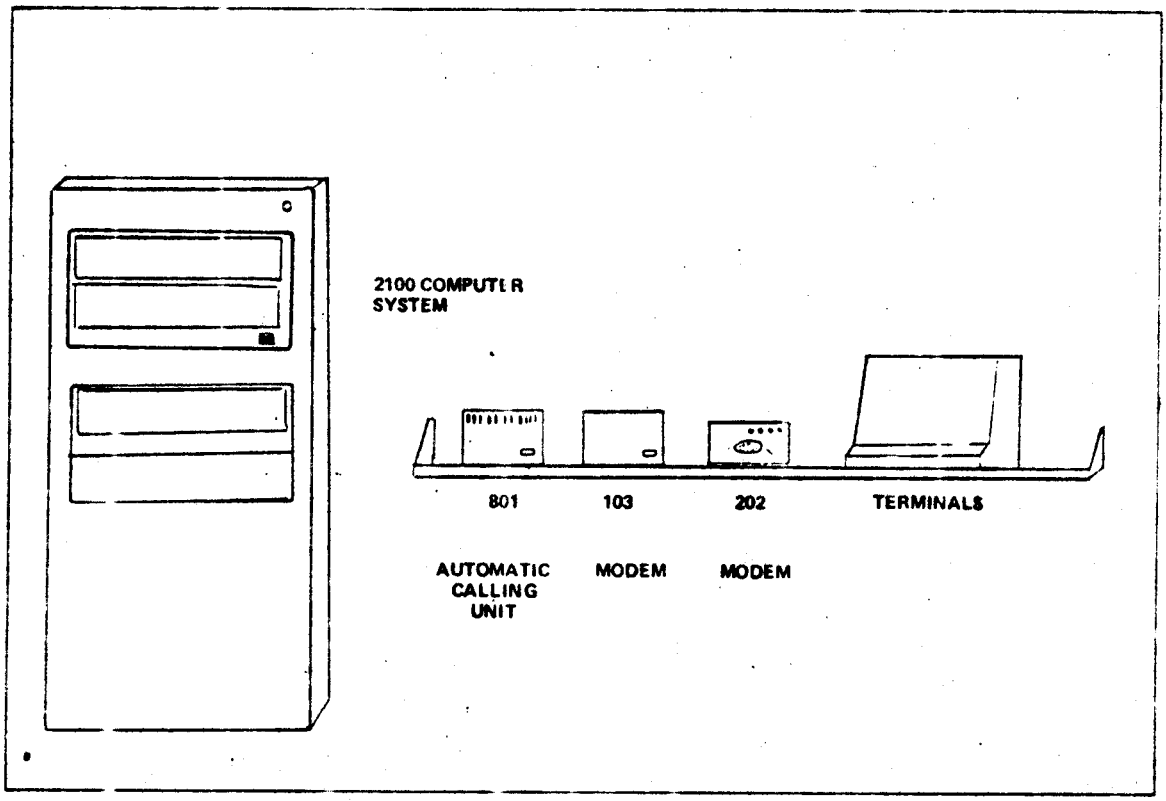
THE 12920
ASYNCHRONOUS MULTIPLEXER
APPLICATION NOTE

RICHARD LYMAN
January 1972

GENERAL DESCRIPTION

The 12920 Asynchronous Channel Multiplexer is an interface to sixteen asynchronous bit serial devices. These may be 103 modems, 202 modems, or hardwired devices. It may also operate up to eight 801 Automatic Calling Units.



A MULTIPLEXER allows signals from several lines to be channeled through a single port. In this case, the single port is the computer I/O system and the multiple lines are communication lines. There are many different types of multiplexers so, for simplicity, the Asynchronous Channel Multiplexer may be likened to sixteen buffered teletype interfaces. There are, of course many differences between the simple teletype interface and the multiplexer, but the mode of communication is the same, asynchronous bit serial.

Asynchronous bit serial devices, numbering in the hundreds comprise CRT's, hardcopy terminals, batch terminals, keyboard-printers, computers, tape readers, tape punches, card readers, tape cassettes, optical character readers, graphics terminals, etc. The requirements are:

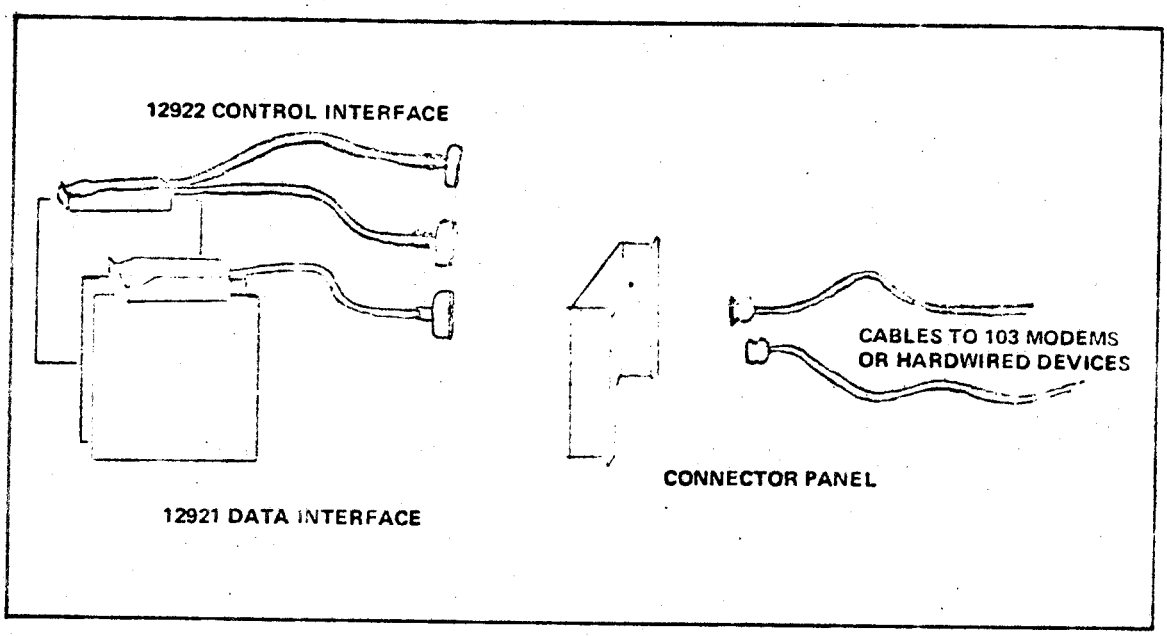
1) Electronic Industries Association RS 232 interface or CCITT V.24 interface.
2) Asynchronous start-stop transmission.
3) Speed in the 60 to 2400 bit per second range.

Essentially, any device which can operate into a 103 or 202 data set can operate with the multiplexer. It is largely because of the adherence of this large class of plug compatible devices to a single mechanical and

electrical interface, in Europe CCITT recommendation V.24 and in the U.S.A. RS 232 specifications, that has allowed manufacturers to develop standard products confidently in this area.

An important point is that the multiplexer can operate simultaneously and independently, any mix of these devices regardless of the differences between the devices in bit rate, line discipline, operating mode, etc.

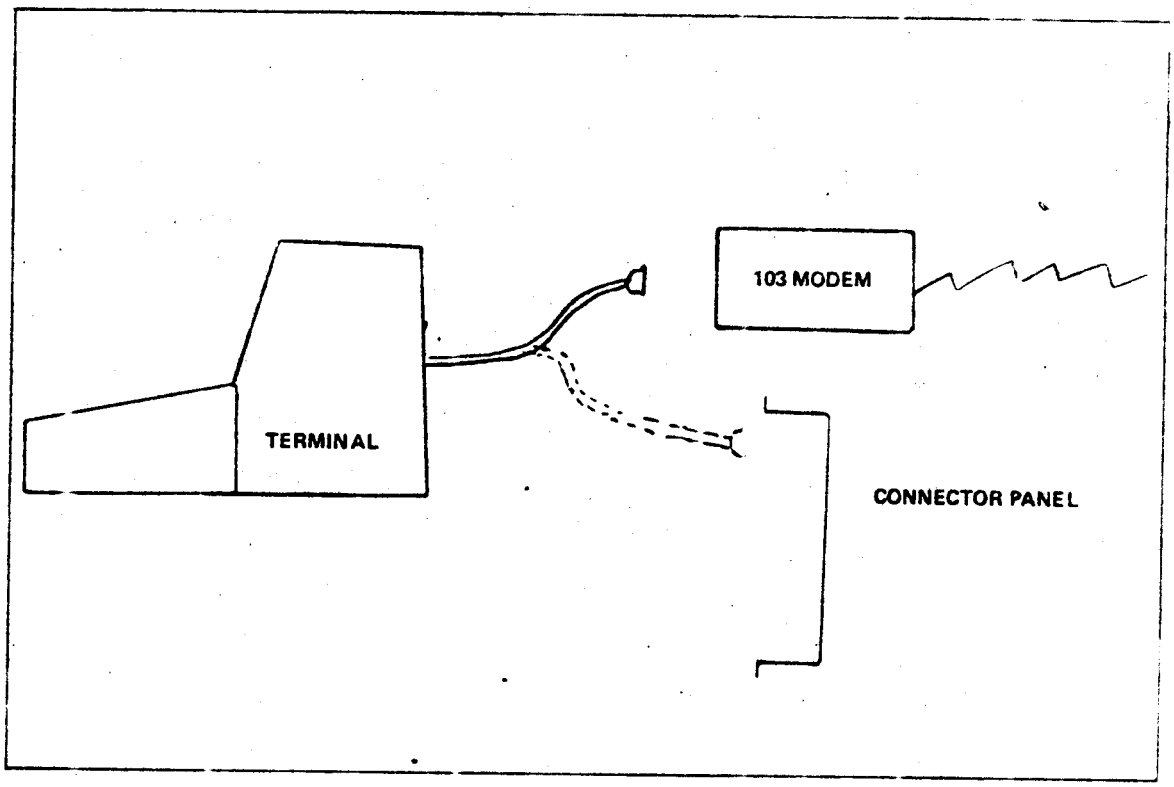
The basic 12920 kit consists of three interface cards, cables, and a connector panel. Two of these cards are the 12921 Data Interface and are linked through a double hooded cable to the panel. The third card is the 12922 Control Interface which, through two cables coming out of its hood, is linked to the panel. The panel, 30062, in turn connects to the devices either through customer supplied cables or the 30332 series cables. The sixteen connectors to the modems or devices are the standard RS 232 twenty-five pin "Cinch" connectors. For devices which go directly to the panel, the computer is simulating a data modem so that any terminal which can work directly into a data modem will be able to work directly into the panel.

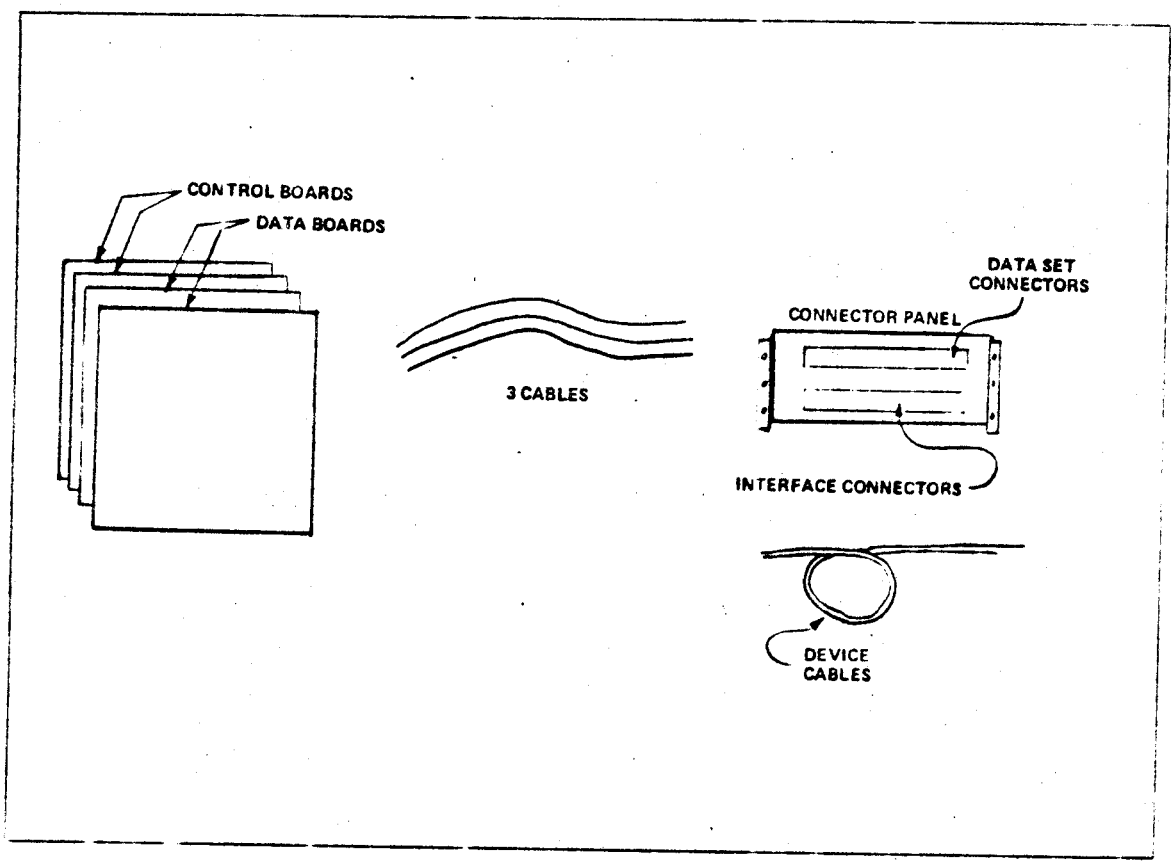


THE BASIC 12920 INTERFACE KIT

The 12920 option 001 adds one more 12922 Control Interface. This increases the number of control and status lines allowing use of up to eight 801 computer controlled Automatic Dialers, or allowing the use of up to sixteen 202 type data sets. Dialers, terminals, and modems may be intermixed with the restriction that each dialer takes up two of the sixteen ports.

From the panel to the device, there are three HP supplied cables, the 30332 which connects to a 103 or 202 modem, the 30332-001 which goes to an 801 automatic dialer, and the 30332-002 which is a brute force pin for pin twenty-five foot extension cable. In most cases, the cable which comes with a terminal, meant to go to a data set, will work plugged directly into the panel or through the extender cable into the panel.

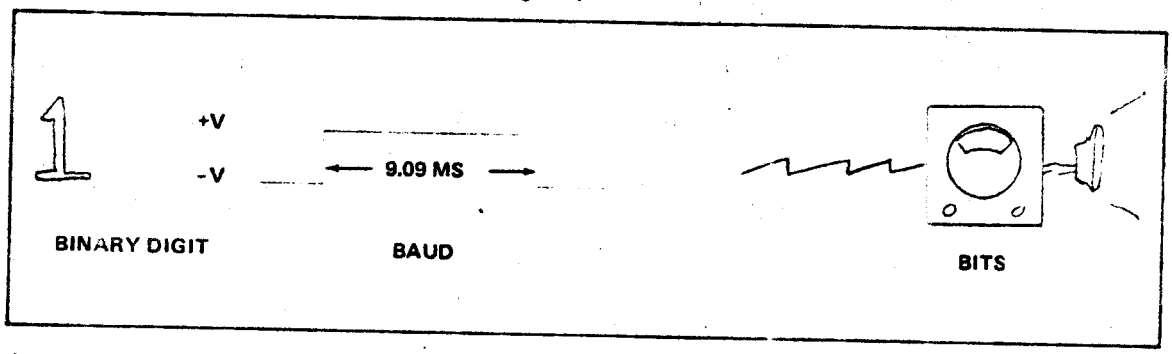

103 MODEM
TERMINAL
CONNECTOR PANEL


CONTROL BOARDS
DATA BOARDS
3 CABLES
DATA SET CONNECTORS
CONNECTOR PANEL
INTERFACE CONNECTORS
DEVICE CABLES

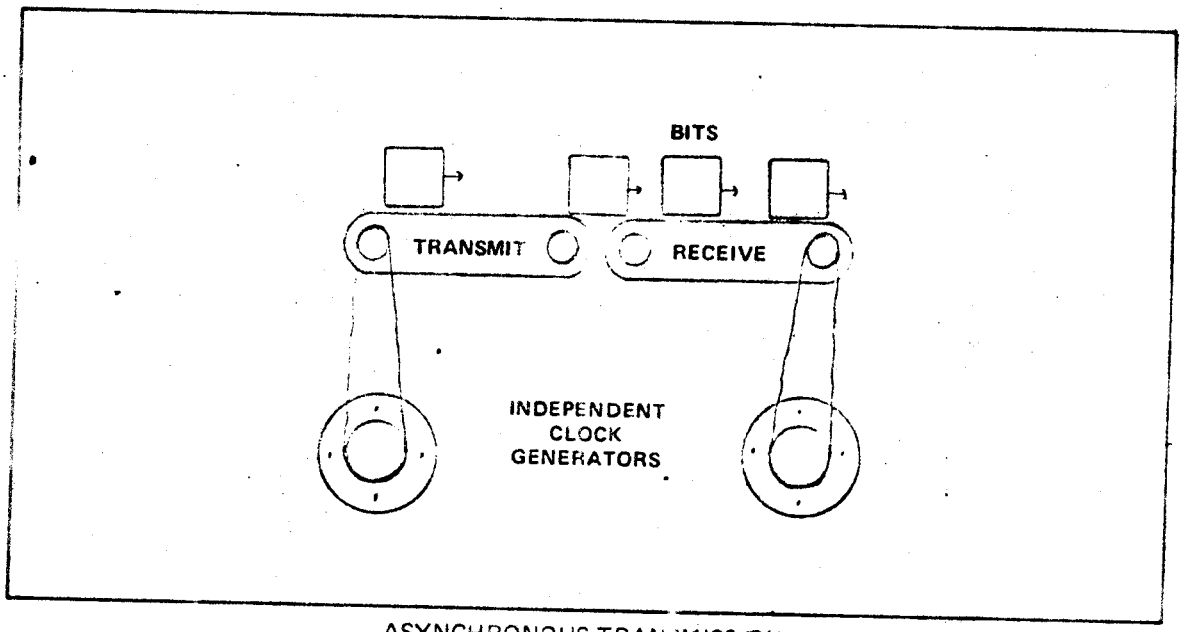
BIT SERIAL TRANSMISSION

The thread of commonality between the devices useable by the multiplexer is that data is transferred one bit at a time. Bit serial transmission uses only a single wire and a ground return to transfer information. Because of this advantage in the cost of transmission medium, this mode of operation has been around since the invention of Morse code. Today's asynchronous bit serial devices use three wires, one for data out, one for data in, and a ground return. Characters are sent by a timed sequence of bits.

A bit is a unit of information and "bit" usually is used instead of a "binary digit." Confused with "bit" is "baud", the shortest duration signalling element. On a 103 or 202 data set, "bit" equals "baud" equals "binary digit" and the terms are used interchangeably.

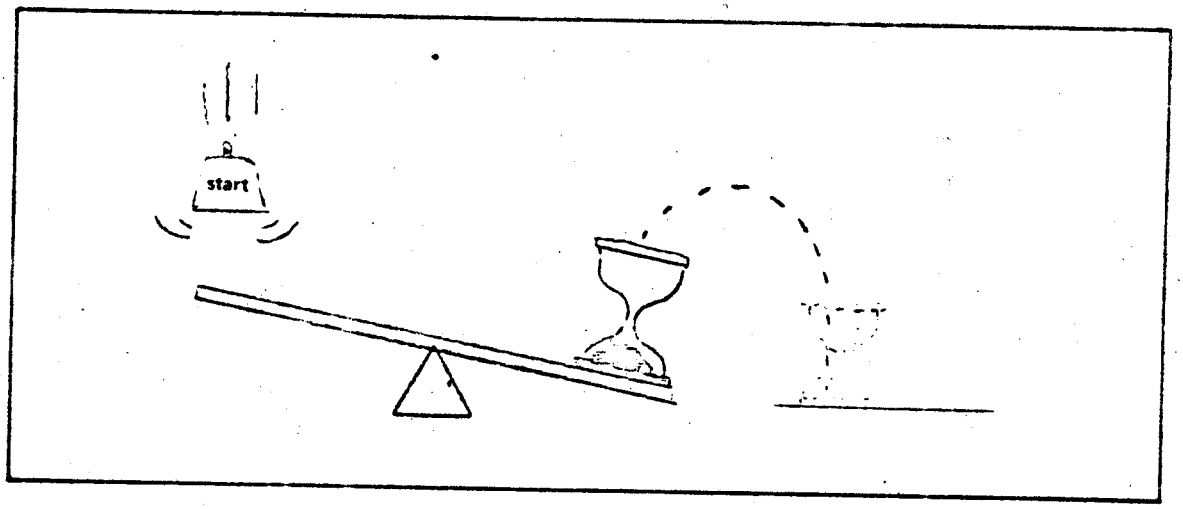


Asynchronous transmission is differentiated from synchronous transmission by the method of generating the bit timing. In asynchronous transmission, the receiving and sending ends have independent clocks to strobe the bits in or out.



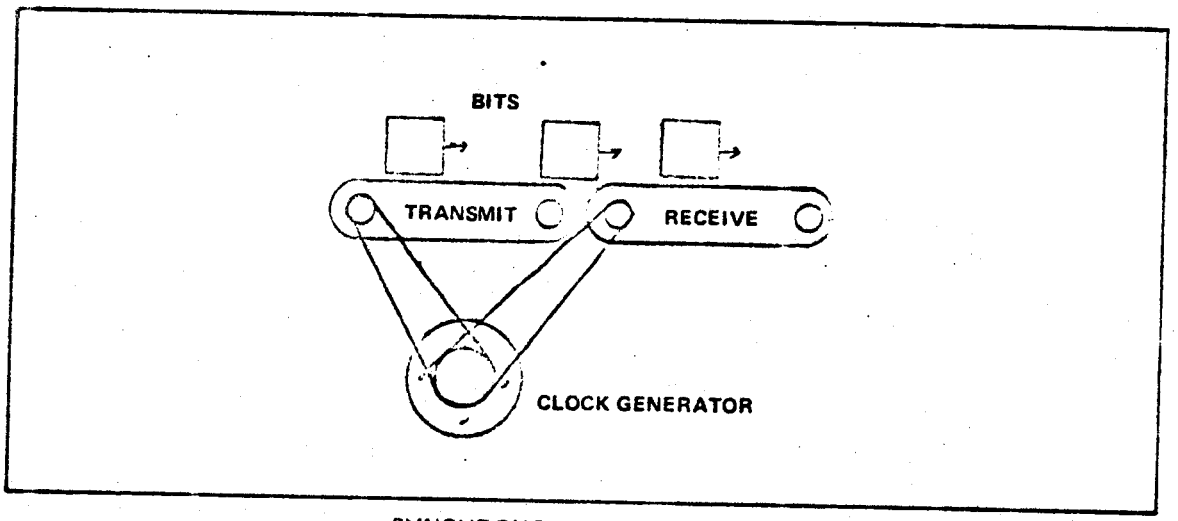
ASYNCHRONOUS TRANSMISSION

The clocks might not be going the [illegible] frequency, but can differ by up to a few percent before transmission failure occurs. Since the [illegible] are not in synchronization, the transmission mode is termed "asynchronous". In order to know wh[illegible] ift a bit in, the receiver must be able to synchronize itself with the bit stream. It does this by [illegible] a start pulse and times to the middle of the bits from this transition. It doesn't matter how muc[illegible]es between the stop bits of one character and the start bit of another because the receiver always [illegible]izes itself.



RESYNCHRONIZATION AT THE BEGINNING OF EACH CHARACTER

Synchronous transmission, on the other hand, allows each end of the transmission link to have access to the same clock. This means that both the receiving and sending end clock bits in and out at exactly the same frequency. Blocks of thousands of bits can be sent without the need for re-synchronization.



SYNCHRONOUS TRANSMISSION

The bit-serial data communications devices are perhaps the largest group of plug to plug compatible devices which exist. There are, however, several difficulties which must be surmounted in order to utilize this resource pool.

First, there are at least a dozen speeds at which these devices transfer their bits. This is known as the device BIT or BAUD rate. The HP 2600 terminal alone can communicate at 10 different switch selectable speeds.

TYPICAL BIT RATES

HP 2749 (ASR 33 teletype)	110 bits per second
IBM 2741 terminal	134.5
HP 2605 (Univac DCT 500)	110, 150, or 300
Memorex 1240 terminal	110, 150, 300, 600
HP 2761 marksense card reader	1050
HP 2600 CRT (Datapoint 3300)	110, 220, 440, 880, 1760, 150, 300, 600, 1200, 2400

This is no problem on the multiplexer as it allows any of the channels to be configured by a simple output of a control word to any one of 256 (249 useable) frequencies covering the spectrum of speeds between 56 bits per second and 2400 bits per second. This completely and cleanly eliminates the speed compatibility problem.

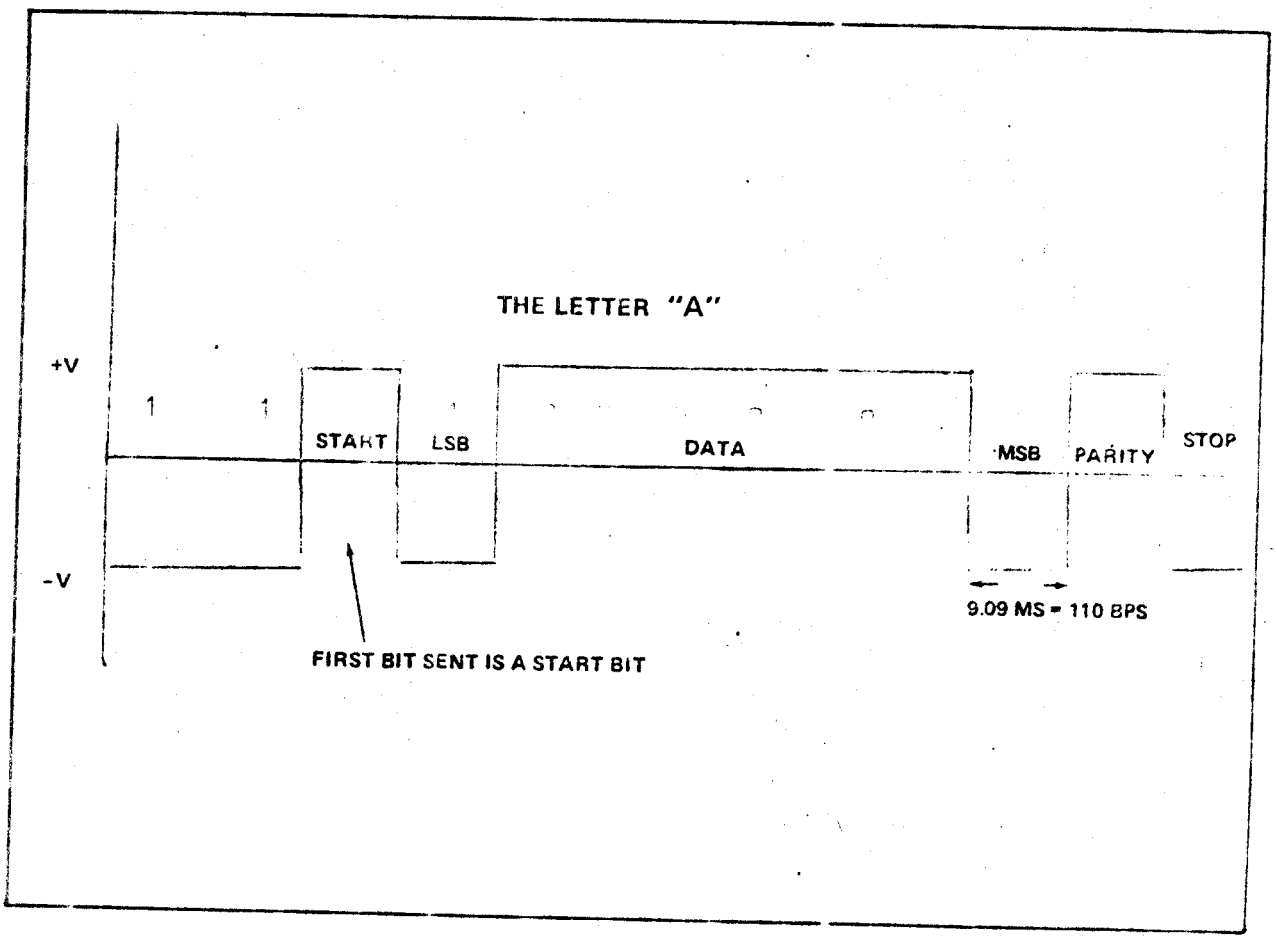
Second, there are a variety of character formats represented, from the 5 data bits of a Baudot code teletype, to the 6 bits of an IBM 2741, to the ordinary ASCII 7 bits plus parity, and to, in the future, 8 bits plus parity USASCII8.

TYPICAL CHARACTER SIZES

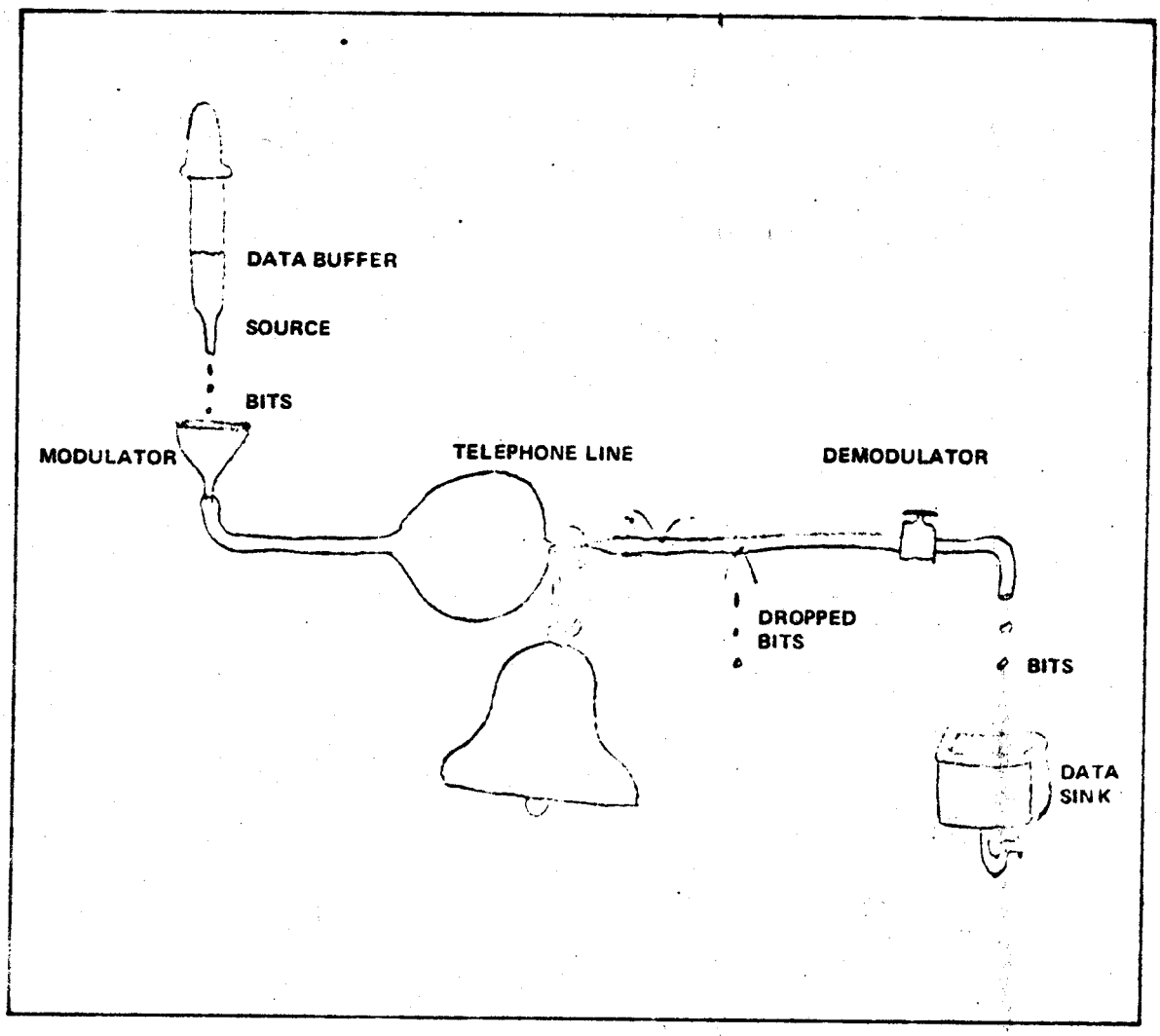
HP 2749 teletype	11 bits = 1 start, 7 data, 1 parity, 2 stops.
IBM 2741	9 bits = 1 start, 6 data, 1 parity, 1 stop.
HP 2605, HP 2761, HP 2600	10 bits = 1 start, 7 data, 1 parity, 1 stop.
Baudot code	7½ bits = 1 start, 5 data, 1½ stop (send 2; receive 1).

Character size and a number of other parameters are programmable for each channel individually. Thus, the interface can simultaneously run any mix of bit serial device types.

The third compatibility factor is control of the device which is in most cases a data set. By providing the important control lines and creating an interface which can be set to give an interrupt on rising, falling, or both edges of a status line signal, generality and flexibility of usage is maintained. For instance, it is possible for an 801 automatic dialer, a 103 data set, a 202 data set, a 103 compatible hardwired terminal, a 202 compatible hardwired terminal, and a Touchtone receiver data set to be operating simultaneously.

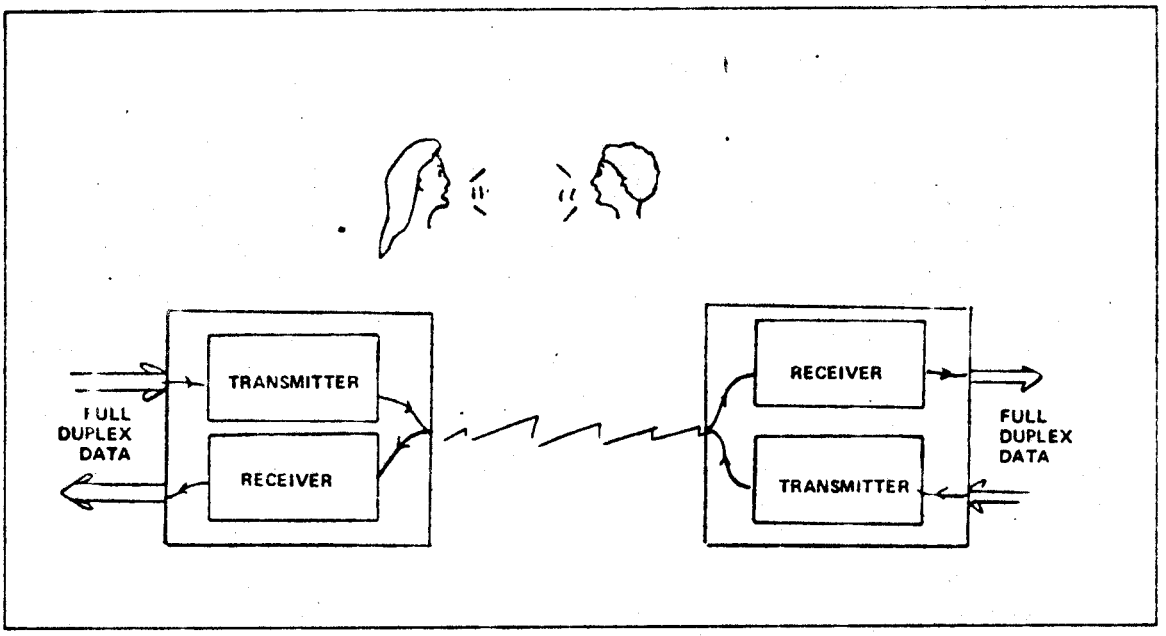


DATA MODEMS



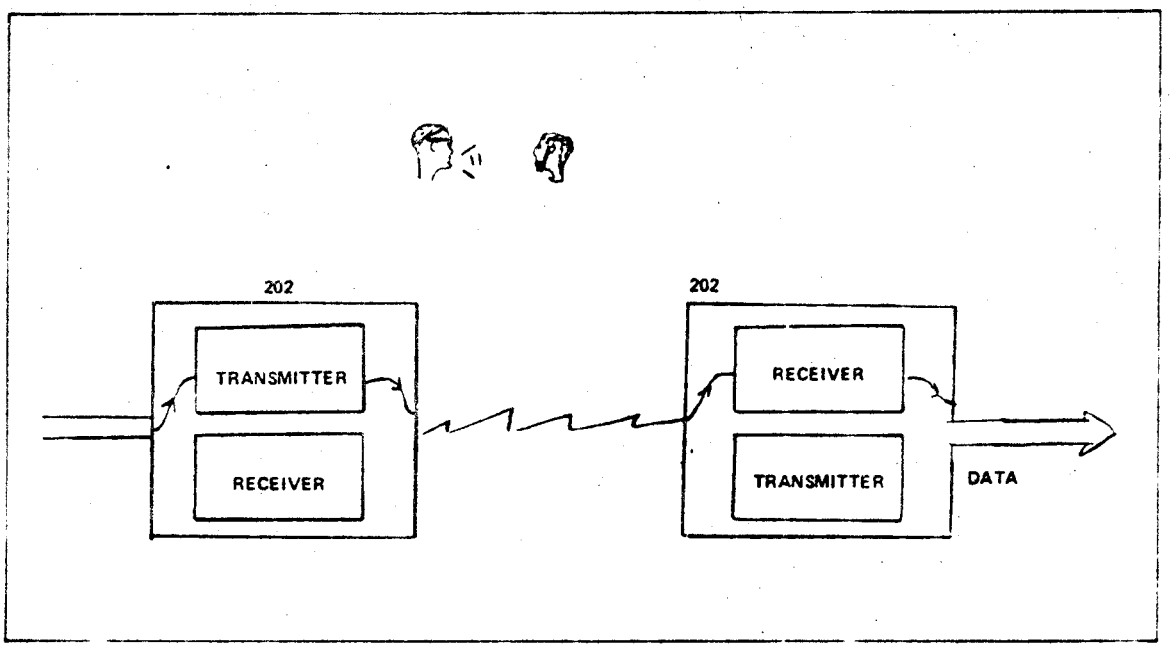
There are two types of asynchronous modems in widespread use. These are the 103 type and the 202 type, patterned after Western Electrics modems. The word "modem" stands for modulator-demodulator which is equivalent to transmitter-receiver.

The 103 is a full duplex, frequency shift keyed, 0 to 300 baud modem.



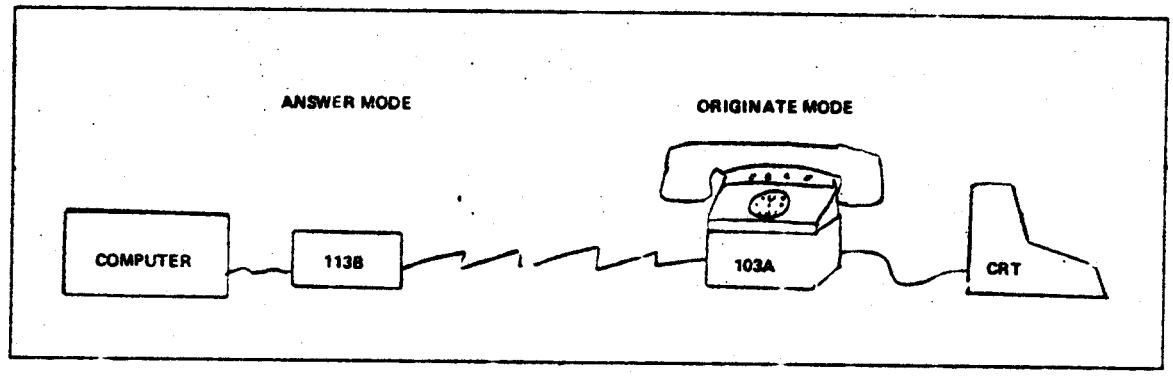
FULL DUPLEX

Full duplex means that each terminal may be transmitting and receiving simultaneously. This means that the data being received can be different from the data sent. Some computer systems use this fact to print out a different character from that typed, or to separate input and output into independent information streams. Most usually, as in the case of the teletype, the full duplex operation is used to echo back the same character that was typed. This provides a simple and very effective method of error control. A bit error on the line will cause a different character to be printed from that which was typed and can be detected visually at the terminal. Some terminals which operate over a 103 are half duplex, that is data can be received or transmitted but not both ways simultaneously. A half duplex terminal usually has a direct connection between the keyboard and the printer to cause printing while typing.

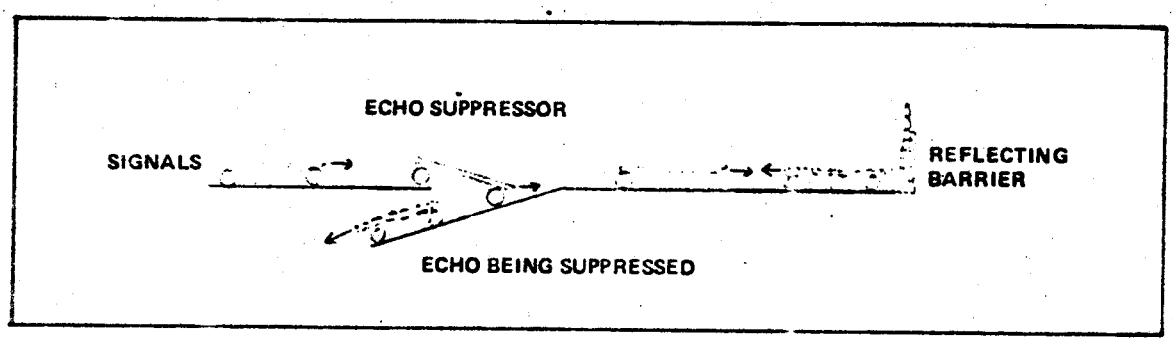


HALF DUPLEX

A 103 trans.mits on one frequency and receives on another, the frequencies being determined by whether the data set originated the call or answered the call. The 103A can be used both for originating and answering, but some data sets can be used only in one mode such as the 113B which can only answer.



The 202 type data set goes up to 1200 bits per second but is half duplex. A line turn around is necessary to switch the modem from send to receive and vice versa. This turn around takes 200 milliseconds when working over the direct dialed network because of the process of disabling echo suppressors, those devices which keep us from hearing our own echoed and delayed voice.



The long duration of the turn around degrades the terminal's throughput in highly interactive systems. The 202 also has a very slow, 5 baud, reverse channel which can be used to send information opposite the direction of the main information flow. This channel can be used to request retransmission, signal errors, send low speed information, or signal ready to receive. In interactive terminals, the reverse channel is connected to the break key so that the terminal may interrupt the computer.

The signals which connect between the computer and the data set are important because interaction with these signals varies widely from terminal to terminal.

BA (BA is the standard mneumonic used with this signal.) This is the serial data to the data modem. The data modem will send a mark frequency if this line is a logical 1 (negative volts) and will send a space frequency if this line is a logical 0 (positive volts). Some time later, normally a few milliseconds, the value of this line will come out the BB line of the distant data set.

BB The BB line is the serial data line from the data modem. It will be either 1) a logical 1 if the data set is not in the data mode, 2) the same value, delayed as the distant BA line, 3) or, on a 202 C modem in the transmit mode, the value of the local BA line. This signal wrap around sometimes causes unexpected situations during program debugging because of reception of one's own data.

CD Data Terminal Ready to the data modem is very important since, unless it is a logical 1, positive volts, the data set will neither send, receive, nor answer the telephone. When it is dropped to 0, the data set will disconnect from the line, a feature useful for eliminating an unwanted caller.

CA Request to Send to the modem places a 202 in the send mode when logical 1 or the receive mode when logical 0. This line must be toggled whenever a line turn around is desired. CA does function on some private line 103's to turn the transmitter on and off, a feature useful for multi-drop configurations.

SA Secondary Channel Send, to the modem, sends a signal on the 202 reverse channel. On a 103 it has no function. The modulation technique on the reverse channel is on-off keying of a 387 Hertz tone which can be heard as a loud whistle on the telephone.

CC Data Set Ready, from the modem, is equivalent to the data button light on the phone, i.e. the data set is in the Data mode. When the data set automatically answers a call, or the data button is pushed, CC comes to a 1. When the data set goes out of the data mode either because the data set automatically disconnects or the call is terminated by lowering CD, then CC goes to 0.

CF Carrier Detect is a status line from the modem which indicates, when on, that a signal from another modem is being received. On a 103 modem, since the distant station is continuously transmitting, it should remain on for the duration of the call. On a 202 modem, in the receive mode, CF indicates the presence of the other carrier, but in the send mode indicates ones own carrier. During the line turn around, carrier detect goes off then on again. A loss of carrier detect during a reception is regarded as a line fault, the normal action being to disconnect by dropping CD, Data Terminal Ready, which places the telephone back on hook.

CB Clear to Send, is request to send brought back to the data terminal through a delay circuit. Providing the necessary timeout for a line turn around, the delay is two hundred milliseconds when strapped for communication links with echo suppressors. When request to send, CA, is removed, then CB goes away within a few milliseconds. On a 103 modem, this line is usually the same as CF, carrier detect.

SB Secondary Received Data, is on a 202 data set, the signal coming from the 5 baud reverse channel. This line reflects 1) the distant station's SA line delayed by 10 to 100 milliseconds when in the send mode, 2) one's own SA line when in the receive mode, or 3) a 0 when the data set is unplugged, the telephone line is down, or the distant end is off the air.

CH Frequency select, a control line to the modem, is not in general use in the United States, but is used in Europe where it determines the frequencies for frequency shift keying. This gives a tradeoff between speed and reliability.

REQUIREMENTS FOR AUTOMATIC ANSWER

1) Data Set "AUTO" button depressed.
2) CD, Data Terminal Ready, ON.

REQUIREMENTS FOR 103 TRANSMISSION OR RECEPTION

1) CC, Data Set Ready, ON.
2) CF, Carrier Detect, ON.
3) CD, Data Terminal Ready, ON.
4) CA, Request to Send, ON.

REQUIREMENTS FOR 202 DATA TRANSMISSION

1) CC, Data Set Ready, ON.
2) CB, Clear to Send, ON.
3) CA, Request to Send, ON.
4) CD, Data Terminal Ready, ON.

REQUIREMENTS FOR 202 DATA RECEPTION

1) CC, Data Set Ready, ON.
2) CA, Request to Send, OFF.
3) CD, Data Terminal Ready, ON.
4) CF, Carrier Detect, ON.

SOME CAUSES FOR DISCONNECTION

1) CD, Data Terminal Ready, OFF.
2) Carrier loss, if data set has option for disconnection on carrier loss.
3) 3 second space, if data set has option for long space disconnect.
4) Pressing "TALK" button on Auto Answer data set.
5) Loss of signal (10 seconds) from the switched network.
6) Housewife call, if data set has option to disconnect if carrier does not come up 10 to 20 seconds after CC, Data Set Ready, ON.
7) Cable or data set not plugged in.

RS-232 DEFINITION	DATA MODEM			
Command Lines	103A, E, G, H, 113B	DATREX	103F	202C/D
Request to Send (CA) "ON" conditions data modem to transmit. With modems strapped for half duplex operation, ON inhibits receive mode, "OFF" conditions data modem to receive.	Not present	Not functional	"ON" turns transmit carrier on.	Conforms to RS-232C
Data Terminal Ready (CD) "ON" allows data modem to be connected to communications lines. "OFF" disconnects data modem from communications line. With modems strapped for automatic answer of ringing, "ON" condition enables automatic answer to occur.	Conforms to RS-232C 50 millisecond "OFF" interval insures disconnection of communication line.	"ON" allows connection of far end terminal. "OFF" disconnects far end terminal and prevents connection of far end terminal. 130 millisecond "OFF" interval insures disconnection.	Not present	Conforms to RS-232C
Status Lines				
Clear to Send (CB) "ON" is a delayed response to CA "ON" and indicates that the timed interval necessary for "setting" of communications line has occurred.	"ANDED" with CF, otherwise conforms to RS-232C.	Installer option connects CB to CC so both carry same information.	Same as 103A	Conforms to RS-232C
Data Set Ready (CC) "ON" indicates data modem is connected to the communications line.	Conforms to RS-232C	"ON" indicates far end terminal connected to line.	Same as 103A	Conforms to RS-232C
Carrier Detect (CF) "ON" indicates presence of carrier from far end. With modems strapped for half duplex "ON" indicates either local or far end carrier.	"ANDED" with CB, otherwise conforms to RS-232C.	"ON" indicates DATREX is operational. "OFF" is a trouble indication.	Same as 103A	Conforms to RS-232C
Ringing (CE) "ON" indicates presence of ringing from communications line.	Conforms to RS-232C	Not present	Not present	Conforms to RS-232C
NOTES	All are full duplex over a dial-up or leased line. 103E,G,H and 113B have an installer option for "ANDING" CB & CF leads to conform to 103A exactly. The 103E,G,H and 113B also have an "abort timer" feature which disconnects modem from communications line if far end carrier not received within 12-20 seconds after data set ready goes from "OFF" to "ON".	DATREX is full duplex.	103F requires a leased line (not dial-up). Effectively the computer interface is identical to a hardwired terminal if CA is maintained in the "ON" state.	202 is half duplex over a dial-up line and requires a leased line (4 wire) for full duplex. If used in half duplex mode, on change from transmit mode to receive mode, CA must remain on for 1 millisecond after last data bit enters "transmit-in" lead from computer. This allows time for D-A conversion in modem.

OTHER MODEMS

Most asynchronous modems are a variation on the 103 or 202 modem. The split speed modems behave like 202's except the reverse channel is fast enough to support meaningful information; like keyboard input.

Touch tone telephones may be used through a 403 D7 data set which is a touchtone receiver with a bit serial interface to the computer. It behaves like a 103.

With all of the non-telephone company modems, a Data Access Arrangement is needed to go between the modem and the telephone line. This protects the line from the modem and the modem from the line. DAA's are available for manual operation and for automatic operation. The Automatic DAA's allow customer generated dial pulses and unattended answer.

THE EIA RS 232 C INTERFACE

The RS 232 specification is a set of electrical and mechanical rules intended to standardize data set interfaces. The document may be obtained from the Electronics Industry Association,

EIA Engineering Department
2001 Eye Street, N.W.
Washington, D.C. 20006

price $4.40.

The basics are the following:

1) The connector to be used on the modem is a 25 pin "Cinch" or "Cannon" connector. The standard for this as used by the telephone company and HP has a grey breakable plastic hood. This is not in the specification, but is an established standard.
2) Any line must be able to be shorted to any other. This means that the connector may be plugged or unplugged without blowing up the computer, or in the case of the 2100, without halting the computer.
3) Input impedances must be within 3000 to 7000 ohms range.
4) All control and status lines are on, logical 1, for 3 to 15 volts and off, logical 0, for -3 to -15 volts.
5) Data lines are 1, mark, for minus volts and space, 0, for positive volts.

PIN CONNECTIONS ON DATA SET CONNECTOR

1	Protective Ground	(this line should not be connected to the computer ground)
2	BA	Transmitted Data
3	BB	Received Data
4	CA	Request to Send
5	CB	Clear to Send
6	CC	Data Set Ready
7	AB	Signal Ground (Common Return)
8	CF	Received Line Signal Detector
11	SA	Secondary Transmitted Data (this is not a standard EIA RS 232 pin number. 202's are different this way)
12	SB	Secondary Received Data (same as SA)
20	CD	Data Terminal Ready
22	CE	Ring Indicator
23	CH/CI	Data Signalling Rate Selector

THE 12920 ASYNCHRONOUS CHANNEL MULTIPLEXER

The following is a "how to do it" reference on using the multiplexer. A complete explanation of what is happening in the hardware is available in the 12920 manual.

ACTION DESIRED	ACTION REQUIRED	EXPLANATION
clear flags set by power on	CLF CNTL CLF MPX CLF REV	Interrupt will occur on the first STC if flags are not clear. CNTL is the select code of the 12920 control board. MPX is the data board. REV is the 12920-001 control board.
detect a status bit change	LDA UNIT ALF, ALF RAL, RAL IOR CWS OTA CNTL STC CNTL	UNIT is the port 0-15 of the device. CWS is the control word requesting an interrupt upon a status bit value.
generate a 100 millisecond timeout	LDA =B161202 JSB OUT LDA =B47777 JSB OUT	Set up the channel to send all marks at teletype speed. Interrupt will occur in 100 ms.
configure a channel	LDA CW JSB OUT	CW configures either a send or a receive channel with bit rate, character size, etc.
send a character	LDA CHAR IOR =B43600 JSB OUT	The 7 bit character is or'd with stop bits and is sent. An interrupt occurs at the end of transmission.
send a long space	LDA =B40000 JSB OUT	The output remains at a space until changed by another output.
output data or parameters	OUT NOP LIB MPX SSB JMP *-2 OTA MPX LDA UNIT ALF, ALF RAL, RAL OTA MPX+1 STC MPX JMP OUT,I	Enter with data in A. Check seeking bit. Output data. Position and output the unit number. Initiate the operation.
acknowledge an interrupt from data transmission or reception	CLF MPX	The multiplexer will not interrupt again until it has received a Clear Flag.
acknowledge an interrupt from status lines	CLF CNTL CLF REV	

TABLE OF COMMON PARAMETERS

OUTPUT	(all numbers are octal)
TTY ASR 33 (11 bit format, ASCII)	161202
IBM 2741 Selectric (9 bit format, PTTC/BCD)	160157
30 cps terminal (10 bit format)	160457
60 cps terminal	160427
120 cps terminal	160413
240 cps terminal	160405
150 baud Baudot code terminal	163537
TTY with parity	171202
TTY with diagnose (data is routed to auxilliary channels)	165202
INPUT	
TTY ASR 33 (with echo)	131202
IBM 2741 Selectric	120157
30 cps terminal (with echo)	130457
60 cps terminal (without echo)	120427
120 cps terminal	120413
240 cps terminal	120405
TTY with diagnose (input data routed to auxilliary channels)	135202
150 baud terminal (Baudot code)	133537
STOP BIT FORMATS (to be inclusive or'd with the character before output)	
TTY (ASCII) (without parity or with odd parity)	43600
TTY with even parity	43400
Selectric	43600
Baudot code	43740
Time delay character on channel configured without parity	47777
Time delay character on channel configured with parity	47577

PARAMETERS FOR THE DATA INTERFACE

BIT RATE PARAMETER

This eight bit number initializes an eight bit counter which measures the time between bits. The parameter is 14,400/bit rate - 1. A jumper on the interface will cut the crystal oscillator frequency in half making the parameter = 7,200/bit rate - 1. This extends the multiplexer range for the 45 or 50 baud applications but deletes 2400 baud operation. The crystal oscillator does not drift and is accurate to within .05%. If the frequency derived from the parameter is over 3% off the actual bit rate of the device, errors may appear.

CHARACTER SIZE

The 3 bit parameter controls the number of bits which will be shifted in or out of the interface. It is the total number of bits per character minus 1. The most significant binary digit is dropped. For a teletype, eleven bits, the parameter is 7011-1=010.

DIAGNOSE

The diagnose bit, when on, creates a data path between a main channel, one of the 16 send or 16 receive channels going to the device, and the five receive-only channels. For a send channel, this allows output of data on a main channel and reception on the extra channel, an attribute useful for on-line diagnostics or monitoring of output. For a receive channel, this routes input data to the extra channels so that data can be received at six speeds simultaneously. This is a great aid in designing device type and device speed recognition algorithms.

PARITY

When on, Parity causes ASCII even parity to be generated on a send channel. When the character is being output, it should be or'd with an octal 43400, the stop bits. To send odd ASCII parity, the stop bits should be an octal 43600. This bit does not influence the parity check bit in the status.

ECHO

On a receive channel, echo transmits a bit with the reception of each bit from the device. The transmitted bit pattern lags 1/2 bit behind that received. Bits transmitted are exactly those that are input to the computer. Whenever a break is received, only one character of break is echoed after which the output line is returned to the mark condition. This avoids the distant data set from hanging up by keeping from sending a long space.

ENABLE

This bit does not turn off the channel but instead acts as an interrupt dump. When off, no interrupts will be generated from the channel, all input characters being thrown away. When on, the channel will interrupt at the completion of each send or receive operation. It is not necessary to ever disable a Send channel since the program, in effect, commands each interrupt individually by output of a character. When no more interrupts are desired, no more characters are sent. On a receive channel, the character interrupts will keep coming unless the enable bit is turned off.

SYNC

Sync is actually the start bit of the character. When set to one, and when the character is all 1's, nothing is sent. Marks, the idle line state, are clocked out for one character time after which there is an interrupt. This

has proved to be very useful because each unit has, in effect, its own time base generator. It usually happens that time outs are needed when data is not being sent so this timing does not interfere with transmission. When parity is being generated, the sync word should be 47577 instead of 47777 because parity inverts bit 7.

IMPORTANT

This can be the source of programming problems. Use of the wrong time delay character will cause the terminal to go out of synchronization. Use 47577 when parity is being generated (bit 12 of the output parameter word is 1). Use 47777 when parity is NOT being generated (bit 12 of the output parameter word is 0).

UNIT NUMBER (CHANNEL NUMBER)

There are sixteen units 0 through 15 each of which has two channels, one send and one receive. There are five more units, 16 through 20 which only can receive. The unit number must be output on the upper select code whenever data or parameters are output. Any unit number outside the legal range will cause the interface to hang up with seeking on, a condition correctable by pushing preset or output of a legal unit number.

DATA

Characters must be right justified with stop bits or'd in. The stop bits are bit 14 and any bits to the left of the character in the character field. If the stop bits are not put in, the line will be set to a space at the end of each character and the terminal will print incorrect data.

STATUS DATA INTERFACE

SEND/RECEIVE

If the interrupt came from the completion of a character transmission, the Send/Receive bit will be 1. If it came from completion of a character reception, the bit will be 0. This, like the other status bits, comes from the last channel to set the flag. After a Clear Flag instruction, another channel can bring its status bits into position.

CHARACTER LOST

When ON, this status bit indicates that two characters were received successively on a channel without being able to set the flag. Each receive channel has a character buffer so that one complete character time is allowed for interrupt service time. There are actually three layers of buffering in the interface so that it is possible for one channel to have three characters in the interface, one having set the flag waiting for input, another in the character buffer, and another being serially shifted in. With a properly designed driver, this status bit should never be on and can be completely ignored.

BREAK

A Break is a completely zero character including the stop bits. This means, at different bit rates, the detected character will be different lengths. For instance, the space must be different lengths. For instance, the space must be 100 milliseconds from a teletype or 5 milliseconds from a CRT in order to set the Break status bit. A break is received just like any other character so that, in order to receive it, the channel must be primed to receive characters. Break is detected on a send channel if, for the duration of a character output, the input line remains a space. After receiving a break on a receive channel, only one interrupt will occur, the interface holding off reception of the next character until the input line returns to a mark.

DIAGNOSE

This status bit is just the Diagnose parameter brought back as status. It is useful in determining which part of a driver to go to following an interrupt.

UNIT NUMBER

The most recent channel to set the flag,displays its unit number as status.

SEEKING

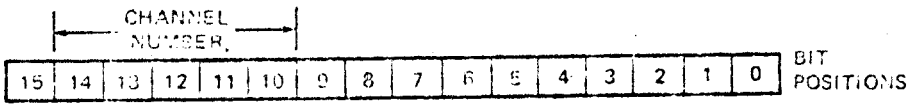
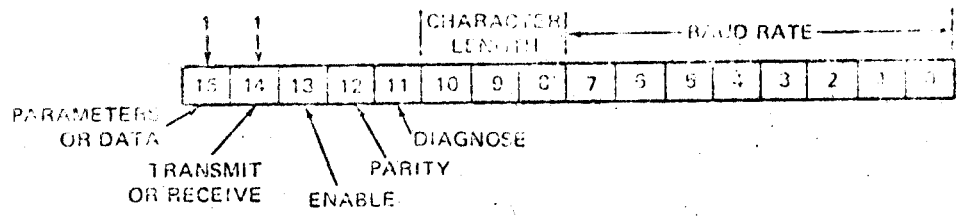
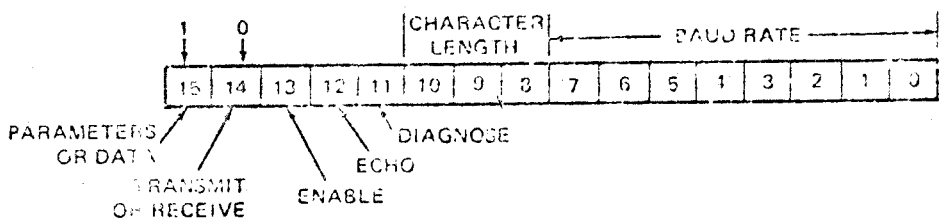
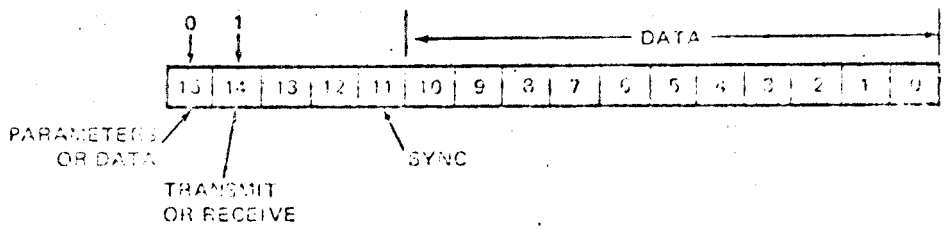
After a Set Control instruction, some period between 0 and 70 microseconds is taken searching for the correct field to fill in the circulating memory. Until the previous information is disposed of, no new data may be output. Thus seeking must be checked before any OTA until it is 0.

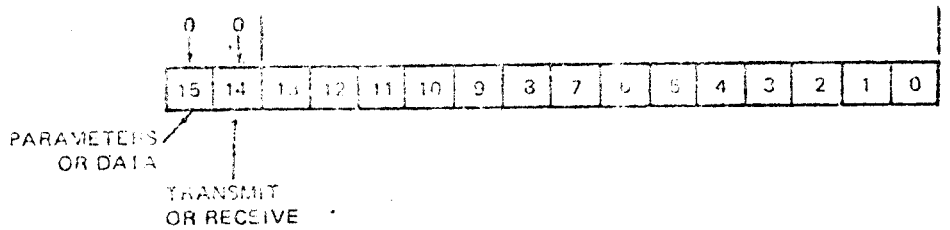
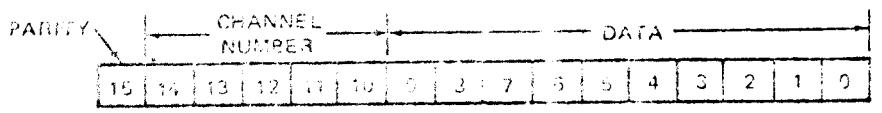
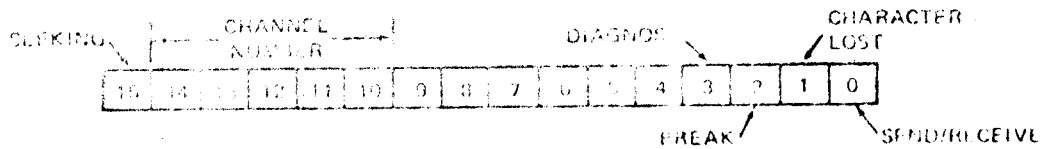
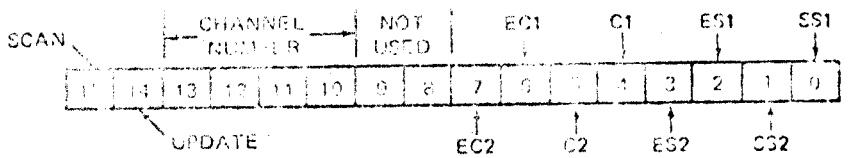
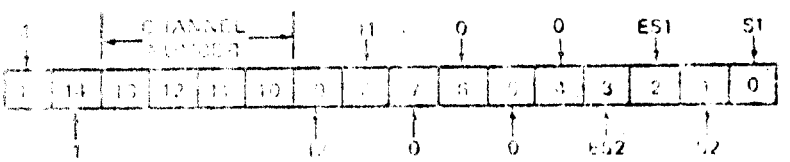
DATA

Characters come in right justified. Unwanted bits to the left of the character should be masked off since they are garbage. Characters with up to 10 bits of data may be input.

PARITY CHECK

When an ASCII character is received with even parity, the parity check bit will be 0. If it has odd parity, the bit will be 1. This bit is the even parity of data bits 0-7.

OPERATION MNEMONIC	DESCRIPTION AND WORD FORMAT
	COMMANDS **Note: All communication between the data interface and computer memory must contain a channel number to designate which of 16 devices is making an input or output transfer.**
OTA (Upper Select Code)	Output channel number to interface. 
CLF (Lower Select Code)	Acknowledge interrupt. (Data interface provides an interrupt per character and must be primed with parameters before channel will transmit or receive.)
STC (Lower Select Code)	Initiate output of parameters or data to the data interface.
	INFORMATION OUTPUT
OTA (Lower Select Code)	Output parameters or transmit to channel. 
OTA (Lower Select Code)	Output parameters or receive from channel. 
OTA (Lower Select Code)	Output data to a transmit channel. 

OPERATION MNEMONIC	DESCRIPTION AND WORD FORMAT
OTA (Lower Select Code)	INFORMATION OUTPUT (Continued) Illegal. May alter received data. 
LIA (Lower Select Code)	RECEIVE DATA AND STATUS Receive data from channel. 
LIA (Upper Select Code)	Receive status from channel. 
OTA/B (Select Code)	Output control word. **12922 CONTROL INTERFACE** 
STC (Select Code)	Enables computer I/O interrupt.
LIA/B (Select Code	Receive status word 

PROGRAMMING

DATA INTERFACE

The Data Interface gives an interrupt per character and must be primed with parameters before a channel will send or receive. All communication from interface to computer and vice versa is accompanied by a unit number which designates which of the 16 devices is making the input or output transfer. The data interface takes up two adjoining computer or extender select codes with data or parameters being output and data being input on the lower numbered select code, and unit number being output and unit number and status input on the upper select code. As an example, data may be output to select code 13(8) and the unit number to select code 14(8). Interrupts, flags, etc. only come from the lower select code (e.g. 13).

All outputs of parameters or data to the interface are followed by a set control which initiates an operation. A clear flag must be issued after each interrupt to acknowledge the interrupt to the interface.

EXAMPLE:

```
        LDA CW1         send control word
        OTA SC          output control word
        LDA UNIT        unit number of interface channel
        ALF, ALF
        RAL, RAL
        OTA SC+1        output unit number in bits 10-14
        STC SC          initiate operation
        .
        .
CW1     OCT 161202      send mode TTY speed.
UNIT    OCT 5
SC      EQU 13b         the data boards occupy I/O slots 13 and 14
```

Unit numbers 0-15 correspond to devices connected to the multiplexer. Numbers 16-20 are extra receive-only channels which obtain their data internally in the interface.

Output of data and parameters is identical except for bit 15 of the word output. This bit determines whether the other 15 bits will be interpreted as data or parameters. The mechanism within the interface for transferring the information into the correct position in the MOS memory is the same for both information types. The order, whether the data or unit number is output first, is unimportant, just so that these two items are output before the set control. An output instruction simply clocks data out of the CPU into registers on the interface. Not until the set control are the contents of the registers important.

The unit number is in bits 10-14 because this positions it above the data in the input data word. The unit number register on the interface is loaded by an OTA and is not changed by the interface until the program does another output. The unused bits 15 and 0-9 in the unit number word output are "don't cares" so that the unit number register could be filled by an LIA followed by and OTA following an interrupt. However, this method is usually not convenient in practice because outputs are not always to the last device which interrupted.

The information output example is incomplete because of neglect of the "seeking" bit. This status bit indicates that previous data output has not been deposited into the interface memory. In other words, the registers on the interface still contain information waiting to drop into the DRUM-LIKE memory and should not be reloaded by the CPU. Reloading would destroy the previous data in transit from CPU to interface memory. The maximum amount of time which the seeking bit will be on after a set control is 70

microseconds unless a nonexistent unit number had been previously output. The logic would spin around forever looking for a memory field to put the information into. Care must be taken that the program will never try to send data or parameters to a nonexistent channel. With the seeking bit check, the example becomes:

```
        LIA  SC+1          check seeking
        SSA
        JMP  *-2           if 1 then loop until 0
        LDA CW1
        OTA SC             information out
        LDA UNIT
        ALF, ALF
        RAL, RAL
        OTA SC+1           UNIT OUT
        STC  SC
        .
        .
CW1     OCT 151202         parameter word
UNIT    OCT 5              unit being configured
SC      EQU 13B
```

Stop bits must be added to the data word before it is output. This is a slight inconvenience but yields much simpler hardware than with automatic stop bit generation and allows transmission of a long space by making stop bits zeroes. The typical data output is:

```
        LDA CHAR           7 bit ASCII character
        IOR  =B43600       add stop bits (bit 14 is the same
        JSB  OUT           send/receive bit as in the parameter word.
        .                  1=send)
        .
OUT     NOP
        LIB  SC+1          check seeking
        SSB
        JMP  *-2
        OTA SC             output data
        LDA UNIT
        ALF, ALF
        RAL, RAL
        OTA SC+1           output unit number
        STC  SC            initiate transmission
        JMP  OUT,I
        .
        .
UNIT    OCT 5
CHAR    OCT 15             carriage return
SC      EQU 13b            lower select code of data interface.
```

The output function has been placed in a subroutine, OUT, which may be used for data or parameter output as it outputs the 16 bits of the A register regardless of type.

In most cases, it is advisable to turn off the interrupt system while doing outputs because of the possibility of trying to output to the interface simultaneously from a driver initiator section, entered from the main program, and the continuator section, entered via an interrupt. The two processes, main program and interrupt service routine could possible conflict when an interrupt occurs during an output routine. For instance, the main program, getting ready to initialize a channel might have just output information and be about to output a unit number when the interrupt occurred. The interrupt service routine might proceed to

reload the information register on the interface. Most I/O interfaces don't have this type of problem because it is impossible for initiator and continuator to operate concurrently.

Input of data and status is by simple LIA's and there is no special procedure needed. After the last LIA, a clear flag executed which tells the interface that the input registers are free to be changed so another channel may load them with its data and interrupt. Up until the clear flag, other channels which were ready to interrupt had to hold off. There is no priority scheme between interface channels so that when the flag is cleared, the channel with the interrupt pending which gets there first will be the next to interrupt.

Thus a service routine to send a carriage return to a channel would execute the following I/O instructions.

```
LIA  SC+1      get the number of the interrupting channel
•
LIA  SC+1      check seeking bit
•
OTA  SC        output information
•
OTA  SC+1      output unit number
•
STC  SC        initiate transmission
•
CLF  SC        acknowledge interrupt
```

Output of data from an initiator section would be similar except there would be no clear flag command since there is no interrupt to acknowledge.

CONTROL INTERFACE

The Control Interface is an input line monitor. The CPU in requesting an interrupt is saying "Give me an interrupt when line x is value y." For instance, a request might be "interrupt when the carrier detect line for unit 5 is a 0." An interrupt will occur when this status condition occurs and bits will tell which line interrupted. In addition to giving interrupts on status, the interface supplies signals to the devices with the program setting the value such as "set Data Terminal Ready of unit 15 to a 1."

EXAMPLE: The program commands an interrupt on data set ready on.

```
        LDA CW3     request for interrupt on condition
        OTA CTL     CTL is the control board select code
        STC CTL
        •
        •
CW3     OCT 152004  scan, update, and enable status
CTL     EQU 15b
```

The scan bit, bit 15, begins an input line scan. The update bit enables bits 0 3 which is actually enabling the 4 bits to be loaded into a random access memory. During scanning, the memory spins through all of the 16 channels seeing if an input condition exists for an interrupt.

Bits 10-13 are the number 5 standing for channel 5.

Bit 2 is a 1 which indicates that status bit 1, in this case, Data Set Ready, for the channel is enabled to interrupt. Bit 3 is a 0 which disables status bit 2 from interrupting, I.E. carrier detect for this channel will not interrupt.

Bit 0 is a 0. The value of this bit is opposite that desired for the interrupt to occur, the reason being that this simplifies programming of the service routine. When the interrupt occurs, an LIA inputs all bits in the proper position to be immediately output. The enable bits read in are the same enable bits which were

output earlier. The value of bits 0 and 1, are the actual input line values for the status bits. When these are output as is, the sense of the interrupt will be reversed and an interrupt will occur on the opposite input line value.

EXAMPLE:

```
        LDA CW3         condition for interrupt
        OTA CTL         on
        STC CTL         DATA SET READY = 1
        .
        .
        interrupt occurs
        LIA CTL         input is 152405
        OTA CTL         output conditions for interrupt of
        STC CTL,C       Data Set Ready = 0
        .
        .
        interrupt occurs
        LIA CTL         input is 152404
        OTA CTL         output conditions for interrupt of
        STC CTL,C       Data Set Ready = 1
        .
        .
CW3     OCT 152004
CTL     EQU 15b
```

In this way, interrupts can be requested on both leading and trailing edges of a signal. For many types of status lines, this mode of edge sensitivity is applicable.

After each interrupt the contents of the memory on the interface MUST be changed or another interrupt will occur from the same input line condition. In other words, the request for an interrupt can be changed, disabled, or enabled only by the program. Thus another mode of operation might be to turn off the request after an interrupt.

EXAMPLE:

```
        LDA CW3         request interrupt
        OTA CTL
        STC CTL
        .
        .
        interrupt occurs
        LIA CTL         input is 152405
        AND =B177700    turn off status enable bits
        OTA CTL
        STC CTL,C
        .
CW3     OCT 152004
CTL     EQU 15b
```

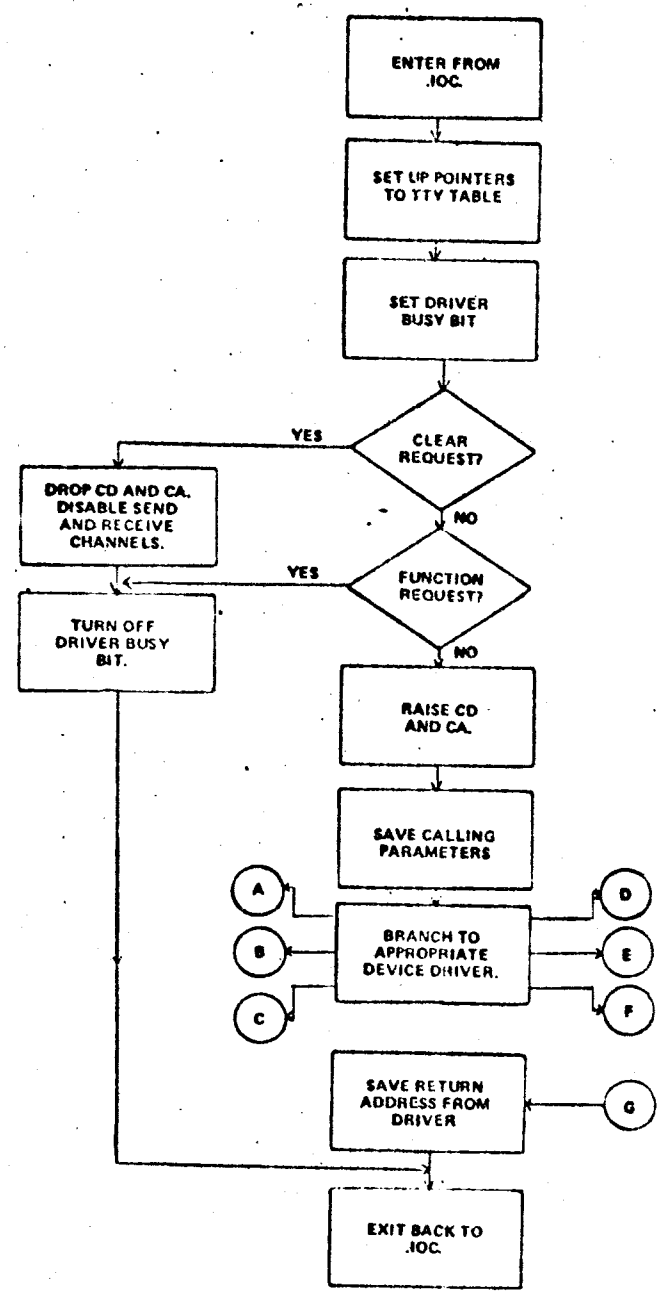
In this case, the interrupt occurs and the 4 bits, 0-3, are returned zero to the interface disabling interrupts from that channel. The program would probably initiate another status interrupt request and repeat the procedure.

```
0001                    ASMB,R,B,L,C
0003  00000                 NAM D,30      NAME OF DRIVER
0004                      ENT D,30      INITIATOR ENTRY POINT
0005                      ENT I,30      CONTINUATOR ENTRY POINT
0006                      SUP
0007*
0008*  THIS DRIVER IS NOT SUPPORTED BY HEWLETT-PACKARD
0009*
0010*  PURPOSE
0011*  THE ASYNCHRONOUS MULTIPLEXER BCS DRIVER PROVIDES A SOFTWARE
0012*  INTERFACE FOR SIXTEEN DEVICES CONNECTED TO A 2100 THROUGH A
0013*  12920 INTERFACE.
0014*
0015*  THESE DEVICES MAY BE:
0016*  HP 2600 CRT'S OPERATING AT 2400 BPS,
0017*  OR IBM 2741 SELECTRICS OPERATING THROUGH A 103 DATASET,
0018*  OR TELETYPES OPERATING HARDWIRED OR THROUGH A DATASET AT 110 BPS.

0020*  CALLING SEQUENCES
0021*
0022*     JSB .IOC,     CLEAR REQUEST (DISCONNECT DATASET),
0023*     OCT 0100N     N = LOGICAL UNIT NUMBER (0-63)
0024*     (RETURN)
0025*
0026*     JSB .IOC,     READ REQUEST
0027*     OCT 0100N
0028*     (REJECT POINT)
0029*     DEF BUFR      BUFFER ADDRESS (MAY BE INDIRECT)
0030*     DEC -LENG     BUFFER LENGTH IN CHARACTERS (NEGATIVE)
0031*     (NORMAL RETURN)
0032*                   IF REJECT THEN A=STATUS WORD,
0033*                   BIT 15=DEVICE BUSY WITH PREVIOUS REQUEST.
0034*
0035*     JSB .IOC,     WRITE REQUEST
0036*     OCT 0200N
0037*     (REJECT)
0038*     DEF BUFR
0039*     DEC -LENG
0040*     (NORMAL RETURN)
0041*
0042*     JSB .IOC,     STATUS REQUEST
0043*     OCT 0400N
0044*                   NOT BUSY-IF-A = 0
0045*                   B = NUMBER OF CHARS TRANSMITTED,(POSITIVE #)

0047*  THE DRIVER MUST BE REASSEMBLED TO CHANGE I/O SLOT POSITION OF THE
0048*  MULTIPLEXER OR THE DEVICE TYPES.
0049*
0050*  TO CHANGE I/O CONFIGURATION, ALTER THESE STATEMENTS
0051*
0052  00013         DATA  EQU 13B       SELECT CODE OF 12621 BOARDS
0053  00015         CNTL  EQU 15B       SELECT CODE OF 12922 BOARD.

0055*  TO CHANGE DEVICE TYPES, PLACE THE STARTING ADDRESSES OF
0056*  THE MINI DRIVER FOR THE DEVICE IN ARRAYS SSTRT AND RSTRT.
0057*
```



```
0058*  PROPER STARTING ADDRESSES:
0059*
0060*  HP 2600 CRT
0061*     DEF REC0 IN RSTRT,
0062*     DEF SEND0 IN SSTRT,
0063*
0064*  IBM 2741 SELECTRIC
0065*     DEF REC1 IN RSTRT,
0066*     DEF SEND1 IN SSTRT,
0067*
0068*  TCP TELETYPES:
0069*     DEF REC2 IN RSTRT,
0070*     DEF SEND2 IN SSTRT,

0072*  TO CREATE A NEW MINI DRIVER:
0073*
0074*  PLACE THE RECEIVE AND SEND STARTING ADDRESSES IN RSTRT AND SSTRT.
0075*  CALL ISFS TO BE AWAKENED AFTER THE FIRST INTERRUPT.
0076*  CALL SFS FOR SUBSEQUENT INTERRUPTS.
0077*  TERMINATE BY JUMPING TO FINI.

0079*  PREPARE CONTROL SYSTEM ONLY INITIALIZES 3 BITS FOR MULTI-
0080*  UNIT CONTROLLERS IN THE EQUIPMENT TABLE.  THE TABLE MUST BE
0081*  PATCHED TO A 4 BIT UNIT NUMBER IF PORTS 8-15 ARE TO BE USED.

0083*  EQUIPMENT TABLE FORMAT
0084*  WORD 1 BITS 0-5, LOGICAL UNIT NUMBER.
0085*  WORD 1 BITS 6-A, UNIT NUMBER, SPECIFIED AS U0, U1, ETC DURING PCS
0086*  WORD 2 BITS 2-7, DEVICE STATUS, IGNORED HERE.
0087*  WORD 2 BITS 8-13, DEVICE TYPE, IGNORED HERE.
0088*  WORD 2 BITS 14-15, BUSY BITS; BIT 15 CAUSES CALL REJECT.
0089*  WORD 3  TRANSMISSION LOG, POSITIVE # OF CHARS TRANSFERED.
0090*  WORD 4  DRIVER ADDRESS.

0092  00000         A     EQU 0         A REGISTER
0093  00001         B     EQU 1         B REGISTER
```

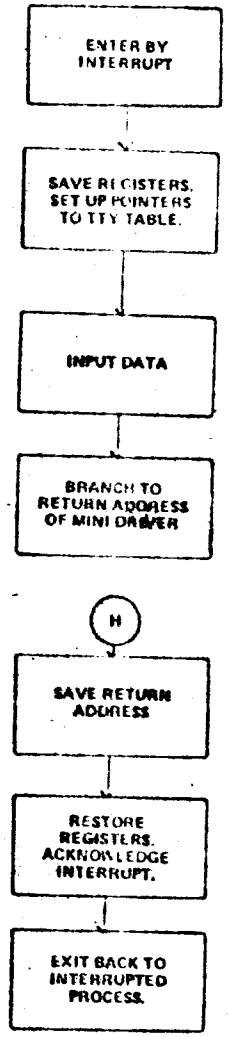
```
0056*  START OF BCS DRIVER
0057*
0058*  INITIATOR SECTION
0059*
0060  00000 000000  0,30  NOP             ENTRY FROM .IOC,
0061  00001 107300        CLF 0           PROTECT FROM CONTINUATOR
0062  00002 074062R       STA TEMP1       EQT ADDRESS
0063  00003 076463R       STB TEMP2       REQUEST CODE ADDRESS
0064  00004 060000        LDA A,I
0065  00005 001700        ALF
0066  00006 012074R       AND =B36??      UNIT X'S 2**10
0067  00007 074061R       STA UNIT
0068  00010 016072R       JSB SETUP       POINTER SETUP FOR UNIT
0069  00011 060062R       LDB TEMP1       ADDRESS OF EQT WORD 1
0070  00012 006004        INB
0071  00013 175400R       STB EQT,I       SAVE EQT ADDRESS IN TABLE
0072  00014 103001        LDA B,I
0073  00015 052075R       IOR =B??,000    SET DRIVER BUSY BIT IN
0074  00016 170001        STA B,I         EQUIPMENT TABLE,
0075  00017 162063R       LDA TEMP2,I     WORD 1 OF USER CALL,
0076  00020 001700        ALF
0077  00021 012076R       AND =B3         LIMIT A TO 0,1,2 OR 3
0078  00022 002003        SZA,RSS         IF 0,
0079  00023 026051R       JMP CLR           CLEAR,
0080  00024 052077R       CPA =B3
0081  00025 026060R       JMP FUNC        IF 3 THEN ILLEGAL FUNCTION CALL.
0082
0083  00026 046472R       LDB CLAY        IF 1,READ, OR 2,WRITE,
0084  00027 044000        ADB A           INDEX TO RSTRT OR SSTRT,
0085  00030 103001        LDB B,I         GET DRIVER ADDRESS
0086  00031 062077R       LDA =B40360     TURN ON CD AND CA
0087  00032 016444R       JSB COUT        OUTPUT TO DATASET BOARD
0088  00033 036463R       ISZ TEMP2       INDEX TO REJECT POINT
0089  00034 036463R       ISZ TEMP2       INDEX TO BUFFER ADDRESS POINT
0090  00035 062463R       LDA TEMP2       GET ADDRESS OF ADDRESS
0091  00036 160000        LDA A,I         RETRIEVE BUFFER ADDRESS
0092  00037 001275        RAL,CLE,SLA,ERA  DESCEND INDIRECT CHAIN
0093  00040 026037R       JMP *-2
0094  00041 001200        RAL             CONVERT TO BYTE POINTER
0095  00042 172466R       STA BUFA,I      SAVE ADDRESS IN TTY TABLE
0096  00043 036463R       ISZ TEMP2       INDEX TO LENTH PARAMETER,
0097  00044 160463R       LDA TEMP2,I     RETRIEVE BUFFER LENGTH
0098  00045 002021        SSA,RSS         IF NOT 2'S COMPLEMENT CHARS
0099  00046 003024        CMA,INA         THEN CHANGE TO 2'S COMPLEMENT,
0100  00047 172467R       STA BUFL,I      SAVE IN TABLE
0101  00050 172470R       STA BUFLI,I     SAVE INITIAL BUFFER LENGTH
0102  00051 000100        HLT             INSERT A HALT FOR DEBUGGING,
0103  00052 124001        JMP B,I         GO TO MINI DRIVER
0104
0105  00053 000000  ISFS  NOP             INITIAL WAIT FOR INTERRUPT RETURN
0106  00054 072473R       LDA ISFS
0107  00055 172471R       STA STATE,I     SAVE RETURN ADRESS
0108  00056 002400  GOOD  CLA             SIGNAL SUCCESS TO .IOC,
0109  00057 103200        STC 2           RE-ENABLE CONTINUATOR INTERRUPTS
0110  00060 126000R       JMP D,30,I      DONE INITIATION
```



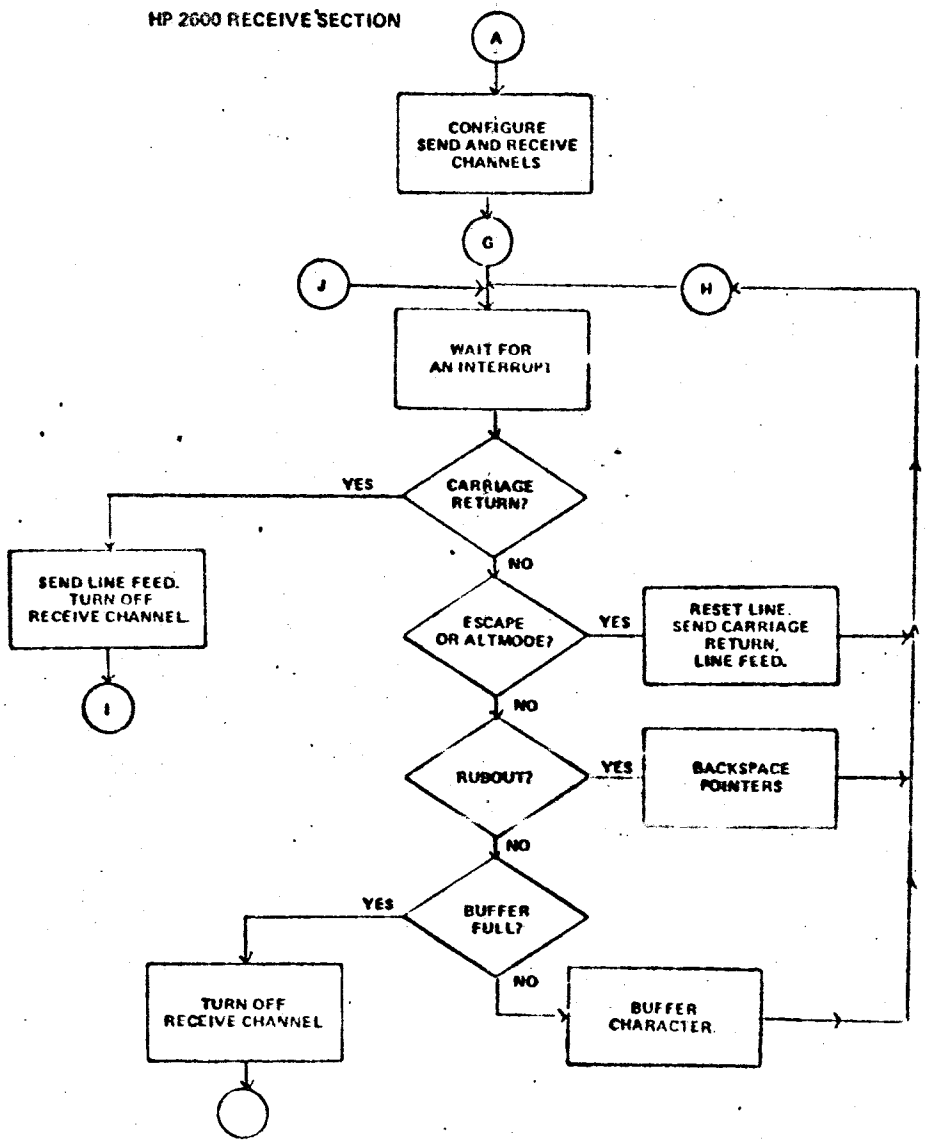
```
0151  00061 062700R CLR   LDA =B40300  HANG UP DATA SETS.
0152  00062 016444R        JSB COUT    OUTPUT TO DATA SET BOARD
0153  00063 016436R        JSB CRF     TURN OFF RECEIVE CHANNEL
0154  00064 062721R        LDA =B14000 TURN OFF SEND CHANNEL
0155  00065 016423R        JSB OUT
0156  00066 016111R FUNC   JSB DONE    TURN OFF DRIVER BUSY BIT.
0157  00067 026056R        JMP GOOD    EXIT

0159  00070 000000  SETUP  NOP         FORM POINTERS TO TABLE
0160  00071 001727         ALF,ALF     UNIT #'S 2**2
0161  00072 001000         ALS         UNIT #'S 2**3
0162  00073 042473R        ADA TBLE    POINT TO TTY TABLE
0163  00074 072465R        STA BUFA    SAVE POINTERS
0164  00075 002004         INA
0165  00076 072466R        STA BUFL
0166  00077 002004         INA
0167  00100 072467R        STA BUFLI
0168  00101 002004         INA
0169  00102 072470R        STA EQT
0170  00103 002004         INA
0171  00104 072471R        STA STATE
0172  00105 002004         INA
0173  00106 072472R        STA DLAY
0174  00107 126070R        JMP SETUP,I

0176  00110 000000  DONE   NOP         TURN OFF BUSY BIT IN EQT
0177  00111 166470R        LDB EQT,I   B IS POINTER TO EQT
0178  00112 160001         LDA B,I
0179  00113 012772R        AND =B77777 SET BIT 15 TO 0
0180  00114 170001         STA B,I
0181  00115 126110R        JMP DONE,I

0183  00116 000000  AMT    NOP         COMPUTE CHARACTERS TRANSFERED (+)
0184  00117 162467R        LDA BUFLI,I NUMBER OF CHARACTERS EQUALS
0185  00120 002004         CMA,INA     INITIAL BUFFER LENGTH,
0186  00121 142466R        ADA BUFL,I  MINUS THE BUFFER LENGTH NOW.
0187  00122 126116R        JMP AMT,I

0189  00123 016111R FINI   JSB DONE    SET DRIVER NOT BUSY
0190  00124 016116R        JSB AMT     COMPUTE NUMBER OF CHARACTERS
0191  00125 166470R        LDB EQT,I   AND PUT IN TRANSMISSION LOG
0192  00126 002004         INA
0193  00127 170001         STA B,I
0194  00130 002400         CLA         FINAL EXIT FROM DRIVER.
0195  00131 026156R        JMP EXIT1
```

```
0197*  CONTINUATOR SECTION
0198*  CONTROL COMES HERE FROM JSB IN TRAP CELL
0199*
0200  00132 000000  I,SC  NOP             ENTRY POINT
0201  00133 072402R       STA TEMP1      SAVE A
0202  00134 076403R       STB TEMP2      SAVE B
0203  00135 001520        ERA,ALS
0204  00136 102201        SOC
0205  00137 002004        INA
0206  00140 072404R       STA TEMP3      SAVE E AND O
0207  00141 102514        LIA DATA+1     GET UNIT NUMBER
0208  00142 012075R       AND =B36000    UNIT X'S 2**10
0209  00143 072401R       STA UNIT
0210  00144 016172R       JSB SETUP      POINTER SETUP FOR UNIT
0211  00145 106471R       LDB STATE,I    B IS MINI DRIVER P REGISTER
0212  00146 102513        LIA DATA       GET CHARACTER
0213  00147 012077R       AND =B177      ISOLATE
0214  00150 000000        NOP            INSERT A HALT FOR DEBUGGING,
0215  00151 006003        SZA,RSS        LEGAL INTERRUPT?
0216  00152 026157R       JMP EXIT2      YES SO EXIT.
0217  00153 124001        JMP B,I        BRANCH TO DEVICE DRIVER

0219  00154 000000  SFS   NOP            RETURN HERE
0220  00155 002154R       LDA SFS        TO WAIT FOR INTERRUPT.
0221  00156 172471R EXIT1 STA STATE,I    SAVE P REGISTER
0222  00157 076404R EXIT2 LDA TEMP3      RESTORE E AND O
0223  00160 103101        CLC
0224  00161 002036        SLA,ELA
0225  00162 103121        STF 1
0226  00163 076402R       LDA TEMP1      RESTORE A
0227  00164 006403R       LDB TEMP2      RESTORE B
0228  00165 103113        CLF DATA       ACKNOWLEDGE INTERRUPT
0229  00166 126132R       JMP I,SC,I     RETURN TO MAIN PROCESS
```


CONTINUATOR SECTION

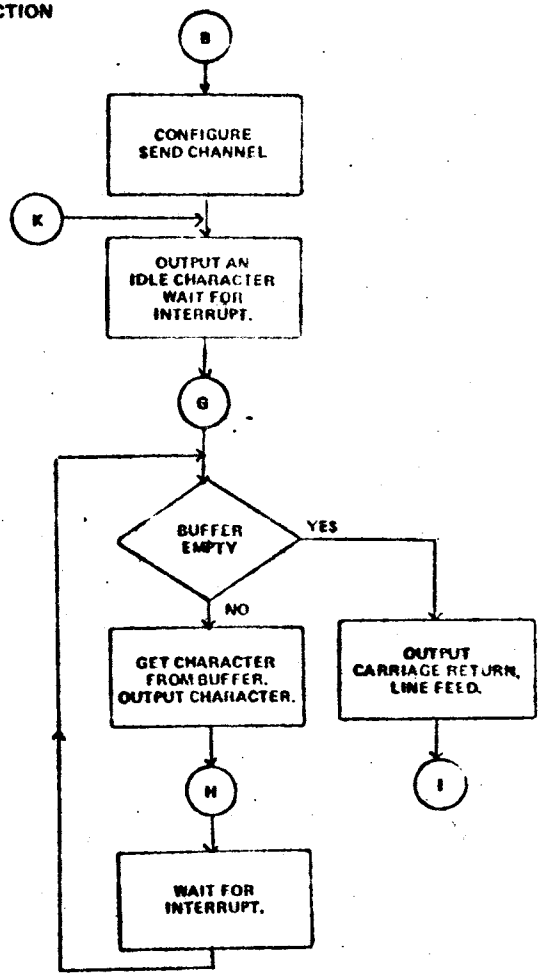


```
2231*  CRT MINI DRIVER RECEIVE SECTION
2232*
2233*  INITIATOR SECTION
2234*
2235*  1)  IF CR THEN LF AND DONE.
2236*  2)  IF ESC THEN CR,LF,AND ELIMINATE LINE.
2237*  3)  IF ALTMD THEN CR, LF, AND ELIMINATE LINE.
2238*  4)  IF RUBOUT THEN BACKSPACE.
2239*  5)  IF FULL BUFFER THEN DONE.
2240*
2241  00167 062704R RECV  LDA =B132405 2400 BAUD WITH ECHO ON
2242  00170 016423R       JSB OUT       OUTPUT TO CHANNEL
2243  00171 062706R       LDA =B160405 SEND CHANNEL PARAMETERS
2244  00172 016423R RV    JSB OUT       OUTPUT TO CHANNEL
2245  00173 016753R       JSB ISFS      WAIT FOR FIRST INTERRUPT

2247*  CONTINUATOR SECTION
2248*
2249  00174 052457R IN0   CPA CR        RETURN HERE FROM INTERRUPT
2250  00175 026222R       JMP ENDR0     CR SO RECEPTION IS DONE
2251  00176 052454R       CPA ESC
2252  00177 026210R       JMP RUB0      ESCAPE SO RESET POINTERS
2253  00200 052456R       CPA ALTM
2254  00201 026210R       JMP RUB0      ALTMODE IS SAME AS ESCAPE
2255  00202 052455R       CPA RUB
2256  00203 026220R       JMP BACK0     RUBOUT SO BACKSPACE
2257  00204 016334R       JSB PUT       PLACE IN BUFFER
2258  00205 026227R       JMP FULL0     BUFFER IS FULL SO EXIT
2259  00206 016154R LO    JSB SFS       WAIT FOR THE NEXT CHARACTER
2260  00207 026174R       JMP IN0

2262  00210 016405R RUB0  JSB RESET     WIPE OUT LINE
2263  00211 052457R       LDA CR
2264  00212 016417R       JSB CHAR      OUTPUT CARRIAGE RETURN
2265  00213 016154R       JSB SFS       WAIT FOR INTERRUPT
2266  00214 052460R       LDA LF
2267  00215 016417R       JSB CHAR      OUTPUT LINE FEED
2268  00216 016154R       JSB SFS
2269  00217 026206R       JMP LO        CONTINUE RECEIVING

2271  00220 016373R BACK0 JSB BACK      BACKSPACE POINTERS
2272  00221 026206R       JMP LO        CONTINUE RECEIVING

2274  00222 016434R ENDR0 JSB OFF       TURN OFF RECEIVE CHANNEL
2275  00223 052460R       LDA LF
2276  00224 016417R       JSB CHAR      OUTPUT LINE FEED
2277  00225 016154R       JSB SFS       WAIT FOR INTERRUPT
2278  00226 026123R       JMP FINI      EXIT

2280  00227 016434R FULL0 JSB OFF       BUFFER FULL EXIT-NO CR,LF
2281  00230 026123R       JMP FINI      EXIT
```



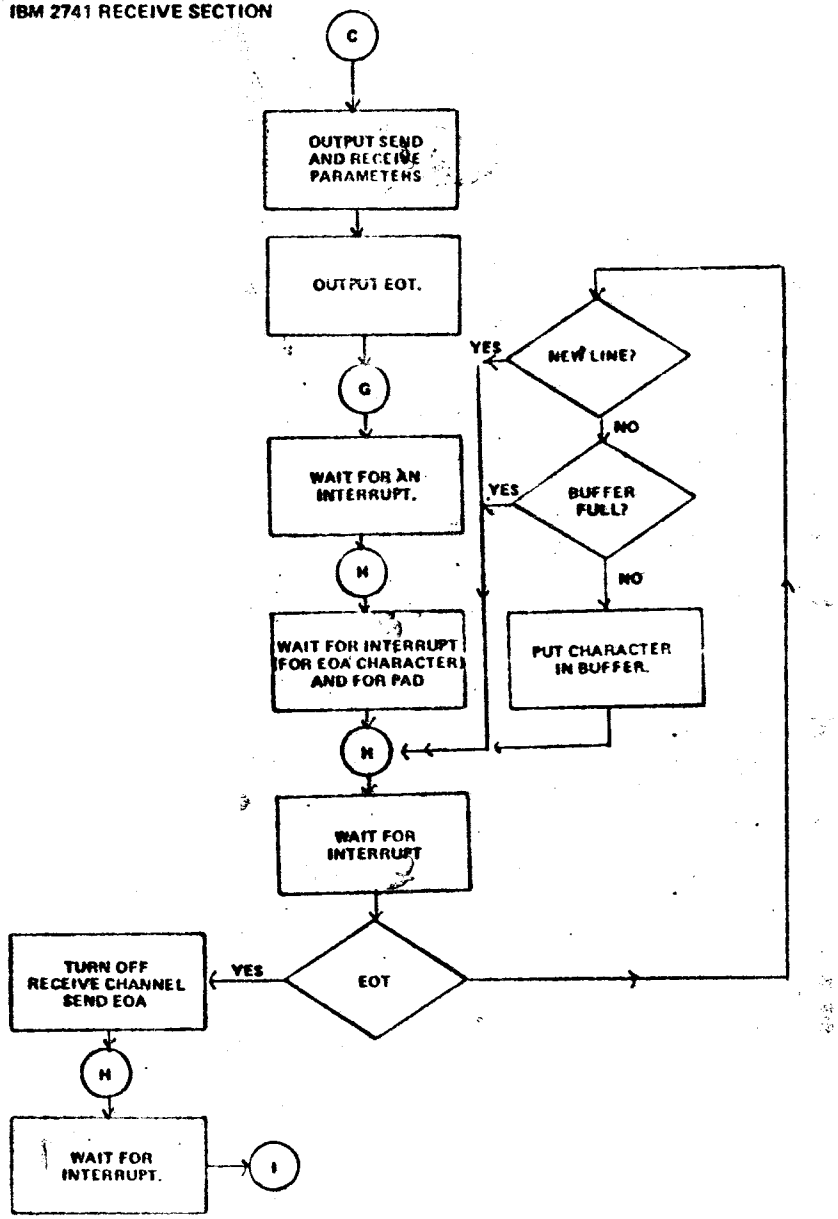
```
02283*  CRT SEND MINI DRIVER
02284*
02285*  INITIATOR SECTION
02286*
02287*  1)  IF BUFFER EMPTY THEN CR,LF AND DONE,
02288*
02289  02231 062726R SENDR LDA =B162405 SEND CRT SPEED
02290  02232 016423R (SJ)    JSB OUT      OUTPUT PARAMETERS
02291  02233 016440R         JSB NULL     SEND SYNCHRONIZING IDLE
02292  02234 016753R         JSB ISFS     WAIT FOR INITIAL INTERRUPT

02294*  CONTINUATOR SECTION
02295*
02296  02235 016354R CUTB    JSB GET      GET A CHAR FROM BUFFER
02297  02236 026242R         JMP ENDSB    IF BUFFER EMPTY THEN EXIT
02298  02237 016417R         JSB CHAR     SEND CHAR
02299  02240 016154R         JSB SFS      WAIT FOR INTERRUPT
02300  02241 026235R         JMP CUTB     SEND NEXT CHARACTER

02302  02242 062457R ENDSB   LDA CR       END LINE-SEND CR,LF
02303  02243 016417R         JSB CHAR
02304  02244 016154R         JSB SFS
02305  02245 026222R         JMP ENDR2    SEND LF AND EXIT
```


HP 2600 SEND SECTION



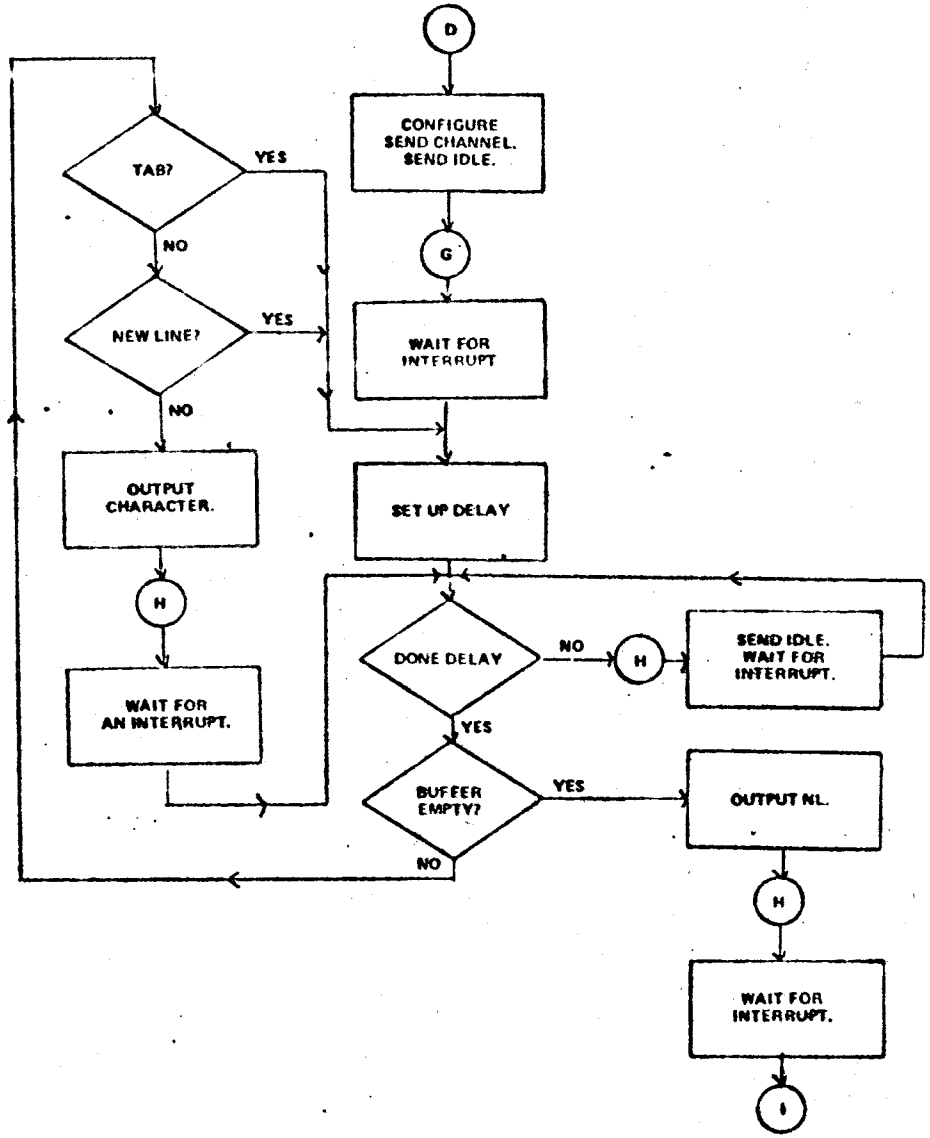
```
0307*  IBM 2741 MINI DRIVER RECEIVE SECTION
0308*
0309*  INITIATOR SECTION
0310*
0311*  1)  SEND EOT
0312*  2)  RECEIVE ECA,
0313*  3)  RECEIVE PAD,
0314*  4)  IF EOT THEN SEND EOA AND EXIT,
0315*  5)  IF BUFFER FULL THEN IGNORE CHARACTER,
0316*
0317  00246 C42726R REC1  LDA =B162152 134,5 BAUD SEND PARAMETERS
0318  00247 C16423R       JSB OUT      OUTPUT TO CHANNEL
0319  00250 C42727R       LDA =B125152 134,5 BAUD RECEIVE PARAMETERS
0320  00251 C16423R       JSB OUT
0321  00252 C02451R       LDA EOT      UNLOCK KEYBOARD
0322  00253 C16417R       JSB CHAR
0323  00254 C16453R       JSB ISFS     WAIT FOR COMPLETION OF EOT

0325*  CONTINUATOR SECTION
0326*
0327  00255 C16154R       JSB SFS      RECEPTION OF CIRCLE D
0328  00256 C16154R       JSB SFS      RECEPTION OF PAD

0330  00257 C02451R IN1   CPA EOT      IF DONE RECEPTION
0331  00260 C02657R       JMP ENDR1    THEN TERMINATE
0332  00261 C02453R       CPA NL       IGNORE NEW LINES
0333  00262 C02665R       JMP L1
0334  00263 C16334R       JSB PUT      BUFFER CHARACTER IF ROOM
0335  00264 000000        NOP          IF BUFFER FULL THEN IGNORE
0336  00265 C16154R L1    JSB SFS      WAIT FOR INTERRUPT
0337  00266 C02657R       JMP IN1

0339  00267 C16434R ENDR1 JSB OFF      TURN OFF RECEIVE CHANNEL
0340  00270 C02452R       LDA EOA      PUT TERMINAL
0341  00271 C16417R       JSB CHAR       IN RECEIVE MODE
0342  00272 C16154R       JSB SFS      WAIT UNTIL TRANSMITTED
0343  00273 C02612R       JMP FINI     EXIT
```


IBM 2741 RECEIVE SECTION



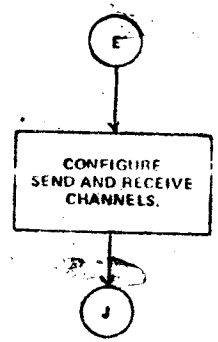
```
                        ASYNCHRONOUS MULTIPLEXER DRIVER

0345*   IBM 2741 SELECTRIC MINI DRIVER SEND SECTION
0346*
0347*   INITIATOR SECTION
0348*
0349*   1)  SET 13 CHARACTER DELAY.
0350*   2)  IF TAB THEN SEND TAB AND 13 CHARACTER DELAYS.
0351*   3)  IF NL THEN SEND NL AND 13 CHARACTER DELAY.
0352*   4)  IF BUFFER EMPTY THEN SEND NL AND EXIT.
0353*
0354  00224 042224R SEND1 LDA =B1647152 134.5 BAUD SEND PARAMETERS
0355  00225 016423R       JSB OUT
0356  00226 016446R       JSB NULL      TRIGGER WITH IDLE
0357  00227 016053R       JSB ISFS      WAIT FOR INITIAL INTERRUPT

0358*   CONTINUATOR SECTION
0359*
0360  00300 042712R WAIT1 LDA =D-12     SET UP FOR DELAY
0361  00301 172472R       STA DLAY,I    OF 12 CHARACTER TIMES
0362  00302 016446R WAIT1 JSB NULL      SPIN UNTIL DELAY IS OVER
0363  00303 016154R       JSB SFS
0364  00304 130472R       ISZ DLAY,I    DONE DELAY?
0365  00305 026302R       JMP WAIT1     NO, REPEAT.

0366  00306 016354R OUT1  JSB GET       GET CHARACTER FROM BUFFER
0367  00307 026317R       JMP ENDS1     NO MORE CHARS SO EXIT
0368  00310 042456R       CPA TAB       TABULATION CHARACTER?
0369  00311 026323R       JMP L2        YES SO DELAY
0370  00312 042453R       CPA NL        DELAY FOR NEW LINE ALSO
0371  00313 026323R       JMP L2
0372  00314 016417R       JSB CHAR      OUTPUT THE CHARACTER
0373  00315 016154R       JSB SFS
0374  00316 026306R       JMP OUT1      SEND NEXT CHARACTER

0375  00317 042453R ENDS1 LDA NL        LINE END, SEND CR+LF
0376  00320 016417R       JSB CHAR
0377  00321 016154R       JSB SFS
0378  00322 026123R       JMP FINI      EXIT

0379  00323 016417R L2    JSB CHAR      OUTPUT CHARACTER THEN DELAY
0380  00324 016154R       JSB SFS
0381  00325 026302R       JMP WAIT1     GO OUTPUT DELAY
```

IBM 2741 SEND SECTION

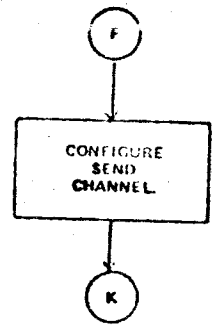


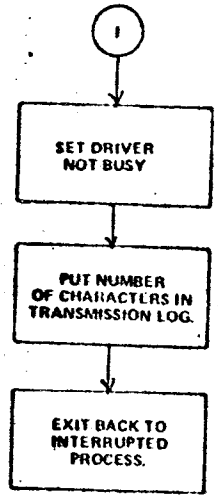
```
PAGE 0011 #01  BSC ASYNCHRONOUS MULTIPLEXER DRIVER

0384*  TTY MINI DRIVER RECEIVE SECTION
0385*
0390*  INITIATOR SECTION.
0391*
0392*  1)  IF CR THEN SEND LF AND EXIT.
0393*  2)  IF ESC THEN SEND CR,LF, AND ELIMINATE LINE.
0394*  3)  IF ALT-MD THEN CR, LF, AND ELIMINATE LINE.
0395*  4)  IF RUBOUT THEN BACKSPACE.
0396*  5)  IF FULL BUFFER THEN EXIT.
0397*
0398  00326 C627118 REC2  LDA =H131202 110 BAUD RECEIVE PARAMETERS
0399  00327 C161218       JSB OUT       OUTPUT TO CHANNEL
0400  00330 C627104       LDA =B161202 110 BAUD SEND PARAMETERS
0401  00331 C261728       JMP RR        REST IS SAME AS CRT

0403*  TTY  MINI DRIVER SEND SECTION
0404*
0405*  INITIATOR SECTION.
0406*
0407*  1)  IF BUFFER EMPTY THEN SEND CR, LF, AND EXIT.
0408*
0409  00332 2627128 SEND2 LDA =B161202 110 BAUD TTY SEND PARAMETERS
0410  00333 C262328       JMP SD        SAME AS CRT.
```


TELETYPE RECEIVE SECTION



TELETYPE SEND SECTION





```
0412*  BUFFER MANAGEMENT ROUTINES
0413*
0414  00334 000000  PUT   NOP             A=CHARACTER TO BE BUFFERED
0415  00335 156456R       LDB BUFL,I
0416  00336 006721        SSB,RSS         IF BUFFER IS FULL ALREADY
0417  00337 126334R       JMP PUT,I         THEN EXIT P+1
0418  00340 156455R       LDB BUFA,I      GET BYTE ADDRESS
0419  00341 004065        CLE,ERB         CONVERT TO WORDS
0420  00342 002041        SEZ,RSS         EVEN OR ODD BYTE?
0421  00343 025346R       JMP EVEN        EVEN BYTE
0422  00344 132021        IOR B,I         ODD BYTE
0423  00345 002021        RSS
0424  00346 001727  EVEN  ALF,ALF         POSITION IN UPPER HALF
0425  00347 172021        STA B,I         STORE IN BUFFER
0426  00350 136455R       ISZ BUFA,I      INCREMENT BYTE ADDRESS
0427  00351 136456R       ISZ BUFL,I      INCREMENT LENGTH
0428  00352 026334R       ISZ PUT         IF BUFFER IS NOT FULL, RETURN P+2
0429  00353 126334R       JMP PUT,I

0431  00354 000000  GET   NOP             GET A CHARACTER FROM BUFFER
0432  00355 156456R       LDA BUFL,I      IF BUFFER EMPTY ALREADY
0433  00356 002721        SSA,RSS
0434  00357 126354R       JMP GET,I       THEN RETURN P+1
0435  00360 156455R       LDB BUFA,I      GET BYTE ADDRESS
0436  00361 004065        CLE,ERB         CONVERT TO WORD ADDRESS
0437  00362 162021        LDA B,I         GET CHARACTER WORD
0438  00363 002041        SEZ,RSS         UPPER OR LOWER HALF?
0439  00364 001727        ALF,ALF         UPPER
0440  00365 012733R       AND =B177       ISOLATE
0441  00366 136455R       ISZ BUFA,I      INCREMENT BYTE POINTER
0442  00367 136456R       ISZ BUFL,I      INCREMENT CHARACTER COUNT
0443  00370 000000        NOP
0444  00371 026354R       ISZ GET         RETURN TO P+2 SINCE NOT EMPTY.
0445  00372 126354R       JMP GET,I       RETURN WITH CHARACTER IN A

0447  00373 000000  BACK  NOP             BACKSPACE POINTER IN BUFFER
0448  00374 156456R       LDA BUFL,I
0449  00375 152572R       CPA BUFL,I      IF ALREADY AT BUFFER BEGINNING
0450  00376 176373R       JMP BACK,I      THEN DON'T DECREMENT
0451  00377 042713R       ADA =D-1
0452  00400 172456R       STA BUFL,I      OTHERWISE, DECREMENT LENGTH,
0453  00401 156455R       LDA BUFA,I
0454  00402 042713R       ADA =D-1
0455  00403 172455R       STA BUFA,I      AND DECREMENT ADDRESS
0456  00404 176373R       JMP BACK,I

0458  00405 000000  RESET NOP             WIPE OUT A LINE
0459  00406 016116R       JSB AMT         GET NUMBER OF CHARS TRANSMITTED
0461  00407 003004        CMA,INA
0461  00410 156456R       LDB BUFL,I
0462  00411 040000        ADB A           SUBTRACT FROM LENGTH (-)
0463  00412 176456R       STB BUFL,I
0464  00413 156455R       LDB BUFA,I      AND
0465  00414 040000        ADB A           ADDRESS
0466  00415 176455R       STB BUFA,I
0467  00416 126405R       JMP RESET,I
```



```
0469*  CHARACTER AND PARAMETER OUTPUT ROUTINES
0470*
0471  00417 000000  CHAR  NOP             OUTPUT THE CHARACTER IN A
0472  00420 032714R       IOR =B43600     ADD STOP BITS
0473  00421 016423R       JSB OUT         OUTPUT TO MULTIPLEXER
0474  00422 126417R       JMP CHAR,I

0476  00423 000000  OUT   NOP             OUTPUT 16 BITS TO MULTIPLEXER
0477  00424 106514        LIB DATA+1      CHECK SEEKING
0478  00425 006020        SSB
0479  00426 026424R       JMP *-2         IF NOT OK THEN WAIT
0480  00427 102613        OTA DATA        SEND OUT DATA
0481  00430 062461R       LDA UNIT        UNIT IS BITS 12-13.
0482  00431 102414        OTA DATA+1      SEND OUT UNIT
0483  00432 102713        STC DATA        INITIATE TRANSFER
0484  00433 126423R       JMP OUT,I

0485  00434 000000  OFF   NOP             TURN OFF A RECEIVE CHANNEL
0487  00435 062675R       LDA =B100000    ENABLE OFF, ECHO OFF
0488  00436 016423R       JSB OUT         OUTPUT TO MULTIPLEXER
0489  00437 126434R       JMP OFF,I

0491  00440 000000  NULL  NOP             SEND A TIME DELAY CHAR
0492  00441 052722R       LDA =B77777     SYNC BIT IS 1
0493  00442 016423R       JSB OUT
0494  00443 126440R       JMP NULL,I

0496  00444 000000  COUT  NOP             OUTPUT TO CONTROL BOARD
0497  00445 062461R       IOR UNIT        ADD UNIT NUMBER , BITS 10-13.
0498  00446 102615        OTA CNTL        OUTPUT
0499  00447 126444R       JMP COUT,I
```

```
2531*  CONSTANTS
2532*
2533   00452 000067  TAB    OCT 57          SELECTRIC TAB CODE
2534   00453 000064  EOA    OCT 64          SELECTRIC CIRCLE D CODE.
2535   00454 000175  NL     OCT 175         SELECTRIC NEW LINE CODE
2536   00455 000176  ESC    OCT 176         ASCII ESCAPE CODE
2537   00456 000177  RUB    OCT 177         ASCII RUBOUT CODE
2538   00457 000033  ALTM   OCT 33          ASCII ALTERNATE MODE CODE
2539   00460 000015  CR     OCT 15          ASCII CARRIAGE RETURN
2540   00461 000012  LF     OCT 12          ASCII LINE FEED

2541*  TEMPORARY STORAGE
2542*
2543   00462 000000  UNIT   NOP             UNIT NUMBER OF INTERRUPTING UNIT
2544   00463 000000  TEMP1  NOP             SAVE A OR  EQT POINTER
2545   00464 000000  TEMP2  NOP             SAVE B OR USER CALL POINTER.
2546   00465 000000  TEMP3  NOP             SAVE E AND O

2547*  TABLE OF POINTERS INTO THE CURRENT TTY TABLE
2548*
2549   00466 000000  BUFA   NOP             BYTE POINTER
2550   00467 000000  BUFC   NOP             CHARACTER COUNT
2551   00470 000000  BUFLI  NOP             INITIAL CHARACTER COUNT
2552   00471 000000  EQT    NOP             EQUIPMENT TABLE WORD 2
2553   00472 000000  STATE  NOP             MINI DRIVER P REGISTER
2554   00473 000000  DLAY   NOP             DELAY COUNT

2555*  TELETYPE TABLES
2556*
2557*  BUFA
2558*  INPUT.  POINTER TO THE NEXT BYTE TO BE INPUT.
2559*  OUTPUT.  POINTER TO THE NEXT CHARACTER TO BE SENT.
2560*
2561*  BUFL
2562*  INPUT.  2'S COMPLEMENT OF BYTES LEFT IN BUFFER.
2563*  OUTPUT.  2'S COMPLEMENT OF CHARACTERS TO BE TRANSMITTED.
2564*
2565*  BUFI
2566*  INITIAL BUFFER SIZE, 2'S COMPLEMENT OF BYTES.
2567*
2568*  EQT
2569*  ADDRESSS OF THE SECOND WORD IN EQT FOR UNIT
2570*
2571*  STATE
2572*  THE SAVED P REGISTER FROM THE MINI DRIVER.
2573*
2574*  DLAY
2575*  2'S COMPLEMENT OF DELAY CHARACTERS REMAINING.
2576*
2577*  RSTRT
2578*  ADDRESS OF RECEIVE ENTRY POINT IN MINI DRIVER.
2579*
2580*  SSTRT
2581*  ADDRESS OF SEND ENTRY POINT IN MINI DRIVER.
```



```
2582   00473 000474R TBLE   DEF *+1
2583   00474 000000  TTY0   OCT 0,0,0,0,0,0,0  BUFA,BUFL,BUFI,EQT,STATE,DLAY
2584   00475 000326R        DEF REC2      110 BAUD TTY RECEIVE
2585   00476 000332R        DEF SEND2     110 BAUD TTY SEND
2586   00477 000000  TTY1   OCT 0,0,0,0,0,0  BUFA,BUFL,BUFI,EQT,STATE,DLAY
2587   00500 000326R        DEF REC2      110 BAUD TTY RECEIVE
2588   00501 000332R        DEF SEND2     110 BAUD TTY SEND
2589   00502 000000  TTY2   OCT 0,0,0,0,0,0  BUFA,BUFL,BUFI,EQT,STATE,DLAY
2590   00503 000326R        DEF REC2      110 BAUD TTY RECEIVE
2591   00504 000332R        DEF SEND2     110 BAUD TTY SEND
2592   00505 000000  TTY3   OCT 0,0,0,0,0,0  BUFA,BUFL,BUFI,EQT,STATE,DLAY
2593   00506 000326R        DEF REC2      110 BAUD TTY RECEIVE
2594   00507 000332R        DEF SEND2     110 BAUD TTY SEND
2595   00510 000000  TTY4   OCT 0,0,0,0,0,0  BUFA,BUFL,BUFI,EQT,STATE,DLAY
2596   00511 000246R        DEF REC1      134.5 BAUD 2741 RECEIVE
2597   00512 000274R        DEF SEND1     134.5 BAUD 2741 SEND
2598   00513 000000  TTY5   OCT 0,0,0,0,0,0  BUFA,BUFL,BUFI,EQT,STATE,DLAY
2599   00514 000246R        DEF REC1      134.5 BAUD 2741 RECEIVE
2600   00515 000274R        DEF SEND1     134.5 BAUD 2741 SEND
2601   00516 000000  TTY6   OCT 0,0,0,0,0,0  BUFA,BUFL,BUFI,EQT,STATE,DLAY
2602   00517 000167R        DEF RECO      2400 BAUD CRT RECEIVE
2603   00520 000231R        DEF SEND0     2400 BAUD CRT SEND
2604   00521 000000  TTY7   OCT 0,0,0,0,0,0  BUFA,BUFL,BUFI,EQT,STATE,DLAY
2605   00522 000167R        DEF RECO      2400 BAUD CRT RECEIVE
2606   00523 000231R        DEF SEND0     2400 BAUD CRT SEND
2607   00524 000000  TTY8   OCT 0,0,0,0,0,0  BUFA,BUFL,BUFI,EQT,STATE,DLAY
2608   00525 000167R        DEF RECO      2400 BAUD CRT RECEIVE
2609   00526 000231R        DEF SEND0     2400 BAUD CRT SEND
2610   00527 000000  TTY9   OCT 0,0,0,0,0,0  BUFA,BUFL,BUFI,EQT,STATE,DLAY
2611   00530 000167R        DEF RECO      2400 BAUD CRT RECEIVE
2612   00531 000231R        DEF SEND0     2400 BAUD CRT SEND
2613   00532 000000  TTY10  OCT 0,0,0,0,0,0  BUFA,BUFL,BUFI,EQT,STATE,DLAY
2614   00533 000167R        DEF RECO      2400 BAUD CRT RECEIVE
2615   00534 000231R        DEF SEND0     2400 BAUD CRT SEND
2616   00535 000000  TTY11  OCT 0,0,0,0,0,0  BUFA,BUFL,BUFI,EQT,STATE,DLAY
2617   00536 000167R        DEF RECO      2400 BAUD CRT RECEIVE
2618   00537 000231R        DEF SEND0     2400 BAUD CRT SEND
2619   00540 000000  TTY12  OCT 0,0,0,0,0,0  BUFA,BUFL,BUFI,EQT,STATE,DLAY
2620   00541 000167R        DEF RECO      2400 BAUD CRT RECEIVE
2621   00542 000231R        DEF SEND0     2400 BAUD CRT SEND
2622   00543 000000  TTY13  OCT 0,0,0,0,0,0  BUFA,BUFL,BUFI,EQT,STATE,DLAY
2623   00544 000167R        DEF RECO      2400 BAUD CRT RECEIVE
2624   00545 000231R        DEF SEND0     2400 BAUD CRT SEND
2625   00546 000000  TTY14  OCT 0,0,0,0,0,0  BUFA,BUFL,BUFI,EQT,STATE,DLAY
2626   00547 000167R        DEF RECO      2400 BAUD CRT RECEIVE
2627   00550 000231R        DEF SEND0     2400 BAUD CRT SEND
2628   00551 000000  TTY15  OCT 0,0,0,0,0,0  BUFA,BUFL,BUFI,EQT,STATE,DLAY
2629   00552 000167R        DEF RECO      2400 BAUD CRT RECEIVE
2630   00553 000231R        DEF SEND0     2400 BAUD CRT SEND
2631                        END 0,30
** NO ERRORS.
```

*B10000	00112	00497					
*B12015	00319						
*B13040	00241						
*B13120	00398						
*B14000	00154						
*B16015	00317	00354					
*B16040	00243	00289					
*B16120	00402	00429					
*B177	00213	00440					
*B3	00116	00119					
*B36000	00135	00229					
*B40320	00151						
*B40360	00125						
*B43520	00472						
*B77777	00179	00492					
*D-1	00451	00454					
*D-12	00361						
A 00092	00123	00123	00138	00462	00465		
ALTM 00509	00253						
AMT 00183	00187	00190	00459				
B 00093	00111	00113	00124	00142	00178	00180	
	00193	00217	00422	00425	00437		
BACK 00447	00271	00450	00456				
BACKO 00271	00296						
BUFA 00522	00134	00163	00418	00426	00435	00441	
	00453	00455	00464	00466			
BUFL 00523	00139	00165	00186	00415	00427	00432	
	00442	00448	00452	00461	00463		
BUFLI 00524	00142	00167	00184	00449			

CHAR 00471	00264	00267	00276	00298	00303	00322	
	00341	00374	00379	00383	00474		
CLR 00151	00118						
CNTL 00053	00498						
COUT 00496	00126	00152	00499				
CR 00510	00249	00253	00302				
D.3P 00099	00204	00149	00606				
DATA 00052	00207	00212	00228	00477	00480	00482	
00483							
DLAY 00527	00122	00173	00362	00365			
DONE 00176	00156	00191	00189				
ENDR0 00274	00252	00305					
ENDR1 00330	00331						
ENDS0 00302	00297						
ENDS1 00378	00369						
EOA 00505	00342						
EOT 00524	00321	00330					
EOT 00525	00110	00169	00177	00151			
ESC 00507	00251						
EVEN 00424	00421						
EXIT1 00221	00195						
EXIT2 00222	00216						
FINI 00189	00278	00281	00343	00361			
FULL0 00280	00258						
FUNC 00154	00120						
GET 00431	00296	00368	00434	00444	00445		
GOOD 00147	00157						
I.3P 00220	00225	00229					
IN0 00249	00262						

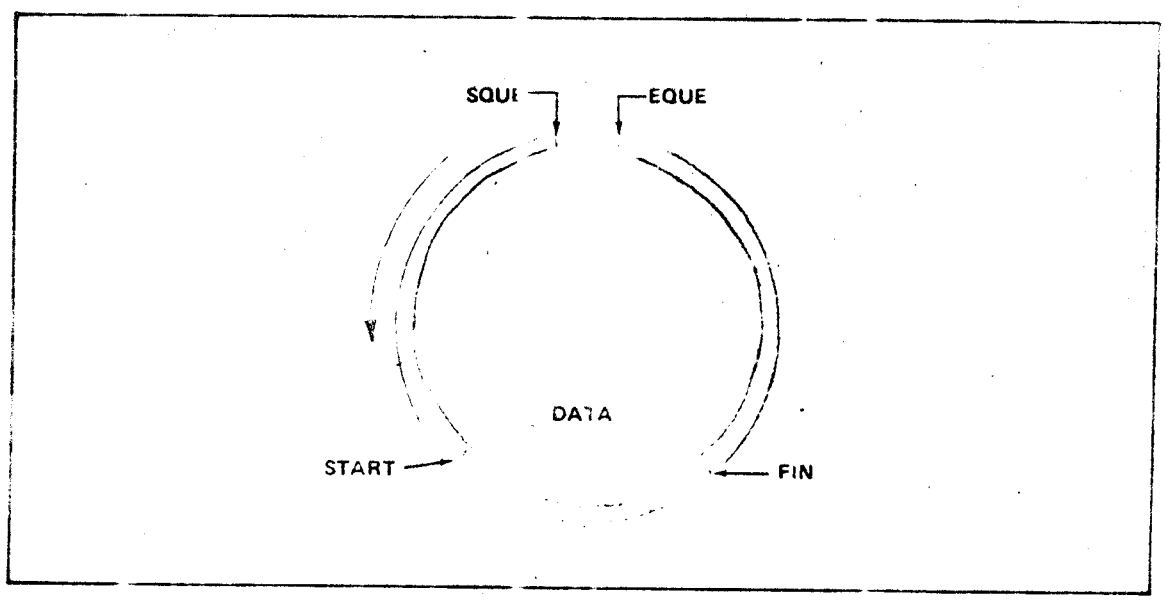
IN1	00338	00337					
IN2	00144	00145	00245	00299	00323	00353	
LC	00269	00269	00272				
L1	00336	00333					
L2	00383	00371	00373				
LF	00511	00266	00275				
NL	00506	00332	00372	00376			
NULL	00491	00291	00356	00363	00494		
OFF	00486	00153	00274	00280	00339	00489	
OUT	00476 00355	00155 00399	00242 00473	00244 00484	00290 00488	00318 00493	00320
OUT2	00284	00283					
OUT1	00368	00376					
PUT	00414	00257	00334	00417	00428	00429	
R3	00244	00491					
RECO	00241 00595	00577 00598	00580 00621	00583 00624	00586	00589	00592
REC1	00317	00571	00574				
REC2	00398	00559	00562	00565	00568		
RESET	00459	00262	00467				
RUB	00256	00255					
RUBO	00262	00252	00254				
S3	00290	00412					
SEND3	00289 00556	00578 00599	00581 00622	00584 00605	00587	00590	00593
SEND1	00354	00572	00575				
SEND2	00429	00560	00563	00566	00569		
SETUP	00159	00197	00174	00210			
SFS	00219 00324 00353	00220 00327 00384	00259 00328	00265 00336	00268 00342	00277 00364	00299 00375

STATE	00526	00146	00171	00211	00221		
TAB	00503	00370					
TBLE	00557	00162					
TEMP1	00516	00191	00198	00201	00226		
TEMP2	00517 00136	00192 00202	00114 00227	00127	00128	00129	00135
TEMP3	00518	00206	00222				
*TTY0	00558						
*TTY1	00561						
*TTY10	00588						
*TTY11	00591						
*TTY12	00594						
*TTY13	00597						
*TTY14	00620						
*TTY15	00623						
*TTY2	00564						
*TTY3	00567						
*TTY4	00570						
*TTY5	00573						
*TTY6	00576						
*TTY7	00579						
*TTY8	00582						
*TTY9	00585						
UNIT	00515	00106	00209	00481	00497		
WAIT1	00363	00366					
WAIT	00361	00385					

PROGRAMMING POINTS

Data transmission is characterized by high peak rates and low average throughput rates. If service routines are long, then the system will not be able to stand very high peak rates. If a computer is limited by the peak rate, then it will be mostly idle. The solution is to have very short service routines which queue data with the processing being done in the background.

```
I.53    NOP  INTERRUPT ENTRY POINT
        STA  SAVA
        LDA  FIN        IF FIN+1=EQUE THEN FIN:=SQUE
        INA             ELSE FIN := FIN + 1 ;
        CPA  EQUE
        LDA  SQUE
        CPA  START      IF FIN = START THEN OVERFLOW ;
        HLT  0
        STA  FIN
        LIA  MPX        (FIN) := INPUT
        STA  FIN,I
        LDA  SAVA
        CLF  MPX        ACKNOWLEDGE INTERRUPT
        JMP  I.53,I

SQUE    DEF  *+1        BEGINNING OF QUEUE
        BSS  16         LENGTH DEPENDS ON PROBABILITY OF ERROR
EQUE    DEF  *
START   DEF  SQUE+1     POINTER TO BEGINNING OF DATA
FIN     DEF  SQUE+1     POINTER TO END OF DATA
SAVA    NOP
MPX     EQU  13B        SELECT CODE OF DATA INTERFACE
```



CIRCULAR QUEUE

This routine should be about 33 microseconds in a 2100 yielding a peak rate of 33,000 characters per second. Another routine will compare the pointers to the beginning and to the end of the queue to see if there is anything in the queue. Since the multiplexer is asynchronous, output does not have to be done immediately after the interrupt and may be done from the background program.

In order to avoid queue overflow when a burst of data occurs, the queue must be long enough so that the probability of queue overflow is less than the probability of error on the telephone line. A very good analysis of queue length is "A Study of Asynchronous Time Division Multiplexer for a Timesharing Computer" by W.W. Chu; Fall Joint Computer Conference 1969 pp 669. For example, if 1) the computer is capable of handling 1500 characters per second in the background processing routine, 2) the average character throughput is 1000 characters per second, and 3) the average message length is 40 characters, then the I/O queue should be about 1000 characters long to achieve a probability of queue overflow of 10^{-6}.

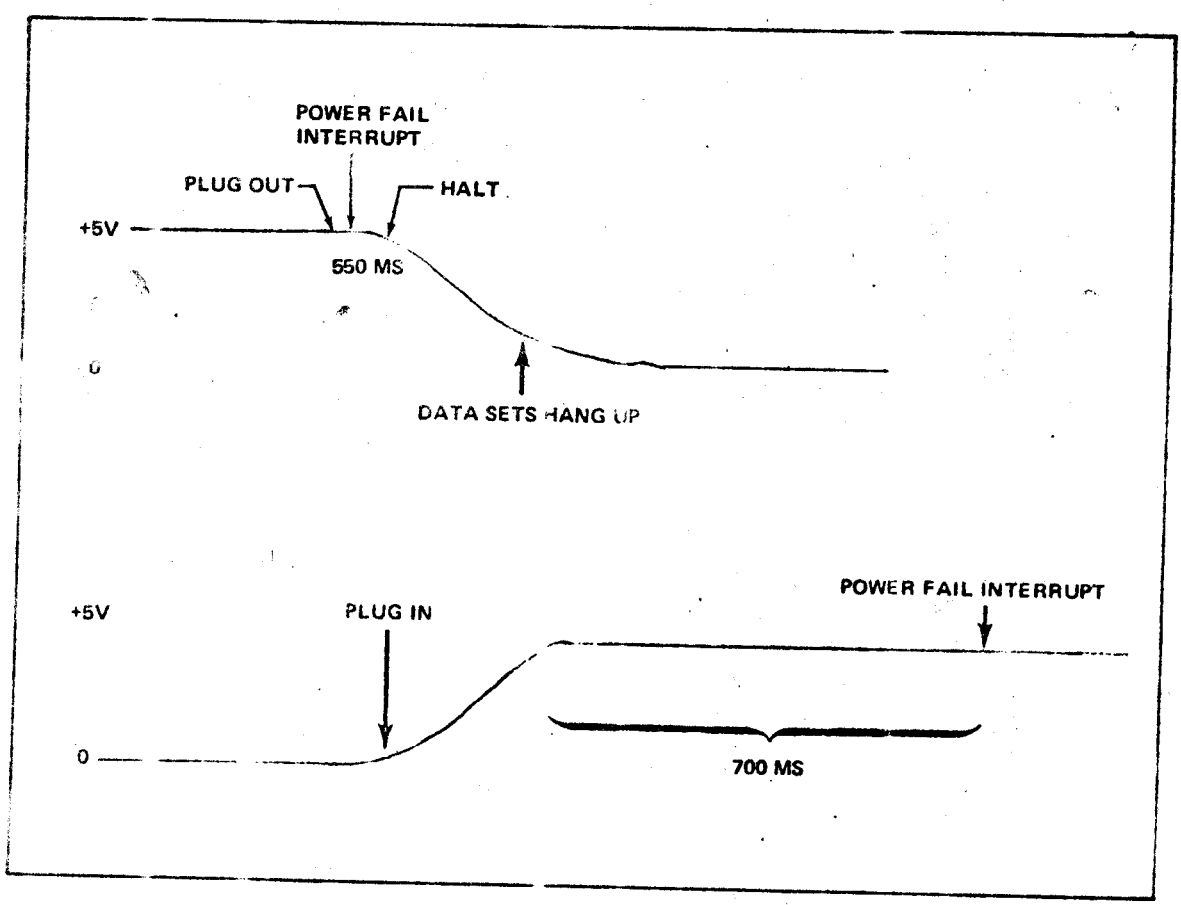
Just as the average throughput rate on a given line may be much less than the peak transfer rate, the average buffer needed per line may be much less than the peak. This implies a buffer allocation scheme.

POWER FAILURE

Upon the loss of power to the computer, a power failure interrupt is generated which gives the program a short time to record its current state. The amount of time should be about 550 microseconds before everything stops dead. When the voltages have come back up following the return of power, a 700 millisecond delay is triggered during which the logic is resetting to its initial state and after which a power up interrupt occurs.

Upon loss of power, the multiplexer returns all of its output lines to the mark state. This is off for the control lines causing any data set on the system to hang up. Internally, all control words, requests for interrupts, etc., are lost. In order to resume communication upon power up, all of the control words for all active channels must be reinstated. The last character transmitted before power failure may have gone out garbled, and characters may have been echoed or partially echoed but not input to the computer.

Power On sets the flag on the interface which should be reset to avoid a false interrupt upon the first STC instruction.



MULTIPLEXER LIMITATIONS AND EXTENSIONS

In the event that requests for interrupts are being generated faster than they are being serviced, data will be lost from the receive channels or data output will slow down. Since operation is asynchronous, it is not critical that output slows down. The output channels may proceed at full output speed and others may slow down. This is a result of the first come first serve method of generating interrupts. When a clear flag has been issued, the next channel ready to interrupt which rotates into position in the circulating memory gets to interrupt while the others are held off. This can lead to a beat note effect between the 70 microsecond period circulating memory and the computer program.

An example of a situation where this was observed was a driver which took 500 microseconds per character. It was called to output 240 characters per second to sixteen channel simultaneously. This would require an interrupt every 260 microseconds; thus, something had to make up for the 500 to 260 microsecond discrepancy. The result was a non-uniform slowing down of output. No characters were lost but the effect was unsightly. This problem does not occur if the interrupt routine can handle the output rate. Reducing the 500 microsecond interrupt routine to 200 microseconds eliminated this problem.

The 2100 computer, like most minicomputers, can handle a few thousand characters per second before the CPU is saturated. For high performance systems, it is adviseable to implement much of the multiplexer driver and as many commonly used routines as possible in microcode. This usually cuts execution time from 80% to 90%. In order to facilitate firmware operation, the flag of the interface will not set if the Seeking bit is on. Thus, it doesn't need to be checked in firmware and eliminates the possibility of being stranded in the microcode for 70 microseconds waiting on Seeking. However, to take advantage of this, Seeking must be checked, with interrupts disabled, before and after output from an initiator routine (any output not immediately following an interrupt).

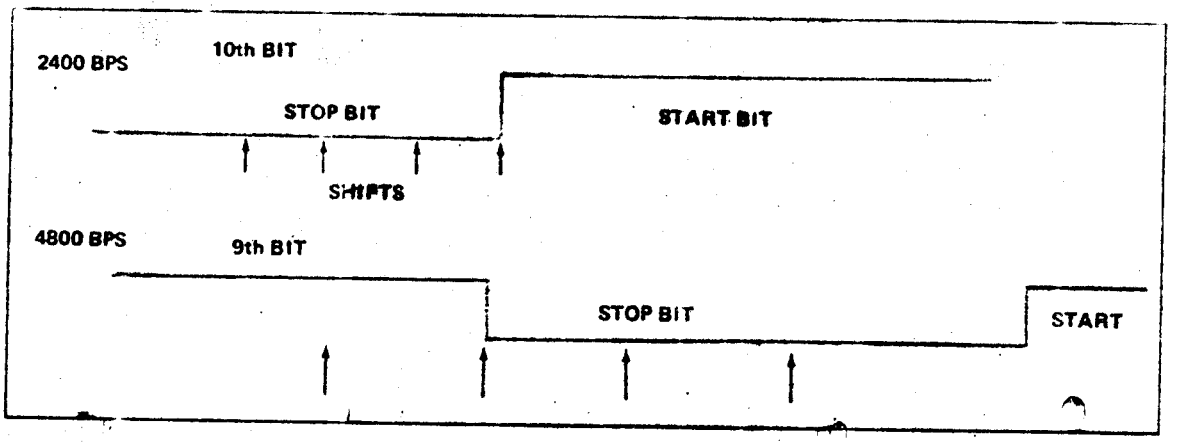
Initiator Section

```
CLF  0          disable interrupts
  .
LIA  MPX+1      check seeking
SSA
JMP  *-2
  .
OTA  MPX        output data
  .
OTA  MPX+1      output unit
  .
STC  MPX        initiate output
  .
LIA  MPX+1      check seeking
SSA
JMP  *-2
  .
STF  0          re-enable interrupts
```

Firmware

```
interrupt       (normal)
  .
  .
OCT  105000     macro call to the
                microcode.

OTA  MPX        outputs done from
OTA  MPX+1      the firmware with
STC  MPX        no seeking check
```

Each channel of the multiplexer is limited to 2400 bits per second by the internal operation of right justifying each character. Each justification shift takes 69.4 microseconds. The number of shifts needed is the difference between the character size and twelve, i.e., a ten bit format needs two shifts. One additional 69.4 microsecond period is needed before the beginning of the next start bit. The two justification shifts and the additional period must occur in one half bit time to be ready for the next start bit. At 2400 bits per second, the three 69.4 microsecond periods take exactly one half bit time, thus the limit of 2400 bits per second. Higher speed operation, e.g., 4800 bps, may be achieved by shortening the programmed character size. The multiplexer then starts justification in the middle of the last data bit instead of in the middle of the stop bit. At 4800 bps, the equivalent of three samples per bit are taken. This sampling rate is regarded by some to be unreliable since the multiplexer would make errors if there were much distortion on the incoming data.



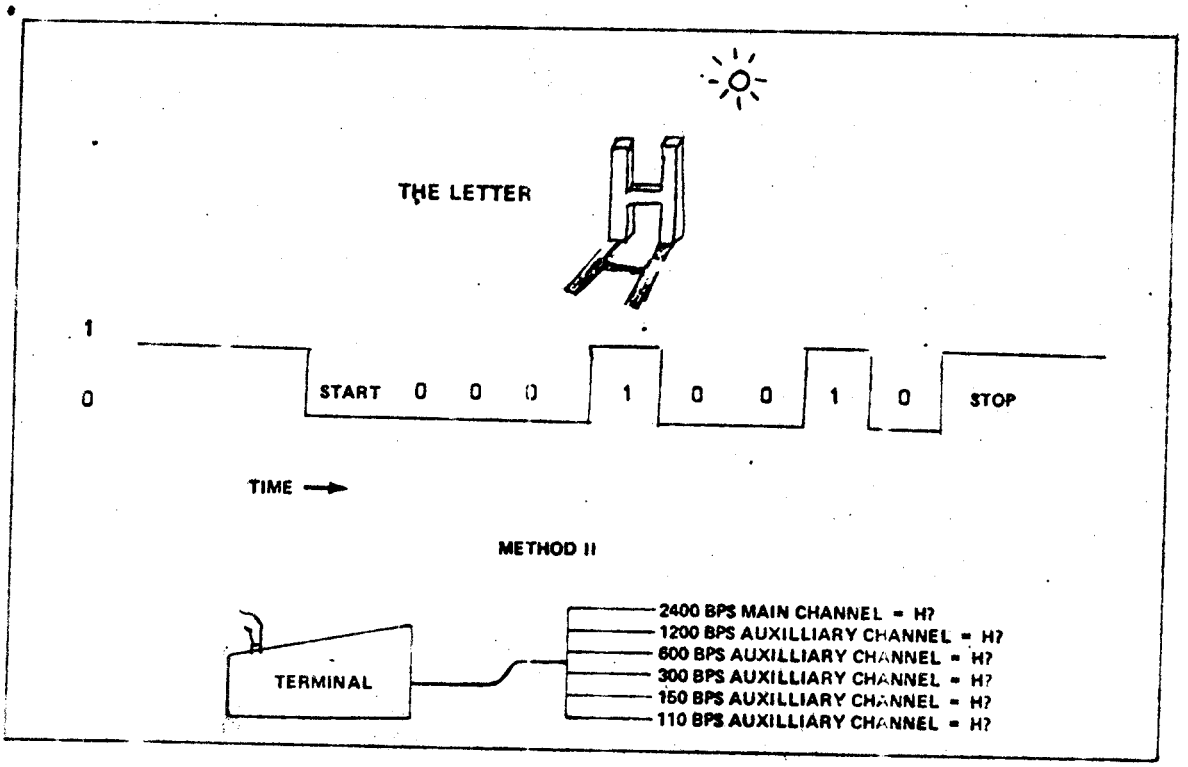
DYNAMIC DEVICE SPEED DETECTION

In order to recognize the bit rate of data coming from an asynchronous bit serial device, the program must make decisions upon the received patterns. The hardware features which can be useful to the program are dynamic speed changing on any channel and reception of data at more than one speed simultaneously by use of the auxilliary channels.

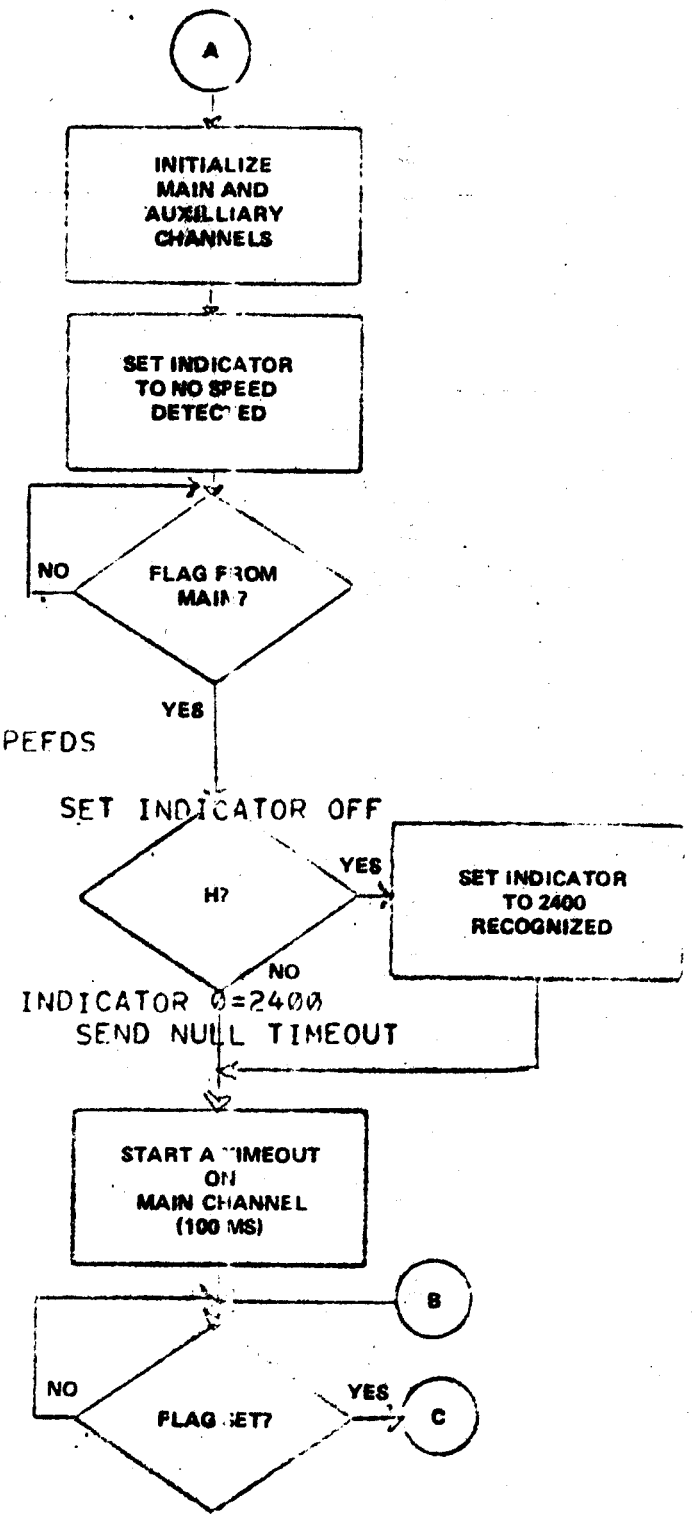
There are basically two approaches to speed detection; 1) receive the data at a frequency differing from the transmitted frequency and make a decision upon the skewed result, or 2) receive the data at the expected frequency and see if it is as expected.

The first method, making decisions on skewed data is difficult to achieve reliably and requires that an in depth analysis be made of the timing of the bits. An example might be reception of data at 150 baud and comparing for the patterns that a character such as "H" might make were it transmitted at 110 baud or 300 baud.

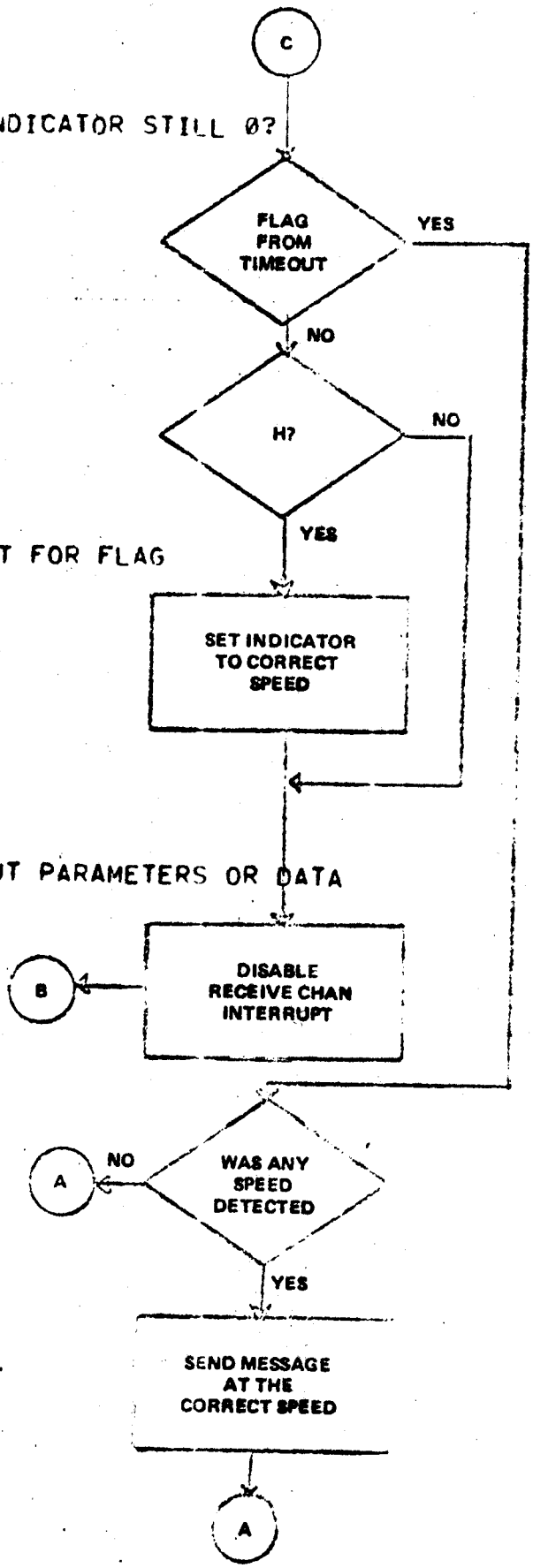
The second method would use the five extra receive-only channels to bring in the data at six separate speeds such as 110, 150, 300, 600, 1200, and 2400 baud. A comparison is made with the expected pattern, "H", to determine which of the speeds is the same as the terminal's.



```
0001                        ASMB,A,B,L
0003  00104                       ORG 1005
0004  00100 000020  INIT    NOP       CONFIGURE MAIN CHANNEL AND 5 AUXILLIAR
0005  00101 060240            LDA C24R
0006  00102 064255            LDB .0
0007  00103 014210            JSB OUT
0008  00104 060254            LDA C11S
0009  00105 064255            LDB .0
0010  00106 014210            JSB OUT
0011  00107 060241            LDA C12R
0012  00110 064256            LDB .1
0013  00111 014210            JSB OUT
0014  00112 060242            LDA C60R
0015  00113 064257            LDB .2
0016  00114 014210            JSB OUT
0017  00115 060243            LDA C30R
0018  00116 064260            LDB .3
0019  00117 014210            JSB OUT
0020  00120 060244            LDA C15R
0021  00121 064261            LDB .4
0022  00122 014210            JSB OUT
0023  00123 060245            LDA C11R
0024  00124 064262            LDB .5
0025  00125 014210            JSB OUT

0027*  DETECT AN H AT DIFFERENT DASPEEDS
0028  00126 003400  DTECT   CCA
0029  00127 070271            STA CW       SET INDICATOR OFF
0030  00130 014201            JSB SFS
0031  00131 050266            CPA H
0032  00132 002001            RSS
0033  00133 024136            JMP LOW
0034  00134 002400            CLA
0035  00135 070271            STA CW       INDICATOR 0=2400
0036  00136 060263  LOW       LDA NULL     SEND NULL TIMEOUT
0037  00137 064255            LDB .0
0038  00140 014210            JSB OUT
0039  00141 060270            LDA CLRI
0040  00142 064255            LDB .0
0041  00143 014210            JSB OUT
0042  00144 014201  LOOP      JSB SFS
0043  00145 050266            CPA H
0044  00146 024155            JMP FOUND
0045  00147 106514            LIB SC+1
0046  00150 004010            SLB
0047  00151 024164            JMP DONE
0048  00152 060267            LDA CLR
0049  00153 014210            JSB OUT
0050  00154 024144            JMP LOOP
0051  00155 102513  FOUND     LIA SC
0052  00156 001727            ALF,ALF
0053  00157 001323            RAR,RAR      POSITION CHANNEL NUMBER
0054  00160 010272            AND B7
0055  00161 002004            INA
0056  00162 070271            STA CW       INDICATOR = FROM WHICH CHANNEL
```



```
0057  00163 024144        JMP LOOP

0059  0016[illegible]  DONE  LDA CW      INDICATOR STILL 0?
0060  00165 022323        SSA
0061  00166 024100        JMP INIT
0062  00167 042246        ADA DCW
0063  00170 [illegible]    LDA A,I
0064  00171 064255        LDB .0
0065  00172 014210        JSB OUT
0066  00173 060267        LDA CLR
0067  00174 064255        LDB .0
0068  00175 014210        JSB OUT
0069  00176 064273        LDB BUFF
0070  00177 000040        CLE
0071  00200 024220        JMP SEND

0073  00201 000000  SFS   NOP         WAIT FOR FLAG
0074  00202 103113        CLF SC
0075  00203 102313        SFS SC
0076  00204 024203        JMP *-1
0077  00205 102513        LIA SC
0078  00206 010265        AND B377
0079  00207 124201        JMP SFS,I

0082  00210 000001  OUT   NOP         OUTPUT PARAMETERS OR DATA
0083  00211 102613        OTA SC
0084  00212 106614        OTB SC+1
0085  00213 102713        STC SC
0086  00214 102514        LIA SC+1
0087  00215 002020        SSA
0088  00216 024214        JMP *-2
0089  00217 124210        JMP OUT,I

0091  OUTPUT MESSAGE AT DETECTED SPEED
0092  00220 054300  SEND  CPB EBUFF
0093  00221 024100        JMP INIT
0094  00222 160001        LDA B,I
0095  00223 002041        SEZ,RSS
0096  00224 001727        ALF,ALF
0097  00225 010265        AND B377
0098  00226 030264        IOR STOP
0099  00227 002200        CME
0100  00230 002041        SEZ,RSS
0101  00231 [illegible]    INB
0102  00232 074271        STB CW
0103  00233 [illegible]    CLB
0104  00234 014210        JSB OUT
0105  00235 014241        JSB SFS
0106  00236 064271        LDB CW
0107  00237 024220        JMP SEND
```



```
0109  00240 124405  C24P    OCT 124405
0110  0024  12[illegible]13  C12P    OCT 120413
0111  00242 120427  C60P    OCT 120427
0112  00243 120457  C30P    OCT 120457
0113  00244 120537  C15P    OCT 120537
0114  00245 120602  C11P    OCT 120602
0115  00246 000247  DCW     DEF *+1
0116  00247 160405  C24S    OCT 160405
0117  00250 160413  C12S    OCT 160413
0118  00251 160427  C60S    OCT 160427
0119  00252 160457  C30S    OCT 160457
0120  00253 160537  C15S    OCT 160537
0121  00254 160602  C11S    OCT 160602
0122  00255 000000  .0      OCT 0
0123  00256 040000  .1      OCT 40000
0124  00257 042000  .2      OCT 42000
0125  00260 044000  .3      OCT 44000
0126  00261 046000  .4      OCT 46000
0127  00262 050000  .5      OCT 50000
0128  00263 047777  NULL    OCT 47777
0129  00264 043600  STOP    OCT 43600
0130  00265 000377  B377    OCT 377
0131  00266 000110  H       OCT 110
0132  00000         A       EQU 0
0133  00001         B       EQU 1
0134  000[illegible]3  SC    EQU 13B
0135  002[illegible]7  10[illegible]000  CLR     OCT 100000
0136  002[illegible]0  10[illegible]000  CLRI    OCT 104000
0137  002[illegible]1  00[illegible]000  CW      NOP
0138  002[illegible]2  000007  B7      OCT 7
0139  002[illegible]3  000274  BUFF    DEF *+1
0140  002[illegible]4  04[illegible]105          ASC 3,HELLO
      002[illegible]5  04[illegible]114
      002[illegible]6  047440
0141  00277 00[illegible]412          OCT 5412
0142  00300 00[illegible]300  ERUFF   DEF *
0143                                  END
* NO ERRORS*
```

DEVICES

HP 2600

There are hundreds of devices which can be used by the 12920 Asynchronous MULTIPLEXER and they are all different. The specifications of each new device to be interfaced to the 2100 must be examined carefully for hidden quirks. Most manufacturers do not give, in marketing handouts, sufficient information to program the terminal leaving the task to trial and error, circuit analysis, or direct contact with technicians from the company. In order to gain a feel for the actions required, the HP 2600 CRT is described. It can be run in three modes, over a 103 data set, through a 202 data set, or hardwired. There is a switch on the back which can select any one of 10 speeds, 110, 220, 440, 880, 1760, 150, 300, 600, 1200, or 2400 baud. In the 110 position, an extra stop bit is added to the characters so that the terminal will be teletype compatible.

HP 2600 VIA 103 DATA SET

The terminal disregards status when being used with a 103. This makes it relatively easy to interface because it does not really care what the computer is doing. It blindly receives and transmits full duplex whether a data set is connected or not. Its data terminal ready lead is always raised when the power is on so that a call may be answered or originated at any time.

The operator usually calls up the computer from the terminal by dialing the computer's number. The computer's data set must have been primed by the computer raising its data terminal ready lead, CD, previous to the call. This is done by the computer output of a control word to the 12922 control board. The computer's data set must be in the "Auto" mode. The telephone will ring once or twice, go "off hook" and transfer itself into the data mode. The data button lights and the CC lead, data set ready, to the computer comes up. If the control board were primed to interrupt on CC, this causes an interrupt into the computer.

The computer's data set puts out a loud whistle on the telephone line which disables echo suppressors and signals to the calling operator that he may push his "data" button. After pushing the button, a carrier is sent to the computer and the computer's carrier detect, CF, lead comes up. If the caller never pushes his data button, then carrier never comes up. Some data sets may be ordered with an option which disconnects them if carrier does not appear 10 to 20 seconds after the call, otherwise a non-data caller could occupy the line indefinitely. Most systems do not have to worry about this "HOUSEWIFE" call, but on those that do, a timeout should be initiated after data set ready comes up, and if no carrier appears, CD, data terminal ready, should be dropped to disconnect the unwanted caller.

After both data sets are sending out carriers, transmission and reception may proceed until the computer, the telephone line, or the terminal operator breaks the connection. Without a connection, no carrier will come through, carrier detect will drop. Data sets equipped with the carrier detect disconnect option will go back "on hook".

The simplest driver, which is adequate in the majority of instances, raises CD, data terminal ready, leaves it up, and disregards all status. In order to have a fully automatic, operatorless system, interrupts should occur on status changes, and the program should be able to disconnect the caller.

HP 2600 HARDWIRED

The hardwired terminal is identical to 103 operation except that there is no problem with broken connections. The HP 2600 can be plugged into the 30062 Connector Panel with the same cable that comes with the unit (see page 2). When the 2600 is plugged in, the carrier detect and data set ready lines to the

computer will come up. These are the request to send and the data terminal ready lines from the terminal. Since status from the HP 2600 looks the same as status from a 103 modem, the same software for a 103 can be used for the hardwired terminal.

HP 2600 & 202 DATA SET

202 operation is different since the terminal must turn the modem around at the end of each transmission. On the 2600, the terminal switches from send to receive if it detects a 1 to 0 transition on the SB, secondary channel data, from the modem. Thus, if the computer wants to get the terminal out of the send mode, it drops its SA, secondary data send, line to its 202 and from 10 to 100 milliseconds later, this signal arriving at the 2600 will cause a line turnaround. The terminal will remain in the receive mode until any key is pressed or until the computer turns its modem around.

The computer will see reverse channel on when it is transmitting to the terminal and will see it drop when the terminal goes into the send mode. Break on the 2600 interrupts its SA line and will be seen at the computer as a loss of the secondary received data signal.

USE OF THE IBM 2741 HARDWIRED

When the 2741 is bought or leased with the 103A dial up option, the carrier and clear to send lines are switched. These lines must be unswitched if the terminal is to operate directly into the connector panel without an intervening data set.

A method for doing this is to interchange wires 5 and 8 on either end of the 30332-002 extender cable if this is used between the terminal and the panel. If no extender cable is used, then the wires must be switched in the 2741 connector or an intermediary cable must be made. There is no cabling problem if the terminal is run through modems.

INCLUSION IN HP SOFTWARE SYSTEMS

BCS SYSTEM

Interfacing with the Basic Control System is straight forward being described in detail in "A Pocket Guide to Interfacing HP Computers". The main complication is that Prepare Control System only configures the equipment table for up to eight units on a controller. This is a minor inconvenience remedied by patching the equipment table or having two entry points to the driver each with eight units assigned to it.

RTE and DOS

DOS and RTE use essentially the same drivers and therefore have the same problems.

In RTE all interrupts go directly to the Central Interrupt Control which immediately clears the flag on the interface. The reason is that interrupts may be of two kinds, scheduling and data handling. The scheduling interrupt can call in a program off of the disk. During the tenth of a second that the disk transfer is preceding, the lower priority devices must be re-enabled for interrupts, this being done by the Clear Flag. As long as the flag and control are both set on an interface, lower priority devices cannot interrupt.

On the multiplexer, a clear flag lets another channel set the flag and replace the status bits. Once the original status bits are overwritten, data is lost.

Another problem which may be cured at the same time is the inability of DOS and RTE to handle multi-unit controllers with I/O operations proceeding simultaneous on several of the units.

Both of these problems are solved by placing a pre-interrupt routine before SCIC. The initiator section puts a JSB to the pre-interrupt routine in the trap cell. The pre-interrupt routine:

1) Disables the interrupt system.
2) Does an LIA of the upper and lower select codes, saving these in temporary locations.
3) Takes the unit number from the status, retrieves the equipment table address which it has remembered from the initiator section, and stashes this in the interrupt table.
4) Transfers control to SCIC+1 after placing the return address in SCIC. The Central Interrupt Control has hopefully been tricked into operation on the correct equipment table entry for the channel which interrupted and the status has been saved so that clear flag won't destroy anything.

INDEX

INDEX (Continued)