

# TCI-500 SERIAL PROTOCOL SPECIFICATION

document version 1.3

Copyright © 1998-2001 Masterclock, Inc.

## Introduction

Serial communication with the TCI-500 is accomplished by the exchange of messages. A command message is sent to the TCI-500 from the user application. A response message is returned to the user application from the TCI-500. A response message will consist of a single response for query-type messages or multiple responses on some specified interval for mode-type messages.

A message will be defined as a group of bytes including a header, ID, data size, and data. The message header is used by both the TCI-500 and the user application to identify the message and message type. A message header consists of the hexadecimal byte sequence 0xFF 0xAD (255 173). A message ID will be one byte from the list of messages which the TCI-500 recognizes.

In this document all numbers are in decimal notation except for those preceded by '0x'. Such numbers are in hexadecimal notation.

## Document Revision History

- 1.1 - Removed classification for generator shutdown status #1.
- 1.2 - Defined byte 9 in message 15 response for TCI-500 firmware version 1.4
  - Changed document name to reflect changed product name.
- 1.3 - Fixed documentation error in rx/tx breakdown chart in messages #15 and #16.

## Command messages - to the TCI-500 from the user application

The format for transmitting command messages to the TCI-500 is as follows:

<msg header><msg id>[<data bytes.....>]<checksum>

The data portion of a command message will vary depending on the type of message being sent. Some messages do not require any data to be transmitted so none should be sent. The TCI-500 knows the required number of data bytes for each command message. If a command message is transmitted without the appropriate number of data bytes it will be rejected.

A checksum is required for all messages sent to the TCI-500. The checksum is calculated by exclusive-or'ing together the message id and all data bytes. The message header sequence (0xFF 0xAD) is not included in the checksum calculation. For command messages that contain no data bytes the message ID is sent (again) as the checksum.

## Response messages - from the TCI-500 to the user application

The format for the response messages from the TCI-500 is as follows:

<msg header><msg id><data size><data bytes...><checksum>

The message ID indicates to which command message the response is directed. The data size specifies the number of response data bytes forthcoming, including the checksum. All response messages contain at least one data byte.

## Mode-type message

Mode-type messages are enabled and disabled. When enabled they output on a given interval, usually once per second at the valid start of the second. Unless otherwise stated these messages, even when enabled, will not begin to transmit on their given intervals until stable time code decoding is established.

## Query-type messages

These messages return a one-time response.

## Firmware version notes

The current TCI-500 firmware version is 1.4. If your firmware version is not current some messages may not be supported. The 'Firmware' section for each message will indicate if that message has been added since firmware version 1.0. All user applications should send message ID ??? to verify that TCI-500 firmware version supports message(s) that the user application requires.

### Physical connections to the TCI-500

The TCI-500 has the following I/O connections for RS-232 communications on the rear DB-9 connector:

- pin 2 - transmit (TX)
- pin 3 - receive (RX)
- pin 5 - ground

For communications with the TCI-500 from a standard IBM PC or compatible computer a standard null-modem cable may be used. Please observe the following requirements:

- All communications with the TCI-500 are at 9600 baud, 8 data bits, 1 stop bit, no parity.
- Connect the transmit (TX) line of the TCI-500 (pin 2 of the DB-9 connector) to the receive (RX) line of the host system.
- Connect the receive (RX) line of the TCI-500 (pin 3 of the DB-9 connector) to the transmit (TX) line of the host system.
- Connect the ground line of the TCI-500 (pin 5 of the DB-9 connector) to a ground on the host system, preferably a ground on the host's serial interface.
- Be sure the host system is capable of communicating by standard RS-232 at 9600 baud, 8 data bits, 1 stop bit, no parity. The TCI-500 can not decode TTL-level serial communications at the DB-9 connector.
- Be sure that any cable you are using for communication with the TCI-500 is within RS-232 standard length and is a working cable. Check the cable wiring with an ohmmeter if necessary.

Name: Auto-transmit generator time data  
ID: 0  
Type: Mode  
Firmware: 1.0

Description: Enable or disable automatic transmission of generator time data once per second at the valid start of the second. The valid start is defined as the 1Hz "on-time mark" of the time code being generated. Automatic transmission of this message begins on the first second after the second during which the enable message is received.

#### Command format:

Byte	Value	Description
1	0xFF	message header
2	0xAD	“ “
3	0x00	command ID
4	0x??	0x01 to enable this mode, 0x00 to disable
5	0x??	Checksum

#### Response format:

Byte	Value	Description
1	0xFF	message header
2	0xAD	“ “
3	0x00	response ID
4	0x04	response data size
5	0x??	hour (0 - 23)
6	0x??	minute (0 -59)
7	0x??	second (0 -59)
8	0x??	Checksum

Name: Auto-transmit generator time/date data  
ID: 1  
Type: Mode  
Firmware: 1.0

Description: Enable or disable automatic transmission of generator time/date data once per second at the valid start of the second. The valid start is defined as the 1Hz “on-time mark” of the time code being generated. Automatic transmission of this message begins on the first second after the second during which the enable message is received. This mode is only available if date information is detected in decoded time code.

Command format:

Byte	Value	Description
1	0xFF	message header
2	0xAD	“ “
3	0x01	command ID
4	0x??	0x01 to enable this mode, 0x00 to disable
5	0x??	Checksum

Response format:

Byte	Value	Description
1	0xFF	message header
2	0xAD	“ “
3	0x01	response ID
4	0x08	response data size
5	0x??	hour (0 - 23)
6	0x??	minute (0 -59)
7	0x??	second (0 -59)
8	0x??	month (1 - 12)
9	0x??	day (1 - 31)
10	0x??	year (low 8 bits of 16 bit value)
11	0x??	year (high 8 bits of 16 bit value)
13	0x??	Checksum

Name: Auto-transmit time code decoder time data  
ID: 4  
Type: Mode  
Firmware: 1.0

Description: Enable or disable automatic transmission of the decoder time data once per second at the valid start of the second. The valid start is defined as the 1Hz “on-time mark” of the time code being decoded. Automatic transmission of this message begins on the first second after the second during which the enable message is received. The data in this message is read-latency corrected so that it is in real-time when output.

Command format:

Byte	Value	Description
1	0xFF	message header
2	0xAD	“ “
3	0x04	command ID
4	0x??	0x01 to enable this mode, 0x00 to disable
5	0x??	Checksum

Response format:

Byte	Value	Description
1	0xFF	message header
2	0xAD	“ “
3	0x04	response ID
4	0x04	response data size
5	0x??	decoder hour (0 - 23)
6	0x??	decoder minute (0 - 59)
7	0x??	decoder second (0 - 59)
8	0x??	Checksum

Name: Auto-transmit time code decoder time/date data  
 ID: 5  
 Type: Mode  
 Firmware: 1.0

Description: Enable or disable automatic transmission of the decoder time/date data once per second at the valid start of the second. The valid start is defined as the 1Hz “on-time mark” of the time code being decoded. Automatic transmission of this message begins on the first second after the second during which the enable message is received. This mode is only available if date information is detected in decoded time code. The data in this message is read-latency corrected so that it is in real-time when output.

Command format:

Byte	Value	Description
1	0xFF	message header
2	0xAD	“ “
3	0x05	command ID
4	0x??	0x01 to enable this mode, 0x00 to disable
5	0x??	Checksum

Response format:

Byte	Value	Description
1	0xFF	message header
2	0xAD	“ “
3	0x05	response ID
4	0x08	response data size
5	0x??	decoder hour (0 - 23)
6	0x??	decoder minute (0 -59)
7	0x??	decoder second (0 -59)
8	0x??	decoder month (1 - 12)
9	0x??	decoder day (1 - 31)
10	0x??	decoder year (low 8 bits of 16 bit value)
11	0x??	decoder year (high 8 bits of 16 bit value)
13	0x??	Checksum

Name: TCI-500 operation information  
 ID: 15  
 Type: Query  
 Firmware: 1.0 (byte 9 “decoder additional status” available in firmware 1.4+)

Description: Retrieves TCI-500 current operational information.

Command format:

Byte	Value	Description
1	0xFF	message header
2	0xAD	“ “
3	0x0F	command ID
4	0x0F	checksum

Response format:

Byte	Value	Description
1	0xFF	message header
2	0xAD	“ “
3	0x0F	response ID
4	0x09	response data size
5	0x??	decoder operation (high nibble=time code type, low nibble=time code group) <sup>1</sup>
6	0x??	generator operation (high nibble=time code type, low nibble=time code group) <sup>1</sup>
7	0x??	decoder status, one of: 0 = searching/calibrating 1 = classifying 2 = decoding 3 = time code loss
8	0x??	generator status, one of: 0 = idle 1 = generating
9	0x??	time code decoder additional status <sup>2</sup>
10	0x00	<i>reserved</i>
11	0x00	<i>reserved</i>
12	0x00	<i>reserved</i>
13	0x??	checksum

<sup>1</sup> Currently defined time code constants – see Appendix B

<sup>2</sup> Currently defined are:

bit 0 - full date (month, day, and year) availability on decoder (1 = yes, 0 = no)

Name: TCI-500 version information  
 ID: 16  
 Type: Query  
 Firmware: 1.0

Description: Retrieves firmware version information for TCI-500 product.

Command format:

Byte	Value	Description
1	0xFF	message header
2	0xAD	“ “
3	0x10	command ID
4	0x10	checksum

Response format:

Byte	Value	Description
1	0xFF	message header
2	0xAD	“ “
3	0x10	response ID
4	0x06	response data size
5	0x??	TCI-500 firmware version (major)
6	0x??	TCI-500 firmware version (minor)
7	0x00	<i>reserved</i>
8	0x00	<i>reserved</i>
9	0x00	<i>reserved</i>
10	0x??	checksum

Name: Auto-transmit diagnostics information  
 ID: 17  
 Type: Mode  
 Firmware: 1.0

Description: Enable or disable automatic transmission of TCI-232 run-time diagnostics information. See

Appendix B for diagnostics packet codes and content.

Command format:

Byte	Value	Description
1	0xFF	message header
2	0xAD	“ “
3	0x11	command ID
4	0x??	0x01 to enable this mode, 0x00 to disable
5	0x??	checksum

Response format:

Byte	Value	Description
1	0xFF	message header
2	0xAD	“ “
3	0x11	response ID
4	0x??	response data size
5	0x??	diagnostics code ID
...	0x??	diagnostics data <sup>1</sup>
...	0x??	checksum

<sup>1</sup>Diagnostics data size will vary and will be reflected appropriately in response byte 4 (data size).

Name: Error Messages  
ID: 255  
Type: Response-only  
Firmware: 1.0

Description: Indicates than an RS-232 command message has been rejected. The error code and extended error code fields in the response indicate why the message was rejected.

Response format:

Byte	Value	Description
1	0xFF	message header
2	0xAD	“ “
3	0xFF	response ID
4	0x04	response data size
6	0x??	ID of message which was rejected (if in response to invalid message, otherwise 0xff)
7	0x??	error code <sup>1</sup>
8	0x??	extended error code <sup>2</sup>
9	0x??	checksum

<sup>1</sup> Possible error codes are:

- 1 = checksum failure
- 2 = invalid request for current TCI-500 operation mode
- 3 = unrecognized packet ID

<sup>2</sup> Extended error codes are message dependent. A value of 0 in this field indicates no extended error information is available. See individual message descriptions for potential extended error responses.

**APPENDIX A - TIME CODE GROUP/TYPE DEFINITIONS**

Time Code Types		Time Code Groups			Value
SMPTE (30 frames/sec)	0x10	SMPTE	0x01		0x11
SMPTE (30 frames/sec drop frame)	0x20	SMPTE	0x01		0x21
SMPTE (25 frames/sec)	0x30	SMPTE	0x01		0x31
SMPTE (24 frame/sec)	0x40	SMPTE	0x01		0x41
IRIG-B unmodulated	0x50	IRIG-B	0x02		0x52
IRIG-B(1) 1kHz modulated	0x60	IRIG-B	0x02	0x62	
ESE TC76	0x70	ESE	0x03		0x73
ESE TC89	0x80	ESE	0x03		0x83
ESE TC90	0x90	ESE	0x03		0x93

A value of 0 in either nibble indicates that the system has not yet determined the information.

## APPENDIX C - DIAGNOSTICS CODES AND CONTENT

Code: 1  
 Description: Padding/slew calculation

Byte	Description
1	Generator bit capture (low)
2	Generator bit capture (high)
3	Generator bit max (low)
4	Generator bit max (high)
5	Generator bit time in microseconds (low)
6	Generator bit time in microseconds (high)
7	Scheduled pad adjustment (low)
8	Scheduled pad adjustment (high)

Code: 2  
 Description: Stack overflow (no associated data)

Code: 3  
 Description: Checking for time code type

Byte	Description
1	Time code type, one of: SMPTE            0x01 IRIG-B(1)       0x50 IRIG-B           0x60 ESE              0x03
2	Current AD8402 setting
3	Working low AD8402 setting
4	Working high AD8502 setting
5	Scan mode

Code: 4  
 Description: Identified time code details

Byte	Description
1	date available (1=yes, 0=no)
2	time code type
3	<i>Reserved</i>
4	<i>Reserved</i>
5	<i>Reserved</i>

Code: 5  
 Description: System restart

Byte	Description
1	flags (bit 0 = reset reason)
2	<i>reserved</i>
3	<i>reserved</i>
4	<i>reserved</i>

Code: 6  
Description: Generator shutdown

Byte	Description
1	reason: 1 = <i>reserved</i> 2 = loss of time on decoder detected 3 = decoder/generator time/date achieved <> 1 second discrepancy 4 = decoder/generator 1Hz reference achieved <> 250ms discrepancy