

НАСТРОЙКА ПЕЧАТИ С КОНТРОЛЛЕРОВ СЕРИИ REGUL RX00

Руководство пользователя

DPA-302.7

Версия документа 1.4

Версия ПО 1.7.1.2

Июнь 2024

История изменений руководства пользователя

Версия системного руководства	Описание изменения
1.4	Добавлена история изменений руководства пользователя. <i>Аннотация:</i> Добавлена информация о типах контроллеров, поддерживающих печать. Внесены небольшие изменения с уточняющей информацией

АННОТАЦИЯ

Настоящий документ содержит сведения о настройке печати документов с промышленных логических контроллеров серии Regul RX00, в частности:

- печать с использованием символов из набора ASCII (AsciiPrinter) реализована на контроллерах R500 CU 00 x51/x61/x71/ x81 (W) и R400 CU 00 071;
- печать html-документов (HtmlPrinter) реализована только на контроллерах R500 CU 00 x71 (W) и R400 CU 00 071.

СОДЕРЖАНИЕ

Введение	5
Поддержка печати	6
Функциональный блок AsciiPrinter	6
Функциональный блок HTMLPrinter	8
Низкоуровневые функции печати	12
Структуры	14
Интерфейсы	15
Применение разметки HTML	17
Поддерживаемые теги	17
Базовые цвета.....	20
Атрибуты.....	22
Свойства CSS	23
Поддерживаемые классы CSS.....	25
Настройка подключения.....	26
Подключение к ПЛК	26
Включение службы печати LPD на контроллере	26
Отключение фильтрации порта печати.....	27
Конфигурация принтеров	27
Печать в файл.....	29

ВВЕДЕНИЕ

Для взаимодействия с контроллером принтер должен поддерживать язык описания PostScript. Практически все плоттеры HP (Hewlett Packard) и инженерные широкоформатные принтеры (Ricoh, Xerox, Kip, Kyocera) имеют опцию с аппаратно - программной поддержкой PostScript. Текстовый файл с программой, написанной на языке PostScript (PS-файл с описанием страниц), преобразуется и выводится на печать с контроллера.

ПОДДЕРЖКА ПЕЧАТИ

Программная поддержка печати реализована в библиотеке PS_Print.

Функциональный блок AsciiPrinter

AsciiPrinter обеспечивает печать документов с использованием символов из набора ASCII моноширинным шрифтом (символы одинаковой ширины). Для печати документа его содержимое построчно добавляется методом *AddTextLine*. При добавлении текста он автоматически конвертируется на язык PostScript и сохраняется в строковом буфере функционального блока. Для печати документа используется метод *Print*. В кодировке *Ascii* представлены только латинские буквы, арифметические цифры и другие служебные символы.

Параметр:

pageFormat: TPageFormat, размер страницы в pt, по умолчанию 595x842 (A4).

pageMargins: TPageMargins, размер полей в миллиметрах, по умолчанию все поля 25 мм.

Возвращаемое значение:

TStringBuffer, строковый буфер, содержащий PostScript.

Пример:

```
FUNCTION Test_Print : BOOL
VAR_INPUT
    Printer      : PsPrint.AsciiPrinter;
END_VAR
VAR
    xInit : BOOL;
    res   : INT;
    Print : BOOL;
    Clear  : BOOL;
END_VAR
...
IF NOT xInit THEN
    Printer.AddTextLine('Hello world!');
    res := Printer.Print('printerName');
    Printer.Clear();
    xInit := TRUE;
END_IF
```

Методы блока

Методы функционального блока приведены в таблице 1

Таблица 1 - Перечень методов функционального блока AsciiPrinter

Метод	Описание
AddTextline	Добавление текстовой строки

Метод	Описание
	<p><u>Параметр:</u> <i>line: STRING</i>, строка</p> <p><u>Пример:</u> <code>Printer.AddTextLine('Hello world!');</code></p>
Clear	<p>Очистка содержимого буфера</p> <p><u>Пример:</u> <code>Printer.Clear();</code></p>
NewPage	<p>Начать новую страницу</p> <p><u>Пример:</u> <code>NewPage();</code></p>
Print	<p>Печать документа на принтере.</p> <p><u>Параметр:</u> <i>printerName: STRING</i>, уникальный идентификатор принтера из файла <code>plc.cfg</code>, секции <code>[Printer]</code>.</p> <p><u>Возвращаемое значение:</u> <i>INT</i>, код ошибки (перечень кода ошибок см. таблицу 2). Возвращает «0» если все ОК, иначе код ошибки.</p> <p><u>Пример:</u> <code>Printer.Print('printername');</code></p>
GetPostscript	<p>Получение документа, описанного на языке PostScript.</p> <p><u>Возвращаемое значение:</u> <i>POINTER TO STRING</i>, указатель на строку, содержащую PostScript</p> <p><u>Пример:</u> <code>PsPrintPostscript (printerName, GetPostscript());</code></p>

Таблица 2 - Перечень кода ошибок

Код ошибки	Описание
0	Отсутствие ошибки
1	Некорректное (INVALID) наименование принтера
2	Некорректный (INVALID) файл
3	Ошибка службы печати
4	Ошибка файловой системы
5	Внутренняя ошибка

Функциональный блок HTMLPrinter

HtmlPrinter предоставляет возможность печати html-документов.

Параметр:

Printer: *STRING*, уникальный идентификатор принтера из файла plc.cfg.

Пример:

```

FUNCTION Test_Main : BOOL
VAR_INPUT
    Printer      : PsPrint.HtmlPrinter;
END_VAR
VAR
    xInit : BOOL;
    res   : INT;
    Print : BOOL;
    Clear : BOOL;
END_VAR
...
IF NOT xInit THEN
    Printer( PrinterName := 'printerName' )
    Printer.Encoding := 'windows-1251';
    Printer.AddText ('<html>');
    Printer.AddText ('<head><meta charset="windows-1251"></head>');
    Printer.AddText ('<body>');
    Printer.AddText ('<h1>Заголовок1</h1>');
    Printer.AddText ('<h2>Заголовок2</h2>');
    Printer.AddText ('<h3>Заголовок3</h3>');
    Printer.AddText ('<h3>Русский алфавит</h3>');
    Printer.AddText ('<p>абвгдеёжзийклмнопрстуфхцчшщъыьэюя</p>');
    Printer.AddText ('<p>АБВГДЕЁЖЗИЙКЛМНОПРСТУФХЦЧШЩЪЫЬЭЮЯ</p>');
    Printer.AddText ('<h3>Греческий алфавит</h3>');
    Printer.AddText ('<p>&alpha; &beta; &gamma; &delta; &epsilon; &zeta; &eta; &theta; &iota; &kappa; &lambda;');
    Printer.AddText ('&mu; &nu; &xi; &omicron; &pi; &rho; &sigma; &sigmaf; &tau; &upsilon; &phi; &chi; &psi;');
    Printer.AddText ('&omega; <sub>p</sub></p><p>&Alpha; &Beta; &Gamma; &Delta; &Epsilon; &Zeta; &Eta;');
    Printer.AddText ('&Theta; &Iota; &Kappa; &Lambda; &Mu; &Nu; &Xi; &Omicron; &Pi; &Rho; &Sigma; &Tau;');
    Printer.AddText ('&Upsilon; &Phi; &Chi; &Psi; &Omega; </p>');
    Printer.AddText ('<p><sub>нижний индекс</sub><sup>верхний индекс</sup></p>');
    Printer.AddText ('</body>');
    Printer.AddText ('</html>');
    res := Printer.Print();
    Printer.Clear();
    xInit := TRUE;
END_IF

```

Методы блока

Методы функционального блока приведены в таблице 3

Таблица 3

Метод	Описание
AddText	Добавление текста документа в буфер. <u>Параметр:</u> <i>line</i> : STRING , строка <u>Пример:</u> <code>Printer.AddText ('<h1>Заголовок1</h1>');</code>
Clear	Очистка содержимого буфера <u>Пример:</u> <code>Printer.Clear();</code>
Print	Печать документа на принтере. <u>Возвращаемое значение:</u> INT , код ошибки (перечень кода ошибок см. в таблице 2) <u>Пример:</u> <code>Printer.Print();</code>

Свойства блока

Свойства блока приведены в таблице 4

Таблица 4 - Перечень свойств функционального блока

Свойство	Тип данных	Значение по умолчанию	Описание
Encoding	STRING	Windows-1251	Кодировка символов входной строки, преобразование текстовых данных из одного формата в другой. Поддерживаются следующие значения кодировки: <ul style="list-style-type: none"> – IBM 850, IBM 866, IBM 874, ISO 8859-1 до 10, ISO 8859-13 до 16, KOI8-R, KOI8-U, Windows-1250 до 1258 <u>Пример:</u> <code>Printer.Encoding := 'windows-1251';</code>
FontFamily	STRING	DejaVuSans	Семейство шрифтов, для оформления текста. Доступные шрифты: <ul style="list-style-type: none"> – DejaVuSans-Bold; – DejaVuSans-BoldOblique; – DejaVuSans-Oblique; – DejaVuSans; – DejaVuSansMono-Bold; – DejaVuSansMono-BoldOblique;

Свойство	Тип данных	Значение по умолчанию	Описание
			<ul style="list-style-type: none"> – DejaVuSansMono-Oblique; – DejaVuSansMono; – DejaVuSerif-Bold; – DejaVuSerif-BoldOblique; – DejaVuSerif-Oblique; – DejaVuSerif; – Vera; – VeraBI; – VeraBd; – VeraIt; – VeraMoBI; – VeraMoBd; – VeraMoIt; – VeraMono; – VeraSe; – VeraSeBd <p><u>Пример:</u> Printer.FontFamily := 'Vera';</p>
FontSize	UINT	10	<p>Размер шрифта.</p> <p>Задается абсолютный размер.</p> <p><u>Пример:</u> Printer.FontSize := 10;</p>
Landscape	BOOL	FALSE	<p>Ориентация страницы (альбомная или портретная/книжная).</p> <p>В зависимости от значения:</p> <ul style="list-style-type: none"> – “FALSE” → книжная; – “TRUE” → альбомная <p><u>Пример:</u> Printer.Landscape := FALSE;</p>
PageMargins	STRING	20,20,20,20	<p>Размер полей в миллиметрах, в формате “l,t,r,b”:</p> <ul style="list-style-type: none"> – l – левое поле; – t – верхнее поле; – r – правое поле; – b – нижнее поле.

Свойство	Тип данных	Значение по умолчанию	Описание
			<p>Допускается указание не всех полей. Например - “25,15”:</p> <ul style="list-style-type: none"> – левое поле → 25 мм; – верхнее поле → 15 мм <p><u>Пример:</u> <code>Printer.PageMargins := '20,20,20,20';</code></p>
PageSize	STRING	A4	<p>Размер страницы (формат).</p> <p>Принимает одно из следующих значений: A4, B5, Letter, Executive, A0, A1, A2, A3, A5, A6, A7, A8, A9, B0, B1, B2, B3, B4, B6, B7, B8, B9, B10, C5E, Comm10E, DLE, Folio, Ladger, Tabloid.</p> <p>При указании неизвестного формата или пустой строки, используется значение по умолчанию.</p> <p><u>Пример:</u> <code>Printer.PageSize := 'A4';</code></p>

Низкоуровневые функции печати

Библиотека PS_Print предоставляет доступ к низкоуровневым функциям печати, реализованным в системном ПО на контроллере (перечень функций см. таблицу 5).

Таблица 5

Функция	Описание		
PsPrintPostscript (PrintPostscript до 1.7.2.0)	<p>Отправляет необработанные данные PostScript на принтер.</p> <p><u>Параметр:</u></p> <p>printerName: STRING, уникальный идентификатор принтера из файла plc.cfg,</p> <p>pPostscriptData: POINTER TO STRING, указатель на нуль-термированную строку, содержащую документ.</p> <p><u>Возвращаемое значение:</u></p> <p>INT, код ошибки (перечень кода ошибок см. в таблице 2)</p> <p><u>Пример:</u></p> <pre>PsPrintPostscript (printerName, GetPostscript());</pre>		
PsPrintPurgePrintQueue (PurgePrintingQueue до 1.7.2.0)	<p>Очистка очереди печати принтера</p> <p><u>Параметр:</u></p> <p>printerName: STRING, уникальный идентификатор принтера из файла plc.cfg</p> <p><u>Возвращаемое значение:</u></p> <p>INT, код ошибки (перечень кода ошибок см. в таблице 2)</p> <p><u>Пример:</u></p> <pre>PsPrintPurgePrintQueue ('printerName');</pre>		
PsPrintHtml (PrintHtml до 1.7.2.0)	Печать документа, составленного на языке HTML.		
	<u>Параметр:</u>		
	Свойство	Тип данных	Описание
	printerName	STRING	Уникальный идентификатор принтера из файла plc.cfg
	pHtmlData	POINTER TO STRING	Указатель на нуль-термированную строку, содержащую документ
	FontFamily	STRING	Семейство шрифтов
FontSize	UINT	Размер шрифта	
PageSize	STRING	Размер страницы (формат). Принимает одно из следующих значений: A4, B5, Letter, Legal, Executive, A0, A1, A2, A3, A5, A6, A7, A8, A9, B0, B1, B10, B2, B3, B4, B6, B7, B8, B9, C5E, Comm10E, DLE, Folio, Ledger, Tabloid.	

Функция	Описание		
			При указании неизвестного размера или пустой строки используется А4
	Свойство	Тип данных	Описание
	Margins	STRING	<p>Размер полей в миллиметрах, в формате “l,t,r,b”:</p> <ul style="list-style-type: none"> – l – левое поле; – t – верхнее поле; – r – правое поле; – b – нижнее поле. <p>По умолчанию все поля равны 20 мм. Допускается указание не всех полей. Например - “25,15”:</p> <ul style="list-style-type: none"> – левое поле → 25 мм; – верхнее поле → 15 мм
	Landscape	BOOL	<p>Ориентация страницы (альбомная или портретная/книжная).</p> <p>В зависимости от значения:</p> <ul style="list-style-type: none"> – “FALSE” → книжная; – “TRUE” → альбомная
	Encoding	STRING	<p>Кодировка символов входной строки, поддерживаются следующие кодировки: IBM 850, IBM 866, IBM 874, ISO 8859-1 до 10, ISO 8859-13 до 16, KOI8-R, KOI8-U, Windows-1250 до 1258</p>
	<p><u>Возвращаемое значение:</u></p> <p>INT, код ошибки (перечень кода ошибок см. в таблице 2)</p> <p><u>Пример:</u></p> <pre>Print := PsPrintHtml(mPrinterName, mBuffer.GetStringPtr(), mFontFamily, mFontSize, mPageSize, mPageMargins, mLandscape, mEncoding);</pre>		

Структуры

В библиотеки PS_Print реализованы следующие структуры.

TPageFormat

Структура с информацией о формате страницы.

Содержит переменные, указанные в таблице 6.

Таблица 6 – Переменные TPageFormat

Переменная	Тип	Описание
PageWidthPt	UINT	Ширина
PageHeightPt	UINT	Высота

TPageMargins

Структура с информацией о размере полей в миллиметрах.

Содержит переменные, указанные в таблице 7.

Таблица 7 – Переменные TPageMargins

Переменная	Тип	Описание
MarginTopMm	UINT	Отступ сверху, мм (по умолчанию 25 мм)
MarginLeftMm	UINT	Отступ слева, мм (по умолчанию 25 мм)
MarginRightMm	UINT	Отступ справа, мм (по умолчанию 25 мм)
MarginBottomMm	UINT	Отступ снизу, мм (по умолчанию 25 мм)

Интерфейсы

Библиотека PS_Print предоставляет доступ к интерфейсам печати, описанным в таблице 8.

Таблица 8

Интерфейс	Описание		
HtmlPrinter	Интерфейс функционального блока принтера, печатающего формат html. <u>Пример:</u> <code>res := Printer.Print();</code> Метод Print Печать документа на принтере. <u>Возвращаемое значение:</u> <i>INT</i> , код ошибки (перечень кода ошибок см. в таблице 2) <u>Пример:</u> <code>Printer.Print();</code> Свойства:		
	Свойство	Тип данных	Описание
	Landscape	BOOL	Ориентация страницы (альбомная или портретная/книжная). В зависимости от значения: – “FALSE” → книжная; – “TRUE” → альбомная <u>Пример:</u> <code>Printer.Landscape := FALSE;</code>
	PageMargins	STRING	Размер полей в миллиметрах, в формате “l,t,r,b”: – l – левое поле; – t – верхнее поле; – r – правое поле; – b – нижнее поле. Допускается указание не всех полей. Например - “25,15”: – левое поле → 25 мм; – верхнее поле → 15 мм <u>Пример:</u> <code>Printer.PageMargins := '20,20,20,20';</code>
PageSize	STRING	Размер страницы (формат). Принимает одно из следующих значений: A4, B5, Letter, Legal, Executive, A0, A1, A2, A3, A5, A6, A7, A8, A9,	

Интерфейс	Описание	
		<p>B0, B1, B10, B2, B3, B4, B6, B7, B8, B9, C5E, Comm10E, DLE, Folio, Ledger, Tabloid.</p> <p>При указании неизвестного размера или пустой строки используется A4</p> <p><u>Пример:</u> Printer.PageSize := 'A4';</p>
IPostscriptPrinter	<p>Интерфейс функционального блока принтера, печатающего формат postscript.</p> <p><u>Пример:</u> res := Printer.Print('printerName');</p> <p>Метод Print</p> <p>Печать документа на принтере.</p> <p><u>Параметр:</u> printerName: STRING, уникальный идентификатор принтера из файла plc.cfg</p> <p><u>Возвращаемое значение:</u> INT, код ошибки (перечень кода ошибок см. в таблице 2)</p> <p><u>Пример:</u> Printer.Print('printerName');</p>	

ПРИМЕНЕНИЕ РАЗМЕТКИ HTML

HTML – это язык, используемый для разметки текстового документа. Текст размечается тегами, которые представляют собой ключевые слова языка. Теги заключают в угловые скобки "<" и ">" и чаще парами, для выделения начала и конца куска текста. Данные пометки служат для определения параметров текста (стиль, формат и т.д.), которые будут использованы при выводе на печать.

Поддерживаемые теги

Все поддерживаемые теги HTML перечислены в таблице 9.

Таблица 9 – Поддерживаемые теги HTML

Тег	Описание	Комментарий
a	Создает ссылку (якорь) на другую страницу своего или чужого сайта	Поддерживает атрибуты «href» и «name»
address	Адрес	
b	Жирный текст	
big	Увеличенный шрифт	
blockquote	Параграф с отступом	
body	Основное содержимое старницы (тело документа)	Поддерживает атрибут «bgcolor» в формате #RRGGBB
br	Разрыв строки	
center	Центрированный параграф	
cite	Встроенная цитата	Схожее поведение тэгу«i»
code	Заключение программного кода любого языка	Схожее поведение тэгу«tt»
dd	Определение данных	
dfn	Определение	Схожее поведение тэгу«i»
div	Формирование документа	Поддерживает стандартные атрибуты блока
dl	Определение списка	Поддерживает стандартные атрибуты блока
dt	Определение составляющих	Поддерживает стандартные атрибуты блока
em	Выделение	Схожее поведение тэгу«i»
font	Размер шрифта, стиль и/или цвет	Поддерживает следующие атрибуты: – size;

Тег	Описание	Комментарий
		<ul style="list-style-type: none"> – face; – color (название цвета из предустановленных цветов или цвет в формате #RRGGBB)
h1	Заголовок уровня 1	Поддерживает стандартные атрибуты блока
h2	Заголовок уровня 2	Поддерживает стандартные атрибуты блока
h3	Заголовок уровня 3	Поддерживает стандартные атрибуты блока
h4	Заголовок уровня 4	Поддерживает стандартные атрибуты блока
h5	Заголовок уровня 5	Поддерживает стандартные атрибуты блока
h6	Заголовок уровня 6	Поддерживает стандартные атрибуты блока
head	Заголовок документа	
hr	Горизонтальная линия	Поддерживает атрибут «width», который может быть указан как абсолютное или относительное (%) значение
html	Документ HTML	
i	Наклонный	
kbd	Текст вводимый пользователем	
meta	Мета информация	
li	Элемент списка	
nobr	Неразрывный текст	
ol	Упорядоченный список	Поддерживает стандартные атрибуты списка
p	Параграф	По умолчанию выравнивается по левому краю. Поддерживает стандартные атрибуты блока
pre	Преформатированный текст	
s	Зачеркнутый	
samp	Пример кода	Схожее поведение тэгу «tt»
small	Маленький шрифт	
span	Группированные элементы	
strong	Жирный	Схожее поведение тэгу «b»
sub	Нижний индекс	
sup	Верхний индекс	
table	Таблица	Поддерживает следующие атрибуты:

Тег	Описание	Комментарий
		<ul style="list-style-type: none"> – border; – bgcolor (<u>название цвета</u> или определение в формате #RRGGBB); – cellspacing; – cellpadding; – width (абсолютная или относительная); – height
tbody	Тело таблицы	Ничего не делает
td	Ячейка данных таблицы	Поддерживает стандартные <u>атрибуты ячейки таблицы</u>
tfoot	Сноска таблицы	Ничего не делает
th	Ячейка заголовка таблицы	Поддерживает стандартные <u>атрибуты ячейки таблицы</u>
thead	Заголовок таблицы	Если тег «thead» указан, то он используется при печати таблицы, которая состоит из множества страниц
tr	Строка таблицы	Поддерживает атрибут «bgcolor», который может содержать <u>название цвета</u> или определение цвета в формате #RRGGBB
tt	Моноширинный шрифт	
u	Подчеркнутый	
ul	Неупорядоченный список	Поддерживает стандартные <u>атрибуты списка</u>
var	Переменная	Схожее поведение тэгу «i»

Базовые цвета

Существует 20 базовых предустановленных названий цветов включая черный и белый, главные и второстепенные цвета, темные версии этих цветов и три оттенка серого (Рисунок 1).



Рисунок 1 – Предустановленные цвета

В HTML предпочтительней всего задавать цвета RGB-кодом. Любой цвет задается сочетанием трех основных цветов (красный, зеленый и синий). При записи, после знака решетки, прописываются три двухзначных шестнадцатеричных числа. Каждое число предусматривает 256 градаций яркости и отвечает за насыщенность основного цвета. Первое число указывает на интенсивность красного цвета (Red), второе – зеленого (Green), третье – синего (Blue).

Последовательность чисел должна строго выдерживаться, пример красного цвета:

```
color: #ff0000;
```

Описание цветов представлено в таблице 10.

Таблица 10 – Описание базовых цветов

Название	Код	Описание
black	#000000	Чёрный
white	#ffffff	Белый
darkGray	#808080	Тёмно-серый
gray	#a0a0a4	Серый
lightGray	#c0c0c0	Светло-серый
red	#ff0000	Красный
green	#00ff00	Зелёный
blue	#0000ff	Синий
cyan	#00ffff	Бирюзовый
magenta	#ff00ff	Пурпурный

Название	Код	Описание
yellow	#ffff00	Жёлтый
darkRed	#800000	Тёмно-красный
darkGreen	#008000	Тёмно-зелёный
darkBlue	#000080	Тёмно-синий
darkCyan	#008080	Тёмно-бирюзовый
darkMagenta	#800080	Тёмно-пурпурный
darkYellow	#808000	Тёмно-жёлтый
transparent	0, 0, 0, 0	Прозрачный цвет

Атрибуты

Блока

Тег `<div>` задает блок, который объединяет в себе другие теги, сам по себе ничего не делает, но в сочетании с атрибутами позволяет обратиться к группе объединенных тегов. Тег `<dl>` входит в тройку элементов `<dl>`, `<dt>`, `<dd>`, предназначенных для создания списка определений. Каждый такой список начинается с контейнера `<dl>`, куда входит тег `<dt>` создающий термин, и тег `<dd>`, задающий определение этого термина. Теги `<h1>...<h6>` - создают заголовки внутри страницы с 1 по 6 уровень, а тег `<p>` - создает абзац. Теги поддерживают следующие атрибуты:

- `align` (выравнивание содержимого блока по краю - `left`, `right`, `center`, `justify`);
- `dir` (направление отображения текста : `ltr` (слева направо), `rtl` (справа налево)).

Списка

Списки предоставляют возможность предоставить отдельные предложения в удобном виде, которые начинаются с маркера или цифры. Списки определяются тегами - `...` (создание маркированного списка) и `...` (создание нумерованного списка), поддерживающие следующие атрибуты:

- `type`:
 - `disc` (закрашенные круги);
 - `circle` (круги);
 - `square` (закрашенные квадраты);
- 1. `decimal` (арабские числа);
- a. `lower-alpha` (строчные латинские буквы);
- A. `upper-alpha` (заглавные латинские буквы).

Таблицы

Таблица определяется тегами - `<td>...</td>` (создание обычной ячейки таблицы) и `<th>...</th>` (создание особой ячейки-заголовка таблицы), поддерживающие следующие атрибуты:

- `width` (ширина колонки - абсолютная, относительная или без значения);
- `bgcolor` (цвет фона - название цвета или определение в формате `#RRGGBB`);
- `colspan` (число ячеек по горизонтали, объединение колонок таблицы);
- `rowspan` (число ячеек по вертикали, объединение рядов таблицы);
- `align` (выравнивание содержимого ячейки строки по горизонтали - `left`, `right`, `center`, `justify`);
- `valign` (вертикальное выравнивание содержимого ячейки строки - `top`, `middle`, `bottom`).

Свойства CSS

Все поддерживаемые свойства перечислены в таблице 11.

Таблица 11 – Поддерживаемые свойства CSS

Свойство	Значения	Описание
background-color	<color>	Цвет фона для элемента
background-image	<uri>	Фоновое изображение для элементов
color	<color>	Цвет текста
font-family	<family name>	Название стиля шрифта
font-size	[small medium large x-large xx-large] <size>pt <size>px	Размер шрифта относительно шрифта документа, указанный в точках или пикселях
font-style	[normal italic oblique]	
font-weight	[normal bold 100 200 300 400 500 600 700 800 900]	Указывает ширность шрифта, используемого для текста, где «normal» и «bold» преобразовываются в вес 50 и 75 соответственно
text-decoration	none [underline overline line-through]	Дополнительные текстовые эффекты
font	[[<'font-style'> <'font-weight'>] <'font-size'> <'font-family'>]	Стенографическое свойство шрифта
text-indent	<length>px	Отступ первой линии текста в пикселях
white-space	normal pre nowrap pre-wrap	Объявляет, как обрабатываются пробелы в HTML
margin-top	<length>px	Верхняя граница параграфа в пикселях
margin-bottom	<length>px	Нижняя граница параграфа в пикселях
margin-left	<length>px	Левая граница параграфа в пикселях
margin-right	<length>px	Правая граница параграфа в пикселях
padding-top	<length>px	Верхний отступ ячейки таблицы в пикселях
padding-bottom	<length>px	Нижний отступ ячейки таблицы в пикселях
padding-left	<length>px	Левый отступ ячейки таблицы в пикселях
padding-right	<length>px	Правый отступ ячейки таблицы в пикселях
padding	<length>px	Стенограмма всех настроек, всех свойств отступов

Свойство	Значения	Описание
vertical-align	baseline sub super middle top bottom	Вертикальное выравнивание текста. Для вертикального выравнивания в ячейках текстовой таблицы применяются только «middle», «top» и «bottom»
border-color	<color>	Цвет границ для текстовых таблиц
border-style	none dotted dashed dot-dash dot-dot-dash solid double groove ridge inset outset	Стиль границ для текстовых таблиц
background	[<'background-color'> <'background-image'>]	Стенограмма свойств фона
page-break-before	[auto always]	Делает возможным разорвать страницу перед параграфом/таблицей
page-break-after	[auto always]	Делает возможным разорвать страницу после параграфа/таблицы
float	[left right none]	Указывает, где изображение или текст будет определен в другом элементе. <u>Свойство «float», поддерживается только для таблиц и изображений</u>
text-transform	[uppercase lowercase]	Выбирает трансформацию, которая будет исполнена над текстом до его отображения
font-variant	small-caps	Выполняет преобразование текста - переход к маленьким буквам перед выводом на экран
word-spacing	<width>px	Указывает дополнительный интервал между каждым словом

Поддерживаемые классы CSS

Поддерживаются все классы селекторов CSS 2.1, кроме псевдоклассов. Класс – это всего лишь один из атрибутов HTML-тегов. В CSS можно задавать стили только для элементов с определённым классом. Для этого используется селектор по классу, который записывается, например:

```
.important { color: red; }– выберет все теги с классом "important".
```

Псевдоклассы определяют динамическое состояние элементов, которое изменяется с помощью действия пользователя, а также положение в документе. Например, такие псевдоклассы, как:

- :first-child (задает стилевое оформление для первого дочернего элемента);
- :visited (применяется к ссылкам, уже посещенным пользователем, и задает для них стилевое оформление);
- :hover (определяет стиль элемента при наведении на него курсора мыши, но при этом элемент еще не активирован).

НАСТРОЙКА ПОДКЛЮЧЕНИЯ

Подключить принтер к контроллеру можно как по сети, так и локально через USB. Принтеры, подключаемые через TCP/IP должны поддерживать службу печати LPD, при этом используется протокол «построчной печати». LPD – демон принтера, выступает в качестве главного сервера для координации и управления очередями. Название доступного принтера и его параметры построчно вносятся в файл **plc.cfg** (см. описание в разделе «Конфигурация принтеров»).

Вначале произведите предварительные настройки:


- включите службу печати LPD на контроллере (файл **runtime.cfg**);
- отключите фильтрацию порта печати (файл **pf.conf**);
- настройте конфигурацию принтера (файл **plc.cfg**).

Подключение к ПЛК

Для начала работы необходимо подключиться к контроллеру через сканер сети (см. «ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ ASTRA.IDE. Руководство пользователя», раздел «Подключение контроллера к сети»).

Включение службы печати LPD на контроллере

По умолчанию служба печати отключена, поэтому необходимо запустить службу, выполнив следующие действия:

- в Astra.IDE на главной вкладке параметров устройства перейдите на вкладку **Файлы**. В области **Исполнение** нажмите кнопку  (**Обновить**). В окне отобразится дерево файлов, имеющих на контроллере. Найдите папку **etc** (Рисунок 2)

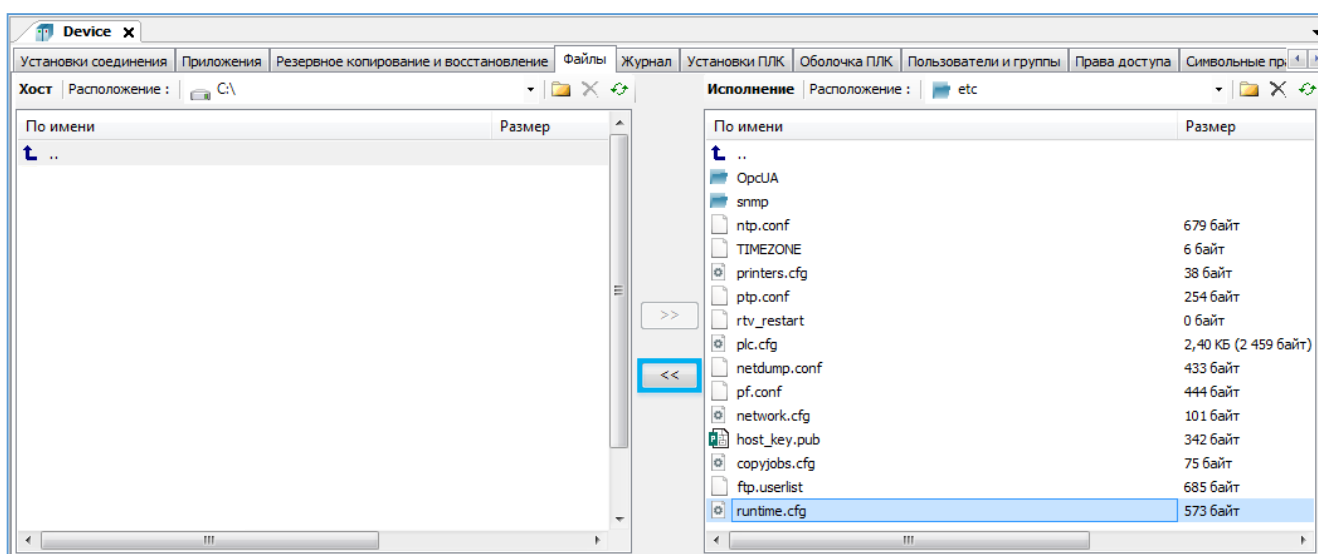




Рисунок 2 - Обмен файлами между ПК и контроллером

- в папке **etc** найдите файл **runtime.cfg**. Кнопкой  скопируйте этот файл с контроллера на ПК (из **Исполнение** в **Хост**);
- откройте на ПК файл **runtime.cfg**. В секции [PlcServices] добавьте строку со значением *EnableLPD=1*;
- в Astra.IDE на вкладке **Файлы** кнопкой  скопируйте измененный файл с ПК на контроллер (из **Хост** в **Исполнение**).

Отключение фильтрации порта печати

Запросы по сети на постановку задания в очередь, удаление из очереди и выдачу информации принимаются через порт 515 (tcp). В целях безопасности сетевой порт закрыт. Для отключения фильтрации порта, зайдите в папку **etc**. Произведите аналогичные действия, которые описаны в разделе «Включение службы печати LPD на контроллере», только уже с файлом **pf.conf**. Для исключения чтения строки о фильтрации порта, добавьте символ (#) в начале строки и получите в результате следующую запись:

```
#block return in quick proto tcp from any to any port 515
```

Конфигурация принтеров

Параметры принтера содержатся в файле **plc.cfg** в папке **etc**. Принтер будет получать задания на печать только в том случае, если он описан в этом файле. Зайдите в папку **etc**. Произведите аналогичные действия, которые описаны в разделе «Включение службы печати LPD на контроллере» только уже с файлом **plc.cfg**, где необходимо прописать параметры принтера. Можно подключить только один принтер. В файле **plc.cfg** в секции [Printer] указываются построчно параметры принтера (Рисунок 3). По завершению перезагрузите контроллер (с помощью сброса питания вкл./выкл. или командой *reboot* на вкладке **Оболочка ПЛК**).

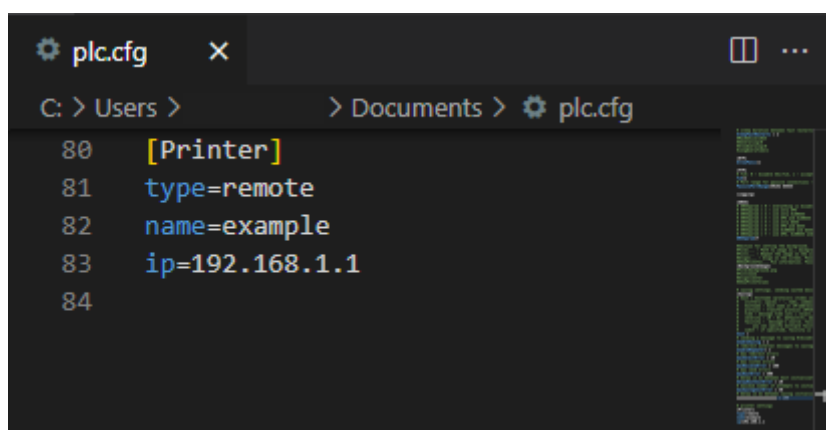


Рисунок 3 – Параметры принтера

Параметры задаются в следующем формате: название параметра, знак равенства (=) и строковое значение.

Добавление сетевого принтера

Для организации доступа, осуществляемого к принтеру по сети, необходимо указать в параметре *type* значение *remote* и добавить ip-адрес сетевого принтера. Пример:

```
[Printer]
type=remote
name=hp304
ip=192.168.1.1
```

Добавление USB принтера

Для организации доступа, осуществляемого к локальному принтеру через USB, необходимо указать в параметре *type* значение *usb*. Пример:

```
[Printer]
type=usb
name=hp304
```

Печать в файл

Можно задать печать в файл при отладке. Для этого необходимо добавить принтер **fileprinter**, который будет отправлять печать не на принтер, а в файл с расширением ***.ps**, который будет создан в папке **logs**.

Примечание: Принтер **fileprinter** в файл конфигурации **plc.cfg** добавлять не нужно!

Пример блока подпрограммы для печати в файл представлен ниже:

```
PROGRAM POU_MAIN
VAR

    xRun: BOOL := TRUE;
    Printer: PsPrint.HtmlPrinter;
    res: INT;

END_VAR

-----

IF xRun THEN
    xRun := FALSE;

    Printer( PrinterName := 'fileprinter' );
    Printer.Encoding:= 'windows-1251';
    Printer.AddText ('<html>');
    Printer.AddText ('<head><meta charset="windows-1251"></head>');
    Printer.AddText ('<body>');
    Printer.AddText ('<h1>Заголовок1</h1>');
    Printer.AddText ('<p>абвгдеёжзийклмнопрстуфхцчшщъыьэюя</p>');
    Printer.AddText ('<p>АБВГДЕЁЖЗИЙКЛМНОПРСТУФХЦЧШЩЪЫЬЭЮЯ</p>');
    Printer.AddText ('<h3>Греческий алфавит</h3>');
    Printer.AddText ('<p>&alpha; &beta; &gamma; &delta; &epsilon; &zeta; &eta; &theta; &iota; &kappa; &lambda;');
    Printer.AddText ('&mu; &nu; &xi; &omicron; &pi; &rho; &sigma; &sigmaf; &tau; &upsilon; &phi; &chi; &psi;');
    Printer.AddText ('&omega; <sub>p</sub></p><p>&Alpha; &Beta; &Gamma; &Delta; &Epsilon; &Zeta; &Eta;');
    Printer.AddText ('&Theta; &Iota; &Kappa; &Lambda; &Mu; &Nu; &Xi; &Omicron; &Pi; &Rho; &Sigma; &Tau;');
    Printer.AddText ('&Upsilon; &Phi; &Chi; &Psi; &Omega; </p>');
    Printer.AddText ('<p><sub>нижний индекс</sub><sup>верхний индекс</sup></p>');
    Printer.AddText ('</body>');
    Printer.AddText ('</html>');
    res := Printer.Print();
    SendTestResult('Checking HtmlPrinter', res = 0 );
    Printer.Clear();

END_IF
```