

сообщения  
объединенного  
института  
ядерных  
исследований  
дубна

3827/2-80

11/8-80  
P11-80-384

Х.Лайх

ДВОИЧНО-ОРИЕНТИРОВАННЫЕ ПРОТОКОЛЫ  
ПЕРЕДАЧИ ИНФОРМАЦИИ

1980

Применение вычислительной техники в последние годы характеризуется переходом от использования отдельно стоящих ЭВМ к созданию сложных вычислительных систем. Вначале это были главным образом иерархические системы с централизованным управлением. В таких системах пользователи осуществляют доступ к ресурсам центральной ЭВМ посредством терминальных устройств, подключенных к ЭВМ через каналы связи и коммутационные устройства.

Появление сравнительно мощных и недорогих мини-ЭВМ обусловило переход от принципа строго централизованного управления в вычислительных системах к принципу "распределенного интеллекта". Это означает, что функции, которые раньше выполнялись центральной ЭВМ, теперь возлагаются на "интеллектуальные" блоки системы. Разработка и широкое внедрение дешевых микро-ЭВМ усиливают эту тенденцию.

В настоящее время наиболее развитой формой вычислительных систем являются сети ЭВМ, в которых территориально распределенные вычислительные мощности (средства) логически и физически связаны между собой коммуникационными каналами. Подключение различных терминалов к сети обеспечивает пользователям возможность доступа ко всем ресурсам такой системы независимо от конкретного географического расположения ресурса.

Создание сложных вычислительных систем, нашедших широкое применение в ядерно-физических центрах /1/, связано с большими затратами времени и средств на разработку общих правил обмена информацией в системе и на создание соответствующего математического обеспечения. Очевидно, что в вычислительных сетях все их компо-



ненты должны обеспечивать единый способ поведения. Набор правил, которые определяют электрические и (или) функциональные параметры процесса обмена информацией между двумя компонентами системы (физическими или логическими) называют протоколом. Существует множество таких протоколов, но все они практически несовместимы друг с другом.

Поскольку в будущем роль вычислительных сетей будет все больше возрастать, одной из наиболее актуальных задач является разработка и внедрение стандартных протоколов для всех возможных связей в системе, с тем чтобы уменьшить затраты на проектирование, реализацию и эксплуатацию вычислительных сетей.

К.Бекманн, председатель двух подкомиссий международной организации по стандартизации (ISO), выразил это так: "Сегодня нужны стандартные интерфейсы и протоколы, которые позволили бы любому человеку, любому терминалу, любому физическому процессу и любой ЭВМ, использующим эти стандарты, обмениваться информацией" /2/. Исходя из этого американская организация по стандартизации ANSI и организация ISO приступили к разработке единой системы протоколов для управления обменом информацией в сетях ЭВМ. Предложенная ими модель предусматривает иерархию протоколов для осуществления этого обмена /2/.

Протокол самого низкого уровня, который согласно модели называется физическим уровнем, определяет механические, электрические и функциональные характеристики, требуемые для работы физической линии связи между двумя компонентами сети. В настоящее время существуют некоторые общепринятые стандарты этого уровня, например, RS-422/RS-423 и RS-232-C, а также стандарт X.21, рекомендуемый МКККТ (CCITT) /3/.

Следующий уровень, называемый уровнем информационного канала, обеспечивает обмен информацией между компонентами сети, именуемыми в дальнейшем станциями. Информация, передаваемая от одной станции в другую, называется сообщением. Сообщение, в свою очередь, состоит из отдельных блоков. Протокол управления информационным каналом определяет правила передачи блоков через информационный канал, добавляет адреса к блокам или декодирует их, управляет потоком информации в канале, а также обнаруживает и исправляет ошибки, возникающие при передаче.

К протоколам этого уровня принадлежат такие известные протоколы, как BSC фирмы IBM и BSCMP фирмы DEC, а также недавно разработанные новые типы протоколов:

HDLC-High Level Data Link Control, рекомендуемый ISO /3/;  
SDLC-Synchronous Data Link Control, фирмы IBM /4/;  
{ ADCCP-Advanced Data Communications Control Procedure, рекомендуемый ANSI /5,7/.

Главным достоинством протоколов HDLC, SDLC и ADCCP является то, что они кодонезависимы. Они уже широко применяются в новых системах, например, HDLC - фирмами Sperry Univac, Honeywell, CDC (в системах Cyber 170), Siemens, а SDLC - фирмой IBM.

На всех последующих уровнях (всего в модели предусмотрено 7 уровней) пока еще отсутствуют общепринятые протоколы и каждый изготовитель соответствующего оборудования разрабатывает и реализует свой специфический вариант.

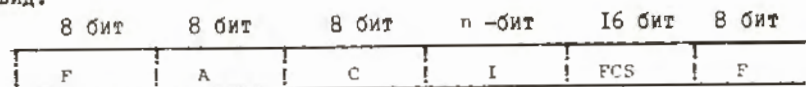
В дальнейшем на примере протокола HDLC - "высокоуровневой процедуры управления линией передачи данных" - более подробно описывается суть новых протоколов, которые в литературе часто называют двоично-ориентированными протоколами (bit-oriented protocols). Поскольку в соответствующих работах на русском языке пока еще нет установившейся терминологии, в данном обзоре используется терминология, принятая в работе /8/.

Протоколы управления информационным каналом SDLC, HDLC и ADCCP выполняют следующие функции:

1. Передачу сообщений в виде отдельных блоков определенной структуры.
2. Установление и окончание логической связи между двумя станциями, связанными через информационный канал.  
Управление передачей и приемом блоков.
3. Обеспечение достоверности передаваемой информации путем применения эффективных методов кодирования информации.
4. Обеспечение информационной прозрачности, т.е. независимости протокола от конкретных кодов в передаваемой информации.
5. Первоначальную загрузку вторичных станций\* первичной.

\* Между станциями, обменивающимися информацией, устанавливается асимметрия: одна станция, которая управляет обменом, является "первичной", а все остальные - "вторичными". Дальше они обозначаются соответственно ПС и ВС.

В протоколе HDLC общая структура блока, который служит для передачи как данных, так и управляющей информации, имеет вид:



F - Flag

Каждый блок ограничивается с обеих сторон специальными группами битов, которые называются "флагом" и имеют вид 0 1 1 1 1 1 0. Если блоки передаются без интервала, то между двумя соседними блоками ставится только один флаг.

A - Address -

адресное поле, в котором всегда указывается адрес вторичной станции, независимо от того, является ли она отправителем или получателем информации.

C - Control -

поле управления (расшифровка дана ниже).

I - Information -

поле, в котором размещается собственно информация, подлежащая передаче. Содержимое этого поля может состоять из произвольного числа битов \* и может быть представлено в любом коде (информационная прозрачность). Чтобы избежать детектирования кода 0111110 (флаг) в A-, C-, I- и FCS-полях, передающая станция после каждой группы из пяти последовательно идущих единиц вставляет нулевой бит, а принимающая станция его автоматически удаляет (bit-stuffing).

FCS - Frame Checking Sequence -

контрольная последовательность блока. Она вычисляется для A-, C-, I- полей путем деления содержимого этих полей на генераторный полином  $x^{16} + x^{12} + x^5 + 1$ , а инвертированный остаток от деления

\* У протоколов SDLC и ADCCP длина этого поля произвольна, но кратна 8 битам.

ния записывается в FCS. На практике генерирование FCS при передаче или проверка при приеме осуществляются схемно с помощью сдвигового регистра с обратными связями.

Протокол HDLC различает 3 типа блоков /3/:

I-блоки: блоки для передачи данных.

S-блоки: блоки для передачи управляющей информации (супервизорные команды/ответы), например, подтверждение правильно полученных блоков, запрос на повторение блоков и др.

U-блоки: нумерованные блоки, они используются для дополнительных функций управления.

В отличие от I-блоков, блоки типа S и U не содержат I-поля.

В таблице I показаны формат и содержание поля управления протокола HDLC.

Таблица I

Тип блока	Мнемоника команды/ответа	Формат C-поля разряды							
		I	2	3	4	5	6	7	8
I - блоки	I	0	N(S)			P/F	N(R)		
S - блоки	RR	I	0	0	0	P/F	N(R)		
	RNR	I	0	I	0	P/F	N(R)		
	REF	I	0	0	I	P/F	N(R)		
ненумерованные блоки (U-блоки)	SNRM	I	I	0	0	P	0	0	I
	SARM	I	I	I	I	P	0	0	0
	DTSC	I	I	0	0	P	0	I	0
	UA	I	I	0	0	F	I	I	0
	CMDB	I	I	I	0	F	0	0	I

N(S) - номер блока в передаваемой последовательности, меняющийся по модулю 8.

N(R) - ожидаемый номер следующего принимаемого блока.

N(R) увеличивается на единицу после приема неискаженного блока, также меняется по модулю 8.



$P/F$  - бит "вызов /конечный";  $P=1$  означает при этом, что данный блок является последним (или единственным) в текущем сообщении, которое ПС посылает в ВС, и ПС ожидает ответ от ВС. Значение  $P=1$  в блоке, отправленном ВС, говорит о том, что данный блок является последним в сообщении вторичной станции, передаваемом в ПС.

Так как  $N(S)$  и  $N(R)$  меняются по модулю 8, то одна станция может передать другой станции максимум 8 блоков, прежде чем получит от приемника подтверждение о безошибочном приеме. В протоколе HDSSR предусмотрено расширение C-поля до 16 бит, что позволяет работать с модулем I28. Это очень выгодно в полудуплексном режиме работы, поскольку в этом случае значительно уменьшается число переключений линии связи по сравнению со старыми протоколами (например, VSC, в котором подтверждается прием каждого блока).

Блоки, передаваемые первичной станцией во вторичную, называются командными, а блоки, передаваемые вторичными станциями, - ответными. В случае командного блока всегда  $P/F=P$ , а в случае ответного блока -  $P/F=F$ . При этом  $P$  означает "вызов" (poll), а  $F$  - "конечный" (final).

Приведенные в таблице I команды и ответы выполняют следующие функции:

I - Information -  
передача данных через линию связи.

RR - Receive Ready

Станция посылает RR с целью:

1. Сообщить другой станции, что она готова принимать I-блоки;
2. Подтвердить безошибочный прием I-блоков вплоть до номера  $N(R)-I$ , где  $N(R)$  есть значение, стоящее в C-поле блока RR.

REJ - Reject

Отказ REJ используется станцией для того, чтобы в случае помех потребовать повторной передачи I-блоков, начиная с блока с номером  $N(R)$ . При этом подтверждается безошибочный прием блоков вплоть до  $N(R)-I$ .

RND-Receive Not Ready

Неготовность к приему RND используется первичной или вторичной станциями для сообщения собственного состояния занятости, т.е. отсутствия возможности принимать последующие I-блоки. Прием блоков вплоть до  $N(R)-I$  подтверждается. Отмена действия RNR достигается передачей блоков UA, RR или REJ.

CMDR-Command Reject

CMDR используется вторичной станцией для сообщения первичной, что переданная ей команда отвергнута по одной из следующих причин:

1. Код переданной команды недопустим;
2. Принят I-блок, длина I-поля которого превосходит имеющийся размер буфера.
3. Значение счетчика  $N(R)$ , переданное в данном блоке, неверно (например, когда  $N(R)$  указывает на I-блок, который был передан и подтвержден ранее).

UA-Unnumbered Acknowledge -

нумерованное подтверждение. Вторичная станция использует UA для подтверждения получения команд SNRM, SARM и DISC. Принятые ВС команды не выполняются до тех пор, пока не будет передана команда UA.

SARM-Set Asynchronous Response Mode -

установление режима "асинхронного ответа". ВС, принимая эту команду, включает у себя режим "асинхронного ответа" и отвечает блоком UA.

DISC-Disconnect

Команда DISC служит для сброса ранее установленного режима работы в ВС. Подтверждение выполнения команды осуществляется путем пересылки блока UA:

SNRM-Set Normal Response Mode

После приема этой команды ВС устанавливает у себя режим "нормального ответа" и отвечает на команду пересылкой UA.

Протокол HDLC допускает два следующих режима работы:

NRM-Normal Response Mode

В режиме "нормального ответа" вторичная станция может начать передачу только после разрешения от ПС в виде блока, у которого P=1.

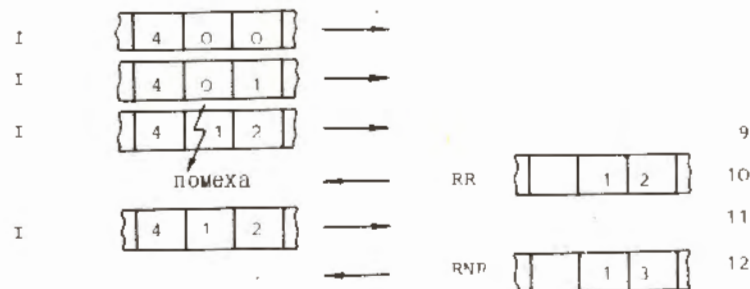
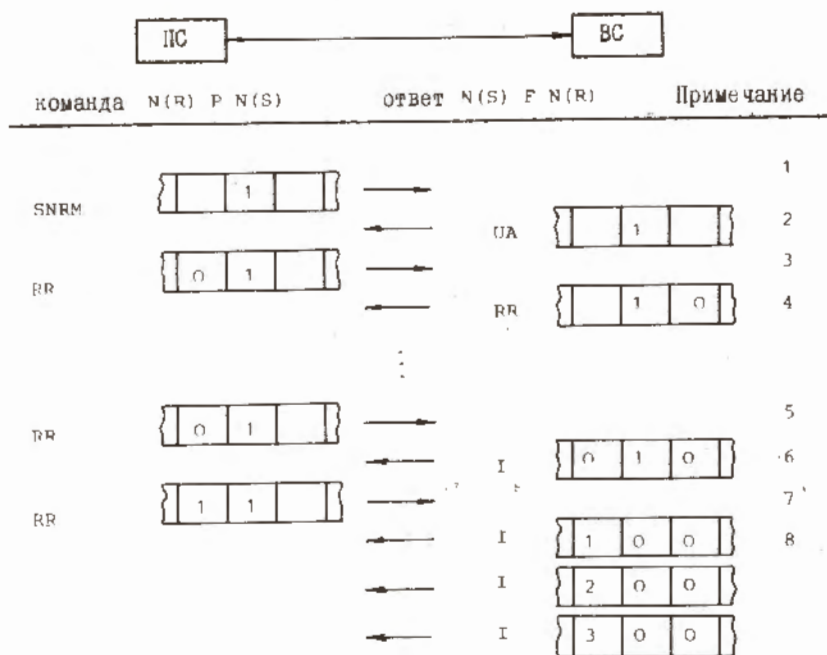
ARM-Asynchronous Response Mode

В режиме "асинхронного ответа" вторичная станция может начать передачу блоков независимо от получения вызова от первичной станции.

Как видно из таблицы I, имеется еще целый ряд свободных комбинаций, которые при дальнейшем совершенствовании протокола могут использоваться для определения дополнительных функций.

Для пояснения механизма передачи информации рассмотрим два примера.

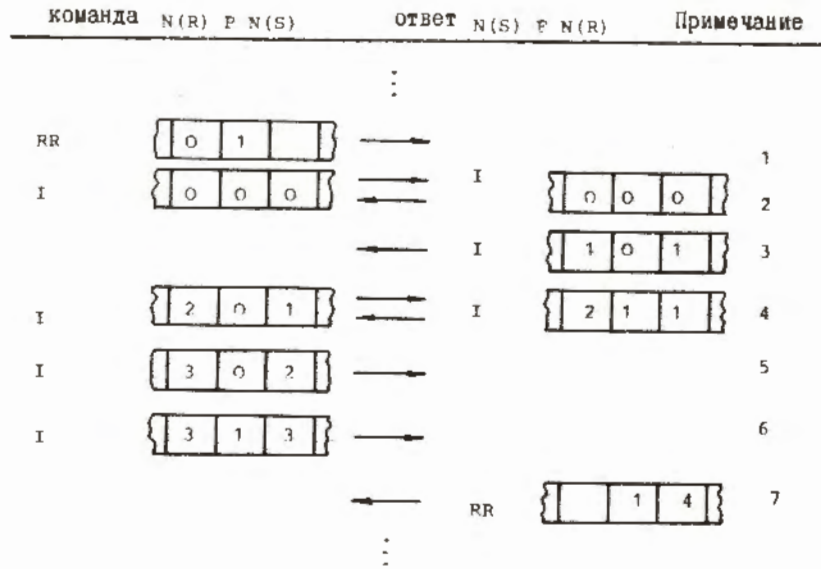
Пример 1: Канал связи типа "point to point" - полудуплексный режим обмена



Пояснения:

- ПС - первичная станция.
- ВС - вторичная станция.
- 1. Инициализация ВС (установление NRM).
- 2. Подтверждение команды SNRM.
- 3. ПС приказывает ВС: "посылай данные".
- 4. Ответ ВС: "ничего посылать; готова принимать данные".
- 5. ПС в ВС: "посылай данные".
- 6. ВС посылает I-блок в ПС.
- 7. Ответ от ПС: получен один I-блок, готова принимать следующую информацию.
- 8. ВС посылает подряд 3 информационных блока.
- 9. ПС посылает 3 I-блока в ВС, при этом последний блок искажается. N(R)=4 сообщает ВС о том, что ПС приняла от ВС блоки вплоть до номера N(R)-1=3.
- 10. ВС сообщает в ПС: приняла блоки вплоть до номера один. Готова к приему.
- 11. ПС повторно посылает последний I-блок.
- 12. ВС сообщает в ПС: приняла блоки вплоть до номера два, больше не могу принимать информацию.

Пример 2: Канал типа "point to point" - дуплексный режим обмена



**Пояснения:**

1. PC приказывает BC: "посылай данные".
2. BC пересылает в PC первый I-блок, одновременно PC пересылает в BC один I-блок.
3. BC пересылает второй блок и подтверждает прием первого блока от PC.
4. PC пересылает второй I-блок в BC и при этом подтверждает прием 2-х блоков, полученных от BC. BC пересылает 3-й блок и подтверждает прием одного блока от PC.
5. PC пересылает 3-й и 4-й I-блоки в BC и получает подтверждение приема 3-х блоков от BC.
6. Четвертый блок является последним в данной передаче.
7. BC подтверждает прием 4-х блоков от PC и одновременно сообщает, что она готова дальше принимать информацию.

Все вышеизложенное может произвести впечатление, что реализация двоично-ориентированных протоколов обмена информацией связана со значительными трудностями. Однако уже сегодня серийно выпускается целый ряд больших интегральных схем (БИС), специально предназначенных для реализации таких протоколов /5,6,7/. В таблице 2 приведены некоторые характеристики указанных БИС.

Таблица 2

Характеристика	БИС					
	(изготовитель и обозначение)					
	Signetics	Zilog	Motorola	Intel	Western Digital	
	2652	SIO	6854	8273	SD1933	
	1	2	3	4	5	6
Макс. скорость передачи Кбит/сек	1000 и 2000	550 и 880	660 и 1000	64	1000	
Разрядность шины данных; бит	8 или 16	8	8	8	8	
Выводы управления модемом	нет	4	4	10	6	
Наличие FIFO -памяти у передатчика	нет	2 байта	2 байта	нет	нет	
Наличие FIFO -памяти у приемника	нет	нет	2 байта	нет	нет	
Прерывание отдельно от передатчика и от приемника	да	нет	нет	да	нет	
Автоматическое распознавание адреса вторичной станции	да	да	нет	да	да	
Распознавание глобального адреса	да	да	нет	да	да	
Возможность расширения А- и С-полей	нет	нет	да	нет	да	



1	2	3	4	5	6
Протоколы, которые можно реализовать с помощью этой БИС	SDLC, DDCMP, BSC	SDLC, HDLC, BSC	SDLC, HDLC, ADCCP	SDLC, HDLC	SDLC, HDLC, ADCCP

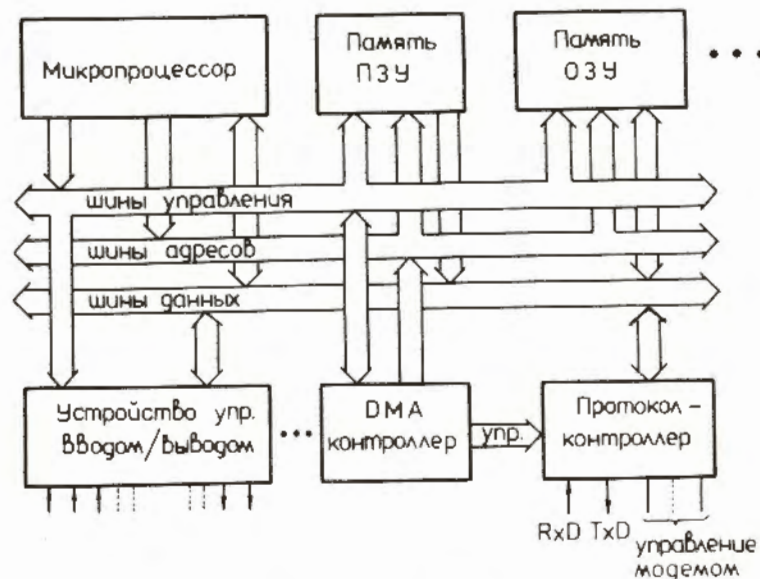
Перечисленные в таблице 2 БИС схемно выполняют следующие функции протокола:

1. Автоматическое генерирование флаг-комбинаций блока при передаче и выделение флага после приема.
2. Преобразование параллельного кода в последовательный и наоборот.
3. Добавление и удаление нулевых битов при передаче/приеме/информации (см. описание I-поля).
4. Генерирование контрольной последовательности (FCS) при передаче и проверка FCS при приеме.
5. Автоматическое распознавание прерывания передачи \*) (abort). Генерирование последовательности, которую приемник расшифровывает как "abort".
6. Автоматическое обнаружение состояния ожидания (idle state)\* линии и генерирование последовательности, которая распознается как состояние ожидания линии.

Таким образом, за программным обеспечением остаются задачи генерирования и анализа C-поля и частично A-поля, а также подготовки и обработки самих данных.

В качестве примера на рисунке показана структура типичной микропроцессорной системы, содержащей БИС для реализации протокола. Такие системы применяют в некоторых терминалах (например, в банковских терминалах фирмы Siemens, которые используют протокол HDLC) и в других современных устройствах. На рисунке протокол-контроллер подключен к общей магистрали через устройство прямого доступа к памяти (DMA), что позволяет работать при больших скоростях передачи данных, например, при 48.000 бит/сек. Возможно и прямое подключение протокола-контроллера к магистрали.

\*) Прерывание передачи: от 7 до 14 последовательно идущих единиц.  
\*\*) Состояние ожидания: 15 и больше последовательно идущих единиц.



Микропроцессорная система с БИС для реализации протокола

В заключение отметим основные преимущества двоично-ориентированных протоколов по сравнению с кодово-ориентированными (например, VSC):

1. Использование единого формата (блок) для всех видов передаваемой информации и способов передачи данных.
2. Использование всего лишь 3-х типов блоков: I-, S- и C-блоков.
3. Длина I-поля не фиксирована. В частности, ее можно установить, исходя из статистики помех в канале связи с тем, чтобы минимизировать число повторений блоков в случае помех.
4. Информационная прозрачность (любая комбинация битов может быть содержанием I-поля в блоке). Тем самым отпадает необходимость в кодировании/декодировании/при передаче/приеме/данных.
5. Применение такой контрольной последовательности (FCS), которая по сравнению с контролем по четности или контрольной



суммой значительно повышает достоверность переданной информации. В отличие от старых протоколов (вс, например) FCS включает не только сами данные (г-поле), но и адресную и управляющую информацию (С- и А-поля). Кроме того, все блоки независимо от их типа содержат FCS.

6. Повышение эффективности работы. Поскольку нет необходимости в подтверждении получения каждого блока в отдельности сразу после его приема, то сильно уменьшается число переключений линии, что дает большой выигрыш во времени.

В новых стандартах предусмотрена также возможность расширения А- и С-полей, что еще больше повышает эффективность; это особенно заметно при использовании линий с большой задержкой, как это имеет место у спутниковой связи.

### Литература

1. Н.Е.Дэвис "The Impact of Computer Networks on Experimental Physics".

Материалы Второго всесоюзного семинара по обработке физической информации. Ереван, сент.1977. Изд-во Ереванского физического института. Ереван, 1978.

2. А.Дурниак: "Special Report: New Networks tie down distributed processing concepts."

Electronics, No 25, 1978, pp. 107-122.

3. Draft Recommendation X.25. APVI-No55-E, pp.80-144; Geneva, 1976.

4. IBM Synchronous Data Link Control. General Information.

GA 27-3093-1, IBM Corp. May 1975.

5. А.Д.Вейсдергер: "Data-link control chips: bringing order to data protocol". Electronics, June 8, 1978, pp.104-112.

6. "Advanced Data Link Controller (ADLC): MC 6854".

Advanced Information AOI-432, Motorola Inc., 1977.

7. В.Меронек: "Data-link control chip supports all three bit-oriented protocols". Electronics, January 18, 1979, pp.137-142.

8. Бутрименко А., Секстон Дж. Протоколы в коммуникационных сетях передачи цифровой информации. Автоматика и вычислительная техника, № 6, 1978, с.56-65.

Рукопись поступила в издательский отдел  
4 июня 1980 года.