

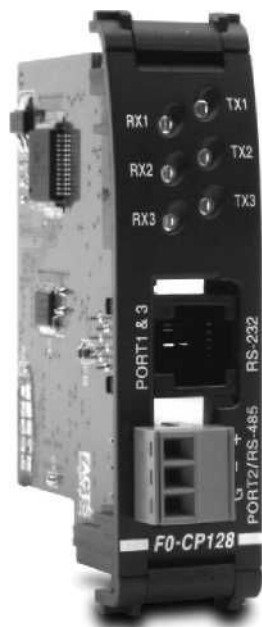
**AutomationDirect.com™**

DirectLogic 05 и 06

Трёхпортовый сопроцессорный модуль,  
программируемый на языке Бейсик

F0-CP128

Руководство пользователя



Код для заказа руководства на английском языке: F0-CP-M



---

## ТОРГОВЫЕ МАРКИ

---

<sup>TM</sup>**Automationdirect.com** – торговая марка компании **Automationdirect.com**

<sup>TM</sup>CoProcessor – торговая марка компании FACTS Engineering, Inc.

---

## АВТОРСКОЕ ПРАВО

---

Авторские права с 2004 года компании FACTS Engineering Inc., 8049 Photonics Dr., New Port Richey, Флорида, 34655. Все права защищены.

Дата последнего издания: сентябрь 2005  
Дата текущего издания: сентябрь 2005



---

## ПРЕДУПРЕЖДЕНИЕ

---

Благодарим за то, что вы купили оборудование для автоматизации компании FACTS Engineering. Мы хотим, чтобы ваше новое оборудование нашей компании работало надежно. Каждый, кто устанавливает или использует наше оборудование, должен прочитать эту инструкцию (и всю относящуюся к этому оборудованию документацию) перед установкой или использованием.

Чтобы минимизировать риск возможных проблем, вы должны следовать всем местным и национальным инструкциям, которые определяют правила установки и использования вашего оборудования. Эти правила различны в разных регионах и обычно меняются со временем. Вы должны сами определить, каким правилам надо следовать, проверить условия установки и функционирования оборудования на соответствие с последними редакциями правил.

Как минимум, вы должны следовать указаниям соответствующих разделов в национальных правилах пожарной безопасности, правилах электробезопасности и указаниям Национальной Ассоциации Производителей Электрооборудования (National Electrical Manufacturer's Association - NEMA). Возможно, существуют государственные организации, которые Вам смогут помочь определить, каким правилам и стандартам следует руководствоваться для безопасной установки и использования оборудования.

*Отказ следовать соответствующим правилам и стандартам может привести к повреждению оборудования или нанести серьезные увечья персоналу. Мы не гарантируем, что продукция, описанная в этой публикации, подходит для вашего конкретного применения, и мы не несем ответственности за ваши проекты, установки и работы.*

Если у вас будут вопросы по установке и работе данного устройства или если вам понадобится дополнительная информация, звоните нам по телефону 1-800-783-3225 или обращайтесь в Российское представительство ООО «ПЛКСистемы»:

тел. +7 495 105-77-98

e-mail: [info@plcsystems.ru](mailto:info@plcsystems.ru)

сайт: [www.plcsystems.ru](http://www.plcsystems.ru)

Эта публикация использует информацию, которая была доступна на момент выхода издания. Несмотря на все наши скрупулёзные усилия, информация, содержащаяся в этом руководстве, не имеет целью осветить все детали или вариации аппаратных средств и программного обеспечения, а также все возможные случаи, связанные с их установкой, работой и обслуживанием. Эта публикация может содержать описания возможностей, которые будут недоступны в определенных версиях наших изделий. Компания FACTS Engineering не берёт на себя обязательств сообщать владельцам этого документа об изменениях, внесённых в него впоследствии. Компания FACTS Engineering оставляет за собой право делать изменения в своей аппаратной и программной продукции в любое время без предупреждения. Компания FACTS Engineering не даёт никаких гарантий, выраженных, подразумеваемых или предписанных законом в отношении своей продукции, и не несёт ответственности за точность, полноту, достаточность или полезность информации, содержащейся в данном руководстве. Никакие обращения о гарантиях товарной пригодности продукции для конкретного использования компанией не принимаются.



## ИЗМЕНЕНИЯ РУКОВОДСТВА

Не забудьте указать название руководства и номер его редакции, которые приведены ниже, если вы будете связываться с отделом технической поддержки по поводу этого руководства.

Название руководства: Direct Logic 05 and 06 Triple Port BASIC CoProcessors User's Manual (Руководство пользователя трёхпортовым сопроцессорным модулем контроллеров Direct Logic 05 и 06 для программирования на языке бейсик)

Код руководства: F0-CP-M

Редакция / Дата	Изменённые страницы	Описание изменений
Предварительная 2/2005		Первый проект
Предварительная 8/2005	Первая страница руководства, 10,12,13,15,16,17,19, 21,22,23,34	Добавлен черно-белый рисунок Различные исправления и удалённые ссылки к COMMAND@2
Предварительная 9/2005	11,14,15,18,19	Различные исправления
Первая 9/2005	17,19	Первая редакция - исправления





# СОДЕРЖАНИЕ

<b>ГЛАВА 1 : ВВЕДЕНИЕ</b> .....	<b>9</b>
<b>СИНХРОНИЗАЦИЯ ЦПУ</b> .....	<b>9</b>
<b>COMMAND@</b> .....	<b>10</b>
<b>ГЛАВА 2 : ОПЕРАТОРЫ СОПРОЦЕССОРА</b> .....	<b>11</b>
<b>BMOVE</b> .....	<b>11</b>
Восьмеричная система счисления и типы данных для операндов оператора BMOVE.....	12
Операнды BMOVE для DL05 .....	12
Операнды BMOVE для DL06 .....	12
<b>DPORT</b> .....	<b>14</b>
Числовые переменные с плавающей запятой в формате IEEE .....	14
<b>S06_</b> .....	<b>18</b>
Числовые переменные с плавающей запятой в формате IEEE .....	18
Восьмеричное исчисление и типы данных для операндов S06_ .....	19
Операнды S06_ для ЦПУ DL05 .....	19
Операнды S06_ для ЦПУ DL06 .....	19
<b>ГЛАВА 3 : ТРЁХПОРТОВЫЙ СОПРОЦЕССОРНЫЙ МОДУЛЬ F0-CP128</b> .....	<b>20</b>
<b>ОБЩИЕ ХАРАКТЕРИСТИКИ F0-CP128</b> .....	<b>20</b>
<b>F0-CP128 ОПИСАНИЕ</b> .....	<b>21</b>
<b>ОПИСАНИЕ И РАСПОЛОЖЕНИЕ ПЕРЕМЫЧЕК В F0-CP128</b> .....	<b>22</b>
CLR ALL .....	22
<b>КОНТАКТЫ ПОРТА F0-CP128</b> .....	<b>23</b>
<b>КОНТАКТЫ ПОРТА РАЗВЕТВИТЕЛЯ (СПЛИТТЕРА)</b> .....	<b>24</b>
<b>ПРИЛОЖЕНИЕ А : ПРИСТУПАЯ К РАБОТЕ С СОПРОЦЕССОРНЫМ МОДУЛЕМ</b> .....	<b>26</b>
<b>ПРЕДВАРИТЕЛЬНЫЕ ДЕЙСТВИЯ С ИСПОЛЬЗОВАНИЕМ АВМ COMMANDER PLUS</b> .....	<b>26</b>
<b>РЕДАКТИРОВАНИЕ ПРОГРАММЫ</b> .....	<b>27</b>
<b>СОХРАНЕНИЕ ПРОГРАММЫ</b> .....	<b>28</b>
<b>РЕЖИМ АВТОЗАПУСКА</b> .....	<b>29</b>
<b>УДАЛЕНИЕ ПРОГРАММЫ</b> .....	<b>29</b>
<b>УДАЛЕНИЕ РЕЖИМА АВТОЗАПУСКА</b> .....	<b>30</b>
<b>ПЕРЕХОД НА ДРУГОЙ ПОРТ ПРОГРАММИРОВАНИЯ</b> .....	<b>30</b>
<b>ПРИЛОЖЕНИЕ В : ПОИСК И УСТРАНЕНИЕ НЕИСПРАВНОСТЕЙ</b> .....	<b>32</b>
<b>НЕ УСТАНАВЛИВАЕТСЯ СВЯЗЬ С СОПРОЦЕССОРНЫМ МОДУЛЕМ</b> .....	<b>32</b>
<b>ПРИЛОЖЕНИЕ С : СХЕМЫ ПОДКЛЮЧЕНИЯ ПО ИНТЕРФЕЙСАМ RS-232 И RS-485</b> .....	<b>34</b>
<b>СТАНДАРТ RS-232</b> .....	<b>34</b>
Название контактов в разъёме RS-232 устройств DTE и DCE и направление тока сигнала..	34
<b>КАБЕЛИ КОМПЬЮТЕРА КОМПАНИИ IBM</b> .....	<b>35</b>
<b>КАК УЗНАТЬ ЯВЛЯЕТСЯ ЛИ ПОРТ ПОРТОМ DCE ИЛИ ПОРТОМ DTE</b> .....	<b>36</b>
<b>RS-232 С КВИТИРОВАНИЕМ (ПОДТВЕРЖДЕНИЕМ УСТАНОВЛЕНИЯ СВЯЗИ)</b> .....	<b>36</b>
<b>СТАНДАРТ RS-485</b> .....	<b>38</b>
<b>ОБМЕН ДАННЫМИ ПО ИНТЕРФЕЙСУ RS-485</b> .....	<b>38</b>
<b>СХЕМА ПОДКЛЮЧЕНИЯ ДЛЯ ОДНОРАНГОВОЙ СЕТИ RS-485</b> .....	<b>38</b>
<b>СХЕМА ПОДКЛЮЧЕНИЯ ДЛЯ ДВУХПРОВОДНОЙ МНОГОАБОНЕНТСКОЙ СЕТИ RS-485</b> .....	<b>39</b>
<b>ЭКРАНИРОВАНИЕ КАБЕЛЯ</b> .....	<b>40</b>
<b>ПОДКЛЮЧЕНИЕ КАБЕЛЕЙ И ЛИНЕЙНЫХ ТЕРМИНАТОРОВ</b> .....	<b>40</b>



## ГЛАВА 1 : ВВЕДЕНИЕ

В этом руководстве описываются некоторые детали, характерные для модуля сопроцессора контроллеров DirectLogic 05 и 06, программируемого на языке бейсик. Этот документ следует использовать как дополнение к руководству FACTS Extended BASIC User's Reference (FA-BASIC-M) – Расширенное руководство пользователя по языку бейсик компании FACTS Engineering – при программировании сопроцессорного модуля вышеназванных контроллеров.

Модуль сопроцессора устанавливается в слот расширения каркаса контроллера D0-05 или в один из четырёх слотов расширения каркаса контроллера D0-06.

Сопроцессорный модуль обменивается информацией с процессором контроллеров DL05 или DL06 с использованием команд S06\_, BMOVE и DPORT.

Для обмена информацией между сопроцессором и контроллером и между контроллером и сопроцессором используется высокоскоростной двухпортовый интерфейс оперативной памяти (RAM<sup>1</sup>) по параллельной шине задней панели контроллеров DL05 или DL06. При использовании команды BMOVE за один цикл контроллера сопроцессор может передать до 256 байт информации. Для обмена данными между сопроцессором и контроллером и между контроллером и сопроцессором не требуется никакой программы на языке релейно-лестничной логики. Сопроцессор не занимает ячеек памяти X и Y в карте памяти CPU<sup>2</sup> PLC<sup>3</sup>.

Программа на лестничной логике в контроллерах DL05 или DL06 командой WX может вызвать прерывание в работе сопроцессора, если в сопроцессоре используется оператор ONPLC. В дополнение к 256 байтам, которые могут быть переданы командой BMOVE, до 256 дополнительных байт информации могут быть переданы в том же цикле ПЛК при использовании такого прерывания.

Сопроцессорный модуль обменивается информацией с внешними устройствами, используя встроенный(е) последовательный(е) порт(ы).

### СИНХРОНИЗАЦИЯ ЦПУ

При включении питания модуль сопроцессора перезагружается и устанавливает связь с ЦПУ контроллеров DL05 или DL06. Затем устанавливается рабочий режим, сохранённый последней командой AUTOSTART. Дополнительную информацию о команде AUTOSTART см в руководстве FACTS Extended BASIC User's Reference

Модуль сопроцессора не перезагружается, если ЦПУ контроллера находится в нерабочем режиме. При необходимости текущее состояние ЦПУ ПЛК можно определить, опросив специальные реле SP11-20. См. Главу 2 (Операторы сопроцессорного модуля), где дано описание оператора S06\_. См. Руководство пользователя контроллеров DL05 или DL06, где приводится описание специальных реле ЦПУ ПЛК.

Пример:	10 IF S06_SP(11) THEN PRINT "Forced running state"	Принудительный рабочий режим (Переключатель режимов ЦПУ в положении RUN)
	20 IF S06_SP(12) THEN PRINT "TERM RUN state"	Терминальный рабочий режим (Переключатель режимов ЦПУ в положении TERM, ЦПУ в рабочем состоянии RUN)
	30 IF S06_SP(13) THEN PRINT "TEST RUN state "	Терминальный режим тестирования (Переключатель режимов ЦПУ в положении TERM, ЦПУ в состоянии тестирования работы)
	40 IF S06_SP(15) THEN PRINT " TEST PGM state"	Терминальный режим тестирования программы (Переключатель режимов ЦПУ в положении TERM, ЦПУ в состоянии тестирования программы)
	50 IF S06_SP(16) THEN PRINT "TERM PGM state"	Терминальный режим программирования (Переключатель режимов ЦПУ в положении TERM, ЦПУ в состоянии программирования)
	60 IF S06_SP(17) THEN PRINT "Forced STOP state"	Принудительный режим останова (Переключатель режимов ЦПУ в состоянии STOP)
	70 IF S06_SP(20) THEN PRINT "TERM STOP state"	Терминальный режим останова (Переключатель режимов ЦПУ в положении TERM, ЦПУ в состоянии STOP)

<sup>1</sup> RAM (Random Access Memory) - оперативная память, оперативное запоминающее устройство

<sup>2</sup> CPU (Central Processing Unit) – центральный процессор, ЦПУ, ЦП

<sup>3</sup> PLC (Programmable Logic Controller) - программируемый логический контроллер, ПЛК

Часто состояния управляющих реле или программных стадий в ЦПУ используются как биты разрешения в программе на языке бейсик. Биты состояния управляющих реле и стадий используются для передачи информации о состоянии программы в сопроцессор. Например, управляющее реле может быть использовано для сигнализации о начале сообщения или просто указывать на то, что ЦПУ ПЛК находится в рабочем режиме.

Пример            10 IF S06\_C(0) THEN PRINT "CR 0 включено"  
                    20 IF S06\_SG(10) THEN PRINT "Стадия 10 активна"

## COMMAND@

Функция            Выбирает порт программирования

Синтаксис            COMMAND@ *port*

Использование      Команда задаёт порт 1 или 3 в качестве порта программирования/порта команд. Бейсик-сoproцессор посылает все сообщения в этот заданный порт и принимает команды только из этого заданного порта.

По умолчанию (заводская установка) в качестве порта программирования/команд используется порт 1 (Port 1) на скорости 9600 Бод.

Для изменения скорости передачи любого последовательного порта при включении электропитания используйте оператор SETPORT.

Для отладки обмена данными с внешним устройством, подключённым к другому порту, используйте команду COMMAND@. Команда COMMAND@ может быть использована для полной загрузки обоих портов с минимальной потребностью в передаче данных по кабелю или использования переключателей.

Пример            Предположим, что программа для диагностического принтера, подключённого к порту 3, завершила свою работу. Теперь на нужно управлять контроллером шагового двигателя, используя порт 1. Перед началом программирования шагового двигателя:

>SETPORT 3, 9600            Установка скорости передачи данных порта 3  
>COMMAND@3            Теперь порт 3 задан как порт программирования

Переключите кабель программирования с порта 1 на порт 3. Для этой цели в комплект поставки модуля сопроцессора включён сплиттер (разветвитель).

Для того чтобы вернуться к программированию через порт 1, ведите  
COMMAND@1

## ГЛАВА 2 : ОПЕРАТОРЫ СОПРОЦЕССОРА

### BMOVE

Функция	Прямой доступ к блоку памяти ЦПУ контроллеров DL05 или DL06
Синтаксис	$BMOVE\ direction, starting\ operand(number), ending\ operand(number)$ $BMOVE\ direction, starting\ operand(number), K\ (number\ of\ bytes)$ <i>direction</i> – направление <i>starting operand</i> – начальный операнд <i>ending operand</i> – конечный операнд <i>number</i> – число <i>number of bytes</i> – число байт
Смотрите также	DPORT, ONPLC, and S06_
Использование	До 256 байт памяти DL05 или DL06 можно прочитать или записать в одном цикле при помощи оператора BMOVE (block move – переместить блок). Память в ЦПУ ПЛК указывается с помощью любого из 11 операндов, заданных восьмеричным адресом <i>number</i> .

Перемещение блока данных начинается в сопроцессоре с ячейки памяти двойного порта DPORT(0) и в ЦПУ ПЛК с *starting operand (number)*. Этот блок данных размещается в ячейках памяти с последовательными адресами вплоть до адреса *ending operand (number)* включительно. Другим способом число передаваемых байт может быть задано как выражение в скобках, которое следует за "K". Если *number of bytes* равно 0, то скопировано будет 256 байт.

Вместо *direction* используйте "R" или "W" для того, чтобы задать чтение (Read) памяти или запись (Write) в неё. "R" приводит к чтению данных из памяти ЦПУ ПЛК и копированию её в память DPORT. "W" приводит к чтению данных из памяти DPORT и копированию её в V-память ЦПУ ПЛК.

Если *starting operand* или *ending operand* имеют битовый (BIT) тип данных, то используется весь адрес V-памяти, содержащий операнд.

### Числовые переменные с плавающей запятой в формате IEEE <sup>4</sup>

Числовые переменные сохраняются в памяти модуля сопроцессора как значение с плавающей запятой в диапазоне от  $\pm 1E-127$  до  $\pm .999999999E+127$ . ЦПУ ПЛК может хранить числа как BCD<sup>5</sup>, BINary<sup>6</sup> или как число с плавающей запятой в формате IEEE в диапазоне  $\pm 3.402822E\pm 38$ . Если вы используете значения с плавающей запятой в ПЛК и хотите обрабатывать эти значения в сопроцессорном модуле, то используйте операторы BMOVE и DPORT с *R portion* или *S06\_VR*.

<sup>4</sup> IEEE (Institute of Electrical and Electronic Engineers) - Институт инженеров по электротехнике и радиоэлектронике, ИИЭР (США)

<sup>5</sup> BCD (Binary Coded Decimal) - двоично-десятичное число

<sup>6</sup> BINary - двоичный

## Восьмеричная система счисления и типы данных для операндов оператора BMOVE

### Операнды BMOVE для DL05

Описание	Операнд	Кол-во	В восьмеричном счислении	Тип данных	Указатель (восьмеричный) пословной V-памяти
Текущие значения таймера (Timer Current)	T	128	0-177	BCD	0-177
Текущие значения счётчиков (Count Current)	CT	128	0-177	BCD	1000-1177
V-память (V-Memory)	VH	3968	1200-7377	HEX <sup>7</sup> или BCD или	1200-7377
Энергозависимая (Volatile)		128	7400-7577	Float <sup>8</sup>	7400-7577
Энергонезависимая (Non-volatile)		128	7600-7777		7600-7777
Системные параметры (System Parameters)					
Входы (Inputs)	X	256	0-377	Bit	40400-40417
Выходы (Outputs)	Y	256	0-377	Bit	40500-40517
Внутренние реле (Internal Relays)	C	512	0-777	Bit	40600-40637
Состояние стадий (Stage Status)	SG	256	0-377	Bit	41000-41017
Состояние таймеров (Timer Status)	TS	128	0-177	Bit	41100-41107
Состояние счётчиков (Counter Status)	CS	128	0-177	Bit	41140-41147
Специальные реле (только чтение) Special Relays (Read Only)	SP	512	0-777	Bit	41200-41237

### Операнды BMOVE для DL06

Описание	Операнд	Кол-во	В восьмеричном счислении	Тип данных	Указатель (восьмеричный) пословной V-памяти
Текущие значения таймера (Timer Current)	T	256	0-377	BCD	0-377
Текущие значения счётчиков (Count Current)	CT	128	0-177	BCD	1000-1177
V-память (V-Memory)	VH	192	400-677	HEX или BCD или	400-677
Энергозависимая (Volatile)		3200	1200-7377	Float	1200-7377
		4096	10000-17777		10000-17777
Энергонезависимая (Non-volatile)		128	7400-7577		7400-7577
Системные параметры (System Parameters)		64	700-777		700-777
		128	7600-7777		7600-7777
		1024	36000-37777		36000-37777
Входы (Inputs)	X	512	0-777	Bit	40400-40437
Выходы (Outputs)	Y	512	0-777	Bit	40500-40537
Внутренние реле (Internal Relays)	C	1024	0-1777	Bit	40600-40677
Состояние стадий (Stage Status)	SG	1024	0-1777	Bit	41000-41077
Состояние таймеров (Timer Status)	TS	256	0-377	Bit	41100-41117
Состояние счётчиков (Counter Status)	CS	128	0-177	Bit	41140-41147
Удалённый ввод/вывод (Remote I/O)	GX GY	2048 2048	0-3777 0-3777	Bit Bit	40000-40177 40200-40377
Специальные реле (только чтение) Special Relays (Read Only)	SP	512	0-777	Bit	41200-41237

<sup>7</sup> HEX (hexadecimal) - шестнадцатеричный

<sup>8</sup> Float – с плавающей запятой

- Пример                      Загрузка таблицы из 6 констант в V-память пользователя, начиная с ячейки памяти V2000
- ```

10 REM Загрузить таблицу в память двойного порта
20 DPORT(0)=10H
30 DPORT(2)=20H
40 DPORT(4)=25H
50 DPORT(6)=30H
60 DPORT(8)=100H
70 DPORT(10)=9798H
80 REM Копировать таблицу в V-память ЦПУ ПЛК
90 BMOVE W, VH(2000), K(12)

```
- Пример                      Умножение данных в некотором диапазоне V-памяти пользователя на постоянную величину
- ```

10 BMOVE R, VH(2000), K(32) : REM Получить значения
20 REM Умножить на 2.5
30 FOR ADDR = 0 TO 31 STEP 2
40 DPORT(ADDR)=DPORT(ADDR)*2.5
50 NEXT ADDR
60 BMOVE W, VH(2000), K(32) : REM Вернуть данные назад

```
- Пример                      Получить таблицу X (входов) контроллера DL240
- ```

10 BMOVE R, X(0), X(477)

```
- Усовершенствование      Если операнд не задан, то адрес является шестнадцатеричным представлением адреса V-памяти в восьмеричном счислении (7FH = 177 – V-память в восьмеричном счислении). BMOVE R, VH(2000), K(10) - это тоже самое, что и BMOVE R, (400H), K(10).
- Это усовершенствование упрощает создание циклов FOR-NEXT и другие типы расчётов при доступе в память ПЛК.
- Пример                      Найти все ячейки пользовательской V-памяти, которые совпадают с константой
- ```

10 K = 1234 : REM Значение константы
15 REM Искать в V-памяти V2000-V7777
20 FOR INDEX=400H TO 1000H STEP 127 : REM 2 BYTES/V-MEM
30 BMOVE R, (INDEX), K(127)
40 FOR ADDR = 0 TO 125 STEP 2
50 IF DPORT(ADDR)<>K THEN 70
60 PRINT1 "Совпадает по шестнадцатеричному адресу V-памяти = ",
62 PRINT1 HEX$(INDEX+ADDR)
70 NEXT ADDR
80 NEXT INDEX

```

## DPORT

Функция	Чтение данных из памяти или запись данных в память, общую с ПЛК DL05 или DL06
Синтаксис	$DPORT (address, portion) = expression$ $variable = DPORT (address, portion)$  <i>address</i> – адрес <i>portion</i> – сегмент <i>expression</i> - выражение <i>variable</i> – переменная (величина)
Использование	<p>DPORT (dual port memory – память двойного порта) используется в связи с прерыванием ONPLC и оператором BMOVE (block move – переместить блок) для доступа к ЦПУ ПЛК.</p> <p>Оператор DPORT находит значение по адресу памяти двойного порта и присваивает это значение переменной.</p> <p>Оператор DPORT сохраняет значение переменной по адресу памяти двойного порта.</p> <p><i>address</i> - это выражение от 0 до 516, с помощью которого выбирается два байта памяти двойного порта. DPORT находит или назначает целочисленное значение (0 - 65535) по адресу</p> <p><i>portion</i> - необязательный элемент и используется для указания позиции бита, нибла (полубайта, группы из 4 бит), байта (группы из 8 бит), двоично-десятичного слова ( 2 байта) или значения с плавающей запятой в формате IEEE (4 байта).</p> <p>Используйте "B(n)", чтобы задать одну из 16 позиций бита, здесь n = 0-15. Используйте "N(n)", чтобы задать один из четырёх ниблов, здесь n = 0-3. Используйте "H", чтобы задать старший байт или "L", чтобы задать младший байт. Используйте "B" чтобы задать преобразование шестнадцатеричного слова в двоично-десятичное. Используйте "R", чтобы задать преобразование значение с плавающей запятой, используемое в языке бейсик, в значение с плавающей запятой IEEE.</p> <p>Первые 256 байт памяти двойного порта, от DPORT(0) до DPORT(255), используются оператором BMOVE при чтении данных из ПЛК или при их записи в ПЛК.</p> <p>Следующие 256 байт памяти двойного порта, от DPORT(256) до DPORT(511), используются вместе с оператором ONPLC. Этот блок памяти доступен ЦПУ ПЛК с помощью команды WX. Последние 5 байт памяти двойного порта, от DPORT(512) до DPORT(516), используются как байты управления для команды WX (см. ONPLC, где приведено полное описание команды).</p>

### Числовые переменные с плавающей запятой в формате IEEE

Числовые переменные сохраняются в памяти модуля сопроцессора как значение с плавающей запятой в диапазоне от  $\pm 1E-127$  до  $\pm 999999999E+127$ . ЦПУ ПЛК может хранить числа как BCD, BINary или как число с плавающей запятой в формате IEEE в диапазоне  $\pm 3.402822E\pm 38$ . Если вы используете значения с плавающей запятой в ПЛК и хотите обрабатывать эти значения в сопроцессорном модуле, то используйте операторы BMOVE и DPORT с R *portion* или S06\_VR.





Пример Извлечение 4-разрядного двоично-десятичного значения (0-9999) из памяти двойного порта

```
10 REM Сохранить двоично-десятичное число в V-памяти по адресу 2000
20 S06_VB(2000)=1234
30 REM Получить это значение с помощью команды BMOVE
40 BMOVE R, VH(2000), K(2)
50 PRINT1 "Двоично-десятичное значение по адресу 2000 V-памяти =",
52 PRINT1 HEX$(DPORT(0))
```

ЗАМЕЧАНИЕ: Воспользуйтесь просмотром данных (DataView) пакета программирования DirectSoft и форматом отображения BCD/HEX для того, чтобы увидеть двоично-десятичные данные в ПЛК.

Пример Сохранение 8-разрядных двоично-десятичных значений (0-99999999) в V-памяти 2000 и 2001 с помощью команды BMOVE

```
10 DPORT(0) = 1234H : REM Константа для V-памяти 2000
20 A = 5678 : REM A должно быть двоично-десятичным значением в диапазоне 0 - 9999
30 DPORT(2) = VAL("0"+STR$(A)+"H") : REM Тоже самое что и DPORT(2,B)=A
40 BMOVE W, VH(2000), VH(2001)
```

ЗАМЕЧАНИЕ: Воспользуйтесь просмотром данных (DataView) пакета программирования DirectSoft и форматом отображения BCD/HEX для того, чтобы увидеть двоично-десятичные данные в ПЛК.

Пример Извлечение шестнадцатеричного/целочисленного значения (0-FFFFH/0-65535d) из памяти двойного порта

```
10 REM Сохранить шестнадцатеричное/десятичное число в V-памяти по адресу 2000
20 S06_VH(2000)=1234
30 REM Получить это значение с помощью команды BMOVE
40 BMOVE R, VH(2000), K(2)
50 PRINT1 " Целочисленное значение по адресу 2000 V-памяти =",
52 PRINT1 HEX$(DPORT(0))
```

ЗАМЕЧАНИЕ: Воспользуйтесь просмотром данных (DataView) пакета программирования DirectSoft и десятичным форматом отображения для того, чтобы увидеть целочисленные данные в ПЛК.

Пример Сохранение значения с плавающей запятой ПЛК, затем извлечение этого значения

```
10 REM Записать значение с плавающей запятой в Write V1400/1401 и прочитать значение с плавающей запятой из V1410/1411
20 DPORT(0,R)= +3.402822E+38
30 BMOVE W,VH(1400),K(4) : REM Значения с плавающей запятой используют 2 слова/4 байта
40 BMOVE R,VH(1410),K(4) : REM Значения с плавающей запятой используют 2 слова/4 байта
50 X=DPORT(0,R)
```

ЗАМЕЧАНИЕ: Воспользуйтесь просмотром данных (DataView) пакета программирования DirectSoft и вещественным или экспоненциальным форматом отображения для того, чтобы увидеть данные с плавающей запятой IEEE в ПЛК.

## Пример

Использование DPORT с модификаторами типа команды PICK

```
1000 V=1120H
1010 DPORT(0)=V : PRINT1 "Извлечение значений из DPORT"
1020 PH1. "DPORT(0) = ",V," в шестнадцатеричном формате"
1030 PRINT1 "1-ый нибл = ",DPORT(0,N(0)), SPC (5),
1040 PRINT1 "3-ий нибл = ",DPORT(0,N(2))
1050 PRINT1 "DPORT(0) в двоичном формате = "; : FOR BT=15 TO 0 STEP -1
1060 IF DPORT(0,B(BT)) THEN PRINT1 "1"; ELSE PRINT1 "0";
1070 NEXT BT: PRINT1
1080 PH1. DPORT(0),
1090 PRINT1 " или ",V," обрабатывается как BCD = ",DPORT(0,B),"
десятичное"
1100 HB=DPORT(0,H) : REM Поменять местами байты
1110 DPORT(0,H)=DPORT(0,L) : DPORT(0,L)=HB
1120 PH1. "Значение с байтами, которые поменялись местами = ",DPORT(0)
1130 PRINT1 : PRINT1 "Присвоение битов и ниблов в DPORT"
1140 DPORT(0)=0
1150 FORBT=0TO15
1160 DPORT(0,B(BT))=1
1170 IFBT=8THEN PRINT1
1180 PH1. DPORT(0), SPC (3),
1190 NEXT : PRINT1
1200 DPORT(0)=0
1210 FORN=0TO3
1220 DPORT(0,N(N))=0FH
1230 PH1. DPORT(0), SPC (3),
1240 NEXT : PRINT1
1250 PRINT1 "Двоично-десятичное назначение"
1260 DPORT(0,B)=1120
1270 PH1. DPORT(0)," = 1120"
```

READY

>run

Извлечение значений из DPORT

DPORT(0) = 1120H в шестнадцатеричном формате

1-ый нибл = 0 3-ий нибл = 1

DPORT(0) в двоичном формате = 0001000100100000

1120H или 4384 обрабатывается как BCD = 1120 десятичное

Значение с байтами, которые поменялись местами = 2011H

Присвоение битов и ниблов в DPORT

0001H 0003H 0007H 000FH 001FH 003FH 007FH 00FFH

01FFH 03FFH 07FFH 0FFFH 1FFFH 3FFFH 7FFFH FFFFH

000FH 00FFH 0FFFH FFFFH

Двоично-десятичное назначение 1120H = 1120

## S06\_

Функция	Прямой доступ к памяти ЦПУ ПЛК
Синтаксис	$S06\_operand(number) = expression$ $variable = S06\_operand(number)$
Сокращённая запись	$S. operand(number)$
Смотрите также	BMOVE, DPORT и ONPLC
Использование	Оператор S06_ перемещает переменную выражения по адресу памяти ЦПУ ПЛК, заданному операндом( <i>number</i> ). Если адрес памяти записывается лестничной программой ЦПУ ПЛК, то оператор S06_ отменяется.

Оператор S06\_ копирует значение из памяти ЦПУ ПЛК по адресу, заданному операндом(*number*) в числовую переменную.

Тип данных значений S06\_ будет двоично-десятичным (VB), шестнадцатеричным (VH), битовым (X,Y,C и т.д.) или с плавающей запятой в формате IEEE(VR) в зависимости от используемого операнда. Дискретные операнды, например, каналы ввода/вывода или управляющие реле работают с битами и возвращают логические значения. Значения таймеров и счётчиков являются двоично-десятичными значениями.

Таблица ниже задаёт восьмеричное исчисление и типы данных для каждого из операндов S06\_ (показано типичное использование операндов VB и VH).

### Числовые переменные с плавающей запятой в формате IEEE

Числовые переменные сохраняются в памяти модуля сопроцессора как значение с плавающей запятой в диапазоне от  $\pm 1E-127$  до  $\pm 999999999E+127$ . ЦПУ ПЛК может хранить числа как BCD, BINary или как число с плавающей запятой в формате IEEE в диапазоне  $\pm 3.402822E\pm 38$ . Если вы используете значения с плавающей запятой в ПЛК и хотите обрабатывать эти значения в сопроцессорном модуле, то используйте операторы BMOVE и DPORT с *R portion* или S06\_VR.

## Восьмеричное исчисление и типы данных для операндов S06\_

### Операнды S06\_ для ЦПУ DL05

Описание	Операнд	Кол-во	В восьмеричном счислении	Тип данных	Указатель (восьмеричный) пословной V-памяти
Таймеры (Timer Current)	T	128	0-177	BCD	0-177
Счётчики (Count Current)	CT	128	0-177	BCD	1000-1177
V-память (V-Memory)	VH			HEX или	
Энергозависимая (Volatile)	VB	3968	1200-7377	BCD или	1200-7377
Энергонезависимая (Non-volatile)	VR	128	7400-7577	Float	7400-7577
Системные параметры (System Parameters)		128	7600-7777		7600-7777
Входы (Inputs)	X	256	0-377	Bit	40400-40417
Выходы (Outputs)	Y	256	0-377	Bit	40500-40517
Внутренние реле (Internal Relays)	C	512	0-777	Bit	40600-40637
Состояние стадий (Stage Status)	SG	256	0-377	Bit	41000-41017
Состояние таймеров (Timer Status)	TS	128	0-177	Bit	41100-41107
Состояние счётчиков (Counter Status)	CS	128	0-177	Bit	41140-41147
Специальные реле (только чтение) Special Relays (Read Only)	SP	512	0-777	Bit	41200-41237

### Операнды S06\_ для ЦПУ DL06

Описание	Операнд	Кол-во	В восьмеричном счислении	Тип данных	Указатель (восьмеричный) пословной V-памяти
Таймеры (Timer Current)	T	256	0-377	BCD	0-377
Счётчики (Count Current)	CT	128	0-177	BCD	1000-1177
V-память (V-Memory)	VH			HEX или	
Энергозависимая (Volatile)	VB	192	400-677	BCD или	400-677
	VR	3200	1200-7377	Float	1200-7377
		4096	10000-17777		10000-17777
Энергонезависимая (Non-volatile)		128	7400-7577		7400-7577
Системные параметры (System Parameters)		64	700-777		700-777
		128	7600-7777		7600-7777
		1024	36000-37777		36000-37777
Входы (Inputs)	X	512	0-777	Bit	40400-40437
Выходы (Outputs)	Y	512	0-777	Bit	40500-40537
Внутренние реле (Internal Relays)	C	1024	0-1777	Bit	40600-40677
Состояние стадий (Stage Status)	SG	1024	0-1777	Bit	41000-41077
Состояние таймеров (Timer Status)	TS	256	0-377	Bit	41100-41117
Состояние счётчиков (Counter Status)	CS	128	0-177	Bit	41140-41147
Удалённый ввод/вывод (Remote I/O)	GX	2048	0-3777	Bit	40000-40177
	GY	2048	0-3777	Bit	40200-40377
Специальные реле (только чтение) Special Relays (Read Only)	SP	512	0-777	Bit	41200-41237

Пример	<p>Использование операндов с битовым типом данных:</p> <pre> 10 REM Отобразить состояние канала ввода X4 20 IF S06_X(4) THEN PRINT1 "Включён" ELSE PRINT1 "Выключен"  10 REM Включить внутреннее управляющее реле ЦПУ ПЛК С400 20 S06_C(400) = 1  10 REM Канал вывод Y23=OFF если CT2 включён и X17 выключен 20 IF S06_CS(2).AND.NOT(S06_X(17)) THEN S06_Y(23) = 0 </pre>
Пример	<p>Использование операндов с двоично-десятичным типом данных:</p> <pre> 10 REM Отобразить текущий отсчёт CNT (счётчика) C10 и TMRF (таймера) T0 20 PRINT1 "Счётчик 10 = ",S06_CT(10) 30 PRINT1 "Таймер 0 = ",S06_T(0)/100  10 REM Разделить текущий отсчёт счётчика C7 на 2 20 S06_CT(7) = S06_CT(7)/2  10 REM Значение канала аналогового ввода находится в ячейке V-памяти 2000 20 REM В ячейку V-памяти 2001 записывается значение для канала аналогового вывода 30 REM Изменять значение канала аналогового вывода пропорционально значению канала аналогового ввода 35 SCALE=.5 : OFFSET=100 40 AOUT = S06_VB(2000) * SCALE - OFFSET 50 REM Ограничить диапазон значения на канале аналогового вывода в пределах (0-4095) 60 IF AOUT &lt; 0 THEN AOUT = 0 70 IF AOUT &gt; 4095 THEN AOUT = 4095 80 S06_VB(2001) = AOUT </pre>
Пример	<p>Использование операндов с шестнадцатеричным типом данных:</p> <pre> 10 REM Отобразить текущее значение времени цикла 20 PRINT1 "Текущее время цикла = ",S06_VH(7775) </pre>
Усовершенствование	<p>Нумерация (адреса) ячеек V-памяти для каждого операнда показана в таблице выше. Операнды VH и VB можно использовать для доступа к любому сегменту V-памяти.</p> <p>Отобразить текущее значение счётчика C0 &gt;P. S.VB(1000)</p> <p>Отобразить состояние первых шестнадцати каналов ввода, X0 - X17 &gt;P. S.VH(40400)</p> <p>Если S06_ используется без операнда, то ячейки V-памяти адресуются в шестнадцатеричном исчислении. Шестнадцатеричный адрес ячейки V-памяти эквивалентен восьмеричному адресу, т.е. S06_VH(2000) это то же самое, что и S06_(400H). Это свойство полезно при создании циклов FOR-NEXT и при других типах доступа к памяти ПЛК, когда требуется производить математические расчёты с данными памяти ПЛК.</p>



## ГЛАВА 3 : ТРЁХПОРТОВЫЙ СОПРОЦЕССОРНЫЙ МОДУЛЬ F0-CP128

### ОБЩИЕ ХАРАКТЕРИСТИКИ F0-CP128

Требования к размещению	- Любой слот расширения для дополнительных модулей
Энергопотребление	-150 мА @ 5 В постоянного тока максимум (от каркаса 05 или 06)
Условия эксплуатации	- 0 - 60 градусов С (32 - 140 градусов F), -влажность 5 - 95% (без конденсации влаги)
Процессор	-CugnalC8051F123
Тактовая частота	-100 МГц, до 100 MIPS <sup>9</sup>
Память пользователя	- Вся память 128К (64К Данные, 64К Программа), энергонезависимая
Физические разъёмы	-1 шестиконтактный разъём RJ12 (Порт 1 и Порт 3 RS-232) -1 трёхклеммная съёмная колодка (Порт 2 RS-485)
Индикаторные светодиоды	- TXD1, RXD1, TXD2, RXD2, RTS1/TXD3, CTS1/RXD3
Порт 1	- RS-232 - Максимум 512000 Бод
Порт 2	- RS-485 - Максимум 512000 Бод
Порт 3	- RS-232 - Максимум 115200 Бод
Дополнительные свойства	- Календарь/Часы с питанием от батареи - Программируется через порт1 или порт 3

<sup>9</sup> MIPS (Million Instructions Per Second) - миллионов (одноадресных) команд в секунду



## F0-CP128 ОПИСАНИЕ

Сопроцессорный модуль, совместимый семейством контроллеров DL05/06 имеет 128К энергонезависимой памяти, три последовательных порта, часы реального времени и календарь, которые питаются от встроенной батареи, арифметику с плавающей запятой и интерпретатор расширенного языка бейсик компании FACTS.

Конвейерная архитектура обработки команд позволяет выполнять 70% процессорных команд в одном или двух системных циклах. Контур с фазовой синхронизацией задаёт внутреннюю тактовую частоту 100 МГц, которая позволяет выполнять до 100 миллионов одноадресных команд в секунду. Скорость выполнения программы бейсик примерно в 10 раз выше, чем в модулях предыдущего поколения. Просто создаваемые и легко поддерживаемые интерпретируемые программы бейсик теперь могут выполняться со скоростью, сравнимой со скоростью компилированной программы бейсик предыдущих версий, языка ассемблера или программ на С.

128К энергонезависимой памяти позволяют сохранять и выполнять несколько программ, поддерживать расширение V-памяти контроллеров DL05 или DL06, сохранять данные и извлекать их. Память поддерживается батареей, и данные в ней сохраняются в течение 10 лет при отсутствии электропитания.

Высокопроизводительный порт 1 может работать на максимальной скорости до 512000 Бод и имеет полностью конфигурируемый последовательный интерфейс RS-232. Высокопроизводительный порт 2 может работать на максимальной скорости до 512000 Бод и имеет полностью конфигурируемый последовательный интерфейс RS-485. Порт 3 может работать на максимальной скорости до 115200 Бод и имеет полностью конфигурируемый последовательный интерфейс RS-232. Все три порта имеют входные буферы клавиатуры ёмкостью в 255 символов для одновременного обмена данными с тремя и более внешними устройствами.

Календарь и часы реального времени, поддерживаемые батареей, сохраняют дату и время, когда происходят нарушения электроснабжения. Прерывания по времени могут быть заданы в программе бейсик с точностью 0,005 секунды.

Арифметика с плавающей запятой позволяет решать сложные формулы с точностью до 8 значащих цифр.

Интерпретатор расширенного языка бейсик компании FACTS имеет множество функций и операторов, которые упрощают программирование, ориентированное на управление.

Программирование через порт 1 или через порт 2 (COMMAND@)

Гибкие команды работы с битами (BITS и PICK)

Прерывания по последовательному порту и по таймеру (ONPORT и ONTIME)

Полный контроль последовательного порта (SETPORT, SETINPUT, PRINT, INPUT, INPLEN, INLEN)

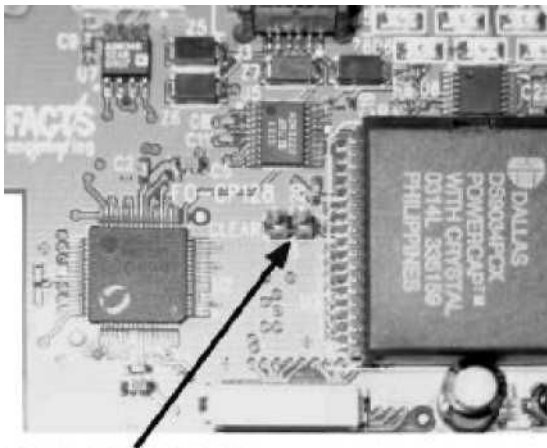
Полный набор команд управления строками (MID\$, LEFT\$, RIGHT\$, REVERSE\$, ASC, CHR\$, LCASE\$, UCASE\$, STR\$, VAL, HEX\$, OCTHEX\$, DATE\$, TIME\$)

Инструменты отладки (TRACE, STOP, CONT)

Сцепление программ (GOPRM)

Операторы и управляющие структуры, общие для большинства языков семейства бейсик.

## ОПИСАНИЕ И РАСПОЛОЖЕНИЕ ПЕРЕМЫЧЕК В F0-CP128



Перемычка CLR ALL (Сбросить всё)

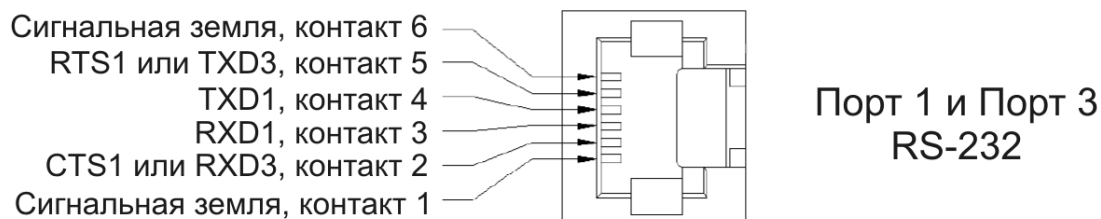
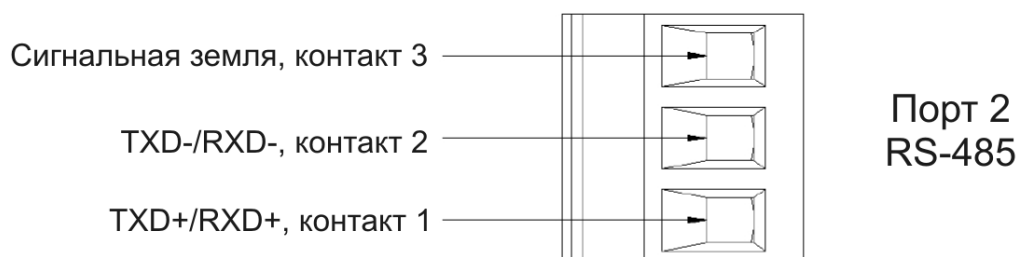
### CLR ALL

Перемычка CLR ALL (Сбросить всё) включает или выключает сброс режима AUTOSTART (Автозапуск). Если установить перемычку на один штырёк переключателя, то будет включён режим AUTOSTART. После включения питания модуль будет использовать параметры, последнего сохранённого режима AUTOSTART. Это положение перемычки является заводской и используется для нормального рабочего режима.

Если соединить перемычкой оба штырька переключателя, то режим AUTOSTART будет отключен, и принудительно все установки будут сброшены. Эту операцию обычно делают только тогда, когда все другие действия по налаживанию связи с модулем сопроцессора не привели ни к какому результату. Это также единственный способ удалить оператор безопасности LOCKOUT (Блокировка). Если включить питание сопроцессорного модуля, когда перемычка CLR ALL соединяет оба штырька, то скорость работы порта 1 и порта 3 устанавливается равной 9600 Бодам.

**ПРЕДУПРЕЖДЕНИЕ:** Замыкание двух штырьков переключателя перемычкой CLR ALL приводит к стиранию программы 0, всех сохранённых переменных, к отмене оператора COMMAND@2, удалению оператора LOCKOUT и сбросу сохранённой информации об AUTOSTART.

## КОНТАКТЫ ПОРТА F0-CP128



Для простого подключения сопроцессорного модуля к компьютеру вместе с модулем поставляется кабель RS-232 с разъёмами на концах и переходник (адаптер) с 9-контактного разъёма на разъём RJ12. Если в вашем портативном компьютере отсутствует порт RS-232, то закажите кабель USB-RS232.

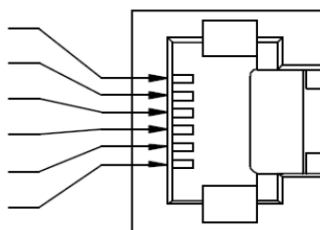
Разъём RS-232 модуля вместе с поставляемым кабелем совместим с разъёмами всех ПЛК. Этот разъём также непосредственно вставляется в FA-15HD (адаптер для 15-контактного разъёма HD DSUB ПЛК), FA-CABKIT (комплект адаптеров RS-232 общего назначения, которые могут использоваться и с модемами, и с разъёмами DB-25) и в FA-ISOCON (преобразователь RS-232 в RS-485 с оптической изоляцией).

## КОНТАКТЫ ПОРТА РАЗВЕТВИТЕЛЯ (СПЛИТТЕРА)

Если сигналы RTS1 и CTS1 не используются, то установите разветвитель портов 1 и 3, показанный на рисунке ниже, на порт RS-232 сопроцессорного модуля. Установка этого разветвителя позволяет просто подключить кабели RS-232 к обоим портам, порту 1 и порту 3.

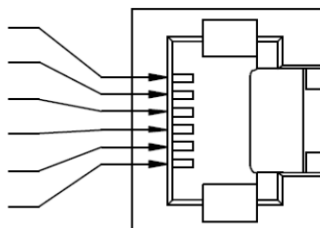


Сигнальная земля, контакт 6  
Не подключён, контакт 5  
TXD3, контакт 4  
RXD3, контакт 3  
Не подключён, контакт 2  
Сигнальная земля, контакт 1



Порт 3  
RS-232

Сигнальная земля, контакт 6  
Не подключён, контакт 5  
TXD1, контакт 4  
RXD1, контакт 3  
Не подключён, контакт 2  
Сигнальная земля, контакт 1



Порт 1  
RS-232



## ПРИЛОЖЕНИЕ А : ПРИСТУПАЯ К РАБОТЕ С СОПРОЦЕССОРНЫМ МОДУЛЕМ

### ПРЕДВАРИТЕЛЬНЫЕ ДЕЙСТВИЯ С ИСПОЛЬЗОВАНИЕМ АВМ COMMANDER PLUS

1. Запустите АВМ Commander для Windows.
2. Познакомьтесь со справочной системой/командами АВМ Commander для Windows.
3. Соедините кабелем компьютер и сопроцессорный модуль. Схема подключения приведена в ПРИЛОЖЕНИИ С.
4. Включите электропитание ПЛК.
5. Выберите меню "Communication" (Коммуникации) с выпадающим списком и в этом списке выберите "Parameters(Port)" (Параметры(Порт)).
6. Выберите последовательный порт в компьютере, которым будете пользоваться. Щёлкните по кнопке "Defaults" (Заводская установка). Установятся следующие параметры связи: 9600, 8, нет, 1, нет. Щёлкните по кнопке "Apply" (Применить).
7. В главном меню выберите "COMMAND MODE Connect to BASIC Module" (КОМАНДНЫЙ РЕЖИМ подключиться к сопроцессорному модулю). В меню COMMAND MODE (Командный режим) выберите "SYstem\_Stats" (Системная статистика).
8. Модуль ответит подсказкой  
READY  
> (">" – символ (значок), указывающий, что BASIC (бейсик) находится в командном (COMMAND) режиме.  
Если вы не получите этот значок после сообщения, то обратитесь к ПРИЛОЖЕНИЮ В, где описаны характерные неисправности и способы их исправления.
9. Сопроцессорный модуль, программируемый на языке бейсик, готов для программирования в режиме реального времени, текущего контроля или загрузки или скачивания программы.

## РЕДАКТИРОВАНИЕ ПРОГРАММЫ

Действия пользователя	Отображение на дисплее
<p>В строке меню выберите 'Auto'. Выберите Mode 0 (Режим 0), Program 0 (Программа 0), и щелкните 'OK'.</p>	<pre>AUTOSTART 0,0  Mode = 0, Edit Program = 0 Port 1 Baud = 9600 Programming (Port 2 = 9600) (Port 3 = 9600)  &gt;</pre>
<p>Введите в поле 'Command Line' (Командная строка): 10 p. &lt;ENTER&gt; 65535 p. &lt;ENTER&gt;</p>	<pre>&gt;10 p. &gt;65535 p.  &gt;</pre>
<p>В строке меню выберите 'Rese<u>T</u>' (Сброс, перезагрузка). Выключение и повторное включение электропитания ПЛК также приводит к перезагрузке сопроцессорного модуля, программируемого на языке бейсик.</p>	<pre>RESET  FACTS Extended BASIC Plus DL05/06 PLCs Warp Drive CoProcessor Version 1.00/HS (c)Copyright FACTS Engineering, Inc. 1988 – 2004  AUTOSTART Mode, Program, Baud Mode = 0, Edit Program = 0 Port 1 Baud = 9600 Programming (Port 2 = 9600) (Port 3 = 9600)  0 stored programs, 65528 program storage bytes free (0 сохранённых программ, 65528 байт свободно для сохранения программы)  PRM 0 READY  &gt;</pre>
<p>В строке меню выберите '<u>L</u>ist' (Список). Обратите внимание, что в режиме 0 используется скорость передачи по последовательному каналу, хранимая в модуле. Программа в буфере редактирования, PROGRAM 0, в режиме 0 сохраняется при пропадании электропитания.</p>	<pre>list 10 PRINT1 65535 PRINT1 PRM 0 READY  &gt;</pre>

## СОХРАНЕНИЕ ПРОГРАММЫ

Действия пользователя	Отображение на дисплее
В строке меню выберите 'Ne <u>W</u> '.	NEW >
Введите в поле 'Command Line' (Командная строка): 10 P."MY FIRST PROGRAM - МОЯ ПЕРВАЯ ПРОГРАММА" <ENTER>	>10 P. " MY FIRST PROGRAM - МОЯ ПЕРВАЯ ПРОГРАММА" >
Выберите 'SaVe'  ПРИМЕЧАНИЕ: F0-CP128 поставляется с предустановленной программой диагностики в PRM1, поэтому первая СОХРАНЁННАЯ программа помещается в PRM2.	SAVE  Saving program 2 (Сохранение программы 2)  2 stored programs, 64310 program storage bytes free (2 сохранённые программы 64310 байт свободно для сохранения программ)  PRM 0 READY >
Введите в поле 'Command Line' (Командная строка): 10 P."MY SECOND PROGRAM - МОЯ ВТОРАЯ ПРОГРАММА" <ENTER>	>10 P. "MY SECOND PROGRAM - МОЯ ВТОРАЯ ПРОГРАММА" >
Выберите 'SaVe'	SAVE  Saving program 3 (Сохранение программы 3)  3 stored programs, 64284 program storage bytes free (3 сохранённые программы 64284 байт свободно для сохранения программ)  PRM 0 READY >



## РЕЖИМ АВТОЗАПУСКА

Действия пользователя	Отображение на дисплее
В строке меню выберите 'Auto' (Автозапуск). Выберите Mode 1 (Режим 1), Program 2 (Программа 2), и щёлкните по 'OK'. Эти действия определяют, что сопроцессорный модуль, программируемый на языке бейсик, после перезагрузки запустит в работу программу 2	AUTOSTART 1,2  Mode = 1, RUN (CLEAR) Program = 2 Port 1 Baud = 9600 Programming (Port 2 = 9600) (Port 3 = 9600) >
В строке меню выберите 'Rese <u>T</u> ' (Сброс, перезагрузка). Выключение и повторное включение электропитания ПЛК также приводит к перезагрузке сопроцессорного модуля, программируемого на языке бейсик.	RESET MY FIRST PROGRAM – МОЯ ПЕРВАЯ ПРОГРАММА  PRM2 READY >
В строке меню выберите ' <u>S</u> el' (Выбор). Щёлкните по кнопке с зависимой фиксацией Program 0' и затем по 'OK'.	>
В строке меню выберите ' <u>L</u> ist' (Список). Подтвердите, что программа в буфере редактирования (PRM0) всё ещё существует.	list 10 PRINT1 "MY SECOND PROGRAM – МОЯ ВТОРАЯ ПРОГРАММА" PRM0 READY >

## УДАЛЕНИЕ ПРОГРАММЫ

Действия пользователя	Отображение на дисплее
В строке меню выберите 'Del' (Удалить). Введите '2', затем 'OK'. В диалоговом окне подтверждения щёлкните по 'Yes' (Да)	DELPRM2 2 stored programs, 64310 program storage bytes free (2 сохранённые программы 64310 байт свободно для сохранения программ)
В строке меню выберите 'Rese <u>T</u> '. Выключение и повторное включение электропитания ПЛК также приводит к перезагрузке сопроцессорного модуля, программируемого на языке бейсик.	RESET MY SECOND PROGRAM – МОЯ ВТОРАЯ ПРОГРАММА  PRM 2 READY

## УДАЛЕНИЕ РЕЖИМА АВТОЗАПУСКА

Действия пользователя	Отображение на дисплее
В строке меню выберите 'Auto' (Автозапуск). Выберите Mode 0 (Режим 0), Program 0 (Программа 0), и щёлкните по 'OK'. Эти действия определяют, что сопроцессорный модуль, программируемый на языке бейсик, после перезагрузки начнёт работу в режиме редактирования	AUTOSTART 0,0 Mode = 0, Edit Program = 0 Port 1 Baud = 9600 Programming (Port 2 = 9600) (Port 3 = 9600) >

## ПЕРЕХОД НА ДРУГОЙ ПОРТ ПРОГРАММИРОВАНИЯ

При обмене данными с двумя или тремя внешними устройствами можно изменить порт программирования с порта 1 на порт 2 или даже на порт 3. Для того чтобы подключить компьютер с портом RS-232 к порту 2 необходим преобразователь интерфейса RS-232 в RS-485. Переход на другой командный порт показан в нижеследующей таблице.

Действия пользователя	Отображение на дисплее
В поле 'Port Select' (Выбор порта) – в командном окне внизу слева – выберите кнопку с зависимой фиксацией 'Port 3' (Порт 3).	На дисплее нет изменений
В поле 'Port Select' (Выбор порта) щёлкните по кнопке 'Command Port (ABM)' (Командный порт (ABM)). Щёлкните по 'Yes' (Да) в диалоговом окне подтверждения.	На дисплее нет изменений
Переключите кабель с порта 1 на порт 3, затем щёлкните по 'OK' в диалоговом окне, в котором предлагается переключить кабель.	На дисплее нет изменений
В строке меню выберите 'SYstem_Stats' (Системная статистика)	>



## ПРИЛОЖЕНИЕ В : ПОИСК И УСТРАНЕНИЕ НЕИСПРАВНОСТЕЙ НЕ УСТАНОВЛИВАЕТСЯ СВЯЗЬ С СОПРОЦЕССОРНЫМ МОДУЛЕМ

1. Если светодиод RXD порта 1 мигает, когда данные поступают на клеммы модуля, то переходите к шагу 2. Если светодиод не мигает, то воспользуйтесь выпадающим окном RS232 для выяснения, в чём причина: в кабеле или в компьютере.
2. Выключите питание, выньте модуль из каркаса контроллера и установите переключку "CLR ALL" на оба контакта переключателя (см. стр.22).

**ПРЕДОСТЕРЕЖЕНИЕ:** Установка переключки CLR ALL приведёт к стиранию программы 0, всех сохранённых данных, удалению COMMAND@2, удалению LOCKOUT и сохранённой информации об AUTOSTART.

3. Запустите ABM Commander для Windows.
4. Загляните в справочную систему ABM Commander для Windows.
5. Соедините кабелем компьютер и сопроцессорный модуль. Схема подключения приведена в ПРИЛОЖЕНИИ С.
6. Включите электропитание ПЛК.
7. Выберите меню "Communication" (Коммуникации) с выпадающим списком и в этом списке выберите "Parameters(Port)" (Параметры(Порт)).
8. Выберите последовательный порт в компьютере, которым будете пользоваться. Щёлкните по кнопке "Defaults" (Заводская установка). Установятся следующие параметры связи: 9600, 8, нет, 1, нет. Щёлкните по кнопке "Apply" (Применить).
9. В главном меню выберите "COMMAND MODE Connect to BASIC Module" (КОМАНДНЫЙ РЕЖИМ подключиться к сопроцессорному модулю). В меню COMMAND MODE (Командный режим) выберите "SYstem\_Stats" (Системная статистика).
10. Модуль ответит подсказкой READY.
11. Наберите следующую команду и нажмите клавишу Return (Ввод)  
>AUTOSTART 0,0
12. Выключите питание контроллера и выньте модуль из каркаса контроллера. Установите переключку "CLR ALL" только на один контакт переключателя.

13. Поместите модуль снова в каркас контроллера, включите питание контроллера. Модуль должен теперь ответить сообщением

```
FACTS Extended BASIC Plus
```

```
...
```

```
READY
```

> (">" – символ (значок), указывающий, что BASIC (бейсик) находится в командном (COMMAND) режиме.

## ПРИЛОЖЕНИЕ С : СХЕМЫ ПОДКЛЮЧЕНИЯ ПО ИНТЕРФЕЙСАМ RS-232 И RS-485

### СТАНДАРТ RS-232

RS-232-C (RS-232) – это стандарт на интерфейс от Electronic Industries Association (EIA)<sup>10</sup>. Этот стандарт даёт название и определение 20 коммуникационным сигналам, которые назначены отдельным контактам в 25-контактном разъёме. На пять остальных неопределённых контактов можно подать нестандартные сигналы, которые необходимы для конкретной системы.

Каждый сигнал представляет собой ток прямого или обратного направления напряжением между 3 и 15 Вольтами (обычно 12 В). Ток сигнала, назначенного данному контакту, течёт только в одном направлении. К примеру, выходной сигнал от компьютера должен втекать в терминальное (внешнее) устройство и наоборот.

Сигналы RS-232 передаются по кабелю последовательного интерфейса, в кабеле может быть до 25 проводов. В связи с тем, что для простого обмена данными большинство сигналов не нужны, то в кабеле может быть всего 2 или три провода. Как показано на следующих схемах, на одном или обоих разъёмах часто устанавливаются переключки для того, чтобы создать необходимые управляющие сигналы.

Сигналы передаются между двумя типами интерфейсных портов, портом аппаратуры передачи данных (data communication equipment (DCE)) и портом оконечного (терминального) оборудования информационной сети (data terminal equipment (DTE)). Названия контактов разъёмов в устройствах DCE и DTE одинаковые, однако, направление тока сигналов обратное.

#### Название контактов в разъёме RS-232 устройств DTE и DCE и направление тока сигнала

Кон-т	Сокраще ние	Название	Направление тока сигнала		Описание
			DCE (модем)	DTE (Компьютер)	
1	FG	Frame Ground (Защитная земля)	Нет	Нет	
2	TXD	Transmit Data (Передача данных)	в	из	DTE передаёт данные
3	RXD	Receive Data (Приём данных)	из	в	DCE передаёт данные
4	RTS	Request to Send (Запрос от DTE на передачу данных)	в	из	У DTE есть данные для передачи
5	CTS	Clear to Send (DCE готово к приёму данных)	из	и	Устройству DTE разрешается передавать данные
6	DSR	Data Set Ready (Запрос от DCE на передачу данных)	из	в	У DCE есть данные для передачи
7	SG	Signal Ground (Сигнальная (рабочая) земля)	в	из	
8	DCD	Data Carrier Detect Обнаружен носитель данных)	из	в	У модема есть носитель (несущая) данных
20	DTR	Data Terminal Ready (DTE готово к приёму данных)	в	из	Устройству DCE разрешается передавать данные
22	RI	Ring Indicator (Указатель вызовов)	из	в	

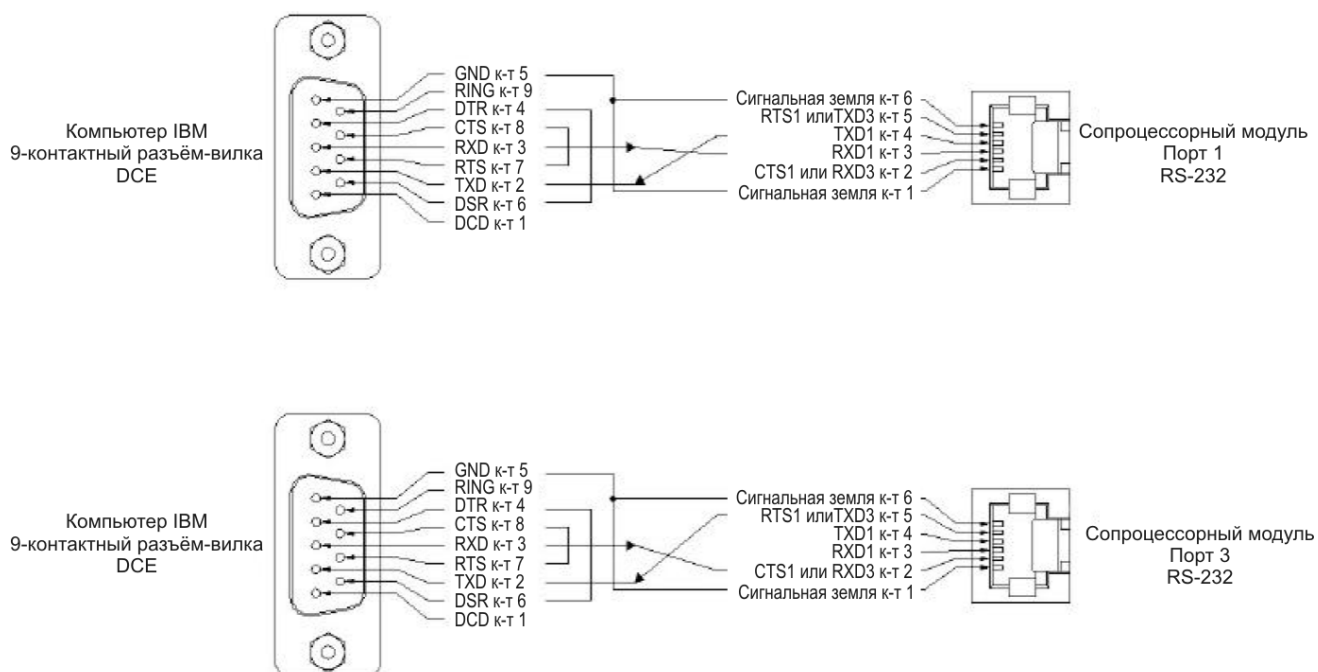
<sup>10</sup> Electronic Industries Association (EIA) – Альянс (объединение) отраслей электронной промышленности, США

## КАБЕЛИ КОМПЬЮТЕРА КОМПАНИИ IBM

В комплект поставки F0-CP128 для подключения к компьютеру входит кабель RS-232 и переходник (адаптер) с 9-контактным разъёмом DSUB. Также в поставку входит разветвитель (сплиттер) портов 1 и 3. Этот сплиттер позволяет подключить компьютер или к порту 1 или порту 3 сопроцессорного модуля.

В наборе дополнительных устройств FA-CABKIT, поставляемых компанией AutomationDirect, имеется кабель RS-232 и адаптеры для подключения к большинству устройств, поддерживающих стандарт RS-232, включая дополнительный компьютер. С помощью этого набора можно быстро и просто создать соединение по сети RS-232 между сопроцессорным модулем и внешним устройством. Если вам нужен экранированный кабель или кабель другой длины, то воспользуйтесь следующими схемами для изготовления интерфейсного кабеля.

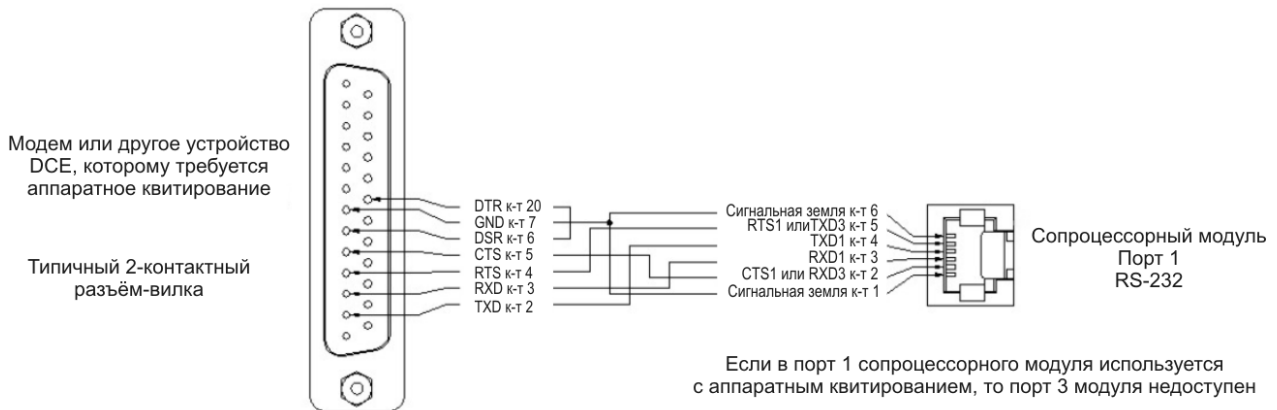
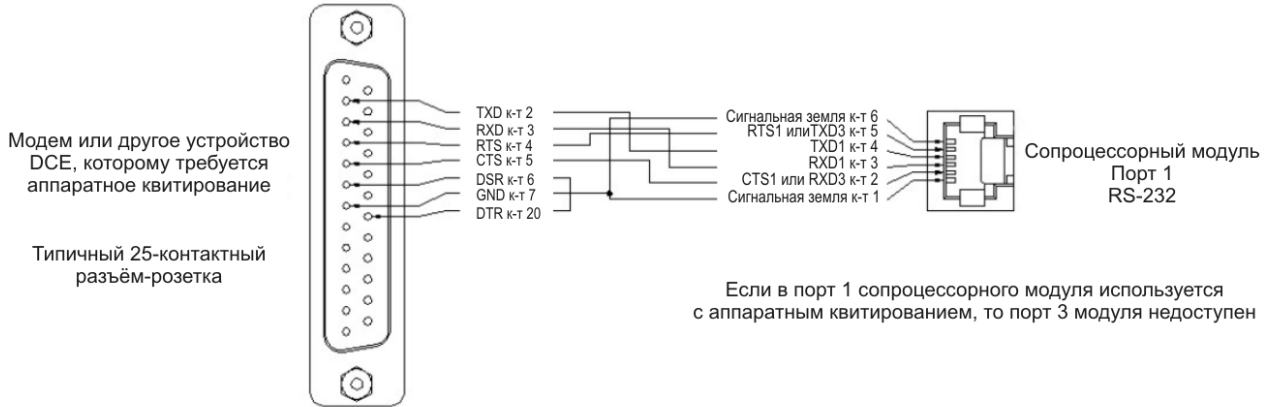
У большинства ноутбуков последних выпусков нет порта RS-232. Для подключения таких компьютеров к сопроцессорному модулю вам понадобится кабель-адаптер USB - 9-контактный разъём RS-232 (Код для заказа у компании AutomationDirect: USB-RS232). Кабель RS-232 и адаптер, поставляемые вместе с F0-CP128, вставляются в кабель-адаптер USB-RS232.



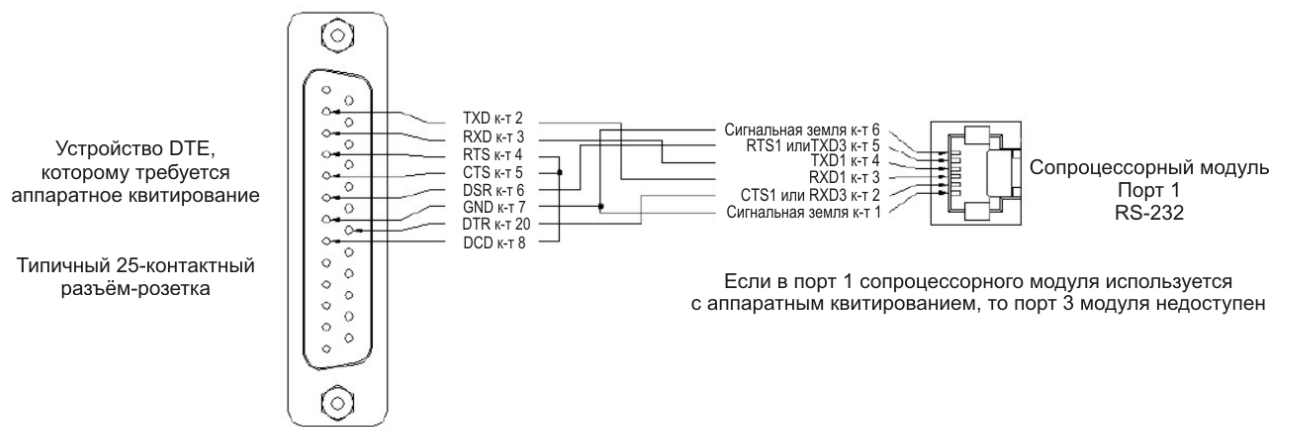
## КАК УЗНАТЬ ЯВЛЯЕТСЯ ЛИ ПОРТ ПОРТОМ DCE ИЛИ ПОРТОМ DTE

Если на устройство подано электропитание, то измерьте напряжение постоянного тока между контактом 2 и контактом 7 (Земля), а также между контактом 3 и земельным контактом порта. Если контакт 2 находится под более отрицательным напряжением, то тогда это порт DTE. Если контакт 3 находится под более отрицательным напряжением, то тогда это порт DCE. Неправильное подключение к контактам 2 и 3 не приводит к повреждению порта.

## RS-232 С КВИТИРОВАНИЕМ (ПОДТВЕРЖДЕНИЕМ УСТАНОВЛЕНИЯ СВЯЗИ)







СХЕМЫ ПОДКЛЮЧЕНИЯ ПО ИНТЕРФЕЙСАМ RS-232 и RS-485

## СТАНДАРТ RS-485

Передачики RS-485 сопроцессорного модуля совместимы с сигналами RS-485.

Стандарт RS-485 модернизированная версия стандарта EIA RS-422-A, в котором предлагаются драйверы с повышенным током и тристабильными выходами. Эти драйверы внутренне защищены от конфликтных ситуаций на шине, вызванных присутствием на одной и той же линии нескольких драйверов. Драйверы RS-485 также выдерживают повышенные напряжения на своих выходах, когда эти выходы отключены (находятся в высокоимпедансном состоянии). Стандарт RS-485 определён для работы систем с несколькими передатчиками и несколькими приёмниками, а также для одно- и многоабонентской (multi-drop) передачи данных. Стандарт RS-485 позволяет включать на одну и ту же линию передачи данных до 32 передатчиков и приёмников.

## ОБМЕН ДАННЫМИ ПО ИНТЕРФЕЙСУ RS-485

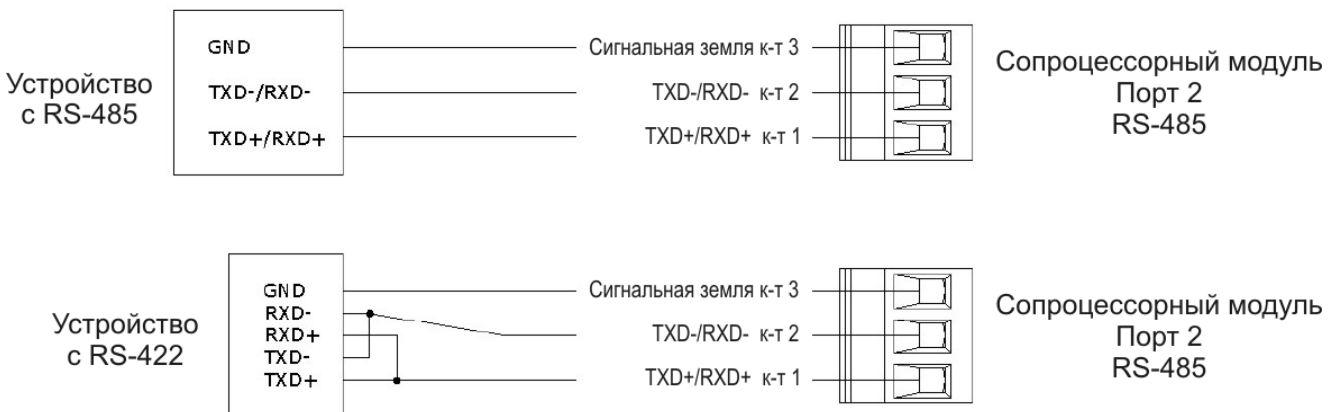
Сопроцессорный модуль осуществляет обмен данными по интерфейсу RS-485 через порт 2. Компенсация эхо-сигналов осуществляется автоматически. Устройство с интерфейсом RS-232 может быть подключено к этому порту через преобразователь RS-232 в RS-485 (код заказа в компании AutomationDirect FA-ISOCN)..

Для того чтобы включить передатчики RS-485 только для печати (PRINTing), воспользуйтесь для выбора многоабонентского режима "M" командой SETPORT. Пользуйтесь многоабонентским режимом, когда сопроцессорный модуль работает как ведомое устройство в конфигурации ведущее/ведомое устройство или когда требуется одноранговая (peer to peer) конфигурация.

Для того чтобы оставить передатчики RS-485 во включённом состоянии даже тогда, когда нет печати (PRINTing), выберите одноранговый режим "P", воспользовавшись командой SETPORT. Пользуйтесь одноранговым режимом, когда сопроцессорный модуль работает как единственное ведущее устройство в конфигурации ведущее/ведомое устройство или в одноранговой конфигурации. Эта конфигурация обеспечивает наивысшую помехозащищённость, так как драйверы RS-485 остаются включёнными и препятствуют приёму шума ведомыми устройствами в сети.

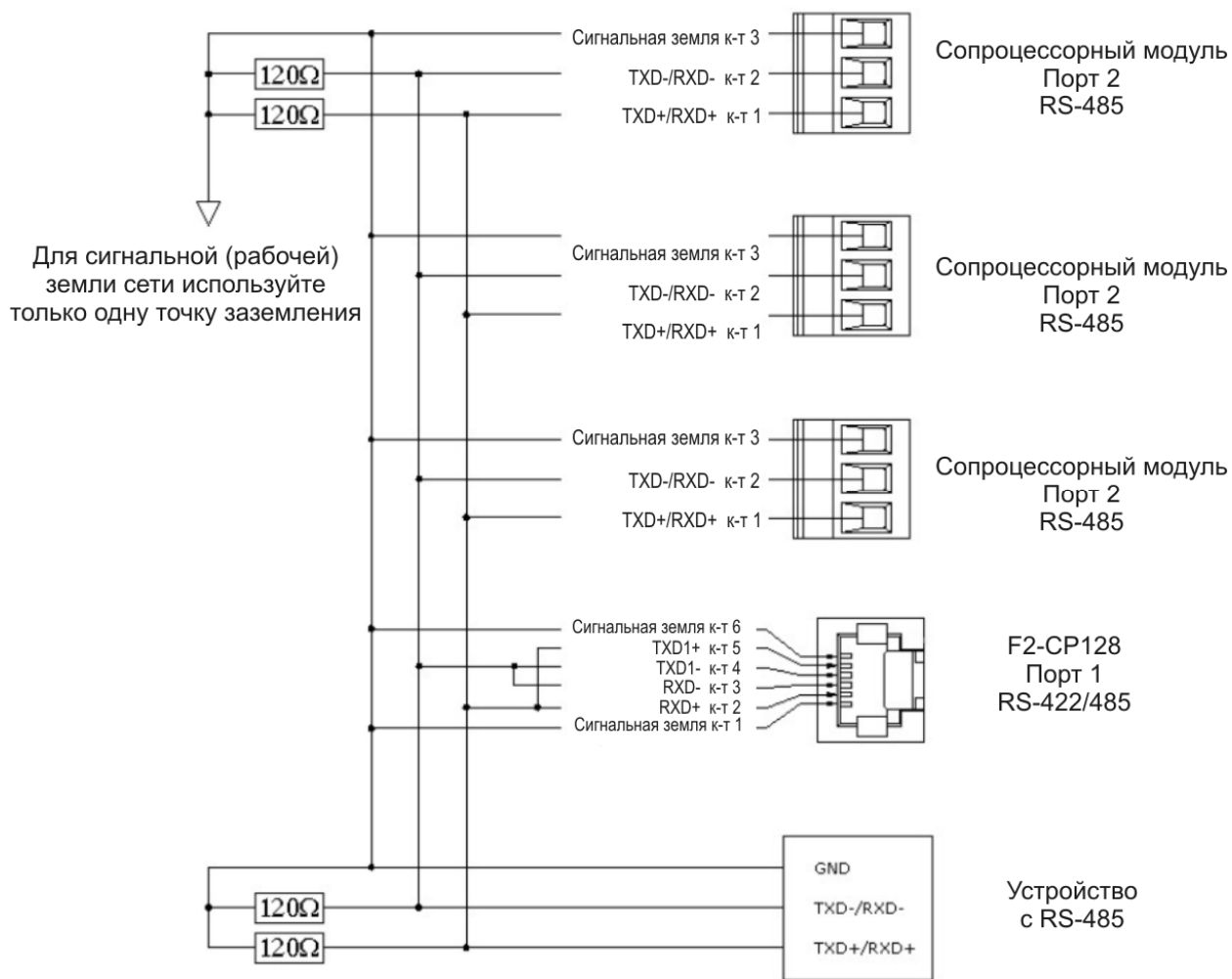
Пример:	Задайте следующие параметры порта 1: 9600 Бод, нет контроля по чётности, длина слова – 8 бит, 1 стоповый бит, программное квитирование XON/XOFF <sup>11</sup> , и многоабонентский режим RS-485. SETPORT 1,9600, N, 8, 1,S, M
---------	--

## СХЕМА ПОДКЛЮЧЕНИЯ ДЛЯ ОДНОРАНГОВОЙ СЕТИ RS-485



<sup>11</sup> XON/XOFF (Transmitter On/Transmitter Off) - протокол XON/XOFF простейший протокол передачи данных между устройствами по асинхронному соединению

## СХЕМА ПОДКЛЮЧЕНИЯ ДЛЯ ДВУХПРОВОДНОЙ МНОГОАБОНЕНТСКОЙ СЕТИ RS-485



СХЕМЫ ПОДКЛЮЧЕНИЯ ПО ИНТЕРФЕЙСАМ RS-232 и RS-485

## **ЭКРАНИРОВАНИЕ КАБЕЛЯ**

Экранирование улучшает помехозащищённость (защиту от воздействия магнитного поля). Важно заземлять экран только на стороне приёмника. Заземление только на стороне приёмника обеспечивает наименьшее ослабление сигнала высокой частоты и наилучшее подавление нежелательных сигналов. Заземление на обоих концах кабеля вызывает протекание через заземление токов помех, наведённых магнитным полем. Шум от этих помех может затем появиться в линиях данных из-за их трансформаторной связи с экраном. Если экран кабеля используется в качестве проводника системного заземления, то тогда для уменьшения токов помех через землю следует установить резистор сопротивлением 100 Ом последовательно с экраном и подключить точку соединения экрана с резистором к заземлению.

## **ПОДКЛЮЧЕНИЕ КАБЕЛЕЙ И ЛИНЕЙНЫХ ТЕРМИНАТОРОВ**

Для двухпроводных сетей RS-485 рекомендуется использовать кабель с витыми жилами и заземление. Для предотвращения ошибок данных в симметричной линии передачи необходимо её согласовать надлежащим образом. Типичный одножильный провод AWG 22 в пластиковой оболочке 0.060 дюйма, скрученный 4,5 раза на каждый фут обладает волновым сопротивлением, величина которого приблизительно равна 120 Ом. Таким образом, выбираем для согласования четыре резистора-терминатора и устанавливаем их между линий и землёй (два сопротивления в параллель на каждой линии – это 60 Ом). Согласование сопротивлением с линии на землю предпочтительнее согласования с линии на линию сопротивлением 120 Ом. В системах с сильно шумящими или длинными линиями лучшая способность ослабления шумов резисторами-терминаторами, включёнными с линии на землю, особенно важна. В многоабонентских сетях линия должна быть согласована только на стороне самого последнего устройства, как показано на двух последних рисунках. Добавление промежуточных согласующих сопротивлений неблагоприятно сказывается на линии передачи.

У некоторых устройств с RS-485 нет заземляющего контакта. У них есть контакт RS-485 плюс и контакт RS-485 минус, но нет контакта для заземления. В этом случае следует использовать изолятор сети (network isolator), например, FA-ISOCAN компании AutomationDirect, для того, чтобы избежать соединения с контактом заземления на сопроцессорном модуле. FA-ISOCAN обеспечивает изоляцию между портом RS-232 сопроцессорного модуля и сетью RS-485. Когда применяется FA-ISOCAN, то порт RS-485 на сопроцессорном модуле не используется.