

Автоматическое согласование электроприводов без обратной связи с применением протокола ModBus

А.Н. Долидзе

Государственный университет аэрокосмического приборостроения, Санкт-Петербург

Аннотация: В статье рассматривается разработка программного модуля управления электроприводами в составе промышленной сети ModBus, предназначенного для модернизации автоматизированной системы управления участка производственной линии хлебопекарного завода «Заря». В существующей системе управления для изменения скорости работы двигателей применяются преобразователи частоты, при этом оператору приходится вручную управлять каждым преобразователем. Управление скоростью вращения двигателей производится путём задания значения частоты, что неудобно для оператора. При таком методе управления невозможно одновременно изменить скорость работы всего оборудования линии, что вызывает рассогласование в работе оборудования. Для решения обозначенной проблемы, на основе уже имеющегося оборудования, строится сеть на основе протокола ModBus. В статье приводится описание основных особенностей протокола ModBus, разрабатывается метод, позволяющий оператору указывать требуемое время работы системы вместо значения частоты для каждого двигателя, и описывается разработанный программный модуль, реализующий предложенный метод.

Ключевые слова: программируемый логический контроллер, язык релейной логики, протокол ModBus, управление без обратной связи, преобразователь частоты.

Введение

Рассмотрим процесс модернизации участка производственной линии хлебопекарного завода «Заря». Модернизируемый участок производственной линии отвечает за доставку, заполнение и погрузку форм с тестом в расстоечный шкаф, а также управление скоростью перемещения форм внутри шкафа. В зависимости от рецептуры выпекаемого хлеба, требуется разное время расстойки, регулируемое путём изменения скорости вращения двигателей оборудования (транспортёров, тестоделителя и расстоечного шкафа). Несмотря на то, что система является автоматизированной, регулировка скорости вращения производится вручную, путём задания выходной частоты на преобразователях частоты (далее ПЧ).

Цель разработки – обеспечить возможность автоматической настройки скорости электроприводов линии для достижения указанного оператором времени работы линии в минутах. В состав сети входят (рис. 1):

программируемый логический контроллер (далее ПЛК) SIEMENS (Germany, Berlin) SIMATIC S7-200 CPU 226 [1], ПЧ VFD-E DELTA ELECTRONICS (Taiwan, Taipei) [2], используемые для управления электроприводами, и сенсорная панель оператора. Для согласования устройств будет использован протокол ModBus.

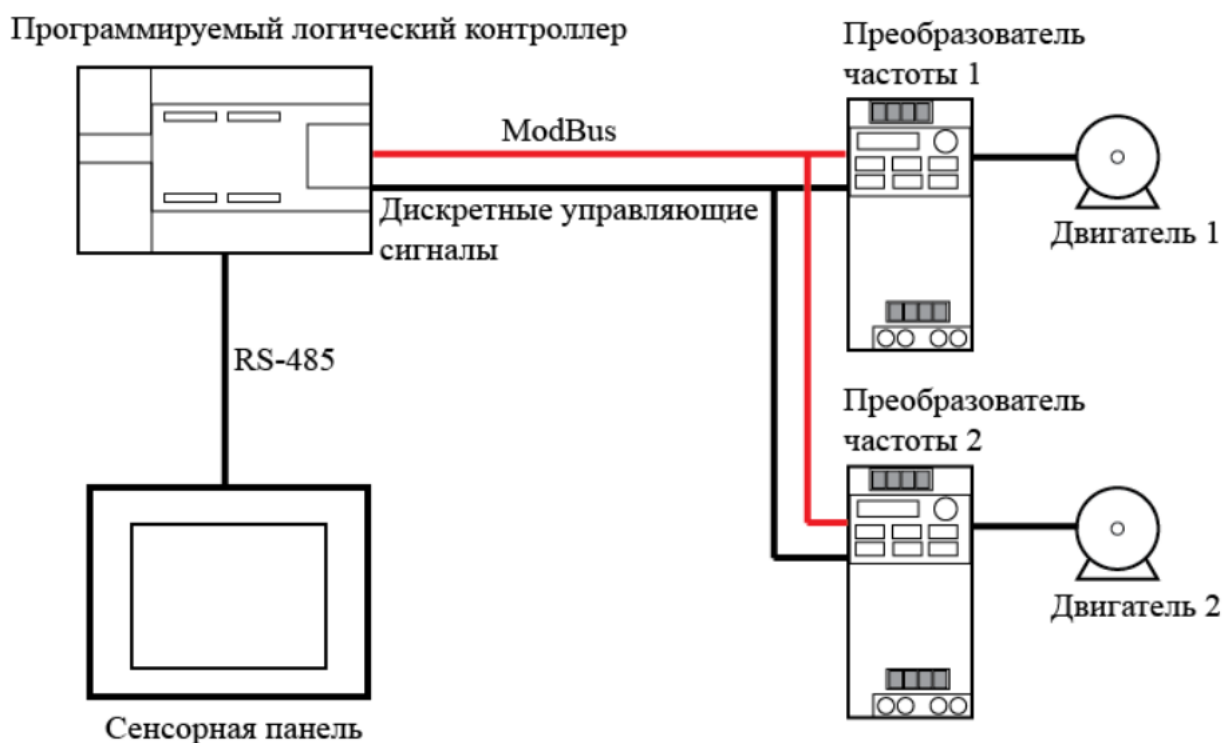


Рис. 1. – Модернизируемая часть системы управления

Протокол ModBus

Применяемый протокол ModBus - открытый коммуникационный протокол для связи между различными электронными устройствами [3]. Большинство современных промышленных систем контроля и управления имеют программные драйвера для работы с ModBus сетями [4]. К недостаткам ModBus можно отнести отсутствие прерываний и невозможность общения между ведомыми устройствами, но, в связи с небольшими размерами сети, эти недостатки не имеют значения. Устройства

в сети делятся на ведущие (master) и ведомые (slave). ПЛК используется в качестве ведущего устройства, а ПЧ в качестве ведомых. Для передачи данных между устройствами используется физический интерфейс RS-485 [5].

В протоколе ModBus для обращения к данным узла используется адрес, состоящий из двух частей (типа данных и локального адреса) [3, 6]. Тип данных определяет размер ячеек памяти 1 или 16 бит и возможные операции над ними. Для данных каждого типа в памяти устройства отводятся специальные таблицы (локальный адрес). Физически все таблицы могут находиться в одной и той же области памяти, в таком случае, адреса 30010 и 40010 будут указывать на одну и ту же ячейку памяти, только с адресом 3xxxx ячейка доступна только для чтения, а с 4xxxx для чтения и записи.

Для работы в промышленных сетях в ПЧ предусмотрены управляющие регистры, с их помощью ПЛК может управлять работой ПЧ, например, в разрабатываемом модуле регистр 2001H используется для задания частоты. Для поддержания протокола ModBus ПЧ не требуется никаких дополнительных адаптеров или написания программ, но понадобится небольшая ручная настройка (при помощи пульта управления) для указания параметров связи (таких как: скорость передачи, режим ASCII или RTU) [2].

Реализация протокола со стороны ПЛК

Управляющая программа для ПЛК написана на языке релейной логики (LAD) в среде разработки STEP 7-Micro/Win. Для организации связи по протоколу ModBus необходима дополнительная библиотека команд «ModBus master» (поставляется вместе со средой разработки), позволяющая ПЛК выступать в роли ведущего устройства в сетях ModBus RTU. Библиотека резервирует некоторые ресурсы ПЛК: 284 байт памяти, коммуникационный порт 0, некоторые системные маркеры, и использует систему прерываний, которые необходимо разрешить при написании программы.

В проекте используются две библиотечные команды: первая (MBUS_CTRL) для настройки и активации ModBus коммуникаций, вторая (MBUS_MSG) для передачи сообщения и приёма ответа. Посредством этих команд в программе реализована передача значения частоты между ПЛК и ПЧ.

Метод преобразования частоты

Так как для удобства оператора будет указываться время работы линии, а не частота, возникает проблема преобразования этих величин. Чем большего быстродействия системы мы хотим добиться, тем выше должна быть частота напряжения подаваемого на двигатели [7], следовательно, чем выше скорость вращения, тем меньше время работы линии. Если схематично построить зависимость частоты от времени: $f(t)$ для двигателя, она будет выглядеть так, как показано на рис. 2.

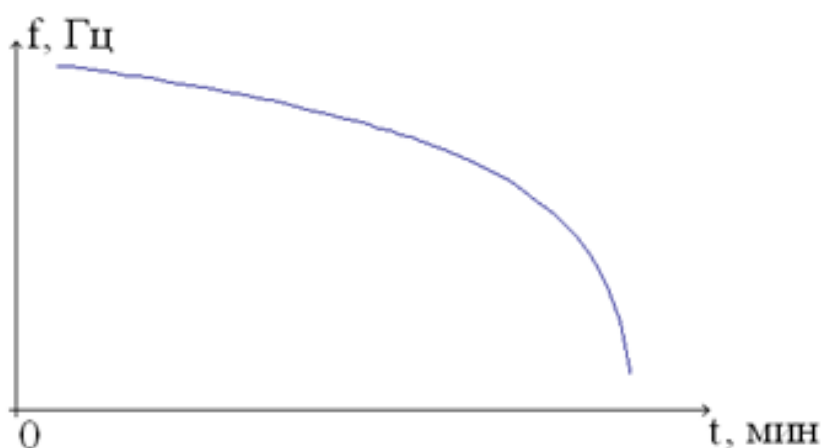


Рис. 2. – Зависимость частоты от времени ($f(t)$)

Если рабочий диапазон времени приходится на линейный участок зависимости $f(t)$, то преобразование времени в частоту можно осуществить по формуле (1):

$$f = tk + f_{\max}, \quad (1)$$

где f – частота напряжения подаваемого на двигатель; t – требуемое время; k – коэффициент; f_{\max} – максимальная частота.

При больших диапазонах времени эта формула будет давать большую погрешность, так как описываемая ей зависимость представляет собой прямую. Для большего соответствия зависимости $f(t)$, можно аппроксимировать её (рис. 3), разделив тем самым на линейные участки, и к каждому участку применять свой коэффициент, в этом случае преобразование времени в частоту будет проводиться по формуле (2):

$$f = \sum_{i=1}^n \Delta t_i k_i + f_{\max}, \quad (2)$$

где f – частота напряжения подаваемого на двигатель; n – количество линейных участков, на которое была разбита зависимость $f(t)$; i – номер линейного участка; Δt_i – отрезок времени, приходящийся на соответствующий участок; k_i – коэффициент участка.

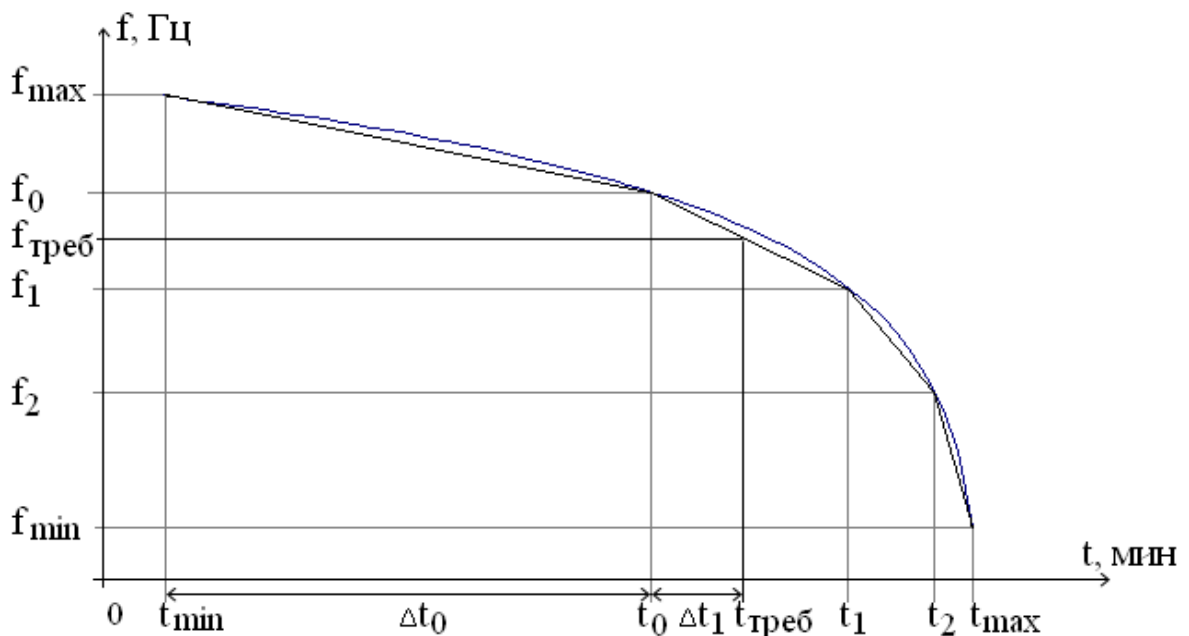


Рис. 3. – Пример аппроксимации зависимости $f(t)$

Для реализации этого подхода необходимо экспериментальным путём измерить характеристики двигателей [8] (точнее узлов системы, приводимых в движение этими двигателями, например: какую частоту нужно подать на двигатель, чтобы время расстойки составляло час, а какую, чтобы полтора часа, фактически оценивается конечный результат, а не скорость вращения конкретного двигателя), затем для каждого построить зависимость, подобную приведённой на рис. 3. По полученным характеристикам можно будет получить все данные, необходимые для вычисления формулы (2).

Для преобразования требуемого времени в частоту на языке релейной логики [9] была разработана подпрограмма TToF (Time To Frequency), реализующая формулу (2). Входными параметрами подпрограммы являются требуемое время и адрес таблицы с параметрами двигателя, а выходным – частота, которую должен генерировать ПЧ для достижения требуемых результатов.

При вводе оператором нового значения времени, управляющая программа по очереди вызывает подпрограмму TToF для каждого двигателя. Значения частот, вычисленные подпрограммой, передаются соответствующим ПЧ. Таким образом обеспечивается согласованность в работе оборудования.

При вызове подпрограмма TToF копирует параметры двигателя в свою область памяти, это реализуется применением косвенной адресации и цикла for- next (рис. 4).

В первом сегменте (Network 1) создаётся косвенный указатель [10] на область памяти, в которую будет происходить копирование. Во втором происходит инициализация цикла (назначение счётчика цикла, выбор начального и конечного значения счётчика). В третьем сегменте происходит копирование одного параметра и приращение косвенных указателей.

Четвёртый сегмент замыкает цикл, он отвечает за возврат ко второму сегменту.

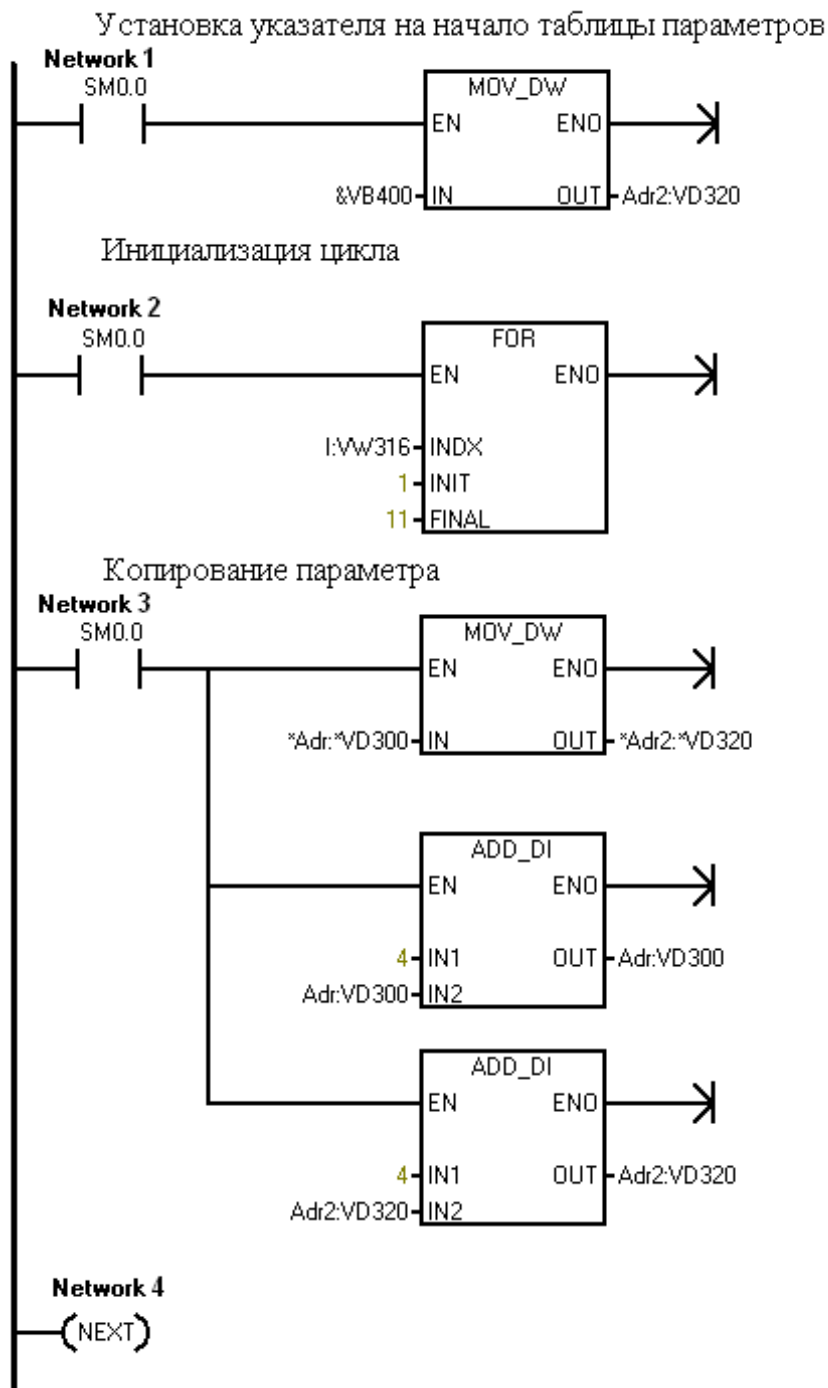


Рис. 4. – Цикл копирования параметров

Главная сложность реализации формулы (2) заключается в вычислении отрезков Δt_i , рассмотрим фрагмент программы (рис. 5), отвечающий за вычисление такого отрезка и умножение его на коэффициент k_i .

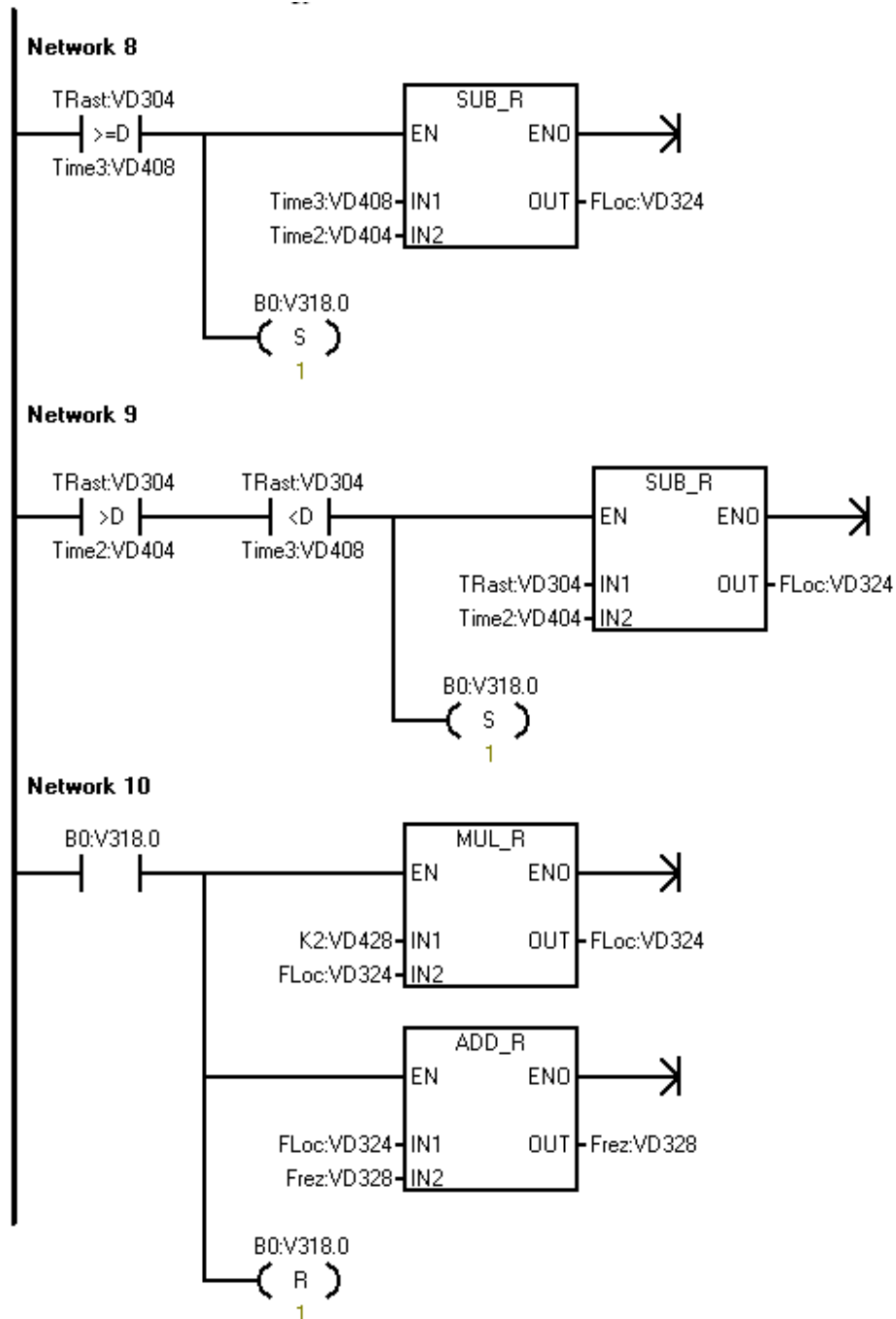


Рис. 5. – Фрагмент программы, отвечающий за нахождение временного отрезка Δt

Предположим, что расположение переменных относительно временной оси соответствует рисунку 6.

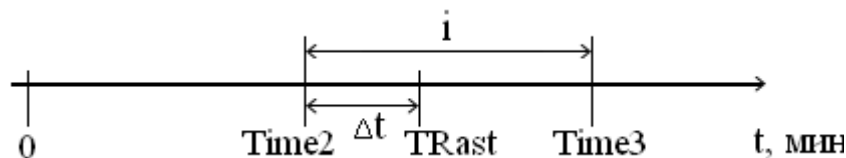


Рис. 6. – Временная ось, поясняющая соотношение переменных в рассматриваемом примере

Первый сегмент (Network 8) отвечает за тот случай, когда время расстойки ($TRast$) больше верхней границы ($Time\ 3$) линейного отрезка i . В этом случае Δt будет равен разнице верхней и нижней ($Time\ 2$) границы отрезка i . Как видно из рисунка 6, время расстойки меньше верхней границы, но больше нижней, за этот случай отвечает второй сегмент программы, здесь Δt будет равно разнице времени расстойки и нижней границы. При срабатывании одного из двух сегментов значение Δt записывается в переменную $FLoc$, кроме того устанавливается бит $B0$, сигнализирующий о наличии Δt на этом линейном отрезке (Δt не будет, если время расстойки меньше нижней границы участка). Третий сегмент включается только при наличии Δt (бит $B0=1$), он отвечает за умножение Δt на коэффициент k_i и реализацию суммирования с предыдущим результатом. Подпрограмма $TToF$ состоит из набора блоков, подобных рассмотренному фрагменту. В последнем сегменте подпрограммы происходит суммирование результата и частоты f_{max} , затем результат округляется и передается основной программе.

Выводы

Рассмотренная реализация согласования работы оборудования не требует обратной связи с двигателями, это уменьшает стоимость системы, но требует довольно сложной настройки, связанной с измерением фактических параметров системы. Со временем, вследствие износа оборудования,

возникнет необходимость корректировать параметры двигателей, указанные в программе; в целом корректировки могут производиться постепенно, без полноценной перенастройки контроллера (если не будет произведена полная замена одного из двигателей).

Литература

1. SIMATIC S7 S7-200 SMART System Manual V2.4. Nuremberg: 2019. 895 p.
2. Sensorless Vector Control Compact Drive VFD-E Series User Manual. Taoyuan: 2016. 435 p.
3. Modicon Modbus Protocol Reference Guide. North Andover: 1996. 121 p.
4. What is Modbus and How does it work? // Schneider Electric Life is On se.com/us/en/faqs/FA168406 (date access: 07.01.2024).
5. Bob Perrin. The Art and Science of RS-485 // CIRCUIT CELLAR ONLINE URL: www.circuitcellar.com (date access: 07.01.2024).
6. Фрасын П.Г., Никитин Н.В., Масанов Д.В., Рыжкова Е.А. Методологические основы работы с протоколом Modbus TCP с примером на высокоуровневом языке программирования Python // Инженерный вестник Дона. 2023. №11. URL: ivdon.ru/uploads/article/pdf/IVD_81__10_Frasyn.pdf_cd9fe09b51.pdf.
7. Усольцев А.А. Частотное управление асинхронными двигателями. Санкт-Петербург: СПбГУ ИТМО, 2006. 94 с.
8. Ларин А.М., Ларина И.И., Манюра В.Г. Определение частотных характеристик асинхронных двигателей из опытов включения на источник однофазного напряжения // Донецкий национальный технический университет URL: masters.donntu.ru/2016/etf/manyura/library/xarakteristik.pdf (дата обращения: 07.01.2024).

9. Programmable controllers – Part 3: Programming languages. International Electrotechnical Commission, International Electrotechnical Commission, 2013. 435 p.

10. Сафаров И.М., Богданова Н.В., Латыпов Т.И. Комплексный критерий оценки эффективности программируемых логических контроллеров // Инженерный вестник Дона, 2023, №9. URL: ivdon.ru/ru/magazine/archive/n9y2023/8706

References

1. SIMATIC S7 S7-200 SMART System Manual V2.4. Nuremberg: 2019. 895 p.

2. Sensorless Vector Control Compact Drive VFD-E Series User Manual. Taoyuan: 2016. 435 p.

3. Modicon Modbus Protocol Reference Guide. North Andover: 1996. 121 p.

4. What is Modbus and How does it work? Schneider Electric Life is On URL: se.com/us/en/faqs/FA168406 (date access: 07.01.2024).

5. Bob Perrin. The Art and Science of RS-485 circuit cellar online. URL: circuitcellar.com (date access: 07.01.2024).

6. Frasn P.G., Nikitin N.V., Masanov D.V., Ryzhkova E.A. Inzhenernyj vestnik Dona. 2023. №11. URL: ivdon.ru/uploads/article/pdf/IVD_81__10_Frasn.pdf_cd9fe09b51.pdf.

7. Usol'tsev A.A. Chastotnoe upravlenie asinkhronnymi dvigatelyami. [Frequency control of asynchronous motors] Sankt-Peterburg: SPbGU ITMO, 2006. 94 p.

8. Larin A.M., Larina I.I., Manyura V.G. Opredelenie chastotnykh kharakteristik asinkhronnykh dvigateley iz opytov vklyucheniya na istochnik odnofaznogo napryazheniya [Determination of frequency characteristics of asynchronous motors from experiments on switching on a single-phase voltage source] Donetskij natsional'nyj tekhnicheskij universitet URL:



masters.donntu.ru/2016/etf/manyura/library/xarakteristik.pdf (date access: 07.01.2024).

9. Programmable controllers – Part 3: Programming languages. International Electrotechnical Commission, International Electrotechnical Commission, 2013. 435 p.

10. Safarov I.M., Bogdanova N.V., Laty`pov T.I. Inzhenernyj vestnik Dona, 2023, №9. URL: ivdon.ru/ru/magazine/archive/n9y2023/8706

Дата поступления: 10.12.2023

Дата публикации: 17.01.2024