

Тема 1. Терминология

АППАРАТНЫЕ СРЕДСТВА

Различают 3 уровня автоматизации:

Полевой – включает в себя совокупность датчиков, исполнительных механизмов и промышленные контроллеры – это автоматизация отдельных технологических участков.

Цеховой уровень – система управления несколькими связанными технологическими процессами.

Уровень предприятия – сюда включают еще и систему документирования.

Обзор продукции SIMATIC

Весь спектр продукции представлен на рис. 1 [1].

- ASI – совокупность аппаратных и программных средств для работы с объектом управления, то есть со сбором информации с датчиков и с выдачей сигналов на исполнительный механизм.
- SIMATIC DP – средства, обеспечивающие работу децентрализованной периферии (DP).
- SIMATIC Controller – семейство программируемых логических контроллеров (ПЛК):
 - S7-200 (рис. 2) и S7-1200 – для малых систем автоматизации;
 - S7-300, S7-400 и S7-1500 – для больших систем.
- SIMATIC PG, SIMATIC PC – средства для программирования всех основных компонентов автоматизации:
 - программатор PG,
 - компьютер PC.

SIMATIC NET – средство для работы сетей.

- a) MPI – многоточечный интерфейс.
- b) PROFIBUS – промышленная полевая сеть.
- c) Industrial Ethernet – для организации уровня автоматизации предприятия.

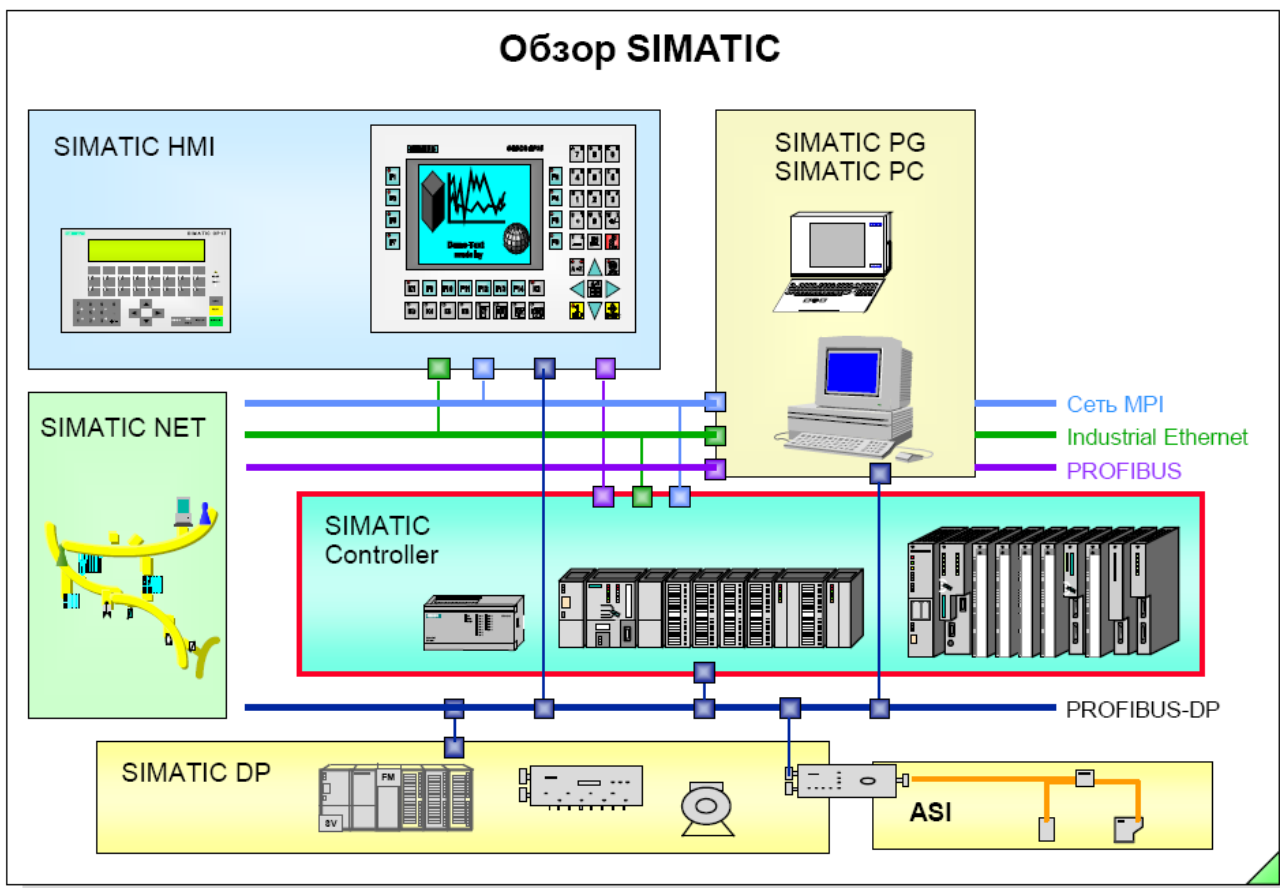


Рис. 1. Обзор продукции SIMATIC

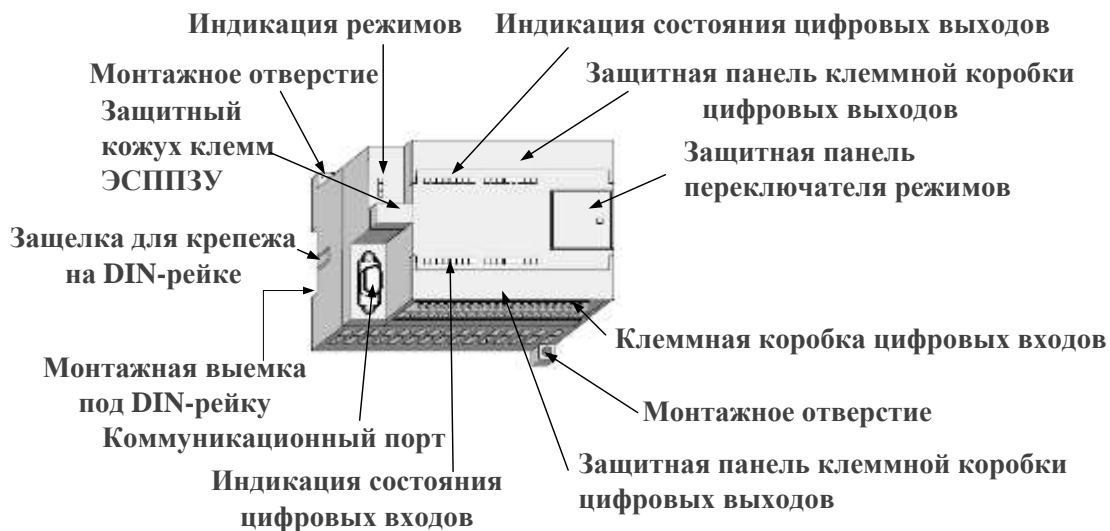


Рис. 1. Компоненты SIMATIC S7-200

CPU семейства S7-200

Проектирование электроустановок осуществляется на базе микроконтроллеров, в частности, программируемых логических контроллеров (ПЛК) S7-200 фирмы SIEMENS [1,2] с программным пакетом STEP7 Micro/Win промышленного программного обеспечения SIMATIC. Данное инструментальное средство применяется в самых простых системах автоматизации.

Основная структура систем автоматического управления с использованием ПЛК S7-200 (рис. 3) включает в себя:

- центральное устройство S7-200;
- персональный компьютер в качестве программирующего устройства для ПЛК;
- программное обеспечение STEP7-Micro/WIN;
- соединительный кабель, осуществляющий коммутацию персонального компьютера с ПЛК.

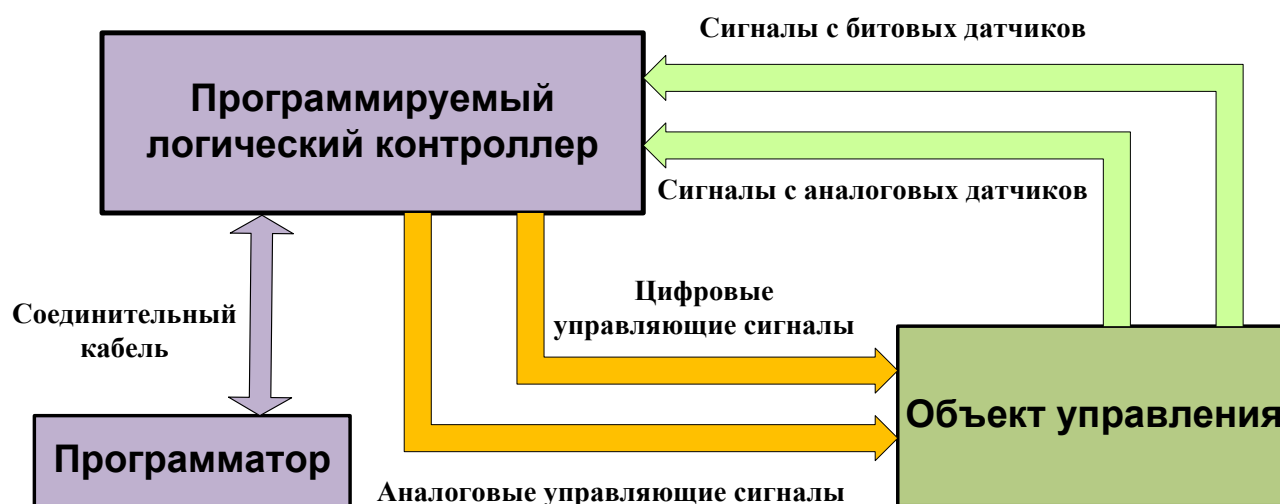


Рис. 3. Структурная схема системы автоматизации на базе ПЛК

В свою очередь ПЛК содержит:

- плата центрального программирующего устройства;
- входы - контролируют сигналы приборов (переключателей, датчиков);
- выходы - управляют электроприводами и другими устройствами;
- коммуникационный порт - через него можно подключить к CPU устройство программирования или другие CPU;
- индикаторы состояния - предоставляют визуальную информацию о режиме работы CPU (RUN - рабочий режим; STOP - режим останова), текущем состоянии сигналов встроенных входов и выходов.

Помимо индикации состояния встроенных цифровых входов и выходов на передней панели CPU реализованы переключатели и индикация режимов работы:

STOP – режим, в котором программа не выполняется, но возможна загрузка программы из программатора, загрузка программы из CPU в программатор.

При этом сам CPU ничего не считает;

RUN – режим выполнения программы, записанной в CPU. Изменение программы невозможно.

В таблице 1 приведены основные характеристики CPU для семейств ПЛК S7-200. В случае недостаточности количества встроенных в CPU входов и выходов в семействе S7-200 реализованы дискретные (таблица 2) и аналоговые (таблица 3) устройства расширения. Адресация встроенных цифровых входов CPU всегда начинается с адреса I0.0, выходов – с Q0.0. Адресация входов и выходов цифровых модулей расширения начинается с младшего бита ближайшего свободного байта соответствующей области памяти.

Таблица 1

Основные характеристики CPU S7-200

CPU 212	CPU 214	CPU 215	CPU 216	CPU 221	CPU 222	CPU 224	CPU 226
1. Физический размер, мм							
160x80x62	197x80x62	218x80x62	218x80x62	90x80x62	90x80x62	120,5x80x62	190x80x62
2. Память программ							
512 слов	2Кслов	4К слов	4К слов	2К слов	2К слов	4К слов	4К слов
3. Данные пользователя							
512 слов	2К слов	2,5К слов	2,5К слов	1К слов	1К слов	2,5К слов	2,5К слов
4. Внутренние маркеры							
128	256	256	256	256	256	256	256
5. Встроенные цифровые входы/выходы, бит							
8DI/6DO	14DI/10DO	14DI/10DO	14DI/10DO	6DI/4DO	8DI/6DO	14DI/10DO	24DI/16DO
6. Максимальное количество модулей расширения							
2	7	7	7	0	2	7	7
7. Отображение процесса ввода/вывода, бит							
64DI/64DO	64DI/64DO	64DI/64DO	64DI/64DO	128DI/128DO	128DI/128DO	128DI/128DO	128DI/128DO
8. Аналоговые входы/выходы (расширение)							
16AI/16AQ	16AI/16AQ	16AI/16AQ	16AI/16AQ	нет	16AI/16AQ	32AI/32AQ	32AI/32AQ
9. Время исполнения булевых операций, мкс/оп							
1,2	0,8	0,8	0,8	0,37	0,37	0,37	0,37
10. Таймеры/счетчики							
64/64	128/128	256/256	256/256	256/256	256/256	256/256	256/256
11. Быстрые счетчики							
1	3	3	3	4	4	6	6
12. Быстрые выходы							
-	2	2	2	2	2	2	2
13. Коммуникационные прерывания (передача/прием)							
1/1	1/1	1/2	2/4	1/2	1/2	1/2	2/4
14. Прерывания, управляемые временем							
1	2	2	2	2	2	2	2

15. Аппаратные прерывания							
1	4	4	4	4	4	4	4

Таблица 2

Модули расширения с цифровыми входами и выходами

Модуль	Описание		Заказной номер
EM221	DI8 x DC24V	8 цифровых входов на 24 В постоянного тока	6ES7 221-1BF20-0XA0
EM222	DO8 x DC24V	8 цифровых выходов на 24 В постоянного тока	6ES7 222-1BF20-0XA0
EM222	DO8 x Rly	8 релейных выходов на 24 В постоянного тока	6ES7 222-1HF20-0XA0
EM223	DI4/DO4 x DC24V	4 входа и 4 выхода на 24 В постоянного тока	6ES7 223-1BF20-0XA0
EM223	DI4/DO4 x DC24 V/Rly	4 входа и 4 релейных выхода на 24 В постоянного тока	6ES7 223-1HF20-0XA0
EM223	DI8/DO8 x DC24V	8 входов и 8 выходов на 24 В постоянного тока	6ES7 223-1BH20-0XA0
EM223	DI8/DO8 x DC24V/Rly	8 входов и 8 релейных выходов на 24 В постоянного тока	6ES7 223-1PH20-0XA0
EM223	DI16/DO16 x DC24V	16 входов и 16 выходов на 24 В постоянного тока	6ES7 223-1BL20-0XA0
EM223	DI16/DO16 x DC24V/Rly	16 входов и 16 релейных выходов на 24 В постоянного тока	6ES7 223-1PL20-0XA0

Таблица 3

Модули расширения с аналоговыми входами и выходами

Модуль	Описание		Заказной номер
EM 231	AI4 x 12 битов	Аналоговые входы на 4 канала по 12 бит	6ES7 231-0HC20-0XA0
EM 232	AQ2 x 12 битов	Аналоговых выходы на 2 канала по 12 бит	6ES7 232-0H B20-0XA0
EM 235	AI4/AQ1 x 12 битов	4 канала AI и 1 канал AQ по 12 бит	6ES7 235-0KD20-0XA0

ПРОГРАММНЫЕ СРЕДСТВА

Машинный цикл CPU состоит из 5 частей:

- 1) Опрос состояния физических входов и его запись в соответствующей ячейке области памяти – сбор информации об объекте управления.
- 2) Выполнение программы с начала до конца по принципу «слева направо, сверху вниз» - отработка программы управления.
- 3) Обработка коммуникационных прерываний (если они есть).
- 4) Самодиагностика CPU.
- 5) Установка физических выходов CPU в состояния, соответствующие рассчитанным в программе значениям – выдача управляющих сигналов.

3 и 4 части не зависят от пользователя.

Редакторы в STEP7 MicroWin

Программный пакет STEP7 Micro/Win работает с тремя редакторами программ:

LAD - Ladder Logic - редактор контактного плана - позволяет формировать программы, имеющие визуальное сходство с релейно-контакторными

схемами (рис. 4).

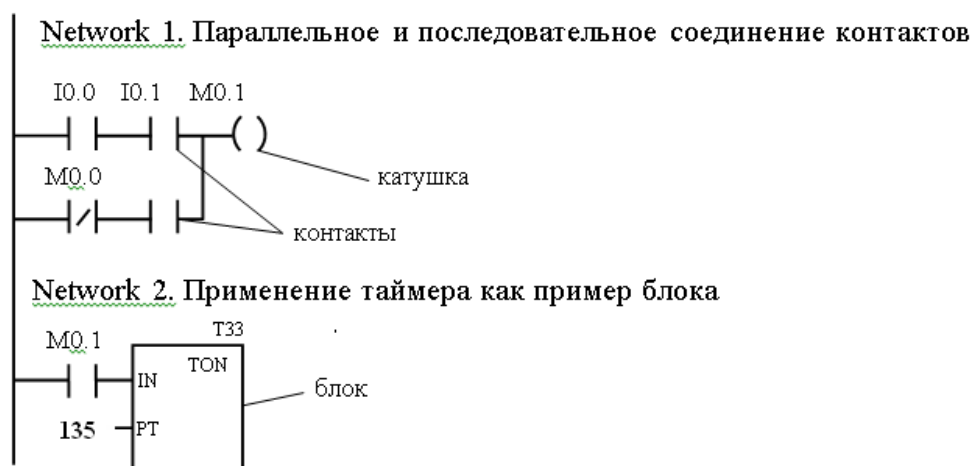


Рис. 4. Пример программы в редакторе контактных планов

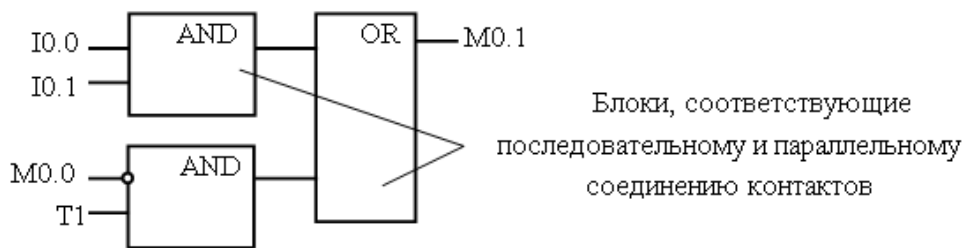
Программа подразделяется на малые, легко понимаемые сегменты, или цепи Networks. Программа выполняется по «цепям» слева направо сверху вниз по программе. По достижении конца программы СРУ снова начинает ее выполнение с начала в следующем машинном цикле (МЦ).

Данный редактор работает с тремя видами графических символов:

- 1) контакты - представляют логические состояния входов ПЛК, соответствующих выключателям, кнопкам, внутренним маркерам программы;
- 2) катушки - представляют логические результаты на выходах ПЛК, к которым подключены исполнительные объекты (лампы, пускатели электродвигателей), либо состояние внутренних маркеров;
- 3) блоки - дополнительные команды такие, как таймеры, счетчики, математические команды.

FBD - Function Block Diagram - редактор функциональных блоков- позволяет формировать программы, имеющие визуальное сходство с логическими схемами (рис. 5).

Network 1. Параллельное и последовательное соединение контактов



Network 2. Применение таймера как примера блока

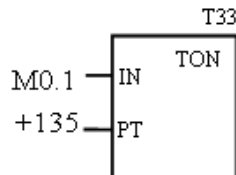


Рис. 5. Пример программы в редакторе функциональных блоков

В нем не используются ни контакты, ни катушки, как в редакторе LAD, но имеются эквивалентные им команды в форме блоков. Так, последовательному соединению контактов в LAD здесь соответствует подача сигналов на вход блока логического умножения AND. Нормально замкнутому контакту в LAD соответствует инверсный вход блока в FBD, параллельному соединению - логическое сложение сигналов на блоке OR.

STL - Statement List - Список команд – визуально напоминает программу на ассемблере. Позволяет создавать управляющие программы в текстовом виде на «родном» языке CPU, состоящем из мнемоник команд (рис. 6), а не в графическом редакторе, где должны применяться некоторые ограничения для правильного начертания схемы.

```
Network 1. Параллельное и последовательное соединение контактов
LD  I0.0      // Загрузка вершины стека из I0.0
A   I0.1      // Логическое умножение I0.0 & I0.1
LDN M0.0     // Загрузка вершины стека величиной M0.0
A   T1        // Логическое умножение вершины стека на T1
OLD                                // Логическая ИЛИ
=   M0.1     // Вершина стека копируется в M0.1

Network 2. Примененне таймера как примера блока
LD  M0.1     // Загрузка вершины стека в M0.1
TON T33,+135 // Таймер T33 с PT= 135
```

Рис. 6. Пример программы в редакторе списка команд

Переход из одного редактора в другой осуществляется с помощью меню «Вид». Из редакторов LAD и FBD можно перейти в редактор STL автоматически, а программа, написанная в STL, не всегда может быть переведена на язык контактных или функциональных планов. Редактор списка команд STL является предпочтительным для более опытных пользователей, а

FBD и LAD - для начинающих.

Константы

- 1) Десятичные – пишутся без всяких дополнительных обозначений.
Пример: **127; 255.**
- 2) Шестнадцатеричные константы обозначаются символами **16#**.
Пример: **16#58 16#9A 16#BF.**
- 3) Двоичные константы обозначаются символами **2#**, при этом пишутся только значащие цифры.
Пример: **2#10111.**

Форматы представления данных

Шина данных CPU 32 разрядная, поэтому возможны 4 формата обращения (рис. 7):

- 1) Формат двойного слова: обращение сразу к 32 разрядам числа.
- 2) Формат слова: обращение к 16-разрядному числу.
- 3) Формат байта: обращение к 8-разрядному числу.
- 4) Формат бита: обращение к 1-разрядной информации.

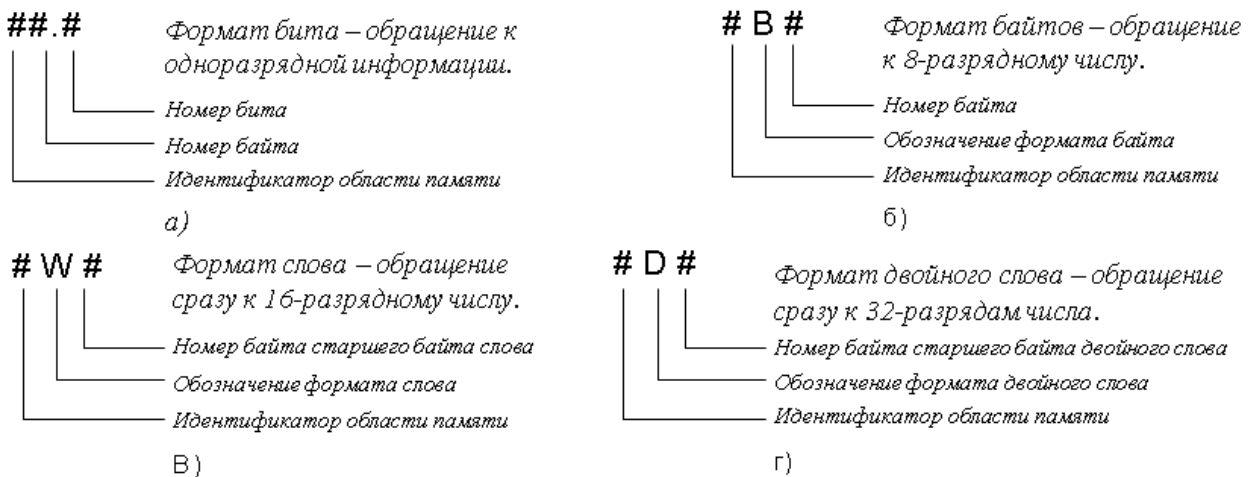


Рис. 7. Форматы данных в ПЛК

Идентификаторы областей памяти с примерами адресации приведены в таблице 4. Соотношение между различными форматами иллюстрирует рис. 8.

Таблица 4

Идентификаторы областей памяти

Идентификатор области памяти	Назначение области памяти	Форматы адресации	Примеры адресации	Примечание
I	Отображение информации на входах	Все	I0.0, IB4, IWO, ID0	
Q	Отображение информации на выходах	Все	Q0.0, QB4, QWO, QD0	
V	Память переменных	Все	V1.0, VB40,	

			VWO, VD100	
M	Внутренние маркеры	Все	MO.O, MB4, MW0, MEIO	
S	Шаговое управление	Все	S0.0, SB4, SWO, SD0	
SM	Специальные маркеры	Все	SMO.O, SMB4, SMWO, SMD0	
T	Таймеры	Бит, слово	T32	Формат - по контексту
C	Счетчики	Бит, слово	CO	Формат - по контексту
AI	Аналоговые входы	Слово	AIW0	Только четные адреса
AQ	Аналоговые выходы	Слово	AQW4	Только четные адреса
AC	Аккумуляторы	Все, кроме бита	ACO	Формат - по контексту
HC	Быстрые счетчики	Двойное слово	HC1	Обращение к текущему значению

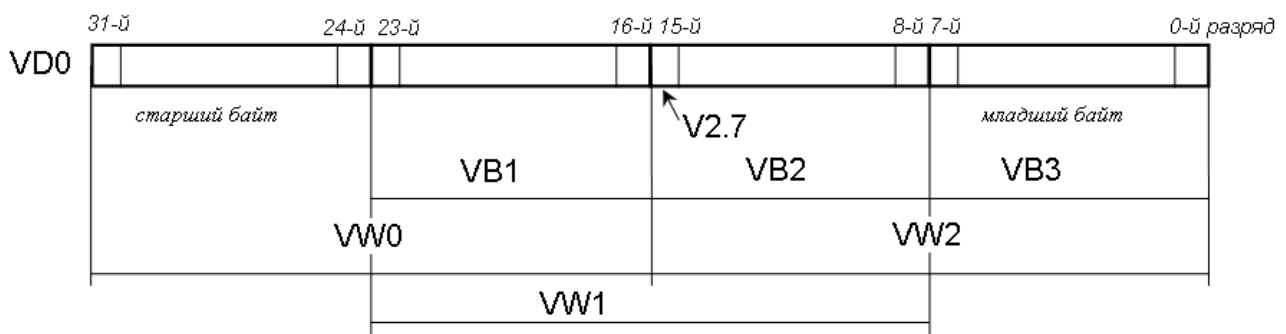


Рис. 8. Соотношение различных форматов ПЛК между собой
Области памяти CPU

Область отображения информации на входах I – обновляется в начале каждого МЦ CPU. Её можно представить как матрицу запоминающих элементов с шириной в 8 бит и длиной, равной $\frac{N}{8}$. Здесь N – количество бит данной области памяти. Например, для модуля S7-212 это количество N=64. Поэтому область I для него можно представить в виде матрицы 8×8 (рис. 9). При этом адреса физических входов CPU занимают самые младшие адреса. У модуля S7-221 шесть физических входов с адресами I0.0 ... I0.5 (показаны серым цветом).

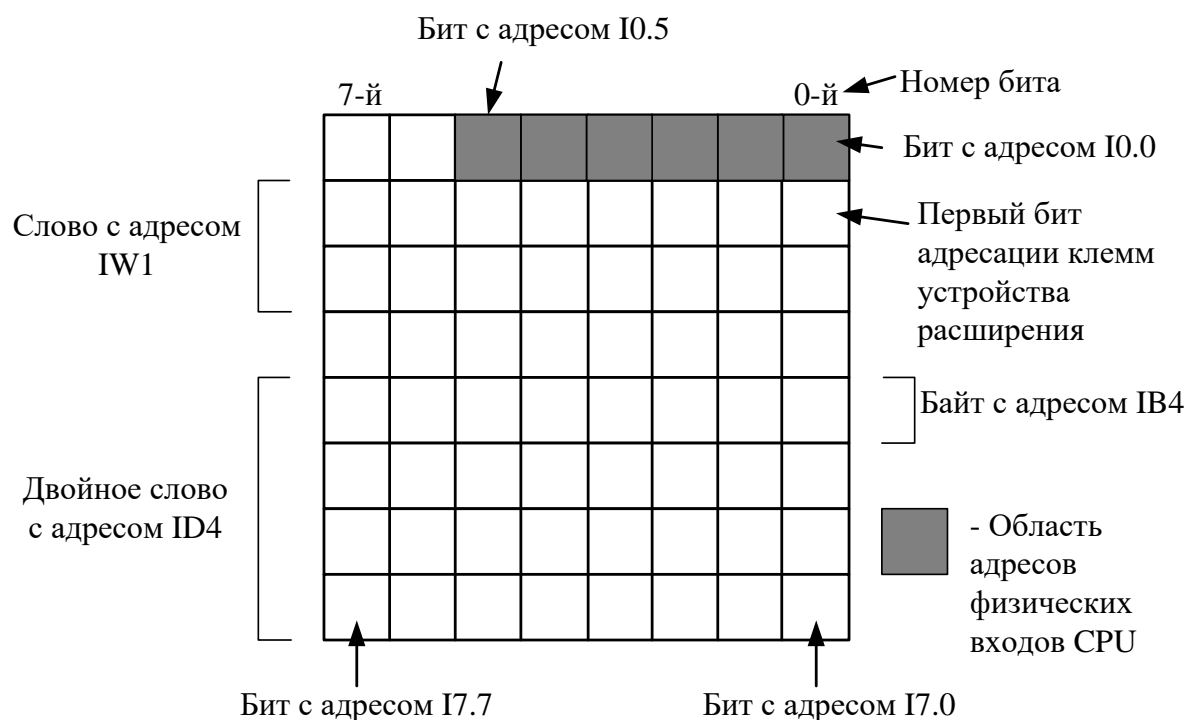


Рис. 9. Область отображения информации на входах для CPU S7-221

Область отображения информации на выходах Q – обновляется во второй части каждого МЦ CPU, а в пятой, последней, части МЦ информация из этой области передается на соответствующие физические выходы CPU.

Адресация физических выходов CPU аналогична адресации входов (рис. 10). Адресация клемм устройств расширения с цифровыми входами и выходами осуществляется с нулевого бита следующего свободного байта (рис. 9 и 10). Например, для S7-221 это байты IB1 и QB1.

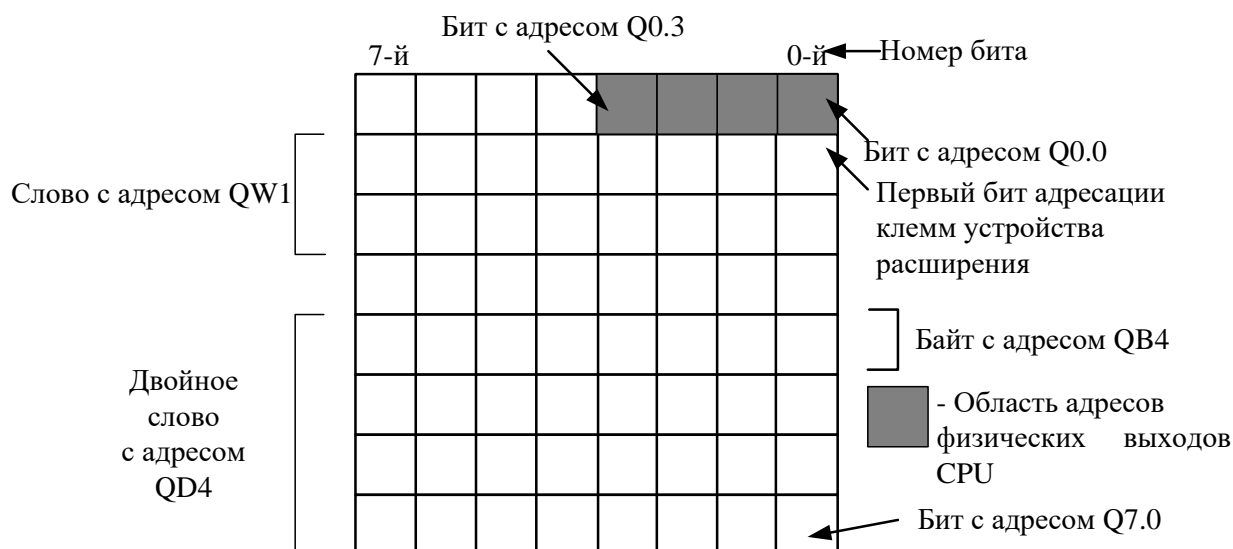


Рис. 1-2. Область отображения информации на выходах для CPU S7-221

Память переменных V – применяется для хранения промежуточных результатов операций, для работы с таблицами, блоками данных, для организации специальных функций CPU.

Маркеры M – в битовом формате могут использоваться в качестве управляющих реле для хранения промежуточного состояния операции или другой управляющей информации, а также для хранения промежуточных результатов операций в соответствующих больших форматах.

Память управления последовательностью операций S – S-биты используются для объединения частей программы в программные сегменты, воспринимаемые ПЛК как единый шаг программы. Такой шаг выполняется, если S-бит установлен и игнорируется, если S-бит сброшен.

Специальные маркеры SM – предназначены для обмена информацией между CPU и программой. Их функции «жестко» связаны с адресом:

- SM0.0 – бит, всегда равный единице. Применяется для безусловного выполнения строки, т.к. строка не может начинаться с блока или катушки.
- SM0.0 – бит, равный единице в первом машинном цикле. Применяется для однократного выполнения операций (например, начальная установка параметров);

- признаки результатов операций (таблица 5);
- *признаки ошибок* при работе составляющих CPU (таблица 6);
- *управляющие биты, биты состояния и параметры* для специальных функций CPU (быстрые счетчики, импульсные выходы – будут приведены в соответствующих темах).

Таблица 5

Формат байта специальных маркеров SMB1

SM-биты	Описание (защищены от записи)
SM1.0	=1 при выполнении определенных команд, когда результат операции равен нулю
SM1.1	=1 при выполнении определенных команд, когда возникает переполнение или когда обнаруживается недопустимое числовое значение
SM1.2	=1, когда арифметическая операция дала отрицательный результат
SM1.3	=1, когда делается попытка деления на нуль
SM1.4	=1, когда команда «Добавить к таблице» пытается переполнить таблицу
SM1.5	=1, когда команда LIFO или FIFO пытается читать из пустой таблицы
SM1.6	=1, когда делается попытка преобразовать в двоичный код величину, не представленную в формате BCD
SM1.7	=1, когда значение в коде ASCII не может быть преобразовано в допустимое шестнадцатеричное значение

Таблица 6

Байт специальных маркеров SMB4 (SM4.0 ... SM4.7)

SM-биты	Описание (защищены от записи)
SM4.0	=1, когда переполнена очередь коммуникационных прерываний
SM4.1	=1, когда переполнена очередь прерываний от ввода
SM4.2	=1, когда переполнена очередь прерываний, управляемых временем
SM4.3	=1, когда во время выполнения обнаруживается ошибка программирования
SM4.4	Отражает состояние разрешения всех прерываний; =1, когда прерывания разрешены
SM4.5	=1, когда передатчик не работает (порт 0)
SM4.6	=1, когда передатчик не работает (порт 1)
SM4.7	=1, когда что-то делается принудительно

Таймеры T – элементы, отсчитывающие приращения времени. С каждым

таймером связаны две одинаково обозначаемые переменные (например, T62), формат которых определяется контекстом программы:

- *текущее значение* – 16-битовое целое число со знаком;
- *бит таймера* – устанавливается или сбрасывается как результат сравнения текущего (CT) и предустановленного (PT) значения. Предустановленное значение вводится как параметр таймера.

Счетчики С – элементы, подсчитывающие каждый нарастающий фронт на входе (входах) счетчика. Как у таймера, у каждого счетчика есть две одинаково адресуемые (например, C24) переменные:

- *текущее значение* – 16-битовое целое число со знаком, хранящее накопленное счетчиком значение;
- *бит счетчика* – устанавливается или сбрасывается как результат сравнения текущего (CV) и предустановленного (PV) значения. Предустановленное значение вводится как параметр счетчика.

Область аналоговых входов AI – служит для хранения 16-разрядных результатов АЦП, поступающих с устройства расширения. Используются только четные адреса в формате слова (AIW0, AIW2 и т.д.). Данная область доступна только для чтения.

Область аналоговых выходов AQ – служит для хранения 16-разрядных кодов, предназначенных для последующего ЦАП в соответствующем блоке расширения. Используются только четные адреса в формате слова (AQW0, AQW2 и т.д.)). Данная область доступна только для записи.

Аккумуляторы AC – это регистры для хранения промежуточных данных, для передачи параметров в подпрограммы и из них. В CPU имеется четыре 32-разрядных аккумулятора с адресами AC0, AC1, AC2 и AC3, к которым возможно обращение в формате двойного слова, слова (два МБ) и байта (самый младший). Формат данных определяется контекстом программы.

Текущее значение быстрых счетчиков HC.