

**«Санкт-Петербургский государственный электротехнический университет
«ЛЭТИ» им. В.И.Ульянова (Ленина)»
(СПбГЭТУ «ЛЭТИ»)**

Направление	13.03.02 – Электроэнергетика и электротехника
Профиль	Электропривод и автоматика
Факультет	ЭА
Кафедра	РАПС

К защите допустить

Зав. кафедрой

Белов М.П.

**ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА
БАКАЛАВРА**

**Тема: РАЗРАБОТКА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ СИСТЕМЫ
УПРАВЛЕНИЯ АВТОНОМНЫМ ИНВЕРТОРОМ НАПРЯЖЕНИЯ**

Студент		Самойленко А.Н.
	<hr/>	<i>подпись</i>
Руководитель	к.т.н., доцент	Татаринцев Н.И.
	(Уч. степень, уч. звание)	<hr/>
		<i>подпись</i>
Консультанты	ст. преп.	Алексеева О.Г.
	(Уч. степень, уч. звание)	<hr/>
		<i>подпись</i>

Санкт-Петербург

2020

КАЛЕНДАРНЫЙ ПЛАН ВЫПОЛНЕНИЯ ВЫПУСКНОЙ КВАЛИФИКАЦИОННОЙ РАБОТЫ

Утверждаю
Зав. кафедрой РАПС
_____ Белов М.П.
« ___ » _____ 20__ г.

Студент Самойленко А.Н.

Группа 6403

Тема работы: Разработка программного обеспечения системы управления автономным инвертором напряжения.

№ п/п	Наименование работ	Срок выполнения
1	Разработка технического задания	28.04.18 – 30.04.18
2	Анализ технического задания и обзор литературы по исследуемой теме	1.05.18 – 12.05.18
3	Анализ математической модели	14.05.18 – 18.05.18
4	Разработка микропроцессорной системы управления	19.05.18 – 1.06.18
5	Техническо-экономическое обоснование	1.06.18 – 3.06.18
6	Оформление пояснительной записки	4.06.18 – 10.06.18

Студент

Самойленко А.Н.

Руководитель

Татаринцев Н.И.

(Уч. степень, уч. звание)

РЕФЕРАТ

Пояснительная записка 83 с., 5 ч., 19 рисунков, 11 таблиц, 17 источников.

АВТОНОМНЫЙ ИНВЕРТОР НАПРЯЖЕНИЯ, МИКРОПРОЦЕССОРНАЯ СИСТЕМА УПРАВЛЕНИЯ, STM32, ЭЛЕКТРОПРИВОД, ПРОГРАММИРОВАНИЕ МИКРОКОНТРОЛЛЕРОВ, НЕЧЕТКИЙ РЕГУЛЯТОР.

Объектом исследования и разработки в данной выпускной квалификационной работе является система управления АИН.

Предметом исследования являются способы управления и их микропроцессорная реализация.

Цель работы – разработка программного обеспечения системы управления АИН в соответствии с техническими требованиями, указанными в задании на ВКР.

В процессе работы использована математическая модель АИН и его системы управления, описаны программные средства, используемые в разработке и моделировании.

ABSTRACT

The explanatory note 83 pages, 5 parts, 19 figures, 11 tables, 17 sources.

AUTONOMOUS VOLTAGE INVERTER, MICROPROCESSOR CONTROL SYSTEM, STM32, ELECTRIC DRIVE, PROGRAMMING MICROCONTROLLERS, FUZZY REGULATOR

The object of study is the control system of the AVI.

The subject of research is controlling methods and their microprocessor implementation.

The purpose of the practice is to develop software for the AVI control system in accordance with the technical requirements specified in the assignments.

In the process, a mathematical model of the AVI and its control system was used, software tools used in the development and modeling are described.

ПЕРЕЧЕНЬ СОКРАЩЕНИЙ И ОБОЗНАЧЕНИЙ

ДПТ – двигатель постоянного тока ВЗ – воздушный зазор

АД – асинхронный двигатель

ЧП – частотный преобразователь

СУ – система управления

МПСУ – микропроцессорная система управления

ПО – программное обеспечение

АИН – автономный инвертор напряжения

ВБ – вентильный блок

ДС – датчик скорости

ДТ – датчик тока

РС – регулятор скорости

РТ – регулятор тока

СУВБ – система управления вентильным блоком

ЭВМ – электронно-вычислительная машина

АИМ – амплитудно-импульсная модуляция

ШИМ – широтно-импульсная модуляция

МП – микропроцессор

МК -микроконтроллер

АЦП – аналого-цифровой преобразователь

ПИД – пропорционально-интегрально-дифференциальный регулятор

БНА – блок нечеткой адаптации

ОГЛАВЛЕНИЕ

Оглавление	7
ВВЕДЕНИЕ	9
1. Объект управления.....	12
2. Обзор основных способов регулирования напряжения	15
2.1. Скалярная форма представления ШИМ	16
2.2. Векторная форма представления ШИМ	16
3. Микропроцессорная реализация	20
3.1. Особенности микропроцессорной реализации.....	20
3.2. Основные способы микропроцессорной реализации ШИМ	23
3.2.1. Программный способ формирования импульсов	24
3.2.2. Аппаратный способ формирования импульсов	25
3.3. Изменение частоты	28
3.3.1. Первый способ	29
3.3.2. Второй способ.....	30
3.3.3. Третий способ	31
3.3.4. Четвертый способ	32
3.4. Регулятор	35
3.5. Настройка регулятора	37
4. Программное обеспечение.....	45
4.1. Определение требований к ПО	45
4.1.1 Требования к функционалу и безопасности.....	45
4.1.2. Требования к эффективности.....	47
4.1.3 Требования к интерфейсу.....	48
4.2 Проектирование ПО.....	50
4.3 Кодирование ПО	54
4.3.1 Настройка микропроцессора.....	54
4.3.2 Кодирование блока управления вентилями	61
4.4. Верификация ПО.....	66

5. Экономическое обоснование ВКР	70
5.1 Концепция	70
5.2 График выполнения ВКР	71
5.3 Себестоимость разработки	71
5.3.1 Затраты на расходуемые материалы.....	72
5.3.2 Основная заработная плата	72
5.3.3 Дополнительная заработная плата.....	73
5.3.4 Затраты на электроэнергию для технологических целей	73
5.3.5 Затраты на социальные нужды	74
5.3.6 Затраты на накладные расходы.....	75
5.3.7 Амортизационные отчисления.....	75
5.3.8 Себестоимость	77
5.4. Заключение по главе	77
ЗАКЛЮЧЕНИЕ	78
ПРИЛОЖЕНИЕ А	81

ВВЕДЕНИЕ

В современном производстве стали широко применяться системы автоматизации, инструментом которой стали двигатели постоянного и переменного тока (ДПТ). При этом, начиная с конца прошлого столетия, наблюдается постепенный переход к использованию двигателей переменного тока, в особенности асинхронные двигатели (АД) с короткозамкнутой обмоткой.

Этот переход в первую очередь обусловлен простотой конструкции, обслуживания и эксплуатации, высокой надежностью и низкой стоимостью производства АД. Повышению надежности и удешевлению обслуживания поспособствовал отказ от использования щеточно-коллекторного аппарата. При этом с переходом на АД, остается потребность в точном регулировании частоты вращения. При использовании ДПТ в зависимости от технологической задачи это решалось с помощью реостатного регулирования либо тиристорных выпрямителей. При использовании АД для обеспечения плавного изменения скорости требуется источник переменного напряжения и переменной частоты, именуемый частотным преобразователем (ЧП). В упрощенном виде ЧП представляет собой управляемый или неуправляемый выпрямитель в связке с управляемым инвертером напряжения либо тока.

Немаловажным фактором, стимулирующим переход к использованию двигателей переменного тока, и АД, в частности, стало развитие и удешевление устройств силовой электроники, связанного с созданием быстродействующих и энергоэффективных MOSFET-транзисторов и транзисторов с изолированным затвором. Развитие и удешевление микропроцессорной техники привело к замене аналоговой системы управления (СУ) и переходу к микропроцессорной системе управления (МПСУ).

В соответствии с [1], выполнение данной ВКР направлено на систематизацию и расширение полученных знаний в области преобразовательной техники, цифровой и аналоговой электроники, микропроцессорной технике и теории

автоматического управления, привитие практических навыков в решении задач профессиональной деятельности, в том числе получение опыта работы с микроконтроллерами семейства ARM, в моделировании работы микропроцессорных устройств и устройств силовой техники.

Таким образом, объектом исследования в ВКР является система управления АИН. Предметом исследования являются способы управления и их микропроцессорная реализация.

Целью ВКР является разработка программного обеспечения СУАИН в соответствии с техническими требованиями, указанными в задании на ВКР.

Согласно [2] программное обеспечение представляет собой совокупность компьютерных программ и документов, необходимых для эксплуатации этих программ. Процесс его разработки включает в себя следующие процессы: определения требований к ПО, проектирование ПО, кодирование ПО, его интеграция и верификация.

Исходя из всего вышесказанного, для разработки ПО необходимо решить следующий ряд задач:

- рассмотрение алгоритмов управления вентильным блоком АИН;
- рассмотрение микропроцессорной реализации СУ ВБ;
- рассмотреть СУ АИН и его микропроцессорную реализацию;
- определение требований к ПО;
- проектирование ПО;
- кодирование ПО;
- интеграция и верификация ПО методом компьютерного моделирования.

Уникальность задач и их решений заключается в первую очередь в том, что в данной работе проделаны все этапы создания программного обеспечения. Обзор различных источников показал, что на сегодняшний день нет литерату-

ры, в которой бы рассматривались все этапы создания современного ПО для устройств преобразовательной техники. Чаще всего все ограничивается обобщенными алгоритмами работы устройств и, в отдельных случаях, реализацией СУ аналоговыми средствами. Немаловажный вклад в рассмотрение этого вопроса был сделан в работе [3], некоторые результаты и рассуждения, из которой будут использованы в данной ВКР.

Несмотря на то, что темой работы является разработка программного обеспечения, будет рассмотрена и аппаратная часть системы управления, т.к. в процессе будут решаться задачи выбора между аппаратной и программной реализацией блоков системы управления.

1. ОБЪЕКТ УПРАВЛЕНИЯ

Зачастую принято рассматривать АИН как совокупность вентильного блока (ВБ) и систему управления ВБ, работающего в составе электропривода переменного тока. В данной же работе АИН как объект управления представляет собой электротехническое устройство, в состав которого входит электропривод. Таким образом, система управления АИН включает себя систему управления ВБ, устройства сравнения, обратные связи по скорости и току и их регуляторы, но не включает в себя систему измерения. Структурная схема электропривода приведена на рисунке 1.

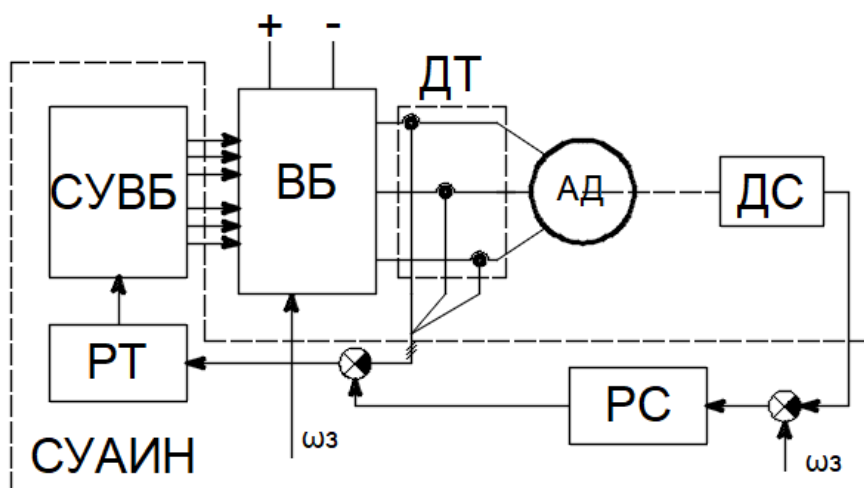


Рисунок 1 – Структурная схема электропривода АД

В электроприводе присутствуют два канала: силовой и информационно-управляющий. В силовой канал входят источник постоянного напряжения, фильтр (в схеме не указаны), вентильный блок, асинхронный двигатель (АД) а также датчики тока (ДТ), как средство системы измерения (СИ). В информационно-управляющий канал входят датчик скорости (ДС), блок регулятора скорости (РС), блок регулятора тока (РТ), сумматоры, и система управления вентильным блоком (СУВБ). СУ АИН представляет собой классическую двухконтурную СУ с обратными связями по току и скорости. При этом задающим сигналом является сигнал требуемой скорости.

В связи тем, что система управления рассматривается с точки зрения разработки ПО, при описании схемы не затронуты вопросы сопряжения с датчиками, гальванической развязки, фильтрации и нормализации, т.к. подобные вопросы должны решаться на этапе проектирования аппаратной части.

Каждый блок регуляторов, в зависимости от задающего сигнала и сигнала обратной связи, формирует управляющий сигнал. Блок регулятора скорости в соответствии с принципом подчинённого регулирования формирует сигнал задания для блока регулирования тока [3]. В свою очередь регулятор тока формирует управляющий сигнал, характеризующий требуемую амплитуду напряжения, на вход блока СУВБ, где на основании полученной информации и требуемой частоты, формируются импульсы управления вентилями.

Помимо основной функции – управления вентильным блоком и обеспечения обратной связи, СУАИН должна решать задачи управления коммутационной аппаратурой и связи с ЭВМ верхнего уровня. Функциональная схема АИН представлена на рисунке 2.

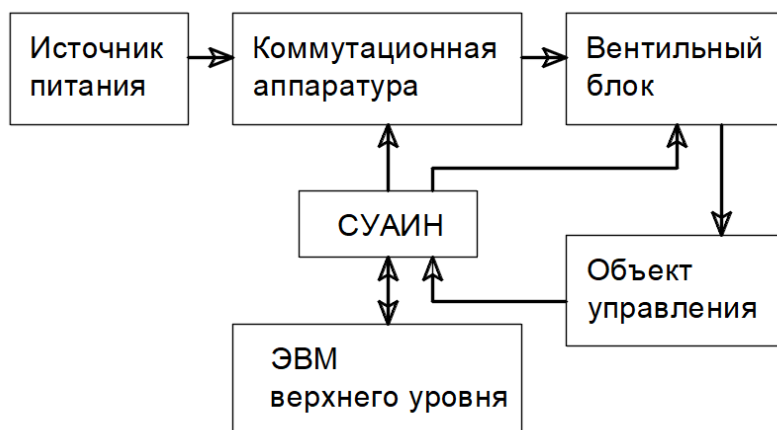


Рисунок 2 – Функциональная схема АИН

Для проектирования системы управления АИН разработчику ПО достаточно рассматривать электрическую схему преобразователя в виде его коммутационной модели. Она основана на представлении управляемых ключей в виде идеальных ключей, находящихся в состоянии «включено» или «выключено».

На рисунке 3 управляемые вентили заменены идеальными ключами. ВБ имеет три плеча, в каждом из которых – два последовательно соединенных ключа.

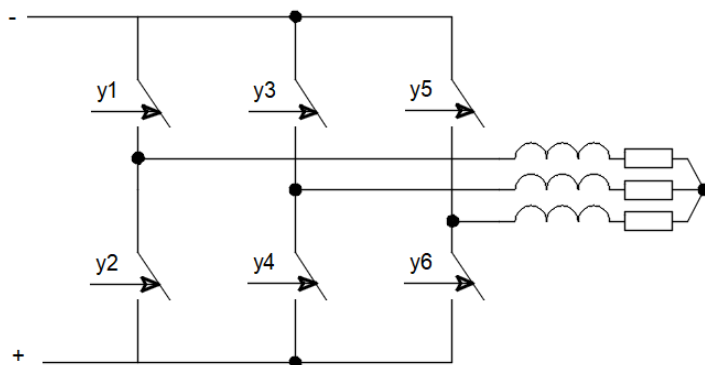


Рисунок 3 – Коммутационная схема ВБ

К средней точке плеча подключен один из выводов соответствующей фазы нагрузки, соединенной в звезду [3]. В соответствии с сигналами управления ключами СУ АИН формирует на нагрузке напряжение. Управляющие сигналы имеют два уровня состояния: 0 либо 1 и называются коммутационными функциями ключей. Чтобы выходное напряжение АИН не зависело от параметров нагрузки, необходимо выполнить следующие условия [4]: в любой момент времени один из ключей в плече инвертора открыт, а другой закрыт. Это так называемый комплементарный режим управления. Значения трех ключей одной вентиляльной группы в один и тот же момент времени будем называть вектором состояния АИН. Способы формирования эти векторов рассмотрены в следующей главе.

2. ОБЗОР ОСНОВНЫХ СПОСОБОВ РЕГУЛИРОВАНИЯ НАПРЯЖЕНИЯ

Выходное напряжение АИН может регулироваться двумя способами: амплитудно-импульсной модуляцией (АИМ) и широтно-импульсной модуляцией (ШИМ). При использовании АИМ изменяется напряжение в звене постоянного тока, а инвертор лишь управляет коммутацией фаз, формируя требуемую частоту; при использовании ШИМ – широтно-импульсным регулированием напряжения в инверторе за счет модуляции напряжения несущей частоты (частоты коммутации силовых ключей) сигналом требуемой частоты. Наиболее распространенным способом модуляции выходных параметров инвертора является ШИМ, ему и посвящена данная работа.

В результате переключения ключей коммутационной схемы (рисунок 3) каждая фаза попеременно подключается к источнику постоянного напряжения (ИПН), обеспечивая тем самым формирование прямоугольных импульсов напряжения. Благодаря фильтрующему свойству обмоток АД, обладающих собственной индуктивностью, на выходе образуется плавно изменяющийся ток. Время подключения фазы к ИПН определяет величину тока. Способ определения этого времени зависит от вида ШИМ. Существуют два основных вида: скалярная и векторная ШИМ. Ряд источников [5], в том числе зарубежных показывает и доказывает эквивалентность скалярной и векторной ШИМ, определяя их как разновидности форм представления. Иными словами, скалярная ШИМ сигнала – есть его рассмотрение во временной области, тогда как в векторной ШИМ сигнал рассматривает в векторном пространстве. Таким образом, принципиальной разницы между векторными и классическими способами ШИМ нет.

2.1. Скалярная форма представления ШИМ

Для реализации скалярной ШИМ требуется высокочастотный опорный сигнал и низкочастотный модулирующий сигнал. Форма модулирующего сигнала является определяющей для гармонического состава выходного напряжения, что в свою очередь влияет на энергоэффективность преобразователя.

Модулирующий сигнал определяет закон изменения длительности импульсов управления (и импульсов выходного напряжения АИН) во времени, а также частоту первой гармоники выходного напряжения. Кроме того, модулирующий сигнал существенно влияет на входные и выходные параметры инвертора напряжения [3]. История развития преобразовательной техники одарила нас большим разнообразием форм модулирующего сигнала. Среди самых простых можно выделить прямоугольный, треугольный, трапецеидальный, синусоидальный, синусоидальный с добавлением третьей гармоники и другие. Вопросы выбора модулирующего сигнала с целью упростить реализацию или наоборот усложнить, чтобы повысить энергоэффективность АИН, посвящено множество статей и даже диссертаций, так что вопрос выбора модулирующего сигнала остается за пределами данной ВКР, перекладывая всю ответственность разработчикам алгоритмов. Для примера будет реализована классическая синусоидальная ШИМ.

Опорный сигнал определяет период пульсаций. Чем он меньше, тем эффективней используются фильтрующие свойства нагрузки. С другой стороны, его уменьшению приводит к увеличению коммутационных потерь и ограничено максимальной частотой работы транзисторов.

2.2. Векторная форма представления ШИМ

В отличие от скалярной формы представления ШИМ, векторная рассматривает сигнал не во временной области, а в векторном пространстве или ком-

плексной плоскости. Так как напряжение трехфазное, может сложиться ложное представление, что в комплексной плоскости одновременно присутствуют три вектора для каждой фазы отдельно. На самом деле для описания трехфазного напряжения используется понятие обобщенного вектора трехфазной системы напряжений. В таком представлении одновременно присутствует лишь один обобщенный вектор. Проекции этого вектора на три оси, развернутые друг относительно друга на 120 электрических градусов, однозначно определяют напряжение каждой из трех фаз. Учитывая тот факт, что в ВБ имеется три плеча, в каждом из которых - два последовательно соединенных взаимно инверсных ключа, всего возможно $2^3 = 8$ положений вектора, два из которых нулевые. На рисунке 4 изображены 6 ненулевых векторов, образующих 6 секторов. Все остальные векторы образуются путем поочередного включения нулевых векторов с одной из пар смежных образующих векторов. В этом и заключается принцип векторной ШИМ.

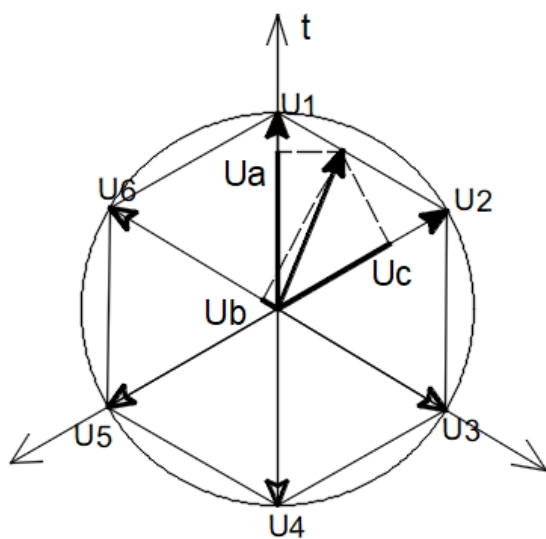


















Рисунок 4 – Векторная диаграмма

Из рисунка видно, что обобщенный вектор может существовать в пределах шестиугольника, вписанного в окружность, радиус которой равен напряжению ИПН. Механизм образования векторов проиллюстрирован в таблице 1, где

каждой комбинации активных ключей y_1, y_2, y_3 соответствует свое состояние обобщенного вектора.

Таблица 1 – Состояние обобщенного вектора

№	y_1	y_3	y_5	Состояние ключей	Обобщенный вектор
0	0	0	0		
1	1	0	0		
2	1	1	0		
3	0	1	0		
4	0	1	1		
5	0	0	1		
6	1	0	1		
7	1	1	1		

Как было сказано ранее, основываясь на геометрическом представлении обобщенный вектор можно разложить в комбинацию нулевых и смежных образующих векторов. Математически это описывается как:

$$\vec{V}_{\text{обобщ}} = T_1 \vec{V}_1 + T_2 \vec{V}_2 + T_0 \vec{V}_0 + T_7 \vec{V}_7;$$

$$T_1 + T_2 + T_0 + T_7 = T_{\text{ШИМ}},$$

где T_1, T_2, T_0, T_7 - время существования одноименных векторов,

$T_{\text{ШИМ}}$ - период ШИМ.

Каждый сектор разбивается при этом на несколько подсекторов или во временной области интервалов усреднения в течение которых положение обобщенного вектора не меняется, т. е. дискретность вращения обобщенного вектора остается, только ее период уменьшается [3].

Теперь, когда стало ясно, как образуется обобщенный вектор, осталось понять, как рассчитываются длительности векторов и в какой последовательности они чередуются [6]. Все это будет описано в дальнейшем при рассмотрении программной реализации.

3. МИКРОПРОЦЕССОРНАЯ РЕАЛИЗАЦИЯ

Освоение и распространение микропроцессорной техники открыли возможности практической реализации и широкого использования ее в сложных системах. Универсальность и гибкость микропроцессора (МП) как устройств с программным управлением позволяют широко применять ее в самых различных системах управления для аппаратной реализации функций управления, контроля измерения и обработки данных. Применение МП в системах управления промышленным оборудованием предполагает, в частности, использование их для управления станками, транспортировочными механизмами, сварочными автоматами, прокатными станами, атомными реакторами, производственными линиями, электростанциями, а также создание на их основе робототехнических комплексов, гибких автоматизированных производств, систем контроля и диагностики. Микропроцессорные средства позволяют создавать разнообразные по функциям устройства управления - от микроконтроллеров несложных приборов до универсальных систем управления в реальном времени [7].

Микропроцессорная реализация с одной стороны, упрощает выполнение многих требований, а с другой - ухудшает свойства сигнала. В случае микропроцессорной реализации АИН это влечет к невозможности формирования непрерывного синусоидального сигнала во времени.

3.1. Особенности микропроцессорной реализации

Как было сказано ранее, для реализации ШИМ требуется низкочастотный модулирующий сигнал. Параметры модулирующего сигнала определяют частоту и амплитуду первой гармоники выходного тока, а также его спектральный состав и, следовательно, энергоэффективность АИН. В данной ВКР в качестве такового используется синусоидальный сигнал. При этом важно обеспечить ре-

гулирование частоты и амплитуды в широких пределах, не допустить присутствие постоянной составляющей и не симметрию между фазами.

Важно понимать, что в цифровой реализации время и значения функции имеет дискретный характер. Частота тактирующих импульсов определяет дискретность во времени. Дискретность значений функции зависит от разрядности памяти, а также от разрядности тех устройств периферии, участвующих в записи или чтении этой функции. В случае использования МК STM32F103R4 шина памяти имеет разрядность 32 бита, тогда как разрядность таймеров, использующих значение функции для генерации ШИМ, составляет 16 бит; аналогово-цифровой преобразователь (АЦП), обеспечивающий обратную связь СУ, имеет разрядность 12 бит, что обеспечивает $2^{12} = 4096$ различных значений функции.

Простейшим примером цифровой реализации является счетчик таймера. Значение этого счетчика меняется по линейному закону с дискретностью во времени, определяемому частотой тактирующих импульсов. Дискретность значений счетчика определяется разрядностью таймера МК STM32F103R4, поддерживающего $2^{16} = 65535$ различных значений. Таким образом, двоичный счетчик можно представить, как цифровой аналог устройства с линейно изменяющимся выходным напряжением (например, генератора опорного напряжения).

В МПСУ, как правило, не добиваются минимально возможной дискретности во времени для генерируемых сигналов, в частности для модулирующего сигнала. Дискретизацию по этой координате стараются «привязать» к процессам, протекающим в объекте управления. В МПСУ вентильным преобразователем дискретизацию непрерывной функции по времени совмещают с периодом дискретности работы этого преобразователя (или интервалом повторения) [3].

Ступенчатая аппроксимация непрерывного, в частности синусоидального, сигнала бывает двух видов: с постоянным приращением амплитуды ΔS и с по-

стоянной длительностью Δt ступеньки [8]. Их особенности продемонстрированы соответственно на рисунке 5, а и б.

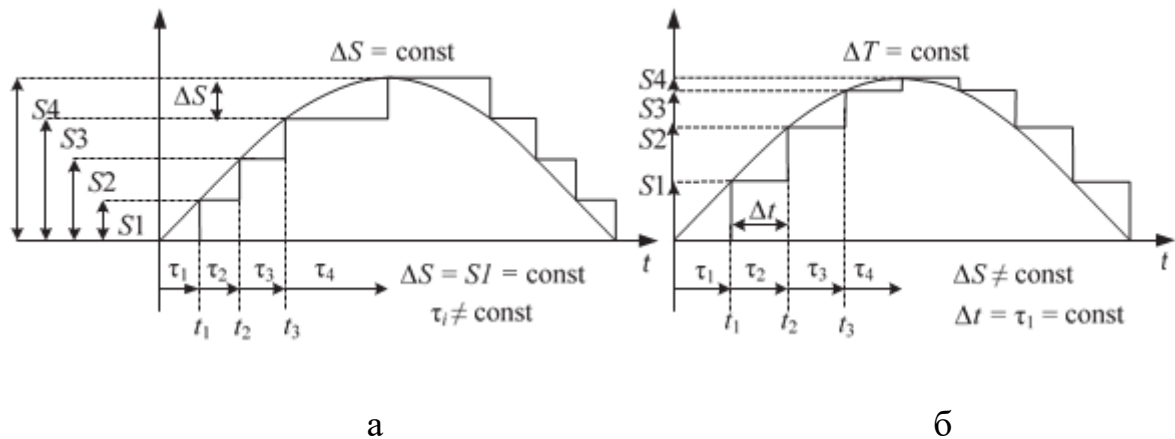


Рисунок 5 – Ступенчатая аппроксимация функции

Аппроксимацию с постоянным приращением амплитуды применяют в силовых устройствах (вентильный преобразователь переменного напряжения с равномерными отпайками трансформатора), а с постоянным приращением времени - в МПСУ.

Исходным источником информации о значениях функции служит ее непрерывный аналог. Для аппроксимации период функции разбивается на N равных интервалов (рисунок б), которому соответствует свое значение функции, определяемой в середине временного интервала.

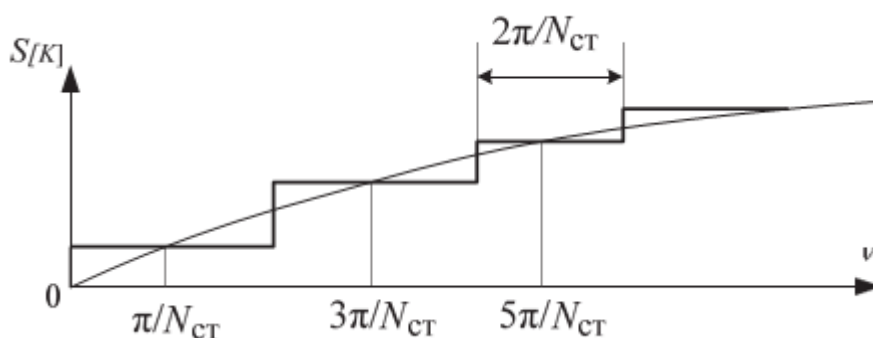


Рисунок 6 – Ступенчатая аппроксимация непрерывной функции

В работе [3] было получено уравнение, определяющее значение функции в зависимости от номера ступеньки:

$$S_{[k]} = \sin \frac{\pi}{N_{CT}} (2k + 1),$$

где – N_{CT} - число временных интервалов, на которое разбивается диапазон изменения функции;

k – номер ступени, лежащий в промежутке от 0 до $N_{CT} - 1$.

Ранее были указаны ограничения сверху, касаясь величины дискретизации функции, но важно понимать, что есть и ограничения снизу. Любая аппроксимация, в том числе и ступенчатая приводит к искажениям ее функции и, следовательно, к ухудшению гармонического состава выходного тока, т.е. чем больше ступенек, тем лучше аппроксимация, тем лучше качество выходного тока. Обычно при выборе N_{CT} руководствуются предельно допустимым коэффициентом гармонических искажений K_G , который можно определить по формуле [8]:

$$K_G = \sqrt{\left(\frac{\pi}{N_{CT} \sin \frac{\pi}{N_{CT}}} \right)^2} - 1.$$

Задавшись величиной K_G можно определить минимально допустимое число ступенек. На практике обычно ограничиваются 30 – 50 ступеньками для единиц герц и 10 – 20 для частоты 50 Гц.

3.2. Основные способы микропроцессорной реализации ШИМ

В данном разделе пойдет речь о методах реализации функции модулятора, формирующего временной интервал, влияющий на скважность. Известны два способа формирования временного интервала: программный и аппаратный. Будет затронута важная техническая проблема – распределение функций между аппаратурой и программными средствами. Ее решение – одна из важнейших

задач в процессе оптимизации разрабатываемой МПСУ. И решается она в каждом конкретном случае по-своему. В этой связи можно отметить лишь два момента. Во-первых, реализация функций аппаратными средствами упрощает программное обеспечение (разгружает МП), и наоборот. Во-вторых, имеющиеся аппаратные средства МПСУ должны быть обязательно использованы. Так, если в МПСУ УВ имеется незадействованный программируемый таймер, то, очевидно, нужно выбрать именно аппаратный способ формирования угла управления. Реализованная с учетом такого подхода МПСУ в большей степени соответствует понятию «оптимальная система» [3].

3.2.1. Программный способ формирования импульсов

Программный способ основан на том факте, что любая команда выполняется за определенное число периодов (тактов) частоты синхронизации микропроцессора. Требуемую временную длительность можно выразить через число тактов или соответственно через определенное число команд, которые должен выполнить микропроцессор для формирования требуемой временной задержки.

Процедура формирования временного интервала обычно заключается в подпрограмму и реализуется в виде программного цикла по известной структуре: счетчик циклов, тело цикла, модификация счетчика циклов, опрос условия выхода из программного цикла [3,9]. Обычно в таком случае в тело цикла вставляют команду пустой операции «nop», т.к. ее выполнение не изменяет текущее состояние программы.

Дискретность получаемого временного интервала пропорциональна времени выполнения одной операции, а, следовательно, тактовой частоте процессора. Можно выделить ряд достоинств и недостатков программного способа формирования временных интервалов.

Достоинство: простота аппаратной реализации, т.к. необходим только микропроцессор.

Недостатки:

- неэффективное использование вычислительных ресурсов микропроцессора из-за большого числа выполнения пустых операций;
- возможен только одноканальный вариант;
- сокращение функций, реализуемых МПСУ.

Таким образом, этот способ имеет право на существование, но лишь в тех случаях, когда вопрос стоимости МК стоит выше функциональных возможностей.

3.2.2. Аппаратный способ формирования импульсов

Аппаратный способ формирования временных интервалов напротив отличается от предыдущего экономным использованием вычислительных ресурсов, но для его реализации необходимы дополнительные периферийные устройства. В микропроцессорной технике функции преобразователя кода модуляции во временной интервал выполняет программируемый таймер.

Программируемый таймер представляет собой периферийное устройство, состоящее из двоичного счетчика и цифровой «обвязки», отвечающей за возможности таймера и информацию о текущем режиме его работы. Можно выделить наиболее важные функциональные возможности таймеров [10]:

- наличие канала выходного сравнения;
- возможность формирования запроса на прерывание по различным событиям;
- режим счета (суммирующий, вычитающий, реверсивный и т. д.);
- возможность останова и сброса счётчика.

Достоинства и недостатки аппаратного способа формирования угла управления противоположны достоинствам и недостаткам программного способа.

Достоинства:

- эффективное использование вычислительных ресурсов микропроцессора;
- возможность расширения функционала (использование регуляторов, алгоритмического распределения, диагностики и пр.).

Недостаток: требует наличие соответствующих аппаратных средств.

Так как в настоящее время наибольшее распространение получил аппаратный способ формирования временных интервалов, далее будет использоваться этот способ.

В современных МПСУ используется компараторная реализация ШИМ, когда текущее значение аппроксимированной функции, хранящееся в специальном регистре, сравнивается со значением двоичного счетчика и в момент их сравнения формируется импульс. По переполнению импульс оканчивается и весь процесс начинается заново. Кроме основного функционала таймера, при реализации ШИМ необходимы дополнительные для обеспечения надежной работы:

- поддержка формирования интервалов мёртвого времени для исключения появления сквозных токов;
- компенсация его влияние на выходное напряжение;
- защита цепей АИН в случае наступления аварийной ситуации.

С этой целью были изобретены и применяются специализированные микроконтроллеры класса «Motion Control» поддерживающие основной дополнительный функционал.

Возможен и другой вариант реализации ШИМ – таймерный. В этом случае достаточно иметь дешевые микроконтроллеры общего назначения с несколькими программируемыми таймерами. Последние должны обладать традиционными свойствами: возможностью предварительного занесения числа в двоичный счетчик и формирования запроса на прерывание по переполнению или обнулению. При этом сервисные функции управления АИН реализуются другими встроенными в микроконтроллер периферийными устройствами или же внешними схемами. Этот интервал также должен формироваться согласно закону, по которому работает ШИМ. Важное достоинство таймерной ШИМ заключается в простоте реализации. Как работа инвертера представляет собой ряд последовательных коммутаций вентиля, так и таймерная ШИМ формирует последовательный ряд межкоммутационных интервалов.

При модуляции постоянного сигнала время, предоставляемое для расчета алгоритма управления тоже постоянно. Но в динамике, когда значение модулирующего сигнала может быть, как больше, так и меньше предыдущих, соответственно будет меняться и время между коммутациями, т. е. время, предоставляемое для расчета алгоритма. Особенно тяжелым является случай, когда сигнал имеет низкую амплитуду, тогда время одного импульса будет уменьшаться и увеличиваться у соседнего. Это может привести к тому, что для расчета новых данных управления просто не хватит времени. Чтобы этого не произошло, в таймерных ШИМ амплитуда модулирующего сигнала ограничивают минимальным значением. Эта величина выбирается из условия гарантированного расчета микропроцессором новых данных для управления [3].

3.3. Изменение частоты

В процессе работы АИН может потребоваться регулирования скорости вращения двигателя и, следовательно, - изменение частоты тока.

При создании ПО необходимо руководствоваться техническими требованиями, исходящими от заказчика. В частности, необходимо знать требуемый диапазон регулирования частоты вращения двигателя, который определяется исходя из требований технологической задачи и как правило ограничен возможностями самого двигателя.

При выборе несущей частоты необходимо руководствоваться требованиями, исходящими на этапе проектирования силовой части АИН и, частности, следующими ограничениями. С увеличением частоты ШИМ возрастают коммутационные потери в АИН, что приводит к снижению КПД, следовательно, частота пульсаций должна быть ограничена максимальной допустимой частотой коммутации транзисторов. При уменьшении частоты коммутации увеличивается нелинейность регулировочной характеристики АИН, что крайне нежелательно [11]. Принцип работы ШИМ требует наличие фильтра для сглаживания пульсаций выходного напряжения. Использование внешнего дросселя в качестве фильтра приводит к увеличению стоимости и габаритов всей системы, поэтому его присутствие крайне нежелательно. На практике стараются ограничить индуктивностью самих обмоток двигателя, увеличивая эффективность их фильтрующего свойства за счет увеличения частоты пульсаций.

В данной работе требования к диапазону регулирования частоты и частоты пульсаций исходят из задания на ВКР. Исходя из этого, при создании ПО СУ необходимо обеспечить плавное изменение частоты выходного тока АИН в диапазоне от 1 до 50 Гц, при этом частота ШИМ должна быть 10 кГц.

Для начала необходимо разобраться с особенностями микропроцессорной реализации компараторной ШИМ. В качестве модулирующего сигнала в МК используется его ступенчатая аппроксимация. В качестве опорного сигнала ис-

пользуется значение счетчика, также представляющий собой ступенчатую функцию. Время начала положительного сигнала отсчитывает от момента сравнения значения счетчика со значением модулирующей функции. На рисунке 7, а показан принцип формирования импульсов периодичностью T .

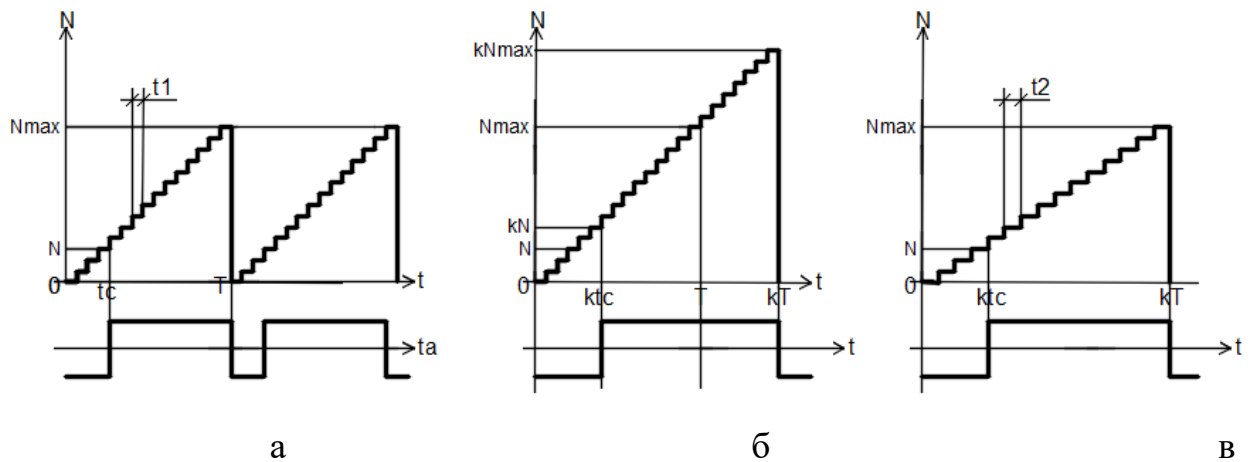


Рисунок 7 – Эпюры сигналов

Частотой выходного сигнала можно управлять тремя способами. Первые два основаны на изменении частоты пульсаций ШИМ сигнала., при этом важно понимать, что значение модулирующей функции изменяется при каждом обнулении счетчика. Следовательно, изменяя скорость обнуления счетчика можно менять скорость изменения модулирующего сигнала, а значит, и частоту.

3.3.1. Первый способ

Первый способ (рисунок 7, б) основан на изменении в k раз пороговой величины счетчика N_{max} , за счет чего изменяется в k раз период ШИМ сигнала T .

Достоинство:

- возможность плавного регулирования.

Недостатки:

- изменение частоты пульсаций ШИМ сигнала;
- усложнение программного обеспечения, т.к. требует постоянного пересчета пороговой величины счетчика и масштабирования сигнала.

3.3.2. Второй способ

Во втором способе (рисунок 7, в) изменяется длительность ступеньки t_1 в k раз до значения t_2 , при этом период T также увеличивается до значения kT . Длительность ступеньки определяется тактовой частотой счетчика.

Достоинство:

- не требует усложнения программного обеспечения для масштабирования сигнала.

Недостатки:

- изменение частоты пульсаций ШИМ сигнала;
- опорная частота определяется тактовым генератором с кварцевой стабилизацией и может перестраиваться только за счёт использования делителя частоты, принимая ряд фиксированных значений, так что плавная перестройка и точное задание частоты синтезируемого сигнала невозможны [12];
- усложнение программного обеспечения для расчета делителя частоты.

Существует иная реализация второго способа, лишённая некоторых недостатков. При этом задействуется дополнительный таймер микроконтроллера, который программируется на формирование импульсов с программируемой частотой. Этот таймер является внешним источником тактирующего сигнала по отношению к таймеру отвечающего за формирование ШИМ [3].

Достоинства:

- не требует усложнения программного обеспечения;
- возможность плавного регулирования.

Недостатки:

- изменение частоты пульсаций ШИМ сигнала;

- требует наличие дополнительного таймера, поддерживающего высокую частоту выходного сигнала.

3.3.3. Третий способ

Для генерации функции синуса выбран следующий способ. В нем значения каналов счетчика для каждого промежутка времени записаны в специальную таблицу значений синуса. Важно отметить, что данная программа и все ПО не изменится в случае изменений формы модулирующего сигнала, необходимо будет лишь загрузить новые значения функции в таблицу. Выбор такого подхода связан со следующими особенностями:

1. в качестве функции модулирующего сигнала выбрана функция синуса. В системе команд МК нет специальной команды для ее вычисления. Эта функция реализуется программно методом разложения в ряд, на что затрачивается часть вычислительных ресурсов МК;
2. Значение функции повторяются через равный промежуток времени.
3. При выборе в качестве модулирующей функцию специальной формы достаточно будет загрузить ее значения в таблицу функции.

При создании алгоритма изменения частоты часто используются идеи, применяемые при создании цифрового вычислительного синтезатора. Основная мысль, лежащая в основе этого метода, гласит что при изменении частоты следует менять величину инкрементации индекса массива. [12]. Иными словами при уменьшении частоты в n раз относительно максимальной, за элементом массива $f(t, \omega_s)_{[k]}$ должен следовать элемент массива $f(t, \omega_s)_{[k+n]}$. При этом этот метод имеет три существенных недостатка:

- при уменьшении частоты уменьшается число точек, по которым формируется сигнал, следовательно, ухудшается качество выходного тока АИН;
- большие затраты ресурсов ПЗУ для хранения всех элементов массива;

- требуется высокая частота ШИМ.

Для устранения первого недостатка стремятся установить минимальное число точек для минимальной частоты, что в свою очередь приводит к избыточности информации о сигнале при максимальной частоте, что еще больше усиливает второй недостаток. Таким образом при реализации данного способа приходится мириться по крайней мере с одним из недостатков, устраняя другой. На деле при минимальной частоте для обеспечения минимального числа точек на период $N_{min} = 30$, приходится создавать массив размером $N_{max} = 1500$. Для обработки такого массива при максимальной частоте потребуется частота ШИМ равная

$$f_{ШИМ} = f_{max} \cdot N_{max} = 50 \cdot 1500 = 75 \text{ кГц}$$

Требуемая частота на порядок отличается от заданной, в связи с чем этот способ реализован не будет. Также при реализации этого способа потребуется 3Кбайт памяти ПЗУ, что тоже немаловажный ресурс.

3.3.4. Четвертый способ

Для генерации ШИМ сигнала периодической функции программа будет использовать ее заранее вычисленные значения. Для понимания следующего способа изменения частоты следует представить генерируемый сигнал как функцию аргумента, значение которого изменяется во времени. В случае генерации синусоидального сигнала его функция имеет вид

$$f(t, \omega_3) = a \cdot \sin(\omega_3 t),$$

где ω_3 – угловая частота, задаваемая программой верхнего уровня, a – амплитуда сигнала.

Эта функция зависит от времени и, если обращаться к элементам массива как к значениям функции от времени это приведет к тому, что ее значения становятся непригодны при изменении частоты и потребуется пересчет таблицы.

Есть все основания полагать, что такой способ является затратным с точки зрения вычислительных возможностей МК и приведет к длительным задержкам. Тогда результат элементарного приращения функции есть функция от приращения ее аргумента

$$f(t, \omega_3)_{[k]} = f(t, \omega_3)_{[k-1]} + \Delta f(t, \omega_3) = a \cdot \sin(\omega_3 t + \omega_3 \Delta t) = a \cdot \sin(\varphi_{[k]} + \Delta \varphi).$$

Таким образом целесообразно обращаться к элементам массива как к значениям функции от ее аргумента (угла). Аргумент функции находится в пределах от 0 до 360 градусов, что в программном исполнении представляет собой диапазон целочисленных чисел от 0 до N_{max} . Для каждого значения из этого диапазона массив ставит в соответствие значение функции.

Поскольку диапазон аргумента является неизменным, следовательно, размер массива также будет оставаться неизменным. Для обеспечения данного условия важно обеспечить постоянство выражения приращения аргумента

$$\Delta \varphi = \omega_3 \Delta t$$

где Δt – периодичность обращения к элементам массива.

Нет необходимости осуществлять выборку из таблицы с периодичностью меньше, чем периодичность опорного сигнала. Следовательно, можно определить минимальное приращение, как

$$\Delta t_{min} = T_{ШИМ}.$$

Этому минимальному временному промежутку будет соответствовать максимальная угловая частота

$$\omega_{max} = 2\pi f_{max}$$

Таким образом величина дискретизации диапазона зависит частоты ШИМ, чем она выше, тем на большее число дискрет можно разбить весь диапа-

зон. При этом важно понимать, что с уменьшением $\Delta\varphi$ увеличивается число элементов массива, которое можно определить как

$$N_{\max} = \frac{2\pi}{\Delta\varphi} - 1 = \frac{2\pi f_{\text{ШИМ}}}{2\pi f_{\max}} - 1$$

В таком случае можно вычислить относительную точность формирования сигнала

$$\delta = \frac{1}{N_{\max} + 1} = \frac{f_{\max}}{f_{\text{ШИМ}}}$$

При $f_{\text{ШИМ}} = 10$ кГц и $f_{\max} = 50$ Гц $\delta = 0,005$, что вполне удовлетворительно для СУАИН.

В момент переполнения счетчика таймер способен генерировать запрос на прерывание, и после этого момента у функции обработки прерывания есть время

$$\Delta t = \frac{1}{f_{\text{ШИМ}}},$$

в пределах которого необходимо произвести все вычисления и сформировать новые данные для кода модуляции. Как правило, при каждом таком прерывании в специальный регистр счетчика заносятся новые данные для сравнения. Такой подход применим для третьего способа.

Идея четвертого способа заключается в изменении частоты занесения данных в регистр. для соблюдения условия, сформулированного ранее, о постоянстве приращения аргумента. Тем самым при уменьшении частоты в k раз, Δt должно увеличиться в k раз. В таком случае целесообразно при занесении новых данных в регистр ориентироваться на прерывания таймера ШИМ, так как его частота постоянна.

В данной работе предложено уникальное решение - задействовать дополнительный таймер, который будет инициализировать прерывания для занесения

новых данных. Таким образом для регулирования частоты модулирующей функции необходимо регулировать частоту срабатывания таймера. Можно отметить ряд преимуществ при использовании данного способа.

Преимущества:

- постоянство частоты пульсации ШИМ;
- число точек не зависит от частоты и, следовательно, качество тока при регулировании частоты неизменно;
- экономия памяти ПЗУ;
- плавность регулирования.

Недостатки:

- требует наличие дополнительного программируемого таймера,

В защиту данного способа стоит сказать, что для его реализации в числе доступной аппаратной периферии МК класса «Motion Control» присутствуют дополнительные таймеры. счетчик. В частности, он присутствует в МК STM32F103R4, на базе которого строится МПСУ в данной работе и очевидно, что необходимо задействовать все имеющиеся аппаратные средства для большего соответствия МПСУ понятию «оптимальная система». В связи с чем этот способ и выбран для микропроцессорной реализации.

3.4. Регулятор

В современном производстве стали широко применяться системы автоматизации, инструментом которой стали промышленные регуляторы, в особенности пропорционально-интегральный-дифференциальный регулятор, призванный повысить быстродействие и точность объекта управления.

ПИД регулятор описывается его передаточной функцией

$$W(p) = k_n + k_d p + \frac{k_u}{p},$$

где k_n, k_d, k_u - пропорциональная, дифференциальная и интегральная составляющие соответственно.

Входным сигналом для регулятора служит разница между заданным значением регулируемой величины и ее фактическим значением:

$$e(t) = u_s(t) - u_{oc}(t)$$

Связи с особенностью цифровой реализации СУ, любые процессы, проходящие во времени, имеют дискретный характер, т.е. от момента получения информации о состоянии входных величин до момента выдачи рассчитанного значения проходит время, именуемое интервалом квантования. Интервал квантования зависит от тактовой частоты МП и его загруженности и для обеспечения качественного управления его величина не должна превышать значение наименьшей постоянной времени СУ [3].

В цифровой реализации дифференциал первого порядка непрерывной функции записывается как:

$$\Delta f_{[k]} = f_{[k]} - f_{[k-1]}.$$

Дифференциал второго порядка записывается как:

$$\Delta^2 f_{[k]} = \Delta f_{[k]} - \Delta f_{[k-1]} = (f_{[k]} - f_{[k-1]}) - (f_{[k-1]} - f_{[k-2]}) = f_{[k]} - 2f_{[k-1]} + f_{[k-2]}.$$

Воспользуемся результатами работы [3], в которой получено уравнение цифровой реализации ПИД регулятора:

$$u_{[k]} = u_{[k-1]} + \left(k_n + \frac{T}{T_u} + \frac{T_d}{T} \right) e_{[k]} - \left(k_n + \frac{2T_d}{T} \right) e_{[k-1]} + \frac{T_d}{T} e_{[k-2]},$$

где T_u, T_d, T_i – постоянные времени ПИД регулятора,

T – интервал квантования.

Вопрос синтеза и реализации регуляторов, в частности цифровых, очень важен и в то же время довольно сложен. Он важен потому, что от регуляторов во многом зависят характеристики всей энергетической системы, а сложность

вопроса заключается в том, что правильный выбор параметров регуляторов и их адекватная реализация требуют больших знаний и опыта.

3.5. Настройка регулятора

Настройка заключается в изменении параметров регуляторов, что является довольно трудозатратным этапом автоматизации. При реализации регулятора в МПСУ предварительно необходимо провести компьютерную проверку разработанного регулятора с использованием прикладных программ, таких как Matlab Simulink или SimInTech. Только после успешных результатов компьютерного моделирования можно приступить к программированию МПСУ.

Настройка регулятора основана на том, что в замкнутой системе известны передаточные функции всех входящих в нее элементов. В нашем случае это передаточные функции и вентильного преобразователя, и двигателя, и его нагрузки. Кроме того, должны быть известны все возмущающие воздействия и т. д. На практике этого, как правило, не бывает: информация приблизительная или вообще отсутствует. Поэтому после разработки и наладки системы ее приходится дополнительно настраивать непосредственно на объекте [3]. Для избежания этого процесса применяют адаптивные СУ, например, с нечеткой логикой.

Нечеткое управление применяется при недостаточном знании объекта управления (его передаточной функции), но при наличии опыта управления. На основании этого опыта эксперт формирует базу правил нечеткого регулятора. Примером может быть водяной насос, параметры которого (потребление воды, момент инерции всей системы) изменяются в широком диапазоне и вызывают большие затруднения при вычислении. При этом эксперт способен достаточно хорошо управлять таким объектом, используя лишь показания приборов и накопленный опыт.

Применение регулятора нечеткого вывода также может быть обосновано нелинейностью объекта управления. Применение пропорционально-интегрально-дифференцирующих (ПИД) регуляторов в таких условиях чаще всего приводит

к низкому качеству управления: большому значению перерегулирования, времени переходного процесса и установившейся ошибки.

В качестве примера при моделировании СУ с нечеткой логикой объект управления будет описываться апериодическим звеном второго порядка

$$W_0(s) = \frac{1}{2s^2 + 3s + 1}.$$

Моделируемая система управления состоит из объекта управления и ПИД регулятора, охваченных обратной связью, блока нечеткой адаптации (БНА) параметров ПИД регулятора и источника задающего сигнала. Ниже (рисунок 8) приведена ее структурная схема.

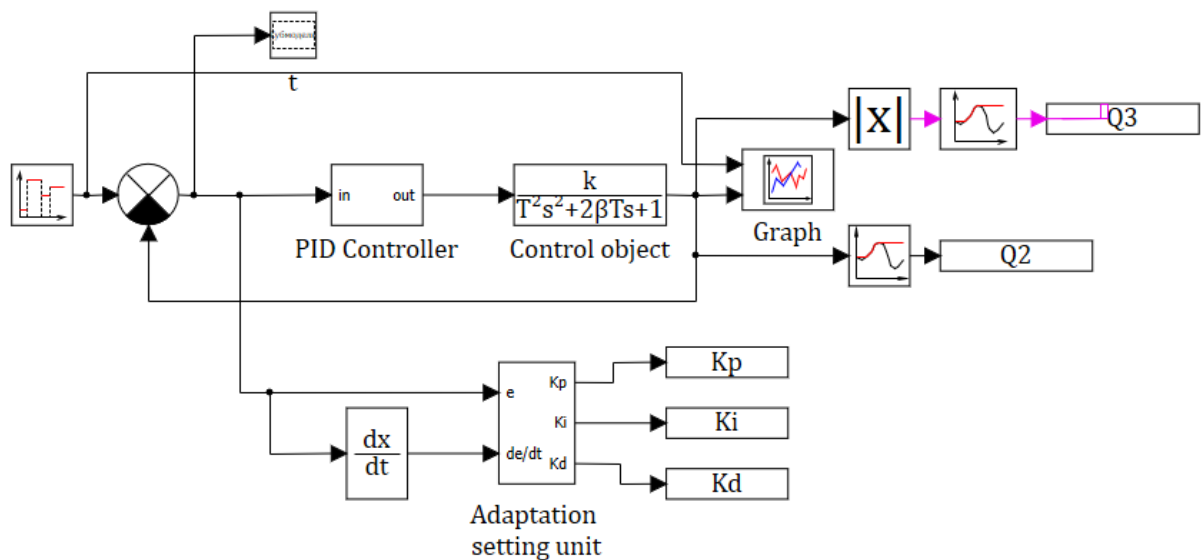


Рисунок 8 - Структурная схема системы управления

Основываясь на работе [13], в которой обосновывается целесообразность применения нечёткого регулятора, в качестве основного регулирующего звена выбран ПИД регулятор с адаптацией его параметров настройки K_p , K_i и K_d посредством блока нечеткой адаптации (БНА). Таким образом, использование БНА позволяет повысить робастность системы. Ниже (рисунок 9) приведена структурная схема ПИД регулятора.

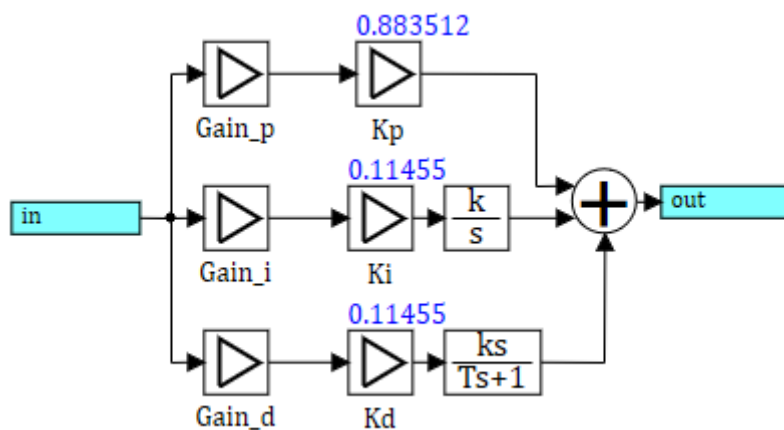


Рисунок 9 - Структурная схема ПИД регулятора

БНА принимает на вход ошибку регулирования e и скорость ее изменения $\frac{\partial e}{\partial t}$ и выдает на выход значения коэффициентов пропорциональной, интегральной и дифференцирующей составляющих ПИД регулятора.

Для применения методов нечеткой логики необходимо преобразовать четкие переменные в нечеткие, когда о истинности высказывании можно судить, основываясь на значении его функции принадлежности. Диапазон значений входной переменной разбивается на подмножества NI, NM, NS, Z, PS, PM, PL. В пределах каждого из них строится функция принадлежности.

Для выходной переменной в условиях нашей задачи диапазон разбивается на 4 подмножества значений Z, PS, PM, PL от нуля до некоторого положительного. Это связано с тем, что коэффициенты ПИД регулятора не могут быть отрицательными.

Процесс управления состоит из 3 основных этапов [14]:

- фаззификация входных переменных, представляющая собой процедуру нахождения значений функций принадлежности нечетких множеств, на основе четко определенных исходных данных;
- определение нечеткого логического вывода с помощью блока правил;

- дефаззификация выходных переменных в системах нечеткого вывода, представляющая собой процедуру или процесс нахождения обычного (четко определенного) значения для каждой из подмножеств.

Цель дефаззификации заключается в том, чтобы, используя результаты логического заключения, получить значение каждой из выходных переменных, которое может быть использовано специальными устройствами, внешними по отношению к системе нечеткого вывода.

Процесс настройки с помощью блока нечеткой логики следует начинать с приближенного вычисления начальных значений коэффициентов регулятора. После формулируется функция принадлежности. В данной работе функция принадлежности для каждого подмножества имеет треугольную форму.

Диапазоны параметров функций принадлежности для входных сигналов составляет $[-1,5 \dots 1,5]$, для выходных $[0 \dots 1]$.

Для корректировки коэффициентов при составлении логики управления необходимо руководствоваться следующими соображениями [15]:

- увеличение K_p ведёт к ускорению переходного процесса, уменьшению устойчивости и уменьшению статической ошибки; уменьшение K_p ведёт к замедлению переходного процесса, увеличению устойчивости (уменьшение колебательной) и увеличению статической ошибки
- увеличение K_i приводит к уменьшению статической ошибки и к увеличению его колебательности; уменьшение K_i приводит к увеличению статической ошибки, но уменьшает его колебательность.
- увеличение K_d приводит к повышению устойчивости процесса и его быстродействию, но при этом регулятор становится восприимчивым к высокочастотным шумам; уменьшение K_d приводит к уменьшению устойчивости и быстродействия процесса, но при этом высокочастотные помехи оказывают меньшее влияние на процесс управления.

Выделим следующие закономерности для вычисления параметров.

1. Чем больше по модулю ошибка рассогласования и её производная, тем больше должна быть K_p и тем меньше K_i и K_d .

2. Чем меньше модулю ошибка рассогласования и её производная, тем меньше должна быть K_p и тем больше K_i и K_d . В таблицах 2 и 3 приведены правила для коэффициентов ПИД-регулятора.

Таблица 2 –Правила для K_p

			e						
			+			0	-		
			LP	MP	SP	Z	SN	MN	LN
de/dt	+	LP	LP	LP	MP	SP	MP	LP	LP
		MP	LP	MP	SP	SP	SP	MP	LP
		SP	MP	SP	SP	Z	SP	SP	MP
	0	Z	SP	SP	Z	Z	Z	SP	SP
		SN	MP	SP	SP	Z	SP	SP	MP
		MN	LP	MP	SP	SP	SP	MP	LP
	-	LN	LP	LP	MP	SP	MP	LP	LP

Таблица 3 –Правила для K_i и K_d .

			e						
			+			0	-		
			LP	MP	SP	Z	SN	MN	LN
de/dt	+	LP	Z	Z	SP	MP	SP	Z	Z
		MP	Z	SP	MP	MP	MP	SP	Z
		SP	SP	MP	MP	LP	MP	MP	SP
	0	Z	MP	MP	LP	LP	LP	MP	MP
		SN	SP	MP	MP	LP	MP	MP	SP
		MN	Z	SP	MP	MP	MP	SP	Z
	-	LN	Z	Z	SP	MP	SP	Z	Z

Основываясь на полученной базе правил, составляется логическая функция управления, фрагмент которой приведен ниже.

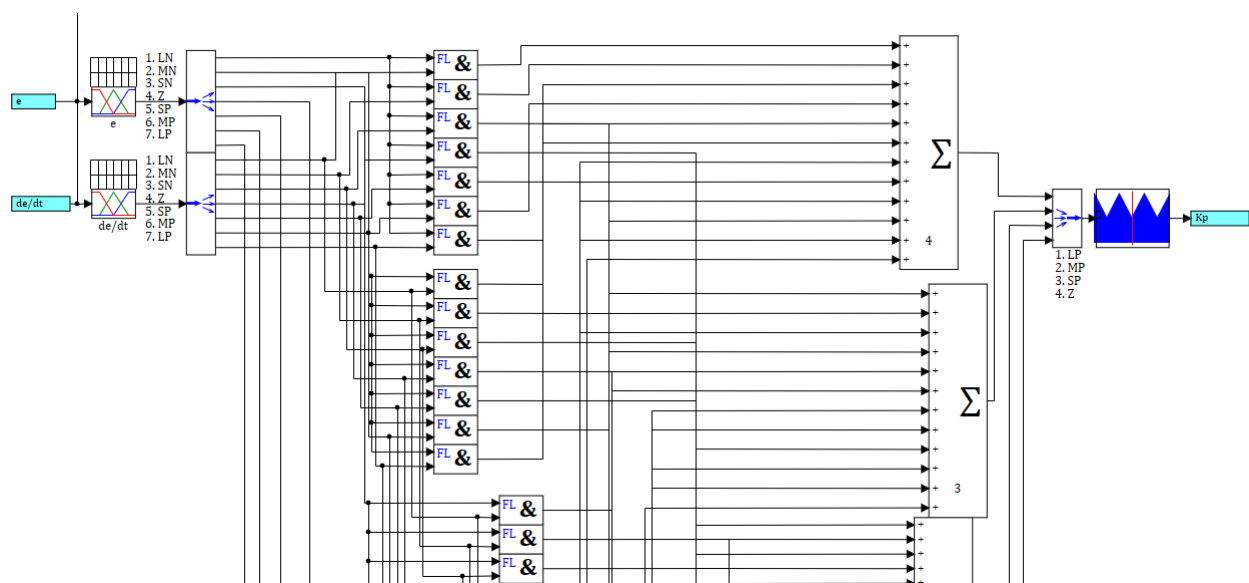


Рисунок 10 - Фрагмент внутренней структуры БНА для вычисления K_p

Применение БНА в качестве источника позволяет избавиться от необходимости настраивать параметры ПИД регулятора, но требует настройки параметров функций принадлежности для входящих и выходящих сигналов. Эта проблема легко решается если в качестве параметров функций принадлежности использовать не абсолютные, а относительные значения, которые потом обнажаются на масштабирующие коэффициенты $Gain_p$, $Gain_i$ и $Gain_d$.

Диапазоны параметров функций принадлежности для входных сигналов составляет $[-1,5 \dots 1,5]$, для выходных $[0 \dots 1]$.

Масштабирующие коэффициенты не могут определены однозначно. Следовательно, для нечеткого ПИД регулятора требуется решить задачу оптимизации, а именно задачу выбора значений, при которых критерий качества переходного процесса достигали допустимого уровня.

В качестве критерии переходного процесса была выбраны время переходного процесса $t < 0.3c$ и перерегулирование 10%. В качестве задающей функции используется ступенчатая функция.

Оптимизация осуществлялась по всему переходному процессу. В результате оптимизации имеем следующие оптимальные значения коэффициентов: $Gain_p = 51,62$, $Gain_i = 28,99$, $Gain_d = 49,29$.

Полученные коэффициенты позволяют достичь времени переходного процесса $t = 0,29$ с и перерегулирования 10%.

Важно отметить, что оптимизация выполняется единожды и далее изменение параметров ПИД регулятора осуществляется с помощью БНА.

Расчеты проводились в среде динамического моделирования SimInTech. Переходный процесс полученные при моделировании представлен на рисунке 11. Для имитации изменения параметров системы коэффициент демпфирования объекта управления менялся от 1 до 2 в течение 5 с. Можно заметить, что выходной сигнал устойчивый с перерегулированием 10% и временем затухания 0,3 с. Статическая ошибка в установившемся режиме составляет 2%.

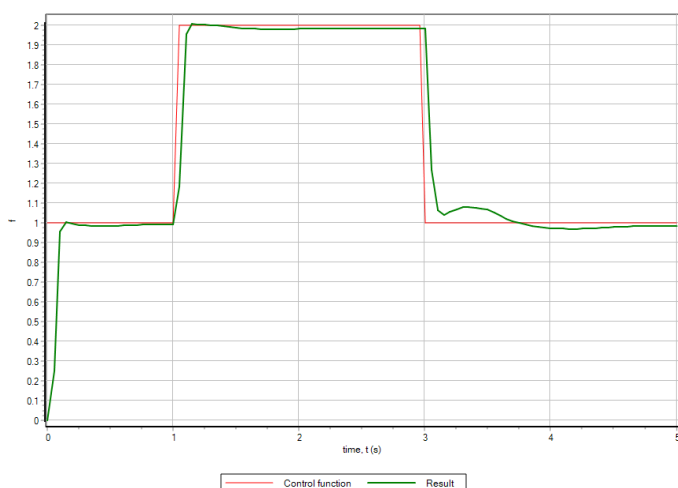


Рисунок 11 - График переходного процесса

В качестве обоснования целесообразности применения БНА ниже приведен (рисунок 12) график переходного процесса без его использования. Коэффициенты ПИД регулятора имеют фиксированные значения, полученные блоком БНА во время предыдущей симуляции: $K_p = 0,866$, $K_i = 0,113$, $K_d = 0,113$.

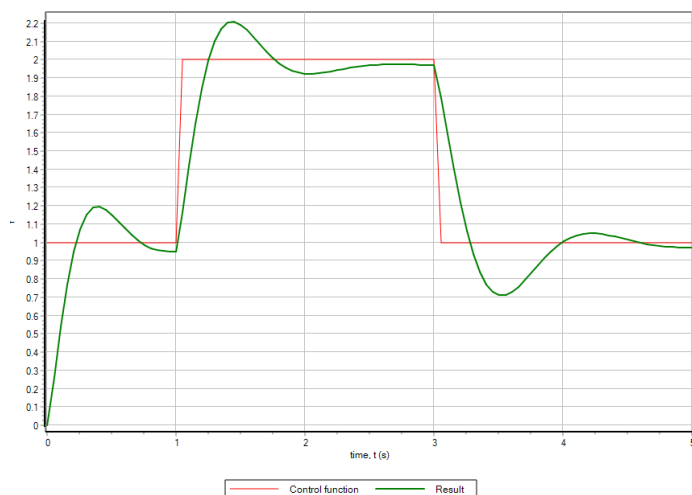


Рисунок 12 - График переходного процесса без БНА

Из полученного графика видно, что такая система отличается меньшим быстродействием и большим перерегулированием 20%.

В результате моделирования СУ можно сделать следующие выводы:

- использование ПИД регулятора с адаптацией его параметров посредством блока нечеткой адаптации (БНА) существенно эффективней чем использование отдельно ПИД регулятора или отдельно нечеткого регулятора;
- использование БНА позволяет повысить робастность системы.

4. ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ

В соответствии с [2], программное обеспечение (ПО) представляет собой совокупность компьютерных программ и документов, необходимых для эксплуатации этих программ. Процесс его разработки включает в себя следующие процессы:

- определение требований к ПО;
- проектирование ПО;
- кодирование ПО;
- интеграция и верификация ПО.

Таким образом, целью создания ПО является четкое определение правил или алгоритмов, которые задают последовательность действий для выполнения конкретных задач.

4.1. Определение требований к ПО

Целью данного процесса является разработка требований верхнего уровня и их оценка с точки зрения безопасности системы. Требования верхнего уровня содержат функциональные требования, требования к эффективности, требования к интерфейсу и требования, связанные с безопасностью [2].

Исходными данными для процесса определения требований к ПО являются системные требования, описание аппаратного интерфейса и архитектуры системы, рассмотренные ранее.

4.1.1. Требования к функционалу и безопасности

Перечислим основные функциональные требования, предъявляемые к микропроцессорной системе управления АИН [3].

Функция мониторинга: считывание показаний датчиков тока, датчика скорости, их преобразование с помощью аналого-цифрового преобразователя (АЦП) в двоичный код и обработка: фильтрация и нормализация.

Функция регулятора: формирование согласно реализуемым алгоритмам цифрового кода управляющего сигнала по вектору задающих сигналов и вектору сигналов обратной связи.

Функция модулятора: преобразование цифрового кода управляющего сигнала в ШИМ сигнал, задающий моменты коммутации вентиля полупроводникового преобразователя.

Функция алгоритмического распределения: формирование слова состояния вентиля в соответствии с заданным алгоритмом переключения, для каждого модулированного по длительности временного интервала.

Кроме этого, МПСУ должна выполнять целый ряд дополнительных сервисных задач, обусловленных требованиями к электроприводам промышленного применения. К ним относятся следующие.

Функция связи. Она необходима для организации двустороннего обмена информацией между МПСУ и вычислительным устройством верхнего уровня. Микропроцессорная система управления получает команды и передает различные параметры системы, необходимые для верхнего уровня управления.

Функция защиты. Параметры силовой части схемы должны постоянно контролироваться. При выходе значений параметров за допустимые границы необходимо сформировать сигнал прерывания работы. Согласно ему, формируется сигнал предупреждения или тревоги и выполняется ряд программных или аппаратных операций для защиты компонентов силового канала или изменения режима его работы.

Функция диагностики системы управления. Задача ее сводится к контролю состояния отдельных модулей МПСУ путем выполнения соответствующих тестовых программ, цель которых заключается в проверке аппаратуры

управления, обнаружении нарушений и, при возможности, выявлении их источников.

Функция отображения информации. В процессе работы электротехнической установки требуется индикация текущих параметров объекта управления и состояния системы на семисегментных индикаторах и светоизлучающих диодах.

Функция адаптивного режима. Наличие этого режима обусловлено необходимостью автоматической адаптации серийно выпускаемой микропроцессорной системы к требованиям и характеристикам конкретного объекта для оптимального управления.

Функция идентификации. Она требует наличия энергонезависимого оперативного запоминающего устройства (ОЗУ) или электрически программируемого постоянного запоминающего устройства (ПЗУ) для хранения настроек системы и другой персональной информации.

Функция внутрисхемной эмуляции. Для этого требуется специальный JTAG- интерфейс, встроенный в микроконтроллер, позволяющий обеспечить внутреннюю эмуляцию и отладку программы путем подключения МПСУ к персональному компьютеру с программной средой разработки и отладки.

4.1.2. Требования к эффективности

Эффективность работы системы определяется прежде всего быстродействием обработки информации, т.к. быстродействие влияет на качество регулирования. При максимально допустимой частоте коммутации управляемых вентилей 10 кГц, время обработки сигнала не должно превышать 100 мкс, т.е. система управления должна работать в режиме реального времени, что подразумевает выполнение следующих требований.

Во-первых, операции по переработке информации в микропроцессоре должны выполняться за меньшие или равные промежутки времени чем период

работы вентиля. Для решения данной проблемы требуется высокое быстродействие микропроцессора с высокой частотой его тактирования; наличие в системе команд соответствующих операций цифровой обработки сигналов и рациональное использование аппаратных ресурсов при создании программного обеспечения.

Во-вторых, работа МПСУ в режиме реального времени делает необходимым «привязку» выполнения программы (или отдельных ее фрагментов) к внешним событиям, что требует наличия в микропроцессоре системы прерывания. Это требование необходимо, например, для быстрой реакции МПСУ в случае возникновения аварии.

В-третьих, работа МПСУ в режиме реального времени подразумевает выполнение различных операций, имеющих дело с реальными временными интервалами, такими как формирование сигнала управления вентилем заданной длительности, генерацию периодических запросов на прерывание, реализацию формирования частоты передачи данных по последовательному каналу или частоты обновления информации на индикаторах и т. д. Для выполнения данных операций необходимы программируемые таймеры (ПТ), как наиболее важные компоненты управляющих микропроцессорных систем.

4.1.3 Требования к интерфейсу

Коммуникационная способность – неотъемлемое свойство МПСУ. Она необходима для организации двустороннего обмена информацией между МПСУ и вычислительным устройством верхнего уровня. Для этого или микроконтроллер должен быть оснащен высокоскоростными последовательным портам ввода/вывода, реализующими один из стандартных промышленных интерфейсов RS-232, RS-422, RS-485, CAN, Modbus, Profibus, а программа должна

уметь работать хотя бы с одним из промышленных протоколов передачи данных.

4.2 Проектирование ПО

Цель процесса проектирования ПО является разработка архитектуры ПО и требований нижнего уровня на основе требований верхнего уровня [2].

В свою очередь, определение архитектуры ПО подразумевает определение ее структуры и поведения ее отдельных блоков. Ниже дано краткое описание, предназначение и поведение каждого из них. Для предотвращения появления возможных отказов или уменьшения уровня критичности в случае их возникновения применяется метод разбиения, позволяющий разложить большую задачу на подзадачи. Каждая задача решается в отдельном программном блоке.

Рассмотренный ранее перечень требуемых основных и сервисных функций позволяет выделить основные программные блоки (отмечены зеленым и красным) и сервисные (отмечены серым), которые в комплексе образуют программное обеспечение, структура которого представлена на рисунке 13.

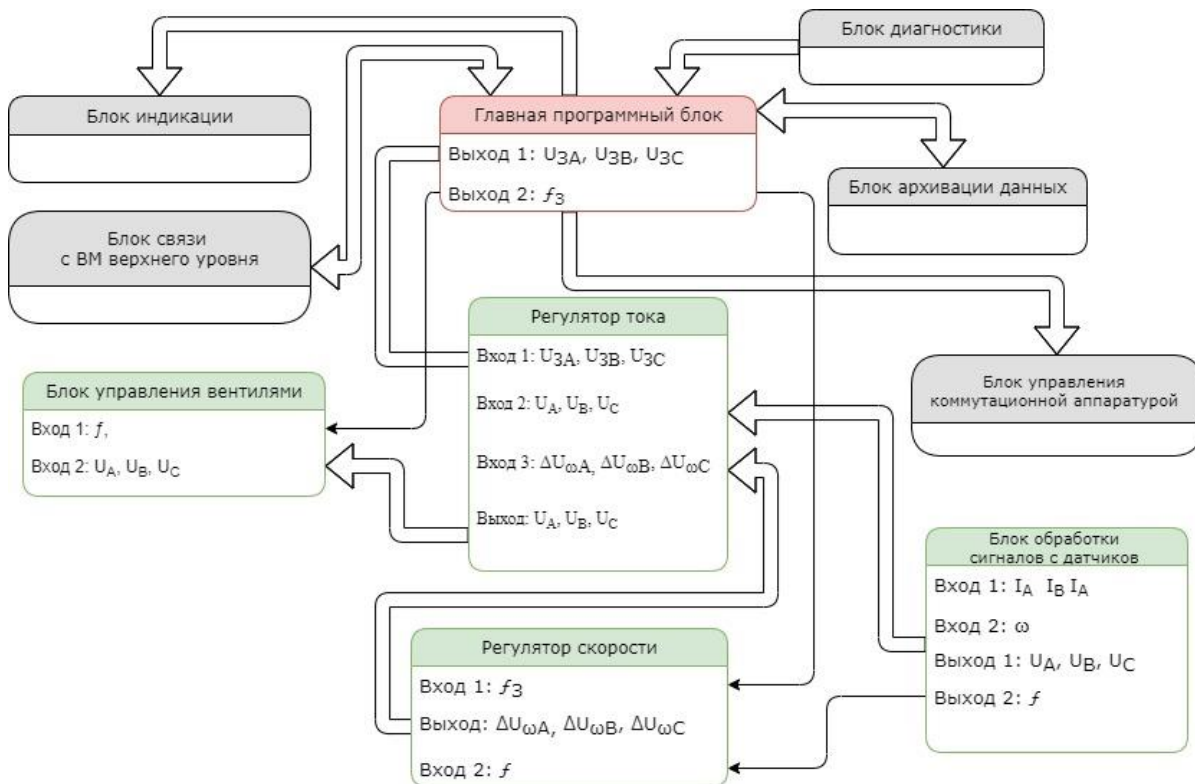


Рисунок 13 – Функциональная схема ПО

Архитектура ПО включает в себя 10 программных блоков.

Главный программный блок

Координирует работу всех остальных блоков, является главенствующим по отношению к остальным. Коммутирует с вычислительным устройством верхнего уровня посредством программного блока связи с ВМ верхнего уровня. Принимает на вход информацию для управления, включающий в себя частоту и выходное напряжение инвертера. Задает требуемое напряжение для регулятора тока и частоту для регулятора напряжения.

Блок обработки сигналов с датчиков

Данный программный блок позволяет считывать аналоговые сигналы обратной связи по скорости и выходному току и преобразовывать их в цифровую информацию с помощью АЦП. Связан непосредственно с блоками регуляторов тока и скорости и передает им информацию для дальнейшей обработки.

Блок регулятора тока

Предназначен для улучшения показателей качества управления по контуру тока. Регулирование осуществляется методом сравнения с задающей величиной. После чего информация поступает в блок управления вентилями. С точки зрения схемы управления (рисунок?) является внутренним. С точки зрения теории автоматического управления по своей природе контур тока является быстрым, т.е. обладает относительно малой постоянной времени, а значит, частота вычисления информации в программном блоке регулятора тока должна быть равна частоте ШИМ.

Блок регулятора скорости

Предназначен для улучшения показателей качества управления по контуру скорости. Регулирование осуществляется методом сравнения с задающей скоростью. После чего информация поступает в блок управления вентилями. С точки зрения схемы управления (рисунок?) является внешним. С точки зрения

теории автоматического управления по своей природе контур скорости является медленным, т.е. обладает относительно большой постоянной времени, а значит, частота вычисления информации в программном блоке может быть уменьшена для экономии вычислительных ресурсов.

Блок управления вентилями

На основании принятых данных о требуемой частоте и амплитуде выходного напряжения и в соответствии с выбранным алгоритмом, данный блок управляет коммутацией вентиляей.

Блок диагностики

В этом блоке микропроцессор производит самодиагностику, т. е. диагностику аппаратных и программных средств самой МПСУ. Она реализуется выполнением ряда тестовых программ, позволяющих оценить состояние памяти (оперативной и постоянной) и всех периферийных устройств (АЦП, таймеров, портов ввода/вывода и т. д.). Кроме того, в этом же блоке, путем считывания соответствующей информации оценивается состояние цепей обратной связи, коммутационной аппаратуры, силовой схемы управляемого выпрямителя и т. п. Для реализации диагностики требуются дополнительные схемы, обеспечивающие реализацию процедуры тестирования []. В зависимости от результатов диагностики посылается соответствующий ответ разрешающий или запрещающий работу устройства. Выполняется 1 раз перед запуском АИН.

Блок индикации

Данный программный блок отвечает за отображение основной информации, характеризующей состояние системы. Вызывается по прерыванию в случае изменения параметров индикации.

Блок управления коммутационной аппаратурой в силовой цепи

Служит для управления автоматических выключателей и других устройств защиты в силовой цепи переменного и постоянного токов. Вызывается по требованию ВМ верхнего уровня.

Блок связи с ВМ верхнего уровня

Обеспечивает двустороннюю связь СУ с вычислительной машиной (ВМ) верхнего уровня. Собирает данные полученные в блоках диагностики, архивации и обработки сигналов с датчиков и посылает ВМ и принимает данные для задания частоты и напряжения, и состояния коммутационного оборудования. Вызывается по прерыванию при обращении одного из устройств.

Блок архивации данных

Данный блок необходим для хранения настроек системы и другой персональной информации. Для экономии вычислительных ресурсов вызывается по прерыванию таймера с определенной периодичностью.

4.3 Кодирование ПО

Целью процесса кодирования является разработка исходного кода. При этом код должен быть прослеживаемым, верифицируемым, непротиворечивым и корректно реализующим требования нижнего уровня. Кодирование ПО происходит на основании архитектуры ПО и требований нижнего уровня [2]. Написание кода происходит в двух программных средах.

Предварительная настройка микроконтроллера и инициализация начального кода производятся в программе STM32CubeMX. Данная программа является визуальным графическим редактором для конфигурирования микроконтроллеров семейства STM32, позволяющим генерировать код на основе языка Си; является частью фирменного программного обеспечения STMicroelectronics STM32CubeMX, призванного облегчить труд разработчиков, сократить длительность исполнения и стоимость проектов [16].

В качестве среды разработки была выбрана программа Keil uVision, представляющая собой набор утилит для выполнения полного комплекса мероприятий по написанию программного обеспечения для микроконтроллеров.

Для обеспечения взаимозаменяемости МК семейства STM32, а также для того чтобы абстрагироваться от уровня регистров, для настройки и программирования используется программная библиотека HAL.

4.3.1 Настройка микропроцессора

Конфигурация выходов

Предварительная настройка МК начинается с конфигурации выходов. Ниже представлена таблица 4 с названиями выходов МК, режимом работы и описанием функционального предназначения.

Таблица 4 – Конфигурация выходов

Выход	Режим работы	Описание
PA8	TIM1_CH1	Управление выходом осуществляется первым каналом первого таймера. Служит для управления вентилем VD1.
PA9	TIM1_CH2	Управление выходом осуществляется вторым каналом первого таймера. Служит для управления вентилем VD3.
PA10	TIM1_CH3	Управление выходом осуществляется третьим каналом первого таймера. Служит для управления вентилем VD5.
PA7	TIM1_CH1N	Управление выходом осуществляется первым комплементарным каналом первого таймера. Служит для управления вентилем VD2.
PB0	TIM1_CH2N	Управление выходом осуществляется вторым комплементарным каналом первого таймера. Служит для управления вентилем VD4.
PB1	TIM1_CH2N	Управление выходом осуществляется третьим комплементарным каналом первого таймера. Служит для управления вентилем VD6.
PA0	ADC1_IN0	Является входом для первого канала первого АЦП. Отвечает за измерение тока фазы А.

Продолжение таблицы 4 – Конфигурация выходов

PA1	ADC1_IN1	Является входом для второго канала первого АЦП. Отвечает за измерение тока фазы В.
PA2	ADC1_IN2	Является входом для третьего канала первого АЦП. Отвечает за измерение тока фазы С.
PA3	ADC1_IN3	Является входом для третьего канала первого АЦП. Отвечает за измерение скорости вращения.

Расположение портов ввода-вывода на МК иллюстрирует рисунок 14.

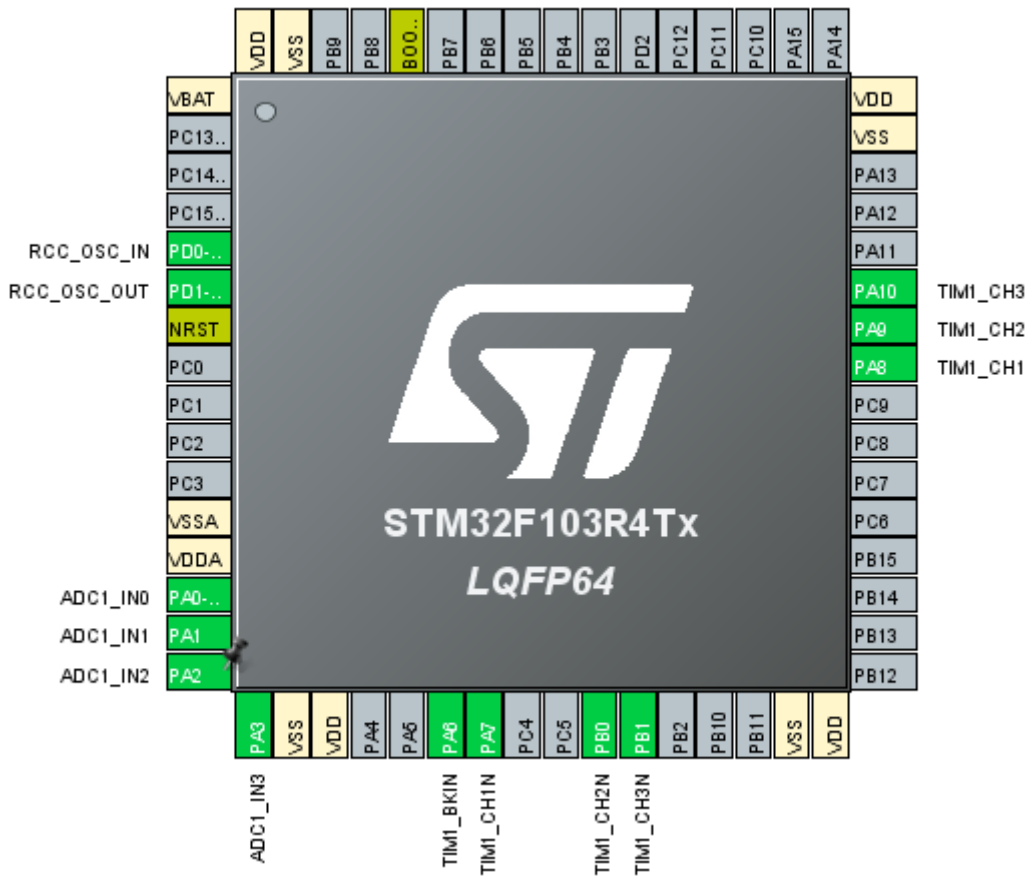


Рисунок 14 – Изображение МК

Настройка тактирования

STM32CubeMX предоставляет упрощенное схематическое изображение всей системы синхронизации контроллера (рисунок 15) с возможностью настройки источников сигналов, делителей и мультиплексоров.

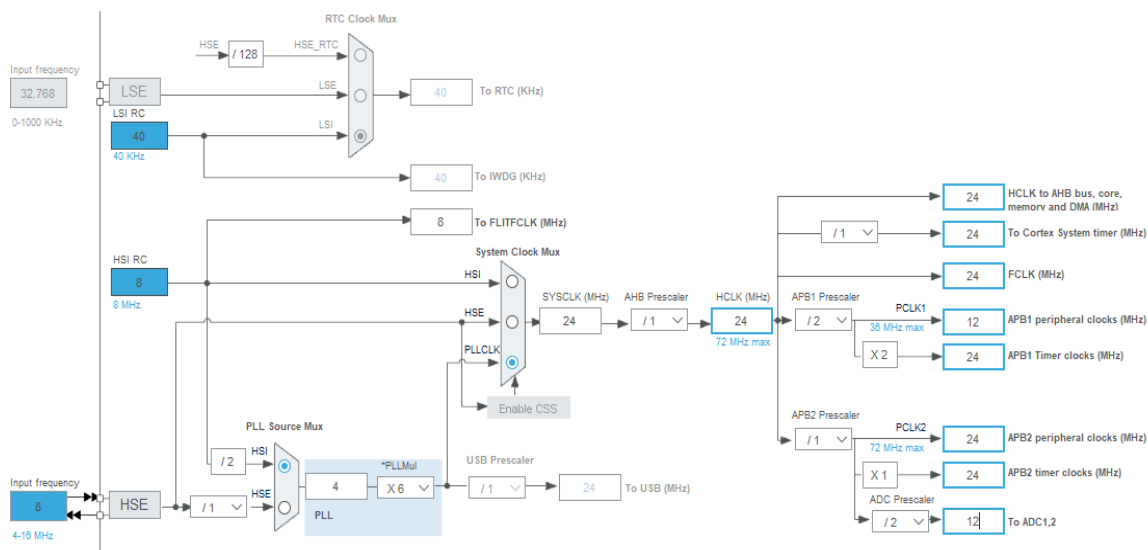


Рисунок 15 - Схема системы синхронизации контроллера

В качестве источника тактирования выбран встроенный в микроконтроллер внутренний HSI RC-генератор (High Speed Internal) на 8 МГц. Выбор связан с тем, что при симуляции работы МК доступен только внутренний генератор. Тактирование шины APB2, отвечающей за работу таймера, настроено на частоту 24 МГц, тактирование шины ADC 12, отвечающей за работу АЦП, - на частоту 12 МГц. Настройки остальных устройств выставлены автоматически по умолчанию.

Настройка таймеров

Для реализации ШИМ необходим таймер класса Advanced-control. Таковым является первый таймер TIM1. Он имеет широкую сферу применения, и в частности позволяет реализовать аппаратную трехканальную ШИМ.

Таймер имеет 4 независимых канала, три из которых помимо основных имеют комплементарные выходы, способные выдавать инвертированный сигнал. Основные параметры таймера представлены в таблице 5

Таблица 5 – параметры таймера TIM1

Параметр таймера	Значение	Описание
Counter Mode	Up	Счетчик будет считать от нуля до значения переполнения
Counter Period	2400	Значение переполнения
Brake State	Enable	Разрешение сигнала сброса
Brake Polarity	Low	Состояние входа для срабатывания сигнала сброса
Dead Time	20	Пауза между переключениями
Chanel 1	PWM Generation CH1 CH1N	Канал в режиме генерации ШИМ с комплементарными выходами
Chanel 2	PWM Generation CH2 CH2N	Канал в режиме генерации ШИМ с комплементарными выходами
Chanel 3	PWM Generation CH3 CH3N	Канал в режиме генерации ШИМ с комплементарными выходами
CH Polarity	High	Состояние выхода при сравнении счетчика и значения в регистре сравнения
CHN Polarity	Low	Состояние комплементарного выхода при сравнении счетчика и значения в регистре сравнения

Параметр «Counter Period» определяет максимальное значение счетчика. Эта величина определяется как максимальное значение, до которого способен досчитать счетчик при тактовой частоте 24 МГц при частоте ШИМ равной 10 кГц. Фактически этот параметр и определяет частоту ШИМ.

Параметр «Brake State» разрешает сигнал сброса. Эта функция позволяет в случае наступления аварийной ситуации перевести все вентили в предустановленное состояние.

Параметр «Brake Polarity» определяет состояние входа для срабатывания сброса.

Функция «Dead Time» позволяет полностью исключить режим сквозных токов. Значение параметра определяется емкостью затвора транзистора и определяется на этапе проектирования силовой части АИН. Фактическое значение времени переключения комплементарных выходов определяется значением параметра «Dead Time» умноженного на период тактового сигнала таймера.

Для работы ШИМ необходимо с некоторой периодичностью извлекать данные, при необходимости обрабатывать их, и заносить в специальный регистр счетчика. Для оптимального использования вычислительных ресурсов МК следует проделывать подобные операции в обработчике прерывания. В разделе «Регулирование частоты» было предложено в качестве инициатора прерывания использовать дополнительный таймер TIM2. Основные параметры настройки таймера представлены в таблице 6.

Таблица 6 – Параметры таймера TIM2

Параметр таймера	Значение	Описание
Counter Mode	Up	Счетчик будет считать от нуля до значения переполнения
Counter Period	1200	Значение переполнения
Prescaler	1	Таймер тактируется с частотой, в 2 раза меньше чем частота тактирование шины APB2
Chanel 1	Output Compare No Output	Канал в режиме сравнения без физического выхода.

Для работы прерывания необходимо активировать параметр TIM2 global interrupt.

Настройка АЦП

Для измерения параметров необходимо настроить 4 канала АЦП. Основные параметры настройки АЦП представлены в таблице?

Таблица 7 – Параметры АЦП

Параметр АЦП	Значение	Описание
Data Alignment	Right alignment	Выравнивание по правому краю
Number of Conversion	4	Кол-во каналов для опроса
Sampling Time	14	Время в тактах затрачиваемое на преобразование. Определяется для каждого канала.
External Trigger Conversion Source	By software	Программный запуск АЦП

В данном МК имеется 2 АЦП с 16 каналами в каждом.

4.3.2 Кодирование блока управления вентилями

Согласно выбранной архитектуры ПО данный программный блок принимает на вход два параметра: напряжение фаз и заданную частоту модулирующего сигнала. Для оптимизации работы программы данный блок будет вызываться по прерыванию. Инициатором прерывания будет событие переполнение счетчика таймера, отвечающего за ШИМ.

Генерация ШИМ сигнала производится в режиме сравнения. На первый вход будет подаваться значение счетчика, на второй – значение модулирующей функции. Принцип формирования ШИМ сигнала для трех каналов продемонстрирован на рисунке 16.

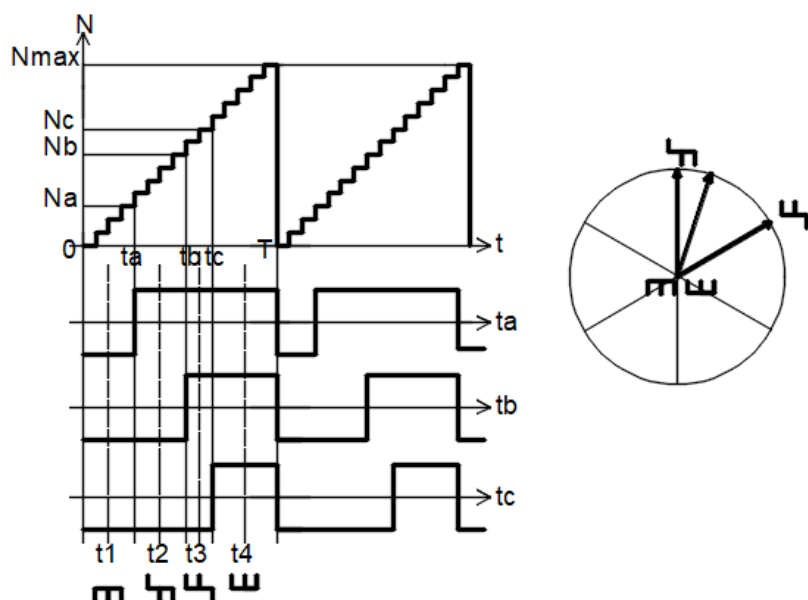


Рисунок 16 – Эпюры сигналов и их векторное представление

Из рисунка видно, что в зависимости от значения счетчика для каждого канала меняется время включения вентиля; в независимости от значения N_a , N_b и N_c за период ШИМ сигнала существуют 4 состояния вентиля, два из которых нулевые. Каждому из состояний соответствует один из шести базисных вектор. Изменяя очередность включения каждого канала, меняется ориентация результирующего вектора.

Как и было сказано в подразделе «Регулирования частоты», значения функции для каждого аргумента из фиксированного ряда записаны в специальную таблицу значений функции.

В таблицу записываются значения функции согласно формуле:

$$S_{[k]} = \sin \frac{\pi}{N_{cm}} (2k + 1),$$

где N_{cm} - число значений фазы, на которое разбивается диапазон ее изменения;

k – номер ступени, лежащий в промежутке от 0 до $N_{cm} - 1$.

Видоизменив функцию получим:

$$S_{[k]} = \left(1 + \sin \frac{\pi}{N_{cm}} (2k + 1) \right) \frac{S_{max}}{2} \pm \beta,$$

где β – дробная величина для округления $S_{[k]}$ до целого числа;

S_{max} – максимально возможное значение счетчика, равное:

$$S_{max} = \frac{24MГц}{10кГц} - 1 = 2399$$

В приложении 1, таблица 1 представлены все значения функции, записываемые в таблицу.

Число значений фазы зависит от частоты ШИМ $f_{ШИМ}$ и максимальной частоты модулирующего сигнала f_c и определяется как

$$N_{cm} = \frac{f_{ШИМ}}{f_c} = \frac{10кГц}{50Гц} = 200$$

Регулирование частоты

Для регулирования частоты будет использован четвертый способ регулирования путем изменения частоты прерываний. В МК STM32F103R4 имеется дополнительный таймер.

Таймер TIM2 тактируется с частотой:

$$f_{TIM} = \frac{f_{APB2}}{Prescaler + 1} = \frac{24\text{МГц}}{1 + 1} = 12\text{МГц}$$

где f_{APB2} – частота тактирования шины APB2.

Прерывание настроено на переполнение счетчика таймера. Для регулирования частоты переполнения отвечает специальный 16-битный регистр, доступ к которому возможен через параметр Counter Period. Этот параметр определяет частоту прерываний согласно формуле:

$$f_{INT} = \frac{f_{TIM}}{CounterPeriod}$$

При этом частота прерываний должна быть равна

$$f_{INT} = N_{cm} f_c$$

Тогда можно найти Counter Period как

$$CounterPeriod = \frac{f_{TIM2}}{f_{INT2}} = \frac{f_{TIM2}}{N_{cm} f_c}$$

При максимальной частоте сигнала $f_c = 50$ Гц значение Counter Period будет равно:

$$CounterPeriod = \frac{12\text{МГц}}{200 \cdot 50} = 1200.$$

При минимальной частоте $f_c = 1$ Гц его значение будет:

$$CounterPeriod = \frac{12MГц}{200 \cdot 1} = 60000$$

Из полученного выражения видно, что максимальный диапазон регулирования частоты при такой настройке находится в пределах от 60 кГц до 0,91 Гц. Для увеличения его максимального значения следует увеличить частоту тактирования, либо число точек $N_{ст}$. Для еще большего уменьшения минимального значения частоты следует уменьшить частоту тактирования.

Можно найти шаг квантования частотного диапазона регулирования:

$$\begin{aligned} \Delta fc &= fc[k] - fc[k-1] = \frac{f_{TIM2}}{N_{cm} \cdot CounterPeriod_{[k]}} - \frac{f_{TIM2}}{N_{cm} \cdot CounterPeriod_{[k-1]}} = \\ &= \frac{f_{TIM2}}{N_{cm}} \left(\frac{1}{CounterPeriod_{[k]}} - \frac{1}{CounterPeriod_{[k]} + 1} \right) = \\ &= \frac{f_{TIM2}}{N_{cm}} \frac{1}{CounterPeriod_{[k]} \cdot CounterPeriod_{[k]} + 1} \end{aligned}$$

При частоте сигнала $f_c = 50$ шаг квантования будет равен 0,041 Гц, при $f_c = 1$ шаг квантования будет 0,00003 Гц.

Исходя из всего вышесказанное, можно утверждать, что предложенный метод позволяет получить условно широкий диапазон регулирования частоты с малым шагом квантования.

Регулирование амплитуды

В этом случае все гораздо проще чем регулирование частоты. Для регулирования амплитуды требуется при извлечении данных из таблицы, предварительно перед занесением в регистр счетчика, делить на число, определяемое выражением:

$$M = \frac{A_{\max}}{A_3}.$$

где A_{\max} – максимальная амплитуда ШИМ сигнала,

A_3 – амплитуда задаваемая регулятором тока, меньшая или равная A_{\max} .

Текущее значение сигнала будет определяться как:

$$N = \left[\frac{S_{[k]}}{M} \right],$$

где квадратными скобками обозначено целочисленное деление, именуемое операцией антье.

4.4. Верификация ПО

Назначение верификации ПО [2] состоит в том, чтобы обнаружить и зарегистрировать ошибки, которые могли быть внесены в ПО во время его разработки (устранение ошибок является задачей разработки ПО). Основное назначение верификации ПО - проверить что:

- системные требования, предназначенные для программной реализации, были должным образом переработаны в требования верхнего уровня к ПО, которые удовлетворяют этим системным требованиям;
- требования верхнего уровня были переработаны в архитектуру ПО и требования нижнего уровня, которые удовлетворяют требованиям верхнего уровня; если разработано несколько уровней требований к ПО между требованиями верхнего уровня и требованиями нижнего уровня, то каждый последующий уровень требований разработан так, чтобы удовлетворять требованиям более высокого уровня;
- архитектура ПО и требования нижнего уровня должным образом преобразованы в исходный код, удовлетворяющий им;
- - исполняемый объектный код удовлетворяет требованиям к ПО;
- инструментальные средства, используемые для выполнения указанных работ, являются технически корректными и полными для заданного уровня ПО.

Процесс верификации включает в себя также процесс тестирования ПО. Тестирование ПО имеет две взаимодополняющие цели. Первая цель - показать, что ПО удовлетворяет требованиям к нему. Вторая цель - продемонстрировать с высокой степенью доверия, что были устранены ошибки, которые могли бы привести к возникновению отказных ситуаций, определенных процессом оценки безопасности системы.

Для достижения целей тестирования ПО может потребоваться более одной среды тестирования. Идеальная среда тестирования включает в себя ПО, загруженное в объектный вычислитель и тестируемое в среде, которая

имитирует среду объектного вычислителя с высокой точностью. Возможно сертификационное доверие для эмулятора или имитатора объектного вычислителя на инструментальном компьютере. Однако рекомендации относительно среды тестирования сводятся к следующему: некоторые тесты должны быть выполнены только в интегрированной объектной вычислительной среде, так как некоторые ошибки могут быть обнаружены только в этой среде [2].

В данной работе в качестве среды тестирования выбрана среда моделирования ISIS Proteus. Была собрана упрощённая модель АИН и МПСУ (рисунок 17), включающая в себя объектный вычислитель, МК STM32F103R4.

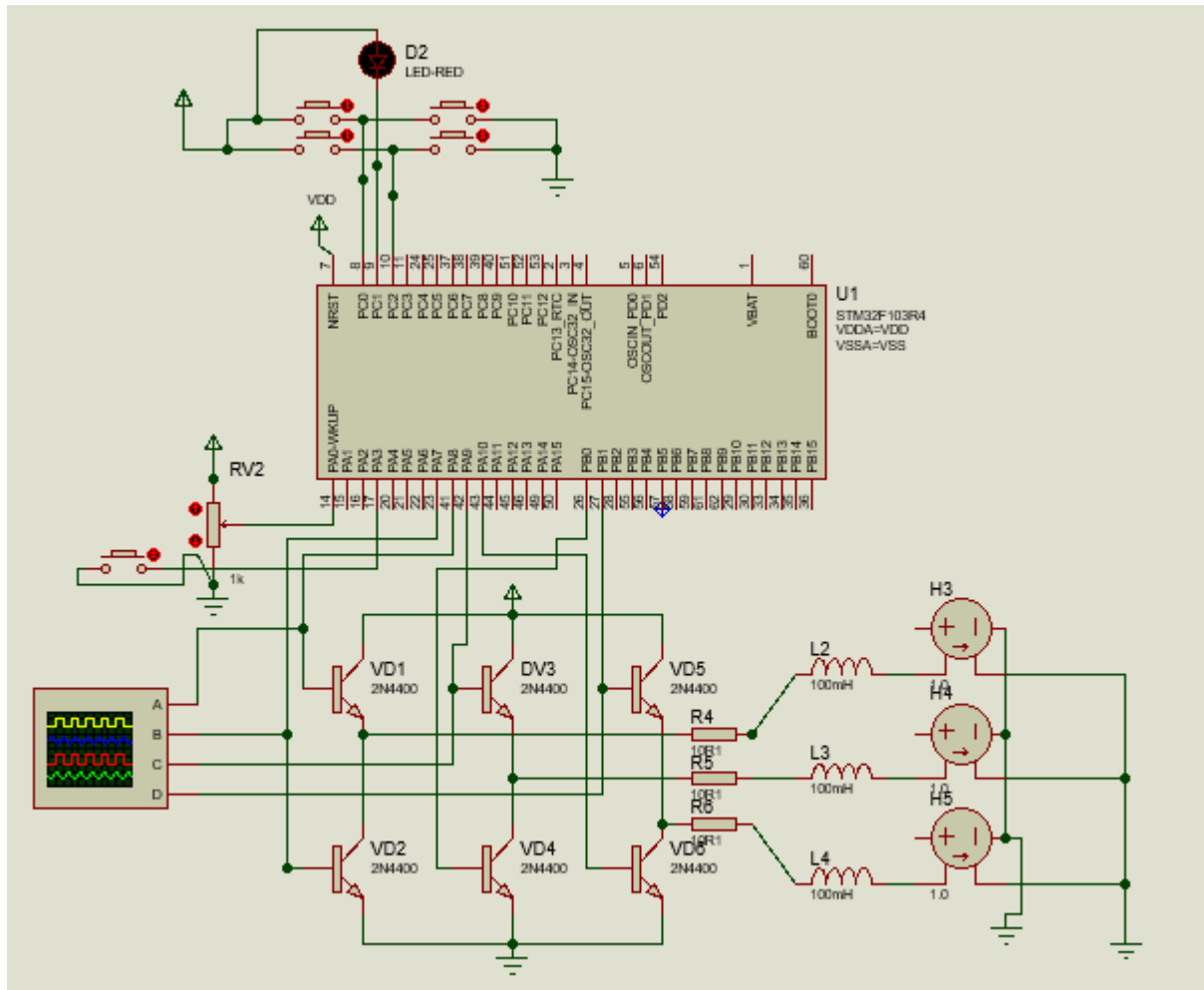


Рисунок 17 – Модель АИН и МПСУ в среде Proteus

На рисунке 18 изображены осциллограммы управляющих сигналов ВВ.

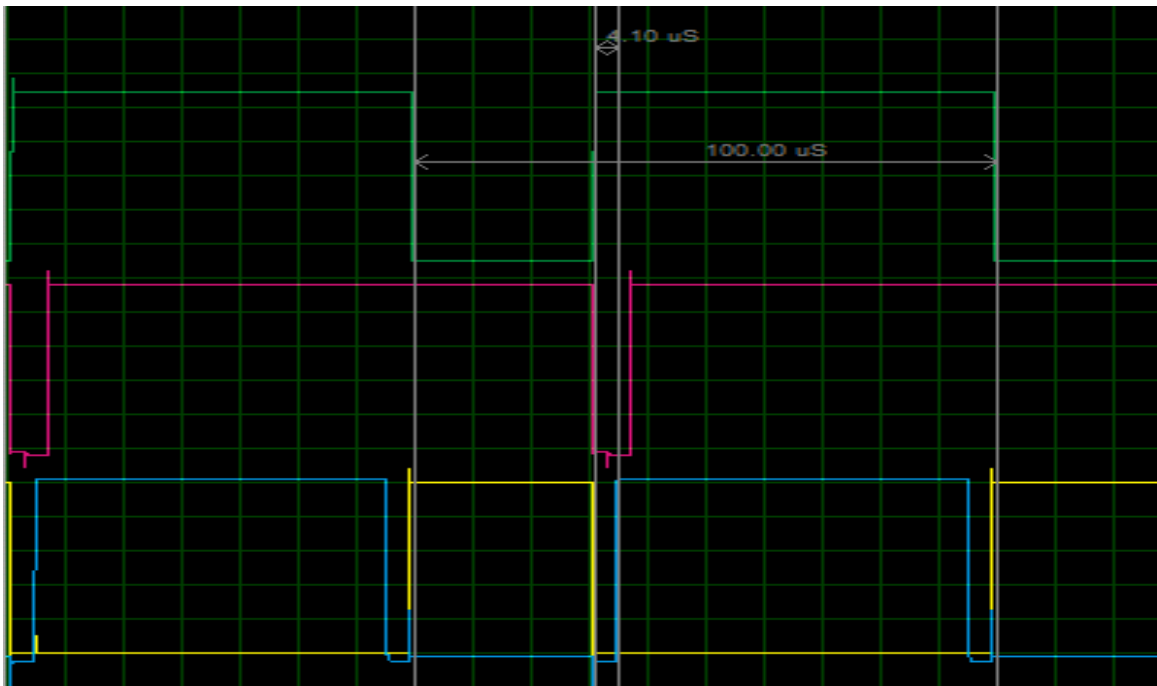


Рисунок 18 – Осциллограммы управляющих сигналов.

Из рисунка видно, что период ШИМ сигнала составляет 100 мкс, что соответствует частоте несущего сигнала 10 кГц. Внизу изображены два элементарных сигнала (синий и жёлтый). Можно заметить, что между ними есть запаздывание, равное 4 мкс. Это и есть так называемое «мертвое время», позволяющее избежать сквозных токов, вследствие немгновенного записывания вентилей.

На рисунке 19 изображены осциллограммы трехфазного выходного тока.

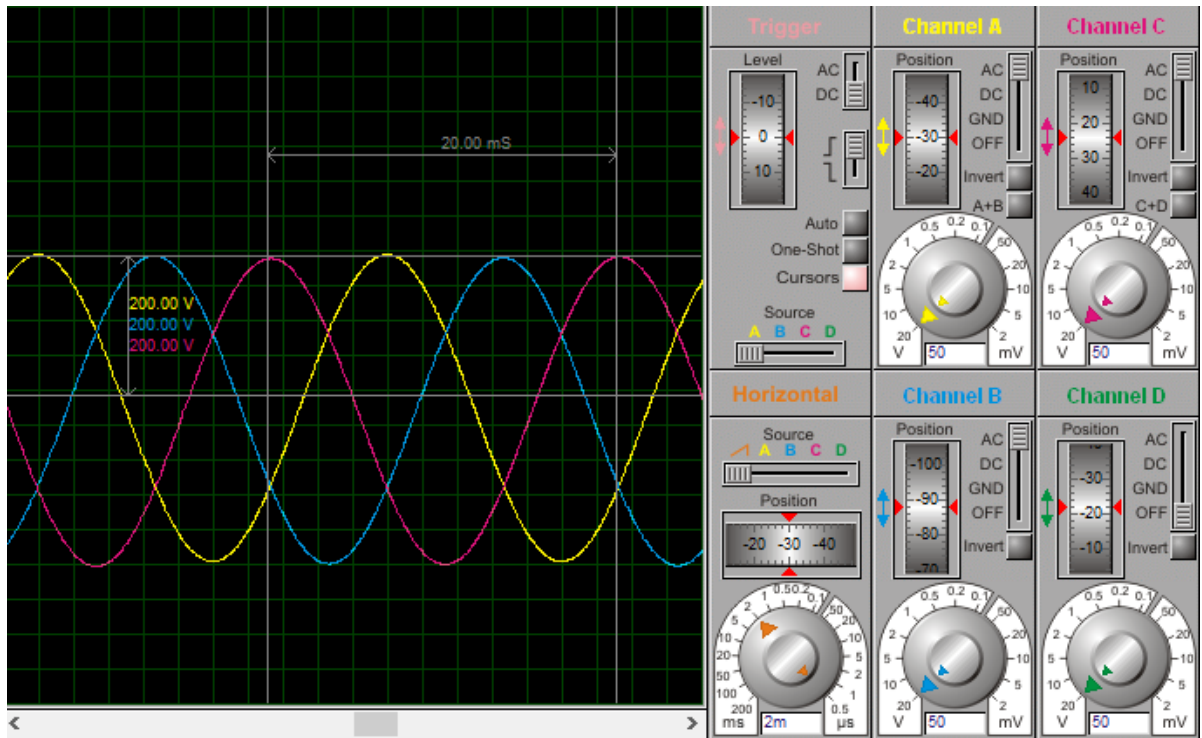


Рисунок 19 – Осциллограммы выходного тока

Из рисунка видно, что период выходного тока составляет 20 мс, что соответствует частоте 50 Гц.

5. ЭКОНОМИЧЕСКОЕ ОБОСНОВАНИЕ ВКР

5.1 Концепция

Целью ВКР является разработка программного обеспечения СУАИН в соответствии с техническими требованиями, указанными в задании на ВКР.

Процесс создания программного обеспечения включая в себя выполнение следующих этапов, рекомендуемых [2]:

- анализ существующих алгоритмов управления вентильным блоком АИН;
- анализ существующих методов микропроцессорной реализации СУ;
- определение требований к ПО;
- проектирование ПО;
- кодирование ПО;
- интеграция и верификация ПО методом компьютерного моделирования;
- оформления документации (пояснительной записки).

В условиях современной экономики, требуется решать задачи оптимизации. Критериями оптимизации в общем случае являются цена и качество готового ПО. При этом при проектировании важно учитывать себестоимость готового ПО, с целью удешевления и повышения эффективности выполнения отдельных его этапов. По этой причине себестоимость является главным предметом рассмотрения экономической составляющей вопроса проектирования.

5.2 График выполнения ВКР

Для расчёта себестоимости необходимо рассчитать заработную плату всех участников проекта, для этого необходимо составить график проводимых работ. В таблице 8 приведены основные этапы разработки ПО.

Таблица 8 – Трудоёмкость работ

Наименование работ	Трудоёмкость, чел./дни	
	Исполнитель	Руководитель
Разработка ТЗ.	5	2
Анализ существующих алгоритмов управления вентильным блоком АИН.	10	1
Анализ существующих методов микропроцессорной реализации СУ.	10	1
Определение требований к ПО.	2	0
Проектирование ПО.	3	0
Кодирование ПО.	20	0
Интеграция и верификация ПО методом компьютерного моделирования.	10	0
Оформления документации (пояснительной записки).	10	1

Итого: 75 чел./дни.

5.3 Себестоимость разработки

Себестоимость рассчитывается на основе калькуляции, главными пунктами которой являются [17]:

- расходуемые материалы;
- основная з/п;

- дополнительная з/п;
- социальные нужды;
- накладные расходы
- амортизационные отчисления
- себестоимость.

5.3.1 Затраты на расходуемые материалы

В данном разделе учитываются все материалы, используемые в разработке. Все затраты приведены в таблице 9.

Таблица 9 –Расходуемые материалы

Материалы	Количество, шт.	Цена, руб..	Сумма, руб..
Бумага А4	1	235,00	235,00
Ручка	1	40,00	40
Дипломная папка	2	225,00	450,00
Итого:			500

5.3.2 Основная заработная плата

Зарботную плату рассчитываем на основе тарифной ставки, принимая 21 рабочий день в месяце. Основная заработная плата определяется по формуле:

$$Z_{\text{осн.з/пл}} = \sum_{i=1}^k T_i C_i,$$

где $Z_{\text{осн.з/пл}}$ - расходы на основную заработную плату исполнителей руб.; k – количество исполнителей; T_i - время, затраченное i -м исполнителем на проведение исследования (дни или часы); C_i - ставка i -го исполнителя руб./день или руб./час

Рассчитаем дневную З/П

$$D_{ст} = \frac{46250}{21} = 2202 \text{ руб.},$$

$$D_p = \frac{52500}{21} = 2500 \text{ руб.};$$

где $T_{ст}$ – Затраченное время работы инженера (студента); T_p – Затраченное время работы руководителя; D_p – дневная ставка руководителя; $D_{ст}$ – дневная ставка студента.

Рассчитаем основную заработную плату

$$Z_{осн.з/пл} = 70 \cdot 2202 + 5 \cdot 2500 = 166670 \text{ руб.}$$

Основная заработная плата составляет 166670 рублей.

5.3.3 Дополнительная заработная плата

К статье «Дополнительная заработная плата» относятся выплаты, предусмотренные законодательством за непроработанное время, оплата отпусков, заработная плата лицам, не достигшим 18 лет и т.п.

$$Z_{доп.з/пл} = Z_{осн.з/пл} \cdot \frac{N_{доп}}{100},$$

где $Z_{осн.з/пл}$ – расходы на дополнительную заработную плату исполнителей, руб.; $Z_{осн.з/пл}$ – расходы на основную заработную плату исполнителей, руб.; $N_{доп}$ – норматив дополнительной заработной платы. В данном случае он равен 14%.

$$Z_{доп.з/пл} = 166670 \cdot 0,14 = 23339 \text{ руб.}$$

Дополнительная заработная плата составила 23339 рублей.

5.3.4 Затраты на электроэнергию для технологических целей

В настоящее время каждый проект не обходится без использования электрооборудования, поэтому важной частью в оценке себестоимости работы является расчет расходов на электроэнергию.

Затраты на электроэнергию определяются в соответствии со следующим выражением:

$$Z_{\text{э}} = \sum_{i=1}^m N_i \cdot t_i \cdot g_i \cdot T_{\text{э}},$$

где m – количество видов специального оборудования, необходимого для реализации проекта; t_i – время использования i – го вида оборудования в исследовательских целях, час; g_i – коэффициент использования мощности i -го вида оборудования; $T_{\text{э}}$ – стоимость одного кВт/ч электроэнергии, которая определяется местной энергетической компанией и зависит от местоположения организации или предприятия, занимающейся разработкой проекта. Все результаты по данной статье затрат объединим и поместим в таблицу 9.

Таблица 9 – Затраты на электроэнергию

Наименование оборудования	Мощность оборудования, кВт	Время использования, час.	Коэффициент использования мощности оборудования	Стоимость одного кВт/час электроэнергии, руб.	Итого материальных затрат, руб.
Компьютер	0,15	560	0,8	4,56	306

Итого: 306 рублей составляют затраты на электроэнергию.

5.3.5 Затраты на социальные нужды

В отчисления на социальные нужды вход на обязательное социальное, пенсионное и медицинское страхование с основной и дополнительной заработной платы. В СПбГЭТУ «ЛЭТИ» ставку отчислений на страховые взносы принято считать равной 30%. Страховые отчисления определяются по формуле:

$$Z_{\text{соц}} = (Z_{\text{осн.з/пл}} + Z_{\text{доп.з/пл}}) \cdot \frac{H_{\text{соц}}}{100},$$

где $H_{\text{соц}}$ – норматив отчислений на страховые взносы на обязательное социальное, пенсионное и медицинское страхование.

$$z_{\text{соц}} = (166670 + 23339) \cdot 0,3 = 57002 \text{ руб.}$$

Отчисления на социальные нужды составили 57002 рублей.

5.3.6 Затраты накладные расходы

В состав накладных расходов входят издержки на освещение и отопление, амортизацию машин и оборудования общего пользования. В СПбГЭТУ «ЛЭТИ» рассчитываются по ставке 34%. Размер накладных расходов определяется по формуле:

$$z_{\text{н.р.}} = (z_{\text{осн.з/пл}} + z_{\text{доп.з/пл}}) \cdot \frac{H_{\text{н.р.}}}{100},$$

где $H_{\text{н.р.}}$ – норматив накладных расходов.

$$z_{\text{н.р.}} = (166670 + 23339) \cdot 0,34 = 64603 \text{ руб.}$$

Накладные расходы составили 64603 рублей.

5.3.7 Амортизационные отчисления

В данном случае учитывает амортизация как оборудования, так и вспомогательного программного обеспечения. Амортизация определяется по нормам амортизации линейным методом. Норматив амортизации составит:

$$H_{\text{ам}} = \frac{1}{T} \cdot 100\% = 20\%$$

где $T = 5$ лет – срок полезного использования.

$H_{\text{ам}}$ – норматив амортизации.

Амортизационные отчисления по основному средству i за год определяются как:

$$A_i = \frac{C_{\text{ни}} \cdot H_{\text{ai}}}{100},$$

где A_i – амортизационные отчисления за год по i -му основному средству (руб.); $C_{\text{пн},i}$ – первоначальная стоимость i -го основного средства (руб.); H_{ai} – годовая норма амортизации i -го основного средства (%).

Величина амортизационных отчислений по i -му основному средству, используемому студентом при работе над ВКР, определяется по формуле:

$$A_{i \text{ ВКР}} = \frac{A_i \cdot T_{i \text{ ВКР}}}{T},$$

где $A_{i \text{ ВКР}}$ - амортизационные отчисления по i -му основному средству, используемому студентом в работе над ВКР (руб.); A_i – амортизационные отчисления за год по i -му основному средству (руб.); $T_{i \text{ ВКР}}$ – время, в течение которого студент использует i -ое основное средство (мес.).

Пример расчета амортизационных отчислений для компьютера приведен ниже.

$$A_k = \frac{C_{\text{пнк}} \cdot H_{ak}}{100} = \frac{62000 \cdot 20}{100} = 12400,$$

$$A_{k \text{ ВКР}} = \frac{A_k \cdot T_{k \text{ ВКР}}}{T} = \frac{12400 \cdot 3}{12} = 3100,$$

Основные средства, стоимость и время их использования представлены в таблице 10.

Таблица 10 – Амортизационные отчисления

Основные средства	Стоимость, руб.	Амортизационные отчисления, руб.	Время использования, мес.
Ноутбук «Acer Nitro 5»	62000	3100	3
Программный пакет «ProteusVSM for ARM Cortex M0»	40000	2000	3
Программный пакет «MS Office 365»	3000	150	3

Итого амортизационные отчисления составят 5250 рублей.

5.3.8. Себестоимость

Для расчёта всех затрат сведём все данные в таблицу 11.

Таблица 11 – Себестоимость

Статья затрат	Сумма, руб.
Расходуемые материалы	500
Основная З/П	166670
Дополнительная З/П	23339
Социальные нужды	57002
Накладные расходы	64603
Амортизационные отчисления	5250
Затраты на электроэнергию	306
Итого себестоимость	317670

5.4. Заключение по главе

Итоговая себестоимость составила 317670 рублей. О величии полученной суммы, можно судить, зная количество устройств (АИН), для которых предназначено это ПО. Очевидно, что для штучной партии создание ПО экономически неэффективно. В таком случае лучше воспользоваться готовыми решениями. Основную долю всех затрат составляют затраты на основную и дополнительную заработные платы. Эту статью можно уменьшить, применяя типовые решения при создании ПО, тем самым уменьшая трудоемкость.

ЗАКЛЮЧЕНИЕ

В результате выполнения работы был проведен обзор современных решений микропроцессорной реализации СУ АИН, изучены и предложены уникальные методы. Для создания ПО были решены все поставленные задачи и проделаны все этапы, рекомендуемые ГОСТом. Произведено математическое моделирование адаптивной системы управления на основе нечеткой логики, внедренной в МПСУ АИН.

Были доведены до стадии готового кода программный блок управления вентилями, блок регулирования частоты и амплитуды.

Материалы и результаты данной работы могут быть полезны при написании учебного пособия по созданию ПО, а также для создания готового лабораторного стенда.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Требования к выполнению и защите выпускной квалификационной работы бакалавра, магистра, специалиста в СПбГЭТУ «ЛЭТИ» //Официальный сайт СПбГЭТУ. URL: https://etu.ru/assets/files/obrazovatel'naya/dokumenty-dlya-obrazovatel'nogo-processa/prikaz_od_0210_pril2_trebovaniya-k-vypolneniyu-i-zashhite-vypusknj-kvalifikacionnoj-raboty.pdf (дата обращения: 30.05.2020).
2. ГОСТ Р 51904-2002. Программное обеспечение встроенных систем. Общие требования к разработке и документированию. М.: Госстандарт России, 2002.
3. Баховцев И.А. Микропроцессорные системы управления устройствами силовой электроники. Структуры и алгоритмы: учеб. пособие. Новосибирск: Изд-во НГТУ, 2018. – 219 с.
4. Зиновьев Г.С. Основы силовой электроники: учеб. пособие. Новосибирск: Изд-во НГТУ, 2004. – 671 с.
5. Солонина А.И. Алгоритмы и процессоры цифровой обработки сигналов: учеб. пособие. Санкт – Петербург: Изд-во «БХВ-Петербург», 2002. - 445с.
6. Шрейнер Р.Т. Математическое моделирование электроприводов переменного тока с полупроводниковыми преобразователями частоты. Екатеринбург: Изд-во УРО РАН, 2000. – 654 с.
7. Структура и архитектура микропроцессоров современных персональных электронных вычислительных машин // Московский государственный университет печати. URL: <http://www.hi-edu.ru/e-books/xbook112/01/part-005.htm> (дата обращения: 30.05.2020).
8. Булатов О.Г. Автономные тиристорные инверторы с улучшенной формой выходного напряжения: учеб. пособие. Кишинев: Изд-во Штиинца, 1980. – 113 с.

9. Каган Б.М. Основы проектирования микропроцессорных устройств автоматики. Москва: Изд-во Энергоатомиздат, 1987. – 304 с.

10. Ремизевич Т.В. Микроконтроллеры для встраиваемых приложений: от общих подходов к семействам HC05 и HC08 фирмы Motorola / под ред. И.С. Кирюхина. Москва: ДОДЭКА, 2000. – 272 с.

11. Бернштейн А.Я. Тиристорные преобразователи частоты в электроприводе / А.Я. Бернштейн, Ю.М. Гусяцкий и др. М.: Энергия, 1980. – 328 с.

12. Использование микроконтроллера для прямого цифрового синтеза сигнала //Микропроцессорная электроника. URL: http://www.rotr.info/electronics/mcu/stm32_dac_table_sin.htm (дата обращения: 30.05.2020).

13. Построение и настройка нечеткого адаптивного ПИД регулятора/ Ю. И. Кудинов., В.А. Колесников, И.С. Дургарян и др. // Интеллектуальные системы. 2016, вып. (№3) 49. С. 87–96.

14. Методы нечеткой логики в задачах автоматизации тепловых процессов электростанций / С.И. Новиков, В.Р. Шахнович, А.В. Сафронов // «Вестник ИГЭУ.2010, вып. №4.С. 1 – 4.

15. Синтез нечёткого регулятора при помощи пакета прикладных программ системы Matlab / А.В. Кузнецов, Г.М. Макарьянц // Самара: Издательство СНИУ, 2016 – 62 с.

16. CubeMX и Workbench: создание проекта на базе STM32 с помощью бесплатного ПО //Компэл. URL: <https://www.compel.ru/lib/75317> (дата обращения: 30.05.2020).

17. Алексеева О.Г. Методические указания по экономическому обоснованию выпускных квалификационных работ бакалавров. СПб.: Изд-во СПбГЭТУ “ЛЭТИ”, 2013. с.

ПРИЛОЖЕНИЕ А

Таблица 1 – Таблица значений модулирующей функции

k	S[k]	k	S[k]	k	S[k]	k	S[k]
0	1218	50	2399	100	1181	150	1
1	1256	51	2398	101	1143	151	1
2	1294	52	2395	102	1105	152	4
3	1331	53	2392	103	1068	153	7
4	1369	54	2387	104	1030	154	12
5	1406	55	2381	105	993	155	18
6	1443	56	2374	106	956	156	25
7	1480	57	2366	107	919	157	33
8	1516	58	2356	108	883	158	43
9	1552	59	2346	109	847	159	53
10	1588	60	2334	110	811	160	65
11	1623	61	2322	111	776	161	77
12	1659	62	2308	112	740	162	91
13	1693	63	2293	113	706	163	106
14	1727	64	2277	114	672	164	122
15	1761	65	2260	115	638	165	139
16	1794	66	2241	116	605	166	158
17	1826	67	2222	117	573	167	177
18	1858	68	2202	118	541	168	197
19	1889	69	2181	119	510	169	218
20	1920	70	2159	120	479	170	240
21	1949	71	2136	121	450	171	263
22	1979	72	2112	122	420	172	287
23	2007	73	2087	123	392	173	312
24	2034	74	2061	124	365	174	338
25	2061	75	2034	125	338	175	365
26	2087	76	2007	126	312	176	392
27	2112	77	1979	127	287	177	420
28	2136	78	1949	128	263	178	450
29	2159	79	1920	129	240	179	479
30	2181	80	1889	130	218	180	510
31	2202	81	1858	131	197	181	541
32	2222	82	1826	132	177	182	573
33	2241	83	1794	133	158	183	605
34	2260	84	1761	134	139	184	638
35	2277	85	1727	135	122	185	672
36	2293	86	1693	136	106	186	706
37	2308	87	1659	137	91	187	740
38	2322	88	1623	138	77	188	776
39	2334	89	1588	139	65	189	811
40	2346	90	1552	140	53	190	847

Продолжение таблицы 1 – Таблица значений модулирующей функции

41	2356	91	1516	141	43	191	883
42	2366	92	1480	142	33	192	919
43	2374	93	1443	143	25	193	956
44	2381	94	1406	144	18	194	993
45	2387	95	1369	145	12	195	1030
46	2392	96	1331	146	7	196	1068
47	2395	97	1294	147	4	197	1105
48	2398	98	1256	148	1	198	1143
49	2399	99	1218	149	1	199	1181