

# Задание на разработку для Кандидата

## 1. Требуется разработать консольное приложение, получающее данные от устройств по протоколу Modbus TCP.

- Приложение должно быть реализовано на языке программирования C#, версия платформы .NET 6/8. Должен использоваться Git в качестве системы контроля версий проекта.
- Реализация протокола ModbusTCP должна быть собственная, согласно открытым описаниям стандарта (напр. <https://ipc2u.ru/articles/prostye-resheniya/modbus-tcp/> или аналогичные)
- При реализации не должны быть использованы .nuget-пакеты, сторонние библиотеки или проекты с готовыми реализациями протокола ModbusTCP.
- Требуется реализовать только функции чтения данных; реализация функций записи данных не нужна.
- Приложение должно поддерживать одновременное подключение и циклический опрос произвольного количества устройств с параллельным получением данных от них.

Приложение должно реализовывать следующий функционал:

### 1.1. Давать возможность сконфигурировать:

- Настройку частоты опроса (в мсек.), отдельно для каждого из подключаемых устройств.
- Настройку IP-адреса и порта для подключения к устройствам, отдельно для каждого из подключаемых устройств.
- Сконфигурированные настройки должны запоминаться между сеансами запуска приложения.

### 1.2. Выполнять циклический опрос устройств и логировать полученные данные отдельно для каждого из устройств

Лог-файлы должны содержать следующую информацию:

- Строка в формате дата/время
- Полученные от устройства данные в формате "RawData = {HexString}", где HexString - байтовое представление ответа устройства по сети по протоколу ModbusTCP
- Полученные от устройства данные в формате "Name = {ParamName}, Value = {ParamValue}", где ParamName - строка содержащая текстовое описание переменной, ParamValue - значение данных полученных от устройства и приведенные к требуемому типу

Пример содержимого лог-файла:

```
01.15.2025 16:14:34.94:
RawData = 00-18-00-00-00-06-01-01-27-10-00-01
Name = IsHold, Value = false
RawData = 00-7B-00-00-00-06-01-03-27-1A-00-01
Name = MachineState, Value = 2
RawData = 00-D4-00-00-00-06-01-03-4E-20-00-01
Name = PartCounter1, Value = 7
...

01.15.2025 16:14:35.04:
RawData = 00-18-00-00-00-06-01-01-27-10-00-01
Name = IsHold, Value = false
RawData = 00-7B-00-00-00-06-01-03-27-1A-00-01
Name = MachineState, Value = 2
RawData = 00-D4-00-00-00-06-01-03-4E-20-00-01
Name = PartCounter1, Value = 7
...
```

### 1.3. Иметь возможность регистрации и работы в качестве службы

1.3.1. Для регистрации приложения в качестве службы, необходимо предусмотреть ключ командной строки с помощью которого будет выполняться регистрация

1.3.2. При работе в качестве службы приложение должно поддерживать функции старт/стоп

## 2. Дополнительное программное обеспечение (ПО) для выполнения задания

2.1. Для выполнения задания необходимо скачать ПО "Эмулятор Modbus TCP/RTU устройств". Данное ПО является бесплатным и доступно по ссылке: <http://www.ardsoft.ru/mEmulator.html>

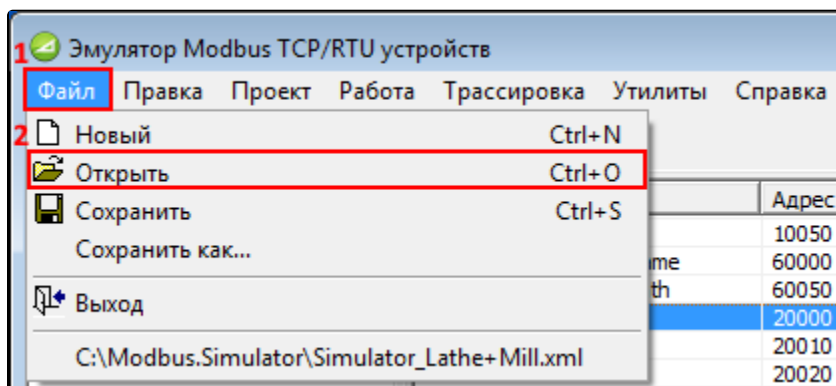
С помощью ПО будет выполняться моделирование (эмуляция) работы реального производственного оборудования по протоколу Modbus TCP.

2.2. Эмуляция оборудования выполняется по предоставленному шаблону доступному для загрузки по ссылке: [Simulator\\_Lathe+Mill.xml](#)

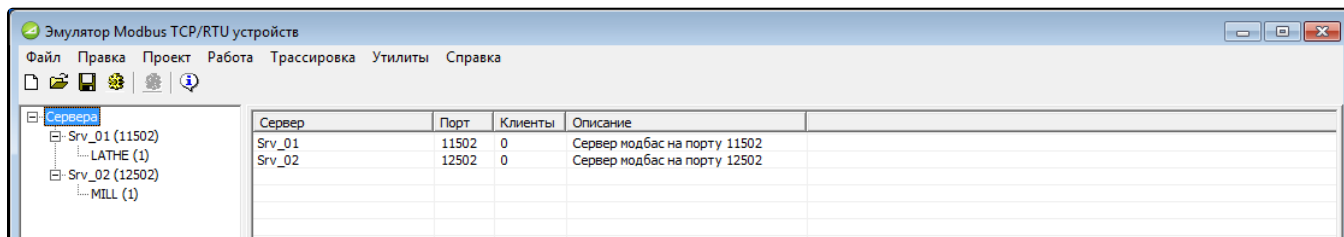
Шаблон содержит набор регистров Modbus для эмуляции работы оборудования следующих типов:

- Токарный станок (Lathe)
- Фрезерный станок (Mill)

Файл шаблона загружается в эмулятор через меню: "Файл" → "Открыть" (см. рис. ниже)

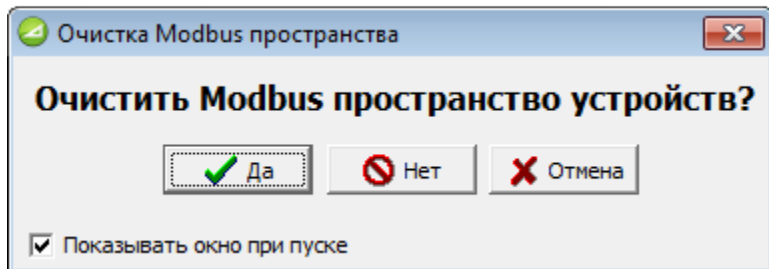


Вид окна эмулятора с загруженным шаблоном:



Запуск эмуляции выполняется при нажатии клавиши "F9", либо пиктограммы шестеренки  расположенной на панели инструментов

На сообщение вида:



Необходимо ответить "Да"

После описанных выше действий приложение перейдет в режим эмуляции данных по заданному в шаблоне набору регистров.

Порты для подключения к эмулятору по протоколу Modbus TCP:

- Токарный станок (Lathe) - 11502/tcp
- Фрезерный станок (Mill) - 12502/tcp

### 3. Получение данных

3.1. Приложение должно обеспечивать получение данных по всем регистрам, содержащимся в шаблоне эмулятора, с последующей записью полученной информации в лог-файлы (см. п.1.2.).

3.2. Поддерживать типы данных заданные для регистров в шаблоне.

3.3. Выполнять преобразование данных (big-endian/little-endian), при необходимости.

3.4. Принимаемые приложением от эмулятора значения регистров должны быть приведены к указанным в таблице:

Наименование параметра	Тип данных	Адрес	Значение параметра	Значение параметра	Примечание
			Токарный станок (Lathe)	Фрезерный станок (Mill)	
IsHold	Coils	10000	False	False	const
IsMDI	Coils	10010	False	False	const
IsMoving	Coils	10020	True	True	const
IsRunning	Coils	10030	False	False	const
IsEStop	Coils	10040	False	False	const
IsProgEnd	Coils	10050	False	False	const
OperationMode	Holding	10000	0..4	0..4	Случайные значения в диапазоне 0 - 4
MachineState	Holding	10010	0..3	0..3	Случайные значения в диапазоне 1 - 3
PartCounter1	Holding	20000	0...∞	0...∞	Значения от 0 до ∞ с приращением 3
PartCounter2	Holding	20010	0...∞	0...∞	Значения от 0 до ∞ с приращением 1
ToolNumber	Holding	20020	8	8	const
FeedRate	Holding	30000	100	100	const
FeedRateCmd	Holding	30010	100	100	const
FeedRateOverride	Holding	20020	60	60	const
RapidOverride	Holding	30030	70	70	const
Spindle1SpeedCmd	Holding	40000	4500	4500	const
Spindle1Override	Holding	40010	1	1	const
Spindle1Speed	Holding	40020	4499,8	4499,8	const
Spindle1Load	Holding	40030	18...99	18...99	Случайные значения в диапазоне 18 - 99
Spindle2SpeedCmd	Holding	40040	-	2300	const
Spindle2Override	Holding	40050	-	1	const
Spindle2Speed	Holding	40060	-	2299,9	const
Spindle2Load	Holding	40070	-	18...99	Случайные значения в диапазоне 18 - 99
Axis1Position	Holding	50000	-11111.1111	-55555.5555	const

Axis2Position	Holding	50010	-22222.2222	-66666.6666	const
Axis3Position	Holding	50020	33333.3333	-	const
Axis1Load	Holding	50100	1...100	1...100	Случайные значения в диапазоне 1 - 100
Axis2Load	Holding	50110	1...100	1...100	Случайные значения в диапазоне 1 - 100
Axis3Load	Holding	50120	1...100	-	Случайные значения в диапазоне 1 - 100
MainProgramName	Holding	60000	Program_11502	Program_12502	const
MainProgramPath	Holding	60050	/bin/var/input/	/bin/var/input/	const

## 4. Предоставление результатов работы

Для проверки работы должно быть предоставлено следующее:

1. Readme-файл в котором описывается конфигурирование, запуск и работа приложения
2. Архив приложения, содержащий проект с исходным кодом, включая папку .git
3. Лог-файлы сформированные приложением в процессе его работы