

Indusoft Web Studio v.6.1.

Краткое руководство пользователя.

scada-support@bbsystems.ru

1. Системные требования

Общие требования: Компьютер с Intel® Pentium IV-совместимым процессором

Разработка: операционная система Windows 2000/NT/XP/Vista или Windows 2003 Server

Исполнение IWS: операционная система Windows 2000/NT/XP/Vista

Исполнение CE View: операционная система Windows CE v3.0, Windows CE.Net.

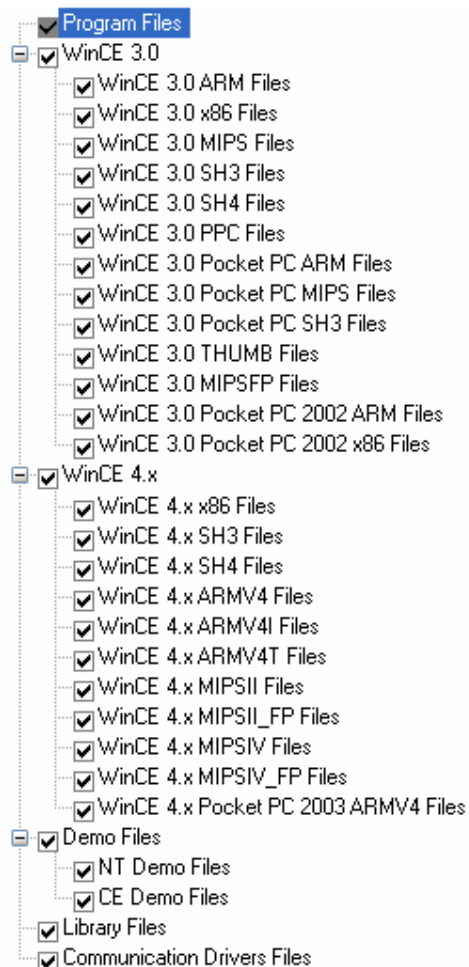
Тип процессора, поддерживаемый IWS (доступные типы перечислены <InduSoft Web Studio Folder>\Redist\<OS Version>\<Processor Type>\)

Тонкий клиент: MS Internet Explorer 6.0 или выше

2. Инсталляция IWS

Разработка и исполнение IWS: IWS можно установить с помощью установочного диска или, скачав файлы для инсталляции на сайте (<http://www.InduSoft.com>).

Запустите установку студии. Следуйте инструкциям указанным в диалоге *InstallShield Wizard*. В диалоге установщика будет предложено выбрать элементы из списка, которые будут установлены на ваш компьютер:



После завершения установки перезагрузите компьютер.

Исполнение CE View: После установки IWS под Windows, файлы необходимые для установки IWS под CE находятся в *<InduSoft Web Studio Folder>\Redist\<OS Version>\<Processor Type>*

Установка:

- 1) При включении WinCE устройства, автоматически запускается Remote Agent диалог. Если он не был запущен автоматически, необходимо скопировать файл CESERVER.EXE из папки *<InduSoft Web Studio Folder>\Redist\<OS Version>\<Processor Type>* на компьютере, с установленной IWS, в папку на WinCE Устройстве, и запустить его.
- 2) Создать соединение между WinCE устройством и компьютером с IWS, используя Remote Agent->Setup (рекомендуется TCP/IP соединение).
- 3) На компьютере с Windows запустить IWS
- 4) На панели управления выбрать **Project->Execution Environment**
- 5) В Диалоге *Execution Environment* выбрать тип соединения с WinCE устройством, затем нажать кнопку Connect.
- 6) Нажать кнопку **Install System Files** для установки файлов на WinCE устройство.

*тип процессора устройства с Windows CE, соответствует допустимым из папки *<InduSoft Web Studio Folder>\Redist\<OS Version>\<Processor Type>*

3. Типы лицензий, ограничения на использование

	CEView Lite	Lite Interface	CEView Standard	Local Interface	CEView Professional	Operator Workstation	Control Room	Advanced Server
Коммуникационные драйверы, одновременно	1	1	3	3	5	5	8	Не ограничено
Кол-во тэгов (точки ввода вывода + переменные)	150	150	1500	1500	4000	4000	64000	512000
элементов в массиве	-	-	256	256	512	512	1024	16348
членов в классе	-	-	32	32	32	32	64	512
Recipe	-	-	+	+	+	+	+	+
Report	-	-	+	+	+	+	+	+
ODBC	-	-	+	+	+	+	+	+
DDE Client / Server	-	-	+	+	+	+	+	+
Group of screens	-	+	+	+	+	+	+	+
Picture (Paste Link)	-	-	+	+	+	+	+	+
Dynamics	-	-	+	+	+	+	+	+

4. Активация лицензий

Разработка, исполнение IWS HardKey: Установить ключ (USB или LTP) и запустить студию. Для активации: **Start -> Programs -> InduSoft Web Studio -> Register** выбрать **HardKey**.

Для изменения лицензии нажмите Check (в появившемся окне будут приведены все параметры текущей лицензии) → Change License, скопируйте код из поля **Site Code** и отправьте его поставщику ПО. В ответ вы получите ключ, который необходимо ввести в поле **Site Key**. После успешной авторизации будет доступна новая лицензия.

Разработка, исполнение SoftKey: Запустите выполнение модуля регистрации: **Start -> Programs -> InduSoft Web Studio -> Register**, выберите **SoftKey** и нажмите Check. Если на компьютере не была установлена лицензия, в появившемся окне в поле Status будет написано **License not found** (иначе параметры текущей лицензии). Нажмите **Change License**, скопируйте код из поля **Site Code** и отправьте его поставщику ПО. В ответ вы получите ключ, который необходимо ввести в поле **Site Key**. После успешной авторизации будет выведено на экран сообщение об успешной инсталляции лицензии.

Исполнение CE View: 2 способа активации лицензии:

- 1) Локально. На WinCE устройстве запустите Remote Agent диалог-> Setup ->Change License. В появившемся окне в поле Status нажмите **Change License**-> скопируйте код из поля **Site Code** и отправьте его поставщику ПО. В ответ вы получите ключ, который необходимо ввести в поле **Site Key**. После успешной авторизации будет выведено на экран сообщение об успешной инсталляции лицензии
- 2) Удаленно. На WinCE устройстве, запустите Remote Agent диалог -> Setup. Создайте соединение между WinCE устройством и компьютером с IWS (рекомендуется TCP/IP соединение). На компьютере с Windows запустить IWS. На панели управления выбрать **Project->Execution Environment**. В Диалоге *Execution Environment* выбрать тип соединения с WinCE устройством, затем нажать кнопку Connect. Выберите вкладку **CE License** -> скопируйте код из поля **Site Code** и отправьте его

поставщику ПО. В ответ вы получите ключ, который необходимо ввести в поле **Site Key**. После успешной авторизации будет выведено на экран сообщение об успешной инсталляции лицензии.

5. Типы переменных

Integer: Численные значения. 32 bit со знаком

String: текстовые значения. Максимальная длина 1024 символа

Real: численные значения с плавающей запятой.

Boolean: численные значения 0 и 1

6. Синтаксис полей переменных. Свойства переменных

Поле переменной – набор свойств, присущих каждой из таблицы переменных.

Синтаксис для просмотра свойств переменной: `<Tag Name>-><Field Name>`

Таблица полей переменных

Поле переменной	Описание	Bool	Int	Real	Str	R / R W
Name	Имя переменной, сформированное в приложении.	Y	Y	Y	Y	R
MemberName	Имя члена класса Синтаксис <code><Class>.<Member>->MemberName</code>	Y	Y	Y	Y	R
Index	Индекс элемента Синтаксис <code><Tag>[<Index>]->Index</code>	Y	Y	Y	Y	R
Description	Описание переменных. Формируется в Tags datasheet.	Y	Y	Y	Y	R
Max	Максимальное значение, которое может принимать переменная в runtime.	N	Y	Y	N	R W
Min	Минимальное значение, которое может принимать переменная в runtime.	N	Y	Y	N	R W
Quality	«Качество» переменной (192=GOOD; 0=BAD). IWS обновляет это поле, каждый раз при присвоении переменной результата выполнения выражения или задач (таких как driver или OPC).	Y	Y	Y	Y	R
Size	Размерность массива. Если переменная не является элементом массива, поле имеет значение 0.	Y	Y	Y	Y	R
TimeStamp	Дата и время последнего изменения значения переменной.	Y	Y	Y	Y	R
Unit	Единицы измерения переменной (не более 9 символов). Например "Kg"	Y	Y	Y	Y	R W
B0 ... B31	Значения (0 или 1) любого из 32 bits (b0, b1, b2, ... b31) переменной типа int.	N	Y	N	N	R W
AlrStatus	Численное значение, связанное с активными, на данный момент, алармами указанной	Y	Y	Y	N	R

	<p>переменной. Каждый бит этого числового результата соответствует определенному статусу:</p> <p>Bit 0: HiHi Alarm активен Bit 1: Hi Alarm активен Bit 2: Lo Alarm активен Bit 3: LoLo Alarm активен Bit 4: Rate Alarm активен Bit 5: Deviation+ Alarm активен Bit 6: Deviation- Alarm активен</p> <p>Пример: Если Tag->AlrStatus возвращает значение 2, это означает "Hi" аларм активен. Если возвращает значение 3, значит одновременной активны алармы "HiHi" и "Hi".</p> <p>Если значение этого поля 0, это означает отсутствие алармов этой переменной.</p> <p>Для переменных типа Boolean, могут быть получены только значения 1 (bit 1), 4 (bit 2) или 16 (bit 4).</p>					
Ack	<p>Это поле может принимать значения 0 и 1:</p> <p>0: Нет алармов, связанных с указанной переменной, нуждающихся в подтверждении; 1: Есть хотя бы один неподтвержденный аларм, связанный с указанной переменной.</p> <p>NOTE: Значение 0 будет только в случае подтверждения всех алармов, связанных с указанной переменной.</p>	Y	Y	Y	N	R
AlrDisable	<p>Это поле может принимать значения 0 и 1:</p> <p>0: Доступны Алармы связаннее с переменной. При достижении аварийного значения, аларм станет активным. 1: Алармы связаннее с переменной не доступны. При достижении аварийного значения, аларм не станет активным.</p>	Y	Y	Y	N	R W
HiHi	0 – HiHi alarm не активен, 1 - HiHi alarm активен	Y	Y	Y	N	R
Hi	0 – Hi alarm не активен, 1 - Hi alarm активен.	Y	Y	Y	N	R
Lo	0 – Lo alarm не активен, 1 - Lo alarm активен	Y	Y	Y	N	R
LoLo	0 – LoLo alarm не активен, 1 - LoLo alarm активен	Y	Y	Y	N	R
Rate	0 – Rate alarm не активен, 1 - RateLo alarm активен	Y	Y	Y	N	R
DevP	0 – DevP alarm не активен, 1 – DevP alarm активен.	N	Y	Y	N	R
DevM	0 – DevM alarm не активен, 1 – DevM alarm	N	Y	Y	N	R

	активен.					
HiHiLimit	Граница возникновения HiHi alarm.	N	Y	Y	N	R W
HiLimit	Граница возникновения Hi alarm.	N	Y	Y	N	R W
LoLimit	Граница возникновения Lo alarm.	N	Y	Y	N	R W
LoLoLimit	Граница возникновения LoLo alarm.	N	Y	Y	N	R W
RateLimit	Граница возникновения Rate alarm.	N	Y	Y	N	R W
DevSetpoint	Уставка для возникновения Deviation alarms.	N	Y	Y	N	R W
DevPLimit	Граница возникновения DevP alarm.	N	Y	Y	N	R W
DevMLimit	Граница возникновения DevM alarm.	N	Y	Y	N	R W

Свойства переменных

Поле переменной	Описание	Bool	Int	Real	Str
Retentive Value	Сохранение значения переменной, в случае остановки приложения.	Y	Y	Y	Y
Retentive Parameters	Сохранение значений полей переменных	Y	Y	Y	Y
Startup Value	Стартовое значение переменной при запуске приложения (если не активно Retentive Value)	Y	Y	Y	Y
Unit	Краткие комментарии к переменной (единицы измерения)	Y	Y	Y	Y
Max, Min	Максимальные, минимальные значения, которые может принимать переменная.	N	Y	Y	N
Dead Band	Величина отклонения от значения переменной, в пределах которой не формируется аларм	N	Y	Y	N
Smoothing	Усреднение изменения значения переменной (при первом считывании tag=50, при следующем Tag=60, тогда новое значение $(50+60)/2$ tag=55)	N	Y	Y	N

Web Data свойства переменных

Переменные могут быть 2-типов: Local и Server.

При обращении тонких клиентов к одной и той же переменной типа Local она принимает различные значения для разных тонких клиентов, независимые друг от друга. Значения переменной типа Server одинаковые для всех тонких клиентов.

7. Массивы и классы

Массив – группа переменных одного типа. Обозначение:

`<ArrayTagName>[ArrayIndex]`. Размерность массива [nx1]. Ограничение на количество элементов массива согласно типу лицензии.

Класс – объединение переменных, имеющих одинаковые свойства. Переменных в пределах одного класса могут быть разных типов. Можно создавать массивы классов. Обозначение: `<ArrayClassName>[<ArrayIndex>].<Member.Name>`.
 Ограничение на количество элементов класса согласно типу лицензии.

8. Переменные типа Inderect

Inderect tag – переменные, обеспечивающие косвенный доступ к другим переменным. Переменная, используемая как Inderect, должна быть типа string. Для косвенного обращения к переменным, необходимо присвоить значение равное названию той переменной, к которой идет обращение, переменной используемой как Inderect, а затем получить доступ, используя символ @.

Пример: Tag=10
 InderectTag="tag"
 Получить значение переменной Tag: value=@InderectTag => value=10
 Записать значение в переменную Tag: @InderectTag=15 => Tag=15

9. Задача Alarm

Формирование аварийных сообщений при достижении переменной, для которой сформирован аларм, предельного значения.

Вид таблицы задачи алармов (Alarm worksheet)

Tag Name	Type	Limit	Message	Priority	Selection
*	HiHi				
*	HiHi				
*	HiHi				

Поля области Header таблицы Alarm worksheet

Поле	Описание
Description	Описание таблицы
Group Name	Название группы алармов. Используется в качестве фильтра алармов в объекте Alarm/Event control при запущенном приложении (runtime).
Email Settings	Настройки, для автоматической отправки Email , содержащих аварийное сообщение. При использовании автоматической отправки email, заранее в проекте необходимо выполнить функцию настройки параметров SMTP протокола (CnfEmail())

Advanced	Дополнительные возможности задачи Alarm (более подробно рассмотрены ниже).
On Line > Display in Alarm Controls	Отображение алармов в объекте Alarm/Event control
On Line > Ack Required	Необходимость подтверждения активных алармов в объекте Alarm/Event control.
On Line > Beep	Звуковое оповещение при существовании активных алармов.
On Line > Send to Printer	Вывод на печать аварийного сообщения при возникновении аларма. Сообщения будут распечатаны на принтере, выбранному по умолчанию. Для вывода на печать с другого принтера, необходимо указать путь к нему в файле <ApplicationName>.APP:
	[AlarmLog] Device=<PrinterPath/PrinterName>
History > Save to Disk	Сохранение аварийных сообщений на диск в виде файлов (ALyymmdd.ALH).
History > Generate Ack Messages	Формирование сообщения о подтверждении активного аларма и сохранение в файл.
History > Generate Norm Messages	Формирование сообщения о нормализации значения переменной, для которой сработал аларм и сохранение в файл.
Colors in Alarm Controls	Цветовая индикация аварийных сообщений в объекте Alarm/Event control.

Поля области Body таблицы Alarm worksheet

Поле	Описание	Синтаксис
Tag Name	Название переменной, для которой формируется аларм. Типы алармов:	Переменная студии
Type	<ul style="list-style-type: none"> • HiHi: Аларм активен при значение переменной >= значению предела. • Hi: Аларм активен при значение переменной >= значению предела. (Tag: Hi limit=3 ->Msg1, HiHi limit=5 ->Msg2 =>если Tag=4 сообщение «Msg1», Tag>5 сообщение «Msg2»)	Combo-box
	<ul style="list-style-type: none"> • Lo: Аларм активен при значение переменной <= значению предела. • LoLo: Аларм активен при значение переменной <= значению предела. • Rate: Аларм активен при изменении значения переменной быстрее заданной скорости (Limit/s, min или hour – выбирается в свойствах переменной) • DevP: Аларм активен при изменении значения 	

- переменной на величину > значение предела + значение уставки (Set Point Deviation).
- DevM: Аларм активен при изменении значения переменной на величину > значение предела - значение уставки (Set Point Deviation)..

При использовании типов алармов Rate, DevP и DevM, необходимо формировать дополнительные условия в настройках переменной (Tag Properties dialog).

Limit	Пределные значения переменных, соответствующие возникновению аларма. Пределы могут изменяться и в процессе работы приложения (runtime), используя поля переменных HiHiLimit, HiLimit, LoLimit, LoLoLimit, Rate, DevP и DevM (пример: TagLevel->HiLimit).	число
Message	Аварийные сообщения, отображаемые в объекте Alarm/Event control, сохраняемые в .alh файл и отправляемые по email.	Текст и/или {Tag} (не более 256 символов)
Priority	Значение приоритета, по которому позыватель может фильтровать и/или сортировать алармы в объекте Alarm/Event contro.	Число (от 0 до 255)
Selection	Текстовое значение, по которому позыватель может фильтровать и/или сортировать алармы в объекте Alarm/Event contro.	Текст (не более 7 символов)

Advanced

Field	Remarks	Syntax
Disable	При значении переменной в этом поле <> 0, алармы данной группы, не формируются	Tag
Total Alarms	Значение переменной, указанной в этом поле, равно числу активных алармов данной группы.	Tag
Total Unack	Значение переменной, указанной в этом поле, равно числу неподтвержденных активных алармов данной группы.	Tag

Remote Ack Trigger	При изменении значения переменной, указанной в данном поле, подтверждаются все активные алармы данной группы.	Tag
Dead Band Time > Activation	Задержка по времени, для формирования сообщения о возникновении аларма.	Tag или число
Dead Band Time > Normalization	Задержка по времени, для формирования сообщения о нормализации аларма.	Tag или число
	Указание времени (time stamp) возникновения (нормализации) аларма. Условия обновления time stamp:	
Dead Band Time > Time Stamp/Value	<ul style="list-style-type: none"> • Activation/Norm (default): Время (с учетом задержки) возникновения (нормализации) аларма. • Last Tag Change: Время изменения значения переменной в течении задержки возникновения (нормализации) аларма. • Start Condition: Время (без учета задержки) возникновения (нормализации) аларма. 	Combo

Продолжительность хранения истории алармов (.alh файлов) указывается в Project > Settings > Options Tab -> Alarm History and Events-> History Life Time (days).

Ограничения для задачи Alarm:

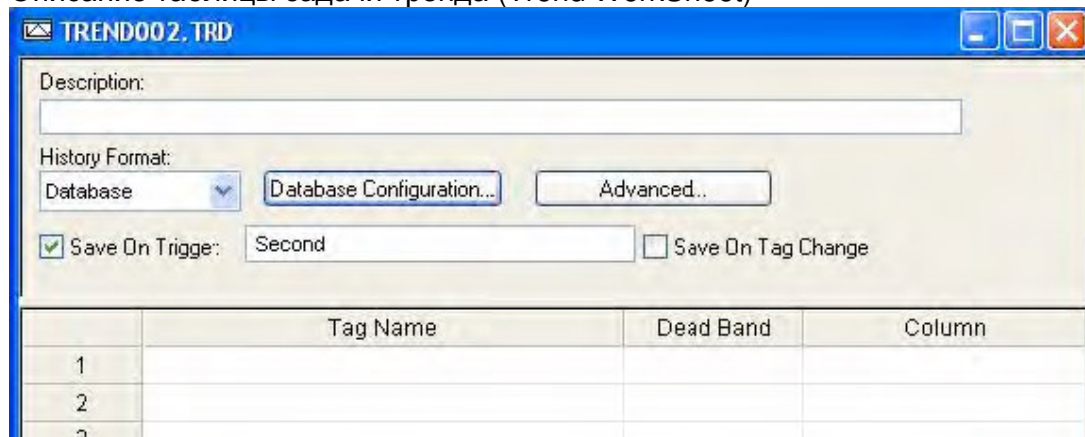
Максимальное количество таблиц (Alarm worksheet) – 255

Максимальное количество строк в таблице (Alarm worksheet) – 32000

10. Задача Trend

Формирование списка точек – значений переменных, в различные моменты времени, в формате .hst файла или базы данных.

Описание таблицы задачи тренда (Trend WorkSheet)



Поля области Header таблицы тренда

History Format:

- Database. Сохранение тренда в виде базы данных (БД). Настройки БД доступны в диалоге Database Cnfiguration
- Proprietary. Сохранение тренда в виде файла формата GGYDDMM.HST

Advanced: дополнительные настройки задачи тренда

Поля настройки Batch файла

- **Start/Stop:** при не нулевом значении переменной в данном поле, IWS будет добавлять новые значения в Batch-файл (в случае выбора Proprietary) или Batch-таблицу (в случае выбора Database)
- **Name:** Путь к Batch-файлу (название Batch-таблицы)
- **Delete:** при изменении значения переменной в данном поле, IWS удалит Batch-файл (поля в Batch-таблице)
- **Existent:** переменная, записанная в данном поле, примет значение 1, если Batch-файл иже существует, в противном случае – 0
- **Description:** (доступно при выборе Database) При не нулевом значении переменной в поле Start/Stop значение, добавляемое в Batch-таблицу, отобразит текст, сформированный в этом поле.
- **Save data even if batch is not running:** если свойство не будет выбрано, данные будут сохраняться только при не нулевом значении переменной в поле Start/Stop

Поля Disk Space Control

- **History Life Time (days):** Количество дней, для хранения файлов истории трендов.
- **Compress After (days):** Количество дней, после которых IWS «сожмет» файлы истории трендов, для освобождения места на диске.

Disable All Data Saving: при не нулевом значении переменной в данном поле, IWS запрещает сохранение истории трендов

Поля области Body таблицы тренда

- **Tag Name:** название переменной, для которой формируется тренд
- **Dead Band:** (доступно при выборе Proprietary) при изменении значения переменной на величину меньшую значению в данном поле, данные не сохраняются в файл тренд.
- **Field:** (доступно при выборе Database) название поля таблицы БД, в которое будет сохранено значение переменной.

Save on Trigger: Значения переменных будут записаны в тренд, при изменении переменной, указанной в данном поле.

Save on Tag Change: При изменении значения переменной из списка, данные будут сохранены в файл тренда.

Ограничения для задачи Trend:

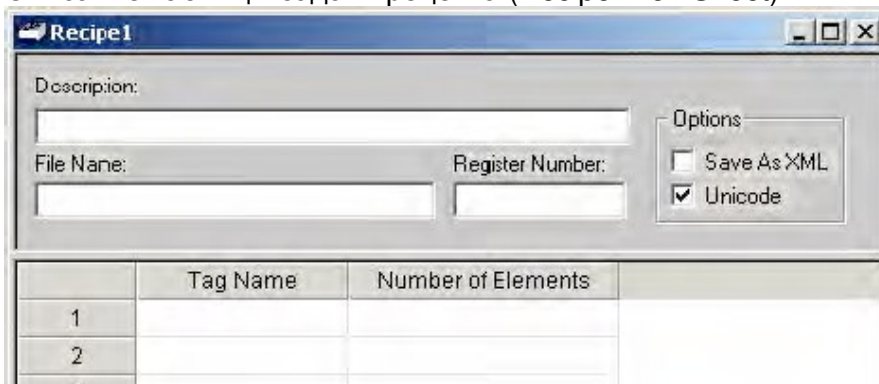
Максимальное количество таблиц (Trend worksheet) – 999

Максимальное количество строк в таблице (Trend worksheet) – 240

11. Задача Recipe

Обмен данными между приложением IWS и файлами в ASCII или DBF формате.

Описание таблицы задачи рецепта (Recipe WorkSheet)



	Tag Name	Number of Elements
1		
2		
3		

File Name: Путь к файлу, с которым будет произведен обмен данными (сохранение или запись). Путь может быть сформирован как переменная {pathTag}. Файл может быть сохранен в формате XML (Save as XML) или Unicod.

Tag Name: Значение переменной, которое будет сохранено в (загружено из) файл

Number of Elements: В случае если нужно сохранить (получить) массив переменных, в данном поле указывается количество элементов массива (начиная с 0-го)

Команды для работы с рецептом:

Recipe("Save:Recipe1") – сохранить рецепт

Recipe("Load:Recipe5") – загрузить рецепт

Ограничения для задачи Recipe:

Максимальное количество таблиц (Recipe worksheet) – 32000

Максимальное количество строк в таблице (Recipe worksheet) – не ограничено

12. Задача Report

Формирование отчетов в формате ASCII или RTF.

Описание таблицы задачи репорта (Report WorkSheet)

Report1

Description:

Output File:

Options

Disk Append

Unicode

Lock Value into the {Tag/Exp} length

Output File: Путь к файлу отчета. Может быть сформирован в виде переменной ({PathTag})

Edit RTF File: Обращение к отчету как RTF файлу. Формирование и редактирование текста отчета

Disk Append: Если поле активно (стоит «галочка»), при вызове команды формирования отчета, новые данные будут добавлены в уже существующий файл. Иначе при вызове команды для формирования отчета, будет создаваться новый файл отчета.

Unicode: формирование отчета в формате Unicode

Lock Value into the {tag/Exp} length: При активном поле, значения переменных, указанные в { }, будут записаны в отчет, имея иметь длину (количество символов) не большую, чем количество символов в { }.

В основном поле формирования отчета, помимо текста, можно использовать выражения с студиными переменными, заключенные в { } ({tag} или {If(tag1>1,1,0)}). Причем в выражениях, заключенных в { } не должно быть пробелов.

Команды для работы с отчетом:

Report("Disk:Report1.rep") – сохранить отчет на диск

Report("Prn:Report2.rep", 0) – распечатать отчет (число – положение страницы: 0 – вертикальное, 1 – горизонтальное)

Report("Pdf:Report1.rep") – сохранить отчет как .pdf

Ограничения для задачи Report:

Максимальное количество таблиц (Report worksheet) – 32000

Максимальное количество строк в таблице (Report worksheet) – не ограничено

13. Задача ODBC

Обмен данными между IWS и любой БД, поддерживающей этот интерфейс.

Для обмена данными, помимо задачи ODBC в студии, необходимо сформировать стандартный интерфейс Windows ODBC.

Описание таблицы задачи ODBC (ODBC WorkSheet)

	Tag Name	Column
1		
2		

Data Source: название интерфейса ODBS, с которым будет производиться обмен данными.

User: имя пользователя для доступа к БД

Password: пароль пользователя для доступа к БД

Table: название таблицы в БД

Condition: поле для выбора или фильтрации данных

Status: состояние соединения (обращения) с БД (ненулевое значение – ошибка)

Transaction: инвертирование значения переменной, указанной в данном поле, после завершения обмена данными.

Select, Next, Insert, Delete, Update Trigger: команды для работы с БД

Tag Name: имя студийной переменной

Column: положение переменной в БД (ячейка)

Задача ODBC запускается автоматически или вручную в настройках проекта (Project → Project Status)

Ограничения для задачи ODBC:

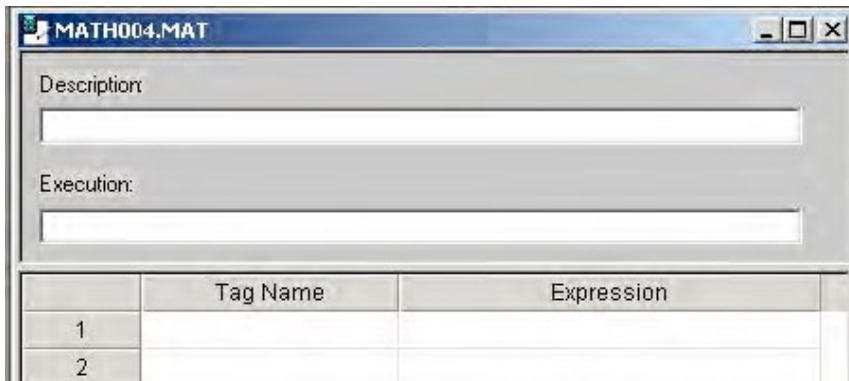
Максимальное количество таблиц (ODBC worksheet) – 999

Максимальное количество строк в таблице (ODBC worksheet) – 4096

14. Задача Math

Формирование скриптов на встроенном языке ISW Built-in Language

Описание таблицы задачи Math (Math WorkSheet)



Execution: при ненулевом значении переменной (выражения или константы), указанной в данном поле, скрипт будет выполняться, до тех пор пока значение переменной не станет =0.

Tag Name: имя студийной переменной

Expression: выражение вычисляемое (выполняемое) в процессе работы скрипта.

Ограничения для задачи Math:

Максимальное количество таблиц (Math worksheet) – 999

Максимальное количество строк в таблице (Math worksheet) – 32000

15. Задача Script

Формирование скриптов на языке VB Script

При использовании студийных переменных в тексте скрипта, перед их названием необходимо поставить символ \$ (\$tag). Внутренние переменные скрипта должны быть объявлены (Dim VBtag), и они могут быть использованы только в пределах данного скрипта.

Запуск скрипта – Execution - при ненулевом значении переменной (выражения или константы), указанной в данном поле, скрипт будет выполняться, до тех пор пока значение переменной не станет =0.

16. Global Script. Graphics Script

Global Script – процедура, вызываемая из любого другого VB скрипта.

Вызов функции – Call FuncName (parameters)

Задание функции – Function FuncName (parameters)

Код программы

End Function

Пример: Передача данных в процедуру и из нее

value=FuncResult(a, b)

Function FuncResult(a, b)

FuncResult=a+ b

End Function


Graphics Script – скрипт, выполняемый при старте (в процессе выполнения и остановки) задачи Viewer-а (графические модули):

Graphics_OnStart(),Graphics_OnWhileRunning(),Graphics_OnEnd ().

17. Задача Scheduler

Выполнение событий (условий) в заданные моменты времени.

Описание таблицы задачи Scheduler (Scheduler WorkSheet)



The screenshot shows a window titled "SCHED002.SCH". Below the title bar is a "Description:" label followed by an empty text input field. Below this is a table with the following columns: Event, Trigger, Time, Date, Tag, Expression, and Disable. The table contains three rows, each starting with an asterisk (*) in the Event column and "Clock" in the Trigger column. Each "Clock" entry has a small blue downward arrow next to it. The other cells in the table are empty.

	Event	Trigger	Time	Date	Tag	Expression	Disable
*	Clock						
*	Clock						
*	Clock						

Event: событие, при котором будет выполнено действие (выражение, указанное в полях Tag и Expression)

- **Clock.** Действие будет выполняться с периодом, указанным в поле Time (00:01:00 – каждую минуту)
- **Calendar.** Действие будет выполнено в указанный в поле Date день и в указанное в поле Time время. При отсутствии в поле Date даты, действие будет выполняться каждый день в указанное в поле Time время.
- **Change.** Действие будет выполнено при изменении значения переменной (триггера) указанной в поле Trigger

Disable: при ненулевом значении переменной в данном поле выполнение действия задачи Scheduler будет запрещено.

Ограничения для задачи Scheduler:

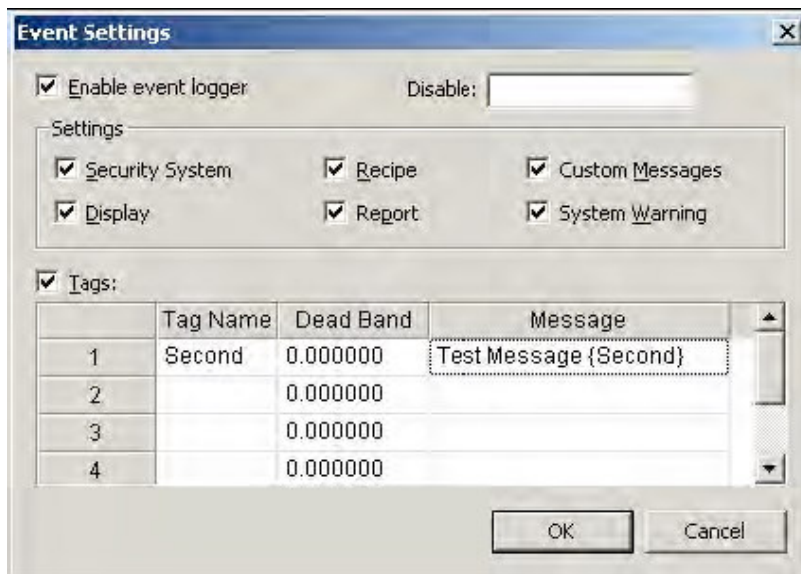
Максимальное количество таблиц (Scheduler worksheet) – 999

Максимальное количество строк в таблице (Scheduler worksheet) – 32000

18. Event

Сохранение событий студии (изменение значения переменной, открытие и закрытие экранов, ошибки и др.) в файл формата evYYMMDD.evт, и отображении их в объекте Alarm/Event control.

Описание таблицы Event



Enable event logger: При активном поле (стоит «галочка») доступно формирование событий и сохранение их в файл

Disable: При ненулевом значении переменной в данном поле, блокируется формирование и сохранение событий.

Security System: сохранение событий системы безопасности IWS (Log on/Log off пользователей, изменение пароля, блокирование пользователей и др.)

Display: сохранение событий открытия/закрытия экранов

Recipe: сохранение событий формирования рецептов

Report: сохранение событий формирования отчетов

Custom Messages: сохранение событий, сформированных с помощью функции *SendEvent(strEvent)*

System Warning: сохранение аварийных событий (ошибка отправки email, деление на 0)

Tag: сохранение событий при изменении значения переменной, указанной в поле **Tag Name**, на величину большую значения указанного в поле **Dead Band**, с текстом, указанным в поле **Message**. Сообщение в поле Message может быть как текстом, так и выражением с использованием переменных, заключенное в { }. Сообщение формируется в ASCII формате. Максимальное количество строк в таблице – 4096.

19. Драйверы

Интерфейс для обмена данными с устройством.

Задача драйвера запускается автоматически или вручную в настройках приложения (Project → Project Status)

Добавление драйвера в проект

- 1) Вкладка Comm->Drivers правой кнопкой мыши Add/Remove Drivers.
- 2) В появившемся окне выберите из списка драйвер к устройству, с которым необходимо связаться -> Select ->Ok

Настройки параметров соединения с устройством

(Вкладка Comm->Drivers правой кнопкой мыши Settings)

Connection Type: Тип соединения с устройством – Direct (последовательное соединение), TCP/IP, Modem, UDP

Serial Port: настройки порта при последовательном соединении. В случае выбора соединения по TCP/IP Или UDP в диалоге будут отображены настройки сети (Network)

Advanced: дополнительные настройки соединения

Main Driver Sheet

Описание таблицы

	Tag Name	Station	I/O Address	Action	Scan	Div	Acd
*				Read+Write	Always		
*				Read+Write	Always		

Disable: При ненулевом значении переменной в данном поле обмен данных с устройством будет запрещен

Read Completed: Значение переменной, указанной в данном поле, будет инвертировано после окончания цикла чтения данных с устройства

Read Status: В переменную, указанную в данном поле, будет возвращен результат чтения данных (ошибка чтения)

Write Completed: Значение переменной, указанной в данном поле, будет инвертировано после окончания цикла записи данных в устройство

Write Status: В переменную, указанную в данном поле, будет возвращен результат записи данных (ошибка записи)

Min/Max: минимальное (максимальное) значение, которое может принимать переменная, указанная в поле Tag Name

Tag Name: имя студийной переменной

Station: станция (адрес устройства в сети)

I/O Address: адрес переменной в устройстве, с которой происходит обмен данными

Action: действия при обмене данными: Read, Write, Read+Write. (Запись производится при изменении значения переменной)

Scan: Условия, при которых выполняется чтение данных (Screen – переменная из списка, существует на открытом, в данный момент экране; Always – постоянное циклическое чтение)

Div/Add: математические операции над полученными данными (tag= val/Div + Add)

Максимальное количество строк в таблице (Main Driver Sheet) = 4096

Driver Sheet

Описание таблицы

The screenshot shows a configuration window titled "MODB1001.DRV". It contains several input fields and checkboxes for configuring the driver's behavior. Below the fields is a table with four columns: Tag Name, Address, Div, and Add. The table has two rows, each starting with an asterisk in the first column.

Tag Name	Address	Div	Add
*			
*			

Increase Priority: приоритет чтения/записи данных данного драйвер-шита в очереди на чтение/запись данных из других драйвер-шитов.

Read Trigger: при изменении значения переменной, указанной в данном поле, будет произведено чтение данных из устройства

Enable Read when Idle: при ненулевом значении переменной, указанной в данном поле, чтение данных будет производиться циклически.

Read Completed: Значение переменной, указанной в данном поле, будет инвертировано после окончания цикла чтения данных с устройства

Read Status: В переменную, указанную в данном поле, будет возвращен результат чтения данных (ошибка чтения)

Write Trigger: при изменении значения переменной, указанной в данном поле, будет произведена запись данных в устройство

Enable Write on Tag Change: при ненулевом значении переменной, указанной в данном поле, запись данных в устройство будет производиться при изменении значения переменной

Write Completed: Значение переменной, указанной в данном поле, будет инвертировано после окончания цикла записи данных в устройство

Write Status: В переменную, указанную в данном поле, будет возвращен результат записи данных (ошибка записи)

Station: станция (адрес устройства в сети)

Header: начальный адрес при обращении к устройстве (индивидуально для каждого типа драйверов)

Min/Max: минимальное (максимальное) значение, которое может принимать переменная, указанная в поле Tag Name

Tag Name: имя студийной переменной

Address: адрес переменной в устройстве, с которой происходит обмен данными

Div/Add: математические операции над полученными данными (tag= val/Div + Add)

Ограничения для драйверов:

Максимальное количество таблиц драйверов (Driver sheet) – 9999

Максимальное количество строк в таблице (Driver sheet) – 4096

NOTE: для соединения с TWCAT на удаленной машине, необходимо положить в <InduSoft Web Studio Folder>\Bin\ следующие файлы:

- \TcAdsDll\AdsDll.dll
- \TcAdsDll\TcAdsDll.dll
- \TcAdsDll\TcAdsTest.exe
- \TcAdsDll\TcAmsRemoteMgr.exe

20. OPC client

Формирование интерфейса между приложением IWS и OPC Server-ом (позволяет IWS соединиться с устройствами, которые выступают как OPC Server), для обмена данными.

Перед использованием OPC Client-а необходимо установить и настроить OPC Server на машине, на которой запущено приложение.

OPC Client запускается автоматически или вручную в настройках приложения (Project → Project Status)

Описание таблицы OPC

opc OPCCL001.OPC

Description: Server Identifier: Disable:

Read Update Rate (ms): Percent Deadband: Status:

Remote Server Name: Read after writing
 Accept Tag Name in the Item column

	Tag Name	Item	Scan	Div	Add
*	<input type="text"/>		Always <input type="text"/>		
*	<input type="text"/>		Always <input type="text"/>		

Server Identifier: Название OPC Server-а, с которым происходит соединение
Disable: при ненулевом значении переменной, указанной в данном поле, обмен данными с OPC Server-ом будет запрещен

Read Update Rate (ms): скорость, с которой будут обновляться данные, получаемые от сервера

Percent DeadBand: при изменении значения переменной на величину (в %) указанную в данном поле, данное значение будет передано на сервер

Status: состояние связи с сервером

Remote Server Name: название удаленного OPC Server-а

Read after writing: флаг поочередного выполнения операций (запись -> чтение)

Accept Tag Name in the Item column: При активном поле, значение тэга, заключенного в { }, в поле Item будет восприниматься как имя переменной OPC сервера.

Tag Name: имя переменной

Item: имя переменной OPC Server-а

Scan: Условия, при которых выполняется обновление данных (Screen – переменная из списка, существует на открытом, в данный момент экране; Always – постоянное циклическое обновление, со скоростью, указанной в поле Read Update Rate)

Div/Add: математические операции над полученными данными (tag= val/Div + Add)

Ограничения для OPC Client:

Максимальное количество таблиц драйверов (OPC Client sheet) – 999

Максимальное количество строк в таблице (OPC Client sheet) – 4096

21. OPC Server

IWS может выступать в роли OPC Server-а. IWS поддерживает модуль OPC Server-а, называемый Studio.Scada.OPC.

Этот модуль автоматически запускается, когда любой OPC Client (локальный или удаленный) пытается соединиться с Studio.Scada.OPC сервером. OPC Client может обмениваться данными со студией (IWS) используя OPC интерфейс. OPC Server можно запустить из студии (автоматически или вручную) используя настройки Project → Project Status.

22. TCP/IP client – server

Модули IWS позволяющие обмен данными между приложениями, используя TCP/IP протокол.

Для использования соединения по TCP/IP необходимо настроить и запустить модули TCP/IP Client и Server.

TCP/IP Server запускается автоматически или вручную в настройках приложения Project → Project Status.

Для настройки TCP/IP Client-а используется конфигурация IWS (в закладке Comm), в которой указывается IP адрес сервера и переменные, которые будут доступны по сети.

Описание таблицы TCP/IP Client

	Tag Name	Remote Tag
*		
*		

Connection Status: Статус соединения с TCP/IP Server-ом

Disable: при ненулевом значении переменной, указанной в данном поле, обмен данными запрещен

Server IP Address: IP адрес TCP/IP Server-а

Send Fields: при активном поле, модули TCP/IP Client и Server будут обмениваться не только значениями переменных, но их значениями их полей (Min, Max, Ack, HiLimit и др)

Tag Name: переменные студии на локальной машине

Remote Tag: переменные студии на удаленной машине

* Если имена переменных на локальной и удаленной машине совпадают, поля Remote Tag можно не заполнять.

Ограничения для TCP/IP Client:

Максимальное количество таблиц драйверов (TCP/IP Client sheet) – 999

Максимальное количество строк в таблице (TCP/IP Client sheet) – 4096

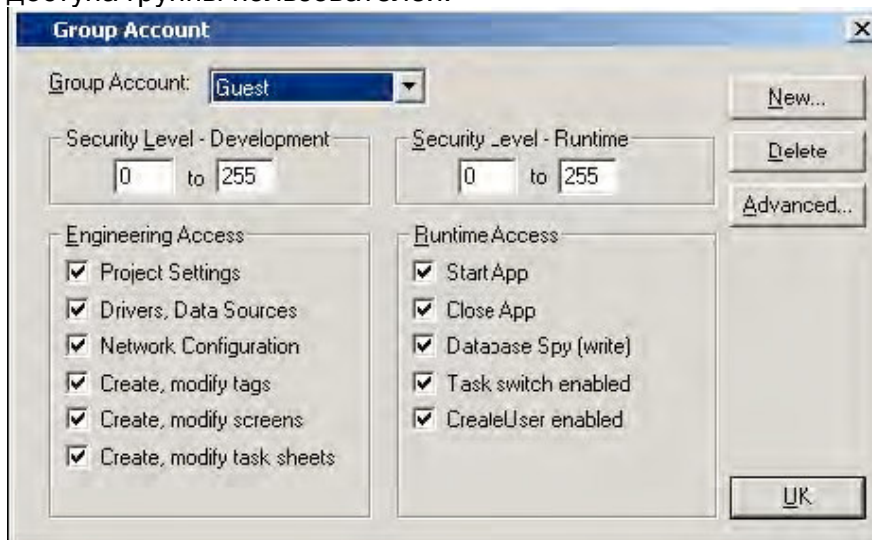
23. Security. Пользователи и группы

Ограничения, накладываемые на выполнение пользователем команд и задач в запущенном проекте (Runtime) и в среде разработки (Development).

Существует 255 уровней доступа – число, указываемое в поле Security в свойствах объекта IWS (команда, ввод/вывод текста, свойства в объекте

Alarm/Event control и др.). Свойство объекта доступно пользователю, если его уровень доступа принадлежит границам доступа пользователя.

Выполнение задач, разрешенных пользователю, выбирается в диалоге настроек доступа группы пользователей.



Group Account: Название группы пользователей

Security Level: границы прав доступа группы пользователей

Engineering (Runtime) Access: Список задач, выполнение которых разрешено группе пользователей при разработке проекта (в работающем приложении)

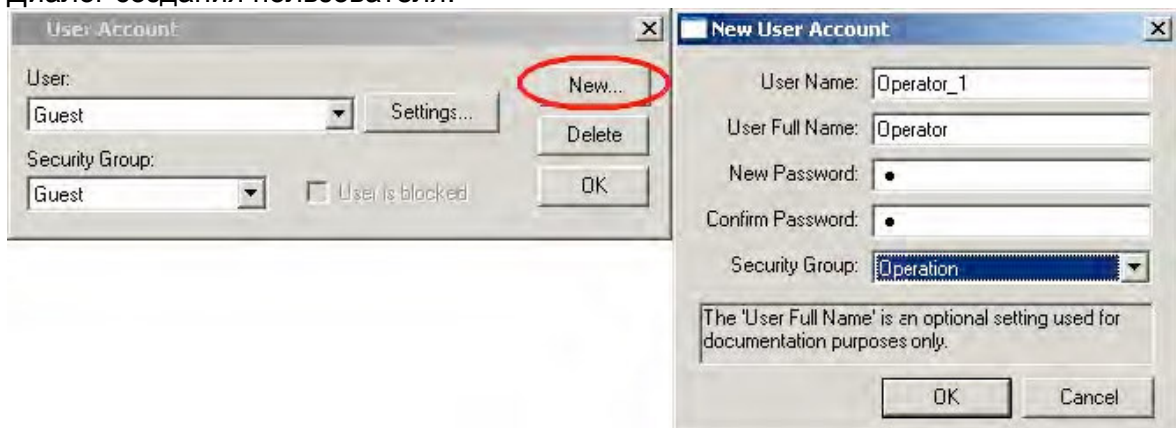
New: добавление новой группы пользователей

Delete: удаление группы пользователей

Advanced: дополнительные настройки группы пользователей

В каждой группе пользователей существует список пользователей, обладающих правами данной группы. Пользователь идентифицируется именем и паролем.

Диалог создания пользователя:



Для создания пользователей в режиме Runtime, используется команда `CreateUser(strUserName, strGroup, strPassw, strUserFullName)`.

Для просмотра всех пользователей (созданных в среде разработки и в Runtime-e) используется программа **ExtUser.exe** (<InduSoft Web Studio Folder>\Bin\).

24. Операции для создания Web ориентированного проекта

Web ориентированного проект – приложение, запущенное на локальной машине, к которому можно обратиться удаленно, используя протокол TCP/IP (Тонким клиентом)

На сервере (машине, на которой запущено приложение) необходимо:

- Создать в проекте .html страницы экранов (File -> Save as Html)
- Запустить TCP/IP Server (автоматически или вручную в настройках приложения Project → Project Status)
- Задать в проекте IP адрес сервера : функция *SetWebConfig(strServerIP)* или в настройках приложения Project->Settings->Web->Data Server IP Address
- Настроить на компьютере порты 80 и порт студии IWS (по умолчанию 1234)
- Запустить Web Server

На клиенте (машина, которая будет удаленно обращаться к серверу) необходимо:

- Наличие MS Internet Explorer 6.0 или выше
- Установить ActiveX компонент IsSymbolControl.ocx
- Разрешить установку ActiveX компонентов

25. Переводчик (Translator)

Формирование таблицы перевода слов и фраз с одного языка на другой, для возможности поддерживать в приложении несколько различных языков.

Таблица формируется в окне *Translation Editor* (Tools -> Translation Editor) и сохраняется в виде .csv файла. Для использования переводчика в приложении, необходимо указать путь к файлу (Project->Settings->Options->Translation File Name) и разрешить перевод, поставив «галочку» в поле Enable Translation.