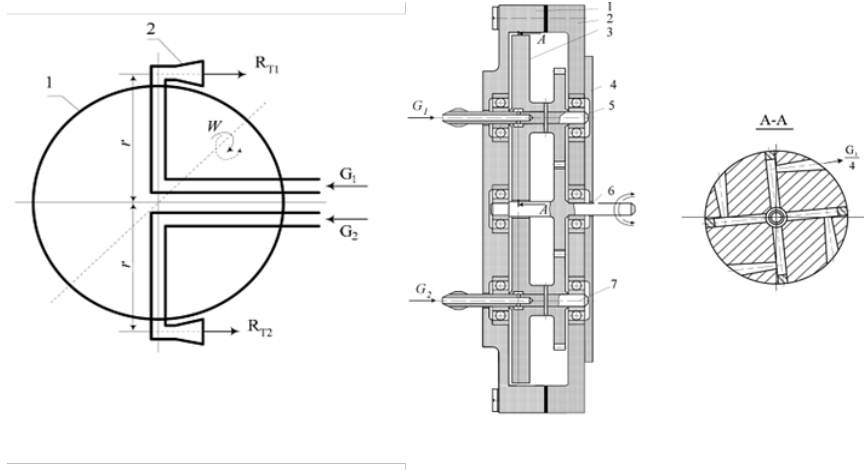


Рисунок 5.21 - Вариант реализации реактивной пары в пневмоавтоматике

Для построения такой пары используются специальные разгоняющие и тормозящие двигатели, отличающиеся малыми габаритами и весом. Включение во время разворота и невозможность какой-либо корректировки после включения требуют разработки особых алгоритмов управления с точным учетом времени включения и отсечки двигателей при реальных параметрах движения (угол, скорость, ускорение).

Вначале включается ускоряющая пара, которая осуществляет разворот на угол  $\theta < \pm 90^\circ$ , а ее отключение от изделия (отсечка) осуществляется при достижении угловой скорости близкой к предельно допустимому значению. Привод такого типа специфичен не только по конструкции, но и по алгоритму управления.

Управление дискретным приводом для разворота изделий на заданный угол за короткое время осуществляется при относительно низкой точности ( $\sim 1^\circ$ ) исполнения команды. На рисунке 5.22 представлено графическое изображение закона управления, но при этом в силу нелинейности всей системы момент времени  $\tau_2$  определяется в виде  $\tau_2 = (\theta_p - \omega_0 T) / (\omega_n + \omega_0)$ , при условии  $\theta_p - \theta - \theta_0 \leq 0$ , где  $\theta$  – угол определяемый в момент  $(\tau_1 + \tau)$ . Проверяются условия  $\omega^* - \omega \leq 0, \theta_0 \leq \theta_p \leq (\pm 90^\circ)$ , где  $\omega^*, \omega_n, \theta_0$  – константы;  $\theta_p$  – расчетное значение угла разворота;  $\theta$  – фактическое значение угла разворота. Точность

выдачи команд управления по углу  $\theta < 0,7^\circ$  при развороте до  $\pm 90^\circ$  позволяет использовать как цифровую, так и аналого-цифровую реализацию Ф- подсистемы. Оценим требуемое быстродействие реализации алгоритма управления дискретным приводом. Заданная точность вычислений по углу  $\theta < 0,7^\circ$  потребует представления угла поворота  $\theta$  восьмиразрядным двоичным кодом, т.к.  $180^\circ / 256 \approx 0,7^\circ$ . Тогда и преобразование сигналов датчиков угла поворота  $\theta$  и угловой скорости  $\omega$  потребуется производить с помощью 8-разрядных АЦП. Наибольший объем вычислений необходимо производить при расчетах  $\tau_2$  на участке поворота после отсечки разгоняющих двигателей. При развороте изделия на  $90^\circ$  за 0,5 сек, даже при равномерном движении, поворот на каждые  $0,5^\circ$  будет проходить за время  $\approx 0,3$  мс. Для расчета и проверки условий выдачи команд управления с точностью до  $0,2^\circ$  потребуется производить два преобразования в двоичный код ( $\omega$  и  $\theta$ ) и определение  $\tau_2$ . Если учесть неравномерность движения на участке разгона, то фактически предельная длительность полного цикла расчета команд  $\leq 100$  мкс. Такое быстродействие и точность реализации вычислительного процесса СПР не является ограничительным фактором для микроконтроллеров.

В алгоритме управления реактивным приводом необходимо учитывать три режима:

1. Изделие не разворачивается, если угол разворота меньше некоторого малого угла;
2. Разворот производится по трапецевидной траектории;
3. Осуществляется релейный режим корректировки по знаку «выхода» изделия за зону допустимой точности «вывода» на заданный угол разворота. Исполнение команд выхода на расчетный угол будет осуществляться с точностью ( $\sim \pm 0,2^\circ$ ) за счет наличия режима слежения.

Алгоритм управления движением представлен на рисунке 5.23.

Расшифровка операторов действия представлена в таблице 5.20, 5.21.

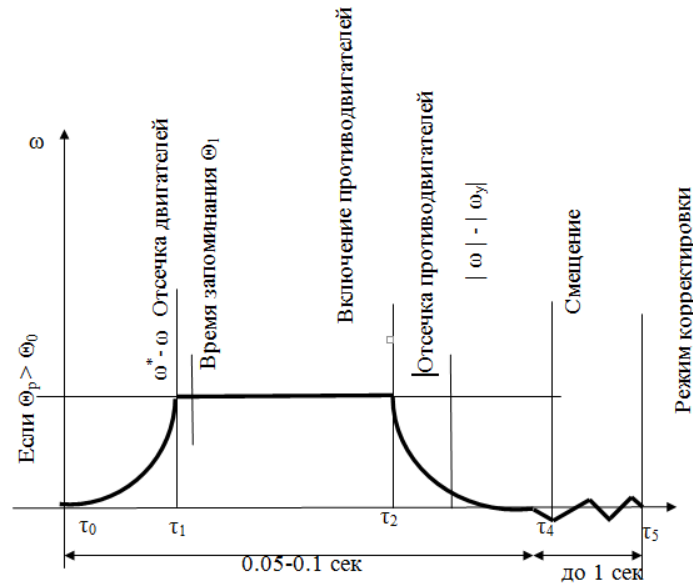


Рисунок 5.22 - Графическое изображение закона управления реактивной пары в пневмоавтоматике.

Таблица 5.20 Расшифровка операторов действия

$A_i$	Семантика	$A_i$	Семантика
1	Нач. установка	12	$t = t+1$
2	Считывание $Q_0$	13	Считывание ПЗУ – Ф3
3	Считывание $Q$	14	Запись $\tau_3$
4	Включение АЦП	15	Вкл. таймера
5	Запись $\theta$	16	Вычисление $\theta - \theta_k$
6	Считывание знач. $\omega$	17	Запись знака
7	Вычисление $ \omega - \omega^* $	18	Вкл. двиг. D1
8	Вычисление $Q_p - \theta$	19	Вкл. двиг. D2
9	Считывание ПЗУ – Ф2	20	Откл. двиг. D1
10	Запись $\tau_2$	21	Откл. двиг. D2
11	Вычисление $t - \tau_2$		

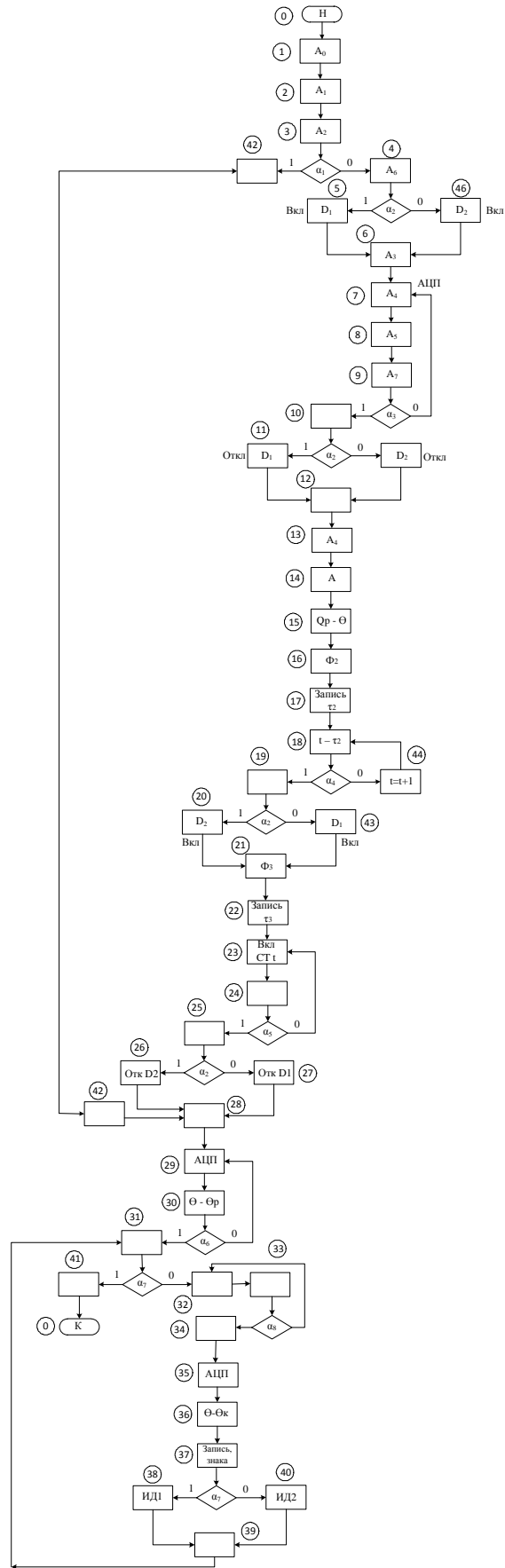


Рисунок 5.23 – Алгоритм управления реактивным приводом

Таблица 5.21 Список логических условий

$\alpha$	Семантика
1	$Q_p > Q_0$
2	Знак $Q_p$
3	$ \omega - \omega^*  \geq 0$
4	$t - \tau_2 > 0$
5	$t - \tau_3 > 0$
6	$\theta - \theta_k > 0$
7	Подрыв

Спецпроцессор управления состоит из датчиков угла, угловой скорости, функциональных преобразователей, счетчика меток времени движения и сумматора- вычитателя. Наиболее сложной подсистемой является управляющая. Как видно по разметке УА относится к классу средней сложности, т.к.  $q=7$ ,  $m=6$ .  $V_p = 8 \cdot 2^{13} = 64$  кб.  $W_p = 8 \cdot 2^7 = 0.1$ кб. Для спецпроцессора управления реактивным движением применение УА нового типа эффективно ( $Q=64$ ) . При этом целесообразно выбрать память со счетчиком, т.к. из 47 состояний номера 39-ти состояний изменяются последовательно.

## 5.6. Управление подсистемами транспортной безопасности

Обеспечение транспортной безопасности (ОТБ) на сегодня является одной из важнейших задач, имеющей общегосударственное значение. Проблема ОТБ является многосторонней (комплексной). В рамках решения задач ОТБ могут выделены отдельные подсистемы, реализуемые на уровне мехатронных систем с микроконтроллерами или автоматическим управлением.

### 5.6.1 Управление охраняемым переездом железнодорожного транспорта

Важное место в обеспечении безопасности на железнодорожном транспорте занимают вопросы управления переездом железнодорожного транспорта на пересечениях с автомобильными трассами. Система управления переездом

гарантирует безопасность как железнодорожного, так и автомобильного транспорта.

Такая задача успешно решается с помощью УА нового типа с  $m=5$ ,  $q=10$ . ГСА представлен на рис.5.24, а соответствующий граф переходов на рис.5.25, расшифровка семантики переходов и операторов действия представлены в таблицах 5.24, 5.25. Как видно из графа переходов все вершины кроме №25 принадлежат траектории счетчика. Эффективность нового УА по сравнению с УА Мура составит  $Q=V/\omega=5/2$ .

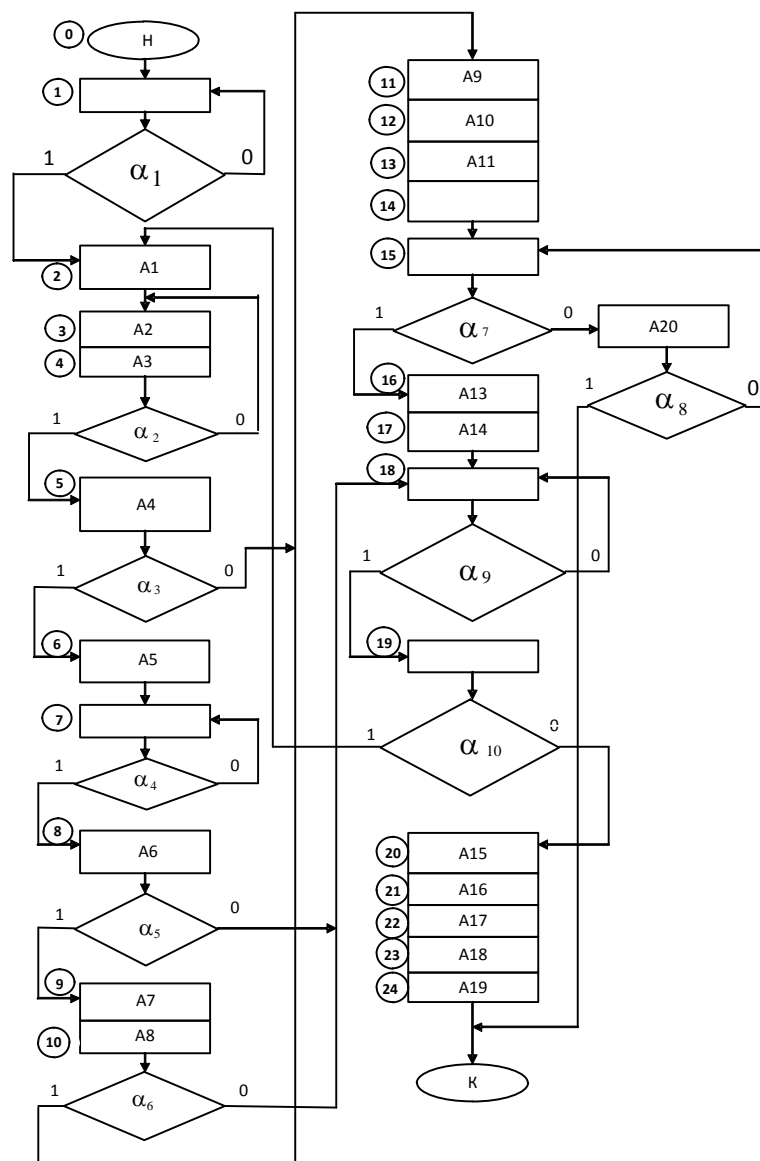


Рисунок 5.24 Алгоритм работы охраняемого переезда



Таблица 5.22 Расшифровка операторов действия

$A_i$	Семантика	$A_i$	Семантика
1	Подключить УОП к АПС	12	Вкл. служебного торможения поезда
2	Измеритель $V_a$ и $L_n$	13	Выкл. заградит. синг.
3	Вычислить $L_n \min$	14	Выкл. кодир. РЦ
4	Включение оповестительной сигнализации	15	Выкл. оповест. сигнализ.
5	Вкл.элемента выдержки $t_3$	16	Выкл. УСЗП
6	Закрыть автошлагбаум	17	УЗП откр. переезд
7	УЗП закрывает переезд	18	Откр. автошлагбаум
8	Вкл.УСЗП	19	Откр. ОУП от АПС
9	Вклю. фоторегистраторов	20	Вкл. аварийного торм.
10	Вкл. заградительных сигн.		
11	Выкл. Кодированных РЦ		

Таблица 5.23 Список логических условий

$\alpha$	Семантика
1	Поезд приближается к переезду
2	$L_n \min < L_n$
3	Оповест. сигн. вкл
4	$t_3$ истекло
5	Автошлагбаум закрыт
6	Препятствие на переезде
7	Препятствие устранено
8	Аварийное торможение вкл.
9	Поезд проследовал переезд
10	2й поезд приближается к переезду

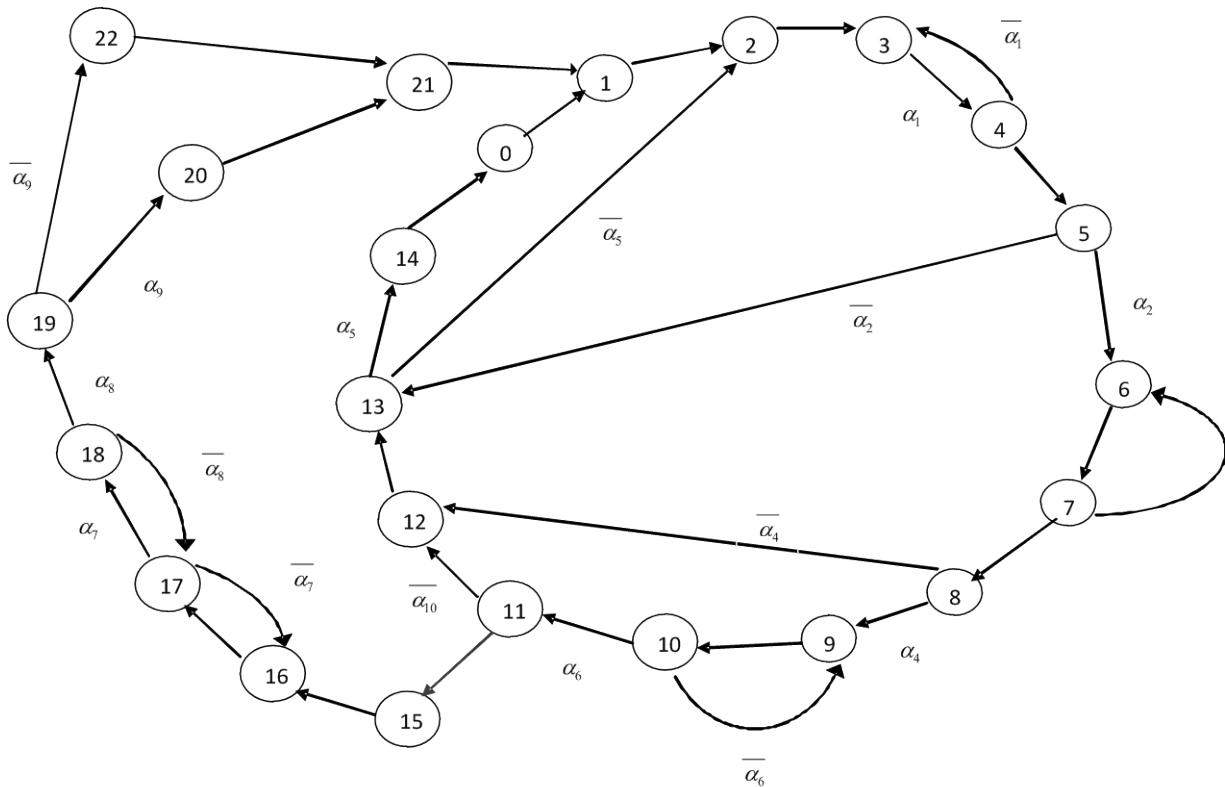


Рисунок 5.25 Переходы УА охраняемого переезда

### 5.6.2. Управление подсистемой автоблокировки железнодорожной автоматики

Система микропроцессорной автоблокировки предусматривает защиту от опасных отказов при коротком замыкании изолирующих стыков сложных рельсовых цепей [151]. Преобразованный алгоритм такой защиты представлен на рис.5.26, а соответствующий ему граф переходов на рис. 5.27. Расшифровка операций и логических условий для оценки характеристики УА не имеет существенного значения.

В табл.5.24 представлено правило выбора номеров логических условий. Как видно из графа переходов в УА целесообразно применить реверсивный счетчик, тогда вне счетчика будет только одна вершина №2, а функция F1 определяется только реализацией переходов 19-22; 22-21; 21-1; 5-13; 8-12; 11-15; 13-2; 14-0.

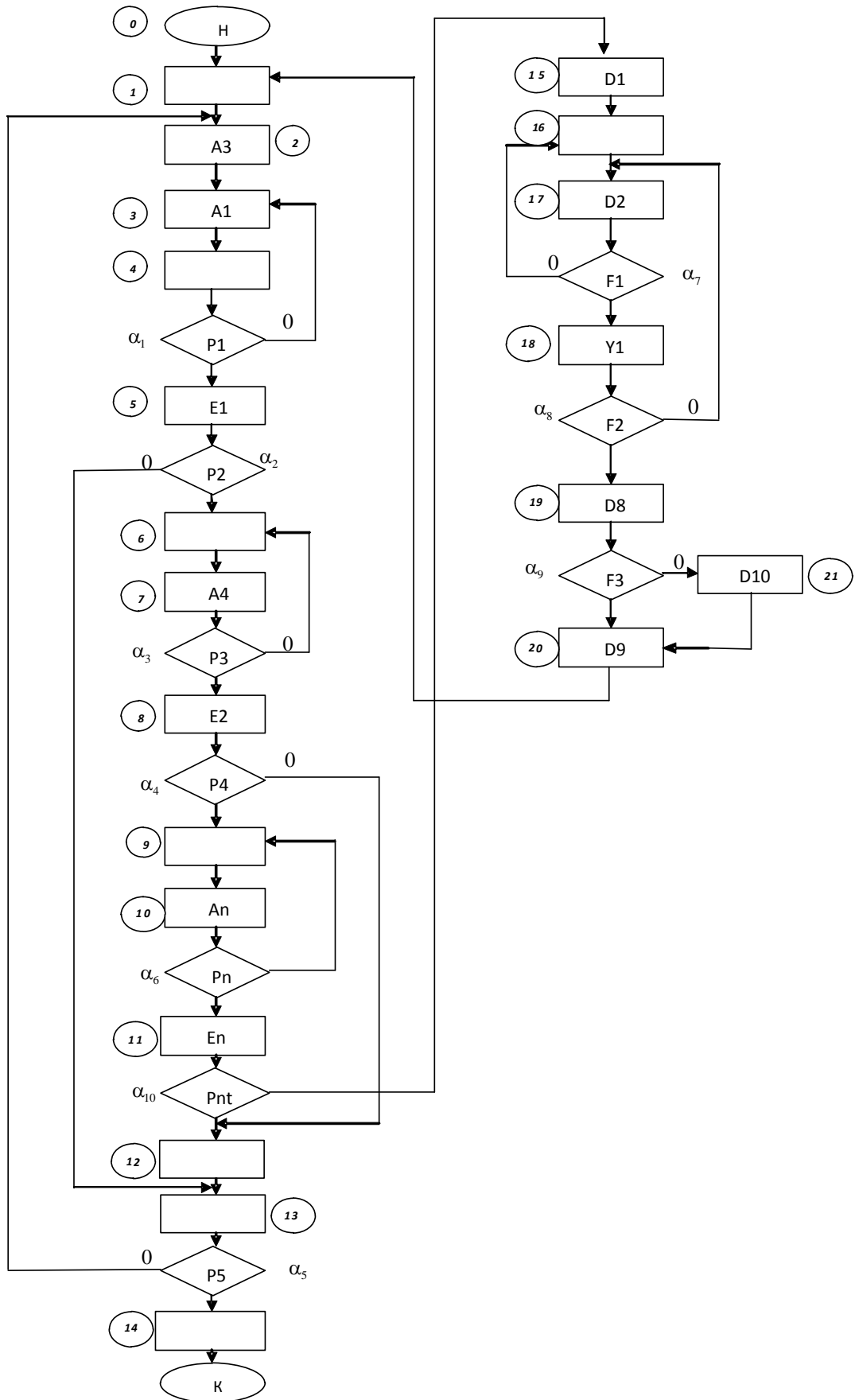


Рисунок 5.26 Алгоритм защиты железнодорожной автоматики

Табл.5.24 Выбор логических условий

$\alpha_1$	4	$\alpha_5$	13	$\alpha_9$	19
$\alpha_2$	5	$\alpha_6$	10	$\alpha_{10}$	11
$\alpha_3$	7	$\alpha_7$	17		
$\alpha_4$	8	$\alpha_8$	18		

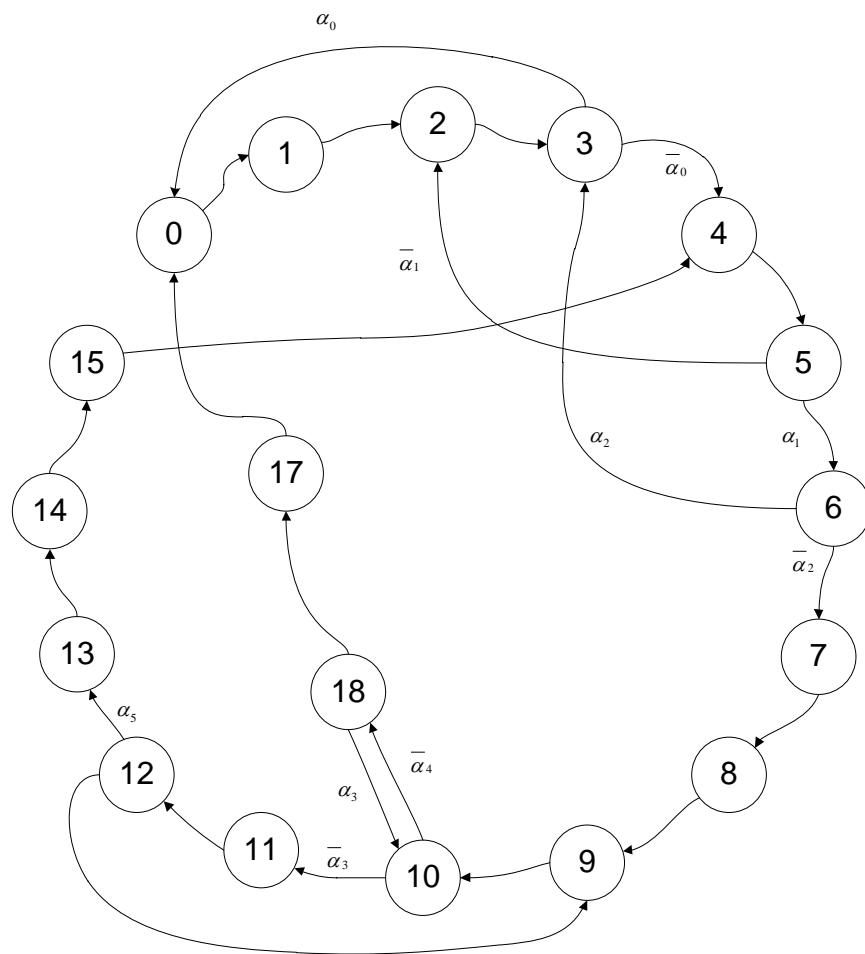


Рисунок 5.26 Граф переходов УА автоблокировки.

### 5.6.3. Управление установкой генерации высоконапорных горячих воздушных струй.

Для очистки промышленных сред (загрязненные поверхности деталей, наледи на железнодорожных стыках, уплотненные снежные заносы на автомобильных дорогах и др.) используются в основном механические способы или высоконапорные горячие воздушные струи.

В работах [12, 45] предложена оригинальная установка получения горячих струй и ультразвука при холодном воздушном напоре в 7-8 атм. без подогрева на основе использования эффекта Ранка (рисунок 5.27).

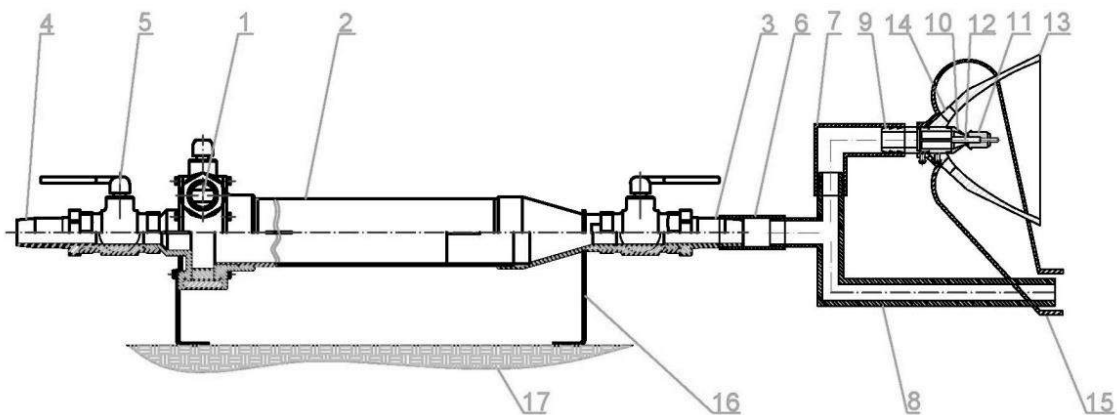


Рисунок 5.27 Общий вид ультразвуковой системы очистки технологических сред с применением вихревого эффекта.

Поддержание соответствующего режима стабильного турбулентного течения воздуха в вихревой трубе реализуется при алгоритмическом управлении включения и отключения четырех насосных подсистем с проверкой почти 20 логических условий.

При традиционном подходе возможно 3 способа реализации ГСА:

- синтез автомата со структурной организацией Мура;
- создание основного ведущего автомата и 4-х автоматов с малым числом состояний, соответствующих повторяющимся подграфам общего графа переходов;
- создание специальной программы для микроконтроллера.

В первом случае комбинационная схема УА при реализации через ПЗУ будет иметь объем  $V = m2^{m+q}$ ,  $m=6$ ,  $q=19$ , но при реальной разрядности ПЗУ равной 8 объем КС составит  $V = 8*2^{6+19} = 0.25\text{Гбит}$ . Реализация ГСА на микроконтроллере потребует разработки также весьма сложной программы.

Отнесем типовой подграф соответствующим переходам (5-22; 9-26; 13-30; 32-35 – рисунок 5.28) в конец новой ГСА состояния (34-39 на рисунке 5.29).

Специфика предложенного метода реализации УА заключается в том, что в едином автомате запоминаются номера состояний (5, 9, 15, 19) обращения к типовому малому подграфу (состояния 34-39), что дает возможность без стековой памяти обеспечить правильность возврата к соответствующему состоянию основного графа переходов после реализации малого подграфа. Такая структурная организация УА обеспечивает дополнительное упрощение комбинационных схем аппаратной реализации УА нового типа, объем которой составит  $W = 8 \cdot 2^{6+1} = 1 \text{ кбит}$ . То есть  $Q = 0,25 \text{ Гбит} / 1 \text{ кбит} = 2,5 \cdot 10^{25}$

Программная реализация УА нового типа осуществляется методом опроса соответствующих зон ОЗУ (ПЗУ) микроконтроллера. В первой зоне записываются функции переходов: 17-2; 2-22; 3-22; 4-23; 5,9,15,19,34; 6-24; 7-25; 8-26; 12-27; 13-28; 12-29; 18-33; 33-30; 32-0.

Во второй зоне ОЗУ записываются функции адресации мультиплексора. Схему реализации выходных команд вместе с дешифратором и мультиплексором целесообразно оставить аппаратно реализуемыми. УА нового типа, реализованный методом структурно-автоматного программирования [44,58], позволяет получить предельно простые программы для микроконтроллера. Такая реализация невозможна (недостижима) ни при одном известном способе программирования микроконтроллеров.

Управление установкой генерации воздушных струй целесообразно осуществлять в виде аппаратно - программной версии автоматов нового типа.

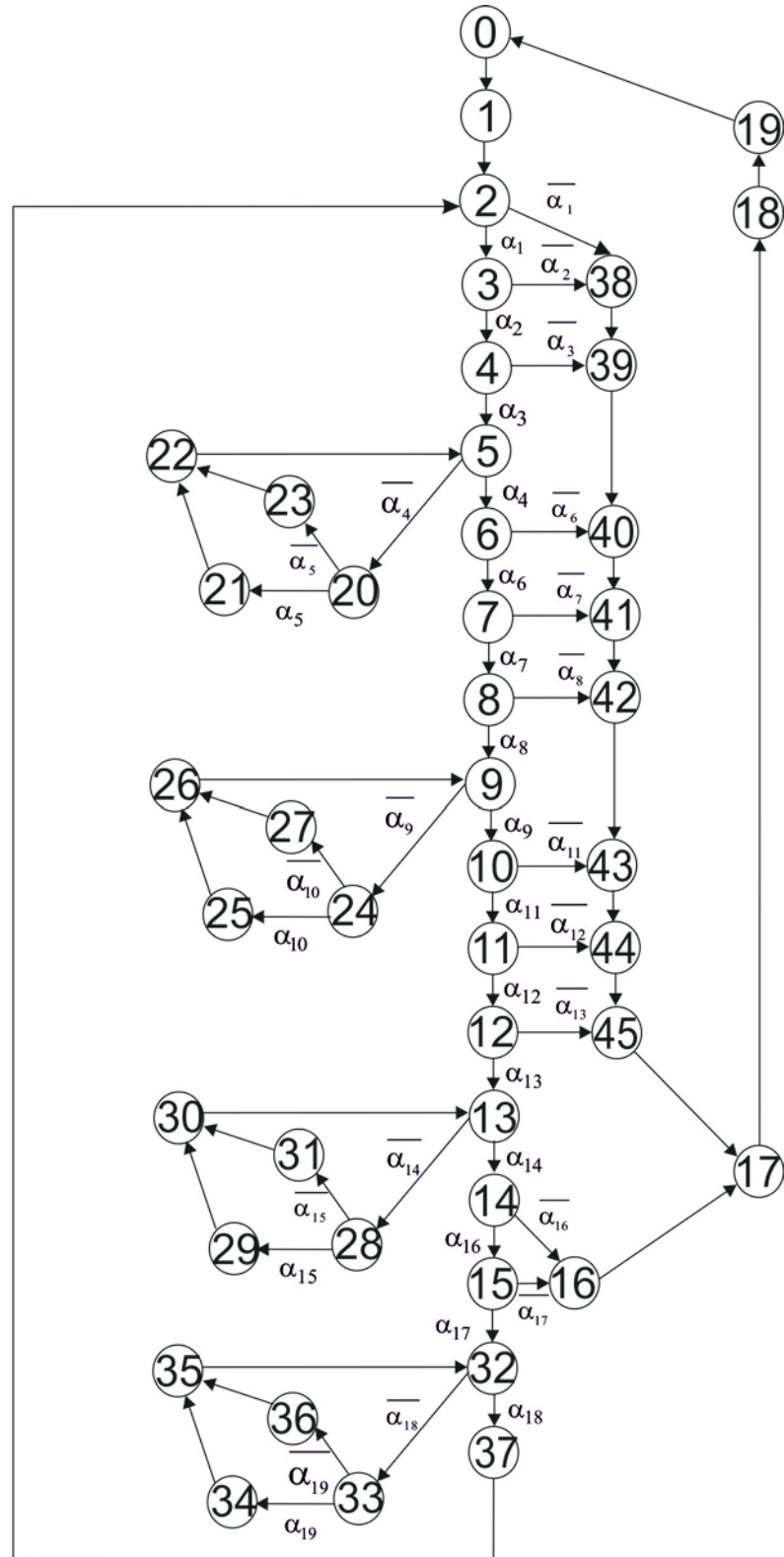


Рисунок 5.28. Граф переходов автоматизированной системы очистки технологических сред

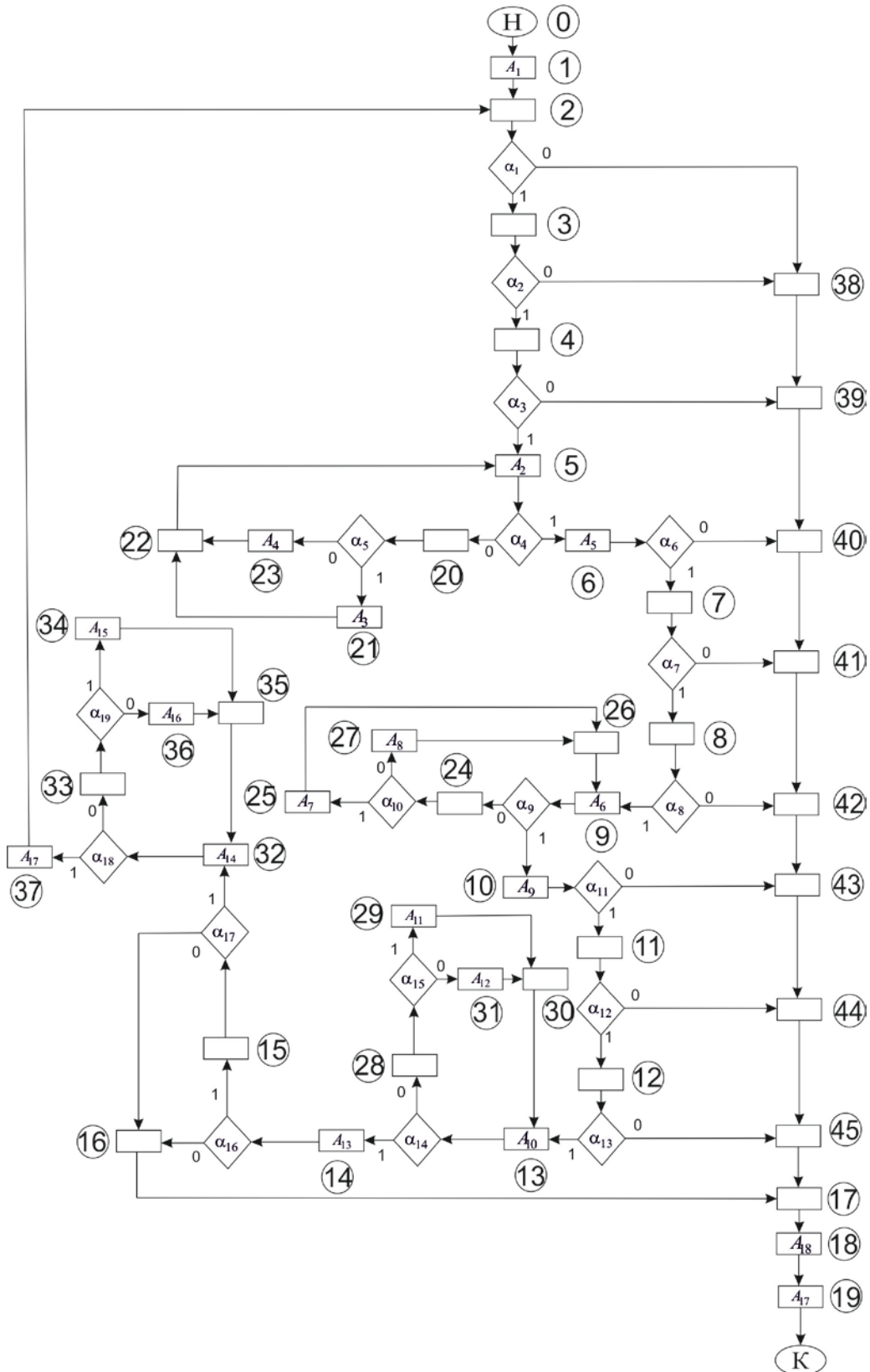


Рисунок 5.29 Алгоритм работы автоматизированной системы очистки технологических сред с включенными пустыми операторами



### 5.7. Управление прибором «Мультимед» для вакуумного и точечного массажа

Рассмотрим оригинальный прибор «Мультимед», представляющий собой 12 специальных медицинских банок с внутренней вставкой.

Основным элементом прибора «мультимед» является типовая медицинская банка с электромагнитной катушкой внутри нее. В катушке имеется выдвижной металлический стержень. Под действием электромагнитного поля катушки стержень нагревается и может оказывать тактильные действия на заданные точки (области), воздействуя на кожные покровы пациента (рисунки 5.30 -5.37).

Закрепление банок на теле осуществляется с помощью системы автоматического поддержания вакуума в банке. Вся система управления обеспечивает комплексные режимы лечения (определяемые врачом). Причем при синтезе по системе ФИЛАУ каждая из подсистем весьма проста по структурной организации, но их взаимодействие во времени требует создания специального управляющего автомата. Управляющий автомат прибора «Мультимед» реализует алгоритм (рис.5.38, с таблицами 5.25, 5.26). Преобразование ГСА и граф переходов представлены на рисунках 5.39, 5.40.

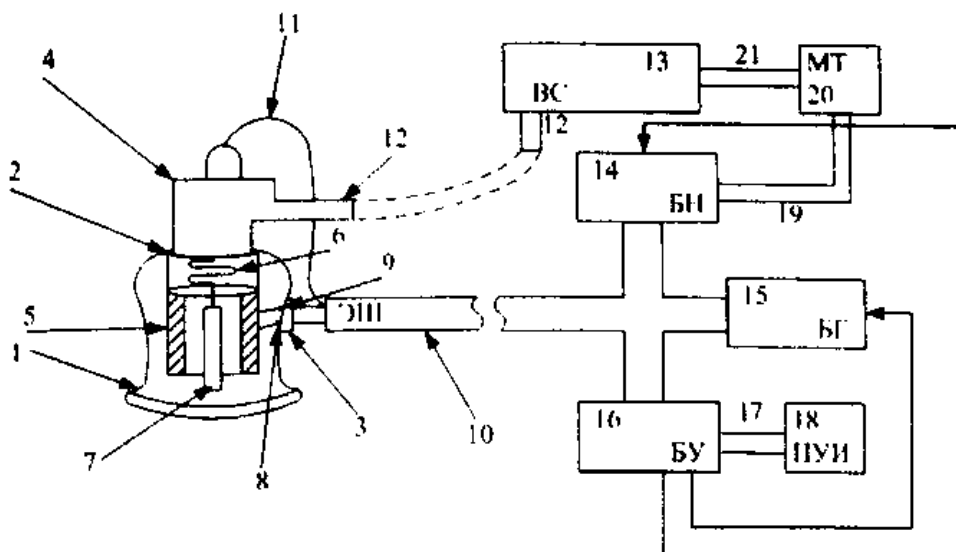


Рисунок 5.30. Структура прибора

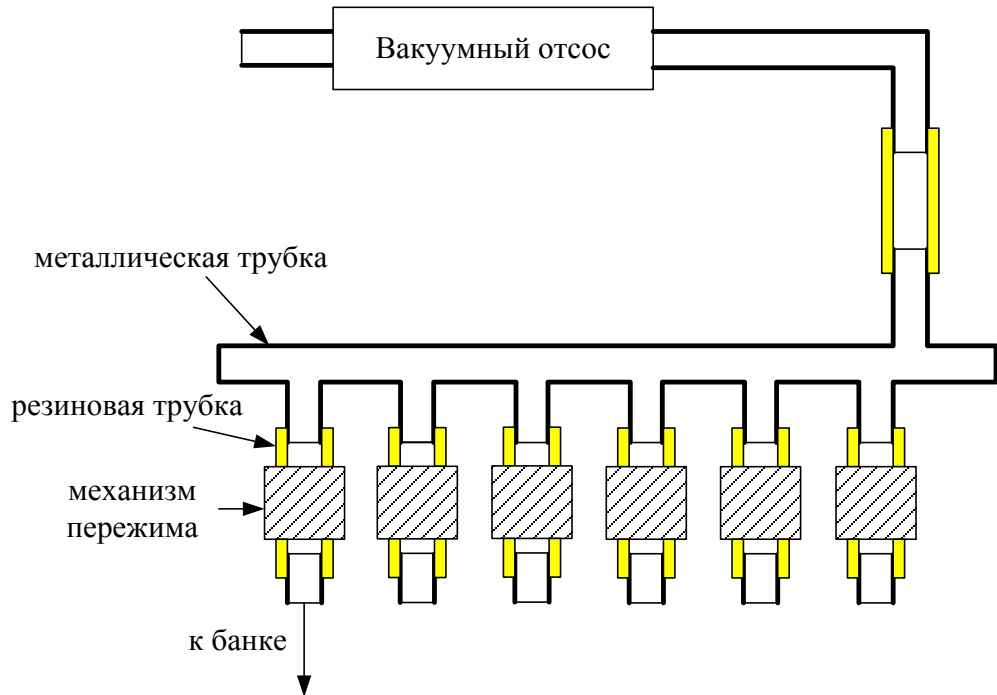


Рисунок 5.31 Схема вакуумного механизма

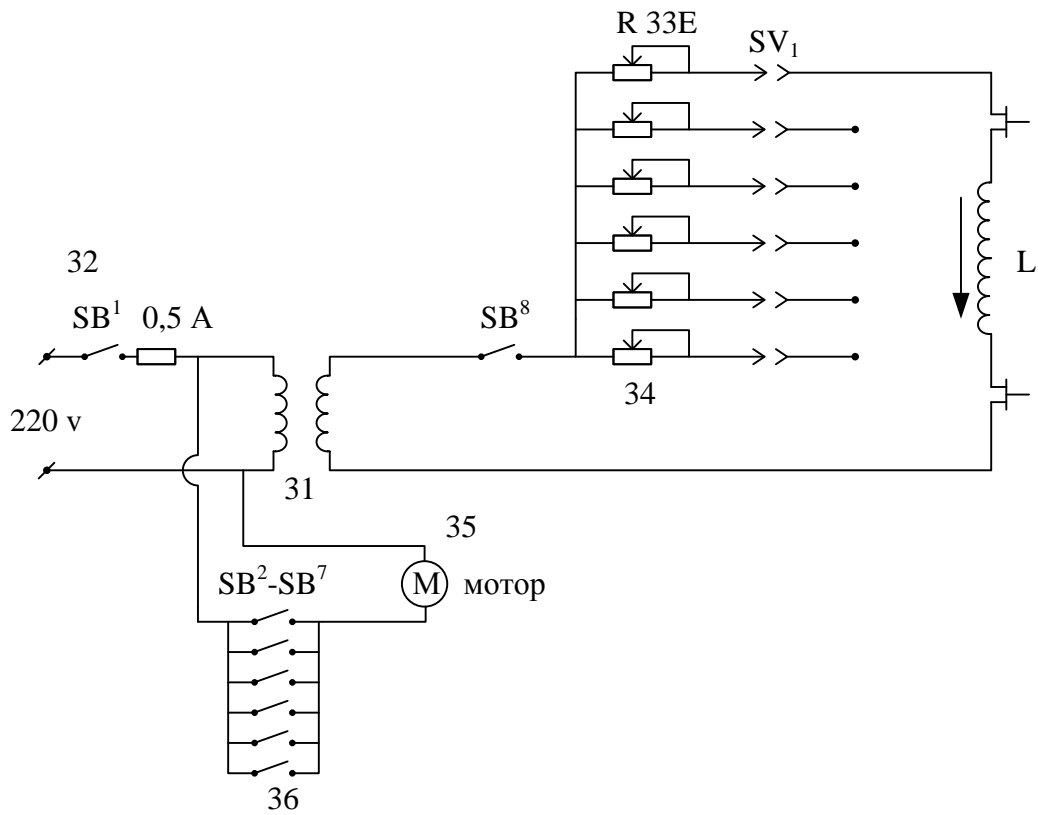


Рисунок 5.32 Электрическая схема нагрева.

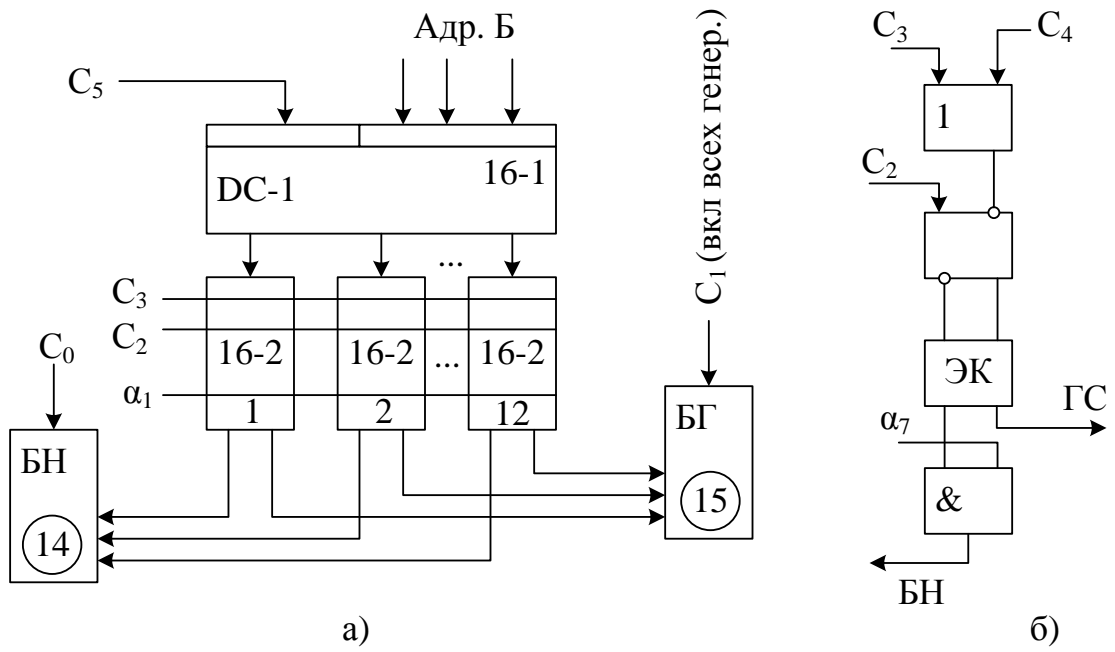


Рисунок 5.33. Блок включения генераторов (а) с элементом переключения б

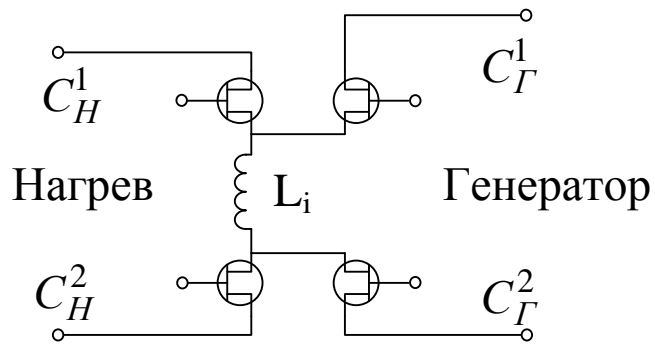


Рисунок 5.34 Элемент коммутации (ЭК)

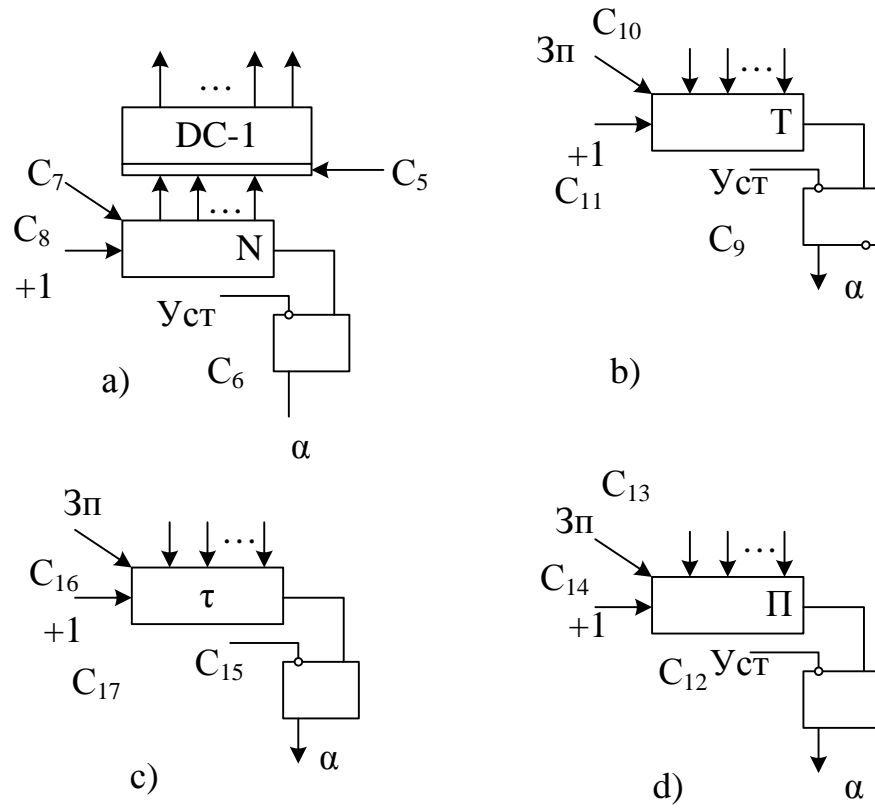
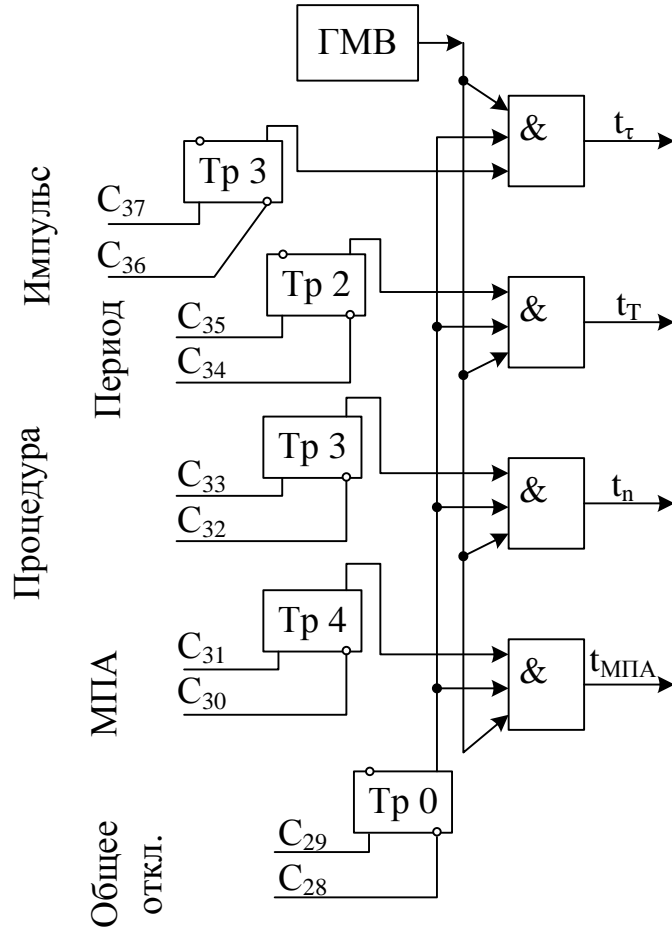


Рисунок 5.35 Блок счетчиков.



$C_{39}$  – Пуск (Вкл МПА)

$C_{38}$  – Ост (Откл МПА)

Рисунок 5.36 Схема подключения меток времени

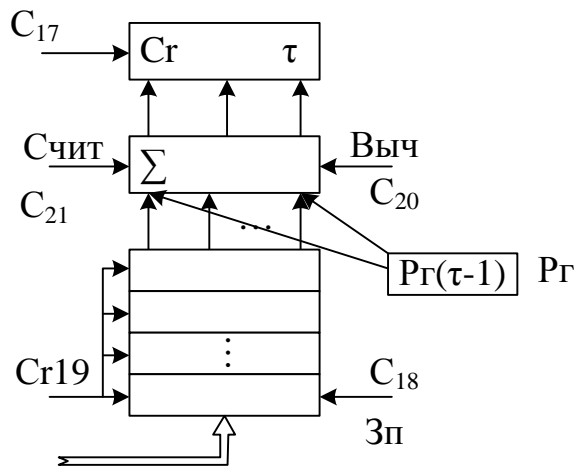


Рисунок 5.37 Спецпроцессор.

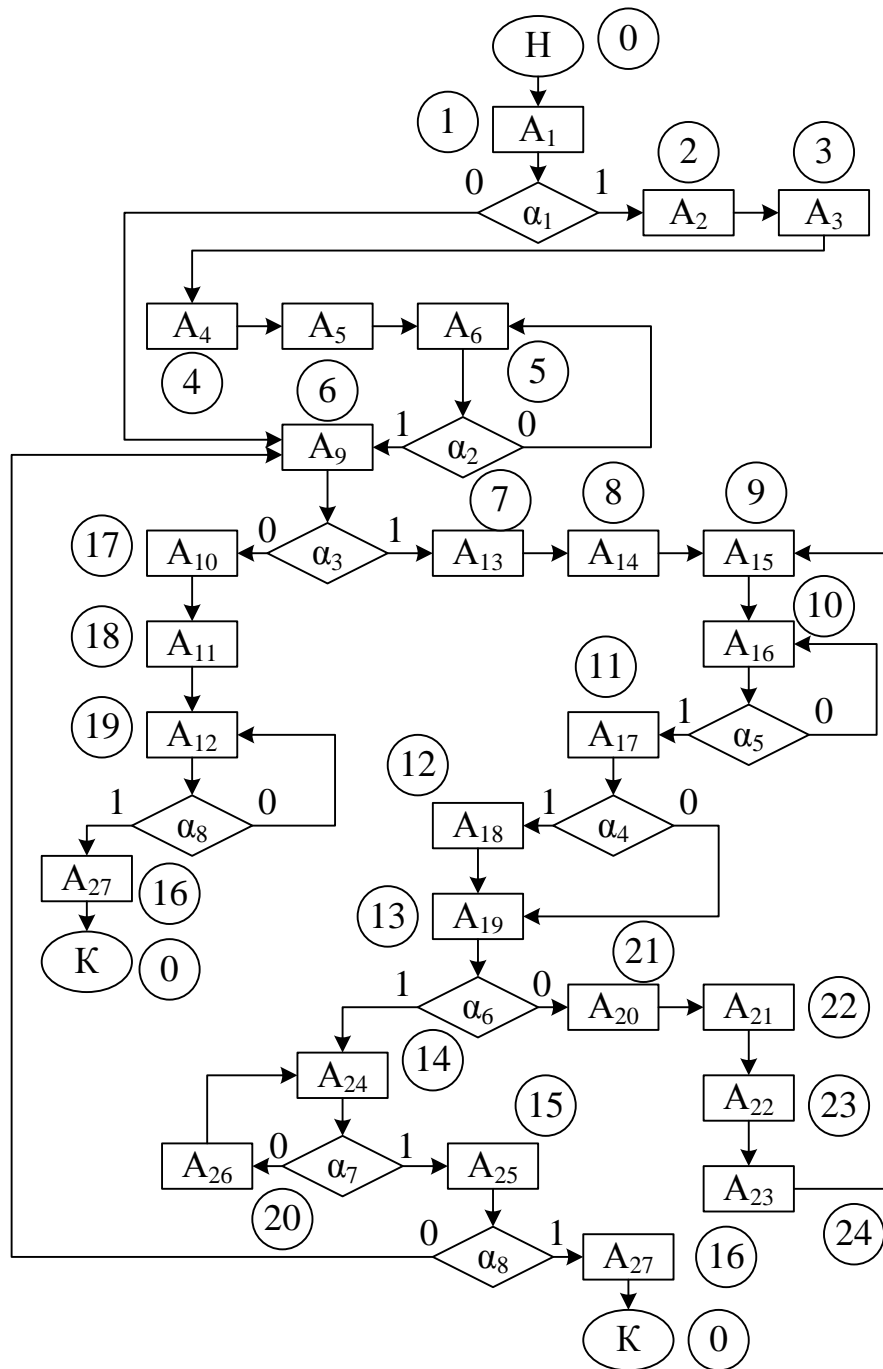


Рисунок 5.38 Алгоритм управления.

Таблица 5.25 Таблица микроопераций

$C_i$	Содержимое	$C_i$	Содержимое
0.	Вкл БН	22.	Сг $t_n$
1.	Вкл всех ГИ	23.	Сг $t_n$
2.	Откл $i$ -того ГИ	24.	Счит $\tau$ в $\sum c P\Gamma(\tau - 1)$
3.	Вкл $i$ -того нагр	25.	Зп на $P\Gamma(\tau - 1)$
4.	Вкл всех N-нагр	26.	Счит БН в счетч. Т
5.	Опрос DC-1	27.	Зп на $P\Gamma(\tau - 1)$
6.	Уст «0» Тр - N	28.	Откл ГВМ (общ)
7.	Зп Сг - N	29.	Вкл ГВМ (общ)
8.	+1 Сг - N	30.	Откл МПА (f)
9.	Уст 0 Тр - Т	31.	Вкл МПА (f)
10.	Зп Сг - Т	32.	Откл Проц (f)
11.	+1 Сг - Т	33.	Вкл Проц (f)
12.	Уст 0 Тр - П	34.	Откл Т (f)
13.	Зп Сг- П	35.	Вкл Т (f)
14.	+1 Сг - П	36.	Откл $\tau$ (f)
15.	Уст 0 Тр - $\tau$	37.	Вкл $\tau$ (f)
16.	Зп Сг - $\tau$	38.	Останов МПА
17.	+1 Сг - $\tau$	39.	Пуск МПА
18.	Зп Стек	40.	Общее отключение
19.	Счит Стек	41.	Вкл $t_n$
20.	Считывание $\sum$	42.	Откл $t_n$
21.	Вычитание		

В табл. 5.25 введены сокращения:

$C_i$  –  $i$ -тая микрооперация

ВКЛ – включить

ОТКЛ – отключить

Нагр – нагреватель

Уст – установка

Зп – запись

Сг – считывание

+1Сг – прибавление 1 к содержимому счетчика (Сг)

Стек – стековая (магазинная) память

ГВМ – генератор временных меток

Таблица 5.26 Таблица команд

№	A	Микрооперации
	0	Пуск
	1	Установка начального состояния $A_9$ ( $A_7, A_8$ )
	2	0, 26
	3	9
	4	29, 31, 38
	5	11
	6	28, 30, 39
	10	23
	11	12, 28, 31
	13	29, 35, 42
	14	16, 19, 27
	15	6, 15, 31
	16	17
	17	11, 14, 30, 36, 39
	18	0
	20	24
	21	19, 21
	22	16
	23	8, 15
	26	11, 14
	27	28, 40
	12, 19, 24, 25	Пропуск такта



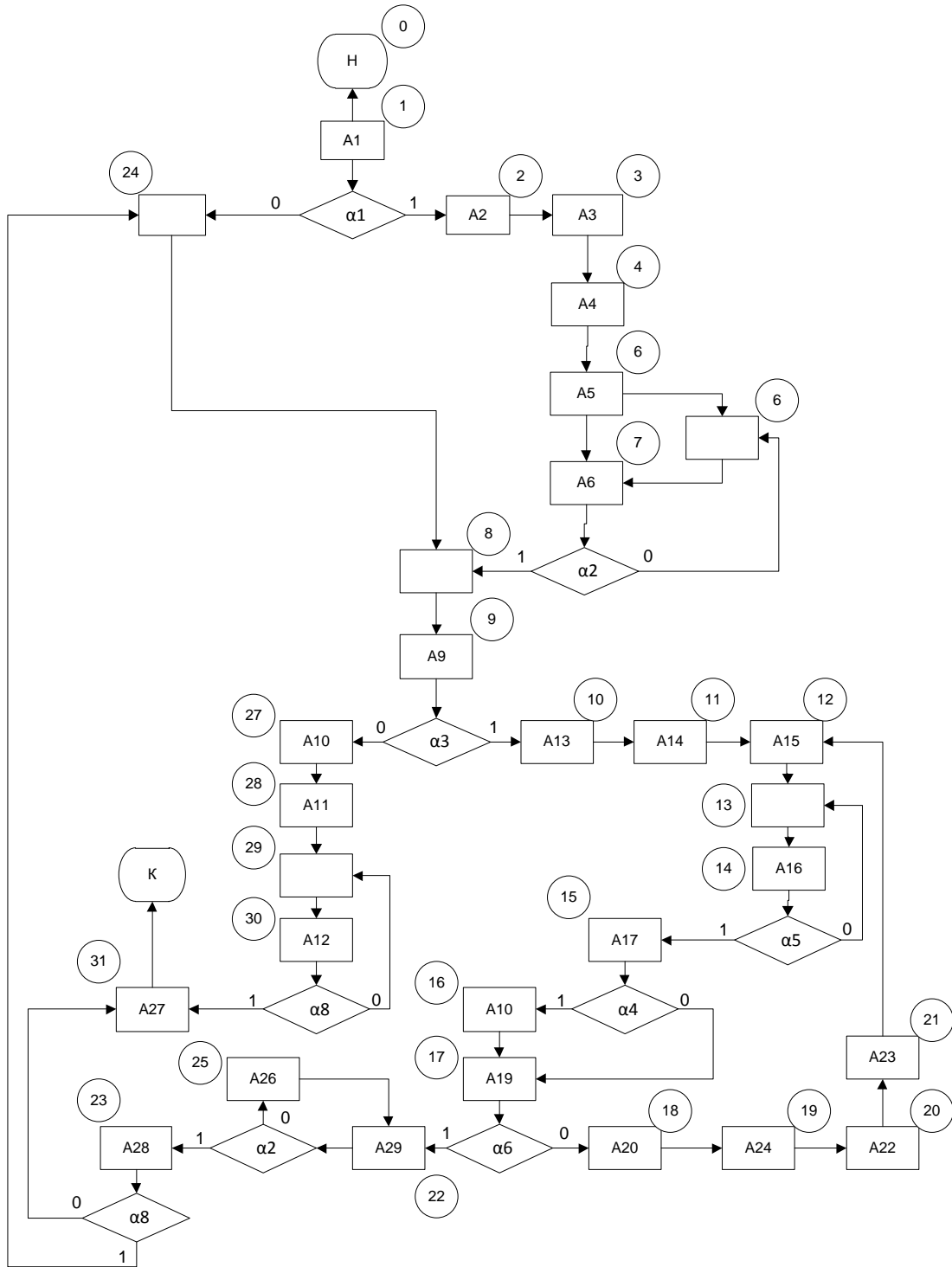


Рисунок 5.39 Преобразованный алгоритм управления

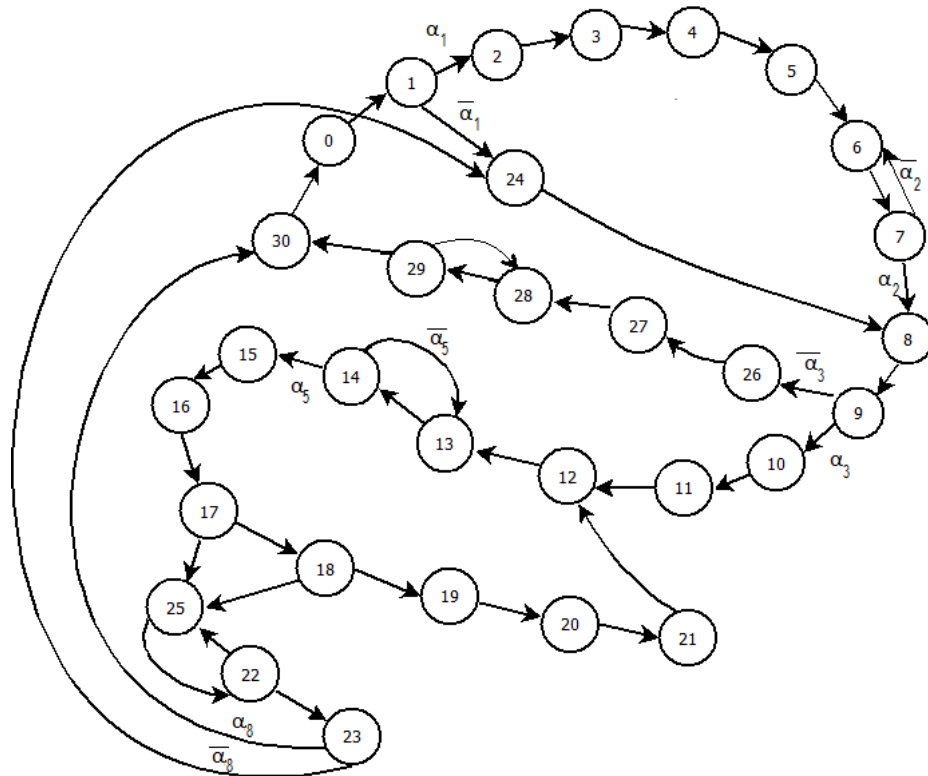


Рисунок 5.40 Граф переходов управляющего автомата

Как видно из графа переходов все вершины графа за исключением 25-ой относятся к счетчику: 0 – 22; 23 – 25; 27 – 0. Т.е. в УА с информационной подсистемой в виде реверсивного счетчика объем комбинационной схемы F1 существенно снижается по сравнению с базовым объемом. Функция F1 сводится в данном случае к генерации кодов  $y_1 \dots y_5$ , соответствующих переходам 1-24; 24-8; 9-26; 17-25; 25-22; 21-12; 30-0.

Для  $m=5$ ,  $q=8$  объем ПЗУ будет снижен в более чем 130 раз.

Самоконтроль УА «Мультимед» может быть эффективно реализован на основе кодов РСN.

## Выводы по главе 5

Спроектированы спецпроцессоры и устройства управления для важных народно-хозяйственных применений, к которым относятся корреляционно-экстремальная навигация, классификация зон изображений, аппаратная реализация криптографической защиты информации, управление реактивным пневмоприводом, управление турбомеханизмами и др. Показана высокая эффективность новой методологии синтеза управляющих автоматов. По криптографической защите получено четыре патента, а по остальным применениям поданы заявки на полезные модели и изобретения.

По вопросам применения методологии проектирования управляющих автоматов и спецпроцессорам (материалам главы 5) автором опубликованы работы [13-21,24,30,34,42,47,49,53,56-58,71], получены патенты [60,61,63,64,66].

## ЗАКЛЮЧЕНИЕ И ОСНОВНЫЕ РЕЗУЛЬТАТЫ

Основным научным результатом диссертационной работы является решение крупной научно-технической проблемы, имеющей важное народно-хозяйственное значение в области разработки систем управления сложными техническими системами и технологическими процессами, а именно создание метода синтеза аппаратно и программно реализуемых управляющих автоматов спецпроцессоров информационно-управляющих систем реального времени.

Разработаны новые методы анализа и синтеза управляющих автоматов, включая синтез взаимодействующих и самоконтролируемых автоматов. Комплекс методов в совокупности составляет основу методологии проектирования управляющих автоматов нового типа – автоматов с выбором одного логического условия из всего множества логических переменных, отличающихся значительным снижением затрат оборудования и энергопотребления, а также высоким уровнем надежности. Управляющие автоматы нового типа могут входить в состав спецпроцессоров, вычислительных комплексов, информационно-измерительных систем, сверхбольших микроэлектронных интегральных схем и/или иметь самостоятельное значение.

При решении основных задач диссертационной работы получены следующие результаты:

1. Выполнен системный анализ методов и средств синтеза спецпроцессоров и управляющих устройств. Показано, что спецпроцессор представляется в виде функциональной (Ф), информационной (И), логической (Л), адресной (А) и управляющей (У) подсистем. Управляющая подсистема сама может быть представлена моделью ФИЛАУ, в которой наиболее сложна Ф подсистема в виде управляющего автомата (УА).

2. Разработан новый метод синтеза управляющих автоматов по заданной операторной схеме алгоритма, основанный на введении пустых операторов для разделения логических условий, если между ними нет операторов действия или к ним передается управление от нескольких операторов. Предложенный метод применим для синтеза автоматов с любым числом логических условий  $q$  при разрядности кода состояний  $m$ . Метод наиболее эффективен для сложных, особо сложных и ультрасложных управляющих автоматов.

3. На основании предложенного метода синтеза разработана оригинальная структурная организация управляющих автоматов с выбором одного логического условия  $\alpha_j$  по коду состояния автомата  $a(t)$  через мультиплексор с импульсно-кодовой организацией памяти и дополнительной комбинационной

схемой определения кода адреса логического условия. Новый тип автомата отличается снижением объема памяти ПЗУ, требующегося для реализации комбинационных схем в  $2^{q-1}$  раз по сравнению с известным вариантом управляющего автомата.

Предложены варианты реализации УА нового типа, позволяющие дополнительно снизить объем ПЗУ в 2-3 раза в УА, спроектированными по новой методике. Разработаны УА с разделенной комбинационной схемой переходов по значению логического условия, а также с независимой реализацией комбинационных схем переходов для младших и старших разрядов.

4. Рассмотрены модификации иерархических и взаимосвязанных автоматов. Предложен метод декомпозиции ГСА, позволяющий создать единый многорежимный УА с сокращенной разрядностью памяти, более простым дешифратором и переключением режимов при динамическом функционировании.

5. Предложен новый метод динамического контроля УА: метод контроля как модификация кода с фиксированным числом единиц ( $kCn$ ), который преобразован в код  $pCn$ , где  $p$  – число трехразрядных групп в выходном коде комбинационной схемы переходов. Комбинационная схема реализуется в виде двух полусхем половинной разрядности с кодом  $pCn$ . Метод с кодами  $pCn$  позволяет контролировать не только выходной код комбинационной схемы переходов, но и код предыдущего состояния  $a(t)$ , представленного в виде двоичного непозиционного кода. Метод на основе кодов  $pCn$  позволяет существенно снизить затраты оборудования встроенных средств контроля.

Предложена модификация метода дублирования за счет представления комбинационной схемы УА двумя схемами половинной разрядности ( $m/2$ ) каждая. К кодам младших и старших разрядов аргументов добавляется по одному разряду, значения которых доопределяются специальным образом.

Предложен новый метод мажоритарного резервирования (троирования) за счет использования третьей схемы не прямого, а обратного преобразования кода состояния  $a(t+1)$  к коду состояния  $a(t)$ . Схема обратного преобразования так же дублируется.

Предлагается так же реализация каждой комбинационной схемы прямого и обратного преобразования в виде двух полусхем с опросом по  $\alpha$  и  $\bar{\alpha}$ , что позволяет обеспечить работоспособность УА в условиях выхода из строя отдельных полусхем, т.е. поднять уровень надежности.

6. Предложен оригинальный метод динамического контроля УА с памятью на счетчике Грея, отличающийся тем, что сравнение кодов  $a(t)$  и  $a(t+1)$  ведется отдельно схемами половинной разрядности. При этом к старшим группам разрядов  $a(t)$  и  $a(t+1)$  добавляется по одному дополнительному разряду, значение

которых доопределяется так, чтобы сумма по mod2 была равна 1. Это условие выполняется только для соседних состояний, не относящихся к последовательным путям. Для путей, определяемых счетчиком, как младшие, так и старшие разряды контролируются по правилу кодов Грея.

7. Предложен новый метод структурно-автоматного программирования, основанный на использовании структурной модели автоматов нового типа. Программный код, полученный методом структурно-автоматного программирования, основан на использовании трех зон (таблиц) памяти ОЗУ, содержание которых меняется при замене операторной схемы алгоритма. Ведущая программа, управляющая обращениями к зонам памяти, составляется один раз и не меняется для любых схем алгоритмов. На сегодня нет более простых и эффективных методов моделирования УА и методов разработки программ информационно-управляющих систем реального времени, чем предложенный метод структурно-автоматного программирования.

Методы 1-8 определяют основы новой методологии проектирования управляющих автоматов сложных технических систем и технологических процессов.

8. Разработаны оригинальные спецпроцессоры с УА нового типа для криптографической защиты информации, корреляционно-экстремальной навигации, классификации типов подстилающей поверхности радиолокационного зондирования местности, управления реактивным пневмоприводом, сглаживания волн давления нефтеперекачивающих станций и др.

9. Предложена новая методология проектирования управляющих автоматов спецпроцессоров сложных технических систем реального времени.

Дальнейшим развитием исследований может быть решение задач функционального и электрического синтеза управляющих автоматов нового типа, а так же синтез адаптивных систем автоматического управления и регулирования с использованием нечеткой логики и нейронных сетей с аппаратно-программной перестройкой их структурной организации на базе предложенной в диссертации методологии проектирования управляющих автоматов.

## СПИСОК СОКРАЩЕНИЙ

АЗУ – ассоциативное запоминающее устройство

ОЗУ – оперативное запоминающее устройство

ПЗУ – постоянное запоминающее устройство

(А) – адресная подсистема

(И) – информационная подсистема

(Л) – логическая подсистема

(Ф) – функциональная подсистема

(У) – управляющая подсистема

АЦП – аналого-цифровой преобразователь информации

ФПИ – функциональный преобразователь информации

МПА – микропрограммный автомат

УА – управляющий автомат

СУА – самоконтролируемый управляющий автомат

Сч – счетчик

БС – блок синхронизации

ИУС – информационно-управляющая система

СПР – спецпроцессор

F – функциональная (комбинационная) схема

ДС – дешифратор

М – мультиплексор

КЗИ – криптографическая защита информации

СТП – стековая память первого типа

СТПН – СТП с неизвестным изображением

СТПЭ – СТП с эталонным изображением

ПЛМ – программируемая логическая матрица

ПЛИС – программируемая логическая интегральная схема

ПИД – пропорциональный интегрально-дифференциальный регулятор

НС – нечеткие системы (системы с элементами нечеткой логики)

БИС – большая интегральная схема

СБИС – сверхбольшая БИС

МИС – схема с малым уровнем интеграции

СИС – схема со средним уровнем интеграции

### СПИСОК ОБОЗНАЧЕНИЙ

$a(t)$  ,  $a(t+1)$  – предыдущее и последующее состояния автомата

$\{A\}$  – множество операторов действия  $A_1, A_2, \dots, A_k$

$\{\alpha\}$  - множество логических условий  $\alpha_1, \alpha_2, \dots, \alpha_q$

$\langle Rr \rangle$  - содержимое регистра памяти

$\langle F \rangle \rightarrow Rr$  – пересылка результата  $F$  на регистр



## СПИСОК ЛИТЕРАТУРЫ

## Основная литература

1. Усков А.А. Системы с нечеткими моделями объектов управления: Монография. – Смоленск: Смоленский филиал АНО ВПО ЦС РФ "Российский университет кооперации", 2013. – 153 с.: ил.
2. [http://attempts-to-see.blogspot.ru/2012/10/blog-post\\_26.html](http://attempts-to-see.blogspot.ru/2012/10/blog-post_26.html)
3. <http://www.mscsoftware.ru>
4. Перельройзен, Е.З. Проектируем на VHDL / Е.З. Перельройзен. - М.: «Солон – Пресс», 2004. - 448 с.
5. Соловьёв, В.В. Логическое проектирование цифровых схем на основе программируемых логических интегральных схем / В.В. Соловьёв, А. Климович.- М.: Горячая Линия – Телеком, 2008. -376 с.
6. Амосов, В.В. Схемотехника и средства проектирования цифровых устройств / В.В. Амосов.- СПб.: БХВ – Петербург, 2007. -542 с.
7. Панченков, А.Н. Методологические и математические вопросы теории больших систем / А.Н. Панченков // В кн. Оптимизация и управление в больших системах энергетики. Том 1. - Иркутск: СЭИ, 1970. – С. 35-66.
8. Труды по теории синтеза и диагноза конечных автоматов и релейных устройств / под ред. В.В. Сапожникова и Вл.В. Сапожникова.- СПб.: Элмор, 2009. – 960 с.
9. Хетагуров, Я.А. Проектирование информационно - вычислительных комплексов / Я.А. Хетагуров, Ю.Т. Древс. - М.: Высшая школа, 1987. - 282 с.
10. Агеев, А.М. Декомпозиция контуров управления самолетом через условия инвариантности / А.М. Агеев, В.Н. Сизых // Инновации в авиационных комплексах и системах военного назначения: сб. научн. статей в гр. Всерос. Научн. - практ. конф. – Воронеж.: изд-во ВАИУ, 2009. – ч.11.- С.5-10.
11. Никищенков, С.А. Автоматизированное диагностирование железнодорожных технологических систем с использованием операторных схем

- процессов / С.А. Никищенков. – Самара: Самарский научный центр РАН, 2007.- 181 с.
12. Чечкин, А.В. Математическая информатика / А.В Чечкин. - М.: Наука,1991.-416с.
13. Чечкин, А.В. Принципы и методы математического моделирования интеллектуальных систем / А.В. Чечкин // Интеллектуальные Системы. – М.: МГУ. - т.3.- вып.1-2, 1998.- С. 63-83.
14. Воевода, А.А. Применение сетей Петри на этапе объектно-ориентированного проектирования / А.А. Воевода, Д.В. Прытков. - Новосибирск: Сб. науч. тр. НГТУ.- №2(60), 2010.- С. 65-76.
15. Новик, К.В. Сеть автоматов для моделирования асинхронного взаимодействия процессов: автореф. дис. ...канд.физ-мат-наук: защищена : утв. / К.В. Новик.- М.: МГУ, 2006.- 23 с.
16. Горбатов, В.А. Теория автоматов / В.А. Горбатов, А.В. Горбатов, М.В. Горбатова. - М.: Астрель, 2008. -560 с.
17. Водяхо, А.И. Функционально-ориентированные процессоры /А.Н. Водяхо., В.Б.Смолов., В.У.Плюснин, Д.В.Пузанков. – Л.: Машиностроение, 1988.-224с.
18. Уилмсхерст, Т. Разработка встроенных систем с помощью микроконтроллеров PIC / Т. Уилмсхерст.- Киев: «МК-Пресс»; СПб.: «КОРОНА – ВЕК», 2008. – 544 с.
19. Кошкин, В.Л. Аппаратные системы числового программного управления / В.Л. Кошкин. - М.: Машиностроение, 1989.- 245с.
20. Видениекс, П.О. Проблемно-ориентированные микропроцессорные системы в производстве РЭА / П.О. Видениекс, Я.Я. Вейнтиш, А.А. Кривченков.- М.: Радио и связь, 1987.-296с.
21. Мухопад, Ю.Ф. Проектирование специализированных микропроцессорных вычислителей / Ю.Ф. Мухопад. - Новосибирск: Наука, 1981.-162 с.
22. Мухопад, Ю.Ф. Микроэлектронные информационно-управляющие системы/ Ю.Ф. Мухопад. - Иркутск: ИрГУПС, 2004.- 407 с.

23. Тарасенко, Ф.П. Прикладной системный анализ / Ф.П. Тарасенко. – М.: КноРус, 2010. - 272с.
24. Сидельников О.В. Теория кодирования / Сидельников О.В. - М.: Физматлит, 2008. – 289 с.
25. Глушков, В.М. Автоматно-алгебраические аспекты оптимизации МПА / В.М. Глушков / Тр. межд. матем. конгресса. - М., 1968. - С. 53 - 55.
26. Мухопад, Ю.Ф. Теория дискретных устройств / Ю.Ф. Мухопад. - Иркутск: ИрГУПС, 2010. - 172 с.
27. Мухопад, Ю.Ф. Анализ структурных схем вычислителей по автоматным моделям / Ю.Ф. Мухопад // Тр. междунар. конгресса IFAC: Сб. Дискретные системы. - Рига.- Т. 5, 1974. - С. 131 - 140.
28. Мухопад, Ю.Ф. Системный анализ машины Тьюринга / Ю.Ф. Мухопад // сб. Микропроцессорные системы. - Новосибирск: НЭТИ, 1990.- С.47-52.
29. Мухопад, Ю.Ф. Системная модель микропроцессорных устройств. / Ю.Ф. Мухопад //сб. Локальные вычислительные сети и распределенная обработка данных. - Новосибирск: НЭТИ, 1991. С. 63-71.
30. Локтюхин, В.Н. Нейросетевые преобразователи импульсно-аналоговой информации / В.Н. Локтюхин, С.В. Челебаев. - М.: Горячая линия-Телеком, 2008.- 144с.
31. Гаврилов, М.А. Теория релейно-контактных схем / М.А.Гаврилов. - М.: Изд-во АН СССР, 1950. – 284 с.
32. Жоффрен, И. Кодирование внутренних состояний и декомпозиция последовательных синхронных устройств. // Булева алгебра и конечные автоматы / И. Жоффрен. - М.: Мир, 1969.- С. 114 - 152.
33. Специализированные ЦВМ /под ред. В.Б. Смолова. -М.: Высшая школа, 1981. – 279 с.
34. Вавилов, Е.Н. Синтез схем электронных цифровых машин / Е.Н. Вавилов, Г.П. Портной. - М.: Сов. радио, 1963. – 440 с.

35. Мухопад, А.Ю. Структурный синтез автоматов управления системами обработки информации реального времени: автореф. дис. канд. техн. наук: / А.Ю. Мухопад. - Братск: БрГУ, 2010.-19с.
36. Попков В.К. Специализированные вычислительные среды / В.К. Попков, Ю.Ф. Мухопад. - Улан-Удэ: Бурятское книжное издательство, 1982. - 189 с.
37. Пупырев, Е.И. Перестраиваемые автоматы и микропроцессорные системы / Е.И. Пупырев. - М.: Наука, 1984. – 191 с.
38. Захаров, В.Н. Системы управления / В.Н. Захаров, Д.А. Поспелов, В.Е. Хазацкий. - М.: Энергия, 1972. – 344 с.
39. Закревский, А.Д. Алгоритмы синтеза дискретных автоматов / А.Д. Закревский. - М.: Наука, 1971. – 512 с.
40. Глушков, В.М. Теория автоматов / В.М. Глушков. - М.: Физматгиз, 1962. - 504с.
41. Гаврилов, М.А. Логическое проектирование дискретных автоматов / М.А. Гаврилов, В.В. Девятков, Е.И. Пупырев. - М.: Наука, 1977, - 363 с.
42. Горбатов, В.А. Теория частично упорядоченных систем / В.А. Горбатов. - М.: Сов. радио, 1976. – 336 с.
43. Сапожников, В.В. Теория дискретных устройств железнодорожной автоматики, телемеханики и связи / В.В. Сапожников, Вл.В. Сапожников, Ю.М. Кравцов. - М.: Транспорт, 2001. – 312 с.
44. Баранов, С.И. Синтез автоматов на элементах с матричной структурой / сб. Проектирование функционально ориентированных вычислительных систем / С.И. Баранов, В.А. Скляр, Н.Я. Янцен. – Л.: ЛГУ, 1990. – С. 90-108.
45. Рыцар Б.Е. Новый метод кодирования подфункций в задачах синтеза цифровых устройств на ПЛИС / Б.Е. Рыцар, А.Б. Кметь. // сб. Кибернетика и системный анализ, 2003. - №2. – С. 63-89.
46. Мухопад, Ю.Ф. Синтез автоматов управления по декомпозированной схеме алгоритма / Ю.Ф. Мухопад, Т.С. Бадмаева // сб. Информационные системы контроля и управления на транспорте. - Иркутск: ИрИИТ, 2002. – С. 14-25.

47. Колосов, В.Г. Проектирование узлов и систем автоматики и вычислительной техники / В.Г. Колосов, В.Ф. Мелехин. - Л.: Энергоатомиздат, 1983. – 255 с.
48. Кравцов, Л.Я. Проектирование микропрограммных устройств управления / Л.Я. Кравцов, Г.И. Черницкий. - Л.: Энергия, 1976.
49. Савельев, А.Г. Минимизация числа внутренних состояний МПА, выполненного с использованием ПЗУ / А.Г. Савельев // Сб. Автоматы и управление. М.: Наука Институт проблем передачи информации АН СССР, 1973. - С 15-26.
50. Тетерин, Ю.Н. О сокращении объема памяти микропрограмм / Ю.Н. Тетерин // Вопросы радиоэлектроники ЭВИ. - вып.- 4, 1983. - С. 35 - 39.
51. Палагин, А.В. Реализация МПА на ПЛИС /А.В. Палагин, А.А. Баркалов. С.И. Юсифов и др. - УСиМ, 1991.- № 8.- С.18 - 22.
52. Мухопад, Ю.Ф. Синтез МПА с ассоциативной памятью / Ю.Ф. Мухопад, П.А. Бабкин / /Сб. Микропроцессорные системы контроля и управления // Тр. Сибирск. научно-техн. конф.- Новосибирск: НЭТИ; Томск: ТИАСУР, 1992. - С. 90 - 99.
53. Баркалов, А.А. Синтез МПА на заказных программируемых СБИС устройств / А.А. Баркалов, Л.А. Титаренко. - Донецк: ДонНТУ; Технопарк УНИТЕХ, 2009. – 336 с.
54. Жинтелис, Г.Б. Автоматизация проектирования микропрограммируемых структур / Г.Б. Жинтелис, Э.К. Карчяускас, Э.К. Мачикенас. - Л.: Машиностроение, 1985. –216 с.
55. Wilkes. M.V. Microprogramming / M.V. Wilkes // Proc. East Joint. Comput.Conf. – Vol NT 114.- New York, 1959.- pp.18-20.
56. Майоров, С.А. Принципы организации ЦВМ / С.А. Майоров, Г.И. Новиков. - Л.: Энергия, 1974.- 434 с.
57. Самофалов, К.Г. Электронные вычислительные машины / К.Г. Самофалов, В.И. Корнейчук, В.П. Тарасенко. - Киев: Высшая школа, 1976. – 480 с.

58. Пийль, Е.И. Размещение микропрограмм в управляющей памяти /Автоматы и управление. Процессы и устройства управления в сетях связи / Е.И. Пийль. - М.: Наука, 1982. - С.20 - 30.
59. Валайтис, В.Ю. Методика и средства диалогового проектирования микропрограммного обеспечения. Архитектура и проектирование вычислительных систем / В.Ю. Валайтис, Э.К. Карчаускас. - Рига: Рижск. политехн. ин-т. им. А.Я. Пельше, 1985. - С.80-88.
60. Бузунов, Ю.А. Реализация неавтономных микропрограммных автоматов на программируемых логических матрицах / Ю.А Бузунов, Н.Н. Шипилов // Управляющие машины и системы.- № 6, 1980. - С.23 - 29.
61. Ожиганов А.А. Теория автоматов.// Учебн. Пособие –СПб: НИУТМО, 2013, 84 с.
62. Баранов, С.И. Матричная реализация управляющих автоматов / С.И. Баранов, Н.Я. Янцен // Теория дискретных управляющих устройств /ред. А.Д.Закревский, И.В. Прангишвили. - М.: Наука, 1982. - С. 57 - 70.
63. Карпов, Ю.Г. Теория автоматов / Ю.Г. Карпов. - СПб.: Питер, 2003. – 208 с.
64. Шалыто, А.А. SWITCH-технология. Алгоритмизация и программирование задач логического управления. СПб.: Наука. 1998, 628 с.
65. Синтез управляющих устройств в однородных средах. / ред В.И. Сифорова. - М.: Наука, 1984, - 166 с.
66. Валиев, Ш.К. Способ построения и реализации асинхронных конечных автоматов. /сб. Совершенствование и повышение надежности железнодорожных систем автоматики, телемеханики и связи. Ш.К. Валиев– Днепропетровск: ДнИИЖТ, 1985. с. 88 – 95.???
67. Гурвиц, Е.А. Синтез полисинхронных дискретных устройств / Е.А. Гурвиц - М.: Связь, 1969 – 172 с.
68. Бибило, П.Н. Синтез дискретных управляющих устройств на базе ПЛИС / П.Н. Бибило // Управляющие системы и машины, 1998. -ч.1 №2.- С. 69 - 75; ч.2.- там же № 3.- С. 48 - 58.

69. Антонов, А.П. Структурный синтез цифровых устройств / А.П. Антонов, В.Ф. Мелехин, Т.М. Митина. - СПб.: Изд. СПбГТУ, 1999. - 88 с.
70. Тимофеев, А.О. Проектирование автомата управления с предельным быстродействием / А.О. Тимофеев, Л.Г. Теницкий // Л.: Изв. ЛЭТИ, 1981. – вып. 291.- С. 47 – 51.
71. Юдитский, С.А. Логическое управление дискретными процессами / С.А. Юдитский, В.З. Магергут. - М.: Машиностроение, 1987. – 175 с.
72. Шалыто, А.А. Логическое управление. Методы аппаратной и программной реализации алгоритмов управления. / А.А. Шалыто. - СПб.: Наука, 2000. – 780 с.
73. Мухопад, Ю.Ф. Методы синтеза автоматов управления на БИС / Ю.Ф. Мухопад, А.Ю. Мухопад // Проблемы информатики. - Новосибирск: СО РАН, 2011. - № 4. - С.17-28.
74. Соловьев, А.В. Специальная математика / А.В. Соловьев. Пермь: ПГТУ, 2001. – С .41 - 54.
75. Лобов, О.Ф. Быстродействующий цифровой автомат / О.Ф Лобов, И.А. Гасников // Вопросы радиоэлектроники ЭВТ, 1989. - № 10.- С. 27 - 33.
76. Пат. 82888 Российская Федерация, МПК 51 G06F 9/00. Микропрограммный автомат / Мухопад А.Ю., Мухопад Ю.Ф. ; заявитель и патентообладатель Иркут. гос.ун - т путей сообщения. - 2008149344/22; заявл.15.12.2008; опубл. 10.05.2009, Бюл. № 13.- 2с.: ил.
77. Агибалов Г.П. Технология решения комбинаторно-логических задач методом сокращенного обхода дерева поиска /Г.П. Агибалов,В.А.. Беляев. – Томск: Изд-во ТГУ, 1981.- 125 с.
78. Закревский, А.Д. Логические основы проектирования дискретных устройств / А.Д. Закревский, Ю.В. Поттосин, Л.Д. Черемисинова. – М.: ФИЗМАТЛИТ, 2007. - 592 с.
79. Плотников, А.В. Микропрограммное устройство управления / А.В. Плотников, А.А. Баркатов, Стародубцев //Управляющие системы и машины. - № 4, 1987. - С.38 - 41.

80. Григорян, А.К. Метод декомпозиции конечных автоматов. / А.К. Григорян // Автоматика и телемеханика, 1968. - №10. - С. 17 - 23.
81. Левитин, М.Е. Оптимизация структуры дискретного автомата методом декомпозиции / М.Е. Левитин. - Тр. IFAC. Дискретные системы, Рига, 1974. - т.5. - С.54-160.
82. Бадмаева, Т.С. Синтез самоконтролируемой системы управления электроавтоматикой / Т.С. Бадмаева, Н.П. Деканова, Ю.Ф. Мухопад // Информационные технологии в энергетике, экономике, экологии. Иркутск: СЭИ СО РАН, 2003. - С.57-64.
83. Горбатов, А.В. Фундаментальные основы дискретной математики / А.В. Горбатов. - М.: Наука – Физматлит, 2000. - 544 с.
84. Горбатов А.В. Логическое управление распределенными системами / А.В. Горбатов, М.И. Смирнов, И.С. Хлытчиев. - М.: Энергоатомиздат, 1991. – 288 с.
85. Ачасова, С.М. Алгоритмы синтеза автоматов на ПЛМ / С.М. Ачасова - М.: Сов. радио, 1987. – 135 с.
86. Туманян, А.К. Построение управляющих автоматов на основе счетчиков / А.К. Туманян // В кн. Автоматика и вычислительная техника.- Вып.1. - Ереван, 1974. - С.13 - 16.
87. Чирков М.К., Пономарева А.Ю. Стационарные детерминированные и вероятностные автоматы. СПб. СПбГУ 2008г. 248 с
88. Вашкевич Н.П. Недетерминированные автоматы в проектировании систем параллельной обработки: учебное пособие. - Пенза: Изд-во Пенз.гос.ун-та, 2004. - 280 с.
89. Закревский А.Д. Параллельные алгоритмы логического управления. Изд.3. - 2012. 200 с.
90. Шамим Эхтер. Многоядерное программирование / Эхтер Шамим, Джейсон Робертс. – СПб: Питер, 2010. -316 с.



91. Варшавский, В.И. Автоматное управление асинхронными процессами в ЭВМ и дискретных системах / В.И. Варшавский, М.А. Кишиневский, В.Б. Мараховский. - М.: Наука, 1986. – 398 с.
92. Евреинов, Э.В. Цифровые автоматы с настраиваемой структурой / Э.В. Евреинов, И.В. Прангишвилли - М.: Энергия, 1974. – 240 с.
93. Барский А.Б. Логические нейронные сети / А.Б. Барский. – М.: Интернет-Университет Информационных технологий; БИНОМ. Лаборатория знаний, 2007. - 352с.
94. Галушкин А.И. Теория нейронных сетей / А.И. Галушкин. - М.: Радиотехника, 2000. - 415 с.
95. Зюзьков, В.М. Моделирование эпидемии с помощью клеточных автоматов / В.М. Зюзьков // Интеллектуальные автоматизированные системы проектирования и управления: под ред. В.П. Тарасенко. - Томск: Изд-во НТЛ, 2000. - С. 239 - 245.
96. Никитин, Г.А. О построении МПА одного класса / Г.А. Никитин, А.С. Боков // Специализированные и комбинированные вычислительные устройства.- Рязань: Ряз.радиотехн. ин - т, 1975. - С. 66 - 69.
97. Хадлстон, К. Проектирование интеллектуальных датчиков с помощью Microchip dsPIC / К. Хадлстон. - Киев: МК - Пресс, 2008.- 312 с.
98. А.с. 1547034 СССР. Устройство для контроля перепрограммируемых ПЗУ / Ю.Ф. Мухопад, Г.С. Скосырский // Бюл. - 1990. - №8.
99. А.с. 991587 СССР. Многоканальный формирователь временных последовательностей / Ю.Ф. Мухопад, Г.С. Скосырский, Ж.Ж. Халудоров и др. // Бюл. - 1983. -№ 3.
100. Угрюмов, Е.П. Цифровая схемотехника / Е.П. Угрюмов. - СПб.: БХВ-Петербург, 2010. – 760 с.
101. Тарасов, И.Е. Разработка цифровых устройств на основе ПЛИС Xilinx с применением языка VHDL / И.Е. Тарасов. - М.: 2-е изд., стереотип, 2015. – 252 с.
102. Врублевская, Е.В. Диагностирование неисправностей подвижного состава / Е.В. Врублевская // сб. Информационные технологии и проблемы

- математического моделирования сложных систем. – Иркутск: ИрГУПС, 2001. – вып.9.- С.26-30
103. Барашенков, В.В. Контроль синтаксической корректности операторных схем алгоритмов / В.В. Барашенков, О.Г. Кокаев, А.А. Гужавин // Вычислительная техника. - Л.: ЛГУ, 1977. – вып. 6 - С. 64 – 71.
104. Закревский, А.Д. Алгоритмический язык ЛЯПАС и автоматизация синтеза дискретных автоматов / А.Д. Закревский. - Томск: ТГУ, 1966.- С. 216.
105. Харлан, С.Н. Моделирование работы устройств автоматики средствами теории автоматов / С.Н. Харлан // Сб. Белоруск. Гос. ун-та, Гомель, 1995.- С. – 102 - 133.
106. Лазарев, В.Г. Построение программируемых управляющих устройств / В.Г. Лазарев, Е.И. Пийль, Е.Н. Турута. - М.: Энергоатомиздат, 1984. -193 с.
107. Булатова, И.Р. Автоматизированное проектирование устройств логического управления из программируемых матричных БИС / И.Р. Булатова, М.П. Друзина, А.В. Галковский и др. //сб. « Разработка и оптимизация САПР и ГАП изделий ЭТ и микро-ЭВМ»- Воронеж: ВПК, 1989. - С. 95 -97.
108. Решетников, Е.О. Инструментальное средство для визуального проектирования автоматных программ на основе Microsoft Domain-Specific Language Tools / Е.О. Решетников. - СПб: СПбГУ ИТМО, 2007.
109. Воевода, А.А. Разработка программного обеспечения с использованием UML диаграмм и сетей Петри на примере АСУ ТП водонапорной станции / А.А. Воевода, А.В. Марков, Д.О. Романников // Тр. СПИИ РАН, 2014. - Вып3(34). - С.218 - 232.
110. Марков, А.В. Применение UML диаграмм и сетей Петри для проектирования ПО технологического процесса / А.В. Марков // Сб. научн. тр. НГТУ №3(77). - Новосибирск: НГТУ, 2014. - С.99 - 118.
111. Шалыто, А.А. Автоматное проектирование программ. Алгоритмизация и программирование задач логического управления / А.А. Шалыто // Известия РАН. Теория и системы управления. - № 6, 2000.- С. 63 – 81.

112. Шалыто, А.А. Алгоритмизация и программирование для систем логического управления и "реактивных" систем / А.А.Шалыто //Автоматика и телемеханика . -№ 1, 2001. - С.3–39.
113. Туккель, Н.И. SWITCH-технология – автоматный подход к созданию программного обеспечения "реактивных" систем / Н.И. Туккель, А.А. Шалыто // Программирование. - № 5, 2001.- С. 45 – 62. Туккель Н.И. Преобразование итеративных алгоритмов в автоматные / Н.И. Туккель, А.А. Шалыто //Программирование.- № 5, 2002. -С.12-26.
114. Шалыто, А.А. Наумов Л.А. Методы объектно-ориентированной реализации реактивных агентов на основе конечных автоматов / А.А. Шалыто, Л.А. Наумов // Искусственный интеллект. - № 4, 2004- С.756–762. Шопырин Д.Г. Графическая нотация наследования автоматных классов / Д.Г. Шопырин, А.А. Шалыто // Программирование. -2007. С. 62 - 74.
115. Поликарпова, Н.И. Автоматное программирование / Н.И.Поликарпова, А.А. Шалыто.-СПб.:Питер, 200. - 176 с. Гуров В.С. Инструментальное средство для поддержки автоматного программирования / В.С. Гуров, М.А. Мазин, А.С. Нарвский, А.А. Шалыто // Программирование, 2009.-№6 . - С. 65 - 80.
116. Вельдер, С.Э. Верификация автоматных программ / С.Э. Вельдер, М.А. Лукин, А.А. Шалыто, Яминов. - СПб.: Наука, 2011. - 242 с.
117. Hamming, R.W. Error detecting and error correcting codes / R.W. Hamming. - Bell System Techn, 1950.-V.29, №2.- p.147 - 160.
118. Гаврилов, М.А. Сигнализация и прогнозирование отказов в дискретных управляющих устройствах со структурной избыточностью / М.А. Гаврилов // Тр. 2-го междунар. конгресса междунар. федерации по автоматическому управлению. - М.: Наука, 1965.
119. Гаврилов, М.А. Структурная избыточность и надежность работы релейных устройств / М.А. Гаврилов // Тр. 1 междунар. конгр. междунар. федерации по автоматическому управлению. М.: Изд-во АН СССР, 1967.- 3т. - С.105 - 112.

120. Сагалович, Ю.Л. Сокращение памяти автомата, устойчивого к повреждению и состязаниям его внутренних элементов / Ю.Л. Сагалович // Проблемы передачи информации. - т. 3, 1967. - вып.2.
121. Закревский, А.Д. Метод синтеза функционально устойчивых автоматов / А.Д. Закревский.- М.: ДАН СССР, 1969.-№4.- С. 381-386.
122. Францис, Т.А. Нижняя оценка выигрыша надежности при использовании кода Хэмминга в комбинационных автоматах / Т.А. Францис, Е.Н. Будзинаускене // Вопросы надежности дискретных автоматов. - Рига: Изд - во Зинатне, 1970. - С.3-16.
123. Tohma Y. Ohyama Sakajr R. Realization on Fail-Safe Sequential Machines by Using a k-out-of-N Code/. Tohma Y. Ohyama Sakajr R.- IEEE Transact on Comput, 1971. №11.- p.22 - 27.
124. Смолов, В.Б. Использование системы геометрических кодов в ПЗУ / В.Б. Смолов, Ю.Д. Чекмарев, Ю.Ф. Мухопад // Изв. Вузов.- сер. Приборостроение, 1971. - т.14. - № 6. - С.73 - 79.
125. Cook, G. Anderson D, Metze G. Design of totally self check Circuits for m-out-of-n Codes / G. Cook, D. Anderson, G/ Metze.- IEEE Transactonus.-1973.-Vc.22,№3.p. 255 - 263.
126. Сапожников, В.В. Синтез асинхронных конечных автоматов, устойчивых к повреждениям элементов логического преобразователя / В.В. Сапожников, Вл.В. Сапожников // Автоматика и телемеханика, 1972. - №8. -С.93 - 99.
127. Францис, Т.А. Избыточность в электронных дискретных устройствах / Т.А. Францис, Г.Ф. Янбых.- Л.: Энергия, 1969. 248.
128. Пархоменко, П.П. Основы технической диагностики / П.П. Пархоменко, Е.С. Согомоян.- М.: Энергоатомиздат, 1981. – 319 с.
129. Иыуду, К.А. Надежность, контроль и диагностика вычислительных машин и систем / К.А. Иыуду.- М.: Высшая школа, 1984. – 214 с.
130. Скосырский, Г.С. Особенности параметрического и функционального контроля БИС ПЗУ с плавающим затвором / Г.С. Скосырский // Проектирование

специализированных вычислителей и управляющих систем. - Иркутск: ИГУ, 1984. - С. 76 - 82.

131. Буинов, А.Н. Построение управляющих автоматов с безошибочным поведением / А.Н. Буинов // Проектирование специализированных вычислителей и управляющих устройств. - Иркутск: ИГУ, 1984. - С.3 - 9.

132. Балакин, В.Н. Синтез устройства диагностирования по схемам алгоритмов управления / В.Н. Балакин, В.В. Барашенков, Ю.Е. Усачев // Автоматика и телемеханика, 1984 .- №6. - С.138 - 144.

133. Тоценко, В.Г. Алгоритмы технического диагностирования дискретных устройств / В.Г. Тоценко. - М.: Радио и связь, 1985. – 238 с.

134. Сапожников, В.В. Методы синтеза надежных автоматов / В.В. Сапожников, Вл.В. Сапожников. - Л.: Энергия, 1980.- 93 с.

135. Согомоян, Е.С. Самопроверяемые устройства и отказоустойчивые системы / Е.С. Согомоян, Е.В. Слабаков. - М.: Радио и связь, 1989. – 208 с.

136. Щербаков, Н.С. Структурная теория аппаратного контроля цифровых автоматов / Н.С. Щербаков, Б.П. Подкопаев. - М.: Машиностроение, 1982. -191 с.

137. Щербаков, Н.С. Алгоритмы технического диагностирования дискретных устройств / Н.С.Щербаков, Б.П. Подкопаев.- М.: Радио и связь, 1990. – 111 с.

138. Балакин, В.Н. Синтез устройства диагностирования по схемам алгоритмов управления / В.Н. Балакин, В.В. Барашенков, Ю.Е. Усачев // Автоматика и телемеханика, 1984. - №6.- С.138 - 144.

139. А.с. СССР 1365986 Устройство для контроля блоков управления / В.Н. Балакин, В.В. Барашенков., А.Ф. Казак., С.А Никищенков, 1988. БИ №1.-10с.

140. Барашенков, В.В. Контроль выполнения программ по ярусно-параллельным алгоритмам / В.В. Барашенков, А.Ф.Казак, С.А Никищенков. -Л.: Известия ЛЭТИ, 1988. - Вып.394. - С.29 - 32.

141. А.с. СССР 1547034 Устройство для контроля перепрограммируемых ПЗУ / Ю.Ф. Мухопад, Г.С. Скосырский. - БИ №8, 1990.

142. Балакин, В.Н. Проектирование самопроверяемых управляющих устройств по тестопригодным схемам алгоритмов / В.Н., Балакин, В.В. Барашенков // Автоматика и телемеханика, 1988. - № 11. – С. 161 – 168.
143. Сапожников, В.В. Самодвойственные дискретные устройства / В.В. Сапожников, Вл.В. Сапожников, М. Гессель. - СПб.: Энергоатомиздат, 2001-331 с.
144. Сапожников, В.В. Основы технической диагностики / В.В. Сапожников, Вл.В. Сапожников. - М.:Маршрут, 2004. – 322 с.
145. Богомолов, А.М. Контроль и преобразования дискретных автоматов / А.М. Богомолов, И.С. Грунский, Д.В. Сперанский. - Киев: Наукова Думка, 1975. -174 с.
146. Барашенков, В.В. Контроль решающих процедур в автоматизированных системах проектирования специализированных цифровых вычислительных устройств / В.В. Барашенков, А.А. Гужавин // Вычислительная техника. - Пенза: Пенз. политехн. институт, 1975.- вып. 5.- С. 20 - 27.
147. Горяшко, А.П. Синтез диагностируемых систем вычислительных устройств / А.П. Горяшко. - М.: Наука, 1987. - 287 с.
148. Щербаков, Н.С. Достоверность работы цифровых устройств / Н.С. Щербаков.- М.: Машиностроение, 1989. – 224 с.
149. Проектирование тестируемых и самотестируемых матричных БИС / в кн. Е.Н. Файзулаева, И.И. Шатурина, А.П. Карамзинского и др.// Быстродействующие матричные БИС и СБИС.- М.: Радио и связь, 1989. - С. 250 - 254.
150. Феоктистов, В.П. Техническое диагностирование транспортных средств / В.П. Феоктистов, И.К. Лакин // Транспорт: Наука, техника, управление, 1997. - № 11. - С. 5 – 19.
151. Никищенков, С.А. Функциональная диагностика реконфигурируемых транспортных технологических систем по информационно - логическим схемам процессов / С.А. Никищенков. - Самара: СНЦ РАН, СамГАПС, 2005. – 159 с.
152. Вильнер, П.Ю. Метод диагностирования отказов сложных технических систем с использованием сетей Петри / П.Ю. Вильнер // Информационные и

- математические технологии в науке, технике и образовании. Тр. X Байкальской Всероссийской конф., ч. I.- Иркутск: ИСЭМ СОРАН, 2005. - С.239 - 335.
153. Пат. 63588 РФ. Самоконтролируемый автомат управления / А.Ю. Мухопад, Ю.Ф. Мухопад, Т.С. Бадмаева // Бюл.- 2007. - № 15. – с.
154. А.с. 1410101 СССР. Постоянное запоминающее устройство с контролем: / Ю.Ф. Мухопад, Ю.Д. Чекмарев. – Бюл. -1988.- № 26. – с.
155. Мухопад, Ю.Ф. Автоматная интерпретация устройств контроля микропроцессорных систем / Ю.Ф.Мухопад, Л.М. Сербуленко // Микропроцессорные системы контроля и управления: материалы Сиб. научн.-техн. конф.– Новосибирск: НЭТИ, 1992. – С. 41-49.
156. Прищепа, Л.С. Устройство для контроля дешифраторов / Л.С. Прищепа, В.А. Илюшкин // Интеллектуальные системы в управлении, конструировании и образовании. - Томск: ТГУСУР, 2002. – С. 160-163.
157. Мухопад, Ю.Ф. Встроенный контроль в автоматах управления / Ю.Ф. Мухопад, А.Ю. Мухопад, Т.С. Бадмаева // Вестник Иркутского государственного технического университета. – Иркутск: ИрГТУ, 2006. - № 2. – С. 148-150.
158. Белоглазов, И.Н. Корреляционно-экстремальные системы / И.Н. Белоглазов, В.П. Тарасенко. - М.: Сов. радио, 1974. - 267 с.
159. Пат. 1107135 РФ G06 F 15/336. Цифровой коррелятор для обнаружения эхо-сигналов / В.Е. Бочаров, А.В. Майструк, В.А. Гудым и др. // Бюл, 1984.- № 29. – 7 с.
160. Странгуль, О. Н. Корреляционно-экстремальные системы навигации и локации подвижных объектов / О. Н. Странгуль, В. П. Тарасенко // Автоматика и телемеханика, 2001. - вып. 7. - С. 201–210
161. Баклицкий, В.К. Корреляционно-экстремальные методы навигации наведения / В.К. Баклицкий. - Тверь: ТО Книжный клуб, 2009.- 365 с.
162. Пат. 2103707 РФ G01S17/00 Корреляционно-экстремальный координатор цели / В.А. Кочкин // БИ, 1998. - №11.

163. Пат. 2528140 РФ. G06K9/48, G06T7/40 Способ автоматического распознавания объектов на изображении / В.А. Ефремов, В.А. Тупиков, Л.В. Московченко и др. // БИ, 2014. - № 25.- 25 с.
164. Пат. 2525601 РФ. G01C23/00 Комплексная корреляционно-экстремальная навигационная система / Г.И. Джанджгава, О.Г. Лещук, Т.В. Сазонова и др. // БИ, 2014. - № 23.
165. Белоглазов И.Н. Основы навигации по физическим полям / И.Н. Белоглазов, Г.И. Джанджгава, Г.П. Чигин. - М.: Наука, 1985.
166. Барабанова, А.М. Результаты распознавания объектов с помощью моделей их фрактальных образов с борта самолета / А.М. Барабанова, Н.В. Усков //Российская научн-техн.конф. «Информатика и проблемы телекоммуникаций». т.1. – Новосибирск: СибГУТИ, 2006. – С.322 -325.
167. Кузнецов, В.А. Алгоритмы автоматизированного обнаружения и распознавания наземных объектов по их радиолокационным изображениям в реальном масштабе времени: автореф.дис. ... канд.техн. наук: / В.А. Кузнецов. – Иркутск, ИрГУПС, 2013. – 24 с.
168. Миронов, Б.М. Алгоритм сегментации изображений подстилающей поверхности когерентного локатора /Б.М. Миронов, Б.В. Лежанкин. С.В. Галиев // Люминесценция и лазерная физика. Тр. межд. школы-семинара. – Иркутск: ИрГУ, 2007. –С. 236-248.
169. Frost V.S. A Data compression technique for synthetic aperture radar images /Frost V.S., Minden G.A. // IEEE Trans. v.AES -22, N 1, Jan, 1986. –р. 47-54
170. Ильющенко, В.В. Автоматизированная система управления насосной станцией на основе отказоустойчивого оборудования: автореф. дисс. ...канд. техн. наук / В.В. Ильющенко. - Иркутск: ИрГУПС, 2011.- 24 с.
171. Левин, М. Криптография без секретов (руководство пользователя) / М. Левин. - М.:ЗАО Новый издательский дом, 2005.-315 с.
172. Чмора, А. Современная прикладная криптография / А. Чмора. - М.: Гелиос АРВ, 2001.- 244 с.



173. Глухов, М.Н., Круглов И.А., Пичкур А.Б., Черемушкин А.В. Введение в теоретико-числовые методы криптографии. М.: Лань, 2011.-400с.
174. Ишмухаметов Ш.Т., Рубцова Р.Г. / Математические основы защиты информации. Казань: КФУ. – 2012,- 135 с.
175. Зайцев, А.П. Технические средства и методы защиты информации / А.П. Зайцев, А.А. Шелупанов, Р.В. Мещеряков и др. – М.: Машиностроение, 2009. - 508с.
176. Самит, В.Н. Криптографические методы и средства защиты информации / В.Н. Самит. - Саратов: СГУ, 2013. – 42 с.
177. Шаньгин, В.Ф. Информационная безопасность компьютерных систем и сетей / В.Ф Шаньгин. - М.: ИД «Форум» - ИНФРА - М, 2008.- 415 с.
178. Фергюсон, Н. Практическая криптография / Н. Фергюсон, Б. Шнайдер.- М.: Изд-во Вильямс, 2005.- 424 с.
179. Пат. на полезн. модель 82974 РФ. H04L 9/00/ Устройство криптографической защиты информации / А.Ю. Мухопад, Ю.Ф. Мухопад, Т.Б. Агафонов // Бюл. - 2009. -№ 13.-23 с.
180. Пат. 82889 РФ. G06 12/16 .Устройство криптографической защиты информации / А.Ю. Мухопад, Ю.Ф. Мухопад, Б.Н. Антошкин // Бюл. -2009. -№ 13.- 22 с.
181. Пат. 82890 РФ. G06F 12/16. Устройство криптографической защиты информации / А.Ю. Мухопад, Б.Н. Антошкин, Ю.Ф. Мухопад // Бюл. – 2009. - № 13.-23 с.
182. Пат. 2475838 РФ. G06F 21/00, H04L 9/00 Устройство криптографической защиты информации / А.Ю. Мухопад, Ю.Ф. Мухопад // Бюл. - 2013. - №5.-16с.: 5ил.
183. Васильев, В.А. Новые подходы к разработке цифрового ПИД регулятора /В.А. Васильев, А.А. Воевода, В.А. Жмудь // сб. научн. тр.- Новосибирск: НГТУ. - № 3(45), 2006. - С.11-18.

184. Колтыгин, Д.С. Технические и программные средства робототехнического комплекса / Д.С. Колтыгин, И.А. Седельников. - Братск: БрГУ, 2014.- 110 с.
185. Мухопад, Ю.Ф. Микроэлектронные системы управления роботами-манипуляторами / Ю.Ф. Мухопад. - Иркутск: ИГУ, 1984.- 124 с.
186. Колтыгин, Д.С. Введение в роботехнику. Цикловое управление манипуляторами и технологическим оборудованием / Д.С. Колтыгин, С.Г. Рудученко. - Братск: БрГУ, 2002.- 234 с.
187. Ковыршин, С.В. Моделирование пневмосистем в среде Flaidsim / С.В. Ковыршин, Н.С. Сегедин. - Иркутск: ИрГУПС, 2011. – 47 с.
188. Колтыгин, Д.С. Пневмоавтоматика / Д.С. Колтыгин. - Братск: БрГУ, 2012.- 142 с.
189. Пашков, Е.В. Электропнеumo-автоматика в производственных процессах / Е.В. Пашков, Ю.А. Осинский, А.А. Четверкин. - Севастополь: СевНТУ, 2013.- 496 с.

#### Дополнительная литература

190. Теория дискретных управляющих устройств / под ред. А.Д. Закревского, И.В. Прангишвили. - М.: Наука, 1982. – 251 с.
191. Дьяченко, В.Ф. О методе перехода от ЛСА к булевым функциям / В.Ф. Дьяченко // Изв. АН СССР. Техническая кибернетика, 1963. - №6.
192. Дьяченко, В.Ф. Управление на сетях связи / В.Ф. Дьяченко, В.Г. Лазарев. - М. : Сов.радио, 1967.- 224 с.
193. Захаров, В.Н. Автоматы с распределенной памятью / В.Н. Захаров. - М.: Энергия, 1975. – 136 с.
194. Горбатов, В.А. Семантическая теория проектирования автоматов / В.А. Горбатов. - М.: Энергия, 1979. – 264 с.
195. Лупал А.М. Теория автоматов. СПб.: ГУАП. – 2000, - 119 с.

196. Лазарев, В.Г. Синтез асинхронных конечных автоматов / В.Г. Лазарев, Е.И. Пийль. - М.: Энергия, 1970. – 400 с.
197. Чулков В.А. Интерполирующие устройства синхронизации и преобразователи информации: монография / Чулков В.А., Москва: Физматлит, 2010, -323 с.
198. Дабаткин, Н.В. Алгоритм расчета характеристик верности передачи информации в сети ЭВМ / Н.В. Дабаткин // Тр. Всесоюзн. семинара «Разработка и оптимизация САПР и ГАП изделий электронной техники новых высокопроизводительных мини и микро ЭВМ.- Воронеж: ВПИ, 1989. -С. 145 - 149.
199. Сагалович, Ю.Л. Кодирование состояний автомата. / Теория дискретных управляющих устройств / Ю.Л. Сагалович - М.: Наука, 1982. С. 175 – 182.???
200. Постников, А.И. Основы теории цифровых автоматов / А.И. Постников. – Красноярск : КрГТУ, 1999. – 251 с.
201. Кирий, В.Г. Теория автоматов / В.Г. Кирий. - Иркутск: ИрГТУ, 2007. – 144 с.
202. Карцев, М.А. Вычислительные системы и синхронная арифметика / М.А. Карцев, В.А. Брик. - М.: Радио и связь, 1981. – 359 с.
203. Овчаренко, Н.И. Цифровые аппаратные и программные элементы микропроцессорной релейной защиты и автоматики энергосистем / Н.И. Овчаренко. - М.: НТФ «Энергопрогресс». «Энергетик», 2006. – 120 с.
204. Басс ,Э.И. Релейная защита электроэнергетических систем / Э.И. Басс, В.Г. Дорогунцев / ред. Дьяков А.Ф. - М.: МЭИ, 2002. – 187 с.
205. Балашов, Е.П. Информационные системы. Табличная обработка информации / Е.П. Балашов, В.Б. Смолков. - Л.: Энергоатомиздат, 1985. – 179 с.
206. Калабеков, Б.А. Цифровые устройства и микропроцессорные системы / Б.А. Калабеков . - М.: Горячая линия -Телеком, 2007 или 2003. – 338 с.

207. Грушвицкий, Р.И. Проектирование систем на микросхемах программируемой логики / Р.И. Грушвицкий, А.Х. Мурсаев, Е.П. Угрюмов - СПб.: БХВ – Петербург, 2002. – 608 с.
208. Бродин, В.Б. Системы на микроконтроллерах и БИС программируемой логики / В.Б. Бродин, А.В. Калинин. - М.: Эком, 2002. – 400 с.
209. Агасандян, Г.А. Разложение автомата на управляющий и управляемый подавтоматы / Г.А. Агасандян, В. Вехлер // Изв. АН СССР. Техническая кибернетика, 1976. - № 6.
210. Бузунов, Ю.А. Метод двухкоординатного кодирования микрокоманд / Ю.А. Бузунов // Автоматика и вычислительная техника, 1976. - № 5.
211. Евлашкин, А.А. Применение алгоритма коллективной оптимизации в организации сетевых структур / А.А. Евлашкин // Тр. конф. «Проблемы, решения, инновации транспорта РФ». - Иркутск: ИрГУПС, 2010. - С. 232 - 237.
212. Кодачиков, В.И. Электронная коммутация информационных каналов / В.И. Кодачиков. - Ростов: Рост.ун - т, 1983. – 207 с.
213. А.с. 1087996 СССР. Устройство для программного управления с применением коммутаторов / Ю.Ф. Мухопад, Т.С. Бадмаева //Бюл. - 1984. - № 15.
214. Буч, Г. Язык UML. Руководство пользователя / Г. Буч, Дж. Рамбо, А. Джекобсон. - СПб: Питер, 2004. – 430 с.
215. Пийль, Е.И. Упрощенный алгоритм получения группы связанных состояний. Принципы построения устройств распределения информации / Е.И. Пийль //Автоматы и управление.- М.: Наука, 1978.- С.113-116.
216. Патенты РФ № 2141729, 2172075, 2188513, 2206120, 2206182, 2211541, 2212108 по разделу 7H04L9/16.
217. Городецкий, В.И. Элементы теории испытаний и контроля технических систем / В.И. Городецкий, А.К. Дмитриев, и др..- Л.: Энергия, 1978. -192 с.
218. В.В. Сапожников, Методы построения безопасных микроэлектронных систем железнодорожной автоматики / В.В. Сапожников, Вл.В. Сапожников, Х.А. Христов и др. –М. Транспорт, 1995 .с- 272 с.

219. Огнев, И.В. Надежность запоминающих устройств / И.В. Огнев, К.Ф. Сарычев. – М.: Радио и связь, 1988. – 224 с.
220. Водяхо, А.Н. Функционально ориентированные процессоры / А.И. Водяхо, В.Б. Смолков, В.У. Плюснин и др. – Л.: Машиностроение, 1988 – 224 с.
221. Шишкин, Г.И. Обеспечение помехоустойчивости цифровых систем / Г.И. Шишкин / Часть 1. – Саров: ФУП «РФЯЦ-ВНИИЭФ», 2004. – 436 с.
222. Артюхов, В.Л. Настраиваемые модули для управляющих логических устройств / В.Л. Артюхов, Г.А. Копейкин, А.А. Шалыто. – Л.: Энергоиздат. 1981. – 168 с.
223. Иванова, Г.С. Технология программирования / Г.С. Иванова. – М.: КНОРУС, 2011. – 336 с.

#### Публикации автора

#### **Монография**

Мухопад, А.Ю. Теория управляющих автоматов технических систем реального времени / А.Ю. Мухопад. – Новосибирск: Наука, 2015. – 176с.

#### **Список публикаций в журналах из списка ВАК**

1. Мухопад, А.Ю. Контроль и диагностика автоматов управления / А.Ю. Мухопад, Т.С. Бадмаева и др. // Журнал «Современные технологии. Системный анализ. Моделирование». – Иркутск: ИрГУПС, 2005. – Вып.5. – С.78-81.
2. Мухопад, А.Ю. Встроенный контроль в автоматах управления / А.Ю. Мухопад, Т.С. Бадмаева // Вестник Иркутского гос. технического университета. – Иркутск: ИрГТУ. – 2006. – № 2. – С. 148-150.
3. Мухопад, А.Ю. Динамический контроль автоматов / А.Ю. Мухопад. – Новосибирск: Сборник научных трудов НГТУ. – № 3. – 2008. – С. 55 – 58.
4. Мухопад, А.Ю. Организация микроэлектронных средств управления сложными техническими системами / А.Ю. Мухопад, Ю.Ф. Мухопад / Журнал «Современные проблемы науки и образования». – Москва: ИД «Академия естествознания» - № 6, 2009. – С. 30-32.
5. Мухопад, А.Ю. Синтез автоматов управления для систем реального времени / А.Ю. Мухопад, А.Ф. Полетаев и др. // Журнал «Современные

- технологии. Системный анализ. Моделирование». - Иркутск: ИрГУПС, 2010. № 3. – С 161-168.
6. Мухопад, А.Ю. Структурные методы повышения надежности автоматов управления / А.Ю. Мухопад // Системы. Методы. Технологии. – Братск: БГУ. - № 3(11), 2011. – С.95 -100.
7. Мухопад А.Ю. Структурная организация самоконтролируемых автоматов для систем реального времени / А.Ю. Мухопад, Ю.Ф. Мухопад // Проблемы информатики.- Новосибирск: ИВМиМГ СО РАН, 2013. - №1. - С.4-15.
8. Мухопад А.Ю. Анализ мехатронных систем ударного действия / А.Ю. Мухопад, А.В. Мельников и др. // Фундаментальные исследования РАЕ, № 8,ч.6.- Москва, 2013.- С.1337-1343.
9. Мухопад А.Ю. Использование теплогенераторов в вагонном депо / В.Ц. Ванчиков, А.Ю. Мухопад и др. // Журнал «Современные технологии. Системный анализ. Моделирование». - №3 (39). - Иркутск: ИрГУПС, 2013.- С. 267-369.
10. Мухопад, А.Ю. Моделирование источника электромагнитного излучения / Р.А. Данеев, А.Ю. Мухопад и др. // Журнал «Современные технологии. Системный анализ. Моделирование». – Иркутск: ИрГУПС, № 4, 2013. – С. 146-151.
11. Мухопад, А.Ю. Ассоциативный автомат адаптивного управления технологическими процессами на основе нейронных сетей / В.Н. Сизых, А.Ю. Мухопад // Научный вестник НГТУ. – Новосибирск.- № 1 (54), 2014. - С. 34-45.
12. Мухопад, А.Ю. Автоматизация управления системой очистки технологических сред с применением вихревого эффекта и ультразвука / Д.А. Филатов, А.Ю. Мухопад и др. // Журнал «Современные технологии. Системный анализ. Моделирование» Иркутск: ИрГУПС, 2015.-№ 3(47). - С.64-72.
13. Мухопад, А.Ю. Микропрограммное управление турбомеханизмами / Д.А. Филатов, А.Ю. Мухопад // Журнал «Современные технологии. Системный анализ. Моделирование» Иркутск: ИрГУПС, 2015.-№3 (47). - С.78-89.
14. Мухопад, А.Ю. Сравнительный анализ комбинационных схем управляющих автоматов / А.Ю. Мухопад // Журнал «Современные технологии. Системный анализ. Моделирование» Иркутск: ИрГУПС, 2015.-№4 (48). - С.106-112.

#### **Научные статьи, доклады на конференциях и др. публикации**

15. Мухопад, А.Ю. Статистическая обработка данных о нестационарных процессах / А.Ю. Мухопад, Ю.Ф. Мухопад // Сб. «Асимптотические методы в

задачах проектирования летательных аппаратов». – Иркутск: ИрГТУ, 1996. – С. 45 – 47.

16. Мухопад, А.Ю. Динамически модифицируемые матрицы для кодирования информации / А.Ю. Мухопад, Т.Б. Агафонов и др. // Сб. науч. трудов «Информационные системы контроля и управления на транспорте». – Иркутск: ИрГУПС, 2001.- Вып 9.- С. 159-161.

17. Мухопад, А.Ю. Таблично-алгоритмический кодер / А.Ю. Мухопад, Т.Б. Агафонов и др. // Сб. науч. трудов «Информационные системы контроля и управления на транспорте». - Иркутск: ИрГУПС, 2002. – Вып. 10.- С. 171-175.

18. Мухопад, А.Ю. Структурная организация автоматов с контролем / А.Ю. Мухопад //Сб. науч. трудов «Информационные системы контроля и управления на транспорте». - Иркутск: ИрГУПС, 2005. - Вып. 13. – С. 75 – 78.

19. Мухопад, А.Ю. Структурная организация программной модели сложных автоматов управления мехатроникой / А.Ю. Мухопад // Труды IV Междунар. конференции «Проблемы механики современных машин». – Улан-Удэ: ВСГТУ, 2009. - том 3. – С. 88 – 94.

20. Мухопад, А.Ю. Структурная организация программной модели сложных автоматов систем реального времени / А.Ю. Мухопад //Труды XXII Международной конференции «Математические методы в технике и технологиях». – Псков: ППИ, 2009. - том 2. – С. 83 – 86.

21. Мухопад, А.Ю. Синтез сложных автоматов управления мехатроникой / А.Ю. Мухопад //Труды VII Всероссийской научно-практической конференции студентов, аспирантов и молодых ученых «Молодежь и современные информационные технологии». Томск: ТПУ, 2009. - ч. 2. – С. 70 – 71.

22. Мухопад, А.Ю. Динамический контроль микропрограммных автоматов / А.Ю. Мухопад // Сб. науч. трудов «Информационные системы контроля и управления в промышленности и на транспорте». – Иркутск: ИрГУПС, 2009. - Вып. 16. - С. 78 - 82.

23. Мухопад, А.Ю. Обеспечение надежности сложных автоматов управления мехатроникой / А.Ю. Мухопад // Труды Международной научной школы «Фундаментальные и прикладные проблемы надежности и диагностики машин и механизмов». – СПб: Научный совет РАН по проблемам машиноведения и технологических процессов ИПМаш РАН, 2009. – С. 17 – 19.

24. Мухопад, А.Ю. Методы повышения безотказности электронного комплекса самолетов / А.Ю. Мухопад, М.П. Дунаев и др. //«Актуальные проблемы развития гражданской авиации России». – Иркутск, Москва: МГТУГА, 2009.- С. 78-83.

25. Мухопад, А.Ю. Минимизация кодового представления переходов в сложных автоматах / А.Ю. Мухопад в кн. Ю.Ф. Мухопада «Микроэлектронные системы управления». – Братск: БГУ, 2009. - С. 171 - 178.
26. Мухопад, А.Ю. Контроль функционирования автоматов управления / А.Ю. Мухопад в кн. Ю.Ф. Мухопада «Микроэлектронные системы управления. - Братск: БГУ, 2009. – С. 185- 193.
27. Mukhopad A.Yr. Microelectronic controlling of realtime complicated technical systems / A.Yr. Mukhopad, Yr. F. Mukhopad //International journal of applied and fundamental research (JSSN 1996- 3955). - № 2, 2009. - p. 26 – 29.
29. Мухопад, А.Ю. Обеспечение надежности сложных автоматов управления мехатроникой. Тр. Международной научной школы «фундаментальные и прикладные проблемы надежности и диагностики машин и механизмов». – СПб: Научный совет РАН по проблемам машиноведения и технологических процессов, ИПМашРАН, 2009. – С. 17 – 19.
30. Мухопад, А.Ю. Системный анализ и моделирование автоматов управления систем обработки информации реального времени / А.Ю. Мухопад, А.Ф. Полетаев и др. // Сб. «Системный анализ в проектировании и управлении» ч.2 – СПбПУ, 2010.- С 151-159.
31. Мухопад, А.Ю. Метод динамического контроля автоматов управления / А.Ю. Мухопад // Сб. науч. трудов «Информационные системы контроля и управления в промышленности и на транспорте». – Иркутск: ИрГУПС, 2010. - Вып. 17. - С. 139-142.
32. Мухопад, А.Ю. Структурный синтез автоматов управления системами обработки информации реального времени: автореф. дис. канд. техн. наук / А.Ю. Мухопад.- Братск, 2010.-19с.
33. Мухопад, А.Ю. Синтез быстродействующих автоматов / А.Ю. Мухопад в кн. Ю.Ф. Мухопада «Теория дискретных устройств» - Иркутск: ИрГУПС, 2010. – С. 102-111.
34. Мухопад, А.Ю. Моделирование автоматов управления / А.Ю. Мухопад в кн. Ю.Ф. Мухопада «Теория дискретных устройств» - Иркутск: ИрГУПС, 2010. – С 150-155.
35. Мухопад, А.Ю. Методы синтеза автоматов управления на больших интегральных схемах / А.Ю. Мухопад, Ю.Ф. Мухопад // Проблемы информатики Новосибирск: ИВМиМГ СО РАН, №4, 2011.- С.17- 28.
36. Мухопад, А.Ю. Методы повышения надежности автоматов управления / А.Ю. Мухопад // Сб. научн. трудов « Информационные системы контроля и управления в промышленности и на транспорте». - Иркутск: ИрГУПС, 2011.- Вып.19.- С. 111-117.



37. Мухопад, А.Ю. Система управления коммутацией сигналов / А.Ю. Мухопад, Т.С. Бадмаева и др. // Сб. научн. трудов «Информационные системы контроля и управления в промышленности и на транспорте». – Иркутск: ИрГУПС. - Вып.19, 2011.- С. 101-110.
38. Мухопад, А.Ю. Аппаратная реализация микропрограммных средств защиты информации / А.Ю. Мухопад, В.С. Жигунов и др. //Труды международной конференции «Транспорт XXI века: исследования, инновации, инфраструктура».- Екатеринбург: Уральский гос.университет путей сообщения.- Вып.97(180) Том 1, 2011.- С.17-21.
39. Мухопад, А.Ю. Методы повышения надежности автоматов управления / А.Ю. Мухопад //Проблемы Транспорта Восточной Сибири: материалы научно-практической конференции молодых ученых, аспирантов и студентов электромеханического факультета, ч.2.- Иркутск: ИрГУПС, 2012.- С.90-95.
40. Мухопад, А.Ю. Структурная организация самоконтролируемых автоматов для систем реального времени / А.Ю. Мухопад // Проблемы информатики.- Новосибирск: ИВМиМГ СО РАН, 2013. №1.- С.4-15.
41. Мухопад, А.Ю. Методика разработки программного обеспечения микропроцессорных информационно-управляющих систем / А.Ю. Мухопад // Информационные системы контроля и управления в промышленности и на транспорте. - Иркутск: ИрГУПС, 2013.- Вып.23.- С. 107-120.
42. Мухопад, А.Ю. Автоматный контроль оборудования авиационной техники / А.Ю. Мухопад // Сб. тр. IV научно-практической конференции «Актуальные проблемы и перспективы развития гражданской авиации России». - Иркутск: Ирк. филиал МГТУГА, 2014.- С.91-95.
43. Мухопад, А.Ю. Адаптивное управление одним классом абсолютно устойчивых систем / В.Н. Сизых, А.Ю. Мухопад и др. // Сборник Всероссийской научно-практической конференции «Авиатор» Актуальные вопросы исследований в авионике: Теория, обслуживание, разработки.- Воронеж: Военно-воздушная академия им. проф. Н.Е.Жуковского и Ю.А. Гагарина, 2014.- С.144-145.
44. Мухопад, А.Ю. Автоматизированное программирование микроконтроллерных средств управления технологическими процессами и сложными техническими системами / А.Ю. Мухопад, Е. А. Безрукова // Информационные системы контроля и управления в промышленности и на транспорте. - Иркутск: ИрГУПС, 2014.- Вып.24.- С. 5-23.
45. Мухопад, А.Ю. Автоматическое управление турбо-механизмами / А.Ю. Мухопад, Д.А. Филатов // Сб. научн. трудов II Международной научно-

практической конференции «Проблемы и достижения в науке и технике». - Омск: ИЦРОН (IDCES), 2015.-С.13-22.

46. Мухопад А.Ю. Управление комбинаторно-нейронной сетью / А.Ю. Мухопад, В.Н. Сизых // Информационные системы контроля и управления в промышленности и на транспорте. - Иркутск: ИрГУПС, 2015.- Вып.25.- С. 11-17

#### **Авторские свидетельства, патенты.**

47. Патент на изобретение № 2475816 Российская Федерация МПК(51) G06F 9/00. Управляющий автомат / Мухопад А.Ю., Мухопад Ю.Ф.; заявитель и патентообладатель Иркут. гос.ун-т путей сообщения. - № 2011145137/08; заявл. 07.11.2011; опубл. 20.02.2013, Бюл. №5.-13с.: 5ил.

48. Патент на изобретение № 2475838 Российская Федерация МПК(51) G06F 21/00, H04L 9/00. Устройство криптографической защиты информации / Мухопад А.Ю., Мухопад Ю.Ф.; заявитель и патентообладатель Иркут. гос.ун-т путей сообщения.-№ 2011145012/08; заявл. 07.11.2011; опубл. 20.02.2013, Бюл. №5.-16с.: 5ил.

49. Патент на изобретение № 2527190 Российская Федерация МПК(51) G06F 9/00. Микропрограммный автомат /Мухопад А.Ю., Пунсык - Намжилов Д.Ц. и др.; заявитель и патентообладатель Иркут. гос.ун-т путей сообщения.- № 2013110986/08; заявл. 12.03.2013; опубл. 27.08.2014, Бюл. № 24. -20 с.: 12 ил.

50. Патент на изобретение № 2502121 Российская Федерация МПК (51) G06F 9/22, G06F 11/00. Самоконтролируемый автомат / Мухопад А.Ю., Мухопад Ю.Ф.; заявитель и патентообладатель Иркут. гос.ун-т путей сообщения № 2011148883/08; заявл. 30.11.2011; опубл. 20.12.2013, Бюл. № 35.- 29с. : 15ил.

51. Патент на полезную модель № 63588. Самоконтролируемый автомат управления / Мухопад А.Ю., Бадмаева Т.С. и др.; заявитель и патентообладатель Иркут. гос.ун-т путей сообщения.- № 2006143930/22; заявл. 11.12.2006; опубл. 27.05.2007, Бюл. №15.- 10 с.: ил.

52. Патент на полезную модель № 82889 МПК(51) G06 12/16. Устройство криптографической защиты информации / Мухопад А.Ю., Антошкин Б.Н., и др.; заявитель и патентообладатель Иркут. гос.ун-т путей сообщения.- № 2008150879/22; заявл. 22.12.2008; опубл. 10.05.2009, Бюл. № 13.- 12с.: ил.

53. Патент на полезную модель № 82890.МПК (51) G06F 12/16. Устройство криптографической защиты информации / Мухопад А.Ю., Антошкин Б.Н. и др.; заявитель и патентообладатель Иркут. гос.ун-т путей сообщения.-2008150881/22; заявл.22.12.2008; опубл. 10.05.2009, Бюл.№ 13.- 2с.:

54. Патент на полезную модель № 82888. МПК (51) G06F 9/00. Микропрограммный автомат / Мухопад А.Ю., Мухопад Ю.Ф.; заявитель и патентообладатель Иркут. гос.ун-т путей сообщения.- 2008149344/22; заявл.15.12.2008; опубл. 10.05.2009, Бюл.№ 13.- 2с.: 1 ил.
55. Патент на полезную модель № 82974. МПК(51) H04L 9/00. Устройство криптографической защиты информации / Мухопад А.Ю., Агафонов Т.Б. и др.; заявитель и патентообладатель Иркут. гос.ун-т путей сообщения.- 2008149331/22; заявл. 15.12.2008; опубл. 10.05.2009, Бюл. № 13.- 23с. : ил.
56. Патент на полезную модель № 111760. МПК(51) A61N 9/00, A61N 2/02. Медицинский прибор «Мультимед» / Хомяков Г.К., Мухопад А.Ю. и др.; заявитель и патентообладатель ООО «Байкальский научно-инженерный центр» - № 2010128849/14; заявл. 12.07.2010; опубл. 27.12.2011, Бюл. № 36.-2с.: 1ил.
57. Патент на полезную модель №158425 РФ МПК (51) H04N 19/00. Устройство классификации изображений / Мухопад А.Ю., Марюхненко В.С., Миронов Б.М. заявитель и патентообладатель Иркут. гос.ун-т путей сообщения.- №2015111830/08; заявл. 01.04.2015; опубл. 27.12.2015 Бюл.№36
58. Свидетельство о государственной регистрации программы для ЭВМ № 2015615404 от 18.05.2015. Программа управляющего автомата автоматизированной системы очистки технологических сред с применением вихревого эффекта и ультразвука / Филатов Д.А., Мухопад А.Ю. и др.
59. Свидетельство о регистрации электронного ресурса №15737 от 11.05. 2010. Алгоритм моделирования автомата в системе управления операционным устройством / Мухопад А.Ю., Полетаев А.Ф. и др.

## ПРИЛОЖЕНИЯ

РОССИЙСКАЯ ФЕДЕРАЦИЯ

**ПАТЕНТ**

НА ИЗОБРЕТЕНИЕ

№ 2502121

**САМОКОНТРОЛИРУЕМЫЙ АВТОМАТ**

Патентообладатель(ли): *Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования Иркутский государственный университет путей сообщения (ФГБОУ ВПО ИргУПС) (RU)*

Автор(ы): *с.м. на обороте*

Заявка № 2011148883

Приоритет изобретения **30 ноября 2011 г.**

Зарегистрировано в Государственном реестре изобретений Российской Федерации **20 декабря 2013 г.**

Срок действия патента истекает **30 ноября 2031 г.**

*Руководитель Федеральной службы по интеллектуальной собственности*

*Б.П. Симонов*



## РОССИЙСКАЯ ФЕДЕРАЦИЯ



## ПАТЕНТ

НА ИЗОБРЕТЕНИЕ

№ 2527190

## МИКРОПРОГРАММНЫЙ АВТОМАТ

Патентообладатель(ли): *Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования Иркутский государственный университет путей сообщения (ФГБОУ ВПО ИрГУПС) (RU)*

Автор(ы): *с.м. на обороте*

Заявка № 2013110986

Приоритет изобретения 12 марта 2013 г.

Зарегистрировано в Государственном реестре изобретений Российской Федерации 07 июля 2014 г.

Срок действия патента истекает 12 марта 2033 г.

*И.о. руководителя Федеральной службы по интеллектуальной собственности*

*Л.Л. Кирий*



РОССИЙСКАЯ ФЕДЕРАЦИЯ



ПАТЕНТ

НА ИЗОБРЕТЕНИЕ

№ 2475816

УПРАВЛЯЮЩИЙ АВТОМАТ

Патентообладатель(ли): *Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования Иркутский государственный университет путей сообщения (ФГБОУ ВПО ИРГУПС) (RU)*

Автор(ы): *с.м. на обороте*

Заявка № 2011145137

Приоритет изобретения 07 ноября 2011 г.

Зарегистрировано в Государственном реестре изобретений Российской Федерации 20 февраля 2013 г.

Срок действия патента истекает 07 ноября 2031 г.

Руководитель Федеральной службы  
по интеллектуальной собственности

Б.П. Симонов



РОССИЙСКАЯ ФЕДЕРАЦИЯ



**ПАТЕНТ**

НА ИЗОБРЕТЕНИЕ

№ 2475838

**УСТРОЙСТВО КРИПТОГРАФИЧЕСКОЙ ЗАЩИТЫ  
ИНФОРМАЦИИ**

Патентообладатель(ли): *Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования Иркутский государственный университет путей сообщения (ФГБОУ ВПО ИрГУПС) (RU)*

Автор(ы): *см. на обороте*

Заявка № 2011145012

Приоритет изобретения **07 ноября 2011 г.**

Зарегистрировано в Государственном реестре изобретений Российской Федерации **20 февраля 2013 г.**

Срок действия патента истекает **07 ноября 2031 г.**

*Руководитель Федеральной службы  
по интеллектуальной собственности*

*Б.П. Симонов*





## РОССИЙСКАЯ ФЕДЕРАЦИЯ



## ПАТЕНТ

НА ПОЛЕЗНУЮ МОДЕЛЬ

№ 63588

248  
3 стр.

## САМОКОНТРОЛИРУЕМЫЙ АВТОМАТ УПРАВЛЕНИЯ

Патентообладатель(ли): *Государственное образовательное учреждение высшего профессионального образования Иркутский государственный университет путей сообщения (ИрГУПС) (RU)*

Автор(ы): *см. на обороте*

Заявка № 2006143930

Приоритет полезной модели 11 декабря 2006 г.

Зарегистрировано в Государственном реестре полезных моделей Российской Федерации 27 мая 2007 г.

Срок действия патента истекает 11 декабря 2011 г.

Руководитель Федеральной службы по интеллектуальной собственности, патентам и товарным знакам



Б.П. Симонов

## РОССИЙСКАЯ ФЕДЕРАЦИЯ



## ПАТЕНТ

НА ПОЛЕЗНУЮ МОДЕЛЬ

№ 158425

## УСТРОЙСТВО КЛАССИФИКАЦИИ ИЗОБРАЖЕНИЙ

Патентообладатель(ли): *Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования Иркутский государственный университет путей сообщения (ФГБОУ ВПО ИРГУПС) (RU)*

Автор(ы): *см. на обороте*

Заявка № 2015111830

Приоритет полезной модели 01 апреля 2015 г.

Зарегистрировано в Государственном реестре полезных моделей Российской Федерации 08 декабря 2015 г.

Срок действия патента истекает 01 апреля 2025 г.

Руководитель Федеральной службы  
по интеллектуальной собственности

*Г.П. Ивлиев* Г.П. Ивлиев



РОССИЙСКАЯ ФЕДЕРАЦИЯ



**ПАТЕНТ**

НА ПОЛЕЗНУЮ МОДЕЛЬ

№ 82888

**МИКРОПРОГРАММНЫЙ АВТОМАТ**

Патентообладатель(ли): *Государственное образовательное учреждение высшего профессионального образования Иркутский государственный университет путей сообщения (ИрГУПС (ИрИИТ)) (RU)*

Автор(ы): *см. на обороте*

Заявка № 2008149344

Приоритет полезной модели 15 декабря 2008 г.

Зарегистрировано в Государственном реестре полезных моделей Российской Федерации 10 мая 2009 г.

Срок действия патента истекает 15 декабря 2018 г.

Руководитель Федеральной службы по интеллектуальной собственности, патентам и товарным знакам



Б.П. Симонов

РОССИЙСКАЯ ФЕДЕРАЦИЯ



**ПАТЕНТ**

НА ПОЛЕЗНУЮ МОДЕЛЬ

№ 82889

**УСТРОЙСТВО КРИПТОГРАФИЧЕСКОЙ ЗАЩИТЫ  
ИНФОРМАЦИИ**

Патентообладатель(ли): *Государственное образовательное учреждение высшего профессионального образования "Иркутский государственный университет путей сообщения" (ИрГУПС (ИрИИТ)) (RU)*

Автор(ы): *см. на обороте*

Заявка № 2008150879

Приоритет полезной модели **22 декабря 2008 г.**

Зарегистрировано в Государственном реестре полезных моделей Российской Федерации **10 мая 2009 г.**

Срок действия патента истекает **22 декабря 2018 г.**

*Руководитель Федеральной службы по интеллектуальной собственности, патентам и товарным знакам*



Б.П. Симонов

РОССИЙСКАЯ ФЕДЕРАЦИЯ



**ПАТЕНТ**

НА ПОЛЕЗНУЮ МОДЕЛЬ

№ 82890

**УСТРОЙСТВО КРИПТОГРАФИЧЕСКОЙ ЗАЩИТЫ  
ИНФОРМАЦИИ**

Патентообладатель(ли): *Государственное образовательное учреждение высшего профессионального образования "Иркутский государственный университет путей сообщения" (ИрГУПС (ИрИИТ)) (RU)*

Автор(ы): *см. на обороте*

Заявка № 2008150881

Приоритет полезной модели 22 декабря 2008 г.

Зарегистрировано в Государственном реестре полезных моделей Российской Федерации 10 мая 2009 г.

Срок действия патента истекает 22 декабря 2018 г.

*Руководитель Федеральной службы по интеллектуальной собственности, патентам и товарным знакам*



*Б.П. Симонов*

РОССИЙСКАЯ ФЕДЕРАЦИЯ



**ПАТЕНТ**

НА ПОЛЕЗНУЮ МОДЕЛЬ

№ 82974

**УСТРОЙСТВО КРИПТОГРАФИЧЕСКОЙ ЗАЩИТЫ  
ИНФОРМАЦИИ**

Патентообладатель(ли): *Государственное образовательное учреждение высшего профессионального образования Иркутский государственный университет путей сообщения (ИрГУПС (ИрИИТ)) (RU)*

Автор(ы): *с.м. на обороте*

Заявка № 2008149331

Приоритет полезной модели 15 декабря 2008 г.

Зарегистрировано в Государственном реестре полезных моделей Российской Федерации 10 мая 2009 г.

Срок действия патента истекает 15 декабря 2018 г.



*Руководитель Федеральной службы по интеллектуальной собственности, патентам и товарным знакам*

Б.П. Симонов

## РОССИЙСКАЯ ФЕДЕРАЦИЯ



ПАТЕНТ

НА ПОЛЕЗНУЮ МОДЕЛЬ

№ 158425

## УСТРОЙСТВО КЛАССИФИКАЦИИ ИЗОБРАЖЕНИЙ

Патентообладатель(ли): *Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования Иркутский государственный университет путей сообщения (ФГБОУ ВПО ИрГУПС) (RU)*

Автор(ы): *см. на обороте*

Заявка № 2015111830

Приоритет полезной модели 01 апреля 2015 г.

Зарегистрировано в Государственном реестре полезных моделей Российской Федерации 08 декабря 2015 г.

Срок действия патента истекает 01 апреля 2025 г.

*Руководитель Федеральной службы  
по интеллектуальной собственности*

*Г.П. Ивлиев*







РОССИЙСКАЯ ФЕДЕРАЦИЯ

Иркутский государственный  
университет путей сообщения  
**ПАТЕНТНЫЙ ОТДЕЛ**


Косовых Артем Родионович Яковлев  
Юрьевич & В. Кошуров & В.

**СВИДЕТЕЛЬСТВО**

о государственной регистрации программы для ЭВМ

**№ 2015615404**

**Программа управляющего автомата автоматизированной  
системы очистки технологических сред с применением  
вихревого эффекта и ультразвука**

Правообладатель: **Федеральное государственное бюджетное  
образовательное учреждение высшего профессионального  
образования «Иркутский государственный университет путей  
сообщения» (ФГБОУ ВПО ИРГУПС) (RU)**

Авторы: **Филатов Дмитрий Алексеевич (RU), Марков Родион  
Юрьевич (RU), Мухомад Александр Юрьевич (RU)**

Заявка № **2015612467**Дата поступления **31 марта 2015 г.**

Дата государственной регистрации

в Реестре программ для ЭВМ **18 мая 2015 г.**

Врио руководителя Федеральной службы  
по интеллектуальной собственности

Л.Л. Курий

УТВЕРЖДАЮ  
Первый заместитель директора Филиала ОАО  
«НТЦ ФСК ЕЭС» – СибНИИЭ

Макеев А.А.



\_\_\_\_\_ 2015 г.

**Справка  
об использовании результатов диссертационной работы**

Методы синтеза управляющих автоматов для сложных технических систем, предложенные в диссертационной работе «Анализ и синтез устройств управления проблемно-ориентированными средствами вычислительной техники и сложными техническими системами», представленной Мухопадом Александром Юрьевичем на соискание ученой степени доктора технических наук по специальности 05.13.05, имеют перспективу практического использования и будут рассмотрены на предмет применения при разработке устройств управления режимами электроэнергетических систем на основе технологий smart grid.

**Главный специалист  
канд. техн. наук**

**А.П. Долгов**

« \_\_\_\_ » \_\_\_\_\_ 2015 г.

Филиал ОАО «НТЦ ФСК ЕЭС» - СибНИИЭ, 630126, Россия, г. Новосибирск, ул. Кленовая, 10/1, <http://www.ntc-power.ru/>; e-mail: [office@ntcsib.ru](mailto:office@ntcsib.ru), тел. (383)2440602.

РОССИЙСКАЯ ФЕДЕРАЦИЯ



ЗАКРЫТОЕ АКЦИОНЕРНОЕ  
ОБЩЕСТВО  
ОГРН 1023801429068

Юридический адрес: 664048, Иркутск, ул. Розы  
Люксембург, 184.  
Для корреспонденции: 664040, г. Иркутск, а/я 87

Тел (3952) 211-140  
Факс (3952) 211-139

www.enerpred.com  
e-mail:  
info@enerpred.com

г. Иркутск

29.09.2015 г.

**Акт**

**о внедрении результатов диссертационных исследований  
Мухопода Александра Юрьевича  
на тему «Анализ и синтез устройств управления проблемно-  
ориентированными средствами вычислительной техники и сложными  
техническими системами»**

Настоящий акт свидетельствует о внедрении в практику проектирования средств производственной автоматизации предприятия ЗАО «Энерпруд» метода синтеза управляющих автоматов, разработанного в диссертационном исследовании Мухопода А.Ю.

Внедрение метода Мухопода А.Ю. позволило:

- 1) упростить и структурировать процесс программирования;
- 2) существенно сократить объем памяти микроконтроллера для управляющей программы;
- 3) повысить быстродействие системы управления;
- 4) добиться упрощения структуры управляющего автомата и, благодаря этому, в целом для разработанных систем управления:
  - снизить затраты на элементную базу;
  - повысить надежность функционирования;
  - снизить энергопотребление.

**Вывод:** результаты диссертационных исследований Мухопода А.Ю. представляют практический интерес и успешно применены на производстве.

Генеральный директор  
ЗАО «Энерпруд»



/ А. Ю. Иванкович



## Открытое акционерное общество «Томская нефтегазовая компания»

Юридический адрес: 636700 с. Каргасок, ул. М. Горького, 93

Местонахождение обособленного подразделения:  
634050 г. Томск, пер. 1905 года д. 18 т/ф: (3822) 51-60-80

### АКТ

О внедрении результатов диссертационной работы А.Ю.Мухопода на тему  
“Анализ и синтез устройств управления проблемно-ориентированными  
средствами вычислительной техники и сложными техническими системами”,  
представленной на соискание ученой степени доктора технических наук по  
специальности 05.13.05 — Элементы и устройства вычислительной техники и  
систем управления.

Настоящий Акт выдан в том, что в ОАО «ТНГК» были внедрены  
результаты исследований А.Ю. Мухопода, а именно:

- реализация систем управления комплексом оборудования нефтяных и газовых агрегатов;
- реализация систем плавного запуска погружных насосов ЭЦНА5А – 50 - 1930; ЭЦНА5 – 30 -2500; ЭЦНА5 – 50 -2426.

Генеральный директор



В.Р.Костылев

## ФЕДЕРАЛЬНОЕ АГЕНТСТВО ЖЕЛЕЗНОДОРОЖНОГО ТРАНСПОРТА

Федеральное государственное бюджетное образовательное учреждение  
высшего профессионального образования

**«Иркутский государственный университет путей сообщения»**

**ФГБОУ ВПО ИргГУПС**

Чернышевского ул., 15, Иркутск, 664074

Тел.: (3952) 63-83-11, факс (3952) 38-77-46. E-mail: [mail@irgups.ru](mailto:mail@irgups.ru), <http://www.irgups.ru>  
22.06.2015 № 01-2434-19 на № \_\_\_\_\_ от \_\_\_\_\_

## АКТ

О внедрении результатов диссертационной работы А.Ю.Мухопода на тему  
“Анализ и синтез устройств управления проблемно-ориентированными  
средствами вычислительной техники и сложными техническими системами”,  
представленной на соискание ученой степени доктора технических наук по  
специальности 05.13.05 — Элементы и устройства вычислительной техники и  
систем управления.

Настоящий Акт выдан в том, что в учебном процессе Иркутского  
государственного университета путей сообщения по курсам «Теория  
дискретных устройств», «Архитектура ЭВМ и систем», а так же в курсовом и  
дипломном проектировании использовались следующие результаты  
исследований А.Ю. Мухопода:

- методика синтеза аппаратной реализации управляющих автоматов систем  
железнодорожного транспорта;
- методика синтеза самоконтролируемых автоматов управления  
информационно-измерительными системами;
- метод структурно-автоматного программирования микропроцессорных и  
микроконтроллерных систем железнодорожного транспорта;
- метод криптографической защиты информации в системах реального  
времени.

Ректор



А.П. Хоменко

«УТВЕРЖДАЮ»

И.о. ректора ФГБОУ ВО «ИРНИТУ»,  
 доктор физико-математических наук,  
 профессор  
 Афанасьев Александр Диомидович



2015 г.

**А К Т**

об использовании результатов научной работы  
 Мухопода Александра Юрьевича

Комиссия, включающая заведующего кафедрой электропривода и электротранспорта (ЭЭТ) Арсентьева О.В., к.т.н., доцента; профессора кафедры ЭЭТ Дунаева М.П., д.т.н., проф.; составила настоящий акт в том, что результаты научной работы Мухопода А.Ю. внедрены в учебный процесс института энергетики федерального государственного бюджетного образовательного учреждения высшего образования «Иркутский национальный исследовательский технический университет» (ФГБОУ ВО «ИРНИТУ»).

Разработанная Мухопадом А.Ю. методика синтеза аппаратной реализации управляющих автоматов и метод структурно-автоматного программирования микропроцессорных и микроконтроллерных систем используются при проведении лабораторных занятий и дипломном проектировании студентов специальности 140604 «Электропривод и автоматика производственных установок и технологических комплексов». Использование вышеуказанных результатов позволяет повысить эффективность процесса обучения студентов и углубить их знания по учебной дисциплине «Автоматизация типовых технологических процессов и установок».

Заведующий кафедрой электропривода  
 и электротранспорта, к.т.н., доцент

Арсентьев О.В.

Профессор кафедры ЭЭТ, д.т.н., проф.

Дунаев М.П.



МИНОБРНАУКИ РОССИИ

Федеральное государственное  
бюджетное образовательное  
учреждение высшего  
профессионального образования  
«Братский государственный  
университет»  
(ФГБОУ ВПО «БрГУ»)

Макаренко ул., д. 40, г. Братск,  
Иркутская обл., 665709  
Тел. (3953) 33-20-08, факс (3953) 33-20-08,  
E-mail: rektor@brstu.ru, http://www.brstu.ru

№ \_\_\_\_\_



УТВЕРЖДАЮ

Первый проректор

В.Б. Кашуба

«\_\_\_» \_\_\_\_\_ 201\_ г.

## АКТ

## о внедрении результатов НИР в учебный процесс

## Результаты

- методы синтеза аппаратно и программно реализуемых управляющих автоматов специпроцессоров информационно-управляющих систем реального времени;
- новые методы анализа и синтеза управляющих автоматов, включая синтез взаимодействующих и самоконтролируемых автоматов;
- метод структурно-автоматного программирования микропроцессорных и микроконтроллерных систем железнодорожного транспорта;
- метод криптографической защиты информации в системах реального времени.

диссертационной работы Анализ и синтез устройств управления проблемно-ориентированными средствами вычислительной техники и сложными техническими системами

представленной Мухопадом Александром Юрьевичем на соискание ученой степени доктора технических наук по специальности 05.13.05 — Элементы и устройства вычислительной техники и систем управления.

в 2014-2015 гг. внедрены в учебный процесс на основании решения кафедры Управление в технических системах, протокол заседания кафедры №9 от 20.05.2015

Указанные результаты включены в дисциплины Автоматизированные информационно-управляющие системы, Технические средства автоматизации и управления в качестве лекционного материала

Зав. кафедрой Управления  
в технических системах

И.В. Игнатьев

«\_\_\_» \_\_\_\_\_ 2015 г.

Начальник управления  
научно-инновационной  
деятельностью

М.Ю. Иванов

«\_\_\_» \_\_\_\_\_ 2015 г.

Начальник учебно-методического  
управления

Г.П. Нежевец

«\_\_\_» \_\_\_\_\_ 2015 г.



УТВЕРЖДАЮ  
Заместитель начальника  
ВУНЦ ВВС «ВВА» (г. Воронеж) по УНР  
генерал-майор

А. Нагалин

2015 г.

АКТ

**реализации в учебном процессе**

Комиссия в составе: председателя - полковника Шуклина И.К. и членов: полковника Аврамова А.В., полковника Кирюшкина В.В., полковника Лущик А.В. составила настоящий акт о том, что научные результаты:

- новая теория синтеза автоматов реального времени, основанная на предварительном преобразовании операторной схемы алгоритма управления за счет ввода пустых операторов в отдельные ветви граф-схем;
- автоматы реального времени нового типа, имеющие оригинальную запатентованную структурную организацию;
- методы синтеза самоконтролируемых автоматов нового типа;
- практическая реализация теории анализа и синтеза управляющих автоматов на примере спецпроцессоров технических систем радионавигации, радиолокации и управления реального времени,

предложенные и представленные в монографии:

А.Ю. Мухопад Теория управляющих автоматов технических систем реального времени / А.Ю. Мухопад / под ред. д.т.н. Ю.Ф. Мухопада. – Новосибирск: Наука, 2015. -224 с.

использованы для разработки учебно-методических материалов по дисциплинам:

«Бортовые цифровые вычислительные машины комплексов авиационного вооружения» (Тема №1 Математические основы построения БЦВМ военного назначения. Лекция №5 Основы синтеза логических схем БЦВМ);

«Системы автоматического управления полетом» (Тема №11 Оптимальное управление движением воздушного судна военного назначения и концепция активного управления. Лекция №34 Идентификация летательных аппаратов и концепция активного управления на 92 кафедре автоматизации управления летательными аппаратами (и вычислительных систем) при подготовке инженеров по специальностям «Робототехнические системы авиационного вооружения» и «Техническая эксплуатация и восстановление электросистем и пилотажно-навигационных комплексов боевых летательных аппаратов»;

«Радионавигационные системы» (Тема №7. Основы комплексирования радионавигационных систем, Лекция № 20 «Навигационные комплексы воздушных судов военного назначения»),

«Моделирование систем и процессов» (Тема №2 Организация и методология моделирования сложных технических систем, Лекция №13 Моделирование авиационных систем управления) на 102 кафедре авиационных систем и комплексов радионавигации и радиосвязи при подготовке инженеров по специальности «Техническая эксплуатация транспортного радиооборудования»;

«Автоматика и управление» (Тема №4 Синтез систем автоматического управления и систем автоматического регулирования, Лекция №12 Способы улучшения показателей качества автоматических систем.) на 101 кафедре авиационных радиоэлектронных комплексов при подготовке инженеров по специальности «Техническая эксплуатация транспортного радиооборудования».

Акт обсуждён и одобрен на расширенном заседании 102 кафедры, протокол № 6 от 5 октября 2015 года.

Председатель комиссии: полковник

И. Шуклин

(воинское звание, подпись, фамилия)

Члены комиссии:

полковник

А. Аврамов

(воинское звание, подпись, фамилия)

полковник

В. Кирюшкин

(воинское звание, подпись, фамилия)

полковник

А. Лущик

(воинское звание, подпись, фамилия)

« 5 » октября 2015 г.



## Листинг работы УА нового типа — вариант процедурного программирования на языке Pascal

```

kommutator[i,j]- массив данных, содержащий значения таблицы переходов (таб.5)
comb[i,j] - массив данных, содержащий значения таблицы КС F1 (таб.4)
dsh[i,j] - массив данных, содержащий значения таблицы КС F2 -дешифратора (таб.6)
F2[i,j] - массив данных, содержащий коды и значения операторов действия (таб.3)

// процедура инициализации
procedure init_state;
var i: byte;

begin
  //обнуление регистра условий
  reg_alfa.Clear;
  //обнуление регистра F3(reg6)
  reg6:="";
  //обнуление регистра состояний
  reg9:="";
  //обнуление регистра состояний
  reg7:="";
end;

//опрос MUX: условие по полученному коду состояния
function mux_opros(reg:string):string;

var i: byte;
begin
  //поиск по всей таблице
  for i:=1 to R do
    if reg=kommutator[i,1] then
      mux_opros:=kommutator[i,2];
  end;

//опрос КС F1
function fl_opros(reg:string):string;
var i:byte;
begin
  //поиск по всей таблице
  for i:=1 to H do
    if (reg=comb[i,2]) and
      (mux_opros(reg6)=comb[i,3]) and
      (reg_alfa[mux_opros(reg6)]=comb[i,4])
      then fl_opros:=comb[i,5];

  //обновление регистра состояний
  reg6:=fl_opros;
  //обновление регистра состояний
  reg7:=fl_opros;
end;

//опрос КС F2 - дешифратора
function comand (y:string):string;
var i:byte;
begin
  //поиск по всей таблице дешифратора
  for i:=1 to R do
    if (y=dsh[i,1]) then
      statecode:=dsh[i,2];
  //выработка команды по коду
  for i:=1 to R do
    if (f2[i,1]=statecode) then
      comand:=f2[i,2];
  end;
//главная процедура

procedure on_time();
var res:string;
begin
  //инициализация
  init_state;
  //такт работы автомата
  res:=comand(fl_opros(reg9));
  //обновление регистров
  reg6:=fl_opros(reg9);
  reg9:=reg7;
  reg7:=reg6;
end;

```

## Листинг работы УА нового типа— вариант для объектно-ориентированного программирования на языке Java

```

kommutator[i,j]- массив данных, содержащий
значения таблицы переходов (таб.5)
comb[i,j] - массив данных, содержащий
значения таблицы КС F1 (таб.4)
dsh[i,j] - массив данных, содержащий
значения таблицы КС F2 -дешифратора
(таб.6)
F2[i,j] - массив данных, содержащий коды и
значения операторов действия (данные из
таб.2,3)

Class NewAutomata
public void init_state() {
byte i;
for (int i = 1; i <= B; i++) {
//обнуление регистра условий
reg_alfa[i]=0;
//обнуление регистров состояний
reg6="";
reg9="";
reg7="";
}

// опрос MUX: условие по полученному коду
состояния
public mux_opros(String reg) {
i: byte;
//поиск по всей таблице
for (int i = 1; i <= R; i++) {
if reg=kommutator[i,1] (
f3_opros=kommutator[i,2])
}
}

//опрос КС F1
public f1_opros(String reg) {
I byte;
//поиск по всей таблице
for (int i = 1; i <= H; i++) {
if (reg=comb[i,2]) &
(mux_opros(reg6)=comb[i,3]) &
(reg_alfa[mux_opros(reg6)]=comb[i,4]) {
f1_opros=comb[i,5];
//обновление регистра состояний
reg6=f1_opros;
//обновление регистра состояний
reg7=f1_opros;
}
}

//опрос КС F2 - дешифратора
public comand(String y) {
i byte;
//поиск по всей таблице дешифратора
for (int i = 1; i <= R; i++) {
if (y=dsh[i,1]) {statecode=dsh[i,2]}
for (int i = 1; i <= R; i++) {
//выработка команды по коду
if (f2[i,1]=statecode) { comand=f2[i,2]}
}
}

public class Main {
public static void main(String[] args) {
String res;
//инициализация
init_state();
//такт работы автомата
res=comand(f1_opros(reg9));
//обновление регистров
reg6=f1_opros(reg9);
reg9=reg7;
reg7=reg6;
}
}

```