CUSTOM PROGRAMMING

FOR THE TCR-500/500PCI

SMPTE - EBU & IRIG-B

TIME CODE READER

document version 1.0

2484 West Clay
St. Charles, MO 63301
phone: (636) 724-3666 fax: (636) 724-3776

TABLE OF CONENTS

Introduction

This document is directed toward those interested in developing device drivers or custom applications to interface directly to the TCR-500/500PCI time code reader cards. It is written for the perspective of IBM-PC/compatible developers, but may also be relevant with respect to the TCR-500PCI in other computer architectures which implement a PCI bus. Note: Windows NT/2000 developers should utilize the API library software and drivers already developed by W C&A.

For the remainder of the document "TCR" will refer to the TCR-500 (ISA) or TCR-500PCI model time code reader cards. The actual products will be differentiated in the documentation where appropriate. All numbers appearing this document followed by the character 'h' are represented in hexadecimal notation. All other numbers are in decimal notation.

The following information is made available as a matter of courtesy to users of the TCR time code reader card who may wish to make use of additional features of the card that are not available with the standard software that is provided. Utilization of the information supplied herein requires some proficiency in computer programming.

Legal

W Clark & Associates, Ltd. does not provide any warranty on the information or features described herein nor will any further support of any kind be given except on a custom basis. W C&A reserves the right to make changes to the TCR without notification and there are no guarantees that the software features described herein will be supported in the future. W C & A makes no warranty of any kind with regard to this material, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. W C&A shall not be liable for errors contained herein or for incidental or consequential damages in connection with the furnishing, performance, or use of this material.

TCR Resource Configuration/Identification

TCR-500 (ISA)

The TCR-500 is an ISA bus time code reader card with SMPTE and IRIG-B(1) decoding capability. Configuration of TCR-500 is performed manually by jumper configurations on the card.

Card jumper J1 (located in the upper-right corner of the card near the LED) sets time code decoding mode. Jumper ON sets IRIG-B(1) decoding. Jumper OFF sets SMPTE decoding (all formats).

I/O ports are selected by the P5 and P6 jumpers in the left-center of the card. The selections may be configured among I/O port address 300h, 310h, 320h, or 330h. See the truth-table silkscreened onto the card underneath of the jumpers for instructions.

Interrupt selection is configured by jumpers in the lower-left hand corner of the hard. Supported interrupts are 5, 7, 10, 11, 12, and 15.

TCR-500PCI

The TCR-500PCI is a PCI bus time code reader card with SMPTE and IRIG-B(1) decoding capability.

The card is PCI bus 2.1 compliant and uses a type-0 PCI configuration header. The TCR-500PCI is identified by Vendor ID 10b5h, Device ID 9050h, Subsystem Vendor ID 5743h, Subsystem Device ID (16-bit binary) 0010 000001 *xxxxxx,* where *xxxxxx* represents EEPROM revision (not firmware version). For example, TCR-500PCI with EEPROM revision 0 will have Subsystem Device ID 4040h.

Card jumper J1 (located in the upper-right corner of the card near the LED) sets time code decoding mode. Jumper ON sets IRIG-B(1) decoding. Jumper OFF sets SMPTE decoding (all formats).

I/O port and interrupt configurations are dynamically programmed by the computer's BIOS during system boot-up. It is possible for these values to change between boots especially if devices are added or removed from the PCI bus.

The I/O port setting used to communicate with the TCR-500PCI will be located in the PCI Base Address 3 configuration register. The least-significant bit will be a '1' indicating I/O address space. This '1' should be masked off to obtain the true address value. The value can be reprogrammed if necessary. If reprogrammed, the least-significant bit must again be set to 1 to indicate I/O address space. The address must be doubleword aligned (bits 2-4 set to zero).

The interrupt setting for the TCR-500PCI will be located in the Interrupt Pin configuration register. This value is set by the system BIOS and routing is programmed to the PCI host bridge. The value cannot be reprogrammed.

Applications which do not wish to incorporate PCI device interrogation code may utilize the Legacy Support Tool (LEGACY.EXE) provided on the TCR-500PCI software diskette to determine the resources assigned to the card after each system startup.

Porting Custom Applications From TCR-500 to TCR-500PCI

Custom applications written for a TCR-500 (ISA) card can be easily ported to a TCR-500PCI. The TCR-500PCI protocol is backward compatible with the TCR-500 ISA card. However, due to the dynamic nature of the PCI bus the following issues should be address when upgrading from the ISA to PCI card model.

1. TCR-500PCI resources are assigned dynamically as discussed in the Resource Configuration – TCR500PCI section. Applications must be prepared to dynamically configure themselves after each system startup. An application may choose to interrogate the TCR-500PCI itself to learn resource configuration settings, or use the Legacy Support Tool to obtain the information.

2. Unlike the TCR-500 ISA card, interrupts are not enabled by default. For interrupts to be generated by the card they must be enabled through the global interrupt enable command 90h. See the section entitled TCR-500PCI Global Interrupt Enable/Disable for more information.

3. Response timing for commands on the TCR-500PCI will not be identical to the TCR-500 (ISA) due to the different bus architecture. Applications should behave either entirely synchronously or asynchronously (via command 29h interrupt-on-each-response) to avoid inconsistencies. Synchronous applications should always read/verify the response for a current command before sending a new command. (The TCR-500PCI cannot necessarily process commands as fast as the host processor can transmit them.)

Communication With TCR Card

Communication with the TCR is accomplished by sending an 8-bit command to its I/O port address then awaiting an 8-bit reply. Reply time can vary depending on command sent but will never exceed a few milliseconds. Generally, a reply to a command will remain available until a new command is sent. The exception to this rule is that an interrupt status value I/O reply will be sent when any interrupt occurs.

Synchronous Communication

Synchronous communication is accomplished by sending a command to the card then busy-waiting for the response while reading the card's I/O port address, checking for an appropriate response. This mechanism

can tie up the computer in "do-nothing" loops but is easier to design interfaces for – practical for most applications.

Asynchronous Communication

Asynchronous communication utilizes the "interrupt on each response" feature of the TCR. When enabled, the card will interrupt the PC directly after placing a command's reply on the I/O port. Applications which do not wish to tie up the computer's CPU waiting for replies from the TCR should use this mechanism. Asynchronous communication requires the development of an interrupt service routine.

Reading Time Code From The Card

Beginning of Read Cycle

The TCR is always reading/processing time code (when present). The last good/complete frame of time code read is kept in an internal buffer. This internal buffer is updated at the time code's frame rate. To enable time code reads that are asynchronous and non-volatile a special read buffer area is implemented.

Command: Receive Time Read Request -- e8h (response e8h)

The command duplicates the last good/complete frame read data into a non-volatile read buffer. The values in the read buffer will not change until another e8h command is sent. During the copy a 4-bit exclusive-or'ed checksum of the read buffer data is calculated.

Command: Request Time Code Frame Checksum -- 2fh (response 02fh then 0*x*h where *x* is checksum)

Reading out the checksum as part of a time code frame read is recommended. Compare the remote checksum with a locally-calculated checksum to verify that no data was corrupted during bus transfer. Use of the checksum feature requires reading out the data values for all time code read commands 68h – 6fh and a8h – afh.

Read Time Code Data

A full time code frame is read from the card in a series of nibbles. Commands 68h – 6fh and a8h – afh read the time code nibbles. The response/confirmation for the command is in the high nibble of the reply and the data value in the low nibble. The data portion in the command's reply will be different for SMPTE and IRIG-B formats but the response mechanism is the same. See Command Reference Chart for response values (*x* indicates data value). Only commands for the data values desired need be sent. They can be sent in any order.

No Time Code Indication

When no time code is present or the card is undergoing time code level calibration/identification no time information is available. Under such circumstances the value 0fh will be read out in the frames units nibble for SMPTE time code and the seconds units nibble for IRIG-B time code.

Time Code Read Latency Correction

For SMPTE and IRIG-B time codes, the time data encoded within the signal is valid at the beginning of signal generation for the given time code frame. However, time elapses as the signal is decoded by the TCR. The elapsed time is the frequency of the time code signal. One SMPTE frame elapses as a SMPTE frame is decoded. One second elapses as an IRIG-B frame is decoded.

As a convenience to applications accessing the TCR, the card internally corrects the time registers such that the data is still in real time when made available to be read by an application. However, other information within the time code frame will not be altered in any way. Therefore, if custom applications require non-time data within a frame (such as a user bits in SMPTE or control functions in IRIG-B) to be time-matched said application subtract one frame from the time data as read. This means subtracting one frame unit from SMPTE time data and one second from IRIG-B time data.

TCR Interrupts

The TCR supports the following interrupt types. Directly before throwing an interrupt the TCR will place the interrupt status value on the I/O port. The first task for an interrupt service routine is to read the TCR's I/O port to determine the interrupt status value. The also has the effect of clearing the interrupt on the TCR card.

Interrupt Types

Type: Time Set
Frequency (SMPTE): Once/minute when seconds=30 and frames=0, when time code is present
Frequency (IRIG-B): Once/minute when seconds=30, when time code is present
Status value: 68h
Enabled by default: Yes
Maskable: Yes, by Once/Second or Start of Time Code interrupts

Type: Gained Time Code (from lost state)
Frequency: When condition occurs
Status value: 69h
Enabled by default: Yes
Maskable: No

Type: Lost Time Code (from reception state)
Frequency: When condition occurs
Status value: 6ah
Enabled by default: Yes
Maskable: No

Type: End of Time Code (SMPTE only, occurs after user group 8-before sync word)
Frequency: SMPTE frame rate
Status value: 6bh
Enabled by default: No, enable with command 2ah
Maskable: Yes, by command 28h (clear enhanced interrupts)

Type: Once/Second (occurs at the on-time mark beginning of each second)
Frequency: Once/second (1Hz)
Status value: 6dh
Enabled by default: No, enable with command 2ch
Maskable: Yes, by command 28h (clear enhanced interrupts) or by Start of Time Code interrupt.

Type: Start of Time Code (occurs at the on-time mark for each time code frame)
Frequency (SMPTE): frame rate
Frequency (IRIG-B): once/second (functionally identical to Once/Second interrupt)
Status value: 6ch
Enabled by default: No, enable with command 2bh
Maskable: Yes, by command 28h (clear enhanced interrupts)

Type: Interrupt On Each Response (interrupt accompanies replies to all commands)
Frequency: as commands are sent to TCR
Status value: reply value for last command sent
Enabled by default: No, enable with command 29h
Maskable: Yes, by command 28h (clear enhanced interrupts)

Note: clear enhanced interrupts (command 28h) clears all currently-enabled enhanced interrupts on the TCR. Enhanced interrupts are any of Once/Second, Start of Time Code, End of Time Code, and Interrupt on Each Response.

TCR-500PCI and Global Interrupt Enable/Disable

The TCR-500PCI has a global interrupt enable/disable function. Command 90h globally enables interrupts and command 91h globally disables them. By default, interrupts are disabled. This feature operates independently of the 28h-2fh commands that enable/disable enhanced interrupt features in the card. Interrupts cannot be enabled for a PCI device unless a driver or application is present to handle then clear them. Before unloading themselves, all custom drivers/applications that enabled interrupts via command 90h must disable them via command 91h or the system will hang.

Other Advanced Features

Gain Control

Custom applications have full control over the gain settings for the TCR. Manual control of gain level is not recommend unless a specific goal has been identified.

Command: Recalibrate Gain Level -- 2dh (response 2dh)

This command instructs the TCR to again perform the signal level calibration procedure that was executed at the start of the last card reset. Normally the TCR performs the calibration once, only at card reset

Command: Set Gain Level -- 7*xxxx* (response 24h), 8*yyyy* (response 25h), eeh (response eeh)

This command manually sets the gain level for the incoming time code signal. A gain value is loaded via the 7*xxxx* and 8*yyyy* commands, where *xxxx* is the low nibble of the new gain setting and *yyyy* is the high nibble of the new gain setting. Command eeh is sent when custom application is ready for the new gain level to be applied. Gain setting is an unsigned 8-bit value in the range 0-255, with 0 representing the largest gain, 254 the smallest gain, and 255 a short.

Calibration Status

Command: Request Calibration Status – e9h (response e9h, then *xx*h)

Requests the current status of gain level/time code detection on the card. A value of 1 in *xx* indicates that the card is still calibrating. During calibration, the card will indicate a no time code status.

Configuration

Command: Request Time Code Configuration – 2eh (response 2eh, then *xx*h)

Requests the configuration for J1 jumper (SMPTE or IRIG-B time code operation). A value of 1 in *xx* indicates SMPTE decoding mode, a value of 2 indicates IRIG-B decoding.

TCR Command Protocol Reference Chart

SMPTE COMMAND SET

Command		Description	Response
7*xxxx*	-	Set AD8402 pot register low nibble	24h
8*xxxx*	-	Set AD8402 pot register high nibble	25h
90h	-	Global interrupt enable **(TCR-500PCI only)**	26h
91h	-	Global interrupt disable **(TCR-500PCI only)**	27h
28h	-	Clear enhanced features	28h
29h	-	Set host interrupt on each response	29h
2ah	-	Set host interrupt at end of time code	2ah
2bh	-	Set host interrupt at start of time code	2bh
2ch	-	Set host interrupt once/second at start of second	2ch
2dh	-	request TCR-500PCI re-calibration	2dh
2eh	-	request TCR-500PCI configuration (*xx*=01h for SMPTE)	2eh,*xx*h
2fh	-	request time code frame checksum	2fh,0*x*h
e8h	-	Receive time read request	e8h
e9h	-	Request TCR-500PCI calibration status (0=done, 1=active)	e9h,*xx*h
edh	-	Request TCR-500PCI mode (always receive)	ebh
eeh	-	Set AD8402 pot to pot register	eeh
efh	-	Dummy command	efh
68h	-	read frame units	0000*xxxx*
69h	-	read user group 1	0001*xxxx*
6ah	-	read frame tens	0010*xxxx*
6bh	-	read user group 2	0011*xxxx*
6ch	-	read second units	0100*xxxx*
6dh	-	read user group 3	0101*xxxx*
6eh	-	read second tens	0110*xxxx*
6fh	-	read user group 4	0111*xxxx*
a8h	-	read minute units	1000*xxxx*
a9h	-	read user group 5	1001*xxxx*
aah	-	read minute tens	1010*xxxx*
abh	-	read user group 6	1011*xxxx*
ach	-	read hour units	1100*xxxx*
adh	-	read user group 7	1101*xxxx*
aeh	-	read hour tens	1110*xxxx*
afh	-	read user group 8	1111*xxxx*

IRIG-B COMMAND SET

Command		Description	Response
7*xxxx*	-	Set AD8402 pot register low nibble	24h
8*xxxx*	-	Set AD8402 pot register high nibble	25h
90h	-	Global interrupt enable **(TCR-500PCI only)**	26h
91h	-	Global interrupt disable **(TCR-500PCI only)**	27h
28h	-	Clear enhanced features	28h
29h	-	Set host interrupt on each response	29h
2bh	-	Set host interrupt at start of time code	2bh
2dh	-	request TCR-500PCI re-calibration	2dh
2eh	-	request TCR-500PCI configuration (*xx*=02h for IRIG-B)	2eh,*xx*h
2fh	-	request time code frame checksum	2fh,0*x*h
e8h	-	Receive time read request	e8h
e9h	-	Request TCR-500PCI calibration status (0=done, 1=active)	e9h,*xx*h
edh	-	Request TCR-500PCI mode (always receive)	ebh
eeh	-	Set AD8402 pot to pot register	eeh
efh	-	Dummy command	efh
68h	-	read second units	0000*xxxx*
69h	-	read second tens	0001*xxxx*
6ah	-	read minute units	0010*xxxx*
6bh	-	read minute tens	0011*xxxx*
6ch	-	read hour units	0100*xxxx*
6dh	-	read hour tens	0101*xxxx*
6eh	-	read day units	0110*xxxx*
6fh	-	read day tens	0111*xxxx*
a8h	-	read day hundreds	1000*xxxx*
a9h	-	read control functions 1 (bits 51-54)*	1001*xxxx*
aah	-	read control functions 2 (bits 55-58)*	1010*xxxx*
abh	-	read control functions 3 (bits 59, 61-63)*	1011*xxxx*
ach	-	read control functions 4 (bits 64-67)*	1100*xxxx*
adh	-	read control functions 5 (bits 68-69, 71-72)*	1101*xxxx*
aeh	-	read control functions 6 (bits 73-76)*	1110*xxxx*
afh	-	read control functions 7 (bits 77-79)*	1111*xxxx*

* The least significant bit within a control function group response nibble is the lowest-numbered IRIG-B bit of the bits it contains. See IRIG-B specification for information on the placement of these bits within the code frame. Bit numbers assume bit numbering starting at 1.