QBXCTV

SoundBlaster(tm) Creative Voice Library

for

QuickBASIC 4.x and BASIC 7.x

by

Cornel Huth

5-May-1991

QBXCTV is a SoundBlaster(tm) compatible digitized voice input/output module for Microsoft QuickBASIC 4.x and BASIC 7.x compilers. Its purpose is to provide digitized voice output and input for my QBXSBC SoundBlaster/AdLib music card software library but it can also be used separately.

Features of QBXCTV are:

1) Linkable module for stand-alone or environment (QLB) use.
2) Written in fast and compact assembly code.
3) All 14 CT-VOICE.DRV functions supported.
4) DMA transfers for background operation of voice input and output.
5) All VOKXIT voice and packing formats supported.

This software package is copyrighted material. You may use it for non-commercial work only. If you are going to be using QBXCTV commercially, or in any work that is to be distributed in any manner, you must register by purchasing a license for the QBXSBC package.

QBXSBC, the package $19.95
- includes single-user licenses for:
1) QBXIOL, fast I/O DOS file module
2) QBXCTV, digitized voice I/O module for SoundBlaster
3) QBXFMI, interface module to the resident FM driver for the SoundBlaster and AdLib music cards. Includes QBXFMI.BAS and resident driver SB-SOUND.COM for the SoundBlaster.
- also includes:
4) Useful sample programs in QB demonstrating how to access AdLib instrument BNK files, voice files, and card detection.
5) Programmer documentation

QBXSBC, assembly source $29.95
- includes:
1) QBXIOL.ASM, MASM 5.1 compatible source
2) QBXCTV.ASM, MASM 5.1 compatible source
3) Additional programmer documentation

To order see the ORDER.FRM file.

LICENSE AGREEMENT - REGISTERED VERSIONS ONLY

This is a legal agreement between you, the end user, and Cornel Huth. By using this software, you are agreeing to be bound by the terms of this agreement.

SOFTWARE LICENSE

1. GRANT OF LICENSE. Cornel Huth grants to you the right to use one copy of the SOFTWARE on a single terminal connected to a single computer (i.e., with a single CPU). You may not network the SOFTWARE or otherwise use it on more than one computer or computer terminal at a time.

2. COPYRIGHT. The SOFTWARE is owned by Cornel Huth and is protected by United States copyright laws and international treaty provisions. Therefore, you must treat the SOFTWARE like any other copyrighted material (e.g., a book or musical recording) except that you may either (a) make one copy of the SOFTWARE solely for backup or archival purposes, or (b) transfer the SOFTWARE to a single hard disk provided you keep the original solely for backup or archival purposes. You may make a single copy of this document for your own use only.

3. OTHER RESTRICTIONS. You may not rent or lease the SOFTWARE, but you may transfer the software and accompanying documentation on a permanent basis provided you retain no copies and the recipient agrees to the terms of this Agreement. You may not reverse engineer, decompile, or disassemble the software. If the SOFTWARE is an update, any transfer must include the most recent update and all previous versions.

NO WARRANTIES. Cornel Huth disclaims all warranties, either expressed or implied, including but not limited to implied warranties of merchantability and fitness for a particular purpose, with respect to the SOFTWARE and the documentation.

NO LIABILITY FOR CONSEQUENTIAL DAMAGES. In no event shall Cornel Huth be liable for any damages whatsoever arising out of use of or inability to use this SOFTWARE.

U.S. GOVERNMENT RESTRICTED RIGHTS

The SOFTWARE and documentation are provided with RESTRICTED RIGHTS. Use, duplication, or disclosure by the Government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause at 52.227-7013. Contractor/manufacturer is Cornel Huth/6402 Ingram Rd/San Antonio, TX 78238.(512)684-8065.

This agreement is governed by the laws of the state of Texas.

FUNCTION LIST

All routines are functions eventhough some do not currently return status codes.

```
    FUNCTION CTVver(soft, hard)
    FUNCTION CTVport(BYVAL port)
    FUNCTION CTVirq(BYVAL irq)
    FUNCTION CTVdetect()
    FUNCTION CTVspeaker(BYVAL onoff)
    FUNCTION CTVusrstat(BYVAL vseg,BYVAL voff)
    FUNCTION CTVoutput(BYVAL vseg,BYVAL voff)
    FUNCTION CTVinput(BYVAL rate,BYVAL lbytes&,BYVAL vseg,BYVAL voff)
    FUNCTION CTVhalt()
    FUNCTION CTVuninstall()
    FUNCTION CTVpause()
    FUNCTION CTVcontinue()
    FUNCTION CTVskip()
    FUNCTION CTVexternal(BYVAL vseg, BYVAL voff)
    FUNCTION CTVstatus()
```

CTVver

Type FUNCTION - INTEGER

Arguments soft - INTEGER (returned)
hard - INTEGER (returned)

Syntax stat = CTVver(soft,hard)

Use Return the software version and the hardware version of the SoundBlaster.

Example 1 stat = CTVver(soft,hard)

Rules CTVdetect() should be called before CTVver().

Notes QBXCTV is compatible with version 1.14 of CT-VOICE.DRV. This software requires a hardware version of 1.03. My SoundBlaster, recently bought, reports a hardware version of 2.00.

Return 0 okay
-1 CTVdetect() not yet called

CTVport

Type FUNCTION - INTEGER

Arguments port - INTEGER (BYVAL)

Syntax stat = CTVport(port)

Use Configure QBXCTV to the base address of the IO port used by the SBC.

Example 1 stat = CTVport(&H220)

Rules The possible ports are 210h, 220h, 230h, 240h, 250h, or 260h, selected via a jumper on the SBC. QBXCTV defaults to port address 220h.

Notes See CTVSCAN.BAS for an example of using this routine.

Return 0 okay
-1 invalid port assignment

CTVirq

Type FUNCTION - INTEGER

Arguments irq - INTEGER (BYVAL)

Syntax stat = CTVirq(irq)

Use Configure QBXCTV to the IRQ line used by the SBC.

Example 1 stat = CTVirq(7)

Rules The possible IRQs are 2, 3, 5, or 7. QBXCTV defaults to IRQ 7.

Notes See CTVSCAN.BAS for an example of using this routine.

Return 0 okay
-1 invalid IRQ assignment

CTVdetect

Type FUNCTION - INTEGER

Arguments none

Syntax stat = CTVdetect

Use Detect the SoundBlaster card.

Example 1 stat = CTVdetect

Rules

This function *MUST* be called before any of the following functions.

Notes

See CTVSCAN.BAS for an example in using this routine. CTVdetect() requires that QBXCTV be configured with the correct IO port and IRQ line used by the SBC.

On the SBC is a DMA ENABLE JUMPER (DRQ1). Normally this is enabled. Since the SBC always uses DMA channel 1 you may be unable to use other hardware devices that also use channel 1. It's much easier removing the jumper than the card. This can also affect software - PCBACKUP, from Central Point Software, won't work properly in high-speed mode unless you un-jump DRQ1.

Return

0 okay
1 voice card fails (or obsolete hardware)
2 I/O R/W fails
3 DMA fails

CTVspeaker

Type

FUNCTION - INTEGER

Arguments

OnOff - INTEGER (BYVAL)

Syntax	stat = CTVspeaker(OnOff)
Use	Turn on or off the speaker output of the SBC.
Example 1	stat = CTVspeaker(0)
Rules	The speaker must be on for a voice to be produced.
Notes	OnOff=0 speaker off, OnOff<>0 speaker on.
Return	none

CTVusrstat

Type	FUNCTION - INTEGER
Arguments	vseg - INTEGER (BYVAL) voff - INTEGER (BYVAL)
Syntax	stat = CTVusrstat(vseg,voff)
Use	Assign the QBXCTV status to a QuickBASIC variable.
Example 1	DIM SHARED STAT AS INTEGER stat = CTVusrstat(VARSEG(STAT),VARPTR(STAT))
Rules	If the assignment is made it must be to a main module variable.
Notes	Since QBXCTV use background DMA processing it is possible for you to move things around within the QB environment while QBXCTV is still playing. If the usrstat variable changes location, then QBXCTV will be updating an unknown area of memory. An alternate but slower method is to use the CTVstatus() function while in the environment.
Return	none

CTVoutput

Type FUNCTION - INTEGER

Arguments vseg - INTEGER (BYVAL)
voff - INTEGER (BYVAL)

Syntax `stat = CTVoutput(vseg,voff)`

Use Output the voice data at vseg:voff.

Example 1

```
DIM Buff(0 TO 32000) AS INTEGER
vseg = VARSEG(Buff(0)): voff = VARPTR(Buff(0))
LoadVoiceData "VOICE.VOC",vseg,voff
'turn on the speaker, speak, wait 'til done
stat = CTVspeaker(1)
stat = CTVoutput(vseg,voff)
DO:LOOP WHILE CTVstatus
stat = CTVuninstall
```

Rules Only one voice can be played at a time.

Notes See CTVOUT.BAS for an example of using this routine. When this routine is active usrstat or CTVstatus() return -1.

Return 0 okay
1 voice already active

CTVinput

Type FUNCTION - INTEGER

Arguments SR - INTEGER (BYVAL)
length& - LONG (BYVAL)
vseg - INTEGER (BYVAL)
voff - INTEGER (BYVAL)

Syntax

```
stat = CTVinput(SR,length&,vseg,voff)
```

Use Input length& bytes of voice data to the buffer at vseg:voff using a sampling rate of SR Hz. The hardware does all writing to RAM in proper voice file format.

Example 1

```
DIM Buff(0 TO 24000) AS INTEGER
vseg = VARSEG(Buff(0)): voff = VARPTR(Buff(0))
turn off speaker, input, wait 'til done
stat = CTVspeaker(0)
stat = CTVinput(12000,48000&,vseg,voff)
DO:LOOP WHILE CTVstatus
stat = CTVuninstall
```

Rules Only one voice can be played or input at a time. Sampling rate can be 4000Hz - 13000Hz. length& can be any length&. Time of recording in seconds is length&/sampling rate.

Notes See CTVIN.BAS for an example of using this routine. When this routine is active usrstat or CTVstatus() return -1.

Return 0 okay
1 voice already active

CTVhalt

Type	FUNCTION - INTEGER
Arguments	none
Syntax	stat = CTVhalt
Use	End the current voice.
Example 1	stat = CTVhalt
Rules	none
Notes	Once the voice has been halted it cannot be restarted.
Return	0 okay 1 no voice being output

CTVuninstall

Type	FUNCTION - INTEGER
Arguments	none
Syntax	stat = CTVuninstall
Use	Halt any voice and turn speaker off.
Example 1	stat = CTVuninstall
Rules	none
Notes	Same as calling CTVhalt() and CTVspeaker(0).
Return	none

CTVpause

Type	FUNCTION - INTEGER
Arguments	none
Syntax	stat = CTVpause
Use	Pause voice output until CTVcontinue().

Example 1	stat = CTVpause
	DoHighSpeedXfer
	stat = CTVcontinue

Rules	none
Notes	none
Return	0 okay 1 no voice to pause

CTVcontinue

Type	FUNCTION - INTEGER
Arguments	none
Syntax	stat = CTVcontinue
Use	Continue voice output that had been CTVpause().

Example 1	stat = CTVpause
	DoHighSpeedXfer
	stat = CTVcontinue

Rules	none
Notes	none
Return	0 okay 1 no voice to continue

CTVskip

Type FUNCTION - INTEGER

Arguments none

Syntax stat = CTVskip

Use Skip to the end of a REPEAT voice BLOCK and continue output from there.

Example 1

```
INPUT "Press a key to end main theme",a$
stat = CTVskip
PRINT "Now playing ending music"
```

Rules none

Notes To build repeat blocks see the JOINTVOC.EXE program on your original SoundBlaster disk.

Return 0 okay
1 not in repeat block

CTVexternal

Type FUNCTION - INTEGER

Arguments vseg - INTEGER (BYVAL)
voff - INTEGER (BYVAL)

Syntax stat = CTVexternal(vseg,voff)

Use Programmer-supplied voice output routine located at vseg:voff.

Example 1

```
LoadCustomCTV vseg,voff
stat = CTVexternal(vseg,voff)
SpeakCustom
DO:LOOP WHILE CTVstatus
stat = CTVexternal(0,0)
```

Rules   none

Notes   This is for advanced use. This allows the programmer to supply his own SPK_ABIT routine (see the QBXCTV.ASM source file for more). Once set you can return to the default SPK_ABIT code by sending vseg and voff both = 0.

Return   none

CTVstatus

Type   FUNCTION - INTEGER

Arguments   none

Syntax	stat = CTVstatus
Use	Returns the current state of the voice driver.

Example 1	stat = CTVspeaker(0)
	stat = CTVinput(12000,48000&,vseg,voff)
	DO:LOOP WHILE CTVstatus
	PRINT "Done playing"

Rules	none
Notes	See CTVusrstat().
Return	0 voice not active -1 voice active