Atari Jaguar Voice Modem

Please note that the* Atari Jaguar Voice Modem *section of the documentation is still undergoing significant revisions to properly outline the various requirements for tasks such as making a call or answering an incoming call. If this section of your documentation is more than two months old, please contact Jaguar Developer Support for an updated revision.

Introduction

The Jaguar voice modem is a high performance (v32terbo) DSP based modem, with many additional features and modes which make it particularly suitable for an interactive and consumer friendly game environment.

In the rest of this section, we discuss:

The Modem Interface
Data Communications and Bandwidth
Flow Control
Data Parsing
Call Waiting

...conclude with a summary of the commands and unsolicited responses used in voice plus data ode. A full reference manual of all commands is available but not complete yet. This manual is only ecessary for full featured fax and data communication systems (without simultaneous voice).

Modem Interface

it, no parity,

ommand to
usually

this case the
kets are

ast

The interface between the Jaguar and modem is via the built in Jaguar UART. Communications in both directions, are in the form of 2 or 3 byte packets, at a baud rate of 57600 or 19200 (1 start b 2 stop bits).

After reset, all communications are initiated by the Jaguar. Typically, Jaguar will send a co the modem, and the modem will respond. In simultaneous voice plus data applications, we reduce the baud rate between the modem and Jaguar, in order to ease the interrupt response requirements.

The Jaguar can also enable various types of "unsolicited" data packets from the modem. In modem may send a data/command packet to the Jaguar unsolicited. These unsolicited pack typically used for incoming data, call waiting detection, loss of the line, and other errors.

Commands from the Jaguar to the Modem are always sent as a two byte packet, with the le significant byte sent first.

© 1995 Atari Corp. Confidential Information Property of Atari Corporation

Replies from the Modem to the Jaguar are sent as two byte packets, with the most significant byte (usually the command byte) first. The modem will also send a padding byte of 0xFF prior to a packet if there was a significant gap since the previous packet.

The Parse data flow diagram shows how to handle received data.

Data Communications And Bandwidth

In voice plus data mode (known hereafter as SVD - simultaneous voice plus data), compressed voice data is sent over the telephone line in packets which have a one byte header. Game data packets can be inserted into this data stream at any time with a one byte overhead. The game data packets actually interrupt the voice data stream to keep transport latency to an absolute minimum (which is necessary for good interactivity).

Developers need to understand the data bandwidth which is available, and then decide which packet sizes are most appropriate for their game. The following equations describe the available bandwidth:

```
Total data bandwidth = Line Speed / 8 (in bytes per second)
```

[Modem data is sent with an embedded clock, with no need for start or stop bits]

```
Voice data bandwidth = (Voice sampling frequency/4) +
  (Voice sampling frequency / (4*Voice packet size))
```

This gives you the voice data bandwidth, in bytes per second. This shows that each voice sample uses 2 data bits - or 4 samples per byte, and each voice packet has a one byte overhead.

```
Game data bandwidth = (number of game data packets per second) *
                      (game data packet size + x)

(x = 1 in normal mode,  2 for error detection mode)
```

The following table shows the voice sampling rates that the modem will use by default (assuming 80 byte voice packets, and the default adaptive voice sampling rates):

Line Speed	Total Bytes Per Second	Voice Sample Rate	Voice Data Rate	Voice Packet Size	Voice Headers	Voice Bytes Per Second	Remaining Bytes Per Second
19200	2400	7200	1800	80	22.5	1822.5	577.5
16800	2100	6800	1700	80	21.25	1721.25	378.75
14400	1800	5600	1400	80	17.5	1417.5	382.5
12000	1500	4400	1100	80	13.75	1113.75	386.25
9600	1200	3200	800	80	10	810	390

The Remaining bytes/second are available for data packets. Game data packets have a one byte overhead each, plus an additional overhead byte for error detection. Note that you MUST use a form of error detection, since errors do occur over the line. Error correction is usually achieved by requesting

 © 1995 Atari Corp.

that the packet be resent. So, assuming a worst case data rate of 378 bytes per second, the following data packet options are possible:

Total Data Rate (Bytes/Sec)	Data Packet Size	Packet Overhead	Total Packet Size	Packets Per Second	Total Data (Bytes/Sec)
378	1	2	3	126.00	126.00
378	2	2	4	94.50	189.00
378	4	2	6	63.00	252.00
378	6	2	8	47.25	283.00
378	10	2	12	31.50	315.00
378	20	2	22	17.18	343.64
378	40	2	42	9.00	360.00

As you can see, the smaller the data packet size, the less efficient this method is (in terms of the total bytes per second). However, the smaller packets do provide a higher packet-per-second rate, which will increase user interactivity.

Flow Control

Example code is provided for initialization and overall flow control, and we suggest everyone use it. Once the two modems have completed "handshaking", the users will be able to talk over their headsets, whilst the Jaguars send each other data packets.

The Jaguar game will need a "Modem" option selection screen. This will allow selection of any of the following items:

1. Call. This brings up an edit field to enter the number to dial. When entered and OK selected, the modem will go off hook and dial the number. The user will hear the dialing via her headset. If the line is answered, she will be able to talk to the answerer via the headset. If there is no answer, she can select "Hang up".

2. Hang up. This will do a graceful cleardown (i.e. cause both ends to hang up together) if the modem was communicating digitally with the other end. If the modem was still in analog mod it will simply hang up the line.

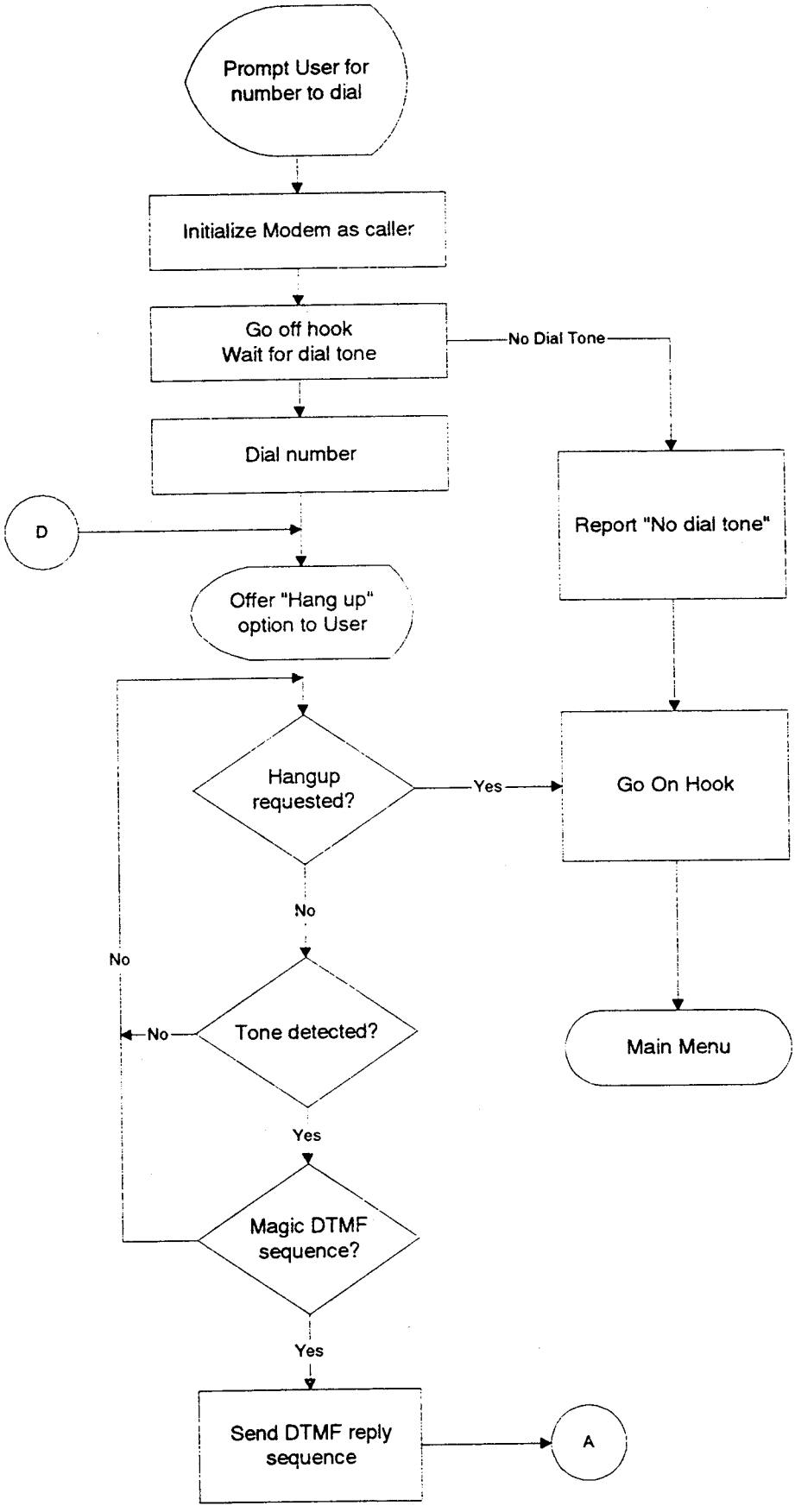
3. Answer. This is the selection used by the answerer after the two parties have verbally agreed over the analog line to play the game. This selection will mute the headsets and commence handshaking.

4. Adjust voice volume

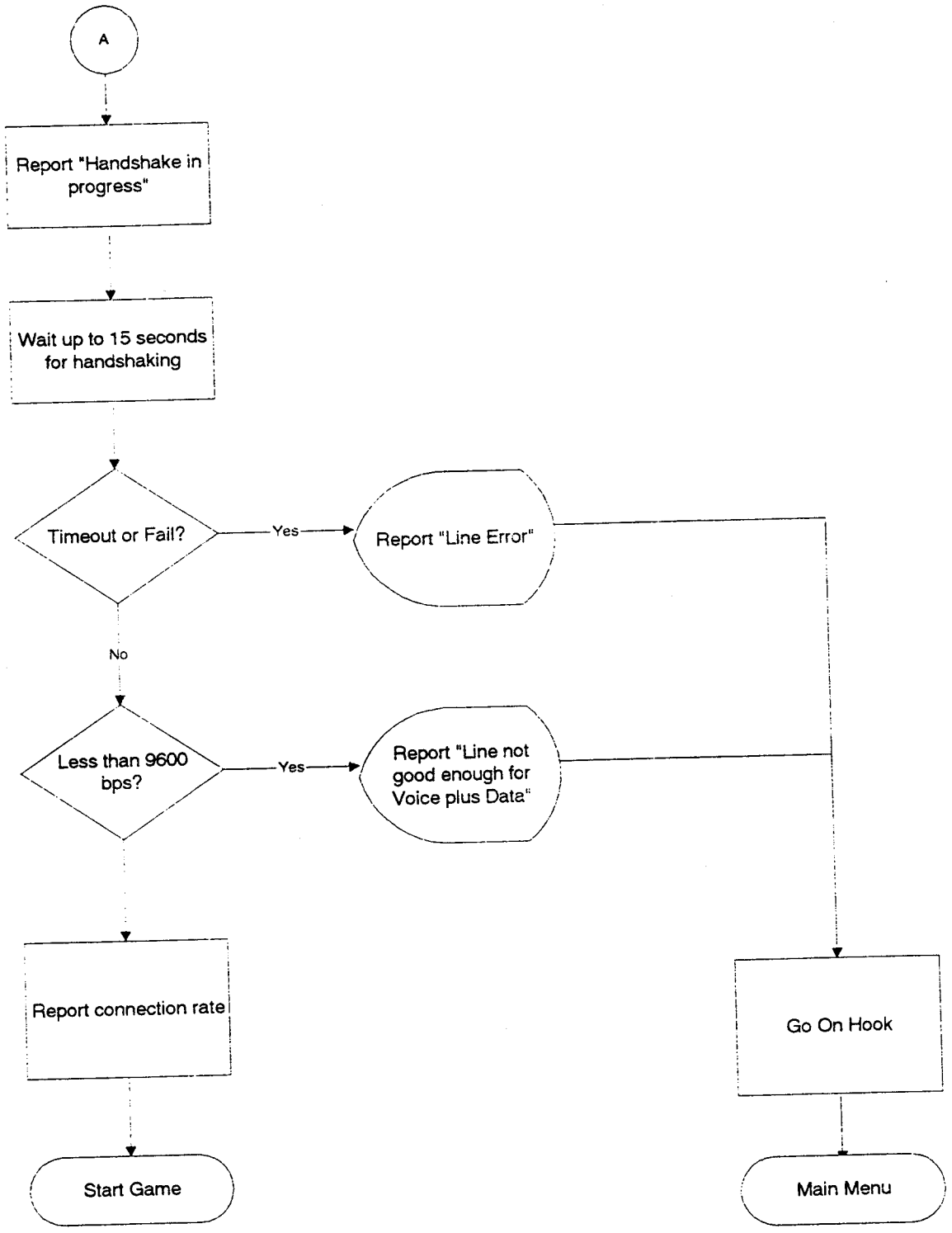
An outline of the Modem commands used for each of the four options listed above is given below. Example code is also available, and a flow chart is included. Details of each command are given at th

© 1995 Atari Corp. *Confidential Information - Property of Atari Corporation*

Call



 © 1995 Atari Corp.



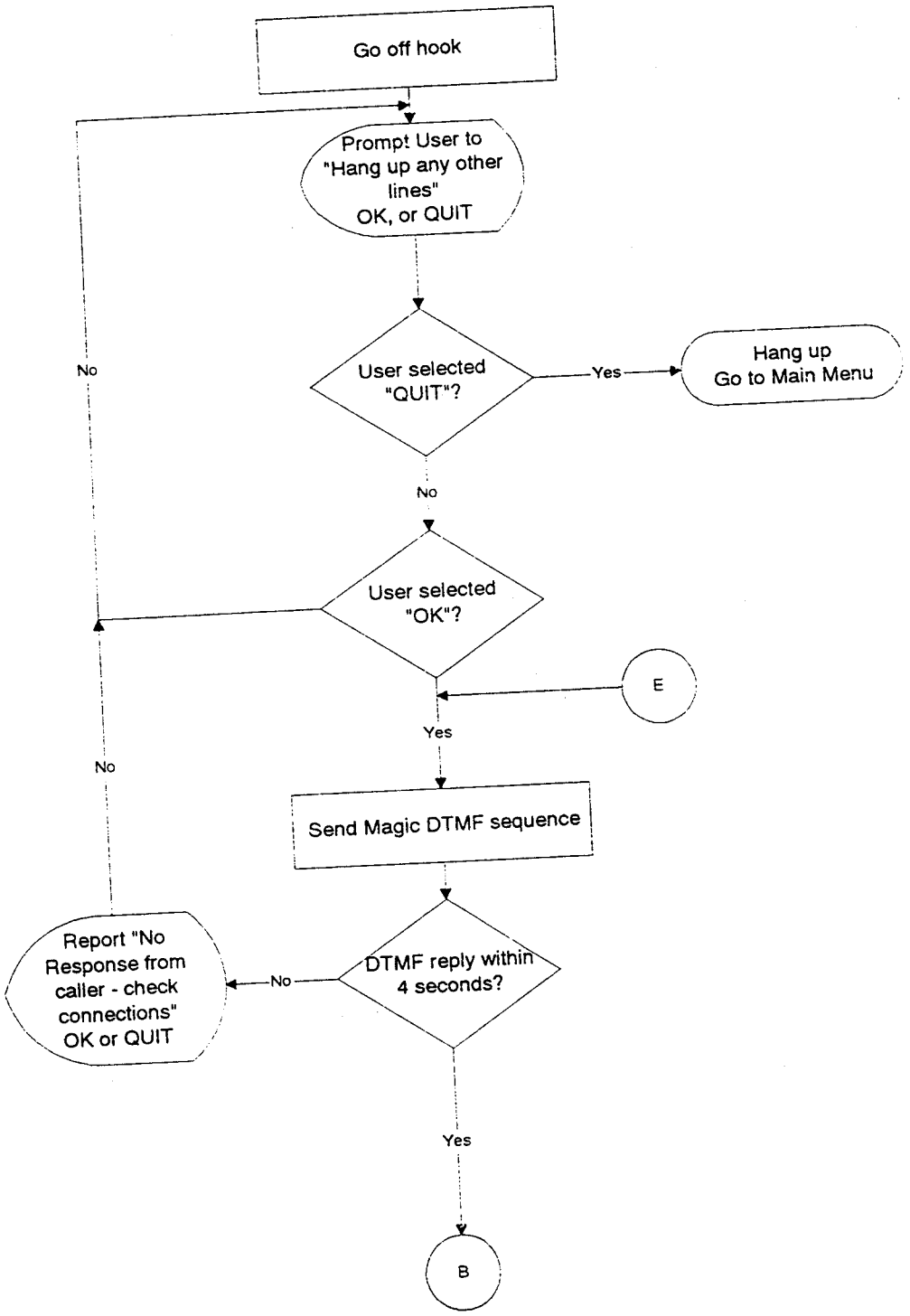
Command	Response	Description
FFFF	B80x	Reset modem and do a self test.
FFFE	none	Set baud rate to 19200
000F	000F	Enable echo back of commands
B000	B000	Enable Analog Line to Headset connection
2C80	2C80	Set this modem up as a Caller, and enable call waiting detection
3952	3952	Set miscellaneous configuration items
A021	A021	Set target error rate to better than 1 in 10e6 bits (i.e. minimum)

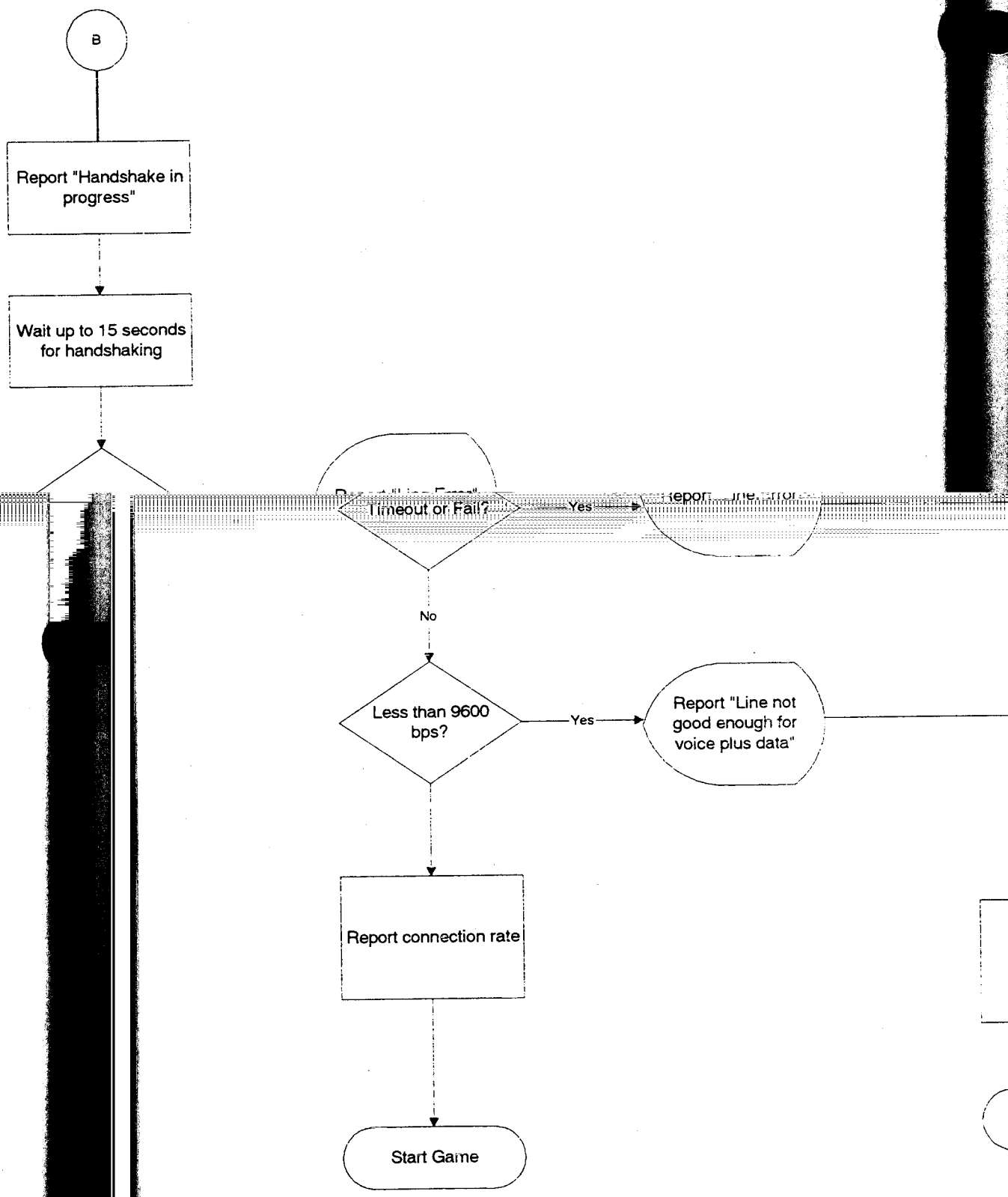
Command	Response	Description
F207	F207	Enable unsolicited error detection codes
B602	B602	Enable error detection mode
B5xx	B5xx	Set data packet size
B405	B405	Set voice packet size to 80 bytes
A37E	A37E	Enable loss of line detection
A060	A060	Go off hook
8C01	8Cxx	Wait for dial tone (4 second timeout if necessary)
8A2x	8A2x	Dial number (Repeat command for each digit in phone number)
6800	000x	Poll DTMF tone detector for magic 10 tone sequence.
	FFFE	(no tone detected). No timeout here - users are talking. If no tone is ever detected, the Caller will never see the "Handshake in progress" status, but the users will still be able to talk and discuss the problem over the analog line. When magic tone is detected...
8A2x	8A2x	Send magic DTMF reply sequence
		Display "Handshake in progress" status message
2C80	2C80	Set me up as the caller
8000	8000	Start Handshake
8100	8xyz	Poll for handshake successful (timeout after 15 seconds)

Hang Up

Command	Response	
9000	9000	Graceful cleardown and hang up if in data mode
A040	A040	Just hang up (if in analog mode)

Answer Sequence





Command	Response	Description
FFFF	B30x	Reset modem and do a self test.
FFFE	none	Set baud rate to 19200
000F	000F	Enable echo back of commands
B000	B000	Enable Analog Line to Headset connection
2480	2480	Set this modem up as an answerer, and enable call

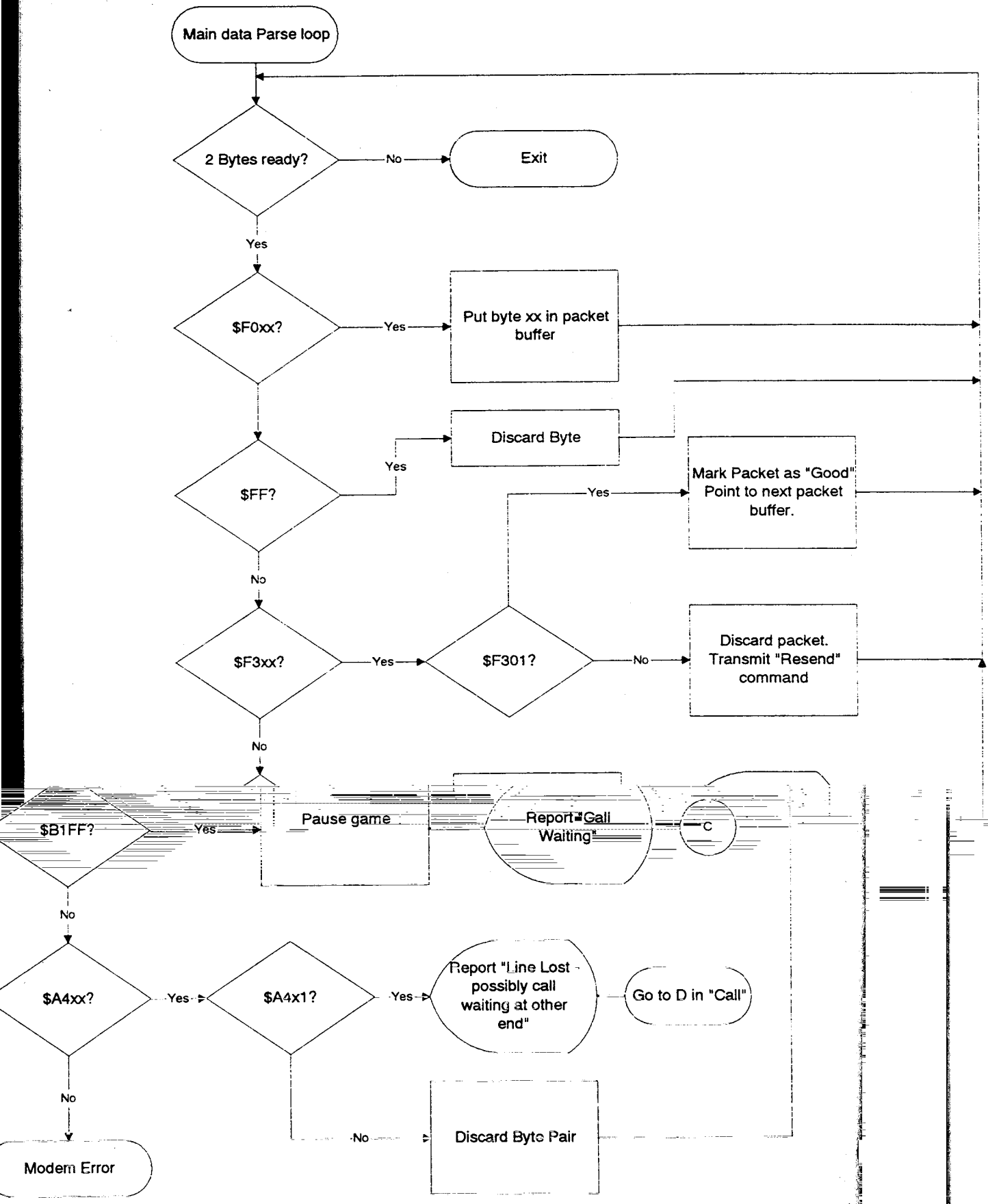
waiting detection

© 1995 Atari Corp.

Command	Response	Description
3952	3952	Set miscellaneous configuration items
A021	A021	Set target error rate to better than 1 in 10e6 bits (i.e. minimum)
F207	F207	Enable unsolicited error detection codes
B602	B602	Enable error detection mode
B5xx	B5xx	Set data packet size
B405	B405	Set voice packet size to 80 bytes
A37E	A37E	Enable loss of line detection
A060	A060	Go off hook Prompt user "Hang up any other hand sets" Wait for user to acknowledge other lines are hung up
3A2x	3A2x	Send magic DTMF sequence
6800	000x	Poll DTMF tone detector for magic DTMF reply sequence
	FFFE	(no tone detected). Timeout after 4 seconds If timeout, prompt user "No response from caller modem. Check modem connections" Wait for acknowledgment, then go to "Send magic DTMF sequence" When magic reply detected ... Display "Handshake in progress" status message.
2480	2480	Make me an answerer
8000	8000	Start Handshake
8100	8xyz	Poll for handshake successful (timeout after 15 seconds)

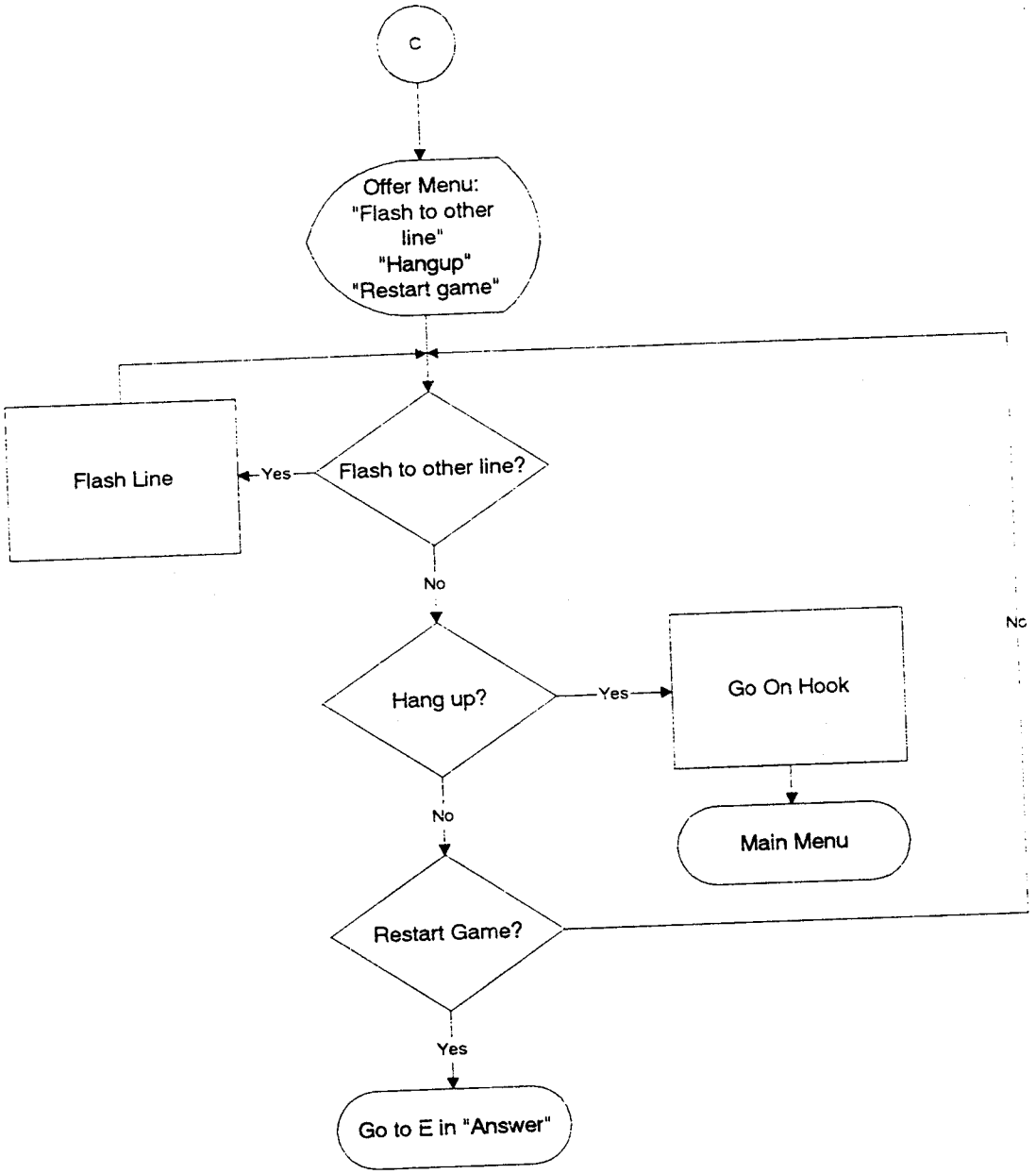
Parsing The Received Data

Once the modem has been initialized, and handshaking has occurred, data transmissions are possible. A flow chart for received data is given below:


Main data Parse loop
2 Bytes ready?
No
Exit
Yes
$F0xx?
Yes
Put byte xx in packet buffer
Discard Byte
Yes
$FF?
Yes
Mark Packet as "Good" Point to next packet buffer.
No
$F3xx?
Yes
$F301?
No
Discard packet. Transmit "Resend" command
No
$B1FF?
Yes
Pause game
Report "Call Waiting"
C
No
$A4xx?
Yes
$A4x1?
Yes
Report "Line Lost - possibly call waiting at other end"
Go to D in "Call"
No
No
Discard Byte Pair
Modem Error

Call Waiting

The line which gets a call waiting tone will receive the unsolicited data packets $B1FF then $A4??. The other line will just get a $A4?? packet. Both ends will then immediately go into analog line mode, which will allow them to talk, and for the call waiting receiver to ask the other party to wait while she picks up the call waiting. She then selects the "go to call waiting" box, which flashes the line for her, has the conversation, then selects "reconnect", which will flash the line again (back to the first party), and send the magic DTMF tone sequence - starting handshake again.



Command Reference For Voice Plus Data

Unless otherwise noted, all values are in hexadecimal.

Initiate-Report Software Reset 0xFFFF

function: This command causes the Voice Modem to reset all parameters to the default conditions. After resetting, the Voice Modem will return the self-test result executed during the previous POR\. This command may be issued at any time. **CAUTION: care should be taken because the command will clear all operating parameters to the default values.**

The Modem will internally issue the following commands during reset:

Command Name	Command Code
HAS_MODE_ON	0xA031
HOST_ECHO_ON	0x000F
AUDIO_LOW	0x0102
Set Configuration Word 1	0x2480
Set Configuration Word 2	0x3952
Enable Unsolicited Error Detection Responses	0xF207
Set Bit Error Rate Target	0xA021
V24_MASK	0xA3FF
Connect Headset to Analog Line	0xB000

Since it is not always possible to determine whether the modem host baud rate is set to 57600 or 19200, the following procedure is recommended for issuing the reset command:

- Send Reset command at 57600
- If a sucessful response (0xB800) is received within 1 second, then exit reset
- If a response is not received within 1 second, issue the reset command at 19200 and ignore the response (if any)
- Then issue a reset command again at 57600, and wait for the response.

response: The response is returned at a host baud rate of 57600, after the reset is completed and within about 1 second. It is in the form 0xB80*x* where *x* has the bit form:

[DSP] [AFE] [ROM] [SRAM]

where 0 is a pass and 1 is a fail. Thus a successful self-test will give a response of 0xB800.

default: N/A

 © 1995 Atari Corp.

Change Host Baud Rate To 19200 — 0xFFFE

function: Set host baud rate to 19200
Only reset {0xFFFF} can change the baud rate back to 57600.

response: none

default: N/A

Connect Headset To Analog Line — 0xB000

function: Allow the headset to be used as a telephone handset, as if it were directly connected to the analog line. (In reality, a digital connection is made between the line Codec and the headset Codec)

This command will also cause the modem to switch to SVD mode immediately after handshaking is complete.

response: The command will be echoed back within 1.2ms

Set Configuration Word 1 — 0x2*nnn*

...writes 12 bits, specified by *nnn*, to the modem Configuration Word 1.

Modem Type	Data Rate(bit s)	Modulation	Bit:	5	4	3	2	1	0
V.27ter	2400	DPSK		0	1	0	1	1	0
V.17	14400	QAM-TCM		1	1	0	0	0	0
V.17	12000	QAM-TCM		1	1	0	0	0	1
V.17	9600	QAM-TCM		1	1	0	0	1	0
V.17	7200	QAM-TCM		1	1	0	0	1	1
V.21 Ch2	300	FSK		1	0	0	0	0	0
Voice Mode	14400	ADPCM		1	1	1	0	0	0

Bit 11: Answer/Call - selects the answer mode or call mode handshake sequence for the modem type selected. This should only be changed when the modem is off-line.

Bit 10: Accept/Reject Remote Loop Request - this will allow or disallow response to remote digital loopback when requested by the far-end modem. This is valid for V.32terbo/V.32bis/ V.32, V.22bis, V.22 and Bell 212 modem types. This may be changed at any time.

Bits 9-8: Tx Clock - this selects the source of the transmit bit timing, either locked to the external clock XTCLK, internal on-board crystal or locked to the received clock RDCLK derived from the far-end modem signal.

Bit 7: Enable call waiting detection

Bit 6: Reserved - this bit is reserved for future use and should be set to 0.

Bits 5-0: Modem Type - these 6 bits select the modem type desired. When selecting a V.32terbo/V.32bis/V.32 configuration, the desired rates should be defined using the Set Rate Sequence Command 1NNN. The combinations of these two commands would have the effect of either setting a single speed, negotiating within a restricted set of speeds or allowing all possible speeds. When using a test command, the highest rate enabled is used.

response: The command is echoed back within 1.2 ms after it was written.

default: 2480 hex

Set Configuration Word 2 — 0x3*nnn*

function: This command writes 12 bits, specified by *nnn*, to the modem Configuration Word 2. The meaning and function of these bits are described below.

Meaning	Bit:	11	10	9	8	7	6	5	4	3	2	1	0
Enable/Disable Answer Tone		0/1											
No Tones			0	0									
550 Hz Guard Tone ON			0	1									
1800 Hz Guard Tone ON			1	0									
Echo Protection Tone ON			1	1									
Enable/Disable Auto-mode					0/1								
Dial-up/Leased-Line						0/1							
Enable/Disable Auto-retrain/Rate Renegotiation							0/1						
Asynchronous Normal								0	0				
Asynchronous Extended/HDLC								0	1				
Reserved								1	0				

 © 1995 Atari Corp.

Meaning		Bit:	11	10	9	8	7	6	5	4	3	2	1	0
Synchronous									1	1				
Serial V.14 Character Length:	8 bits										0	0		
	9 bits										0	1		
	10 bits										1	0		
	11 bits										1	1		
Serial/Parallel Data Mode													0/1	
Enable/Disable Adaptive MSE/RLSD Thresholds														0/1

Bit 11: Enable/Disable Answer Tone - the function of this bit depends on the state of Bit 11 of Configuration Word 1. When an answer mode handshake is selected (Configuration Word 1, Bit 11 = 0), clearing this bit enables the transmission of 3600 ms of 2100 Hz tone prior to beginning the appropriate handshake sequence according to V.25 recommendation. Setting this bit to one causes no 2100 Hz tone to be transmitted prior to the handshake sequence. When an originate mode handshake is selected (Configuration Word 1, Bit 11 = 1) this bit has no effect. This bit is not used with Bell 103 or Bell 212A modem types and will have no effect if these modem types are selected. This bit may be changed at any time.

Bits 10-9: Tones Selection - these two bits allow the generation of 550 and 1800 Hz guard tones for V.22bis and V.22 answer modes and echo protection tone for V.33, V.17, V.29 and V.27ter half-duplex modes. For other modem types, no tone (00) should be selected. These bits should only be changed when the modem is off line.

Bit 8: Enable/Disable Auto-mode - this feature supports Annex A of V.32terbo/V.32bis/V.32 CCITT recommendations and EIA PN-2330 (draft proposal) for automode handshake which allows the Voice Modem to automatically determine the mode of the far-end modem during handshake and to reconfigure itself appropriately. This feature works if the far-end modem is a V.32terbo/V.32bis/V.32, V.22bis, V.22, V.21, V.23, Bell 212A or Bell 103.

Bit 7: Dial-up/Lease-Line - this bit modifies the handshake from normal dial-up to a specified leased-line sequence if applicable.

Bit 6: Enable/Disable Auto-retrain and Auto-rate Renegotiation - if this feature is enabled, the Voice Modem will initiate a retrain or a rate renegotiation if the actual mean square error (MSE), which represents signal quality, is higher or lower than a dynamically set threshold. For a more detailed explanation refer to Section 8.2.

Bits 5-4: Async/Sync Select - these bits function in conjunction with Configuration Word 2, bit 1 as follows: If Configuration Word 2, bit 1=0 (serial data), then async mode is selected with bit 5=0. Bit 4 allows the choice of normal operation in the +1.0% to -2.5% rate range or extended operation in the +2.3% to -2.5% rate range according to V.14 recommendations. However, if bit 1=1 (i.e. parallel data), then bit 4=1 configures the data interface for HDLC operation and bit 4=0 for asynchronous (8,N,1) operation as described in the parallel data mode section. Synchronous operation, either in serial or parallel data mode, is selected by setting bit 4=1, bit 5=1.

Bit 1	Bit 5	Bit 4	Function
0	0	0	Serial Async Normal Rate
0	0	1	Serial Async Extended Rate
0	1	0	Reserved
0	1	1	Serial Synchronous
1	0	0	Parallel Async 10-bit Character
1	0	1	Parallel Sync w/HDLC
1	1	0	Reserved
1	1	1	Parallel Sync Bit Stream

Bits 3-2: Character Length - These bits are used to select the correct character length for the Serial V.14 async/sync converter. They are only used when the modem is operating in asynchronous serial mode (Configuration Word 2, bit 5=0, bit 1=0). The character length includes one start bit and one stop bit. Thus, the commonly used 7 data bit even parity one stop bit character format would require a character length of 10 bits (10). In asynchronous parallel mode (Configuration Word 2, bit 5=0, bit 4=0, bit 1=1), the character length is always 10 bits.

Bit 1: Serial/Parallel Data Mode - This bit configures the Voice Modem to pass data serially through the V.24 Pins RXD, TXD or in bytes through the controller interface. It is used in conjunction with Word 2, bit 4 and bit 5. ***Note:** Serial mode is not available in "V.32terbo" 19,200 bit/s mode.*

Bit 0: Enable/Disable Adaptive RLSD Detection - This bit enables or disables the adaptive determination of RLSD thresholds to enable fast and consistent RLSD\ loss detection. For a more detailed explanation refer to Section 8.5.

response: The command is echoed back within 1.2 ms.

default: 3000 hex

Set Bit Error Rate Target 0xA02*n*

function: This command sets the BER target for the auto-speed selection feature. This feature enables Voice Modem to automatically select the highest data rate allowable by the modems and supported by the line conditions such that BER does not exceed the target value. The command variable "*n*" assumes the following values:

n = 0 ; Disabled
n = 1 ; BER = 10E-6
n = 2 ; BER = 10E-5
n = 3 ; BER = 10E-4
n = 4 ; BER = 10E-3

response: The command is echoed back within 1.2 ms.

default: A021 hex

 © 1995 Atari Corp.

Enable Unsolicited Error Detection Responses 0xF207

function: The command allows the modem to return the 0xF3xx error check responses (if enabled) at the end of data packets

response: The command is echoed back within 1.2 ms.

Enable Error Detection Mode 0xB60n

function: Selects data modes:

Value of *n*	Meaning
0	non real-time data
1	real-time data, without error detection
2	real time data, with error detection

response: The command is echoed back within 1.2 ms.

0xB5xx

ta packet size to *xx* bytes.

will be echoed back within 1.2ms.

ed Line Status 0xA8F1

e unsolicited responses 0xA4xx (see unsolicited response section below)

nand will be echoed back within 1.2ms.

Detector 0x8C01

nand is used to detect presence or absence of dial tone within a very short

se of 8C01 means that a dial tone has been detected.

one was not detected, the response will be 8Cxx, where xx is not 01.

Set Data Packet Size

function: Set real time da

response: The command

default: 0xB504

Enable Unsolicit

function: Enable th

response: The comm

Report Dial Tone

function: This comm interval.

response: A respons

If a dial t

© 1995 Atari Corp.

The response is returned within 1.2 ms after the command was issued.

Set Voice Sampling Frequency 0xB30x

function: Set the compressed voice sampling frequency, as shown below:

Sample Rate	*x*
Adaptive sampling (Default)	0
3200 Hz	1
3600 Hz	2
4000 Hz	3
4400 Hz	4
4800 Hz	5
5200 Hz	6
5600 Hz	7
6000 Hz	8
6400 Hz	9
6800 Hz	A
7200 Hz	B

The default adaptive sampling rates are as follows:

Connection Speed	Sampling rate
19200 bps	7200 Hz
16800 bps	6800 Hz
14400 bps	4400 Hz
12000 bps	4400 Hz
9600 bps	3200 Hz

response: The command will be echoed back within 1.2ms

Dial Number/Transmit DTMF Tone 0x8A2x

function: This command is used to dial a digit based on the mode selected using the Set Dial Mode. The command is of the form 8A2*x* hex, where *x* denotes the digit to be dialed. The status of digit dialling can be known using the **Report Call Progress Detector** command.

x = 0 1 2 3 4 5 6 7 8 9 A B C D E F
Number = 0 1 2 3 4 5 6 7 8 9 * # A B C D

response: The command will be echoed back within 1.2ms.

Poll DTMF Detector 0x6800

function: This command starts the DTMF tone detector and returns the status of the detector with a response of 000x hex. The least significant digit of the response reports the DTMF tone pair received as follows:

x = 0 1 2 3 4 5 6 7 8 9 A B C D E F
DTMF Tone Pair = 0 1 2 3 4 5 6 7 8 9 * # A B C D

 © 1995 Atari Corp.

If no digit is detected, a response of FFFE hex is returned. The digit detected is held until it is read by the controller or another digit is detected.

response: A response is returned within 1.2 ms after it was written.

Report Handshake Status 0x8100

function: This command causes the Voice Modem to return a 12-bit response indicating the progress through the handshake, retrain or rate renegotiation.

response: The response is returned in the form of 8xyz hex, where x, y and z are shown below.

Example:
V.32bis handshake completed at 14.4k bit/s: 0x86B2
V.32bis handshake before rate determination: 0x8002
Auto-moding, no mode or rate is determined: 0x8000

Handshake/Retrain State	y
Undetermined	0
1200/75	1
75/1200	2
0-300	3
1200	4
2400	5
4800	6
7200	7
9600 Non-trellis	8
9600	9
12000	A
14400	B
16800	C
19200	D

Data Rate Response	z
Undetermined	0
V.32	1
V.32terbo/V.32bis	2
V.22bis	3
V.22	4
Bell 212	5
V.21	7
Bell 103	8
V.23	9
V.27	A
V.29	B
V.33	C
V.17	D

State	x
Auto-mode Handshake in Progress	0
Non-Automode Handshake in Progress	1
Abort/Idle	4
Retrain in Progress	3
Rate Renegotiation in Progress	5
Data Mode	6

response: The response is returned within 1.2 ms after the command is written.

Set Voice Volume 0xBA0x

function: Adjust voice volume. The allowable values for *x* are:

Level	x
Maximum volume (default)	0
-3 db	1
-6 db	2
-3 x db	x
Mute	F

response: The command will be echoed back within 1.2ms

Send Real Time Data 0xF0xx

function: Send data byte *xx* in real-time (low latency) mode.

The data byte *xx* will be sent once the controller has received a full packet of bytes (packet size is set by the B5*xx* command). The typical latency is around 18ms.

response: The command will be echoed back within 1.2ms

 © 1995 Atari Corp.

Unsolicited Response Reference

This section summarises the various types of unsolicited data that can be expected from the modem.

Receive Real Time Data — 0xF0xx

function: The byte *xx* was received from the remote modem.

If error detection has been enabled, note that the packet error status will only be received at the end of the packet (after all packet bytes have been received).

Packet Error Status — 0xF3xx

function: If error detection has been enabled (with the $B602 command), this reponse will be received after all bytes in a packet have been received. The format is:

F301 = No Errors in packet
F311 = Error ocurred in packet

Call Waiting Detected — 0xB1FF

function: When call waiting detection has been enabled with the 2C80 or 2480 command, this response indicates that a call waiting tone has been detected.

This response will be followed by a A4?? response, indicating that the line has been lost (see below).

Line Lost — 0xA4xx

function: This unsolicited response type is enabled with the command 0xA3FE. When enabled, the modem will report line lost, and occasionally also report that the line is still good. As shown in the parse data flow chart, the line good response needs to be taken into account, and discarded.

The least significant bit of the response indicates the line status:

%xxxx xxx1 = Line Lost
%xxxx xxx0 = Line Good

Only the LSB is valid. All other bits must be ignored.

Immediately subsequent to losing the line, the modem will switch back to analog mode, where the headset and microphone are connected to the analog line.

When a call waiting tone is detected by the remote modem, the local modem will just get this lost line response on its own. Both ends will in fact switch to analog mode, allowing the users to talk, take care of the call waiting, and then restart communications and handshaking.