Федеральное агентство связи
Федеральное государственное бюджетное образовательное учреждение высшего образования
«Сибирский государственный университет телекоммуникаций и информатики»
(СибГУТИ)

На правах рукописи

Шабронов Андрей Анатольевич

УДК 630.02

Исследование синтеза управляющих систем с конечным набором команд на основе сингулярной сигнальной модели

специальность 05.12.03
«Системы, сети и устройства телекоммуникаций»

Диссертация на соискание ученой степени
кандидата технических наук

г.Новосибирск - 2020г

Содержание.

- Введение;
- Глава 1. Обзор управляющих систем:
    - Управляющие автоматы первого уровня;
    - Модель "клиент-сервер" в двух уровневой системе управления;
    - Модель "главный сервер" - "подчиненные сервера" в трех уровневой системе управления.
- Глава 2. Анализ последовательности управляющих сигналов:
    - Ограничения старт-стопных каналов в управляющих системах;
    - Анализ синхронных каналов, преимущества и недостатки;
    - Анализ преобразования сигналов в управляющие воздействия на примере электродвигателя постоянного тока;

- Глава 3. Синтез сигнала сингулярной сигнальной модели:
    - Расчет точности для управления двигателем постоянного тока;
    - Расчет ограничений на синтезируемые сигналы на примере модели управления двигателем постоянного тока.
    - Расчет ограничений компьютера на примере модели "кванта учета" вычислительной мощности;

- Глава 4. Описание управления двигателем постоянного тока с использованием сигналов сингулярной сигнальной модели:
    - Блочная схема;
    - Принципиальная схема управления по интерфейсу старт-стопного RS485 канала связи электродвигателем постоянного тока;
    - Достоинства и недостатки предлагаемой схемы управления;

- Выводы.
- Литература

Введение.

В современном мире все более важное место занимает проблема построения систем управления, будь то управление и измерения в промышленном секторе или в частной жизни. В общем случае это вызвано растущем спросом на обработку большого количества данных и оптимального принятия решения. Наиболее проблематично дела обстоят в области разработки программного обеспечения.

Особенно большие затраты, видны при разработке специальных программ, которые "записываются" в микроконтроллеры. Микропрограммы - это слабое звено в эксплуатации и в расширении системы управления. Поменять микропрограмму в микроконтроллере иногда бывает просто не возможно, т.к. нет тех специалистов, которые это могут сделать. Или стоимость модификации старой

микропрограммы превышает стоимость замены всей системы управления.

Кроме того, при длительной эксплуатации, возникает ситуация что неисправный микроконтроллер невозможно заменить, т.к. он уже не выпускается. Или если есть в наличии микросхема, то нет возможности ее запрограммировать.

Один из вариантов решения вопроса "микропрограмм", это перенос "веса затрат" программ на платформу "открытого кода" и "внешнего управления". Эта возможность появляется при наличии высоко скоростных каналов связи и "мощных компьютеров".

Наличие высоко скоростных каналов связи, а сейчас оптоволоконные каналы работают со скоростями в гигагерцы, предполагают разные решения в системах управления.

Сингулярная сигнальная модель команд предполагает для расчета решения задач управления использовать "интеллект центра", а "периферии" оставить только выполнение.

В данной работе исследовано использование электронных ключей для управления элементами механики: электрическими двигателями автоматизации, различными элементами механического управления, сервоприводами и т.п., что увеличивает ремонтопригодность, расширяемость и надежность. В предлагаемой модели управления "микропрограммы и микроконтроллеры" реализованы на языках высокого уровня в центральном компьютере, что увеличивает доступность и информативность системы управления.

Научная ценность работы заключается в том, что найден, исследован и подтвержден экспериментальной моделью принципиально новый метод формирования сигналов управления физическими устройствами,

который создает возможность уменьшения «цены вопроса» при решении практических производственных задач.

Глава 1. Обзор управляющих систем. Управляющие автоматы первого уровня.

Управляющие автоматы или системы первого уровня, называют еще иначе автоматы с "жесткой логикой".

Управляющий автомат (УА) генерирует последовательность управляющих сигналов из множества Y1....Ym

На рисунке 1. представлен управляющий автомат как "черный ящик".

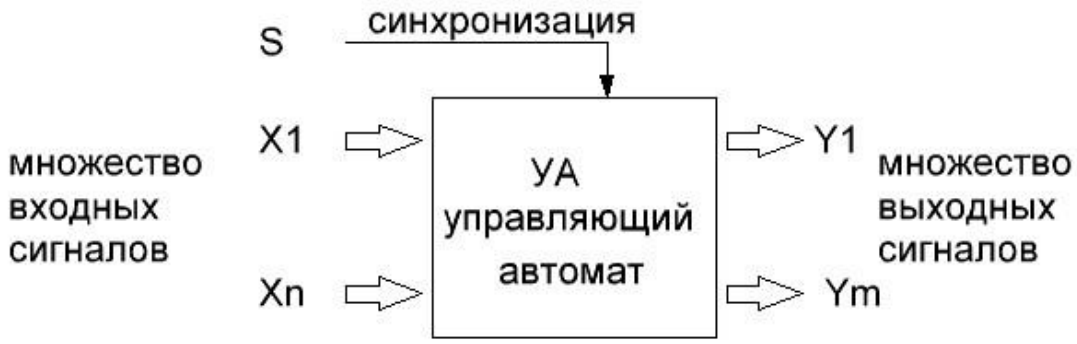


Рисунок 1. Абстрактный управляющий автомат

Сигналы Y1...Ym называются микрооперациями, каждый из сигналов может принимать только одно из значений 1 или 0, предписанную микропрограммой У и соответствующую значениям логическим условий X1...Xn.

При выполнении процессором пакета микропрограмм на его входы последовательно подаются коды операции, которые соответствуют той или иной микропрограмме.

На входы процессора могут поступать внешние сигналы логических условий, а с выходов сниматься сигналы для управления внешними устройствами.

Переход на новый шаг алгоритма осуществляется только с приходом специального сигнала синхронизации (S).

Выходные сигналы Y1...Ym могут иметь различную длительность. Математической моделью управляющих автоматов, формирующих короткие выходные сигналы, является модель Мили[Л1], а для автоматов, формирующих длинные выходные сигналы - модель Мура [Л2].

Примером подобных автоматов является контроллер температур или контроллер каких либо переключений по событиям.

Эти системы управления автономны и использование в них сингулярной сигнальной модели не применимо.

Модель "клиент-сервер" в двух уровневой системе управления.

Развитие модели "клиент-сервер" исторически начиналось с систем управления базами данных [Л3]. На рисунке 2 показана построение СУБД.

Что дает архитектура клиент-сервер?

Надежность

Сервер баз данных осуществляет модификацию данных на основе механизма транзакций, который придает любой совокупности операций, объявленных как транзакция, следующие свойства:

- атомарность - при любых обстоятельствах будут либо выполнены все операции транзакции, либо не выполнена ни одна;

- целостность данных при завершении транзакции;
- независимость - транзакции, инициированные разными пользователями, не вмешиваются в дела друг друга;

устойчивость к сбоям - после завершения транзакции, ее результаты уже не пропадут.

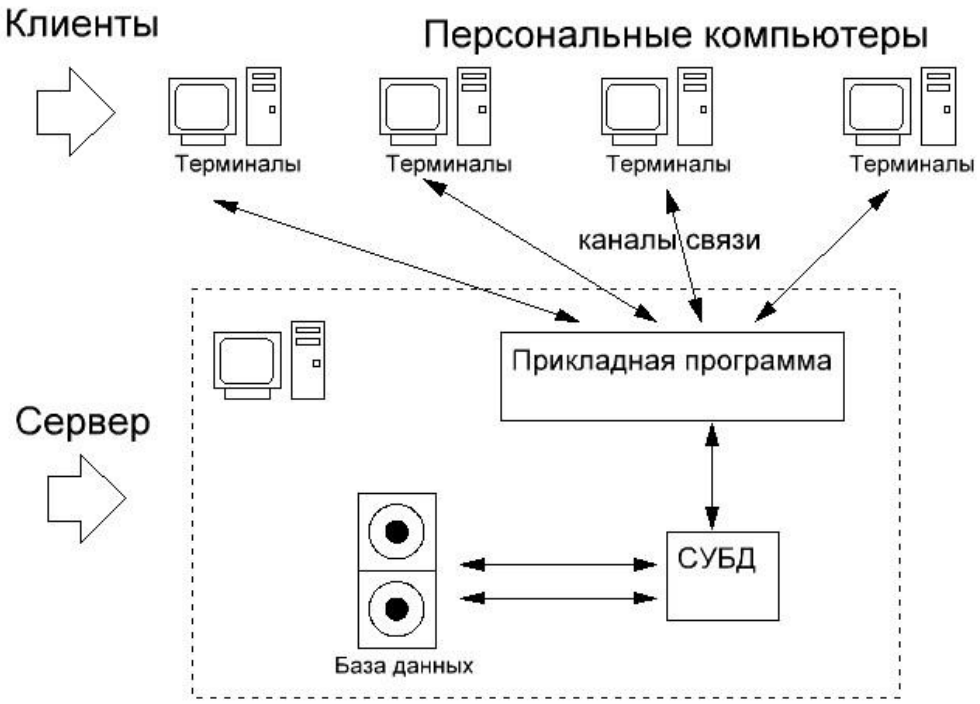


Рисунок 2. Клиент-серверая СУБД

- в файл-серверной системе сбой на любой из рабочих станций может привести к потере данных и их недоступности для других рабочих станций, в то время, как в клиент-серверной системе сбой на клиенте, практически, никогда не сказывается на целостности данных и их доступности для других клиентов.
- Масштабируемость - способность системы адаптироваться к росту количества пользователей и объема базы данных при адекватном повышении производительности аппаратной

платформы, без замены программного обеспечения. Системы же на основе серверов баз данных могут поддерживать тысячи пользователей и сотни ГБ информации - дайте им только соответствующую аппаратную платформу.

- Безопасность .Сервер баз данных предоставляет мощные средства защиты данных от несанкционированного доступа, невозможные в настольных СУБД. При этом, права доступа администрируются очень гибко - до уровня полей таблиц. Кроме того, можно вообще запретить прямое обращение к таблицам, осуществляя взаимодействие пользователя с данными через промежуточные объекты - представления и хранимые процедуры. Так что администратор может быть уверен - никакой слишком умный пользователь не прочитает то, что ему читать неположено.
- Гибкость. В приложении, работающем с данными, можно выделить три логических слоя:
    - пользовательского интерфейса ;
    - правил логической обработки;
    - управления данными;

Приведенные достоинства двух уровневой модели СУБД в настоящее время широко используются при построении SCADA систем. [Л4]

SCADA (сокращенно от английского: **S**upervisory **C**ontrol **A**nd **D**ata **A**cquisition - диспетчерское управление и сбор данных) [Л5].

Основная задача таких система визуализация управляемых процессов. На рисунке 3 представлен вид рабочего место оператора, управляющий производством нефтепродуктов, из отечественной демо-версии SCADA системы TRACE MODE V6.2

SCADA-системы решают следующие задачи:

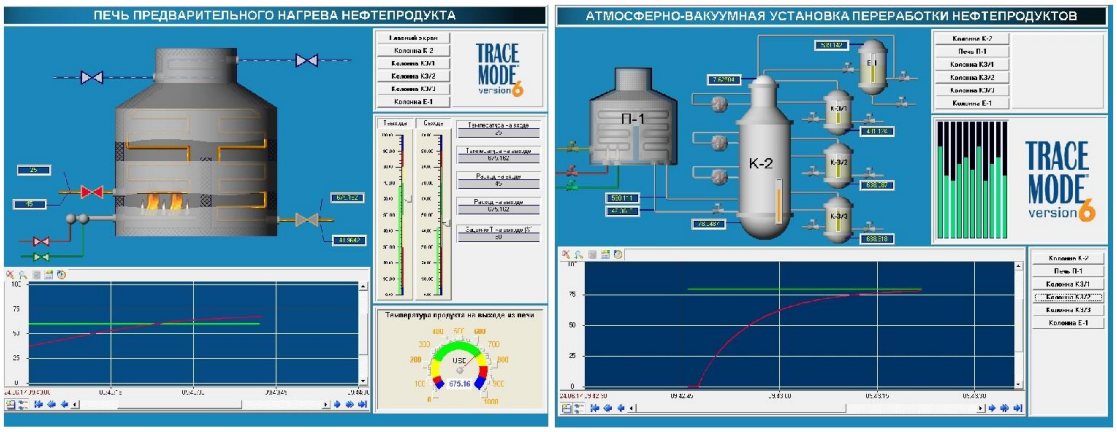


Рисунок 3. Варианты визуализации процессов управления

- Обмен данными с «устройствами связи с объектом» (то есть с промышленными контроллерами и платами ввода-вывода) в реальном времени через драйверы;
- Обработка информации в реальном времени;
- Логическое управление;
- Отображение информации на экране монитора в удобной и понятной для человека форме;
- Ведение базы данных реального времени с технологической информацией;
- Аварийная сигнализация и управление тревожными сообщениями;
- Подготовка и генерирование отчетов о ходе технологического процесса;
- Осуществление сетевого взаимодействия между SCADA ПК;
- Обеспечение связи с внешними приложениями (СУБД, электронные таблицы, текстовые процессоры и т. д.);

Необходимо отметить важную составляющую всех двух уровневых систем управления - всегда наличие канала связи между управляемым объектом и системой управления.

Другая важная часть - это программы обмена с контроллерами, которые имеют общее функциональное название OPC.

OPC (сокращенно от английского **O**le for **P**rocess **C**ontrol) — семейство программных технологий, предоставляющих единый интерфейс для управления объектами автоматизации и технологическими процессами.

В контексте далее изложенных предложений, необходимо отметить независимость программ, которые объединяются в процессе формирования SCADA системы. Это важный признак, который позволяет расширять и изменять программы нижнего уровня, независимо от верхнего. Важно выполнить требования по протоколам обмена.[Л6]

Модель "главный сервер" - "подчиненные сервера" в трех уровневой системе управления.

В трех уровневой системах, происходит объединение систем "клиент-сервер".

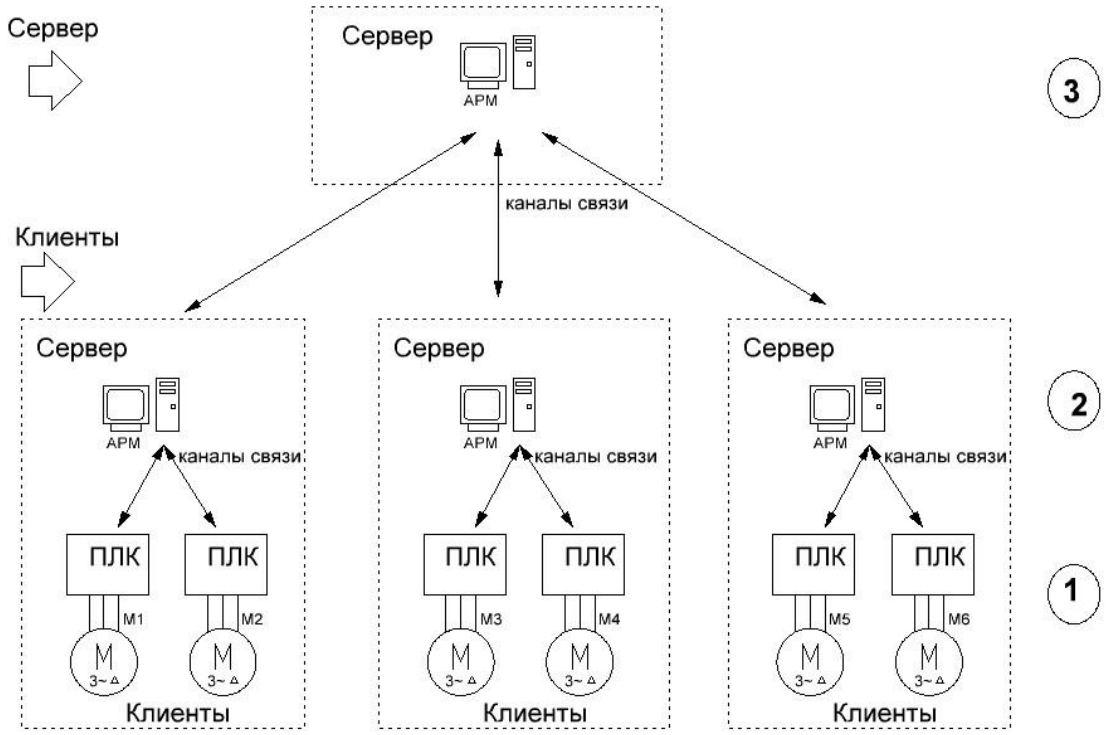


Рисунок 4. Трех уровневая структура =клиент-сервер=

В этом случае, возможны разные модификации систем управления, в которых допускается централизованное и децентрализованные структуры. На рисунке 4 представлена "древовидная" структура трех уровневой системы на модели объединения нескольких SCAD. Цифрами в кружочка указаны условные уровни системы управления. ПЛК - программируемый логический контроллер. М- мотор, которым управляет ПЛК. Например, такая структура просматривается при построении различных конвейерных линий.

Выводы по главе 1.

В контексте далее изложенных предложений, начиная с двух уровневой системой и в трех уровневой системе управления, остается задача программирования микроконтроллеров. И трудозатраты по прежнему не уменьшаются. Они накапливаются!

Глава 2. Анализ последовательности управляющих сигналов. Ограничения старт-стопных каналов в управляющих системах.

Рассмотрим ограничения интерфейса RS485, как одного из основных старт-стопных интерфейсов, из используемых в управляющих системах.

Термин "интерфейс" и "канал связи" имеют подобные и сходные параметры, которые и рассмотрим:[Л7]

Свойства интерфейса стандарта RS485.

- Двунаправленная полудуплексная передача данных. Поток последовательных данных передаётся одновременно только в одну сторону, передача данных в другую сторону требует переключения приёмопередатчика. Приёмопередатчики принято называть "драйверами"(driver), это устройство или электрическая цепь, которая формирует физический сигнал на стороне передатчика.
- Симметричный канал связи. Для приёма/передачи данных используются два равнозначных сигнальных провода. Провода означаются латинскими буквами "A" и "B". По этим двум проводам идет последовательный обмен данными в обоих направлениях (поочередно). При использовании витой пары симметричный канал существенно повышает устойчивость сигнала к синфазной помехе и хорошо подавляет электромагнитные излучения создаваемые полезным сигналом.

- Дифференциальный (балансный способ передачи данных). При этом способе передачи данных на выходе приёмопередатчика изменяется разность потенциалов, при передаче "1" разность потенциалов между АВ положительная при передаче "0" разность потенциалов между АВ отрицательная. То есть, ток между контактами А и В, при передачи "0" и "1", течёт (балансирует) в противоположных направлениях.
- Многоточечность. Допускает множественное подключение приёмников и приёмопередатчиков к одной линии связи. При этом допускается подключение к линии только одного передатчика в данный момент времени, и множество приёмников, остальные передатчики должны ожидать освобождения линии связи для передачи данных.
- "Низкоимпендансный" выход передатчика. Буферный усилитель передатчика имеет низкоомный выход, что позволяет передавать сигнал ко многим приёмникам. Стандартная нагрузочная способность передатчика равна 32-м приёмникам на один передатчик. Кроме этого, токовый сигнал используется для работы "витой пары" (чем больше рабочий ток "витой пары", тем сильнее она подавляется синфазные помехи на линии связи).
- Зона нечувствительности. Если дифференциальный уровень сигнала между контактами АВ не превышает ±200мВ, то считается, что сигнал в линии отсутствует. Это увеличивает помехоустойчивость передачи данных.

Так как стандарт, RS485 описывает только физический уровень процедуры обмена данными, то все проблемы обмена, синхронизации и

квитирования, возлагаются на более высокий протокол обмена. Это стандарт RS232 или другие верхние протоколы (ModBus , DCON и т.п.).

Сам RS485 выполняет только следующие действия:

- Преобразует входящую последовательность "1" и "0" в дифференциальный сигнал.
- Передает дифференциальный сигнал в симметричную линию связи.
- Подключает или отключает передатчик драйвера по сигналу высшего протокола.
- Принимает дифференциальный сигнал с линии связи.

При организации систем управления любого уровня приходится объединять большое число различных ЭВМ. Чтобы такое объединение происходило по возможности легко, т. е. разные типы компьютеров и сетей могли быть соединены между собой, и эффективно обмениваться информацией, Международной организацией по стандартизации (ISO) была разработана базовая модель взаимодействия открытых систем (OSI – Open System Interconnection). [Л8]

На сегодняшний день эта модель является международным стандартом для передачи данных.

В данной модели для описания взаимодействующих систем используется так называемый метод иерархической декомпозиции. Это означает разбиение сложной системы на уровни, связанные односторонней функциональной зависимостью.

Таким образом, в рамках данной модели каждая так называемая «открытая система», под которой понимается любая система от отдельного компьютера до глобальной сети, состоит из семи уровней:

- Прикладной
- Представительский
- Сеансовый
- Транспортный
- Сетевой
- Канальный
- Физический

Каждый из этих уровней отвечает за выполнение своего собственного круга задач и функций.

Физический уровень. На этом уровне обеспечивается взаимодействие со средой передачи данных (различные виды кабелей и т. п.), определяются физические: механические, электрические — и процедурные параметры связи. Данный уровень отвечает за готовность среды передачи данных к эксплуатации в любой момент времени. Здесь обеспечивается физический и логический доступ к среде передачи данных. На этом уровне также реализуются некоторые механизмы зашиты информации, например шифрование.

Канальный уровень. На этом уровне в передаваемое сообщение вносится некоторый «порядок»: оно разбивается на «кадры» (в различных системах название может быть разным), формируются

последовательности этих кадров. Также канальный уровень отвечает за управление доступом к среде, используемой несколькими ЭВМ, синхронизацию, обнаружение и исправление ошибок.

Более высокие уровни выходят за рамки исследований данной работы и поэтому их характеристики пропускаются.

Наиболее используемый в настоящее время порядок "канального" уровня - это старт-стопный формат передачи. Формат представлена на рисунке 5

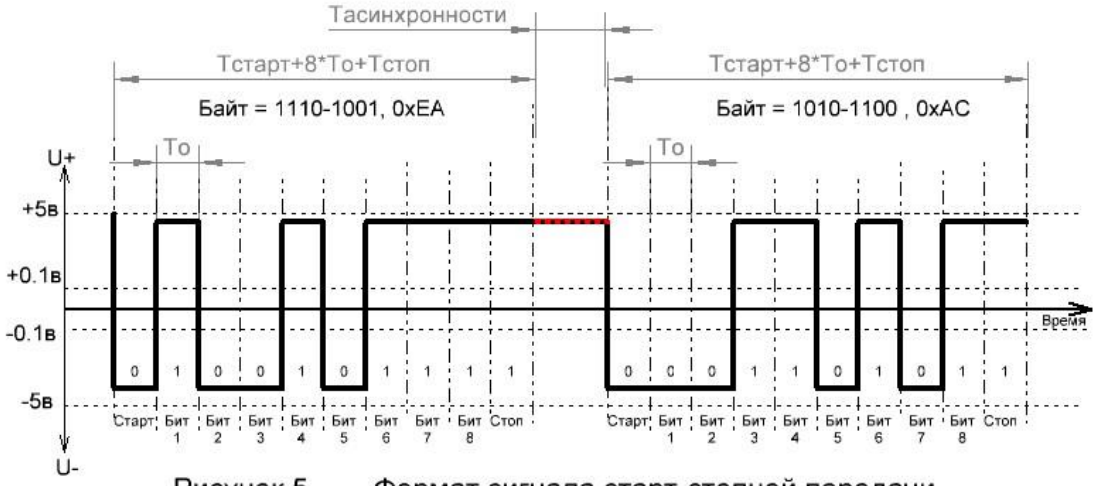


Рисунок 5. Формат сигнала старт-стопной передачи

В асинхронном режиме каждый байт данных сопровождается специальными сигналами «старт» и «стоп» . Назначение этих сигналов состоит в том, чтобы, во-первых, известить приемник о приходе данных и, во-вторых, чтобы дать приемнику достаточно времени для выполнения некоторых функций, связанных с синхронизацией, до поступления следующего байта.

Сигнал «старт» имеет продолжительность в один тактовый интервал, а сигнал «стоп» может длиться один, полтора или два такта.

Асинхронным описанный режим называется потому, что

каждый принятый байт может быть
смещен во времени относительно переданного байта на
случайный промежуток времени.

Это резко снижает требования к характеристикам системы передачи. В то же время, такая асинхронность передачи не влияет на корректность принимаемых данных, так как в начале каждого байта происходит дополнительная синхронизация приемника с источником за счет битов «старт».

Необходимо, учитывать, что все последовательности байт группируются в определенные группы. И группирование - это уровень сетевого протокола.

Важный фактор, показанный на рисунке 5, время Тасинхронности. Исследования показали, что данный показатель в пределах передаваемой группы и при хорошем канале связи практически постоянен.

Группа, это определенное количество байт, которая передается от одного источника к другому, или наоборот.

Примем считать хорошим каналом связи канал с вероятностью ошибки передачи бита не менее чем на 1 миллион передаваемых бит.

Количество байт в группе определяется исходя из вероятности ошибки, а так же из верхних протоколов взаимодействия системы управления. В теме исследования и предложенной модели вполне хватает этого параметра для построения системы управления.

Анализ синхронных каналов, преимущества и недостатки.

Необходимо различать два подхода к понятию синхронный канал.

Первый подход - синхронизм создается в самом канале связи. Пользовательские данные собираются в кадр, который предваряется байтами синхронизации (на рис.3 — флаги). Старт-стопные биты между соседними байтами отсутствуют. Байт синхронизации — это байт, содержащий заранее известный код, например 0111110, который оповещает приемник о приходе кадра данных. Его обычно называют флагом. При его получении приемник должен войти в байтовый синхронизм с передатчиком, то есть правильно понимать начало очередного байта кадра. Иногда применяется несколько синхробайт для обеспечения более надежной синхронизации приемника и передатчика. [Л9] [Л10]

На промежутке времени в интервале передачи групп, можно считать такой канал так же асинхронным, т.к. всегда требуется время на вхождение в синхронизацию.

Другой подход - синхронизм создается дополнительным каналом связи, т.е. отдельным проводом. Примером такого решения протокол I2C , который широко используется в управлении микроконтроллерами или между ними. [Л11] На рисунке 6 представлен пример линий протокола I2C.

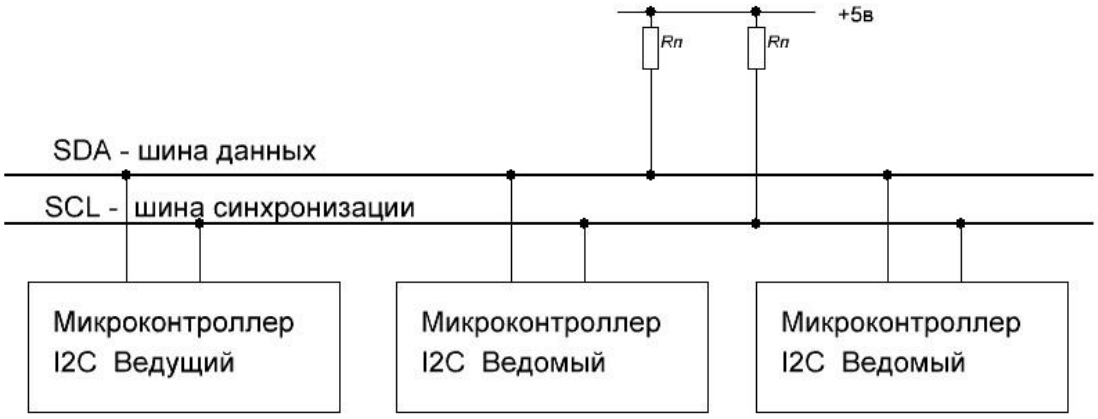


Рисунок 6. Конфигурация оборудования протокола I2C

Ведущий - это устройство, которое инициирует передачу данных и вырабатывает сигналы синхронизации. При этом любое адресуемое устройство считается ведомым по отношению к ведущему. Типичная конфигурация оборудования показана на рисунке 6:

- SDA - шина данных;
- SCL - шина сихронизации;

Обратите внимание, что к линиям SDA и SCL можно подключить несколько ведомых устройств. SDA и SCL являются двунаправленными линиями и подключаются к источнику питания через поттягивающий резистор Rп. В большинстве случаев ведущим устройством выступает микроконтроллер, а ведомыми могут быть часы реального времени и датчики температуры, АЦП или другие микроконтроллеры.

Данные могут передаваться только после создания стартовых условий. На рисунке 7 показаны сигналы взаимодействия шины I2C. Переход линии SDA из ВЫСОКОГО состояния в НИЗКОЕ, в то время как SCL находится в ВЫСОКОМ состоянии, означает START. После

каждой передачи данных выполняется условие остановки. Переход линии SDA из НИЗКОГО состояния в ВЫСОКОЕ при SCL в ВЫСОКОМ состоянии означает STOP. [Л12]

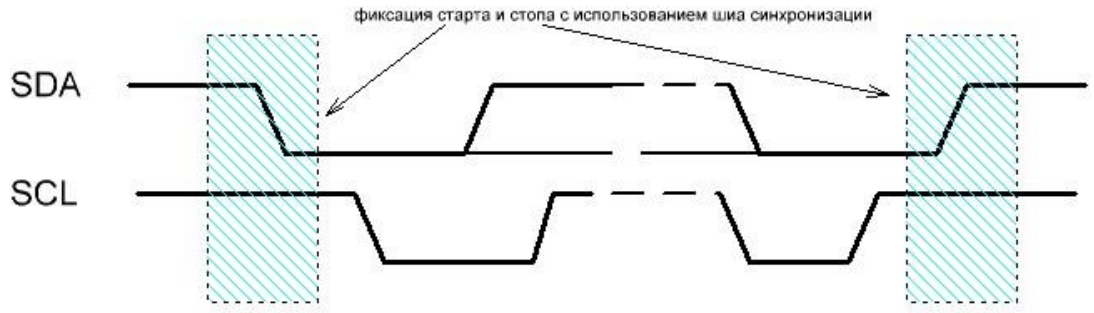


Рисунок 7. Сигналы в шине I2C

Преимущества I2C - большая скорость обмена при высокой помехозащищенности. Явный недостаток один - вторая линия шины данных.

Другой не явный недостаток - программирование микроконтроллеров для протокола обмена представляет специальную задачу и имеет достаточно большую трудоемкость.

Анализ сигналов такой синхронной линии показывает, что имеются так же временные промежутки для старта, передачи информации и стопа. Это подтверждает асинхронность всех систем управления. Всегда существует время для синхронизации, затем время на передачу информации и затем окончание передачи.

Однако, системы управления с разделением на каналы синхронизации и данных не входят в исследование данной работы.

Анализ преобразования сигналов в управляющие воздействия на примере электродвигателя постоянного тока

Рассмотрим используемые сигналы для управления двигателем постоянного тока в системе управления. Блочная схема управления двигателем постоянного тока в системе управления "клиент-сервер" представлена на рисунке 8.

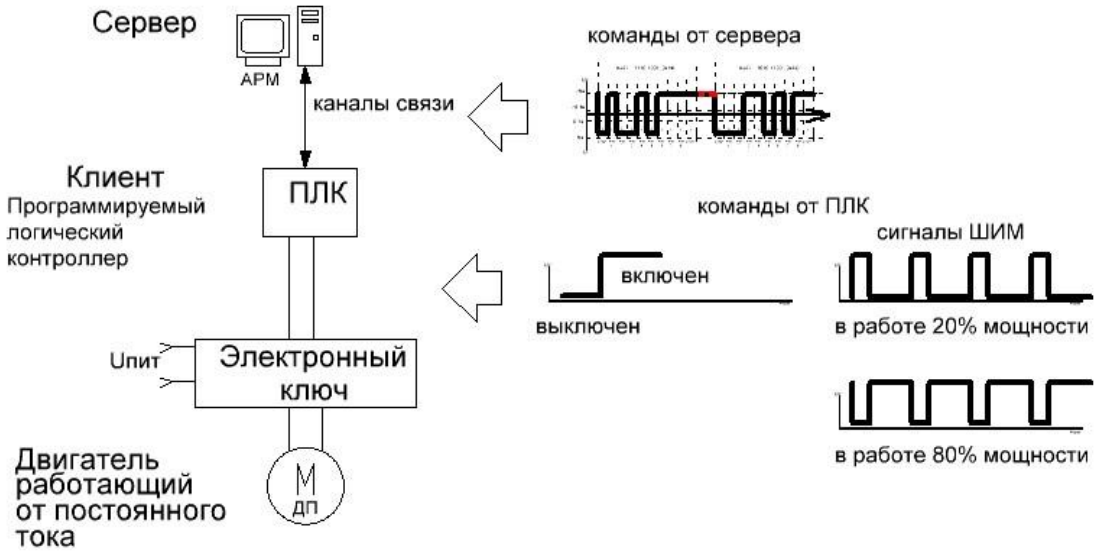


Рисунок 8. Блочная схема управления двигателем постоянного тока

ПЛК - программируемый логический контроллер получает команду от АРМ - автоматизированного рабочего места по каналу связи. Это может быть и байт и группа байт. Зависит от алгоритма и задачи решаемой общей системой.

Получаемый сигнал - это сигнал стандарта RS485. Протокол канальный и сетевой так же зависит от общей задачи.

Нам важно, что контроллер в любом случае вырабатывает только два типа сигналов для управления двигателем постоянного тока:

- Включить-выключить. Если режим работы двигателя имеет только два состояния.
- Сигнал включить-выключить но со скважностью. Если режим работы двигателя должен иметь разные обороты вращения. В этом случае используется управление по методу ШИМ - широтно-импульсной модуляции;

Сигналы ШИМ [Л13] очень широко используются для управления скоростью двигателя постоянного тока.

Необходимо отметить, что длительность сигналов ШИМ, должна быть сопоставима с числом оборотов двигателя, или желательно меньше. В этом случае интегральное значение суммарных токов распределяется на меньшем угле вращения ротора что формирует его плавное вращение.

Интересно отметить предложение в [Л14], для стабилизации вращения при переменной нагрузки. Вместе с тем, программное обеспечение подобных систем стабилизации выполняется на ПЛК и требует больших затрат. Но самое главное, оно не может быть перенесено на другие типы ПЛК и микроконтроллеры.

Аналогично и в работе [Л15] стабилизация достигается методом нейронных сетей. Но, все эти системы предусматривают программирование микроконтроллеров, и соответственно большие трудозатраты и не переносимость на другие типы изделий.

Множество публикаций, смотрите [Л16] [Л17] [Л18], по управлению двигателями постоянного тока имеют общую задачу - программирование микроконтроллера. И почему то, умалчивается о проблемах программирования микроконтроллеров и отладки микропрограмм.

Кроме того, метод ШИМ используется и для регулировки других, относительно медленных процессов, например в терморегуляторах на основе угольных или проволочных нагревательных элементов [Л19].

Выводы по главе 2.

В контексте далее изложенных предложений, для сигналов управления физическими устройствами достаточно сигнала типа ШИМ.

Все сигналы, передаваемые от сервера к клиенту являются вспомогательными.

И требуется преобразователь, роль которого выполняет обычно микроконтроллер. Соответственно появляются затраты на программирование микроконтроллера и формируется зависимость программы к типу микроконтроллера.

Глава 3. Синтез сигнала сингулярной сигнальной модели. Расчет точности для управления двигателем постоянного тока.

Рассмотрим вариант использования прямого управления сигналами интерфейса RS485 старт-стопного канального вида. На рисунке 9 представлена последовательность 5 байт x00. Отрицательное значение сигнала при интегрировании не используется, по этому указываем только U+. Нам можно передавать коды сигналов x00 до xFF (0000-0000 и 1111-1111 в битовой записи).

Весь ансамбль состоит из 9 сигналов x00, x01, x03, x07, x0F, x1F, x3F, x7F, xFF.

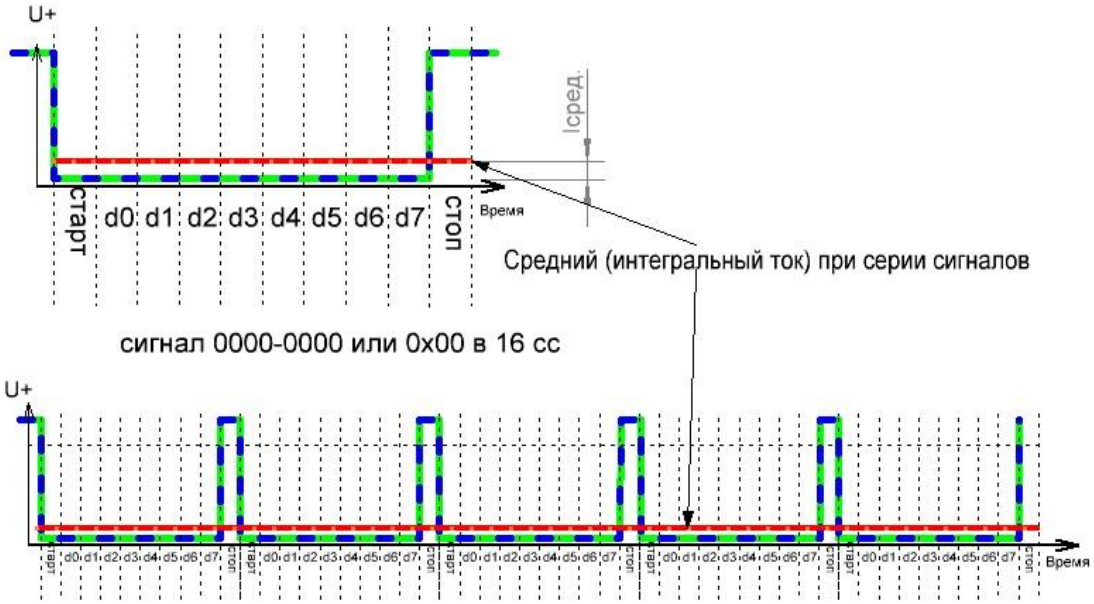


Рисунок 9. Вид сигнала в серии из 5 байт с значением 0x00

На рисунке отображается по оси X-время, по оси Y- напряжение на выходной шине D+ относительно общей шины питания. Противофазное напряжение формируется на D- [Л1]. Средний ток определяется, как средний ток в нагрузке на шине RS485.

Очевидно, что средний (интегральный) ток на нагрузке при последовательности всех x00 будет минимальный. И максимальный ток, можно сформировать при передаче последовательности сигналов xFF.

Промежуточный ток формируют сигналы имеющегося ансамбля.

Необходимо, так же учитывать и инверсию отсчета сигнала, т.к. на рисунка сигналы младших бит расположены слева, а записываются

справа. В этом случае ансамбль из 9 сигналов состоит из: xFF, xFE, xFC, xF7, xF0, xE0, xC0, x70, x00

Это означает, что обе одинаковых составляющих из выборки в сумме равны 0xFF.

1. xFF+x00 = xFF
2. xFE+x01 = xFF
3. xFC+x03 = xFF
4. xF7+x07 = xFF
5. xF0+x0F = xFF
6. xE0+x1F = xFF
7. xC0+x3F = xFF
8. x70+x7F = xFF
9. x00+xFF = xFF

На рисунке 10 показана последовательность для xE0 (1110-0000).

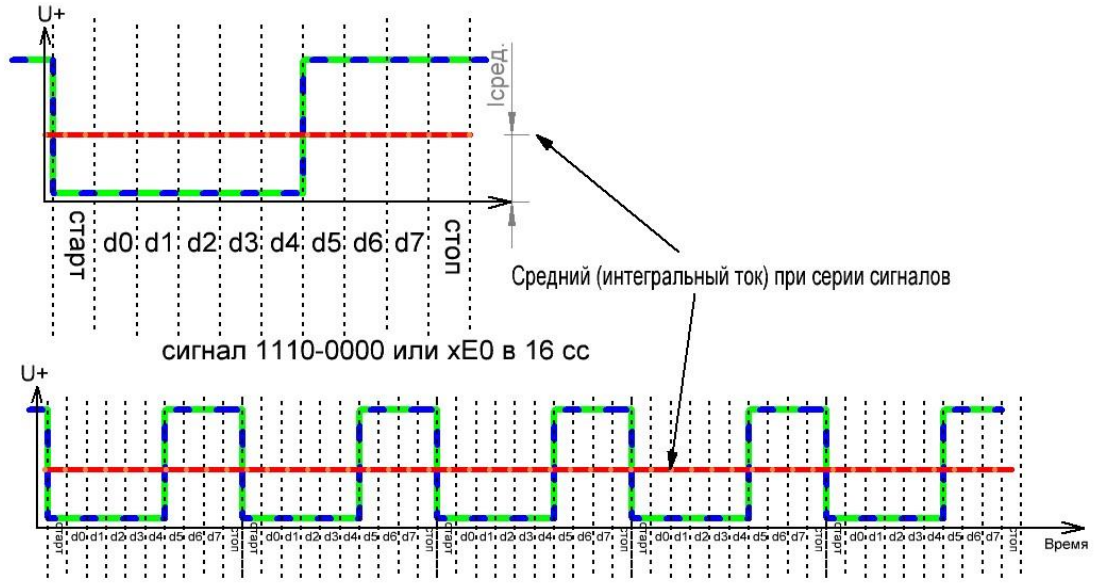


Рисунок 10. Вид сигнала в серии из 5 байт с значением xE0

Подсчет всех возможных уровней интегрального тока проводим по тактовым интервалам. Период состоит из 8 бит, 1-го стартового интервала и 1-го стопового интервала. Всего 10. Время передачи сигнала может меняться от 1-го тактового интервала – это стартовый интервал и передача кода x00. До 9 – передается код xFF. Отношение сигнала к периоду (коэффициент скважности) варьируется от 1/ 10 и до 9/ 10. Итого 9 значений выходного тока. На рисунке 11 показана зависимость выходного тока от передаваемого кода.

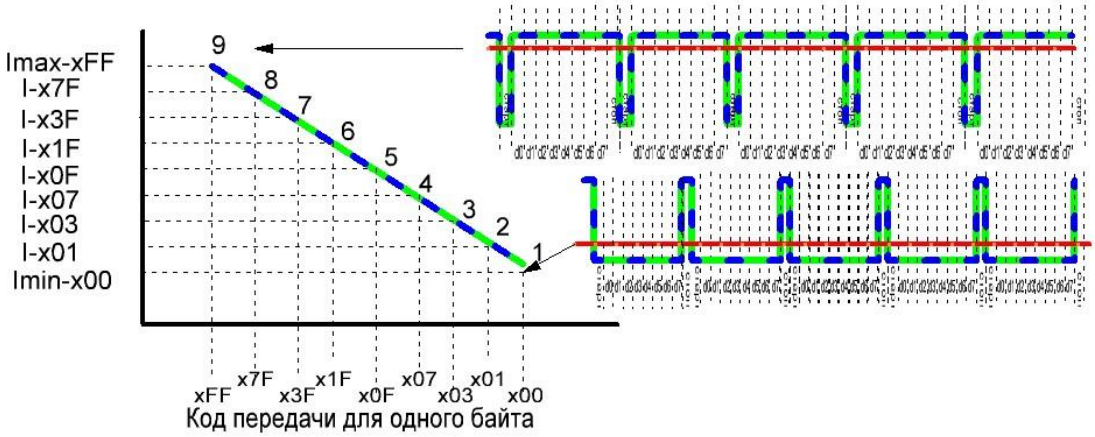


Рисунок 11 График зависимости срднего (интегрального) тока от разных кодов

- **Сингуля́рность** от лат. *singularis* — единственный, особенный: сингулярность в философии (от лат. *singularis* — единственный) — единичность существа, события, явления;

Для увеличения количества градаций выходного тока необходимо сформировать такой ансамбль сигналов, что бы средний (интегральный) ток был уникальный, т.е. сингулярный.

Рассмотрим формирование среднего тока для группы сигналов из 8 байт. Период для учета среднего тока состоит из 8 байт, это 10*8

тактовых интервалов. Получается, что градация скважности меняется от 8 /80 до 72/80 на интервале в 8 байт.

На рисунке 12 показаны градации выходных токов для управляющей группы из 8 байт.

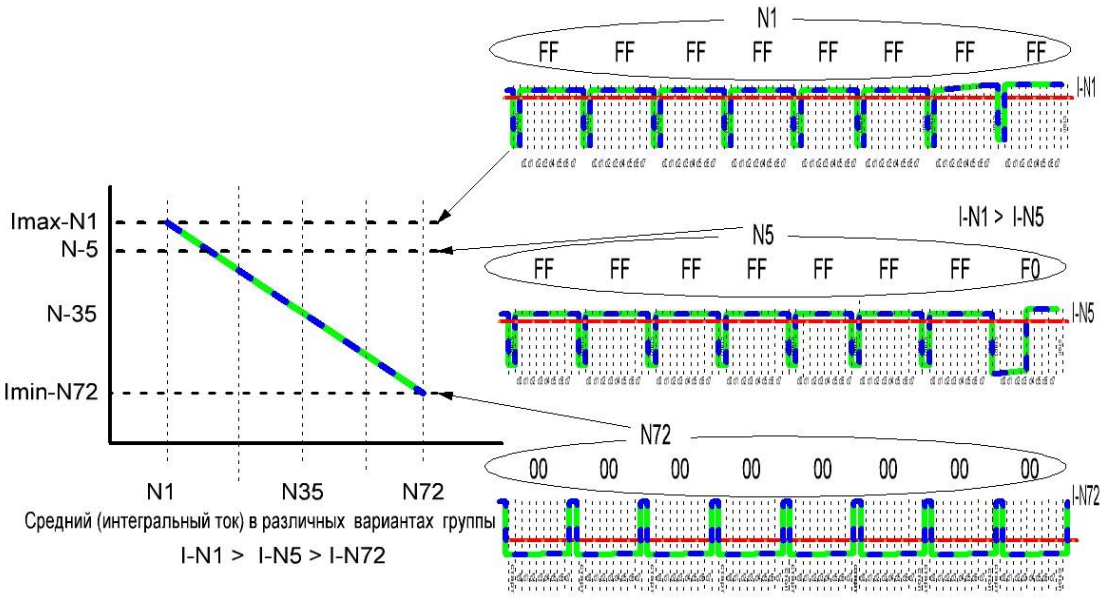


Рисунок 12. График зависимости срднего (интегрального) тока в группе из 8 байт

Использование группы из 8 байт увеличивает градацию среднего (интегрального) тока с 9 до 8*9=72 ступеней. А использование 80 байт в группе увеличивает градацию до 80*9=720 ступеней. Таким образом, увеличивая количество байт в группе, точность можно повысить до любой заданной величины.

Данная группа названа **сингулярной сигнальной моделью управления или воздействия,** т.к. она уникальна для определенного физического устройства на определенном времени.

Однако, нельзя забывать о физических устройствах которые создают свои ограничения. Рассмотрим ограничения предложенной модели далее.

Расчет ограничений на синтезируемые сигналы на примере модели управления двигателем постоянного тока.

Диапазон частот ШИМ определяется по многим критериям и лежит обычно в интервале от 10 до 100 килогерц. [Л22..Л24] В приведенных ссылках на литературу много критериев выбора, однако данная тема выходит за рамки исследований. Просто посчитаем по верхнему диапазону.

Рассчитаем необходимый тактовый интервал To группы в 8 байт для управления двигателем постоянного тока с частотой ШИМ в 100 кгц.

Нас интересует возможно ли использовать сигналы старт-стопного канала для замены сигнала ШИМ на группу сигналов.

F=100 килогерц - T-периода.шим= 10 микросекунд

Тп.шим/N8= 10/8 = 1.25 микросекунд на передачу байта.

To-rs485= 1.25/10= 125 наносекунд - такой To - требуется создать от RS485.

Это возможно. Например адаптеры [Л21] работают на скоростях не менее 2000 кбит. в секунду. Автор экспериментально установил что работа продолжается до 2800 кбит. в секунду.

Приведем расчетный ряд скоростей и соответствующих им To:

- 9.6 кбит. в секунду, To=104 микросекунды
- 115 кбит. в секунду, To=8.6 микросекунд;
- 500 кбит. в секунду , To= 2 микросекунды;
- 1000 кбит в секунду, To=1 микросекунда;
- 2000 кбит в секунду, To=500 наносекунд;

На рисунке 13 показано графически масштаб и значение пересчета ШИМ

Ряд скоростей старт-стопного канала начинается от 300 бит в секунду и далее: 300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200.

Однако, выбор скорости определяется программно, и установить можно любую скорость в интервале определяемом аппаратными средствами.

Следовательно, для управления ДП (двигатель постоянного тока) с точностью в 72 ступени и частотой ШИМ 100 кгц требуется установить скорость передачи в 833 кбит в секунду.

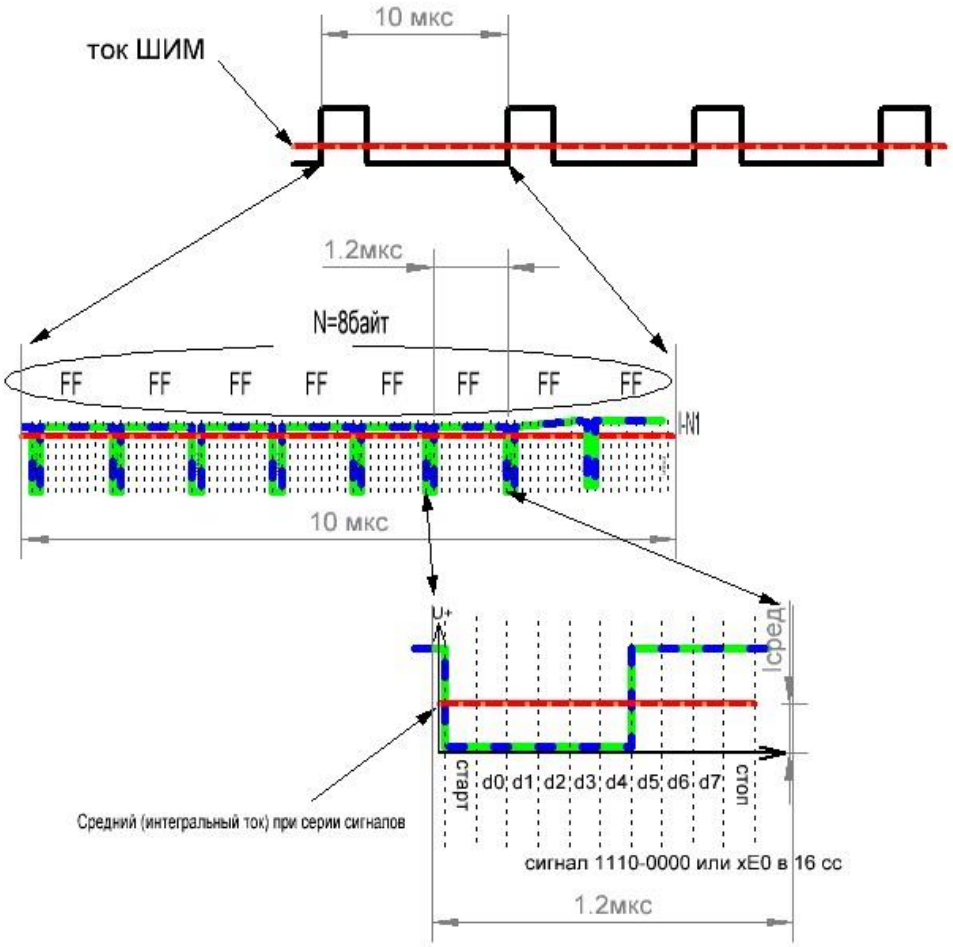


Рисунок 13. Масштабы сигналов для ШИМ=10кгц и группы в 8 байт старт-стопного сигнала

Рассмотрим вероятность ошибки передачи группы сигналов, которая считается как одна команда.

Вероятность появления ошибки бита равновероятна для всех бит, по этому она суммируется.

Ранее во 2-ой главе были даны параметры вероятности ошибки для канала на 1-ин бит из 1 000 000 передаваемых бит.

Следовательно, для группы из 8 байт в количестве 8*10=80 бит, вероятность ошибки передачи 1-ой группы составляет 1000000/80=12500.

На 12500 передаваемых групп-команд имеем только 1 ошибочную группу. Эта величина достаточна мала, и не может влиять на основные параметры управления двигателем постоянного тока.

Однако, необходимо учитывать этот параметр, если линия длинная и "медь".

Таким образом, для управления двигателем постоянного тока по методу ШИМ достаточно передавать в цикле группу сигналов определенного кода с определенной скоростью.

Повторим предложенное определение: - группа сигналов названа сингулярной сигнальной моделью, т.к. она уникальна и формирует определенное воздействие в определенное время.

Для получения групп сигналов управления, можно использовать два варианта их создания:

- подготовить их заранее в виде файла, и далее передавать по запросу алгоритма управления;
- подготовить подпрограмму синтеза и далее выполнять подпрограмму по запросу алгоритма управления. Например, в виде отдельного потока;

Оба варианта, имеют свои плюсы и минусы. Но всегда надо учитывать "вычислительную мощность" используемого компьютера.

Автором подготовлено программное обеспечение для управление двигателями постоянного тока, которое использовалось для построения тренажера.[Л39]

Расчет ограничений компьютера на примере модели "кванта учета" вычислительной мощности;

Для уяснения модели "квант учета", вспомним что такое "лошадиная сила".[Л25] Это "внесистемная" единица измерений.

Небольшая историческая справка.

Термин "лошадиная сила" был изобретён инженером Джеймсом Ваттом. Ватт жил с 1736 по 1819 годы и является одним из самых известных и заслуженных учёных за его работу по повышению эффективности паровых двигателей. Мы также произносим его фамилию почти каждый день, когда говорим о 60-ваттных лампочках.

История гласит, что Ватт работал на угольной шахте, где уголь поднимался из шахты с помощью пони. Ватт хотел найти способ утвердить и говорить о силе, производимой этим животным.

Он обнаружил, что в среднем пони могла сделать 22 000 фут-фунтов работы за одну минуту. Затем он увеличил это число на 50 процентов и привязал измерение одной лошадиной силы к 33 000 фут-фунтам работы за одну минуту.

Эта внесистемная единица измерения проделала свой путь в течение столетий, и теперь ей измеряется производительность Вашего автомобиля, газонокосилки, цепной пилы и даже в некоторых случаях пылесоса.

Сегодня также существует метрическое измерение лошадиной силы - не вдаваясь в подробности, она равна около 735,5 Ватт, или 75 кгс·м/с (работа, которая производится при поднятии груза массой 75 кг на высоту в 1 метр за 1 секунду, и всё это с учётом земного значения ускорения свободного падения).

Воспользуемся для формирования внесистемной единицы измерения мощности компьютера тем же принципом, а в качестве пони будет работать "стандартная программа" или в авторской интерпретации "квант учета".

Для наглядного представления работы "стандартной программы-кванта учета":

- программа должна функционировать на экране;

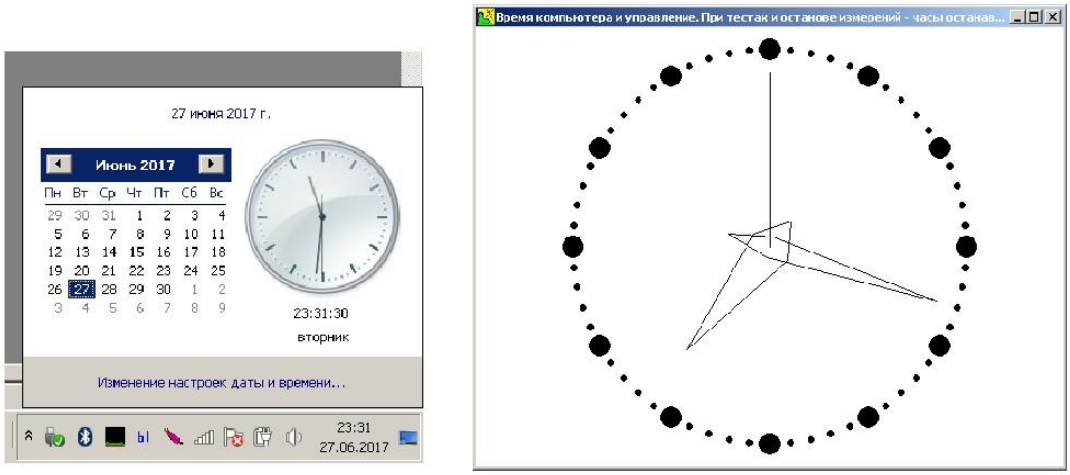


Рисунок 14. Часы на экране компьютера и часы =кванта учета=

- иметь численные и постоянные временные параметры своей работы;
- иметь понятный и однозначный алгоритм работы.

Автором предлагается в качестве "кванта учета" программа часов на экране компьютера системы Windows. Всем она известна, если кликнуть на время в компьютере.

Фрагмент этих часов на экране от Windows и от автора представлен на рисунке 14.

На рисунке 15 приведен скан экрана работающих "квантов учета" Подробное описание, текст и применение этой программы приведено в [Л26]

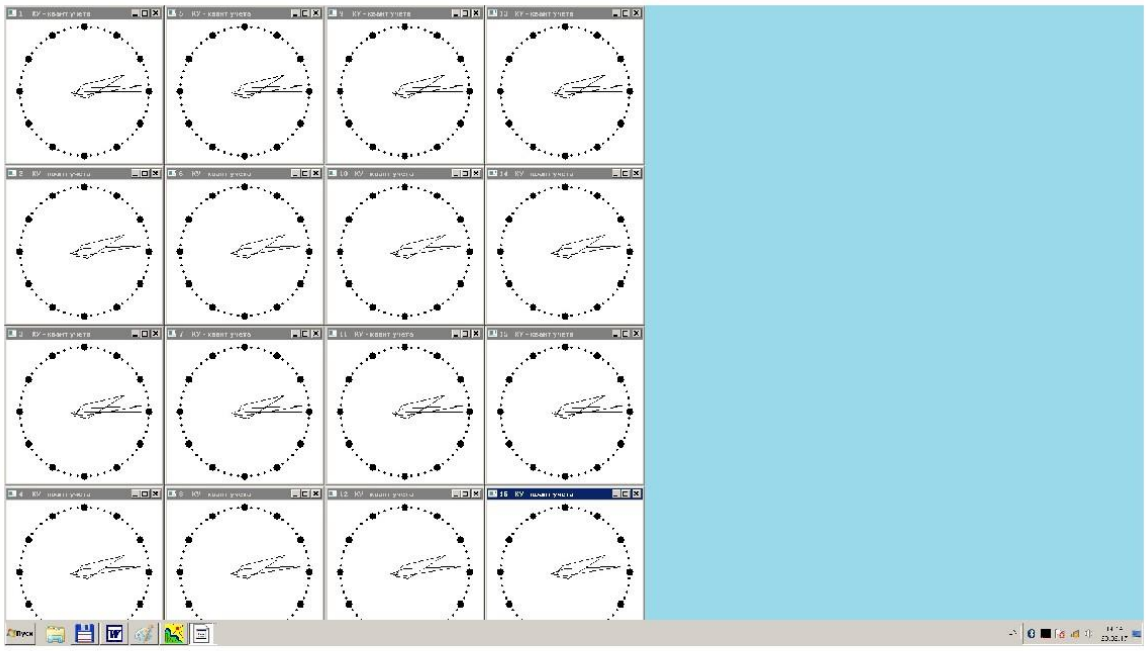


Рисунок 14. Экран компьютера отображает 16 ку - программ-часов

Методика определения "мощности" компьютера заключается в том, что бы определить сколько "квантов учета" т.е. программ-часов может **одновременно** работать на данном компьютере.

Слово "одновременно" специально выделено, т.к. это ключевой момент определения "мощности компьютера".

Чем больше работает программ-часов, тем больше "вычислительная мощность" данного компьютера.

Естественно, что "пони" т.е. программ-часы могут быть разные и иметь разные способности и потребности в ресурсах.

Для однозначности подсчета программа написана с открытым кодом. Ее текст доступен, извлекается из текста приложения, и алгоритм доступен для повторения на других языках

программирования. Использован язык программирования Форт [Л27,Л28].

Для приведенной программной модели КУ кванта учета, компьютер в 1.8 ггц с оперативной памятью в 4 ГБ имеет около 100 ку.

Модель КУ позволяет оценить "вычислительную мощность" и оставить ее запас на случай включения дополнительных и тестовых программ.

Например, если до включения Вашей программы было 100 КУ, а после включения осталось только 20 КУ, то это означает что 80% Ваших ресурсов используется Вашей программой. И вторую такую же программу Вы включить без специальных средств операционной системы не сможете.

Под специальными средствами операционной системы, понимается работа с распределенными ресурсами по времени, по файловой системе, по оперативной памяти. Например, одна программа "ожидает" а другая в "работе".

Таким образом, включая программы-часы КУ и зная их максимальное работающее количество, появляется возможность оценить запас вычислительной мощности.

Выводы по главе 3.

Продемонстрирована теоретическая возможность использования прямого управления сигналами старт-стопного вида для управления физическими устройствами по типу сигналов ШИМ.

Приведены расчеты по ограничениям на частоту ШИМ и соответствующую ей скорость старт-стопного канала.

Приведена методика определения "мощности компьютера" при использовании компьютера для прямого управления по каналам связи.

Глава 4. Описание управления двигателем постоянного тока с использованием сигналов сингулярной сигнальной модели.

Блочная схема.

На рисунке 15 представлена блочная схем управления двигателем постоянного тока.

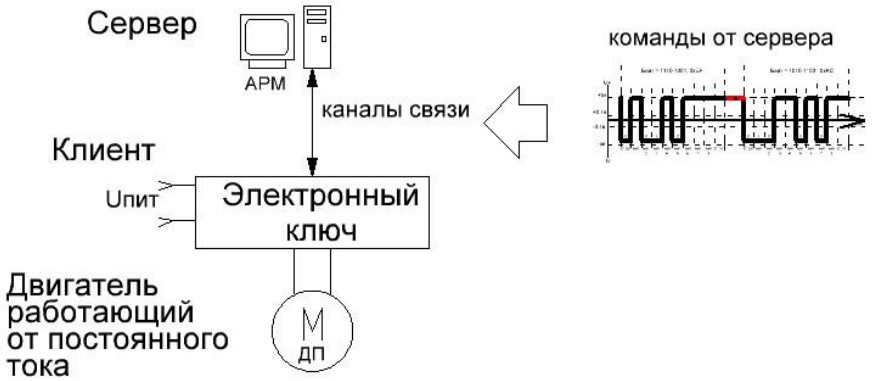


Рисунок 15. Блочная схема управления двигателем постоянного тока

Схема повторяет рисунок 8, но без ПЛК - программируемого логического контроллера. В контексте всех изложенных предложений о "лишнем звене" микроконтроллере, блочная схема получается очень простой, всего три блока: компьютер, электронный ключ и двигатель постоянного тока.

Принципиальная схема управления по интерфейсу старт-стопного RS485 канала связи электродвигателем постоянного тока.

В промышленности и в реальных условиях эксплуатации одно из важных требований к схемам управления это требования гальванической развязки. В данном исследовании этот вопрос не изучается, поскольку существуют общие решения задачи гальванической развязки.

Одно из таких решений для приема информации по CAN-интерфейсу применено автором и описано в [Л30].

Схема управления без гальванической развязки по интерфейсу старт-стопного RS485 канала связи электродвигателем постоянного тока представлена на рисунке 16.

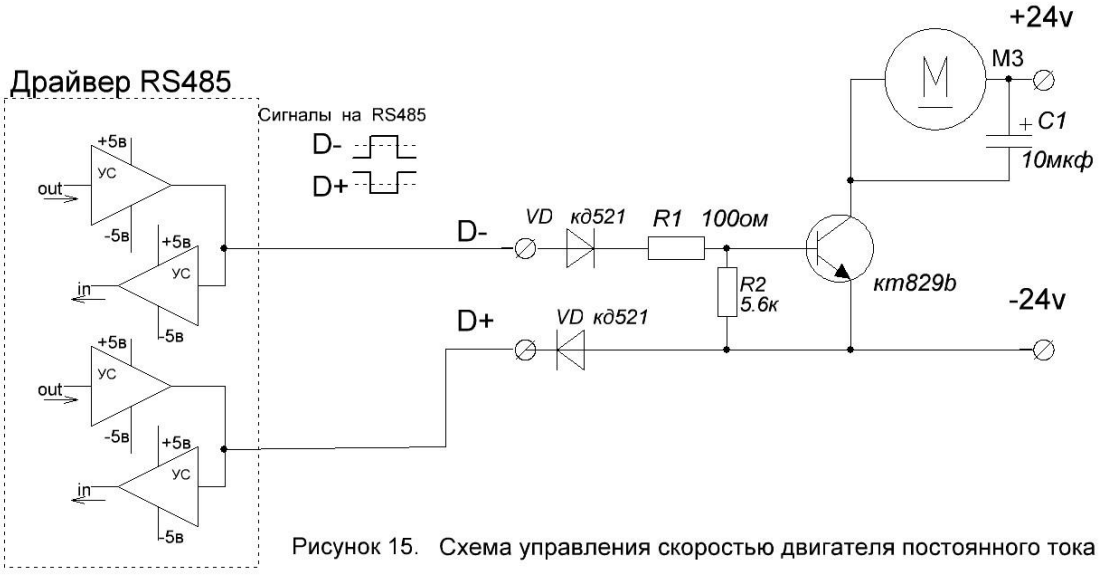


Рисунок 15. Схема управления скоростью двигателя постоянного тока

Принципиальная схема состоит из диодов VD1 VD2, разделяющих положительный и отрицательный сигнал импульса для управления электронным ключом КТ829. Ключ открывается от положительной части импульса. Сопротивления R1 и R2 создают требуемый ток, для

открытия силового ключа при положительной части импульса, и режим закрытия ключа при отрицательной части импульса.

Силовой ключ выполнен на транзисторе типа КТ829. Конденсатора С1 сглаживает импульсные пульсации электронного ключа на двигателе постоянного тока. Напряжение питание зависит от типа двигателя и возможностей транзистора и в приведенной схеме возможно до 45..100 вольт в зависимости от буквы типа электронного ключа.

На схеме показана структура драйвера RS485 в виде отдельного блока.

Показаны силовые ключи, которые коммутируют выходное напряжение.

В схемах связи по RS485 обычно используются так называемые сопротивления согласования шлейфа [Л29]. В данной схеме управления они не нужны, т.к. через них будет уходить ток управления для транзисторного ключа.

Обычно, сопротивления согласования подключаются по мере необходимости и коммутируются специальными переключателями.

Расчет сопротивлений определяющих режим работы силового ключа, проведен по данным стандартных параметров RS458 из [Л31,Л32]. Эти данные сведены в таблицу 1.

Табл. 1 . Параметры интерфейса RS-485, установленные стандартом

Параметр	**Условие**	**Мин.**	**Макс.**	**Единица измерения**
Выходное напряжение передатчика без нагрузки	$R_{нагр} = 0$	1,5 -1,5	6 -6	В В

Выходное напряжение передатчика с нагрузкой	$R_{нагр} = 54\ Ом$	1,5 -1,5	5 -5	В В
Ток к. з. передатчика	К. з. выхода на источник питания +12 В или на -7 В	-	±250	мА
Длительность переднего фронта импульсов передатчика	$R_{нагр} = 54\ Ом$ $C_{нагр} = 50\ пФ$	-	30	% от ширины импульса
Синфазное напряжение на выходе передатчика	$R_{нагр} = 54Ом$	-1	3	В
Чувствительность приемника	При синфазном напряжении от -7 до +12 В	-	±200	мВ
Синфазное напряжение на входе приемника		-7	+12	В
Входное сопротивление приемника		12	-	кОм
Максимальная скорость передачи	Кабель длиной: 12 м 1200 м	10 100	-	Мбит/с Кбит/с

Передатчик должен выдерживать режим короткого замыкания как между своими выходами, так и замыкание их на +12 В или -7 В.

Параметры транзистора приведены [Л33]. Расчеты ключей на биполярных транзисторах широко освещен в литературе [Л34,Л35,Л36]

Как видно из приведенных данных, напряжения и тока от RS485, с избытком хватает для работы мощных ключевых транзисторов.

Расчет мощности рассеяния и теплоотводящих параметров радиаторов так же выходит за рамки данной исследовательской работы, поскольку хорошо освещен в литературе. [Л37,Л38]

Достоинства и недостатки предлагаемой схемы управления.

Предлагаемая схема на транзисторном ключе, предназначена только для доказательства возможности управления по методу сингулярной сигнальной модели.

Достоинства:

- Приведенная не затратная и не сложная схема экспериментально доказывает возможность работы без микроконтроллера и использовать сингулярную модель управления физическими устройствами.
- Схема легко повторяема. Элементы не дефицитны. Ремонтируется и поиск неисправности не сложный.
- Схема масштабируема на разные мощности управления;
- Схему для двигателей постоянного тока без всяких переделок можно использовать для работы с теплонагревателями или источниками света.

Недостатки:

- отсутствует гальваническая развязка;
- отсутствует цепь обратной связи, и не используется возможность приема информации о состоянии работы двигателя постоянного тока; Например, предусмотреть после переданной группы байт для управления принимать один байт о состоянии;
- Нет индикации о режимах работы и возможных авариях, например из-за перегрева транзистора;

Выводы.

В представленном исследовании рассматривались следующие вопросы по предложенной сингулярной сигнальной модели для управляющих систем с конечным набором команд:

- В первой главе дан общий обзор построения систем управления с различной иерархией. Рассматривалось обоснование для исследования "узкого места" программирование микроконтроллеров. Указывалось на "лишнее звено" в некоторых случаях управления;
- Во второй главе определено место исследования – сигналы управления. Дается общий обзор существующих сигналов и шин. Указывается на общность и подобие сигналов по сигнальной шине и сигналов управляющих физическим устройством. Сравниваются сигналы ШИМ и сигналы старт-стопного канала связи.
- В третье главе предлагается, на основе проведенных исследований, расчетная управляющая конструкция сингулярной сигнальной модели. Приводятся расчеты для управления двигателем постоянного тока только сигналами старт-стопного канала связи. Приводятся расчеты ограничений на управление предложенной моделью. Приводится методика оценки вычислительной мощности компьютеров, которые планируются использоваться для управления с программным управлением и сингулярной сигнальной моделью. Указывается на необходимость предварительной оценки затрат вычислительной мощности при использовании сингулярной сигнальной модели.

- В четвертой главе приведено экспериментальное подтверждение предлагаемой сингулярной сигнальной модели для создания систем управления. Приведена принципиальная схема эксперимента. Приводится анализ схемного решения и указывается на экспериментальный характер схемы.

В целом, все исследование направлено, на предложение заменить «микропрограммное» программирование, на программирование общей задачи. Использовать языки высокого уровня и открытый исходный код. Предлагается отказаться от микропроцессорных решений и оставить только силовые ключ для управления физическими объектами. Использовать «мощность» больших компьютеров и высокоскоростные каналы связи.

Данные предложения увеличивают ремонтопригодность и надежность систем управления.

Используемая литература и источники интернета:

- Л1: Автоматы Мура http://cxem.net/beginner/beginner106.php
- Л2: Автоматы Мили: http://masters.donntu.org/2007/fvti/maluk/library/lib11.htm
- Л3: Клиент-серверная модель http://lektsii.org/6-79020.html
- Л4: Модели СУБД: http://mirznanii.com/a/122195/tekhnologiya-klient-server
- Л5: SCADA системы http://bourabai.ru/dbt/scada.htm
- Л6: OPC - стандарты: https://www.itweek.ru/industrial/article/detail.php?ID=59064

- Л7: RS485 рекомендованный стандарт: http://www.softelectro.ru/rs485.html
- Л8: 7-ми уровневая модель: http://www.intuit.ru/studies/courses/636/492/lecture/11116
- Л9: Синхронные каналы связи: http://www.jetinfo.ru/stati/uslugi-operatorov-svyazi-dlya-sozdaniya-korporativnykh-setej-peredachi
- Л10: Процессы и методы передачи данных: http://kom-seti.narod.ru/index.files/3.htm
- Л11: Использование протокола I2C: http://avr.ru/ready/inter/i2c/i2c
- Л12:Описание шины I2C: http://www.itt-ltd.com/reference/ref_i2c.html
- Л13: Виды импульсной модуляции: http://electrono.ru/2-3-2-vidy-impul-snoy-modulyacii-elekpromust
- Л14: СИНТЕЗ СИСТЕМЫ СТАБИЛИЗАЦИИ СКОРОСТИ ВРАЩЕНИЯ ДВИГАТЕЛЯ ПОСТОЯННОГО ТОКА С ШИМ В КАНАЛЕ УПРАВЛЕНИЯ НА ОСНОВЕ МЕТОДА РАЗДЕЛЕНИЯ ДВИЖЕНИЙ Н.А. Степанов, В.Д. Юркевич НГТУ: ДОКЛАДЫ АН ВШ РФ июль-декабрь 2011г. http://journals.nstu.ru/all.
- Л15: Нейросетевая реализация и настройка ШИМ-элементов в автоматических системах. Вестник НГТУ: http://journals.nstu.ru/vestnik/catalogue/contents/view_article?id=8004
- Л16:Широтно-импульсная модуляция: https://myrobot.ru/stepbystep/rce_pwm.php
- Л17: Альтернативная ШИМ: http://www.privod-news.ru/November/invert.htm
- Л18: Высокоточный метод ШИМ... http://elib.altstu.ru/elib/books/Files/pv2011_03_01/pdf/055soloviov.pdf

- Л19: Термодат - управление мощностью электронагревателя: http://www.termodat.spb.ru/upravl.htm
- Л20: Описание стандарта RS485: http://ru.wikipedia.org/wiki/RS-485
- Л21: Описание конверторов интерфейсов usb-rs-485. URL: http://icbcom.ru/store/converters/icb-usb-02.html
- Л22: ШИМ определение использование: http://www.texnic.ru/books/electronika/012.html
- Л23: AVR использование ШИМ: http://easyelectronics.ru/avr-uchebnyj-kurs-ispolzovanie-shim.html
- Л24: Схемы с ШИМ: http://artradiolab.ru/3.6.htm
- Л25: Что такое "лошадиная сила": http://howcarworks.ru/
- Л26: Определение вычислительной мощности компьютера http://shabronov_s2.dyn-dns.ru/doc_sh/nagruzika_v3/
- Л27: Язык Форт, версия spf4.exe – автор А. Черезов http://www.forth.org.ru
- Л28: Л. Броуди Начальный курс программирования на языке Форт: http://www.nncron.ru/download/sf.pdf
- Л29: Организация канала связи RS485: https://bolid.ru/projects/iso-orion/communication-channels/rs-485/
- Л30: Схема опторазвязки CAN-линии: http://shabronov_s2.dyn-dns.ru/doc_sh/kmk_tester_imitator_v5/
- Л31: Интерфейсы RS485, RS422, RS232: http://www.bookasutp.ru/Chapter2_3.aspx
- Л32: RS485 рекомендованный стандарт...: http://www.softelectro.ru/rs485.html

- Л33: Технические данные транзисторов кт-829.: http://lib.chipdip.ru/031/DOC001031308.pdf
- Л34: Транзисторный ключ: http://electronicsclub.ru/tranzistornyj-klyuch-chast-1/
- Л35: Программы для расчетов схем: http://e-dep.ru/forum/topic.php?forum=2&topic=13
- Л36: Электронные ключи в моделях на компьютере: http://riostat.ru/elektron_sistem/8_2.php
- Л37: Простой расчет радиаторов для транзисторов: http://meandr.org/archives/26706
- Л38: Графики для расчета радиаторов: http://full-chip.net/soft-dlya-radiolyubitelya/47-prostoy-raschet-ploschadi-radiatora-dlya-tranzistorov-i-mikroshem.html
- Л39: Опытный вариант управления теннисным тренажером: http://shabronov_s2.dyn-dns.ru/temp/tenis_bam_v2/