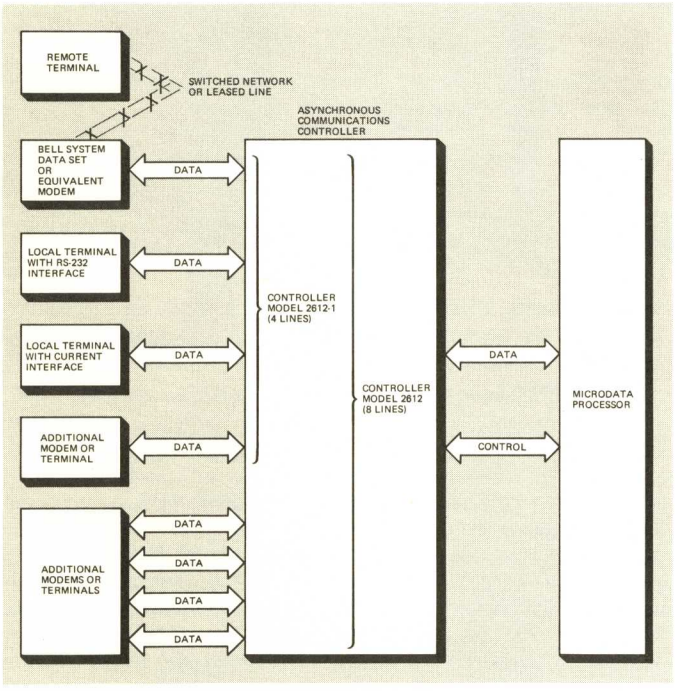
Microdata Interface

Asynchronous Communications Controller, Models 2612 and 2612-1

GENERAL DESCRIPTION

The Asynchronous Communications Controller enables the Micro 1600 series of computers to communicate with asynchronous local or remote devices. Model 2612 services eight full-duplex communication channels, and Model 2612-1 services four full-duplex channels. The interface can be installed in any I/O card slot of the Microdata processor. Connectors on the Interface board accommodate cables to terminals, to Bell System Data Sets, or to equivalent modems. The Asynchronous Communications Controller can be used with terminals having either 20-ma current-loop interfaces, or interfaces conforming to RS-232-C standards.



The flexibility of the Asynchronous Communications Controller provides the Microdata processor with many local and remote data communications options.

STANDARD FEATURES

Full Duplex

Simultaneous sending and receiving is provided for each communication channel of the Controller. Each Send and Receive subchannel operates separately, and each is double buffered so that a full character time is available for the computer program to transfer characters between the Controller and the processor.

Data Transfer Mode

Data is communicated from the Controller to a modem or terminal serially, through separate input and output serializing registers. When a full character is received, it is transferred to a buffer register until accepted by the program. Similarly, a buffer register holds an output character until it is put into a separate serializing register for transmission.

An interrupt occurs whenever a new character is placed in the receive buffer; and an interrrupt occurs whenever a character is removed from the send buffer for transmission. This double buffering technique allows over one millisecond for the program to service an interrupt on a channel handling 10-bit characters at 9600 baud.

Unique Interrupt Response Addresses

Each communication channel has two dedicated interrupt response addresses — one for the receive subchannel, and one for the send subchannel. This simplifies programming and shortens the processing time associated with handling data transfers between the processor and the eight communication channels.

Unique Baud Rates for Different Channels

Each channel has three jumper-selectable and seven switch-selectable baud rates ranging from 75 to 9600 bps.

Unique Transmission Codes for Different Channels

Transmission codes include one start bit; 5, 6, 7, or 8 jumper-selectable information bits; and 1 or 2 switch-selectable stop bits.

Parity Options

Odd parity, even parity, and parity-disable are jumper-selectable options.

Error Control

A status byte from the Asynchronous Communications Controller can be sampled by the processor to detect three types of errors in received data: parity errors, framing errors (absence of a stop bit), and overrun errors (loss of a character).

RS-232-C and Current Loop

The Asynchronous Communications Controller enables the Microdata processor to communicate with modems or terminals using either the Electronic Industries Association RS-232-C standard interface, or Teletype Corporation's 20-milliamp current-loop interface.

OPERATION

The diagram below shows how the Microdata processor communicates with terminals and modems via the Asynchronous Communications Controller.

The Controller connects to the processor via the processor's I/O bus. The bus has 8 input-data lines, 8 output-data lines, plus control, interrupt, and acknowledge lines. Standard Microdata firmware transfers information on the bus in the Programmed I/O mode using a two-byte instruction.

The first instruction byte contains the device address of the Controller (bits zero through four) and the identification of the communication channel (bits five through seven).

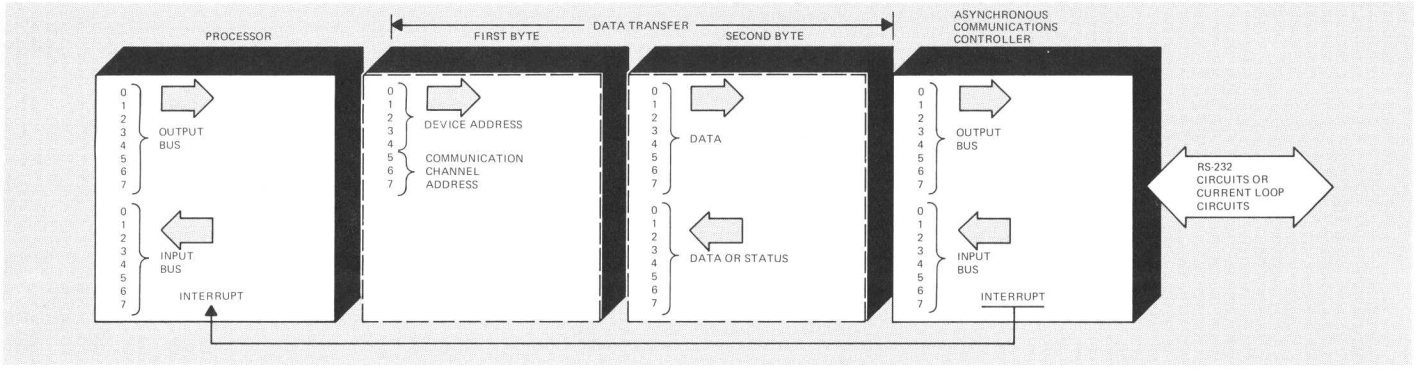
For an output operation, the second instruction byte is data. For an input operation, the second byte can be either data, or a status byte to check the received data for validity.

Each communication channel has a separate send and receive subchannel which can function independently in the full-duplex mode, with sending and receiving operations occurring simultaneously. Each channel has a unique interrupt response address for its send subchannel and for its receive subchannel.

The interrupt from the Controller is enabled automatically each time the Controller is addressed by the processor — whether for input or output. An output interrupt is generated by the Controller whenever a character is moved from a communication channel's output buffer register to the channel's output serializing register. Similarly, an input interrupt is generated by the Controller whenever a character is moved from a channel's input deserializing register to the channel's input buffer register.

When the processor inputs a byte from the Controller, the processor receives only the byte from the communication channel which caused the last interrupt. Thus, if the interrupt from the Controller is disabled by the program, or if a second interrupt occurs before the processor responds to a prior interrupt, data and status information from the prior interrupt will be lost. However, the loss of data can be detected by subsequently reading the channel's status byte (see RECEIVED DATA ERROR CONTROL DISCUSSION).

Each communication channel is independent of other channels, and each channel can have a different baud rate and code structure.



Device Address and Communication Channel Identification

A device address of hexadecimal 1B is usually assigned to the Asynchronous Communications Controller. By means of jumpers on the Controller board, device addresses 18 through 1F can be used. For a device address of 1B, output and input instructions use the following channel addresses.

Output Byte To Channel Number	Address Used With Output Byte Instruction	Address Used With Input Byte Instruction
0	1B	
1	3B	3B Data In
2	5B	5B Status In
3	7B	
4	9B	
5	BB	BB Disable Interrupts
6	DB	
7	FB	

For a device address of 1B, an instruction to input a data byte uses address 3B; and an instruction to input a status byte uses address 5B. An input command using these addresses transfers the data or status byte from the buffer register of the communication channel which generated the most current interrupt. Address BB disables the interrupt from the Controller.

When a device other than 1B is used, the above addresses must be modified accordingly. The use of I/O commands with device addresses is summarized in the COMPOSITE INSTRUCTION summary.

Received Data Interrupts

Each receive subchannel is double buffered. As data is received from a local or remote device on a communication channel, it is accepted one bit at a time and assembled in a deserializing register. When the character is complete, it is automatically transferred to a buffer register, and an interrupt is generated. Each receive subchannel has its own interrupt response address. Following are hexadecimal memory address assignments which are used when the Asynchronous Communications Controller is assigned a device address of 1B.

Receive Channel Number	Interrupt Response Address
0	170
1	172
2	174
3	176
4	178
5	17A
6	17C
7	17E

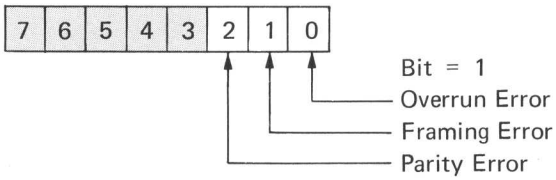
When a device address other than 1B is used, the interrupt response address is modified as follows:

Device Address	Add To Address Shown	Device Address	Subtract From Address Shown
1C	20	1A	20
1D	40	19	40
1E	60	18	60
1F	80		

The unique interrupt response addresses provide implicit identification of the communication channel, and provide an efficient means of entering routines for handling data from the channel. Executing an input-data-byte instruction (input instruction using address 3B) transfers data from the channel's buffer register to the processor, and clears the interrupt.

Received Data Error Control

The validity of received data can be checked by executing an input-status-byte instruction (input instruction using address 5B) immediately before an input-data-byte instruction. The diagram shows the bit pattern of the Status Byte.



Overrun Error. The processor did not remove the preceding character from the input buffer, and the character has been lost.

Framing Error. The stop bit was not received.

Parity Error. Odd or even parity violated. The Controller board is supplied without parity. Odd or even parity can be enabled by adding jumpers as noted in the TRANSMISSION CODE discussion.

Transmitted Data Interrupts

Each transmit subchannel is double buffered. An output instruction transfers the data from the processor to the channel's buffer register in the Controller. When the Controller transfers the character from the buffer register to a serializing register for transmission, the Controller generates an interrupt to indicate the channel is ready to accept another character. Each transmit subchannel has its own interrupt response address. Following are the hexadecimal memory addresses which are used when the Asynchronous Communications Controller is assigned a device address of 1B.

Transmit Channel Number	Interrupt Response Address
0	160
1	162
2	164
3	166
4	168
5	16A
6	16C
7	16E

When a device address other than 1B is used, the interrupt response address is modified as shown in the RECEIVED DATA INTERRUPTS summary.

Disable Interrupts

Interrupts from the Controller are enabled automatically whenever the Controller is accessed to execute an input or output instruction. The Disable-Interrupts instruction prevents the Controller from interrupting the processor, but does not disable the generation and storage within the Controller of an input interrupt or an output interrupt. The interrupts from the Controller can be disabled by using the two-byte instruction hexadecimal 31BB, provided the Controller's device address is 1B. For other device addresses, the BB portion of the instruction must be modified accordingly.

Baud Rates

Each communication channel of the Asynchronous Communications Controller can be made to operate at a unique baud rate by means of a jumper and a switch. Each jumper and switch is labeled A through J to correspond to communications channels 0 through 7, respectively.

Baud Rate	Jumper	Switch Position
75	19 to 21	No Affect
110	19 to 20	4
134.5	19 to 23	No Affect
150	19 to 20	5
300	19 to 20	6
600	19 to 22	No Affect
1200	19 to 20	7
2400	19 to 20	8
4800	19 to 20	9
9600	19 to 20	0

Any of the above baud rates can be used in conjunction with EIA-RS-232-C circuits. For 20-ma current-loop circuits, noise reduction filters are included, and data rates are limited to a maximum baud rate of 150.

Transmission Codes

Each communication channel can use a unique code comprising one start bit; 5, 6, 7, or 8 information bits; and 1 or 2 stop bits. Parity can be disabled, or can be odd or even as the eighth bit. Following are instructions for implementing the various code options with jumpers and switches. Each jumper and switch for a communication channel is labeled A through J to correspond to channels 0 through 7, respectively.

Information Bits	Jumper 1	Jumper 2
5	25 to 24	27 to 26
6	n.c.	25 to 26
7	25 to 24	n.c.
8	n.c.	n.c.

Stop Bits	Switch Position
1	1
2	2

Parity	Jumper 1	Jumper 2
Disable	n.c.	n.c.
Odd	31 to 30	33 to 32
Even	31 to 30	n.c.

Transmission Codes (Continued)

An 8-bit data byte is always transferred between the processor and the Asynchronous Communications Controller. When fewer than eight information bits are contained in the byte, they will be arranged as shown in the word diagram.

Number of Information Bits	Byte Configuration
7	7 6 5 4 3 2 1 0
6	7 6 5 4 3 2 1 0
5	7 6 5 4 3 2 1 0

Shaded areas indicate locations of non-information bits, which may be 1 or 0.

Terminal and Data Set Connectors

Each communication channel of the Asynchronous Communications Controller can be connected either to a 20-ma current loop, or to an EIA-RS-232-C interface. Connection details are tabulated below.

Function	CHANNEL NUMBER 1	2	3	4
Interface Connector J2	1	2	3	4
Interface Connector J3*	5	6	7	8
Function	Interface Connector Pin Assignments**			
Send Data (RS-232)	1	13	25	37
Logic Ground & Send Current Loop Return	2	14	26	38
Send Data (Current)	3	15	27	39
Receive Data (RS-232 & Current)	4	16	28	40
+12V via 1.3K ohms***	5/6	17/18	29/30	41/42
Low Speed Filter	7	19	31	43
Logic Ground	8	20	32	44
Receive Current Loop Return	9	21	33	45

Jumpers For Current Loop Operation

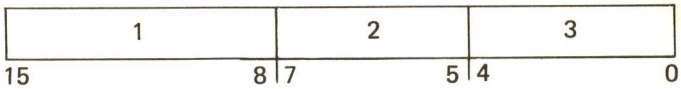
*Communications channels 5, 6, 7, and 8 are available only on Asynchronous Communications Controller Model 2612.

**J2 and J3 are 50-pin connectors. Pin numbers not shown in table are spares.

***Internally connected.

COMPOSITE INSTRUCTION

The instruction word for controlling the Asynchronous Communications Controller is divided into 3 sections as shown.



Section 1:

Output Instructions		Input Instructions	
39	from A Register	31	to A Register
3A	from B Register	32	to B Register
3B	from Memory	33	to Memory

Section 2: Function Code

The following examples use the A register, a device address of $1B_{16}$ and communication channel 1.

395B	Data Out from A
313B	Data In to A
315B	Status Byte In to A
31BB	Disable Interrupts

Section 3: Address

Standard device address is $1B_{16}$. The address can be strapped differently, depending upon system requirements. For communication channel identification, see DEVICE ADDRESS AND COMMUNICATION CHANNEL IDENTIFICATION.

SPECIFICATIONS

Character Format	1 start bit, Data, 1 or 2 stop bits
Data	5, 6, 7, or 8 bits
Bit Rate	To 9,600 bps
Error Control	Framing, Overrun, and VRC
VRC (Parity)	Odd, Even, or Disabled
Line Modes	Full- or Half-Duplex, with Transmit and Receive Circuits Completely Independent
I/O Bus Mode	Programmed Data Transfer
Interrupts	Data Service
Processor Interface	Standard Microdata I/O Bus
Modem or Terminal Interface	RS-232-C or 20-ma current loop
Environment	Operating 0°C to 50°C Storage –40°C to 80°C
Power Requirements	+5V; 1200 ma –16.75V; 300 ma +12V; 200 ma

Microdata™

Microdata Corporation
17481 Red Hill Avenue
Irvine, California 92705
Telephone (714) 540-6730
TWX 910-595-1764