Акционерное общество "СМС-Автоматизация"
(АО "СМС-Автоматизация")

УТВЕРЖДАЮ

Технический директор

АО "СМС-Автоматизация"

А.А. Сидоров

"___" _____ 2018 г.

Модуль СМС WinCC Modbus

СЕРИЯ СМС-КОМКОН

Руководство по настройке и использованию

На 28 листах

2018

СОДЕРЖАНИЕ

1 Общая информация 3
2 Настройка приложения 5
3 Типы переменных 10
4 Описание работы 12
4.1 Серверный поток 12
4.2 Поток для работы с WinCC 12
5 Форматы команд 14
6 Работа с архивами 17
7 Конфигурирование 22
8 Запуск приложения 25
9 Контакты 26

РОССИЙСКАЯ ФЕДЕРАЦИЯ

СВИДЕТЕЛЬСТВО

о государственной регистрации программы для ЭВМ

№ 2016617967

СМС WinCC Modbus - обмен данными по протоколу Modbus TCP для SIMATIC WinCC/PCS7

Правообладатель: ***Акционерное общество «СМС-Автоматизация» (RU)***

Авторы: ***Не указаны***

Заявка № **2016615220**

Дата поступления **23 мая 2016 г.**

Дата государственной регистрации

в Реестре программ для ЭВМ ***19 июля 2016 г.***

Руководитель Федеральной службы
по интеллектуальной собственности

Г.П. Ивлиев

1 ОБЩАЯ ИНФОРМАЦИЯ

Данная библиотека входит в состав СМС-КомКон.

СМС-КомКон - программное обеспечение поддержки коммуникационных протоколов.

Программа представляет собой библиотеку программных модулей, которые реализуют двусторонний обмен данными между ПК и/или ПЛК.

СМС WinCC Modbus – модуль предназначен для организации коммуникации SCADA-системы в формате протокола Modbus TCP с другими системами.

СМС WinCC Modbus – предназначен для доступа к тегам WinCC по Modbus протоколу.

Simatic WinCC (Windows Control Center) — система HMI, программное обеспечение для создания человеко-машинного интерфейса, составная часть семейства систем автоматизации Simatic, производимых компанией Siemens AG. Работает под управлением операционных систем семейства Microsoft Windows и использует базу данных Microsoft SQL Server.

Modbus — открытый коммуникационный протокол, основанный на архитектуре «клиент-сервер».

Modbus-TCP - это симбиоз стандартного протокола Modbus и протокола Ethernet-TCP/IP, как средства передачи данных.

Система архивации тегов в WinCC строится на основе сервера баз данных Microsoft SQL Server 2005. Фирма Siemens предоставляет простой и удобный доступ к архивам WinCC посредством более современного интерфейса – OLE DB. OLE DB – это стандартизованный интерфейс баз данных, который поддерживается всеми современными языками программирования.

Описание возможностей сервера СМС WinCC Modbus

В СМС WinCC Modbus реализована работа с тегами WinCC, имеющими тип:

- binary tag;
- unsigned 16-bit value;
- floating-point number 32-bit IEEE 754.

В СМС WinCC Modbus реализован механизм передачи архивных данных по Modbus-TCP протоколу.

Схема взаимодействия компонентов системы

Для работы СМС WinCC Modbus необходимы файлы: настройки, лог и конфигурационный файл. Схема взаимодействия на рисунке 1.

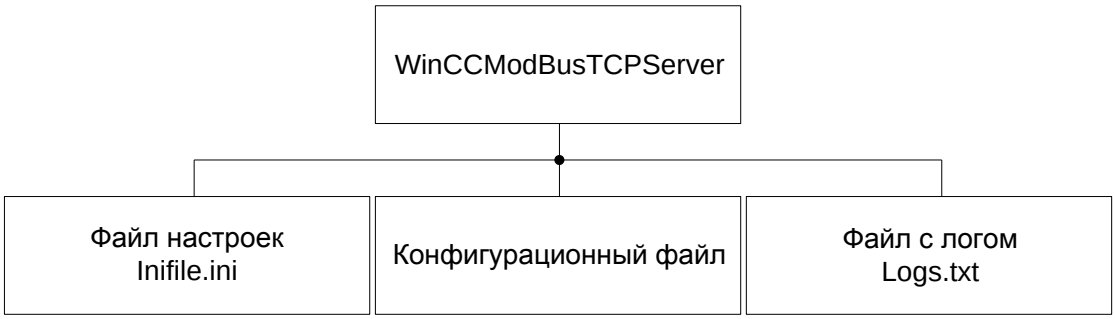

Рисунок 1 – Схема взаимодействия компонентов системы

В файле Inifile.ini содержатся общие настройки системы, которые подробно рассмотрены в разделе 7.

Файл Logs.txt хранит в себе лог работы сервера СМС WinCC Modbus.

Конфигурационный файл содержит таблицы с описанием тегов WinCC, которые описаны в разделе 2.

2 НАСТРОЙКА ПРИЛОЖЕНИЯ

Конфигурационный файл содержит в себе таблицы с описанием тегов. Каждому тегу назначается адрес Modbus регистра. Таблицы должны идти строго друг за другом в том порядке, в котором идет их описание. Слово «table» является ключом, и его использование в качестве имени тега не допускается. В таблице 1 представлено количество регистров под один элемент каждой таблицы.

Таблица 1 – Количество регистров, выделяемых под один элемент

Таблица	Описание	Тип	Регистров, шт.
1	Дискретный	bool	1
2	Механизм	word	1
3	Аналоговый	real	2
4	Время	timestamp	4
5	Управление	-	1
6	Дополнительно	word	1..n

Название столбца «Имя_тега» - служит в алгоритме подтверждением правильности шапки. Изменять его имя нельзя. Для каждой из таблиц важен порядок столбцов. Менять его без доработки алгоритма распознавания нельзя. Конфигурационный файл удобнее создавать в Excel, а затем сохранять его формате txt с разделителями табуляции.

★ После изменения конфигурационного файла необходимо перезапустить программу WinCC2Rakurs.

таблицы 1

Таблица 1 предназначена для описания тегов типа «binary tag».

Имя_тега – имя тега из WinCC, которому ставится в соответствие регистр;

Тип_параметра – признак, не участвующий в алгоритме. Используется для наглядности таблиц. Тип параметра определяется порядковым номером таблицы.

Регистр – номер регистра Modbus, по которому можно обратиться к тегу WinCC. Формат регистра в разделе 3.

Описание – признак, не участвующий в алгоритме. Используется для наглядности таблиц.

Аварийный – если 1, то аварийный;

Предупредительный – если 1, то предупредительный;

Требует записи в WinCC – разрешение на запись, 1 – разрешено, 0 – запрещено.

Таблица 1 – Описание дискретных тегов

table	Дискретный					
Имя_тега	Тип_параметра	Регистр	Аварийный	Предупредительный	Требует записи в WinCC	Описание
b1	Bool	120	0	0	0	Описание
b2	Bool	121	0	0	0	Описание

таблица 2

Предназначена для описания тегов типа «unsigned 16-bit value». Также можно в эту таблицу добавлять теги типа «binary tag».

Признак_основной – в алгоритме не используется, но заполнять обязательно. Может принимать значение 0 или 1.

Ном_связ_регистра – используется для слов состояния и уставок. Для механизмов можно указывать любые регистры. В алгоритме эти номера не используются. Нужны для структуризации конфига (для более простого поиска).

Таблица 2 – Описание механизмов

Table	Механизмы					
Имя_тега	Тип_параметра	Регистр	Признак_основной	Ном_связ_регистра	Требует записи в WinCC	Описание
W1	mehanizm	11111	1	201	0	Описание
W2	mehanizm	22222	1	201	0	Описание

таблица 3

Предназначена для описания тегов типа «floating-point number 32-bit IEEE 754».

Признак_основной – используется для наглядности таблиц;

Признак_состояния уставок – используется для наглядности таблиц;

Регистр_состояния – используется для наглядности таблиц;

Регистр_уставок – используется для наглядности таблиц.

Таблица 3– Описание аналоговых тегов

table	Аналоговый							
Имя_тега	Тип_параметра	Регистр	Признак_основной	Признак_состояния/уставок	Регистр_состояния	Регистр_уставок	Требует записи в WinCC	Описание
r1	real	0	1	0	60	90	0	Описание
r2	real	2	1	0	61	91	0	Описание

таблица 4

Представлен формат описания меток времени. Одна метка времени занимает 4 регистра.

Минуты_секунды – используется для наглядности таблиц;

Дни_часы – используется для наглядности таблиц;

Годы_месяцы – используется для наглядности таблиц;

Миллисекунды – используется для наглядности таблиц;

Имя_тега – не используется в алгоритме, но заполнять обязательно.

Таблица 4 – Описание меток времени

table	Время						
Имя_тега	Тип_параметра	Регистр	Минуты_секунды	Дни_часы	Годы_месяцы	Миллисекунды	Описание
t1	timestamp	48	0	0	0	0	Описание
t2	timestamp	282	0	0	0	0	Описание

таблица 5

Представлено описание команд управления.

Код команды – (см. формат команд);

Регистр – используется в алгоритме, но заполнять обязательно.

Таблица 5 – Описание команд управления ГА

table	Управление ГА			
Имя_тега	Тип_параметра	Регистр	Код команды	Описание
c1	Comand	1001	1	Описание
c2	Comand	1002	2	Описание

таблица 6

Представляет описание дополнительных регистров.

Имя_тега – не используется в алгоритме, но заполнять обязательно;

Разрешение на запись – если 1, то командой записи по модбасу можно записать значение, иначе нельзя.

Таблица 6 – Дополнительные регистры

table	Дополнительные регистры			
Имя_тега	Тип_параметра	Регистр	Разрешение на запись	Описание
U1	word	32000	0	Адрес управления
U2	word	32500	1	Счетчик внешнего буфера
U3	word	32502	0	Адрес внешнего буфера

3 ТИПЫ ПЕРЕМЕННЫХ

При считывании тегов из WinCC есть особенности. Каждый тег имеет определенный тип. В связи с этим в конфигурационном файле были созданы таблицы. Каждая таблица предназначена для описания определенного типа данных.

Чтение булевских параметров

Читать переменные типа «binary tag» можно двумя способами, поместив тег либо в первую таблицу, либо во вторую. При этом значение будет занимать 1 регистр.

Если тег помещен в первую таблицу, то для него будет применяться механизм квитирования. Формат регистра представлен в таблице 1.1. Биты «Авария» и «Предупреждение» выставляются, в зависимости от настроек из конфигурационного файла. Бит «Текущее состояние» всегда совпадает с реальным значением тега. Бит «Квитировано» выставляется в 1 по команде квитирования (раздел 5). Бит «Хранимое состояние» устанавливается в 1 передним фронтом дискретного сигнала и сбрасывается, когда приходит сигнал «Квитирование». Если одновременно выставлены биты «Авария», «Предупреждение», то сигнал маскируется. Это означает, что драйвер принудительно пишет нули в «Текущее состояние», «Хранимое состояние», «Квитировано».

Если тег помещен во вторую таблицу, то он будет интерпретироваться как число типа word. Формат представлен в таблице 1.2.

Таблица 1.1 – Формат регистра в первой таблице для булевского параметра

Бит	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Описание	Резерв	Резерв	Резерв	Резерв	Резерв	Резерв	Резерв	Резерв	Резерв	Резерв	Резерв	Квитировано	**Хранимое состояние**	**Текущее состояние**	Авария	Предупреждение

Таблица 1.2 – Формат регистра во второй таблице для булевского параметра

Бит	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Описание	Резерв	Резерв	Резерв	Резерв	Резерв	Резерв	Резерв	Резерв	Резерв	Резерв	Резерв	Резерв	Резерв	Резерв	Резерв	Значение

Чтение параметров типа word

Для чтения параметров типа «unsigned 16-bit value» необходимо добавить тег во вторую таблицу. Параметр занимает один регистр.

Чтение параметров типа real

Для чтения параметров типа «floating-point number 32-bit IEEE 754» необходимо добавить тег в третью таблицу. Параметр занимает 2 регистра.

Чтение временных меток

Временные метки имеют формат как в таблице 1.3. Занимают 4 регистра.

Таблица 1.3 – Формат временной метки

формат	BCD						HEX	
бит	0	1	2	3	4	5	6	7
Описание	мин	сек	число	час	год	мес	№ миллисекунды	

4 ОПИСАНИЕ РАБОТЫ

Сервер разработан с использованием технологии сокетов. Режим работы многопоточный. Поддерживает подключение нескольких клиентов.

Поддерживает стандартные функции Modbus протокола:

Функция 3 – чтение из нескольких регистров хранения.

Функция 16 – запись в несколько регистров хранения.

Согласование тегов WinCC и карты адресов Modbus находится в конфигурационном файле.

При запуске приложения анализируется конфигурационный файл на наличие ошибок, затем формируется мультиструктура, происходит подключение к WinCC. Запускаются потоки: работы с тегами WinCC, работы с архивами WinCC, серверный поток.

4.1Серверный поток

При старте сервера, все его настройки берутся из Ini_file.ini (описание в разделе 7). При подключении клиента сервер создает поток и циклически слушает порт. Если пришел запрос, то он анализируется и формируется ответное сообщение.

Если пришла команда на чтение, то из карты адресов MB_Map берутся регистры и из них формируется ответ.

Если пришла команда на запись, то:

Если это управление или запись в WinCC, то добавляем его в буфер записи в WinCC;

Если это квитирование, то добавляем его в буфер квитирования.

В конце цикла поток на время SleepThreadTCP (настройка из Ini_file.ini) бездействует. При уменьшении этого времени загрузка процессора увеличивается.

Если в течении таймаута сервера (TimeOutServer) клиент не посылает запросов, то он принудительно отключается сервером, а поток, работающий с этим клиентом, терминируется. При завершении работы приложения, все потоки клиент-сервер терминируются.

4.2 Поток для работы с WinCC

Для взаимодействия реализован поток WinCC. Для чтения/записи значений из WinCC используется драйвер ODK.dll. ODK – открытый пакет разработки для интеграции.

Цикл работы потока WinCC:

Выполнение команды из буфера на запись. За один цикл выполняется одна самая старая команда на запись в WinCC.

Выполнение команд из буфера квитирование. За один цикл выполняются все команды на квитирование. При квитировании анализируются регистры из MB_Map.

Чтение дискретных, механизмов, аналоговых из WinCC и сохранение значений в MB_Map.

Запись временных меток. Поиск их в мультиструктуре и запись в MB_Map.

5 ФОРМАТЫ КОМАНД

Стандартная команда чтения регистров (функция 3)

К регистрам можно обращаться стандартной командой чтения через функцию 3. Пример в таблице 2.4.

- В 5-ом байте длина с 6-го по 11-ый байт = 6.
- В 6-ом байте адрес устройства = 10.
- В 7-ом байте номер функции чтения = 3.
- В 8-ом и 9-ом байте адрес регистра, с которого будем читать = 0.
- В 10-ом и 11-ом байте количество регистров для чтения = 5.

Таблица 2.4 – Формат команды чтения регистров

байт	0	1	2	3	4	5	6	7	8	9	10	11
значение, hex	0	0	0	0	0	6	A	3	0	0	0	5

Стандартная команда записи в регистры (функция 16)

Для записи используется команда 16. Пример в таблице 2.5.

- В 5-ом байте длина с 6-го по 24-ый байт = 19.
- В 6-ом байте адрес устройства = 10.
- В 7-ом байте номер функции записи = 16 = 10h.
- В 8-ом и 9-ом байте адрес регистра, с которого будем писать = 100 = 64h.
- В 10-ом и 11-ом байте количество регистров для записи = 6.
- В 12-ом байте длина с 13-го по 24-ый байт = 12 = 0Ch.
- В 13-24 байтах новые значения, которые запишутся в регистры.

Таблица 2.5 – Формат команды записи регистров

байт	0	1	2	3	4	5	6	7	8	9	10	11	12
значение, hex	0	0	0	0	0	13	A	10	0	64	0	6	C

байт	13	14	15	16	17	18	19	20	21	22	23	24
значение, hex	val	val	val	val	val	val	val	val	val	val	val	val

Для записи в регистры необходимо в конфигурационном файле выставить разрешение на запись в 1.

Формат команд от Rakursa

Формат команд от Rakursa разработан на основании протокола Modbus. Заголовок команды стандартный. После заголовка идут байты, расшифровка которых представлена в таблице 2.6.

Примечания:

1) Нумерация групп с 1, нумерация кодов команд с 1, нумерация устройств внутри группы с 0.

2) ID компьютера/пользователя может при необходимости использоваться для ограничения доступа к некоторым командам.

Таблица 2.6 – Формат команды

8 байт	Слово (HEX)	Слово (HEX)	Слово (HEX)	Слово (HEX)	Слово (HEX)
Заголовок Modbus посылки	Группа	№ в группе	Код команды	Данные 1	ID компьютера/пользователя

Формат команды квитирования

Для квитирования дискретных параметров от Rakursa приходит посылка как в таблице 2.2.

- Байты 0-4 как в стандарте Modbus.
- В 5-ом байте содержится длина Modbus посылки от 6-го до 24-го байта = 19 байт.
- В 6-ом байте адрес устройства. Нам он неважен.
- В 7-ом байте – номер MB функции = 16 → функция записи.
- В 8-ом и 9-ом байте – адрес регистра управления = 32000=7D00h.
- В 10-ом и 11-ом байте – количество регистров из этой MB посылки = 6.
- В 12-ом байте – длина в байтах от 13-го до 24-го байта.
- В 13-ом и 14-ом байте – код команды квитирования = 32=20h.
- В 15-ом и 16-ом байте – номер регистра = 113=71h, который необходимо сквитировать.
- В 17-ом и 18-ом байте – сама команда квитирования = 1.
- Байты с 19-го по 24-ый пока не используются.

Таблица 2.2 – Формат команды квитирования

байт	0	1	2	3	4	5	6	7	8	9	10	11
значение, hex	0	0	0	0	0	13	0B	10	7D	0	0	6

байт	12	13	14	15	16	17	18	19	20	21	22	23	24

значение, hex	C	0	20	0	71	0	1	0	0	0	0	0	0

Расшифровка цветов:

	заголовок
	адрес регистра управления
	количество регистров
	длина в байтах
	код команды квитирования
	номер регистра
	квитировать

Для квитирования всех дискретных сигналов необходимо послать команду, в которой указать номер квитируемого регистра 0.

Нельзя использовать регистр с адресом 0 для дискретных тегов! Иначе, при его квитировании, сквитируются все дискретные параметры.

Формат команды управления от Rakursa

Отличается от квитирования только кодом команды. Для управления ID=0.

Таблица 2.3 – Формат команды управления

байт	0	1	2	3	4	5	6	7	8	9	10	11
значение, hex	0	0	0	0	0	13	0B	10	7D	0	0	6

байт	12	13	14	15	16	17	18	19	20	21	22	23	24
значение, hex	C	0	0	0	1	0	1	0	0	0	0	0	0

Расшифровка цветов:

	заголовок
	адрес регистра управления
	количество регистров
	длина в байтах
	код команды управления(ID)
	код команды из конфига
	значение команды

6 РАБОТА С АРХИВАМИ

Формат БД

WinCC работает с базой данных Microsoft SQL. Архивные теги хранятся в представлении ALARMVIEW (таблица 2.1). Для передачи на верхний уровень используются не все столбцы. Используемые выделены заливкой.

Описание столбцов:

1) State = :
    - 1 – пришло событие;
    - <>1 – ушло событие.
2) DateTime – момент события.
3) Ms – мс.
4) Counter – счетчик событий (первичный ключ).
5) Class = :
    - 4 – Alarm;
    - 7 – Warning.
6) Text1 – название тега.

Таблица 2.1 – Формат представления ALARMVIEW

0	1	2	3	4	5	6	7	8	9
MsgNr	State	DateTime	Ms	Instance	Flags1	PValueUsed	PValue1	PValue2	PValue3
	Пришло/ушло								
10	11	12	13	14	15	16	17	18	19
PValue4	PValue5	PValue6	PValue7	PValue8	PValue9	PValue10	PText1	PText2	PText3
20	21	22	23	24	25	26	27	28	29
PText4	PText5	PText6	PText7	PText8	PText9	PText10	ComputerName	Application	Comment
30	31	32	33	34	35	36	37	38	39
UserName	Counter	TimeDiff	ClassName	TypeName	Class	Type	Text1	Text2	Text3
	Счетчик				A/W		Название тега		
40	41	42	43	44	45	46	47	48	49
Text4	Text5	Text6	Text7	Text8	Text9	Text10	AG_NR	CPU_NR	CrComeFor
50	51	52	53	54	55	56	57	58	59
CrComeBack	CrAckFor	CrAckBack	LocalID	Priority	AP_type	AP_name	AP_par	InfoText	TxtCame
60	61	62	63	64	65	66			
TxtWent	TxtCameNWent	TxtAck	AlarmTag	AckType	Params	Servername			

	Эти столбцы используются для формирования сообщения во внешнем буфере
	Столбцы, используемые для фильтрации сообщений из архива в SQL запросе в IniFile.ini.

Подключение к БД SQL

Для поиска активной БД используется подключение через стандартный драйвер SQL OLEDB Provider. Ниже приведен пример строки для подключения.

```
ADOConnectionSQL.ConnectionString := 'Provider=SQLOLEDB.1;' +
                    'Integrated Security=SSPI;' +
                    'Initial Catalog=master;' +
                    'Data Source=.\WINCC;'+
                    'Auto Translate=True;'+
                    'Packet Size=4096;';
```

Для получения имени активной БД используется запрос:

```
ADOCommandSQL.CommandText     := 'SELECT RTRIM( name),dbid FROM sysdatabases
WHERE (name LIKE "CC%") and (name LIKE "%R")';
```

Подключения к БД с архивами

```
ADOConnectionWINCC.LoginPrompt:= false;
ADOConnectionWINCC.Provider   := 'WinCC OLEDB Provider for Archives';
ADOCommandWINCC.Connection    := ADOConnectionWINCC;
ADOCommandWINCC.CommandType   := cmdUnknown;
ADODataSetWINCC.Connection    := ADOConnectionWINCC;
```

Алгоритм формирования архивных сообщений

Для формирования архивных сообщений имеется внутренний и внешний буфер. Внутренний буфер предназначен для сохранения сообщений после выполнения SQL запроса к представлению ALARMVIEW. Программно размер буфера ничем не ограничен.

Внешний буфер находится в карте Modbus. Его размер задается в настройках IniFile.ini. Он состоит из счетчика сообщений и самих сообщений.

В каждом цикле потока работы с архивами проверяется регистр со счетчиком внешних сообщений. Программа ВУ непрерывно опрашивает счетчик количества сообщений. Как только счетчик количества сообщений становится отличным от нуля, программа ВУ считывает буфер обмена. Указанное количество сообщений заносится в базу. Счетчик сообщений сбрасывается в ноль. Если счетчик равен нулю, то во внешний буфер копируем максимально возможное количество сообщений (не больше размера внешнего буфера) из внутреннего буфера. Если во внутреннем буфере нет сообщений, то делается запрос к базе данных. Если получены сообщения из базы данных, то они добавляются во внутренний буфер. Затем поток приостанавливается на время SleepThreadNormRegim – настройка из IniFile.ini.

Формат сообщений во внешнем буфере

Для обмена с программой верхнего уровня (Rakurs) необходимо формировать буфер сообщений формата, как в таблице 2.2. В действительности формируются сообщения, формат которых представлен в таблице 2.3.

Таблица 2.2 – Формат сообщения во внешнем буфере (как в Rakurse)

Слова и форматы														
1(BCD)		2(BCD)		3(BCD)		4(HEX)	5(HEX)	6(HEX)	7(HEX)		8(REAL,DINT)	10(REAL,DINT)	4(HEX)	4(HEX)
мин	сек	число	час	год	мес	№ миллисекунды	Группа	№ в группе	Код неисправности	Код события	Данные 1	Данные 2	Слово состояния	Номер источника

Таблица 2.3 – Формат сообщения во внешнем буфере (так реализовано)

бит	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
Описание	мин	сек	число	час	год	мес	№ миллисекунды		Группа(32,33,34)		Номер регистра		Код неисправности										Слово состояния			

Зеленым помечены регистры, к которым в первую очередь привязывается Rakurs.

Зависимости таблиц 2.3 от 2.1 в списке:

- 0 – 5 байты в таблице 2.3 формируются из столбца 2 таблицы 2.1;
- 6 – 7 байты в таблице 2.3 из 3 столбца таблицы 2.1;
- 8 – 9 байты в таблице 2.3 из 35 столбца таблицы 2.1;
- 10 – 11 байты в таблице 2.3 из 37 столбца таблицы 2.1;
- 12 – 13 байты в таблице 2.3 из 1 столбца таблицы 2.1;
- 22 – 23 байты в таблице 2.3 из 1 и 35 столбца таблицы 2.1.

7 КОНФИГУРИРОВАНИЕ

IniFile.ini предназначен для конфигурирования СМС WinCC Modbus. Содержит в себе общие настройки сервера (раздел Aplication_settings), настройки приема/передачи по протоколу Modbus (раздел Modbus_settings), настройки для передачи архивных сообщений (раздел Arhive_settings). Пример IniFile.ini:

```
[Aplication_settings]
Config_file=config_2011_12_19_all_defence.txt
Log_file=Logs.txt
sizeLogKb=10000
ExtendedLog=1
Start_auto=1
TCP_IP=127.0.0.1
port=502
TimeOutServer=4000
SleepThreadTCP=10

[Modbus_settings]
FirstByteHigh=0
FirstWordHigh=1
IDKvitirov=32
IDUpravl=0

[Arhive_settings]
EnableArhive=0
sql=ALARMVIEW: SELECT TOP 65000 * FROM AlgViewRus WHERE DateTime > 'DateOld' AND DateTime <= 'DateNow' AND (Text1 LIKE '%MON') AND ((State = 1) OR (State = 2)) ORDER BY DateTime ASC
DateOld=2011-12-23 07:41:04
BufExternalSize=10
UploadIntMin=60
SleepThreadNormRegim=10000
SleepThreadArhive=600
NumLastParam=5688864305890496512
SufficsTag=.I;.Q;
```

Раздел Aplication_settings

Содержит общие настройки драйвера. Для фиксации событий и отладки приложения используется лог. Для него задается максимальный размер в байтах. При превышении максимального размера делается копия лога с расширением *.hys и создается новый пустой файл Logs.txt. Если вновь произойдет переполнение, то старый лог *.hys удалится и вместо него создастся новый. Благодаря такому механизму не происходит утечки памяти.

Config_file=config_2011_12_19_all_defence.txt – название конфиг файла;

Log_file=Logs.txt – название лог файла;

sizeLogKb=10000 – размер лога в байтах;

ExtendedLog=1 – опция работы с логом. 0=сокращенный (системные сообщения), 1=расширенный (расшифровка входящих команд, отображение времен: поиска тега из архива, чтения тегов из WinCC, выполнения команд, записи в WinCC);

Start_auto=1 – 1=автоматический запуск, 0=запуск по кнопке на форме;

TCP_IP=127.0.0.1 – не используется;

port=502 – номер порта, на котором начинает работать драйвер;

TimeOutServer=4000 – таймаут сервера, мс. После этого времени происходит разрыв соединения с клиентом;

SleepThreadTCP=10 – задержка для потока, работающего с сокетом, мс.

Раздел Modbus_settings

Содержит настройки для Modbus.

FirstByteHigh=0 – порядок байт в регистре. 1 – старшим байтом вперед, 0 – младшим байтом вперед;

FirstWordHigh=1 – порядок слов для вещественных чисел. 1 – старшим словом вперед, 0 – младшим словом вперед;

IDKvitirov=32 – идентификатор квитирования. (См. формат команд);

IDUpravl=0 – идентификатор управления. (См. формат команд).

Раздел Arhive_settings

Настройки для работы с архивными данными. Внешний буфер расположен в карте адресов Modbus.

EnableArhive=0 – включить/отключить передачу архивных данных;

sql=ALARMVIEW: SELECT TOP 65000 * FROM AlgViewRus WHERE DateTime > 'DateOld' AND DateTime <= 'DateNow' AND (Text1 LIKE '%MON') AND ((State = 1) OR (State = 2)) ORDER BY DateTime ASC – шаблон sql запроса в WinCC;

DateOld=2011-12-23 07:41:04 – время, с которого будут выбираться записи из архива WinCC; При считывании из внешнего буфера сообщений и обнуления счетчика сообщений, происходит запись в DateOld из DateNow.

BufExternalSize=10 – максимальное количество сообщений во внешнем буфере;

UploadIntMin=60 – интервал времени между 'DateOld' и 'DateNow' при докачке записей из архива, мин;

SleepThreadNormRegim=10000 – время слипа для потока работы с архивом WinCC, если докачка завершена, мс;

SleepThreadArhive=600 – время слипа для потока работы с архивом WinCC, если идет докачка, мс;

NumLastParam=5688864305890496512 – номер последней переданной записи по Modbus. Используется для предотвращения повторной передачи сообщений. При первом запуске приложения счетчик выставить в 0, иначе передача сообщений из архива может не пойти.

SufficsTag=.I;.Q; - суффиксы для тегов. В архиве названия тегов хранятся в столбце «Text1» без суффиксов. Имя тега + суффикс используются для поиска номера регистра из конфигурационного файла. Суффиксы должны разделяться знаком «;».

8 ЗАПУСК ПРИЛОЖЕНИЯ

Подключение к проекту WinCC происходит, как в режиме RunTime, так и в остановленном режиме. Если удалось подключиться к проекту, то связь ОК. Если к серверу подключился хотя бы один ModbusTCP клиент, то связь со SCADA = OK.

В полях буфер архива отображается количество сообщений во внутреннем и внешнем буферах. В поле Data_OLD отображается время из IniFile.ini.

Панель для тестов используется для быстрого поиска тега по номеру регистра. Также на ней отображается количество подключенных клиентов в поле активные соединения. Экранная форма приложения на рисунке 1.

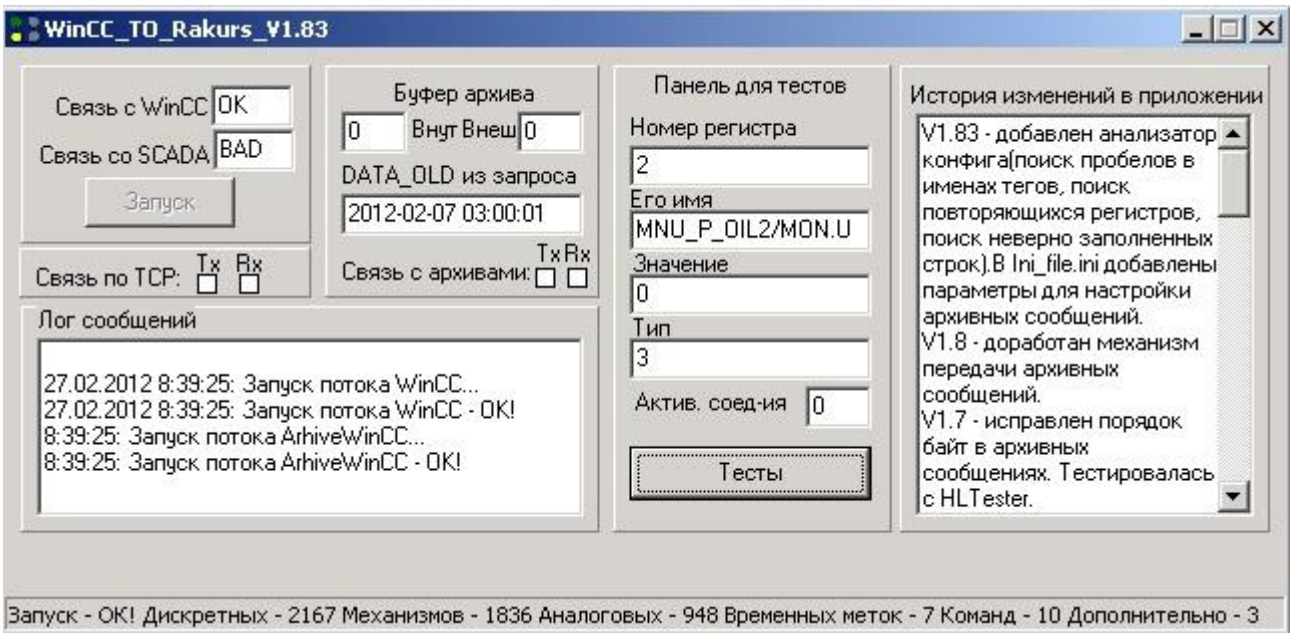

Рисунок 1 – Экранная форма приложения

9 КОНТАКТЫ

Если у Вас возникли вопросы по поводу работы библиотек СМС WinCC Modbus или Вы хотите получить совет по телефону, мы Вам поможем.

По вопросам приобретения библиотек и техническим вопросам обращаться:

Тел./факс: (846) 993-83-83

E-mail: license.plc@sms-a.ru

Web: www.sms-automation.ru

Лист регистрации изменений									
Изм.	**Номера листов (страниц)**				**Всего листов (страниц) в документе**	**Номер документа**	**Входящий номер сопроводительного документа и дата**	**Подпись**	**Дата**
	измененных	**замененных**	**новых**	**аннулированных**					