# Cohu Protocols

**30 August 2006**

## TABLE OF CONTENTS

**Section** **Page**



[1] $Header: d:/TXB-Cohu/RCS/Cohu.tex,v 1.16 2006-08-30 12:31:56-07 Hamilton Exp Hamilton $

# LIST OF FIGURES

**Figure** **Page**



# LIST OF TABLES

**Table** **Page**

# 1 Introduction to the Cohu Protocols

Cohu has many different systems deployed among their various customers. This results in several different versions of their control protocol. Most of it is similar but there are many differences.

A result of the variations in the various active Cohu protocols, is that it is difficult to determine exactly what commands are valid and will need to be translated. This publication has been extracted from various Cohu protocol sources to put in one place some indication of the breadth of the variations in Cohu's protocol suite. An "annotated" index has been provided to give a quick reference as to which sub-protocol is documented to generate each command.

## 1.1 The Cohu Web Site

Cohu has a helpful web site at `www.cohu-cameras.com`. This web site has much useful data about the Cohu product line and includes a downloadable GUI, instructions on how to use the GUI, information about their protocols and information about their product line.

When the web site is accessed, select the "`Tech Center`" option to get to the most interesting page which has pointers to most of their technical documents. The web site does not require "registration" prior to being allowed to access/download information.

A typical abridged version of the Cohu Web Site is in Appendix A, page A-1.

## 1.2 Basic Types of Cohu Systems

Cohu utilizes several similar protocols to control their line of PTZ units. There are two basic groups of these protocols:

1. The older types of units have fixed speed control in pan/tilt. These include at least the following types:

   1.1. MPC-D Mode,

   1.2. ER2221B,

   1.3. ER2222 and

   1.4. 27XX

   These type names were obtained by reading the choices of camera types in WinMPC rev 4.703.

2. The newer set of units all have variable speed operation in pan/tilt. These include at least the following types:

   2.1. iDome,

   2.2. Dome,

   2.3. Positioner,

   2.4. LCU,

   2.5. iView2,

   2.6. 36XX and

   2.7. 4200

   These type names were obtained by reading the choices of camera types in WinMPC rev 4.703.

[2] `$Header: d:/TXB-Cohu/RCS/Intro.inc,v 1.2 2006-08-25 11:06:21-07 Hamilton Exp Hamilton $`

## 1.3 Differences Between Various Cohu Systems

In addition to having, or not, variable speed operation other changes include:

1. Number of presets, 10 or 64.
2. Presence of a digitally controlled camera type.

## 1.4 Cohu Protocol Documents

The net result of this is that Cohu has many variations of their basic protocol. So far the following protocol documents have been downloaded from the Cohu web site.

1. 3500PRO.PDF A modified version of this document is in Section 7, page 87.
2. 3600PRO.PDF This subversion of the protocol did not seem different enough to warrant extracting its information.
3. 3810PRO.PDF This subversion of the protocol did not seem different enough to warrant extracting its information.
4. 3830PRO.PDF This subversion of the protocol did not seem different enough to warrant extracting its information.
5. 3850PRO.PDF A modified version of this document is in Section 5, page 49.
6. IDOMEIVIEWLCUCOMMPROTOCOL.PDF A modified version of this document is in Section 6, page 61.
7. MPC2221PRO.PDF A modified version of this document is in Section 8, page 107.
8. MPC232PRO.PDF A modified version of this document is in Section 4, page 39.
9. MPC422PRO.PDF A modified version of this document is in Section 3, page 31.

Due to the quantity of differing protocol documents, some customers of Cohu extract a subset of what they expect a Cohu compliant PTZ unit to do and produce their own document. A typical one is this one from Georgia's Department of Transportation: CCTV (REV 8.0) 936 04-18-05 (1).PDF

Cohu provides, on their web site, a GUI (WINMPC) which is configurable to control many of their PTZ systems. On their web site there is an out of date, downloadable manual on how to use it (1032V42.PDF)

## 1.5 Other Cohu Originated Documentation

Other items on the Cohu web site that are useful in determining how a Cohu system ties together and works are: (More is available, but these seem to be the most relevant.)

1. 1027_R0 (IVIEW).PDF
2. 1032V42.PDF
3. 1037.PDF
4. 1050A.PDF

5. 5030.PDF
6. 5035.PDF
7. 829A.PDF
8. 836A.PDF

## 1.6 Unexpected and Undocumented Commands

Using WinMPC has reveled some unexpected features of the Cohu Protocol. This section also discusses some important design decisions and other general information about the TXB-C project. As far as known/decided these are:

1. Commands may be stacked. I.e. it OK to say stop pan and stop tilt in the same command such as:

   <auto-baud> <address> P S T S <checksum>. (F8 01 PSTS 85)

   There have been no observed cases where more than two commands have been stacked in the same command unit.

2. To send an "Iris Close" (which is useful in Spectra menu mode) the camera must report that it is in manual iris mode. Since Pelco does not differentiate between auto and manual mode, the status response says that the camera is always in manual mode.

3. It is unclear what the format of pan/tilt/iris/focus responses should be. In the Georgia DOT procurement document it specifies:

   > Encoding an azimuth position value of decimal $2748_{10}$ or 0xABC, we break it up into three nibbles and add 0x30 to each so:
   >
   > $A_2$ = 0x3A,
   > $A_1$ = 0x3B, and
   > $A_0$ = 0x3C.
   >
   > Azimuth data is sent in this order: $A_2A_1A_0$.

   When this example is used for pan information, WinMPC displays on the screen: "CBA" which wasn't as expected. Reversing the order of the bytes as they are sent ($A_0A_1A_2$) results in a display of "ABC", which is isn't too much better. (274.8 should have shown up in at least one of the attempts.)

   I haven't seen any indication of where Pan $90^o$ is. Is it East or West of $0^o$? There is a short description of how the TXB-C treats this problem here: Section 1.6.1, page 14.

   Additionally:

   3.1. Where is $0^o$ Tilt? Up, Down, horizontal?

   3.2. How about $90^o$ in tilt? Up, Down, etc.

   3.3. Same for $180^o$, $270^o$.

   3.4. Does $360^o$ result in an error?

[3] $Header: d:/TXB-Cohu/RCS/UnDoc.inc,v 1.11 2006-08-29 12:47:37-07 Hamilton Exp Hamilton $

Does anything display on the camera screen? And if so how is it numbered? Does it match the readout? (Pelco's does not match for tilt values.)

4. Several undocumented commands have been generated with WinMPC rev 4.703. A table of these commands is in Table 1, page 12.

5. Preset numbering is some what unusual. For starters, on WinMPC presets are numbered from $1 \longrightarrow n$, but the actual protocol values start at ASCII `0` (which is really `0x30`) (MPC-D mode) or `0x10` (iDome mode).

   5.1. With the following: **iDome** Mode, Dome, Positioner, LCU, iView2, 36XX and 4200 choices, the numbering is as follows: (these values were obtained by running WinMPC rev 4.703.) A table of these values is located here: Table 2, page 13.

      5.1.1. Presets $1 \longrightarrow 47$ it is `0x10` = 1. `0x11` = 2, etc., and `0x3E` = $47_{10}$. (**Note the gap.**)

      5.1.2. Presets $48 \longrightarrow 63$ it is `0x60` = $48_{10}$. `0x61` = $49_{10}$, etc., and `0x70` = $64_{10}$.

   5.2. With the following: **MPC-D** Mode, ER2221B, ER2222 and 27XX choices, the numbering is as follows: (these values were obtained by running WinMPC rev 4.703.)

      5.2.1. Presets $1 \longrightarrow 64$ it is `0x30` ("`0`", Zero) = 1. `0x31` ("`1`", One) = 2, etc., and `0x6F` ("`o`", lower case OH) = $64_{10}$. (**With no "gaps".**)
      *If WinMPC has previously run in iDome mode these values are obtained. However if a "fresh" run is being made, then the highest preset # is 10. This probably is indicative of a bug in WinMPC and is not a capability.*

6. The results of the above preset logic is that there may be confusion if presets are sent in both modes.

   As can be seen from the above the two modes of preset ranges have some conflicts. This is because the MPC-D preset IDs are in the middle of the iDome preset IDs.

   Take '0' (`0x30`), for MPC-D systems this is preset 1 however for iDome systems it is preset 33.

   The overall results of this is that it is not possiable to uniquely identify some preset IDs. The TXB-C's internal logic makes the following assumtions:

   6.1. Communications will be with a MPC-D type of unit that has a total of ten presets, numbered from ASCII `0` through ASCII `9` (`0x30` through `0x39`).

   6.2. If a preset in the iDome range, that does not include the MPC-D range, is detected then the preset mode changes to support only the iDome preset numbering sequence.

   6.3. The only way to return to MPC-D mode is to power cycle the TXB-C.

   Thus what is preset 5 for an MPC-D user will be preset 38 for a iDome user. And there is no way to notify the user of this problem.

7. So far there is nothing uniquely identified as being sent out by WinMPC to indicate which type of unit it is talking to.

8. For putting a label on a preset the following command string is sent:

   The label is blank padded and for camera 1 preset 1 has a default label of:

   "`Preset␣1␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣`" (24 characters total)

   `f8 01 64 50 10 50 72 65 73 65 74 20 31 20 20 20 20 20 20 20 20`

   `20 20 20 20 20 20 20 81`

9. When a command is received and ACKed, WinMPC assumes that the function works. I.e. when commands for "Manual Color", and "Manual Integration", are sent, then WinMPC enables those sections of the GUI. This happens even if a query is sent that indicates those functions are not there.

10. There are some indications that the checksum is not checked by WinMPC.

11. When using WinMPC rev 4.703, there are two modes of operation that may be selected with the "`Camera Setup`" option on the main menu. They are:

    11.1. `PC to MPC Master`: This is the method of communication to a Master Control Unit. In this mode messages are framed with an STX/ETX character pair (`0x02`/`0x03`). Inside the two delimiters the only difference in the command is that there is no "auto baud" character used. In this mode WinMPC ignores the replies and reports a "`Com Error`". The commands are decoded transparently by the TXB-C, however it appears that a problem is occurring because of the internal bug of WinMPC.

    11.2. `PC to MPC Receiver`: This is the method of communication to the dome. In this mode an "auto baud" character (`0xF8`) is sent as the first byte of a message and the last byte of the message is a checksum. In this mode everything seems to work correctly. It is the default mode of operation of WinMPC.

| Command | Description/use |
|---|---|
| `dpE` | In the Set ID Display/Azimuth Elevation menu, selects `Enable` |
| `dpD` | In the Set ID Display/Azimuth Elevation menu, selects `Disable` |
| `dpt` | In the Set ID Display/Azimuth Elevation menu, selects `Permanent` |
| `dpT` | In the Set ID Display/Azimuth Elevation menu, selects `TimeOut` |
| `dNE` | In the Set ID Display/Direct Display menu, selects `Enable` |
| `dND` | In the Set ID Display/Direct Display menu, selects `Disable` |
| `dNt` | In the Set ID Display/Direct Display menu, selects `Permanent` |
| `dNT` | In the Set ID Display/Direct Display menu, selects `TimeOut` |
| `dNI` | In the Set ID Display selects `Grouped with IDs` |
| `dNi` | In the Set ID Display selects `Separated from IDs` |
| `mNs` | In the North Direction selects `Set` |
| `mNc` | In the North Direction selects `Clear` |
| `mN0` | In the North Direction selects `Set 8 Zones` |
| `mN1` | In the North Direction selects `Set 16 Zones` |
| `mNE` | On the main display in the NTCIP Features, selects `Enable NTCIP Mode` |
| `mND` | On the main display in the NTCIP Features, selects `Disable NTCIP Mode`.<br>An NTCIP Protocol message is sent which followed by the Cohu protocol command.<br>54 byte message sent in NTCIP Protocol:<br>`7e 05 13 c1 30 2d 02 01 00 04 06 70 75 62 6c 69 63 a3`<br>`20 02 01 00 02 01 00 02 01 00 30 15 30 13 06 0e 2b 06`<br>`01 04 01 89 36 03 1d 01 06 01 02 0a 02 01 03 d3 ee 7e`<br>Followed by this command sent in Cohu Protocol:<br>`f8 01 6d 4e 44 86` (`mND`) |
| `L?H` | On the main display in Camera, is used to determine what the camera's address is. |

Table 1. Known undocumented commands

| Protocol | Preset # | Protocol | Preset # | Protocol | Preset # | Protocol | Preset # |
|---|---|---|---|---|---|---|---|
| 0x10 | $1_{10}$ | 0x20 | $17_{10}$ | 0x30 | $33_{10}$ | 0x61 | $49_{10}$ |
| 0x11 | $2_{10}$ | 0x21 | $18_{10}$ | 0x31 | $34_{10}$ | 0x62 | $50_{10}$ |
| 0x12 | $3_{10}$ | 0x22 | $19_{10}$ | 0x32 | $35_{10}$ | 0x63 | $51_{10}$ |
| 0x13 | $4_{10}$ | 0x23 | $20_{10}$ | 0x33 | $36_{10}$ | 0x64 | $52_{10}$ |
| 0x14 | $5_{10}$ | 0x24 | $21_{10}$ | 0x34 | $37_{10}$ | 0x65 | $53_{10}$ |
| 0x15 | $6_{10}$ | 0x25 | $22_{10}$ | 0x35 | $38_{10}$ | 0x66 | $54_{10}$ |
| 0x16 | $7_{10}$ | 0x26 | $23_{10}$ | 0x36 | $39_{10}$ | 0x67 | $55_{10}$ |
| 0x17 | $8_{10}$ | 0x27 | $24_{10}$ | 0x37 | $40_{10}$ | 0x68 | $56_{10}$ |
| 0x18 | $9_{10}$ | 0x28 | $25_{10}$ | 0x38 | $41_{10}$ | 0x69 | $57_{10}$ |
| 0x19 | $10_{10}$ | 0x29 | $26_{10}$ | 0x39 | $42_{10}$ | 0x6A | $58_{10}$ |
| 0x1A | $11_{10}$ | 0x2A | $27_{10}$ | 0x3A | $43_{10}$ | 0x6B | $59_{10}$ |
| 0x1B | $12_{10}$ | 0x2B | $28_{10}$ | 0x3B | $44_{10}$ | 0x6C | $60_{10}$ |
| 0x1C | $13_{10}$ | 0x2C | $29_{10}$ | 0x3C | $45_{10}$ | 0x6D | $61_{10}$ |
| 0x1D | $14_{10}$ | 0x2D | $30_{10}$ | 0x3D | $46_{10}$ | 0x6E | $62_{10}$ |
| 0x1E | $15_{10}$ | 0x2E | $31_{10}$ | 0x3E | $47_{10}$ | 0x6F | $63_{10}$ |
| 0x1F | $16_{10}$ | 0x2F | $32_{10}$ | 0x60 | $48_{10}$ | 0x70 | $64_{10}$ |

Table 2. Extended Preset Values

| Command | Data | Response |
|---|---|---|
| Pan/Tilt Position Request | "P?" | ACK + Pan/Tilt Response (Table 14, page 28) |
| Pan/Tilt Goto Command | "p$A_2A_1A_0E_2E_1E_0$" | ACK |
| Zoom/Focus Position Request | "V?" | ACK + Zoom/Focus Response |
| Zoom/Focus Goto Command | "v$Z_2Z_1Z_0F_2F_1F_0$" | ACK (Table 15, page 28) |

Table 3. Position Commands

### 1.6.1 Position Encoding

The message data for the "Pan/Tilt Goto Command" above begins with an ASCII '`p`' followed by the azimuth ($A_2A_1A_0$) and elevation ($E_2E_1E_0$) positions. The positions are 12-bit values encoded four-bits each in the least significant nibbles of the three bytes whose most significant nibbles are always `0x30`. The subscript-2 indicates the byte containing the most significant bytes of the position and subscript-0 indicates the least significant.

#### 1.6.1.1 Position Encoding Example:

(*This information comes from the GA DOT document.*)

Encoding an azimuth position value of decimal $2748_{10}$ or `0xABC`, we break it up into three nibbles and add `0x30` to each so:

$A_2$ = `0x3A`,
$A_1$ = `0x3B`, and
$A_0$ = `0x3C`.

The same encoding scheme is used for the zoom and focus positions in the "Zoom/Focus Goto Command".

### 1.6.2 Pan/Tilt Positions

Pan readouts are based on the $CAL_0$ (North or $0^o$) value of the PTZ. Angular readouts then increment to $90^o$ (East), $180^o$ (South) and to $270^o$ (West) when looking down on the PTZ unit from above.

Tilt readouts are based on the horizon, with $0^o$ meaning that the PTZ is looking at the horizon. As the PTZ points down the protocol readout changes to $45^o$ and then $90^o$ when the PTZ is pointing straight down. The PTZ on-screen display will indicate - $45^o$ and then -$90^o$. The discrepancy in the two types of readouts is that the protocol value is unsigned and the on-screen is signed. As tilt passes through straight down (not physically possible, but conceptually convenient, the protocol values increment until a straight up readout is $270^o$ and the on-screen display shows $90^o$. At $45^o$ above the horizon the protocol readout will be $315^o$ and the on-screen display will be $45^o$. Here is a short table (Table 4, page 14) and a figure (Figure 1, page 15) to attempt and explain these values.

| Position | D readouts | Displays as |
|---|---|---|
| $90^o$ up | 270 | 90 |
| $45^o$ up | 315 | 45 |
| Horizontal - $1^o$ | 359 | 1 |
| Horizontal | 0 | 0 |
| $45^o$ down | 45 | -45 |
| $90^o$ down | 90 | -90 |

Table 4. Protocol readouts *vs.* On-screen tilt readouts

**Note**

1. **Position:** Pointing direction of the enclosure/camera.
2. **D readouts:** D protocol returned value for this angle.
3. **Displays as:** What is displayed on the Spectra screen.

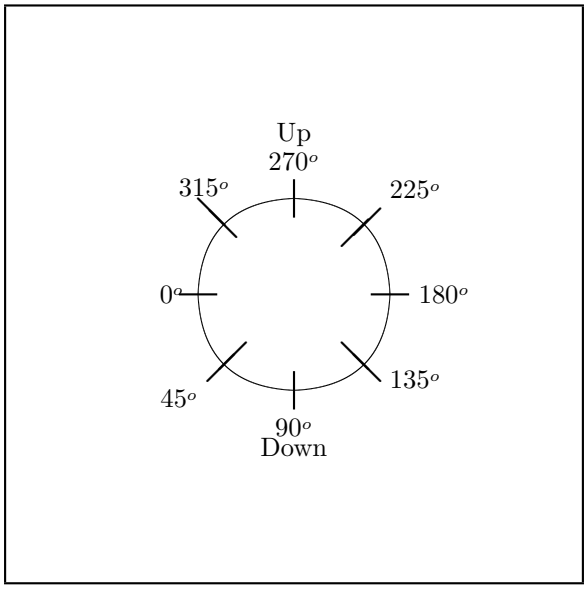



Figure 1. D Protocol Tilt Readout Positions

## 1.7 Notes on using WinMPC

WinMPC is a Windows based GUI (Glass Keyboard) from Cohu. The version used during development of the TXB-Cohu was rev 4.703 and was obtained from the Cohu web site. It consists of a single executable (.EXE) module which self installs on a PC. It has been used on Windows 98 SE and Windows XP and gives about the same level of performance.

Cohu provides a manual, which is slightly out of date, (1032V42.PDF) on how to use it.

Through use of the WinMPC program, several unexpected problems have been discovered. Some of these are:

1. Whenever "`Camera Setup`" is accessed from the main menu, the program sends a "`reset camera`" (`RS`) command to the camera.
2. Often times the program seems to get "locked up" sending "queries" to the camera. Usually by "clicking" on the "`status`" button on the main display it will stop doing that, in some cases the "`status`" button must be clicked twice. In a few cases clicking does no good at all.
3. Near the bottom of the main menu, there is a display of what the GUI is sending out (`Tx`) and receiving back in (`Rx`). The format is very convenient, but unexpected.

   A typical transmit command would display as: `F8␣01␣cDS␣85` [4]

   - `F8`: This is the "auto baud" character that is the first byte of all commands except for single byte ACKs. It is displayed in hex.
   - `01`: This is the camera address that this command is being sent to. It is displayed in hex.
   - `cDS`: This is the actual command being sent to the camera. It is displayed in ASCII. In this example if it was being displayed in hex, the display would be `63 44 53`.
   - `85`: This is the checksum. It is displayed in hex.

   An ACK is displayed as "`Ack`".
4. When first started, the GUI, send nothing out to the camera until the program user tells it to.
5. When the GUI "locks up" sending a command, the only way out is to get out of the program and restart.
6. The GUI does "remember" the last camera, etc., set up data that was selected.
7. The "log" function does work, however it stops logging the first time that an error is detected. Errors are indicated on the screen by having the red "LED" marked `Comm Error` turn on.
8. The only way of getting the `Comm Error` LED to turn off, is to receive a good message or an ACK from the camera. (Or restarting the GUI.)
9. `Lens Function Control` items do not become activated until a reply is received from the camera that indicates that function is present. The only exception for this is that `Zoom` is always turned on.

[4] The response from the TXB-C, as of 25AUG06, is
`F8 01 63 57 30 53 30 59 30 4D 31 44 30 4C 37 41 30 58 30 48 49 50 30 76 30 70 30 46 6F 73 6F 86` or
`<auto baud> <Camera Address> c W 0 S 0 Y 0 M 1 D 0 L 7 A 0 X 0 H I P 0 v 0 p 0 F o s o <CKSM>`.

10. Since different cameras have different capabilities, options in `Latch Commands` are sometimes enabled and sometimes "grayed out". If a specific latch function is desired, fool around with the choices in `Camera Setup` to find a camera type that uses that latch function.

    The TXB-C always returns a status command to the `L?` command (`L7A0`) that says:

    - The Iris is in Manual mode,
    - The Lens is in Fast mode,
    - That the Camera is Powered on and
    - That all three Auxes are turned off.

11. After looking a long time, there is no specific command string that is sent from the GUI to indicate which type of camera it thinks that it is talking to. The most that happens is that if an iDome type is being addressed, then variable speed commands are sent and if a MPC type dome is being accessed then the commands are fixed speed types.

12. Sometimes the GUI will change the camera address from whatever it used to be to `0xFF`. The way to get out of this is to click the `Reset` and then the `Find` buttons and let the GUI reacquire the correct address.

This page intentionally left blank

# 2 GA DOT, CCTV Standard Protocol

## 2.1 General

Use[6] the NaviGAtor Standard CCTV Control Protocol (hereinafter called the "CCTV standard protocol" or "standard protocol") for CCTV system control communications between the user control interface unit and the NaviGAtor central system. Implement the standard protocol as specified in this subsection for all CCTV equipment installed on this project. The CCTV standard protocol governs all control communications between the NaviGAtor central system (hereinafter called the "host") and the CCTV system.

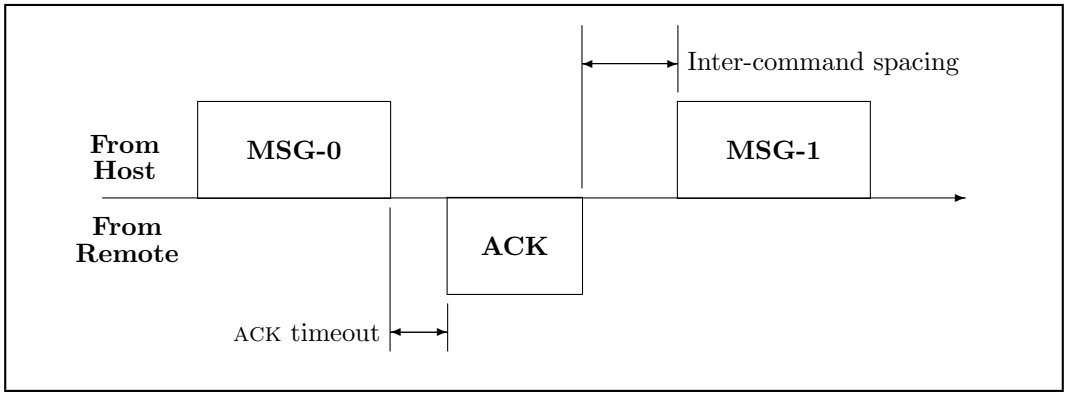
## 2.2 Interface

The remote shall communicate using the CCTV standard protocol over an RS-232 serial interface. Data is transmitted using 1 start bit, 8 data bits, and 1 stop bit (no parity) at a baud rate of 9600.

Each remote shall be identified by a unique integer address between 1 and $233_{10}$ (*This is probably an error, it should probably read* $223_{10}$.). This address is set during installation and shall not be altered using the standard protocol. The address is included in messages to identify the intended recipient of commands from the host and responses from the remote.

## 2.3 Message Sequence

All communication between the host and the remote is initiated by the host. The remote shall not transmit except in response to a properly formatted message from the host addressed to the remote's assigned address. The remote shall respond to such messages with an ACK (a single ASCII character `0x06`) within 500 milliseconds ("ACK timeout"). The remote shall be ready to receive and process another command in as little as 150 milliseconds after the ACK is transmitted ("Inter-command spacing"). Figure 2, page 19 depicts this sequence.



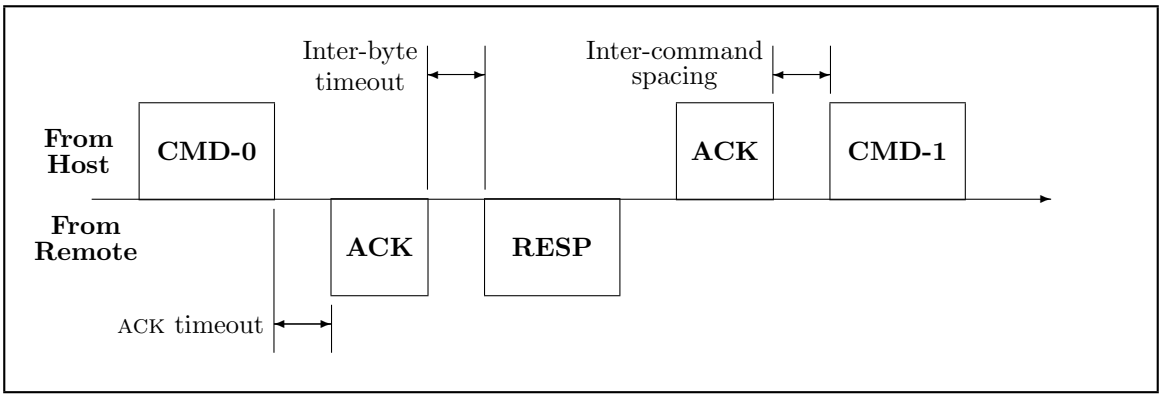

$RCSfile: BMS.inc,v $

Figure 2. Basic Message Sequence

[5] `$Header: d:/TXB-Cohu/RCS/GaCohu.inc,v 1.14 2006-08-29 10:15:17-07 Hamilton Exp Hamilton $`

[6] This document was derived from the Georgia Department of Transportation solicition: "CCTV (Rev 8.0) 936 04-18-05 (1).pdf"

Some messages from the host require a subsequent response message from the remote. In this case, the remote's ACK shall be followed immediately by a response message. When the host receives the properly formatted response, it will transmit an ACK back to the remote[7]. This sequence is depicted in Figure 3, page 20. The timing constraints presented above apply. Additionally, the host will wait a maximum of 500 milliseconds for each byte in the response message ("Inter-byte timeout").




$RCSfile: RMS.inc,v $

Figure 3. Response Message Sequence

## 2.4 Message Format

Command and response messages shall be formatted as shown in Table 5, page 20.

| Size | Data | Description |
|---|---|---|
| 1 byte | `0xF8` | Message header |
| 1 byte | Address | Controller address (`0x01` to `0xDF`) |
| 2+ bytes | Message | Data (see Command and Response Message sections (Section 2.5, page 21 and Section 2.7, page 28) |
| 1 byte | Checksum | `0x80` to `0x8F` (Section 2.4.1, page 20) |

Table 5. Message Format

Messages start with a single-byte header followed by a single-byte address. Two or more message data bytes follow the address. The message is terminated with a single-byte checksum.

### 2.4.1 Checksum Calculations

The message checksum byte is defined as the least significant nibble of the exclusive-or of all previous bytes in the message (excluding the `0xF8` header) added to `0x80`. The resulting byte will be between `0x80` and `0x8F` inclusive.

[7] *There is no requirement to resend a message that does not get an* ACK*, but these* ACK*'s must be ignored.*

#### 2.4.1.1 Checksum Example

- The bytes of a "Pan Left" command addressed to controller 1 would be:

  `0xF8 0x01 0x50 0x4C 0x??`

- The bitwise exclusive-or of the bytes between the `0xF8` header and the checksum comes out to 0x1D:

  `0x01 ^ 0x50 ^ 0x4C = 0x1D`

- Adding the least significant nibble to `0x80`, we get `0x8D` for the checksum byte:

  `0x80 | (0x0F & 0x1D) = 0x8D`

- The resulting 5-byte command would be:

  `0xF8 0x01 0x50 0x4C 0x8D`

## 2.5 Command Messages

The tables included present the message data contained within command messages sent by the host to the remote. In most cases (exceptions will be noted), the "Data" column contains the ASCII characters of the message data to be sent. For example, the Pan Left command data is "`PL`", an ASCII '`P`' (`0x50`) followed by an ASCII '`L`' (`0x4C`).

An "ACK"[8] in the "Response" column indicates no response message is expected from the remote beyond the ACK itself as shown above in Figure 2, page 19. An "ACK + * Response" indicates the ACK from the remote is to be followed by the specified response message as shown above in Figure 3, page 20.

| Command | Length | Data | Response |
|---|---|---|---|
| Pan Left | 5 | "PL" | ACK |
| Pan Right | 5 | "PR" | ACK |
| Pan Stop | 5 | "PS" | ACK |
| Tilt Up | 5 | "TU" | ACK |
| Tilt Down | 5 | "TD" | ACK |
| Tilt Stop | 5 | "TS" | ACK |
| Zoom In | 5 | "ZI" | ACK |
| Zoom Out | 5 | "ZO" | ACK |
| Zoom Stop | 5 | "ZS" | ACK |
| Focus Near | 5 | "FN" | ACK |
| Focus Far | 5 | "FF" | ACK |
| Focus Stop | 5 | "FS" | ACK |
| Iris Open | 5 | "IO" | ACK |
| Iris Close | 5 | "IC" | ACK |
| Iris Stop | 5 | "IS" | ACK |

Table 6. Momentary Commands

[8] *This* ACK *(`0x06`) is a standard ASCII control byte. There is also another ASCII control byte of* NAK *(`0x15`) which is not to be used. It is defined in other Cohu Documents.*

| Command | Length | Data | Response |
|---|---|---|---|
| Manual Iris Toggle | 5 | "LM" | ACK |
| Camera Power Toggle | 5 | "LP" | ACK |
| Lens Speed Toggle | 5 | "LL" | ACK |
| Latch Status Request | 5 | "L?" | ACK + Latch Response (Table 16, page 29) |

Table 7. Latch Commands

| Command | Length | Data | Response |
|---|---|---|---|
| Goto Preset 0 ⟶ 9 | 5 | "H0" ⟶ "H9" | ACK (H0, H1, H2, H3, H4, H5, H6, H7, H8, H9) |
| Store Preset 0 ⟶ 9 | 5 | "P0" ⟶ "P9" | ACK (P0, P1, P2, P3, P4, P5, P6, P7, P8, P9) |
| Preset Status Request | 5 | "H?" | ACK + Preset Response (Table 17, page 29) |

Table 8. Preset Commands

## 2.6 Queries and their Responses

All of this data is repeated where it logically appears (according to the original document).

### 2.6.1 Latch Query and its Response

| Command | Length | Data | Response |
|---|---|---|---|
| Latch Status Request | 5 | "L?" | ACK + Latch Response (Table 16, page 29) |
| Latch Response | 7 | "L$D_1$A$D_0$" | |
| | 1 | "L" | Data ID |
| | 1 | $D_1$ = 0x30 to 0x38 | The bits of $D_1$, the second byte in the data, indicate: (L0, L1, L2, L3, L4, L5, L6, L7, L8)<br>0: Iris Auto(0) or Manual(1)<br>1: Camera Power Off(0) or On(1)<br>2: Lens Speed Slow(0) or Fast(1)<br>3: Comm. Error No(0) or Yes(1)<br>4: (always 1)<br>5: (always 1)<br>6: (always 0)<br>7: (always 0) |
| | 1 | "A" | Data ID |
| | 1 | $D_0$ = 0x30 to 0x37 | The bits of $D_0$, the fourth byte in the data, indicate: (A0, A1, A2, A3, A4, A5, A6, A7)<br>0: Aux-1 Off(0) or On(1)<br>1: Aux-2 Off(0) or On(1)<br>2: Aux-3 Off(0) or On(1)<br>3: (always 0)<br>4: (always 1)<br>5: (always 1)<br>6: (always 0)<br>7: (always 0) |

Table 9. Command "Latch Query" and its Response

### 2.6.2 Preset Status Query and its Response

| Command | Length | Data | Response |
|---|---|---|---|
| Preset Status Request | 5 | "H?" | ACK + Preset Response (Table 17, page 29) |
| Preset Response | 5 | "H$D_0$" | $D_0$ is a single ASCII character indicating: |
| | 1 | "H" | Data ID |
| | | | '0' $\longrightarrow$ '9': at the indicated preset<br>H0, H1, H2, H3, H4, H5, H6, H7, H8, H9)<br>'A': (HA) Active, going to a preset<br>'I': (HI) Inactive, not at a preset<br>'E': (HE) Error, unable to get to preset |

Table 10. Command "Preset Status Query" and its Response

### 2.6.3 Pan/Tilt Query and its Response

| Command | Length | Data | Response |
|---|---|---|---|
| Pan/Tilt Position Request | 5 | "P?" | ACK + Pan/Tilt Response (Table 14, page 28) |
| Pan/Tilt Position Response | 10 | "P$A_2A_1A_0E_2E_1E_0$" | Using the same data encoding as Pan/Tilt Goto Command described earlier.<br>Note: (see Position Encoding above) while the command uses a lower-case 'p',<br>the response uses an upper-case 'P') |
| | 1 | "P" | Data ID |
| | 3 | $A_2A_1A_0$ | Azimuth data |
| | 3 | $E_2E_1E_0$ | Elevation data |

Table 11. Command "Pan/Tilt Query" and its Response

### 2.6.4 Zoom/Focus Query and its Response

| Command | Length | Data | Response |
|---|---|---|---|
| Zoom/Focus Position Request | 5 | "`V?`" | ACK + Zoom/Focus Response |
| Zoom/Focus Position Response | 10 | "`V`$Z_2Z_1Z_0F_2F_1F_0$" | Using the same data encoding as Zoom/Focus Goto Command described (see Position Encoding above) earlier. Note: while the command uses a lower-case '`v`', the response uses an upper-case '`V`') |
| | 1 | "`V`" | Data ID |
| | 3 | $Z_2Z_1Z_0$ | Zoom position |
| | 3 | $F_2F_1F_0$ | Focus position |

Table 12. Command "Zoom/Focus Query" and its Response

### 2.6.5 Position Encoding

The message data for the "Pan/Tilt Goto Command" above begins with an ASCII '`p`' followed by the azimuth ($A_2A_1A_0$) and elevation ($E_2E_1E_0$) positions. The positions are 12-bit values encoded four-bits each in the least significant nibbles of the three bytes whose most significant nibbles are always `0x30`. The subscript-2 indicates the byte containing the most significant bytes of the position and subscript-0 indicates the least significant.

#### 2.6.5.1 Position Encoding Example:

Encoding an azimuth position value of decimal $2748_{10}$ or `0xABC`, we break it up into three nibbles and add `0x30` to each so:

$A_2$ = `0x3A`,
$A_1$ = `0x3B`, and
$A_0$ = `0x3C`.

The same encoding scheme is used for the zoom and focus positions in the "Zoom/Focus Goto Command".

| Command | Data | Response |
|---|---|---|
| Pan/Tilt Position Request | "`P?`" | ACK + Pan/Tilt Response (Table 14, page 28) |
| Pan/Tilt Goto Command | "`p`$A_2A_1A_0E_2E_1E_0$" | ACK |
| Zoom/Focus Position Request | "`V?`" | ACK + Zoom/Focus Response |
| Zoom/Focus Goto Command | "`v`$Z_2Z_1Z_0F_2F_1F_0$" | ACK (Table 15, page 28) |

Table 13. Position Commands

### 2.6.6 Pan/Tilt Positions

Pan readouts are based on the $CAL_0$ (North or $0^o$) value of the PTZ. Angular readouts then increment to $90^o$ (East), $180^o$ (South) and to $270^o$ (West) when looking down on the PTZ unit from above.

Tilt readouts are based on the horizon, with $0^o$ meaning that the PTZ is looking at the horizon. As the PTZ points down the protocol readout changes to $45^o$ and then $90^o$ when the PTZ is pointing straight down. The PTZ on-screen display will indicate - $45^o$ and then -$90^o$. The discrepancy in the two types of readouts is that the protocol value is unsigned and the on-screen is signed. As tilt passes through straight down (not physically possible, but conceptually convenient, the protocol values increment until a straight up readout is $270^o$ and the on-screen display shows $90^o$. At $45^o$ above the horizon the protocol readout will be $315^o$ and the on-screen display will be $45^o$. Here is a short table to attempt and explain these values.

| Position | D readouts | Displays as |
|---|---|---|
| $90^o$ up | 270 | 90 |
| $45^o$ up | 315 | 45 |
| Horizontal - $1^o$ | 359 | 1 |
| Horizontal | 0 | 0 |
| $45^o$ down | 45 | -45 |
| $90^o$ down | 90 | -90 |

**Note**

1. **Position:** Pointing direction of the enclosure/camera.
2. **D readouts:** D protocol returned value for this angle.
3. **Displays as:** What is displayed on the Spectra screen.

## 2.7 Response Messages

In Table 14, page 28 through Table 17, page 29, presents the message data contained within response messages sent by the remote to the host.

| Message | Data | Comment |
|---|---|---|
| Pan/Tilt Position Response | "P$A_2A_1A_0E_2E_1E_0$" | Using the same data encoding as Pan/Tilt Goto Command described earlier.<br>Note: (see Position Encoding above) while the command uses a lower-case 'p',<br>the response uses an upper-case 'P') |

Table 14. Pan/Tilt Response Messages

| Message | Data | Comment |
|---|---|---|
| Zoom/Focus Position Response | "V$Z_2Z_1Z_0F_2F_1F_0$" | Using the same data encoding as Zoom/Focus Goto Command described<br>Note: see Position Encoding above) earlier. (note: while the command uses a lower-case 'v',<br>the response uses an upper-case 'V') |

Table 15. Zoom/Focus Response Messages

While the remote is going to a position as a result of a "Pan/Tilt Goto", "Zoom/Focus Goto", or "Preset Goto" command, "Preset Responses" will return 'A' in the $D_0$ byte (HA). If the remote is unable to get to the position in the "Goto command" (i.e. the pan/tilt assembly reaches the mechanical stops), the "Latch Response" will return 'E' in the $D_0$ byte (HE).

| Message | Data | Comment |
|---|---|---|
| Latch Response | "L$D_1$A$D_0$" | |
| | $D_1$ = 0x30 to 0x38 | The bits of $D_1$, the second byte in the data, indicate: (L0, L1, L2, L3, L4, L5, L6, L7, L8)<br>0: Iris Auto(0) or Manual(1)<br>1: Camera Power Off(0) or On(1)<br>2: Lens Speed Slow(0) or Fast(1)<br>3: Comm. Error No(0) or Yes(1)<br>4: (always 1)<br>5: (always 1)<br>6: (always 0)<br>7: (always 0) |
| | $D_0$ = 0x30 to 0x37 | The bits of $D_0$, the fourth byte in the data, indicate: (A0, A1, A2, A3, A4, A5, A6, A7)<br>0: Aux-1 Off(0) or On(1)<br>1: Aux-2 Off(0) or On(1)<br>2: Aux-3 Off(0) or On(1)<br>3: (always 0)<br>4: (always 1)<br>5: (always 1)<br>6: (always 0)<br>7: (always 0) |

Table 16. Latch Response Messages

| Message | Data | Comment |
|---|---|---|
| Preset Response | "H$D_0$" | $D_0$ is a single ASCII character indicating:<br>'0' $\longrightarrow$ '9':at the indicated preset<br>(H0, H1, H2, H3, H4, H5, H6, H7, H8, H9)<br>'A': (HA) Active, going to a preset<br>'I': (HI) Inactive, not at a preset<br>'E': (HE) Error, unable to get to preset |

Table 17. Preset Response Messages

This page intentionally left blank

# 3 MPC System RS-422 Interface

## 3.1 General Description

The **MPC**[10] Master Control Panel contains an 8031 microcomputer with on-board UART for RS-422 communications with MPC Remote Control Panels and MPC Control Receivers and RS-232 for communications with a Host Computer. (The RS-232 interface is described in publication **6X-5030**.)

This section supplies the details necessary to understand RS-422 communications between the MPC Master and Control Receivers as used in the camera control system (Figure 4, page 32).

## 3.2 Message Format

Any time a command message is sent from the MPC Master Control Panel to a Control Receiver, the Receiver will respond with an ACK or an NAK (Figure 4, page 32). Data is transmitted using 1 start bit, 8 data bits, and 1 stop bit (no parity). The baud rate is set using the monitor 97 function as an entry code. It is typically set to 9600 baud. This process is described in the installation instructions of the Control Panel manual.

The commands are sent using the command message format show in Section 3.6.1, page 32. The various commands that may be transmitted are listed in Section 3.6.2, page 33.

1. If the command message contains a latch function command, then the latch status response (Section 3.6.3, page 34, Format of Latch Status Response) will be sent back to the Master Control Panel.

2. If the command message contains a home function command, then the home status response (Section 3.6.3, page 34, Format of Home Status Response from MPC) will be sent back to the Master Control Panel.

3. If the command is a position message, the response will be as shown in Section 3.6.4, page 35.

4. If the command message contains only a momentary function, no further response beyond the ACK or NAK will occur.

5. If the command message contains a communications error (e.g., an address to an non-existent control receiver or a failure in the communications system), then the command error message response (in Section 3.6.3, page 34, with B3 of byte 3 equal to 1) will be sent back.

6. Section 3.6.5, page 36 gives the ID message format for communications from the master control panel to a control receiver. An ACK or NAK is returned.

## 3.3 Maintenance

No maintenance adjustments are required for the RS-422 circuits on the processor board.

## 3.4 Parts List

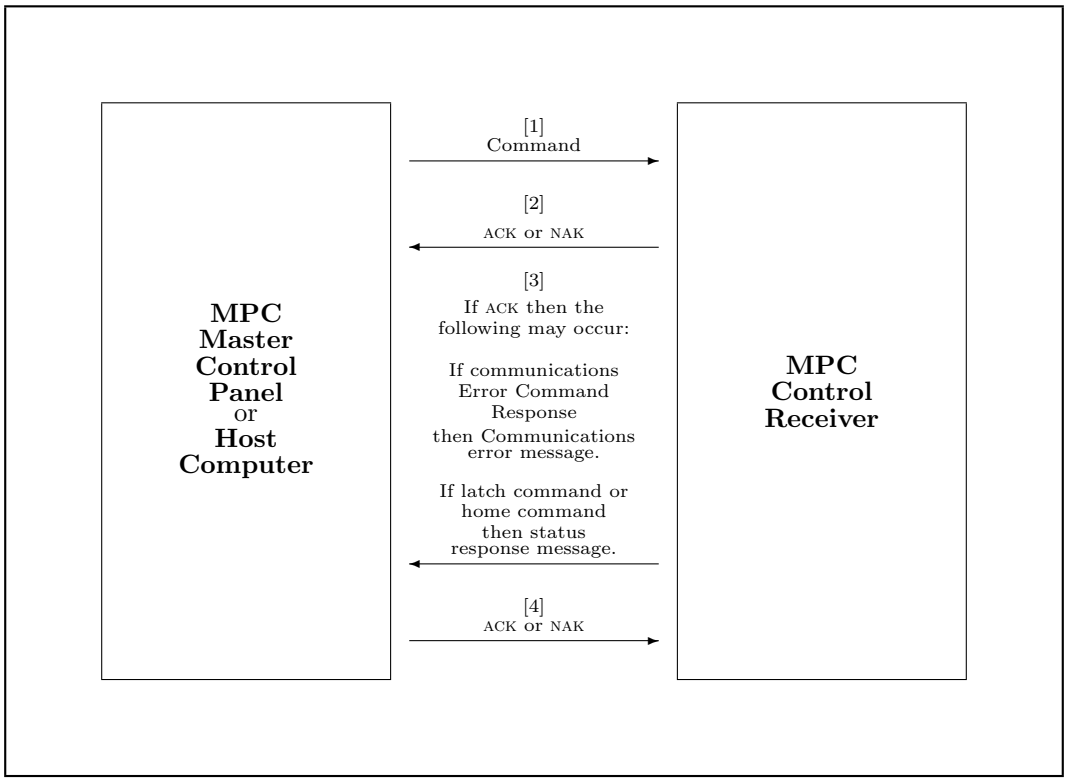
Refer to the processor board for parts identifications.

[9] `$Header: d:/TXB-Cohu/RCS/422Pro.inc,v 1.13 2006-08-24 14:43:13-07 Hamilton Exp Hamilton $`

[10] This document was derived from Cohu's document: "mpc422pro.pdf"

## 3.5 Schematic Diagrams

Refer to the processor board for the schematic diagram.




$RCSfile: TCP.inc,v $

Figure 4. Typical Communications Protocol (Similar to the one in Cohu Document 6X-5035

## 3.6 Commands and Responses

### 3.6.1 RS-422 Command Message Format

| Byte | Data | Description |
|---|---|---|
| 0 | `0xF8` | Autobaud character |
| 1 | Address in hex | Camera address (`0x01` to `0xDF`) |
| 2 to $2n+1$ | Command Data | See Section 3.6.2, page 33 |
| $2n+2$ | CHECKSUM | `0x80` to `0x8F` Least significant nibble is XOR of all previous bytes (LS nibble only) except `0xF8` |

**Note**

$n$ is the number of commands in the message.

### 3.6.2 RS-422 Command Data

| **Momentary** | | |
|---|---|---|
| Reset Control Receiver | "rs" | |
| Pan Left | "PL" | |
| Pan Right | "PR" | |
| Pan Stop | "PS" | |
| Tilt Up | "TU" | |
| Tilt Down | "TD" | |
| Tilt Stop | "TS" | |
| Zoom In | "ZI" | |
| Zoom Out | "ZO" | |
| Zoom Stop | "ZS" | |
| Focus Near | "FN" | |
| Focus Far | "FF" | |
| Focus Stop | "FS" | |
| Iris Open | "IO" | |
| Iris Close | "IC" | |
| Iris Stop | "IS" | |
| P/T Position Request | "P?" | |
| P/T Goto Command | "p$A_2A_1A_0E_2E_1E_0$" | See Section 3.6.4, page 35 for position format |
| Lens Position Request | "V?" | |
| Lens Goto Command | "v$A_2A_1A_0E_2E_1E_0$" | |
| **Latch** | | |
| Manual Iris Toggle | "LM" | |
| Camera Power Toggle | "LP" | See Section 3.6.3, page 34 for response format |
| Lens Speed Toggle | "LL" | |
| Latch Status Request | "L?" | |
| Aux Functions (Option 1) | "L1" $\longrightarrow$ "L3" | Toggles Aux 1, Aux 2, or Aux 3 |
| Color Balance (Option 2) | "L1" | Select auto/Manual Mode |
| Increase Blue | "B1" | |
| Increase Red | "B2" | |
| Balance Stop | "B0" | |
| **Home** | | |
| Home Position Goto Preset 0 $\longrightarrow$ 9 | "H0" $\longrightarrow$ "H9" | A home active HA will be immediate response. Later a home position H0 to H9 or home error HE will be generated. (H0, H1, H2, H3, H4, H5, H6, H7, H8, H9) |

*Continued on the next page.*

| *Continued from the previous page.* | | |
|---|---|---|
| Home Position Store Program 0 ⟶ 9 | "P0" ⟶ "P9" | (P0, P1, P2, P3, P4, P5, P6, P7, P8, P9) |
| Home Position Status Request | "H?" | Will generate a home status response See Section 3.6.3, page 34 |

**Note**

Commands consist of the ASCII codes for the two letters shown:

- "rs" = ASCII codes for "r" and "s"
- "PL" = ASCII codes for "P" and "L", etc.

### 3.6.3 RS-422 Response Message

#### 3.6.3.1 Format of Latch Status Response from MPC Control Receiver

| Byte | Data | Description |
|---|---|---|
| 0 | 0xF8 | Autobaud character |
| 1 | Address in Hex | Camera address (0x01 to 0xDF) |
| 2 | L | Latch Status |
| 3 | 0x30 to 0x37<br>or<br>0x38 | LS nibble is four bits of status<br><br>(Comm error)<br>Bit Value 0/1<br>$B_0$ = Iris Auto/Manual<br>$B_1$ = Camera power Off/On<br>$B_2$ = Lens Speed Slow/Fast<br>$B_3$ = Communications Error No/Yes<br>(L0, L1, L2, L3, L4, L5, L6, L7, L8) |
| 4 | A | Aux Status |
| 5 | 0x30 to 0x37 | LS nibble is four bits of status<br>Bit Value 0/1<br>$B_0$ = Aux 1<br>$B_1$ = Aux 2<br>$B_2$ = Aux 3<br>(A0, A1, A2, A3, A4, A5, A6, A7) |
| 6 | Checksum | 0x80 to 0x8F Least significant nibble is XOR of all previous bytes (LS nibble only) except 0xF8 |

#### 3.6.3.2 Format of Home Status Response from MPC Control Receiver

| Byte | Data | Description |
|---|---|---|
| 0 | 0xF8 | Autobaud character |
| 1 | Address in Hex | Camera address (0x01 to 0xDF) |
| 2 | H | Home Status |
| 3 | 0 ⟶ 9 | ie, 2 = Home position 2<br>(H0, H1, H2, H3, H4, H5, H6, H7, H8, H9) |
| | A<br>I<br>E | (HA) Home active<br>(HI) Not-at-Home position, or active<br>(HE) Home error — could not get to home position |
| 4 | Checksum | 0x80 to 0x8F Least significant nibble is XOR of all previous bytes (LS nibble only) except 0xF8 |

### 3.6.4 MPC Position Message

#### 3.6.4.1 Format of Pan/Tilt Position Message

| Byte | Data | Description |
|---|---|---|
| 0 | 0xF8 | Autobaud character |
| 1 | Address in Hex | Camera address (0x01 to 0xDF) |
| 2 | P<br>p | Upper case P is response<br>Lower case p is Goto command |
| 3<br>4<br>5 | $A_2$<br>$A_1$<br>$A_0$ | (MS nibble)<br>Azimuth position (See Notes)<br>(LS nibble) |
| 6<br>7<br>8 | $E_2$<br>$E_1$<br>$E_0$ | (MS nibble)<br>Elevation position (See Notes)<br>(LS nibble) |
| 9 | Checksum | 0x80 to 0x8F Least significant nibble is XOR of all previous bytes (LS nibble only) except 0xF8 |

#### 3.6.4.2 Format of Zoom/Focus Position Message

| Byte | Data | Description |
|---|---|---|
| 0 | 0xF8 | Autobaud character |
| 1 | Address in Hex | Camera address (0x01 to 0xDF) |
| 2 | V<br>v | Upper case V is response<br>Lower case v is Goto command |
| 3<br>4<br>5 | $A_2$<br>$A_1$<br>$A_0$ | (MS nibble)<br>Zoom position (See Notes)<br>(LS nibble) |
| 6<br>7<br>8 | $E_2$<br>$E_1$<br>$E_0$ | (MS nibble)<br>Focus position (See Notes)<br>(LS nibble) |
| 9 | Checksum | 0x80 to 0x8F Least significant nibble is XOR of all previous bytes (LS nibble only) except 0xF8 |

**Note**

The 12-bit position data is contained in nibbles $0 \rightarrow 2$, with nibble $0 = b_0 \rightarrow b_3$, nibble $1 = b_4 \rightarrow b_7$, and nibble $2 = b_8 \rightarrow b_{11}$. The nibble is then added to `0x30` to produce a value from `0x30` to `0x3F`.

Example: Azimuth data $b_0 \rightarrow b_3$ is expressed as $A_0$ (azimuth nibble 0)

### 3.6.5 ID Message Format

#### 3.6.5.1 Format of Select Mode Message

| Byte | Data | Description |
|---|---|---|
| 0 | `0xF8` | Autobaud character |
| 1 | Address in Hex | Camera address (`0x01` to `0xDF`) |
| 2 | `d` | Display message |
| 3 | `M` | (`dM`) Mode select |
| 4 | `M`<br>`I` | (`dMM`) Menu mode<br>(`dMI`) ID mode |
| 5 | Checksum | `0x80` to `0x8F` Least significant nibble is XOR of all previous bytes (LS nibble only) except `0xF8` |

#### 3.6.5.2 Format of ID Enable/Disable Message

| Byte | Data | Description |
|---|---|---|
| 0 | `0xF8` | Autobaud character |
| 1 | Address in Hex | Camera address (`0x01` to `0xDF`) |
| 2 | `d` | Display message |
| 3 | `I` | (`dI`) ID display |
| 4 | `D`<br>`E` | (`dID`) Disable ID display<br>(`dIE`) Enable ID display |
| 5 | Checksum | `0x80` to `0x8F` Least significant nibble is XOR of all previous bytes (LS nibble only) except `0xF8` |

#### 3.6.5.3 Format of ID Clear Screen Message

| Byte | Data | Description |
|---|---|---|
| 0 | `0xF8` | Autobaud character |
| 1 | Address in Hex | Camera address (`0x01` to `0xDF`) |
| 2 | `d` | Display message |
| 3 | `C` | (`dC`) Clear screen |
| 4 | Checksum | `0x80` to `0x8F` Least significant nibble is XOR of all previous bytes (LS nibble only) except `0xF8` |

#### 3.6.5.4 Format of Select ID Top Message

| Byte | Data | Description |
|---|---|---|
| 0 | 0xF8 | Autobaud character |
| 1 | Address in Hex | Camera address (0x01 to 0xDF) |
| 2 | d | Display message |
| 3 | I | (dI) ID display |
| 4 | T | (dIT) ID at top |
| 5 | Checksum | 0x80 to 0x8F Least significant nibble is XOR of all previous bytes (LS nibble only) except 0xF8 |

#### 3.6.5.5 Format of Select ID Bottom Message

| Byte | Data | Description |
|---|---|---|
| 0 | 0xF8 | Autobaud character |
| 1 | Address in Hex | Camera address (0x01 to 0xDF) |
| 2 | d | Display message |
| 3 | I | (dI) ID display |
| 4 | B | (dIB) ID at bottom |
| 5 | Checksum | 0x80 to 0x8F Least significant nibble is XOR of all previous bytes (LS nibble only) except 0xF8 |

#### 3.6.5.6 Format of Update Line Message

| Byte | Data | Description |
|---|---|---|
| 0 | 0xF8 | Autobaud character |
| 1 | Address in Hex | Camera address (0x01 to 0xDF) |
| 2 | d | Display message |
| 3 | L | (dL) Line of text |
| 4 | 1 $\longrightarrow$ < | (0x31 $\longrightarrow$ 0x3C) Line number (1 $\longrightarrow$ 12) + 0x30 (dL1, dL2, dL3, dL4, dL5, dL6, dL7, dL8, dL9, dL:, dL;, dL<) |
| 5 | (ID text) | ASCII characters (up to 24) for the line starting with the left most character |
| N | Checksum | 0x80 to 0x8F Least significant nibble is XOR of all previous bytes (LS nibble only) except 0xF8 |

#### 3.6.5.7 Format of Alarm Display Message

| Byte | Data | Description |
|---|---|---|
| 0 | 0xF8 | Autobaud character |
| 1 | Address in Hex | Camera address (0x01 to 0xDF) |
| 2 | d | Display message |
| 3 | A | (dA) Alarm message |
| 4 | 1 ⟶ 3 | (0x31 ⟶ 0x33) Display line number (1 ⟶ 3) + 0x30 (dA1, dA2, dA3) |
| 5 | B | Blink alarm message (dA1B, dA2B, dA3B) |
| | E | Enable alarm message (dA1E, dA2E, dA3E) |
| | D | Disable alarm message (dA1D, dA2D, dA3D) |
| 6 | Checksum | 0x80 to 0x8F Least significant nibble is XOR of all previous bytes (LS nibble only) except 0xF8 |

**Note**

For all messages from master control panel to control receiver, ACK or NAK is returned[11].

[11] ACK (0x06) is a standard ASCII control byte as is NAK (0x15).

# 4 MPC System RS-232 Interface

Document: 6X-5030
January 10, 1994
Rev 1. 7-26-94
Rev 2. 2-13-95

## 4.1 General Description

The MPC[13] Master Control Panel contains an 8031 microcomputer with on-board UART and RS-232 interface circuitry for communications with other MPC control panels, control receivers, and a Host Computer.

This section supplies the details necessary to understand RS-232 communications between the Host Computer and the MPC Master Control Panel as used in the Camera control system (Figure 5, page 40).

## 4.2 Message Format

Anytime a command message is sent from the Host Computer to the MPC Master Control Panel, the MPC Master will respond with an ACK or an NAK[14] (Figure 5, page 40). Data is transmitted using 1 start bit, 8 data bits, and 1 stop bit (no parity). The baud rate is set by a DIP switch located on the RS-232 board. It is typically set to 9600 baud.

The commands are sent using the command message format show in Section 4.5.1, page 40.

The various commands that may be transmitted are listed in Section 4.6, page 41.

If the command message contains a latch function command, then the latch status response (Section 4.7.1, page 43, Format of Latch Status Response) will be sent back to the Host Computer. If the command message contains a home function command, then the home status response (Section 4.7.2, page 43, Format of Home Status Response from MPC) will be sent back to the Host Computer.

If the command is a position message, the response will be as shown in Section 4.8, page 44.

If the command message contains only a momentary function, no further response beyond the ACK or NAK will occur. If the command message contains a communications error (e.g., an address to a non-existent control receiver or a failure in the communications system), then the command error message response (in Section 4.7.1, page 43, with $B_3$[15] of byte 3 equal to 1) will be sent back.

Section 4.9, page 45 gives the RS-232 switch message format.

Section 4.10, page 45 gives the ID message format for communications from a host computer to the master control panel. An ACK or NAK is returned.

## 4.3 Maintenance

No maintenance adjustments are required for the RS-232 circuits on the processor board.

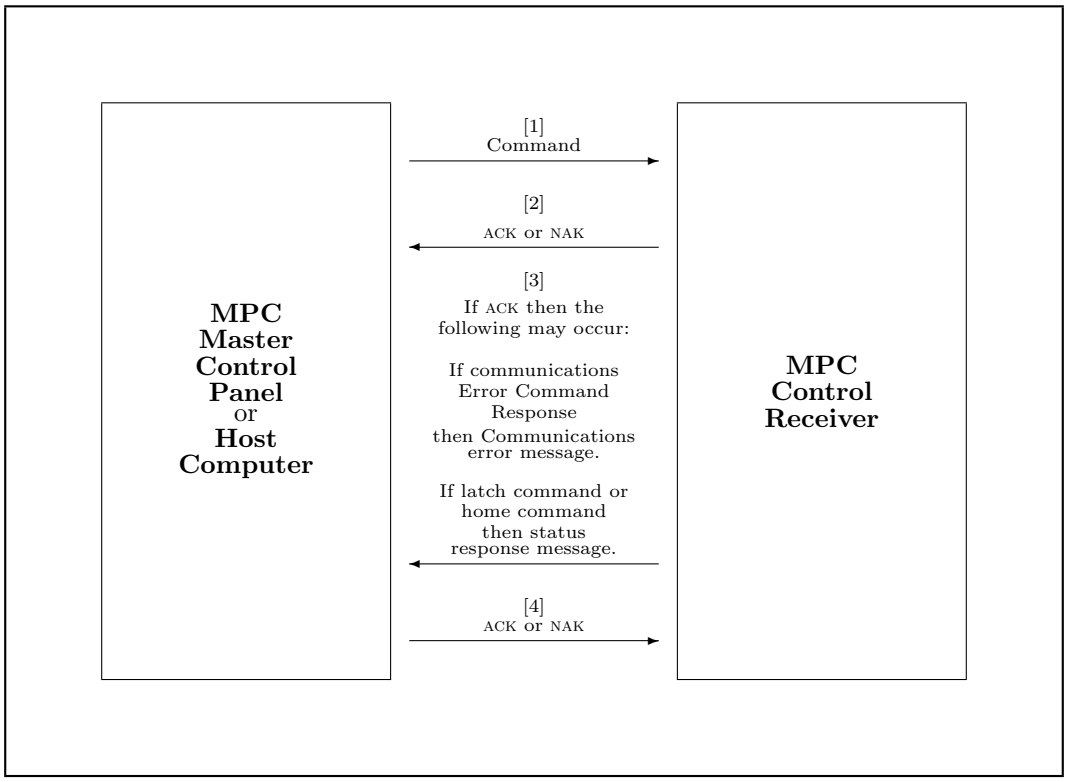
## 4.4 Parts List

Refer to the processor board for parts identifications.

[12] `$Header: d:/TXB-Cohu/RCS/232Pro.inc,v 1.9 2006-08-24 14:43:12-07 Hamilton Exp Hamilton $`
[13] This document was derived from Cohu's document: "mpc232pro.pdf"
[14] ACK = `0x06`, NAK = `0x15`.
[15] The phrase "$B_3$" means bit 3 with numbering starting at 0 in the least signifiant position of a byte.

## 4.5 Schematic Diagrams

Refer to the proces or board for the schematic diagram.




$RCSfile: TCP.inc,v $

Figure 5. Typical Communications Protocol (From Cohu Document 6X-5046(A)

### 4.5.1 RS-232 Command Message Format

| Byte | Data | Description |
|---|---|---|
| 0 | STX | Start of message (0x02) |
| 1 | Address in hex | Camera address (0x01 ⟶ 0xDF) |
| 2 to $2n+1$ | Command Data | See Section 4.6, page 41 |
| $2n+2$ | CHECKSUM | 0x80 to 0x8F Least significant nibble is XOR of all previous bytes (LS Nibbles only) except STX |
| $2n+3$ | ETX | End of Message (0x03) |

NOTE: $n$ is the number of commands in the message

## 4.6 RS-232 Command Data

DATA consists of the ASCII codes for the two letters shown:
"`rs`" = ASCII codes for "`r`" and
"`s`", "`PL`" = ASCII codes for "`P`" and "`L`", etc.

### 4.6.1 RS-232 Command Data, Momentary

| Function | Data | Description |
|---|---|---|
| Reset Control Receiver | "`rs`" | |
| Pan Left | "`PL`" | |
| Pan Right | "`PR`" | |
| Pan Stop | "`PS`" | |
| Tilt Up | "`TU`" | |
| Tilt Down | "`TD`" | |
| Tilt Stop | "`TS`" | |
| Zoom In | "`ZI`" | |
| Zoom Out | "`ZO`" | |
| Zoom Stop | "`ZS`" | |
| Focus Near | "`FN`" | |
| Focus Far | "`FF`" | |
| Focus Stop | "`FS`" | |
| Iris Open | "`IO`" | |
| Iris Close | "`IC`" | |
| Iris Stop | "`IS`" | |
| Auto Scan On | "`AS`" | |
| Auto Scan Off | "`AO`" | |
| P/T Position Request | "`P?`" | See Section 4.8.1, page 44 for position format. |
| P/T Goto Command | "`p`$A_2A_1A_0E_2E_1E_0$" | |
| Lens Position Request | "`V?`" | |
| Lens Goto Command | "`v`$A_2A_1A_0E_2E_1E_0$" | See Section 4.8.2, page 44 for position format. |

### 4.6.2 RS-232 Command Data, Latch

| Function | Data | Description |
|---|---|---|
| Manual Iris Toggle | "LM" | |
| Camera Power Toggle | "LP" | See Section 4.7.1, page 43 for response format |
| Lens Speed Toggle | "LL" | |
| Latch Status Request | "L?" | |
| Aux Functions (Option 1) | "L1" ⟶ "L3" (L1, L2, L3) | Toggles Aux 1, Aux 2, or Aux 3 |
| Color Balance (Option 2) | "L1" | Select auto/manual mode |
| | "B1" | Increase blue |
| | "B2" | Increase red |
| | "B0" | Balance stop |

### 4.6.3 RS-232 Command Data, Home

| Function | Data | Description |
|---|---|---|
| Home Position Goto | Preset 1 ⟶ 9<br>"H0" ⟶ "H9" | A home active "HA" will be immediate response. Later a home position "H0" to "H9" or home error "HE" will be generated<br>(H0, H1, H2, H3, H4, H5, H6, H7, H8, H9) |
| Home Position Store | "P0" ⟶ "P9"<br>Program 0 ⟶ 9 | (P0, P1, P2, P3, P4, P5, P6, P7, P8, P9) |
| Home Position Status Request | "H?" | Will generate a home status response See Section 4.7.2, page 43 |

## 4.7 RS-232 Response Messages

### 4.7.1 Format of Latch Status Response from MPC

| Byte | Data | Description |
|---|---|---|
| 0 | STX | Start of message (0x02) |
| 1 | Address in hex | Camera address 0x01 to 0xDF |
| 2 | "L" Latch Status | |
| 3 | 0x30 to 0x37<br>0x38 (Comm error) | LS Nibble is four bits of status<br>(L0, L1, L2, L3, L4, L5, L6, L7, L8) |
| | | Bit Value 0/1<br>$B_0$ = Iris Auto/Manual<br>$B_1$ = Camera power Off/On<br>$B_2$ = Lens Speed Slow/Fast<br>$B_3$ = Communications Error No/Yes |
| 4 | "A" Aux Status | |
| 5 | 0x30 to 0x37 | LS Nibble is four bits of status<br>(A0, A1, A2, A3, A4, A5, A6, A7)<br>Bit Value 0/1<br>$B_0$ = Aux 1<br>$B_1$ = Aux 2<br>$B_2$ = Aux 3 |
| 6 | CHECKSUM | 0x80 to 0x8F Least significant nibble is XOR of all previous bytes (LS Nibble only) except STX |
| 7 | ETX | End of message (0x03) |

### 4.7.2 Format of Home Status Response from MPC

| Byte | Data | Description |
|---|---|---|
| 0 | STX | Start of message (0x02) |
| 1 | Address in hex | Camera address 0x01 0xDF |
| 2 | "H" Home Status | |
| 3 | "0" $\longrightarrow$ "9" or "A", or "I", or "E" | ie, "2" = Home position 2<br><br>"A" = (HA) Home active<br>"I" = (HI) Not-at-Home position, or active<br>"E" = (HE) Home eror could not get to home position |
| 4 | Checksum | 0x80 to 0x8F Least significant nibble is XOR of all previous bytes (LS Nibble only) except STX |
| 5 | ETX | End of message (0x03) |

## 4.8 MPC Position Messages

### 4.8.1 Format of Pan/Tilt Position Message

| Byte | Data | Description |
|---|---|---|
| 0 | STX | Start of message (0x02) |
| 1 | Address in hex | Camera address (0x01 to 0xDF) |
| 2 | "P" or "p" | Upper case "P" is response;<br>Lower case "p" is GOTO command |
| 3 | $A_2$ (MS nibble) | |
| 4 | $A_1$ | Azimuth position (See Notes) |
| 5 | $A_0$ (LS nibble) | |
| 6 | $E_2$ (MS nibble) | |
| 7 | $E_1$ | Elevation position (See Notes) |
| 8 | $E_0$ (LS nibble) | |
| 9 | CHECKSUM | 0x80 to 0x8F Least significant nibble is XOR of all previous bytes (LS Nibble only) except STX |
| 10 | ETX | End of message (0x03) |

### 4.8.2 Format of Zoom/Focus Position Message

| Byte | Data | Description |
|---|---|---|
| 0 | STX | Start of message (0x02) |
| 1 | Address in hex | Camera address (0x01 to 0xDF) |
| 2 | "V" or "v" | Upper case "V" is response;<br>Lower case "v" is GOTO command |
| 3 | $A_2$ | (MS nibble) |
| 4 | $A_1$ Zoom position | (See Notes) |
| 5 | $A_0$ | (LS nibble) |
| 6 | $E_2$ | (MS nibble) |
| 7 | $E_1$ Focus position | (See Notes) |
| 8 | $E_0$ | (LS nibble) |
| 9 | CHECKSUM | 0x80 to 0x8F Least significant nibble is XOR of all previous bytes (LS Nibble only) except STX |
| 10 | ETX | End of message (0x03) |

**Encoding position data**

The 12-bit position data is contained in nibbles $0 \longrightarrow 2$, with:

nibble $0 = b_0 \longrightarrow b_3$,
nibble $1 = b_4 \longrightarrow b_7$, and
nibble $2 = b_8 \longrightarrow b_{11}$.

The nibble is then added to 0x30 to produce a value from 0x30 to 0x3F.

Example: Encoding an azimuth position value of decimal $2748_{10}$ or 0xABC, is broken up into three nibbles and added 0x30 to each so:

$A_2$ = 0x3A,
$A_1$ = 0x3B, and
$A_0$ = 0x3C.

The same encoding scheme is used for the zoom and focus positions in the "Zoom/Focus Goto Command".

## 4.9 RS-232 Switch Message Format

| Byte | Data | Description |
|---|---|---|
| 0 | STX | Start of message (0x02) |
| 1 | Address in hex | The remote control panel address in hex that the switch message will be displayed on. This remote will also be given control of the control receiver for the camera number in the switch message (if applicable). Binary 0 to 30 valid for remote control panels. Binary 31 refers to the master control panel. Data greater than 31 will provide for video switching only; no control panel is associated. |
| 2 | "S" | ASCII character for letter S |
| 3 | C100 | ASCII character for the camera number. C100 is the hundred's digit, |
| 4 | C10 | C10 is the ten's digit, |
| 5 | C1 | and C1 is the unit's digit |
| 6 | M100 | ASCII character for monitor number. M100 is the hundres's digit, |
| 7 | M10 | M10 is the ten's digit, |
| 8 | M1 | and M1 is the unit's digit |
| 9 | CHECKSUM | CHECKSUM is a hexadecimal number 0x80 to 0x8F. The least significant (nibble) is the XOR of all previous bytes (LS Nibble only) except STX |
| 10 | ETX | End of Message (0x03) |

## 4.10 ID Message Formats

### 4.10.1 Format of Select Mode Message

| Byte | Data | Description |
|---|---|---|
| 0 | STX | Start of Message (0x02) |
| 1 | Address in hex | Camera address (0x01 to 0xDF) |
| 2 | "d" | Display message |
| 3 | "M" | (dM) Mode select |
| 4 | "M"<br>"I" | (dMM) Menu mode<br>(dMI) ID mode |
| 5 | CHECKSUM | 0x80 to 0x8F Least significant nibble is XOR of all previous bytes (LS Nibble only) except STX |
| 6 | ETX | End of Message (0x03) |

### 4.10.2 Format of ID Enable/Disable Message

| Byte | Data | Description |
|---|---|---|
| 0 | STX | Start of Message (0x02) |
| 1 | Address in hex | Camera address (0x01 to 0xDF) |
| 2 | "d" | Display message |
| 3 | "I" | (dI) ID display |
| 4 | "D"<br>"E" | (dID) Disable ID display<br>(dIE) Enable ID display |
| 5 | CHECKSUM | 0x80 to 0x8F Least significant nibble is XOR of all previous bytes (LS Nibble only) except STX |
| 6 | ETX | End of Message (0x03) |

### 4.10.3 Format of ID Clear Screen Message

| Byte | Data | Description |
|---|---|---|
| 0 | STX | Start of Message (0x02) |
| 1 | Address in hex | Camera address (0x01 to 0xDF) |
| 2 | "d" | Display message |
| 3 | "C" | (dC) Clear screen |
| 4 | CHECKSUM | 0x80 to 0x8F Least significant nibble is XOR of all previous bytes (LS Nibble only) except STX |
| 5 | ETX | End of Message (0x03) |

### 4.10.4 Format of Select ID Top Message

| Byte | Data | Description |
|---|---|---|
| 0 | STX | Start of message (0x02) |
| 1 | Address in hex | Camera address (0x01 to 0xDF) |
| 2 | "d" | Display message |
| 3 | "I" | (dI) ID display |
| 4 | "T" | (dIT) ID at top |
| 5 | CHECKSUM | 0x80 to 0x8F Least significant nibble is XOR of all previous bytes (LS Nibble only) except STX |
| 6 | ETX | End of Message (0x03) |

### 4.10.5 Format of Select ID Bottom Message

| Byte | Data | Description |
|---|---|---|
| 0 | STX | Start of message (0x02) |
| 1 | Address in hex | Camera address (0x01 to 0xDF) |
| 2 | "d" | Display message |
| 3 | "I" | (dI) ID display |
| 4 | "B" | (dIB) ID at bottom |
| 5 | CHECKSUM | 0x80 to 0x8F Least significant nibble is XOR of all previous bytes (LS Nibble only) except STX |
| 6 | ETX | End of Message (0x03) |

### 4.10.6 Format of Update Line Message

| Byte | Data | Description |
|---|---|---|
| 0 | STX | Start of message (0x02) |
| 1 | Address in hex | Camera address (0x01 to 0xDF) |
| 2 | "d" | Display message |
| 3 | "L" | (dL) Line of text |
| 4 | "1" $\longrightarrow$ "<"<br><br>(0x31 $\longrightarrow$ 0x3C) | Line number<br>(dL1, dL2, dL3, dL4, dL5, dL6, dL7, dL8, dL9, dL:, dL; dL>)<br>(1 $\longrightarrow$ 12) + 0x30 |
| 5 | (ID text) | ASCII characters (up to 24) for the line starting with the left most character |
| $n$ | CHECKSUM | 0x80 to 0x8F Least significant nibble is XOR of all previous bytes (LS Nibble only) except STX |
| $n+1$ | ETX | End of Message (0x03) |

### 4.10.7 Format of Alarm Display Message

| Byte | Data | Description |
|---|---|---|
| 0 | STX | Start of message (0x02) |
| 1 | Address in hex | Camera address (0x01 to 0xDF) |
| 2 | "d" | Display message |
| 3 | "A" | (dA) Alarm message |
| 4 | "1" $\longrightarrow$ "3"<br><br>(0x31 $\longrightarrow$ 0x33) | Display line number<br>(dA1, dA2, dA3)<br>(1 $\longrightarrow$ 3) + 0x30 |
| 5 | "B"<br>"E"<br>"D" | (dA1B, dA2B, dA3B) Blink alarm message<br>(dA1E, dA2E, dA3E) Enable alarm message<br>(dA1D, dA2D, dA3D) Disable alarm message |
| 6 | CHECKSUM | 0x80 to 0x8F Least significant nibble is XOR of all previous bytes (LS Nibble only) except STX |
| 7 | ETX | End of Message (0x03) |

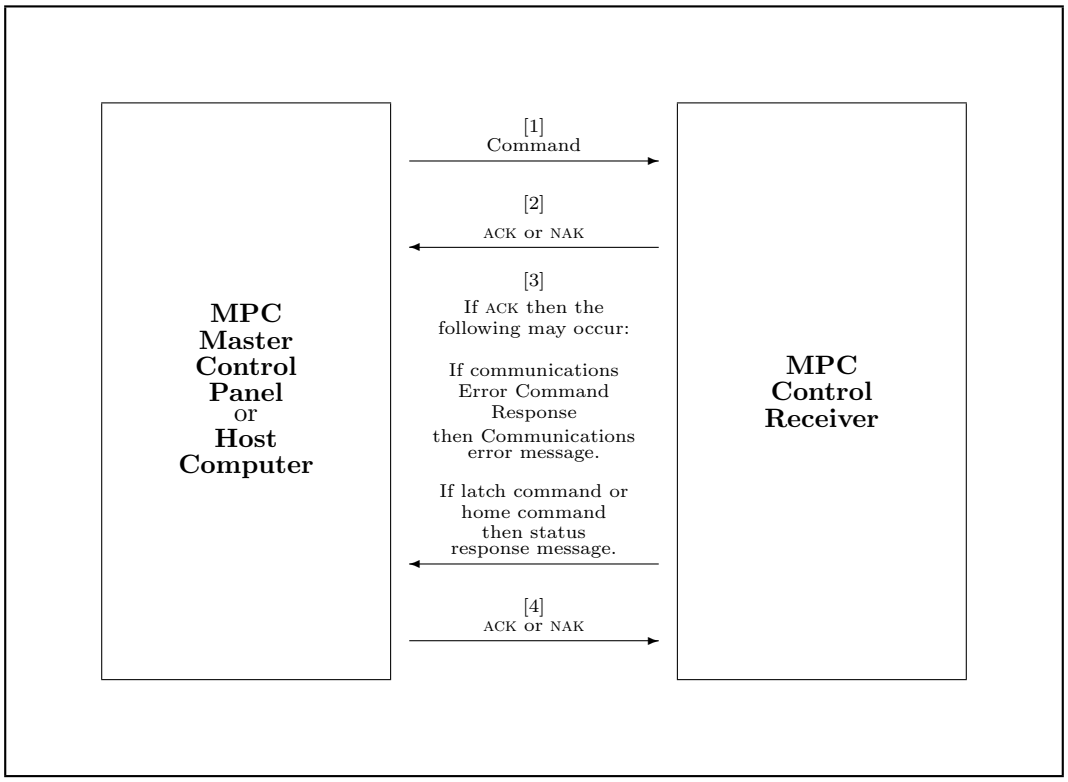
NOTE: For all messages from a host computer to the master control panel, ACK or NAK is returned.

This page intentionally left blank

# 5 3850 Variable Speed Pan/Tilt Communication Interface Protocol

## 5.1 General

This document[17] defines the Asynchronous communication protocol between the MPC Master Control Panel/Host Computer and a Positioner/Dome camera. Figure 1. illustrates the handshake protocol. The communication is a full duplex. Typical or default communication parameters will be 9600 baud, 8 data bits, 1 stop bit and no parity.




$RCSfile: TCP.inc,v $

Figure 6. Typical Communications Protocol (Similar to the one in 3850pro.pdf)

Notes: The time between two consecutive commands to the camera is 250 milliseconds. For example, Zoom In and follow right after that is Zoom Stop.

## 5.2 Message Format

Anytime a command message is sent to the Camera the camera will respond with an ACK or NAK. The commands are sent using the command message format show in Section 5.2.1, page 50. The response message

[16] `$Header: d:/TXB-Cohu/RCS/3850pro.inc,v 1.17 2006-08-30 12:31:56-07 Hamilton Exp Hamilton $`
[17] This document was derived from Cohu's document: "3850pro.pdf"

from the camera is shown in Section 5.2.6, page 58.

If the command message contains only a momentary function, no further response beyond the ACK or NAK will occur .

The control characters to be used are:

- `0xF8` Start Character
- ACK Acknowledge proper receipt of transmission
- NAK Bad transmission — retransmit
- CS Checksum (End of Message)
- ID Camera address
- `##` One byte
- `#` Number in ASCII
- $n$ is the number of command data bytes in the message.

All transmissions will be in format:

`0xF8`/ID/`c`/DATA BYTES/CHECKSUM

Note: `Wa` means ASCII characters for "`W`" and "`a`".

### 5.2.1 3850 Command Message Format

| Byte | Data | Description |
|---|---|---|
| 0 | `0xF8` | Start of message |
| 1 | Camera Address | (`0x01` to `0xDF`) Note: `0xFF` address is used to read camera address or to assign camera address |
| 2 | `c` | Message type (for camera control) |
| 3 to $n+3$ | Command Data | See Section 5.2.2, page 51 (for camera control)<br><br>See Section 5.2.3, page 56 (for variable speed pan/tilt) |
| $n+4$ | CHECKSUM | End of message `0x80` $\longrightarrow$ `0x8F` Least significant nibble is XOR of all previous bytes (LS nibbles only) except `0xF8` |

**Note**

1. $n$ is the number of command data bytes in the message.
2. Assign camera address msg. does not contain the `c`.
3. Set Camera/Preset ID msg. does not contain the `c`.
4. Preset programming and goto does not contain `c`.
5. All standard commands won't contains "`c`". (Section 5.2.4, page 56)
6. Var. Speed pan/tilt msg. does not contain the `c`. `r` `l` `u` `d` (For variable speed pan/tilt).

### 5.2.2 3850 Command/Response Data Command

| Command | Description |
|---|---|
| `cAS` | Read Camera Address |
| `cA###` | Return the current CamAddr in ASCII<br>`###` = $001_{10} \longrightarrow 255_{10}$ |
| `As###` | Assign Camera Address<br>`###` = $001_{10} \longrightarrow 255_{10}$<br>NOTE: The above command is also valid if Camera Address `0xFF` is used. |
| `cB1`<br>`cB0` | Back Light ON<br>Back Light OFF |
| `cBS`<br>`cB0`<br>`cB1` | Request for current Status of Back Light<br>Return the current Status of Back Light Off<br>Return the current Status of Back Light On |
| `cD#` | Digital Zoom Range<br>`cD0` OFF<br>`cD1` 2X<br>`cD2` 4X<br>`cD3` 8X |
| `cDS` | Request for current Status of Digital Zoom Range of Digital Zoom Mode |
| `cZ#` | Response to Return the current Status of Digital Zoom Range of Digital Zoom Mode<br>`cZ0` OFF<br>`cZ1` 2X<br>`cZ2` 4X<br>`cZ3` 8X |
| `cS#` | Shutter<br>`cS0` 1/4 `cS7` 1/250<br>`cS1` 1/8 `cS8` 1/500<br>`cS2` 1/15 `cS9` 1/1000<br>`cS3` 1/30 `cS:` 1/2000<br>`cS4` Auto `cS;` 1/4000<br>`cS5` 1/60 `cS<` 1/10000<br>`cS6` 1/100 — |

*Continued on the next page.*

*Continued from the previous page.*

| Command | Description |
|---|---|
| cSS | Request for current Status of Shutter |
| cS# | Return the current Status of Shutter<br>cS0 1/4 cS7 1/250<br>cS1 1/8 cS8 1/500<br>cS2 1/15 cS9 1/1000<br>cS3 1/30 cS: 1/2000<br>cS4 Auto cS; 1/4000<br>cS5 1/60 cS< 1/10000<br>cS6 1/100 — |
| cIA | Iris Auto |
| cIM | Iris Manual |
| cIS | Request for Current Status of Iris Mode |
| cI# | Return the current Status of Iris Mode<br>cI0 Auto<br>cI1 Manual |
| cIO | Iris Open |
| cIC | Iris Close |
| cIs | Iris Stop |
| cW# | White Balance<br>cW0 Auto<br>cW1 Set<br>cW2 Lock<br>cW3 Indoor<br>cW4 Outdoor<br>cW5 Flourescent |
| cWS | Request for current Status of White Balance |
| cW# | Return the current Status of White Balance<br>cW0 Auto<br>cW1 Set<br>cW2 Lock |

*Continued on the next page.*

*Continued from the previous page.*

| Command | Description |
|---|---|
| | cW3 Indoor<br>cW4 Outdoor<br>cW5 Florescent |
| cz# | Zoom Wide<br>cz0 Low Speed<br>cz1 Medium Speed<br>cz2 Fast Speed |
| cZ# | Zoom Tele<br>cz0 Low Speed<br>cz1 Medium Speed<br>cz2 Fast Speed |
| cZs | Zoom Stop |
| cFA | Focus Auto |
| cFM | Focus Manual |
| cFS | Request for current Status of Focus Mode |
| cF# | Return the current Status of Focus Mode<br>cF0 Auto<br>cF1 Manual |
| cF# | Focus Far<br>cF0 Low Speed<br>cF1 Medium Speed<br>cF2 Fast Speed |
| cN# | Focus Near<br>cN0 Low Speed<br>cN1 Medium Speed<br>cN2 Fast Speed |
| cFs | Focus Stop |
| P## | Program Preset Position See Table 18, page 55.<br>P## = 0x10 to 0x3E (first 47 positions)<br>P## = 0x60 to 0x6F (last 16 positions) |
| H## | Goto a Preset Position See Table 18, page 55<br>H## = 0x10 to 0x3E (first 47 positions)<br>H## = 0x60 to 0x6F (last 16 positions) |
| cRS | Firmware Revision Query |
| | Return Firware Revision (Maximum 24 Characters and blank padded to 24 total characters)<br>Model# Firmware Ver. 1.0 |

*Continued on the next page.*

| *Continued from the previous page.* | |
|---|---|
| Command | Description |
| | For example:<br>3855␣Firmware␣Ver.␣2.2. |
| dIE | Enable ID display |
| dID | Disable ID display |
| dIT | Display ID on top of screen |
| dIB | Display ID on bottom of screen |
| dL#<DATA> | Set content of ID messages<br>dL1<DATA>: line 1 (ID line 1)<br>dL2<DATA>: line 2 (ID line 2)<br>dL3<DATA>: line 3 (Alarm ID line 1)<br>dL4<DATA>: line 4 (Alarm ID line 2)<br>dL5<DATA>: line 5 (Alarm ID line 3)<br>DATA = 24 ASCII characters |
| dP##<DATA> | Store Preset ID display in the receiver<br>Preset ID will be displayed when a preset is executed, and disappears when a P/T/Z/F command is executed.<br>## = 0x10 to 0x3F (first 48 positions) See Table 18, page 55<br>## = 0x60 to 0x6F (last 16 positions)<br>DATA = 24 ASCII characters |
| cd##S | Get Preset ID display stored in receiver to the GUI |
| | ## = 0x10 to 0x3F (first 48 positions) See Table 18, page 55<br>## = 0x60 to 0x6F (last 16 positions) |
| dP | ##<DATA> |
| dAE | Enable Alarm display |
| dAD | Disable Alarm # display |
| dAB | Enable Alarm Blinking |
| dAb | Disable Alarm Blinking |

| Protocol | Preset # | Protocol | Preset # | Protocol | Preset # | Protocol | Preset # |
|---|---|---|---|---|---|---|---|
| 0x10 | $1_{10}$ | 0x20 | $17_{10}$ | 0x30 | $33_{10}$ | 0x61 | $49_{10}$ |
| 0x11 | $2_{10}$ | 0x21 | $18_{10}$ | 0x31 | $34_{10}$ | 0x62 | $50_{10}$ |
| 0x12 | $3_{10}$ | 0x22 | $19_{10}$ | 0x32 | $35_{10}$ | 0x63 | $51_{10}$ |
| 0x13 | $4_{10}$ | 0x23 | $20_{10}$ | 0x33 | $36_{10}$ | 0x64 | $52_{10}$ |
| 0x14 | $5_{10}$ | 0x24 | $21_{10}$ | 0x34 | $37_{10}$ | 0x65 | $53_{10}$ |
| 0x15 | $6_{10}$ | 0x25 | $22_{10}$ | 0x35 | $38_{10}$ | 0x66 | $54_{10}$ |
| 0x16 | $7_{10}$ | 0x26 | $23_{10}$ | 0x36 | $39_{10}$ | 0x67 | $55_{10}$ |
| 0x17 | $8_{10}$ | 0x27 | $24_{10}$ | 0x37 | $40_{10}$ | 0x68 | $56_{10}$ |
| 0x18 | $9_{10}$ | 0x28 | $25_{10}$ | 0x38 | $41_{10}$ | 0x69 | $57_{10}$ |
| 0x19 | $10_{10}$ | 0x29 | $26_{10}$ | 0x39 | $42_{10}$ | 0x6A | $58_{10}$ |
| 0x1A | $11_{10}$ | 0x2A | $27_{10}$ | 0x3A | $43_{10}$ | 0x6B | $59_{10}$ |
| 0x1B | $12_{10}$ | 0x2B | $28_{10}$ | 0x3B | $44_{10}$ | 0x6C | $60_{10}$ |
| 0x1C | $13_{10}$ | 0x2C | $29_{10}$ | 0x3C | $45_{10}$ | 0x6D | $61_{10}$ |
| 0x1D | $14_{10}$ | 0x2D | $30_{10}$ | 0x3D | $46_{10}$ | 0x6E | $62_{10}$ |
| 0x1E | $15_{10}$ | 0x2E | $31_{10}$ | 0x3E | $47_{10}$ | 0x6F | $63_{10}$ |
| 0x1F | $16_{10}$ | 0x2F | $32_{10}$ | 0x60 | $48_{10}$ | 0x70 | $64_{10}$ |

Table 18. Extended Preset Values

### 5.2.3 Variable Pan/Tilt Command Data

| Command | Description |
|---|---|
| r## | Pan Right ##: 16 variable speed controls (from 0x30 to 0x3F) |
| | (r0, r1, r2, r3, r4, r5, r6, r7, r8, r9, r:, r;, r<, r=, r>, r@, rA) |
| l## | Pan Left ##: 16 variable speed controls (from 0x30 to 0x3F) |
| | (l0, l1, l2, l3, l4, l5, l6, l7, l8, l9, l:, l;, l<, l=, l>, l@, lA) |
| PS | Pan Stop |
| u## | Tilt Up ##: 16 variable speed controls (from 0x30 to 0x3F) |
| | (u0, u1, u2, u3, u4, u5, u6, u7, u8, u9, u:, u;, u<, u=, u>, u@, uA) |
| d## | Tilt Down ##: 16 variable speed controls (from 0x30 to 0x3F) |
| | (d0, d1, d2, d3, d4, d5, d6, d7, d8, d9, d:, d;, d<, d=, d>, d@, dA) |
| TS | Tilt Stop |

### 5.2.4 Standard Momentary Commands

Note: All standard command don't contain "c" as DSP commands.

| Command | Description |
|---|---|
| PR | Pan Right |
| PL | Pan Left |
| PS | Pan Stop |
| TU | Tilt Up |
| TD | Tilt Down |
| TS | Tilt Stop |
| rn | Pan Right — Nudge Notes: Nudge commands are optional |
| ln | Pan Left — Nudge |
| un | Tilt Up — Nudge |
| dn | Tilt Down — Nudge |

Note: All above command will generate variable speed pan/tilt based on the Camera Zoom position.

| Command | Description |
|---|---|
| ZI | Zoom In |
| ZO | Zoom Out |
| ZS | Zoom Stop |
| FN | Focus Near |
| FF | Focus Far |
| FS | Focus Stop |
| IO | Iris Open |
| IC | Iris Close |

*Continued on the next page.*

*Continued from the previous page.*

| Command | Description |
|---|---|
| `IS` | Iris Stop |
| `rs` | Software Reset |

### 5.2.5 Latch Commands

| Command | Description |
|---|---|
| LM | Manual Iris toggle |
| LL | Lens Speed toggle |
| L? | Latch Status Request. See Section 5.2.7, page 58 for response status |
| L1 | Aux 1 Function (Auto/Manual Focus) |
| B1 | Integration Up |
| B2 | Integration Down. Continue to do this to go back to 1/60th Shutter |

### 5.2.6 Variable Pan/Tilt Command Data

**Home Commands**

| Command | Description |
|---|---|
| H0 to H9 | Goto Presets 1 to 10<br>(H0, H1, H2, H3, H4, H5, H6, H7, H8, H9) |
| P0 to P9 | Program Presets 1 to 10<br>(P0, P1, P2, P3, P4, P5, P6, P7, P8, P9) |
| H? | Home Position Request. See Section 5.2.7, page 58 for response status |

### 5.2.7 Standard Latch/Home Response Message Format

**Latch Status Response**

| Byte | Data | Description |
|---|---|---|
| 0 | 0xF8 | Start of message |
| 1 | Camera Address | (0x01 to 0xDF) |
| 2 | L | Latch status |
| 3 | 0x30 $\longrightarrow$ 0x37 | LS nibble is four bits of status<br>Bit Value 0/1<br>$B_0$ = Auto Iris/Manual Iris<br>$B_1$ = Camera Power Off/On<br>$B_2$ = Lens Speed Slow/Fast<br>$B_3$ = Not Used<br>(L0, L1, L2, L3, L4, L5, L6, L7) |
| 4<br>5 | A<br>0x30 $\longrightarrow$ 0x37 | Aux status<br>LS nibble is four bits of status<br>Bit Value 0/1<br>$B_0$ = Auto Integration/Manual Integration<br>$B_1$ = Not Used<br>$B_2$ = Not used<br>$B_3$ = Not Used<br>(A0, A1) |
| 6 | CHECKSUM | End of message 0x80 $\longrightarrow$ 0x8F Least significant nibble is XOR of all previous bytes (LS nibbles only) except 0xF8 |

### 5.2.8 Home Status Response

| Byte | Data | Description |
|---|---|---|
| 0 | `0xF8` | Start of message |
| 1 | Camera Address | (`0x01` to `0xDF`) |
| 2 | `H` | Latch status |
| 3 | `0x30` ⟶ `0x39` | Home Position (`H0`, `H1`, `H2`, `H3`, `H4`, `H5`, `H6`, `H7`, `H8`, `H9`) |
| 4 | CHECKSUM | End of message `0x80` ⟶ `0x8F` Least significant nibble is XOR of all previous bytes (LS nibbles only) except `0xF8` |

### 5.2.9 DSP Response Message Format

| Byte | Data | Description |
|---|---|---|
| 0 | `0xF8` | Start of message |
| 1 | Camera Address | (`0x01` to `0xDF`) Note: `0xFF` address is used to read camera address or to assign camera address |
| 2 | `c` | Message type (camera) |
| 3 to $n+3$ | Response Data | Refer to Response Data of those commands above |
| $n+4$ | CHECKSUM | End of message `0x80` ⟶ `0x8F` Least significant nibble is XOR of all previous bytes (LS nibbles only) except `0xF8` |

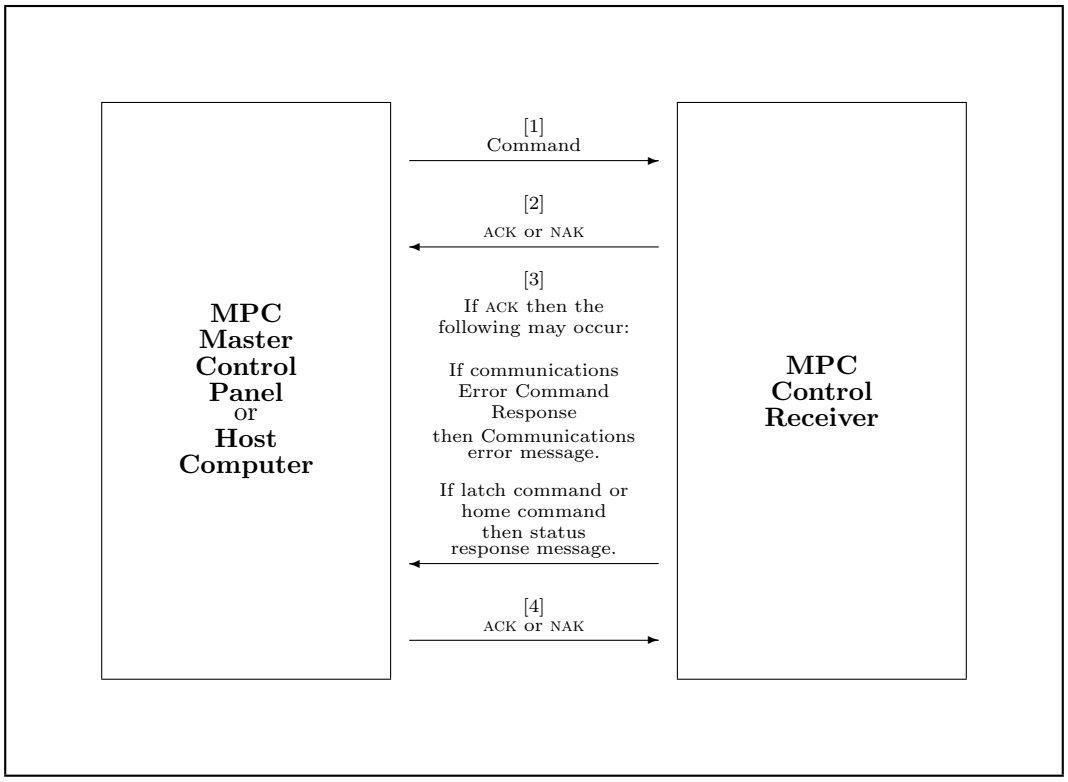
This page intentionally left blank

# 6 iDome/iView/LCU Communication Protocol

Rev 3.0
3/29/2004
COHU, INC[19]

## 6.1 General

This document[20] defines the Asynchronous communication protocol between the MPC Master Control Panel/Host Computer and an iDome/iView/LCU. Figure 7, page 61, Illustrates the handshake protocol. The communication is full duplex. Typical or default communication parameters will be 9600 baud, 8 data bits, 1 stop bit and no parity.




$RCSfile: TCP.inc,v $

Figure 7. Typical Communications Protocol (From Cohu Document 6X-5046(A)

[18] `$Header: d:/TXB-Cohu/RCS/iDome.inc,v 1.14 2006-08-29 10:32:13-07 Hamilton Exp Hamilton $`
[19] Electronics Division, P.O. Box 85623, San Diego, CA 92186-5626 — Telephone: (619) 277-6700 Fax: (619) 277-0221 Web: `www.cohu.com/cctv`
[20] This document was derived from the Cohu document:"iDomeiViewLCUCommProtocol.pdf"

## 6.2 Message Format

Anytime a command message is sent to the Camera the camera will respond with an ACK or NAK. The commands are sent using the command message format show in Section 6.2.1, page 62 to Section 6.2.4, page 63. The response message from the camera is shown in Section 6.4, page 72 and Section 6.5.1, page 76.

If the command message contains only a momentary function, no further response beyond the ACK or NAK will occur. If the command requires a status message response, then a message will be returned following the ACK.

The control characters to be used are:

- `0xF8` Start Character
- ACK Acknowledge proper receipt of transmission
- NAK Bad transmission — retransmit
- CS Checksum (End of Message)
- ID Camera address
- `##` One byte in Hex
- "`#`" Number in ASCII
- "`*`" ASCII Asterisk
- "`c`" ASCII `c`
- "`d`" ASCII `d`

All transmissions will be in formats:

1. <`0xF8`> <CamID> <DATA> <CS> (Section 6.2.1, page 62).
   or:
2. <`0xF8`> <CamID> <`c`> <DATA> <CS> (Section 6.2.2, page 63).
   or:
3. <`0xF8`> <CamID> <`*`> <DATA> <CS> (Section 6.2.3, page 63).
   or:
4. <`0xF8`> <CamID> <`d`> <DATA> <CS> (Section 6.2.4, page 63).

### 6.2.1 Standard Command Message Format

| Byte | Data | Description |
|---|---|---|
| 0 | `0xF8` | Start of message |
| 1 | Camera Address | (`0x01` to `0xDF`) Note: `0xFF` address is used to read camera address or to assign camera address |
| 2 to $n+2$ | Command Data | |
| $n+3$ | CHECKSUM | End of message `0x80` — `0x8F` Least significant nibble is XOR of all previous bytes (LS Nibbles only) except `0xF8` |

NOTE: $n$ is the number of command data bytes in the message

### 6.2.2 DSP Camera Command Message Format

| Byte | Data | Description |
|---|---|---|
| 0 | `0xF8` | Start of message |
| 1 | Camera Address | (`0x01` to `0xDF`) Note: `0xFF` address is used to read camera address or to assign camera address |
| 2 | "`c`" | Message type (for camera control) |
| 3 to $n+3$ | DSP | Command Data |
| $n+4$ | CHECKSUM | End of message `0x80` — `0x8F` Least significant nibble is XOR of all previous bytes (LS Nibbles only) except `0xF8` |

### 6.2.3 Sector/Touring Command Message Format

| Byte | Data | Description |
|---|---|---|
| 0 | `0xF8` | Start of message |
| 1 | Camera Address | (`0x01` to `0xDF`) Note: `0xFF` address is used to read camera address or to assign camera address |
| 2 | "`*`" | Message type (for sector/touring) |
| 3 to $n+3$ | Touring/Position | Command Data |
| $n+4$ | CHECKSUM | End of message `0x80` — `0x8F` Least significant nibble is XOR of all previous bytes (LS Nibbles only) except `0xF8` |

### 6.2.4 Id command Message Format

| Byte | Data | Description |
|---|---|---|
| 0 | `0xF8` | Start of message |
| 1 | Camera Address | (`0x01` to `0xDF`) Note: `0xFF` address is used to read camera address or to assign camera address |
| 2 | "`d`" | Message type (for ID) |
| 3 | Command 1 | See Section 6.3.4, page 66 for command types |
| 4 | Command 2 | and reponse |
| 5 to 29 ID Data | 24 ID | ASCII characters, or Enable/Disable, or Blinking, etc. |
| $n+30$ | CHECKSUM | End of message `0x80` — `0x8F` Least significant nibble is XOR of all previous bytes (LS Nibbles only) except `0xF8` |

## 6.3 Standard Commands

| Command | Description |
|---|---|
| PR | Pan Right |
| PL | Pan Left |
| PS | Pan Stop |
| TU | Tilt Up |
| TD | Tilt Down |
| TS | Tilt Stop |
| ZI | Zoom In |
| ZO | Zoom Out |
| ZS | Zoom Stop |
| FN | Focus Near |
| FF | Focus Far |
| FS | Focus Stop |
| IO | Iris Open |
| IC | Iris Close |
| IS | Iris Stop |
| P## | Program Preset Position |
| | ## = 0x10 to 0x3E (first 47 positions) See Table 19, page 68 |
| | ## = 0x60 to 0x70 (last 16 positions) |
| H## | Goto a Preset Position |
| | ## = 0x10 to 0x3E (first 47 positions) See Table 19, page 68 |
| | ## = 0x60 to 0x70 (last 16 positions) |
| B1 | Integration Up — Continue up to 1/8th shutter speed |
| B2 | Integration Down — Continue back to 1/60th shutter speed |

### 6.3.1 Latch Commands

| Command | Description |
|---|---|
| LP | Camera Power ON/OFF |
| LM | Manual Iris toggle |
| LL | Lens Speed toggle |
| L1 | Manual Focus toggle |

Notes: All latch commands will return the status as the L? command described below.

### 6.3.2 Status request

| Command | Description |
|---|---|
| L? | Latch Status Request — See Section 6.4, page 72 for response status |
| H? | Home Position Request — See Section 6.4, page 72 for response status |

### 6.3.3 Variable Speed P/T/Z Commands

| Command | Description |
|---|---|
| `r##` | Pan Right — Var. Speed<br>`##`: 16 variable speed controls (from `0x30` to `0x3F`)<br>(`r0`, `r1`, `r2`, `r3`, `r4`, `r5`, `r6`, `r7`, `r8`, `r9`, `r:`, `r;`, `r<`, `r=`, `r>`, `r@`, `rA`) |
| `l##` | Pan Left — Var. Speed<br>`##`: 16 variable speed controls (from `0x30` to `0x3F`)<br>(`l0`, `l1`, `l2`, `l3`, `l4`, `l5`, `l6`, `l7`, `l8`, `l9`, `l:`, `l;`, `l<`, `l=`, `l>`, `l@`, `lA`) |
| `PS` | Pan Stop |
| `u##` | Tilt Up — Var. Speed<br>`##`: 16 variable speed controls (from `0x30` to `0x3F`)<br>(`u0`, `u1`, `u2`, `u3`, `u4`, `u5`, `u6`, `u7`, `u8`, `u9`, `u:`, `u;`, `u<`, `u=`, `u>`, `u@`, `uA`) |
| `d##` | Tilt Down — Var. Speed<br>`##`: 16 variable speed controls (from `0x30` to `0x3F`)<br>(`d0`, `d1`, `d2`, `d3`, `d4`, `d5`, `d6`, `d7`, `d8`, `d9`, `d:`, `d;`, `d<`, `d=`, `d>`, `d@`, `dA`) |
| `TS` | Tilt Stop |
| `rn` | Pan Right — Nudge |
| `ln` | Pan Left — Nudge |
| `un` | Tilt Up — Nudge |
| `dn` | Tilt Down — Nudge |
| `i#` | Zoom In — Var. Speed<br>"`i0`" Low Speed<br>"`i1`" Medium Speed<br>"`i2`" Fast Speed |
| `o#` | Zoom Out — Var. Speed<br>"`o0`" Low Speed<br>"`o1`" Medium Speed<br>"`o2`" Fast Speed |
| `f#` | Focus Far — Var. Speed<br>"`f0`" Low Speed<br>"`f1`" Medium Speed<br>"`f2`" Fast Speed |
| `n#` | Focus Near — Var. Speed<br>"`n0`" Low Speed<br>"`n1`" Medium Speed<br>"`n2`" Fast Speed |

### 6.3.4 ID Commands

| Command | Description |
|---|---|
| dIE | Enable ID Display |
| dID | Disable ID Display |
| dIT | Display ID on top of screen |
| dIB | Display ID on bottom of screen |
| dAE | Enable Alarm Display |
| dAD | Disable Alarm # Display |
| dAB | Enable Alarm Blinking |
| dAb | Disable Alarm Blinking |
| dME | Maintenance Display Enable |
| dMD | Maintenance Display Disable |
| dIC | Clear All ID Data, include Presets, Sectors, Privacy Zones |
| dL#<DATA> | Set content of ID messages<br>"dL1" : line 1 (ID line 1)<br>"dL2" : line 2 (ID line 2)<br>"dL3" : line 3 (Alarm ID line 1)<br>"dL4" : line 4 (Alarm ID line 2)<br>"dL5" : line 5 (Alarm ID line 3)<br>DATA = 24 ASCII characters |
| dP##<DATA> | Store Preset ID display in the receiver<br>Preset ID will be displayed when a preset is executed, and disappears when a P/T/Z/F command is executed. |
| | P## = 0x10 to 0x3E (first 48 positions) See Table 19, page 68<br>P## = 0x60 to 0x70 (last 16 positions)<br>DATA = 24 ASCII characters |
| dST#<DATA> | Set Sector ID Text<br>dST# = Sector Number (0x31 to 0x40)<br>(dST1<DATA>, dST2<DATA>, dST3<DATA>, dST4<DATA>, dST5<DATA>, dST6<DATA>, dST7<DATA>, dST8<DATA>, dST9<DATA>, dST:<DATA>, dST;<DATA>, dST<<DATA>, dST=<DATA>, dST><DATA>, dST@<DATA>, dSTA<DATA>)<br>DATA = 24 ASCII characters |
| dSS#s | Set Sector ID State<br># = Sector Number (0x31 to 0x40)<br>s = State "0"=Off: (dSS10, dSS20, dSS30, dSS40, dSS50, dSS60, dSS70, dSS80, dSS90, dSS:0, dSS;0, dSS<0, dSS=0, dSS>0, dSS@0, dSSA0) |

*Continued on the next page.*

| *Continued from the previous page.* | |
|---|---|
| Command | Description |
| | s = State "1"=On: (dSS20, dSS21, dSS31, dSS41, dSS51, dSS61, dSS71, dSS81, dSS91, dSS:1, dSS;1, dSS<1, dSS=1, dSS>1, dSS@1, dSSA1) |
| cdSc | Get ID Control Data, i.e Enable, Top, Alarm Blink ... |
| | Reply is: dC#<br># = Control Byte (xxx11111) |
| cdSl# | Get ID Messages<br>"cdSl1", "cdSl2", "cdSl3", "cdSl4", "cdSl5" |
| | Reply is: dL#<DATA> |
| cdS## | Get Preset ## ID stored in camera<br>## = 0x10 to 0x3E (first 48 positions) See Table 19, page 68<br>## = 0x60 to 0x70 (last 16 positions) |
| | Reply is: dP##<DATA> |
| cdSS## | Get Sector ## ID stored in camera<br>## = 0x31 to 0x40 (16 sectors) |
| | (cdSS1, cdSS2, cdSS3, cdSS4, cdSS5, cdSS6, cdSS7, cdSS8, cdSS9, cdSS:, cdSS;, cdSS<, cdSS=, cdSS>, cdSS@, cdSSA)<br>Reply is: dST##<DATA><br>(dST1<DATA>, dST2<DATA>, dST3<DATA>, dST4<DATA>, dST5<DATA>, dST6<DATA>, dST7<DATA>, dST8<DATA>, dST9<DATA>, dST:<DATA>, dST;<DATA>, dST<<DATA>, dST=<DATA>, dST><DATA>, dST@<DATA>, dSTA<DATA>) |
| cdSs## | Get Sector ID State<br>## = Sector Number (0x31 to 0x40) |
| | (cdSs1, cdSs2, cdSs3, cdSs4, cdSs5, cdSs6, cdSs7, cdSs8, cdSs9, cdSs:, cDSs;, cdSs<, cdSs=, cdSs>, cdSs@, cdSsA)<br>Reply is: dSS#s<br>s = State ("1"=On; "0"=Off) |
| | (dSS10, dSS20, dSS30, dSS40, dSS50, dSS60, dSS70, dSS80, dSS90, dSS:0, DSS;0, dSS<0, dSS=0, dSS>0, dSS@0, dSSA0)<br>(dSS11, dSS21, dSS31, dSS41, dSS51, dSS61, dSS71, dSS81, dSS91, dSS:1, DSS;1, dSS<1, dSS=1, dSS>1, dSS@1, dSSA1) |

### 6.3.5 Misc. Commands

| Command | Description | Response |
|---|---|---|
| A? | Read Camera Address | "A###" Return the current CamAddr in ASCII |
| | | *Continued on the next page.* |

| Protocol | Preset # | Protocol | Preset # | Protocol | Preset # | Protocol | Preset # |
|---|---|---|---|---|---|---|---|
| 0x10 | $1_{10}$ | 0x20 | $17_{10}$ | 0x30 | $33_{10}$ | 0x61 | $49_{10}$ |
| 0x11 | $2_{10}$ | 0x21 | $18_{10}$ | 0x31 | $34_{10}$ | 0x62 | $50_{10}$ |
| 0x12 | $3_{10}$ | 0x22 | $19_{10}$ | 0x32 | $35_{10}$ | 0x63 | $51_{10}$ |
| 0x13 | $4_{10}$ | 0x23 | $20_{10}$ | 0x33 | $36_{10}$ | 0x64 | $52_{10}$ |
| 0x14 | $5_{10}$ | 0x24 | $21_{10}$ | 0x34 | $37_{10}$ | 0x65 | $53_{10}$ |
| 0x15 | $6_{10}$ | 0x25 | $22_{10}$ | 0x35 | $38_{10}$ | 0x66 | $54_{10}$ |
| 0x16 | $7_{10}$ | 0x26 | $23_{10}$ | 0x36 | $39_{10}$ | 0x67 | $55_{10}$ |
| 0x17 | $8_{10}$ | 0x27 | $24_{10}$ | 0x37 | $40_{10}$ | 0x68 | $56_{10}$ |
| 0x18 | $9_{10}$ | 0x28 | $25_{10}$ | 0x38 | $41_{10}$ | 0x69 | $57_{10}$ |
| 0x19 | $10_{10}$ | 0x29 | $26_{10}$ | 0x39 | $42_{10}$ | 0x6A | $58_{10}$ |
| 0x1A | $11_{10}$ | 0x2A | $27_{10}$ | 0x3A | $43_{10}$ | 0x6B | $59_{10}$ |
| 0x1B | $12_{10}$ | 0x2B | $28_{10}$ | 0x3B | $44_{10}$ | 0x6C | $60_{10}$ |
| 0x1C | $13_{10}$ | 0x2C | $29_{10}$ | 0x3C | $45_{10}$ | 0x6D | $61_{10}$ |
| 0x1D | $14_{10}$ | 0x2D | $30_{10}$ | 0x3D | $46_{10}$ | 0x6E | $62_{10}$ |
| 0x1E | $15_{10}$ | 0x2E | $31_{10}$ | 0x3E | $47_{10}$ | 0x6F | $63_{10}$ |
| 0x1F | $16_{10}$ | 0x2F | $32_{10}$ | 0x60 | $48_{10}$ | 0x70 | $64_{10}$ |

Table 19. Extended Preset Values

*Continued from the previous page.*

| Command | Description | Response |
|---|---|---|
| | | `###` = "`001`" — "`223`" |
| `AS###` | Assign Camera Address | `###` = "`001`" — "`223`" |
| | NOTE: The above command is also valid if Camera Address `0xFF` is used. | |
| `E#####` | Elevation Level where the camera is mounted. This is used to calibrate the Low Pressure indicator in the camera.<br>`#####` = 0 to 10000 in ASCII (in feet) | |
| `RS` | iDome/iView Firmware Revision Query | Return Firmware Revision (Maximum 24 Characters blank padded)<br>"`Model#␣Firmware␣Ver.␣1.0`"<br>For example:<br>`RModel#␣Firmware␣V1.0␣␣␣` |
| `Rs` | LCU Firmware Revision Query | Return Firmware Revision (Maximum 24 Characters blank padded)<br>"`Model#␣Firmware␣Ver.␣1.0`"<br>For example:<br>`rLCU␣Firmware␣V1.0␣␣␣␣␣␣` |
| `Lp#` | Line Lock Phase Adjust<br>`Lp0` (Camera is in Crystal mode)<br>`Lp1` (Line Lock — Phase 0 mode — 0 degree)<br>`Lp2` (Line Lock — Phase 1 mode — 30 degree)<br>`Lp3` (Line Lock — Phase 2 mode — 60 degree)<br>`Lp4` (Line Lock — Phase 3 mode — 90 degree)<br>`Lp5` (Line Lock — Phase 4 mode — 120 degree)<br>`Lp6` (Line Lock — Phase 5 mode — 150 degree) | |
| `Bs####` | Set Camera Baud Rate<br>`####` = 0300, 0600, 1200, 2400, 4800, 9600 | |

*Continued on the next page.*

*Continued from the previous page.*

| Command | Description | Response |
|---|---|---|
| | (`Bs0300`, `Bs0300`, `Bs0600`, `Bs1200`, `Bs2400`, `Bs4800`, `Bs9600`) | |
| `bs####` | Set LCU Host Baud Rate<br>`####` = 0300, 0600, 1200, 2400, 4800, 9600<br>(`bs0300`, `bs0300`, `bs0600`, `bs1200`, `bs2400`, `bs4800`, `bs9600`) | |
| `b?####` | Query LCU Host Baud Rate<br><br>(`b0300`, `b0300`, `b0600`, `b1200`, `b2400`, `b4800`, `b9600`) | `b####` = 0300, 0600, 1200, 2400, 4800, 9600 |
| `fE` | Tilt Flip Enabled | |
| `fD` | Tilt Flip Disabled | |
| `fS` | iDome/iView Flip Enable Query | Return Tilt Flip Enable Status<br>"`fSE`" (Tilt Flip Enabled)<br>"`fSD`" (Tilt Flip Disabled) |
| `h1` | RS-232 Flow Control (RTS/CTS) Enabled | |
| `h0` | RS-232 Flow Control (RTS/CTS) Disabled | |
| `hS` | RS-232 Flow Control Query | Return RS-232 Handshake Status<br>"`h1`" (Enabled)<br>"`h0`" (Disabled) |
| `Jc0` | Set Javelin Continous Motor Mode for Camera OFF | |
| `Jc1` | Set Javelin Continous Motor Mode for Camera ON | |
| `JcS` | Query Camera for Javelin Continous Motor Mode status in Camera | "`Jc0`" (OFF)<br>"`Jc1` (ON) |
| `Jb0` | Set Javelin 2 Bytes Addr for camera OFF | |
| `Jb1` | Set Javelin 2 Bytes Addr for camera ON | |
| `JbS` | Query Camera for Javelin 2 Bytes Addr status in camera | "`Jb0`" (OFF)<br>"`Jb1` (ON) |

*Continued on the next page.*

*Continued from the previous page.*

| Command | Description | Response |
|---|---|---|
| `jc0` | Set Javelin Continous Motor Mode for LCU OFF | |
| `jc1` | Set Javelin Continous Motor Mode for LCU ON | |
| `jcS` | Query LCU for Javelin Continous Motor Mode status in LCU | "`jc0`" (OFF)<br>"`jc1` (ON) |
| `jb0` | Set Javelin 2 Bytes Addr for LCU OFF | |
| `jb1` | Set Javelin 2 Bytes Addr for LCU ON | |
| `jbS` | Query LCU for Javelin 2 Bytes Addr status in LCU | "`jb0`" (OFF)<br>"`jb1` (ON) |
| `mPS` | Query the internal pressure data | "`mPxx.x`"<br>`xx`: 2 digits — ASCII data and 1 digit decimal<br>For example: `10.5` |
| `mTS` | Query the internal temperature data | "`mPxx`"<br>`xx`: 2 digits — ASCII data in degree C<br>For example: `30 degree C` |
| `mSN` | Set True North Offset | |

## 6.4 Standard Latch/Home Response Message Format

### 6.4.1 Format Latch Status Response

| Byte | Data | Description |
|---|---|---|
| 0 | 0xF8 | Start of message |
| 1 | Camera Address | (0x01 to 0xDF) |
| 2 | "L" | Latch status |
| 3 | 0x30 — 0x37 | LS nibble is four bits of status<br>Bit Value 0/1<br>$B_0$ = Manual Iris/Auto Iris (1/0)<br>$B_1$ = Camera Power On/Off (1/0)<br>$B_2$ = Lens Speed Fast/Slow (1/0)<br>$B_3$ = Not Used<br>(L0, L1, L2, L3, L4, L5, L6, L7, L8) |
| 4 | "A" | Aux status |
| 5 | 0x30 — 0x37 | LS nibble is four bits of status<br>Bit Value 0/1<br>$B_0$ = Auto/Manual Focus (1/0)<br>$B_1$ = Auto Integration/Manual Integration (1/0)<br>$B_2$ = Not used<br>$B_3$ = Not Used<br>(A0, A1, A2, A3, A4, A5, A6, A7) |
| 6 | "X" | Misc. status |
| 7 | 0x30 — 0x37 | LS nibble is four bits of status<br>Bit Value 0/1<br>$B_0$ = Not Used<br>$B_1$ = Low Pressure (1)<br>$B_2$ = Video Loss (1)<br>$B_3$ = Not Used<br>(X0, X1, X2, X3, X4, X5, X6, X7) |
| 8 | CHECKSUM | End of message 0x80 — 0x8F Least significant nibble is XOR of all previous bytes (LS Nibbles only) except 0xF8 |

### 6.4.2 Format Home Status Response

| Byte | Data | Description |
|---|---|---|
| 0 | 0xF8 | Start of message |
| 1 | Camera Address | (0x01 to 0xDF) |
| 2 | "H" | Home status |
| 3 | 0x10 — 0x3E<br>0x60 — 0x70<br>"HI" | Home Position See Table 19, page 68<br>(H0, H1, H2, H3, H4, H5, H6, H7, H8, H9)<br>Home is inactive (camera is not at a preset position) |
| 4 | CHECKSUM | End of message 0x80 — 0x8F Least significant nibble is XOR of all previous bytes (LS Nibbles only) except 0xF8 |

## 6.5 DSP Commands

| Command | Description | Response |
|---|---|---|
| cD# | Digital Zoom Range<br>cD0 (OFF)<br>cD1 (2X)<br>cD2 (5X)<br>cD3 (10X) | |
| cS# | Shutter<br>cS0 (Auto) cS8 (1/180)<br>cS1 (1/2) cS9 (1/250)<br>cS2 (1/4) cS: (1/500)<br>cS3 (1/8) cS; (1/1000)<br>cS4 (1/15) cS< (1/2000)<br>cS5 (1/30) cS= (1/4000)<br>cS6 (1/60) cS> (1/10000)<br>cS7 (1/120) cS@ (1/30000) | |
| cW# | White Balance<br>cW0 (Auto)<br>cW1 (Manual) | |
| cWR | Increase RED (in Manual mode only) | |
| cWr | Decrease RED | |
| cWB | Increase BLUE | |
| cWb | Decrease BLUE | |
| cWQ | Stop Increase/Decrease color | |
| cY1 | Wide Dynamic Range ON | |
| cY0 | Wide Dynamic Range OFF | |
| cM0 | Camera in Day/Night Auto Mode | |

*Continued on the next page.*

*Continued from the previous page.*

| Command | Description | Response |
|---|---|---|
| cM1 | Camera in Day Mode (Color) | |
| cM2 | Camera in Night Mode (Monochrome) | |
| cIA | Iris Auto | |
| cIM | Iris Manual | |
| cz# | Zoom Wide — Var. Speed<br>cz0 (Low Speed)<br>cz1 (Medium Speed)<br>cz2 (Fast Speed) | |
| cZ# | Zoom Tele — Var. Speed<br>cZ0 (Low Speed)<br>cZ1 (Medium Speed)<br>cZ2 (Fast Speed) | |
| cZs | Zoom Stop | |
| cFA | Focus Auto | |
| cFM | Focus Manual | |
| cFs | Focus Stop | |
| cP# | Camera Communication Protocol Selection<br>cP0 (Cohu)<br>cP1 (American Dynamics)<br>cP2 (Pelco D)<br>cP3 (Javelin)<br>cP4 (Philips)<br>cP5 (Vicon)<br>cP6 (Kalatel)<br>cP7 (Ultrak) | |
| cPs | Query LCU Communication Protocol | P# LCU Protocol<br># = 0x30 (Cohu)<br>P1 (American Dynamics)<br>P2 (Pelco D)<br>P3 (Javelin)<br>P4 (Philips)<br>P5 (Vicon)<br>P6 (Kalatel) |
| csO | Set Progressive Scan On | |
| cso | Set Progressive Scan Off | |
| cFF | Video Freeze ON/OFF toggle | |

*Continued on the next page.*

*Continued from the previous page.*

| Command | Description | Response |
|---|---|---|
| `cF#` | Set Video Freeze ON/OFF<br>`cFO` (ON)<br>`cFo` (OFF) | |
| `cb` | Enable Boot Loader to update new firmware in the iDome/iView | |
| `cB` | Enable Boot Loader to update new firmware in the LCU | |
| `cLN` | Set Normal Light mode (ER8439E — special Camera module) | |
| `cLn` | Set Sodium Vapor Light mode (ER8439E — special Camera module) | |
| `cLS` | Status request of Light mode setting (ER8439E — special Camera module) | `cLN`: Normal mode<br>`cLn`: Sodium Vapor mode |
| `cC#` | Set Camera Head Comm. Baud Rate (ER-8682 only)<br>`cC0` for 4800<br>`cC1` for 9600<br>`cC2` for 19.2K<br>`cC3` for 38.4K | |
| `cn#` | Set Poll Camera (ER-8682 only)<br>`cn0` for OFF<br>`cn1` for ON | |
| `cDS` | DSP Status Request, See Section 6.5.1, page 76 for response formats | |

### 6.5.1 DSP Response Message Format

| Byte | Data | Description |
|---|---|---|
| 0 | `0xF8` | Start of message |
| 1 | Camera Address | (`0x01` to `0xDF`) |
| 2 | `c` | DSP status |
| 3 | `W` | White Balance Status |
| 4 | `W0` = Auto | |

*Continued on the next page.*

*Continued from the previous page.*

| Byte | Data | Description |
|---|---|---|
| | W1 = Manual | |
| 5 | S | Shutter Status |
| 6 | 0x30 — 0x3F | S0 = Auto<br>cS0 (Auto) cS8 (1/180)<br>cS1 (1/2) cS9 (1/250)<br>cS2 (1/4) cS: (1/500)<br>cS3 (1/8) cS; (1/1000)<br>cS4 (1/15) cS< (1/2000)<br>cS5 (1/30) cS= (1/4000)<br>cS6 (1/60) cS> (1/10000)<br>cS7 (1/120) cS@ (1/30000) |
| 7 | Y | Wide Dynamic Range Status |
| 8 | | Y0 = Off<br>Y1 = On |
| 9 | M | Day Night Mode |
| 10 | | M0 = Day/Night in Auto mode<br>M1 = in Day mode<br>M2 = in Night mode |
| 11 | D | Digital Zoom Mode |
| 12 | | D0 = Off<br>D1 = 2X<br>D2 = 5X<br>D3 = 10X |
| 13 | L | Latch status |
| 14 | 0x30 — 0x37 | LS nibble is four bits of status<br>Bit Value 0/1<br>$B_0$ = Manual Iris/Auto Iris (1/0)<br>$B_1$ = Camera Power On/Off (1/0)<br>$B_2$ = Lens Speed Fast/Slow (1/0) |

*Continued on the next page.*

*Continued from the previous page.*

| Byte | Data | Description |
|---|---|---|
| | | $B_3$ = Not Used<br>(L0, L1, L2, L3, L4, L5, L6, L7, L8) |
| 15 | A | Aux status |
| 16 | 0x30 — 0x37 | LS nibble is four bits of status<br>Bit Value 0/1<br>$B_0$ = Auto/Manual Focus (1/0)<br>$B_1$ = Auto Integration/Manual Integration (1/0)<br>$B_2$ = Not used<br>$B_3$ = Not Used<br>(A0, A1, A2, A3, A4, A5, A6, A7) |
| 17 | X | Misc. status |
| 18 | 0x30 — 0x37 | LS nibble is four bits of status<br>Bit Value 0/1<br>$B_0$ = Not Used<br>$B_1$ = Low Pressure (1)<br>$B_2$ = Video Loss (1)<br>$B_3$ = Not Used<br>(X0, X1, X2, X3, X4, X5, X6, X7) |
| 19 | H Home status | |
| 20 | 0x10 — 0x3E<br>0x60 — 0x70<br>HI | Home Position See Table 19, page 68<br>(H0, H1, H2, H3, H4, H5, H6, H7, H8, H9)<br>Preset is inactive (camera is not at a preset position) |
| 21 | P | Camera Communication Protocol |
| 22 | | P0 (Cohu)<br>P1 (American Dynamics)<br>P2 (Pelco D)<br>P3 (Javelin)<br>P4 (Philips)<br>P5 (Vicon) |
| 23 | v | Video Type |
| 24 | | v0 (NTSC)<br>v1 (PAL) |
| 25 | p | Line Lock Phase Status |
| 26 | | p0 (Camera is in Crystal mode)<br>p1 (Line Lock — Phase 0 mode — 0 degree)<br>p2 (Line Lock — Phase 1 mode — 30 degree)<br>p3 (Line Lock — Phase 2 mode — 60 degree) |

*Continued on the next page.*

*Continued from the previous page.*

| Byte | Data | Description |
|---|---|---|
| | | `p4` (Line Lock — Phase 3 mode — 90 degree)<br>`p5` (Line Lock — Phase 4 mode — 120 degree)<br>`p6` (Line Lock — Phase 5 mode — 150 degree) |
| 27 | `F` | Video Status |
| 28 | | `Fo` (Video Live mode)<br>`F0` (Video Freeze mode) |
| 29 | `C` | Camera Head Comm. Baud Rate (ER-8682) |
| 30 | | `C0` 4800<br>`C1` 9600<br>`C2` 19.2K<br>`C3` 38.4K |
| 31 | `n` | Poll Camera ON/OFF Status (ER-8682) |
| 32 | | `n0` No Poll<br>`n1` Poll |
| 33 | `s` | Progressive Scan |
| 34 | | `so` Off<br>`s0` On |
| 35 | CHECKSUM<br>End of message | `0x80` — `0x8F` Least significant nibble is XOR of all previous bytes (LS Nibbles only) except `0xF8` |

## 6.6 Sector/Zone/Touring Message Format

### 6.6.1 Sector Commands

| Command | Description | Response |
|---|---|---|
| *SC# | Clear Sector Limits<br># = Sector Number (0x30 to 0x40)<br>If 0x30 clear all, otherwise clear sector only | |
| *SX | Set Calibration Position (optional) | |
| *Sx | Get Calibration Position (optional) | *SXpppp<br>pppp = ASCII 0000 to 3599 |
| *SL#s | Set Sector Limit<br># = Sector Number (0x31 to 0x40)<br>s = "L" — Left Limit or<br>"R" — Right Limit | |
| *Sl# | Get Sector Limits<br># = Sector Number (0x31 to 0x40) | *SS#llllrrrr<br>llll and rrrr = ASCII 0000 to 3599 |

### 6.6.2 Privacy Zone Commands

| Command | Description | Response |
|---|---|---|
| *PC# | Clear Privacy Zone Limits<br># = Zone Number (0x30 to 0x38)<br>If 0x30 clear all, otherwise clear zone only | |
| *PL#s | Set Privacy Zone Limit<br># = Zone Number (0x31 to 0x38)<br>s = "L" — Left Limit or<br>"R" — Right Limit | |
| *Pl# | Get Privacy Zone Limits<br># = Sector Number (0x31 to 0x38) | *Pp#llllrrrr<br>llll and rrrr = ASCII 0000 to 3599 |
| *PS#s | Set Privacy Zone State<br># = Zone Number (0x31 to 0x38)<br>s = State ("1"=On; "0"=Off) | |
| *Ps# | Get Privacy Zone State<br># = Sector Number (0x31 to 0x38) | *PS#s<br>s = State ("1"=On; "0"=Off) |

### 6.6.3 Tour Commands

| Command | Description | Response |
|---|---|---|
| `*TC#` | Clear all preset/dwell data for Tour `#`<br>`#` = Tour Number (`0x31` to `0x38`) | |
| `*TP#<data>` | Set Tour Presets<br>`#` = Tour Number (`0x31` to `0x38`)<br>`<data>` = 32 preset values in the range `0x10` to `0x50` (i.e. 0 to 64)<br>`Note: 0x10` indicates No Preset | |
| `*Tp#` | Get Tour Presets<br>`#` = Tour Number (`0x31` to `0x38`)<br><data> = 32 preset values in the range of `0x10` to `0x50` (0 to 64) | `*TP#`<data> |
| `*TD#<data>` | Set Tour Dwell Times<br>`#` = Tour Number (`0x31` to `0x38`)<br>`<data>` = 32 dwell times in the range `0x10` to `0x4D` (0 to 60 sec) | |
| `*Td#` | Get Tour Dwell Times | `*TD#`<data><br>`#` = Tour Number (`0x31` to `0x38`)<br><data> = 32 dwell times in the range of `0x10` to `0x4D` (0 to 60 sec) |
| `*TS#s` | Set Tour State<br>`#` = Tour Number (`0x31` to `0x38`)<br>`s` = State (“`0`”=Stop; “`1`”=Start) | |
| `*Ts#` | Get Tour State<br>`#` = Sector Number (`0x31` to `0x38`)<br>`s` = State (“`0`”=Stop; “`1`”=Start) | `*TS#s` |

## 6.7 Position

This section describes how to GET the position/presets data and GOTO a position of the camera.

### 6.7.1 Standard Position Response Data

| Command | Description | Response |
|---|---|---|
| P? | Pan/Tilt Position Request | Ppppttt<br>ppp = 12 bit data of Pan (0x000 to 0xFFF)<br>Format: 0x303030 — 0x3F3F3F |
| | | ttt = 12 bit data of Tilt (0x000 — 0xFFF)<br>Format: 0x303030 — 0x3F3F3F |
| V? | Zoom/Focus Position Request | Pzzzfff<br>zzz = 12 bit data of Zoom (0x000 to 0xFFF)<br>Format: 0x303030 to 0x3F3F3F |
| | | fff = 12 bit data of Focus (always 0x000)<br>Format: 0x303030 |

### 6.7.2 Standard Goto A Position

| Command | Description | Response |
|---|---|---|
| paaabbb | Goto a Pan/Tilt Position<br>aaa = 12 bit data of Pan (0x000 to 0xFFF)<br>Format: 0x303030 — 0x3F3F3F | |
| | bbb = 12 bit data of Tilt (0x000 — 0xFFF)<br>Format: 0x303030 — 0x3F3F3F | |
| vcccddd | Goto Zoom/Focus Position<br>ccc = 12 bit data of Zoom (0x000 to 0xFFF)<br>Format: 0x303030 to 0x3F3F3F | |
| | ddd = 12 bit data of Focus (always 0x000)<br>Format: 0x303030 | |

### 6.7.3 Actual Position Response Data

| Command | Description | Response |
|---|---|---|
| cPS | Pan/Tilt Position Request | cPppppttttt<br>pppp = Pan data (0000 to 3599 in ASCII)<br>tttt = Tilt data (0000 to 1220 in ASCII) |

### 6.7.4 Actual Goto A Position

| Command | Description | Response |
|---|---|---|
| cpaaaabbbb | Goto a Pan/Tilt Position<br>aaaa = Pan data (0000 to 3599 in ASCII)<br>bbbb = Tilt data (0000 to 1220 in ASCII) | |

### 6.7.5 Get Actual Presets Position Data

| Command | Description | Response |
|---|---|---|
| `*pp##` | Get Preset data stored in camera<br>`##` = `0x10` to `0x3E` (first 48 positions)<br>`##` = `0x60` to `0x70` (last 16 positions) | `pP##ppppttttzzzzffff` See Table 19, page 68<br>`pppp` = Pan Preset Position data<br>`tttt` = Tilt Preset Position data<br>`zzzz` = Zoom Preset Position data<br>`ffff` = Focus Preset Position data |

### 6.7.6 Position Encoding

The message data for the "Pan/Tilt Goto Command" above begins with an ASCII '`p`' followed by the azimuth ($A_2A_1A_0$) and elevation ($E_2E_1E_0$) positions. The positions are 12-bit values encoded four-bits each in the least significant nibbles of the three bytes whose most significant nibbles are always `0x30`. The subscript-2 indicates the byte containing the most significant bytes of the position and subscript-0 indicates the least significant.

#### 6.7.6.1 Position Encoding Example:

(*From the GA DOT document.*)

Encoding an azimuth position value of decimal $2748_{10}$ or `0xABC`, we break it up into three nibbles and add `0x30` to each so:

$A_2$ = `0x3A`,
$A_1$ = `0x3B`, and
$A_0$ = `0x3C`.

The same encoding scheme is used for the zoom and focus positions in the "Zoom/Focus Goto Command".

### 6.7.7 Pan/Tilt Positions

Pan readouts are based on the $CAL_0$ (North or $0^o$) value of the PTZ. Angular readouts then increment to $90^o$ (East), $180^o$ (South) and to $270^o$ (West) when looking down on the PTZ unit from above.

Tilt readouts are based on the horizon, with $0^o$ meaning that the PTZ is looking at the horizon. As the PTZ points down the protocol readout changes to $45^o$ and then $90^o$ when the PTZ is pointing straight down. The PTZ on-screen display will indicate - $45^o$ and then -$90^o$. The discrepancy in the two types of readouts is that the protocol value is unsigned and the on-screen is signed. As tilt passes through straight down (not physically possible, but conceptually convenient, the protocol values increment until a straight up readout is $270^o$ and the on-screen display shows $90^o$. At $45^o$ above the horizon the protocol readout will be $315^o$ and the on-screen display will be $45^o$. Here is a short table to attempt and explain these values.

| Position | D readouts | Displays as |
|---|---|---|
| $90^o$ up | 270 | 90 |
| $45^o$ up | 315 | 45 |
| Horizontal - $1^o$ | 359 | 1 |
| Horizontal | 0 | 0 |
| $45^o$ down | 45 | -45 |
| $90^o$ down | 90 | -90 |

**Note**

1. **Position:** Pointing direction of the enclosure/camera.
2. **D readouts:** D protocol returned value for this angle.
3. **Displays as:** What is displayed on the Spectra screen.

## 6.8 Upload Camera Configuration to Camera

This section describes how to upload camera configuration data to the camera. For example, camera IDs, Presets position, Tour/Sector position, etc.

| Command | Description |
|---|---|
| dC# | Upload ID Control Data, i.e Enable, Top, Alarm Blink ... to camera<br># = Control Byte(xxx11111) |
| dL#<DATA> | Upload ID Messages<br># = "1", "2", "3", "4", "5"<br>DATA = 24 ASCII characters |
| dP##<DATA> | Upload Preset ## ID to camera See Table 19, page 68<br>## = 0x10 to 0x3E (first 48 positions)<br>## = 0x60 to 0x70 (last 16 positions) |
| dST##<DATA> | Upload Sector ## ID to camera<br>## = 0x31 to 0x40 (16 sectors) |
| dSS##s | Upload Sector ID State to camera<br>## = Sector Number (0x31 to 0x40)<br>s = State ("1"=On; "0"=Off) |
| *SS#llllrrrr | Upload Sector Limits to camera<br># = Sector Number (0x31 to 0x40)<br>llll and rrrr = ASCII 0000 to 3599 |
| *Pp#llllrrrr | Upload Privacy Zone Limits to camera<br># = Sector Number (0x31 to 0x38)<br>llll and rrrr = ASCII 0000 to 3599 |
| *PS#s | Upload Privacy Zone State to camera<br># = Sector Number (0x31 to 0x38)<br>s = State ("1"=On; "0"=Off) |
| *TP#<data> | Upload Tour Presets to camera<br># = Tour Number (0x31 to 0x38)<br><data> = 32 preset values in the |
| *TD#<data> | Upload Tour Dwell Times to camera<br># = Tour Number (0x31 to 0x38)<br><data> = 32 dwell times in the range of<br>0x10 to 0x4D (0 to 60 sec) |
| *TS#s | Upload Tour State to camera<br># = Sector Number (0x31 to 0x38)<br>s = State ("0"=Stop; "1"=Start) |
| *pP##pppptttzzzzffff | Upload Presets Data to camera |

*Continued on the next page.*

*Continued from the previous page.*

| Command | Description |
|---|---|
| | `##` = `0x10` to `0x3E` (first 48 positions)<br>`##` = `0x60` to `0x70` (last 16 positions)<br>`pppp` = Pan Preset Position data<br>`tttt` = Tilt Preset Position data<br>`zzzz` = Zoom Preset Position data<br>`ffff` = Focus Preset Position data |

# 7 Cohu 3500 DSP Camera, RS-232 Interface

rev 1.15. June 01 1998

## 7.1 General

This document[22] defines the Asynchronous communication protocol between the PC and a 3500 DSP Camera. The communication is a full duplex. Typical or default communication parameters will be 9600 baud, 8 data bits, 1 stop bit and no parity.

## 7.2 Message format

Anytime a command message is sent to the Camera the camera will respond with an ACK or NAK. The commands are sent using the command message format show in Table 20, page 88. The various commands that may be transmitted are listed in Table 21, page 89 ⟶ Table 32, page 93. The response message from the camera is shown in Table 33, page 94 ⟶ Table 43, page 104.

If the command message contains only a momentary function, no further response beyond the ACK or NAK will occur.

The control characters to be used are:

- STX Start of text or (`0xF8` if RS-422/RS-485)
- ACK Acknowledge proper receipt of transmission
- NAK Bad transmission — retransmit
- ETX End of Message (not used for RS-422)
- CS Checksum
- ID Camera address

All transmissions will be in this format:

| STX | ID | "c" | DATA BYTES | CHECKSUM | ETX |
|---|---|---|---|---|---|

**Note**

1. "`Wa`" means ASCII characters for `W` and `a`.
2. **Since all commands are to have a "`c`" in byte position 2 (after the Camera Address and before the Command Data), a `c` has been added into each command ID.**

NOTE: $n$ is the number of command data bytes in the message.

[21] `$Header: d:/TXB-Cohu/RCS/3500pro.inc,v 1.9 2006-08-30 12:31:56-07 Hamilton Exp Hamilton $`
[22] This document was derived from the Cohu document: "3500pro.pdf".

| Byte | Data | Description |
|---|---|---|
| 0 | STX | Start of message (0x02) |
| 1 | Address | Camera address (0x01 — 0xDF) |
| 2 | "c" | Message type (camera) |
| 3 to $n+3$ | Command Data | See Table 21, page 89 through Table 32, page 93 |
| $n+4$ | CHECKSUM | 0x80 — 0x8F Least significant nibble is XOR of all previous bytes (LS Nibbles only) except STX |
| $n+5$ | ETX | End of message (0x03) |

Table 20. RS-232 Command Message Format

### 7.2.1 RS-232 Command Data

| Message | Data | Description | Response |
|---|---|---|---|
| WB Status Request | "cWS" | | See Table 33, page 94 |
| WB Sample/Hold | "cWa" | | ACK/NAK |
| WB auto tracking | "cWt" | | ACK/NAK |
| WB manual | "cWm" | | ACK/NAK |
| WB red gain | "cWr###" | "000" ⟶ "895" | ACK/NAK |
| WB green gain | "cWg###" | "000" ⟶ "895" | ACK/NAK |
| WB blue gain | "cWb###" | "000" ⟶ "895" | ACK/NAK |
| Hue | "cWh###" | "000" ⟶ "063" | ACK/NAK |
| Saturation | "cWs###" | "000" ⟶ "255" | ACK/NAK |
| Indoor Lighting | "cWi" | | ACK/NAK |
| Outdoor Lighting | "cWo" | | ACK/NAK |

Table 21. White Balance Commands

| Message | Data | Description | Response |
|---|---|---|---|
| Sync Status Request | "cSS" | | See Table 34, page 95 |
| Horizontal Phase | "cSc#" | 0x30 ⟶ 0x3F 16 steps | ACK/NAK (cSc0, cSc1, cSc2, cSc3, cSc4, cSc5, cSc6, cSc7, cSc8, cSc9, cSc:, cSc;, cSc<, cSc=, cSc>, cSc@) |
| Sub-Carrier phase | "cSp###" | "000" ⟶ "127" | ACK/NAK |
| Sync line lock | "cSl#" | "0" ⟶ "5" 6 steps | ACK/NAK (cSl0, cSl1, cSl2, cSl3, cSl4, cSl5) |
| Sync mode | "cSm#" | "0" ⟶ "4" | ACK/NAK (cSm0, cSm1, cSm2, cSm3, cSm4) |

Table 22. Synchronize Modes Commands

Note: These commands depend on the sync board specified
Sync mode

"0" = GenLock,
"1" = H/V,
"2" = vert reset,
"3" = LineLock and
"4" = Crystal.

| Message | Data | Description | Response |
|---|---|---|---|
| Sht/Int Status Request | "cVS" | | See Table 35, page 96 |
| Shutter Automatic | "cVa" | | ACK/NAK |
| Shutter Manual | "cVm" | | ACK/NAK |
| Shutter speed | "cVs#" | "0" ⟶ "7" 8 steps | ACK/NAK (cVs0, cVs1, cVs2, cVs3, cVs4, cVs5, cVs6, cVs7, cVs8) |
| Integration Auto | "cVA" | | ACK/NAK |
| Integration manual | "cVM" | | ACK/NAK |
| Integration external | "cVE" | | ACK/NAK |
| Integration time | "cVi##" | "01" ⟶ "99" frames | ACK/NAK |

Table 23. Shutter/Integration Commands

| Message | Data | Description | Response |
|---|---|---|---|
| Gain Status request | "cGS" | | See Table 36, page 97 |
| Automatic gain | "cGa" | | ACK/NAK |
| Manual gain | "cGm" | | ACK/NAK |
| Manual gain level | "cGg###" | "000" ⟶ "126" | ACK/NAK |

Table 24. Gain Commands

| Message | Data | Description | Response |
|---|---|---|---|
| ID Status Request | "cTS" | | See Table 37, page 98 |
| Enable ID text | "cTE" | | ACK/NAK |
| Disable ID text | "cTD" | | ACK/NAK |
| Top of screen | "cTT" | | ACK/NAK |
| Bottom of screen | "cTB" | | ACK/NAK |
| Text up to 24 chars | "cTL" | | ACK/NAK |

Table 25. Operator"s ID Commands

| Message | Data | Description | Response |
|---|---|---|---|
| Iris status request | "cIS" | | See Table 38, page 99 |
| Automatic Iris | "cIa" | | ACK/NAK |
| Iris Manual | "cIm" | | ACK/NAK |
| DC Iris Auto | "cIA" | | ACK/NAK |
| DC Iris manual | "cIM" | | ACK/NAK |
| Iris open | "cIo" | | ACK/NAK |
| Iris close | "cIc" | | ACK/NAK |
| Iris stop | "cIs" | | ACK/NAK |

Table 26. Iris Commands

| Message | Data | Description | Response |
|---|---|---|---|
| Zoom Status request | "cZS" | | See Table 39, page 100 |
| Zoom magnification factor | "cZm#" # | cZm1 = x1,<br>cZm2 = x2,<br>cZm4 = x4,<br>cZm8 = x8 | ACK/NAK |
| Zoom in | "cZi" | | ACK/NAK |
| Zoom out | "cZo" | | ACK/NAK |
| Zoom stop | "cZs" | | ACK/NAK |
| Scroll up | "cZu" | | ACK/NAK |
| Scroll down | "cZd" | | ACK/NAK |
| Scroll right | "cZr" | | ACK/NAK |
| Scroll left | "cZl" | | ACK/NAK |
| Scroll stop | "cZq" | | ACK/NAK |
| Scroll center | "cZc" | | ACK/NAK |
| Zoom ctrl Boundary | "cZB" | | ACK/NAK |

Table 27. Zoom/Scroll Commands

| Message | Data | Description | Response |
|---|---|---|---|
| Upload Configurations | "cCG#" | 0x30 ⟶ 0x39 | See Table 42, page 103 (cCG0, cCG1, cCG2, cCG3, cCG4, cCG5, cCG6, cCG7, cCG8, cCG9) |
| DownLoad Configurations | "cCP#",parameters | 0x30 ⟶ 0x39 | ACK/NAK (cCP0, cCP1, cCP2, cCP3, cCP4, cCP5, cCP6, cCP7, cCP8, cCP9) |

Table 28. Configuration of Camera

Note: parameters — see defination in Table 30, page 93.

| Message | Data | Description | Response |
|---|---|---|---|
| Camera Format status request | "cFS" | | See Table 40, page 101 |
| Camera Format | "cFC##" | statusF1, statusF2A | ACK/NAK |

Table 29. Format of Camera

Note: statusF1, statusF2 — see defination in Table 40, page 101

| Message | Data | Description | Response |
|---|---|---|---|
| Misc. Status Request | "MS" | | See Table 41, page 102 |
| Backlight comp OFF | "Bf" | | ACK/NAK |
| Backlight comp ON | "Bn" | | ACK/NAK |
| Pedestal ON | "On" | | ACK/NAK |
| Pedestal OFF | "Off" | | ACK/NAK |
| Y Gamma | "Yy#" | "0" ⟶ "7" 8 steps | ACK/NAK (Yy0, Yy1, Yy2, Yy3, Yy4, Yy5, Yy6, Yy7) |
| Aperture Correction | "A##" | "00" ⟶ "31" | ACK/NAK |
| Av. Picture Lv (APL) | "L##" | "00" ⟶ "99" | ACK/NAK |
| Contrast | "c##" | "00" ⟶ "63" | ACK/NAK |
| Black Level | "D###" | "000" ⟶ "255" | ACK/NAK |
| Date & Revision num | "RS" | | See Table 44, page 105 |

Table 30. Miscellaneous Commands

| Message | Data | Description | Response |
|---|---|---|---|
| Store Presets | "cPs#" | "0" ⟶ "9" | ACK/NAK (cPs0, cPs1, cPs2, cPs3, cPs4, cPs5, cPs6, cPs7, cPs8, cPs9) |
| Recall Presets | "cPr#" | "0" ⟶ "9" | ACK/NAK (cPr0, cPr1, cPr2, cPr3, cPr4, cPr5, cPr6, cPr7, cPr8, cPr9) |
| Factory Defaults | "cPd" | | ACK/NAK |

Table 31. Preset Commands

| Message | Data | Description | Response |
|---|---|---|---|
| Window frontlight Status | "cwFS" | | See Table 43, page 104 |
| Window defaults Status | "cwFD" | | See Table 43, page 104 |
| Window HV weights | "cwX", 0x01 ⟶ 0x05, v1 ⟶ v5, w1 ⟶ w6 | X = "F" for front light and "B" for backlight coefficient | ACK/NAK |

Table 32. Windows Areas Weights Commands

## 7.3 Response Message Format

| Byte | Data | Description |
|---|---|---|
| 0 | STX | Start of message |
| 1 | ID | Camera address |
| 2 | "c" | Message type |
| 3 | "W" | (cW) WB |
| 4 | StatusWB | WB status |
| 5-7 | Rgain | Red gain "000" ⟶ "895" |
| 8-10 | Ggain | Green gain "000" ⟶ "895" |
| 11-13 | Bgain | Blue gain "000" ⟶ "895" |
| 14-16 | Hue | Hue "000" ⟶ "063" (RGain & BGain) |
| 17-19 | Saturation | Saturation "000" ⟶ "255" (BYGain) |
| 20 | CHECKSUM | 0x80 — 0x8F LS nibble is XOR of all previous bytes (LS Nibbles only) except STX |
| 21 | ETX | End of message (0x03) |

Table 33. White Balance Status

StatusWB byte

- statusWB 0x30 - 0x38
  - bit0 0 = Tracking 1 = manual
  - bit1 0 = indoor 1 = outdoor
  - bit2 not defined always 0
  - bit3 0 = no error 1 = WB error
- cW################## One 1 byte field and five 3 byte fields (cW#,###,###,###,###,###).

| Byte | Data | Description |
|---|---|---|
| 0 | STX | Start of message |
| 1 | ID | Camera address |
| 2 | "c" | Message type |
| 3 | "S" | (cS) |
| 4 | StatusSM | Sync Board status |
| 5 | 0x30 ⟶ 0x3F | Horizontal delay 16 steps |
| 6 | 0x30 ⟶ 0x35 | Line Lock AC Phase. |
| 7-9 | "000" ⟶ "127" | Sub carrier phase (0 ⟶ 353 degree) |
| 10 | CHECKSUM | 0x80 — 0x8F Least significant nibble is XOR of all previous bytes (LS Nibbles only) except STX |
| 11 | ETX | End of message (0x03) |

Table 34. Sync Status

StatusSM Byte

- statusSM 0x30 - 0x34
  - 0x30 = GenLock (Auto 1)
  - 0x31 = H and V (Auto 2 )
  - 0x32 = Vertical Reset
  - 0x33 = LineLock
  - 0x34 = Crystal
- Line lock phase @60 degree interval
  - 0x30 = 300 degree
  - 0x31 = 240
  - 0x32 = 180
  - 0x33 = 120
  - 0x34 = 60
  - 0x35 = 0 degree
- cS##### Two 1 byte fields and one 3 byte field (cS#,#,###).

| Byte | Data | Description |
|---|---|---|
| 0 | STX | Start of message |
| 1 | ID | Camera address |
| 2 | "c" | Message type |
| 3 | "V" | (cS) |
| 4 | StatusSI | Shutter/Integration mode |
| 5 | "0" ⟶ "8" | Shutter speed |
| 6-7 | "01" ⟶ "99" | Integration time in # of frames |
| 8 | CHECKSUM | 0x80 — 0x8F Least significant nibble is XOR of all previous bytes (LS Nibbles only) except STX |
| 9 | ETX | End of message (0x03) |

Table 35. Shutter/Integration Status

Shutter/Integration Byte statusSI

- statusSI 0x30 - 0x37
  - 0x30 = Auto shutter
  - 0x31 = Manual Shutter
  - 0x32 = Manual Integration
  - 0x33 = Auto Integration
  - 0x34 = External Integration
- Shutter Speed byte

| | "0" | "1" | "2" | "3" | "4" | "5" | "6" | "7" |
|---|---|---|---|---|---|---|---|---|
| NTSC | 1/60 | 1/100 | 1/250 | 1/500 | 1/1000 | 1/2000 | 1/4000 | 1/8000 |
| PAL | 1/50 | 1/120 | 1/250 | 1/500 | 1/1000 | 1/2000 | 1/4000 | 1/8000 |

- cV#### Two 1 byte fields and one 2 byte field (cV#,#,##).

| Byte | Data | Description |
|---|---|---|
| 0 | STX | Start of message |
| 1 | ID | Camera address |
| 2 | "c" | Message type (Camera) |
| 3 | "G" | (cG) |
| 4 | StatusGM | Gain Status |
| 5-7 | "000" ⟶ "126" | Gain value |
| 8 | CHECKSUM | 0x80 — 0x8F Least significant nibble is XOR of all previous bytes (LS Nibbles only) except STX |
| 9 | ETX | End of message (0x03) |

Table 36. Gain Status

statusGM

- statusGM 0x30 - 0x31
  - bit0 0 = AGC 1 = manual
- cF#### One 1 byte field and one 3 byte field (cF#,###).

| Byte | Data | Description |
|---|---|---|
| 0 | STX | Start of message |
| 1 | ID | Camera address |
| 2 | "c" | Message type (camera) |
| 3 | "T" | (cT) |
| 4 | statusID | Status of ID message |
| 5-29 | [1-24] | text |
| 30 | CHECKSUM | 0x80 — 0x8F Least significant nibble is XOR of all previous bytes (LS Nibbles only) except STX |
| 31 | ETX | End of message (0x03) |

Table 37. Operator"s ID Status

Operator ID Status Byte (statusID)

- statusID 0x30 - 0x33
  - 0x30 ID disable and dpy @ bottom of screen
  - 0x31 ID enable and dpy @ bottom of screen
  - 0x32 ID disable and dpy @ top of screen
  - 0x33 ID enable and dpy @ top of screen
- cT#<24 ASCII> One 1 byte field and 24 ASCII characters.

| Byte | Data | Description |
|---|---|---|
| 0 | STX | Start of message |
| 1 | ID | Camera address |
| 2 | "c" | Message type (camera) |
| 3 | "I" | (cI) |
| 4 | StatusIR | Status of Iris<br>(cI0, cI1, cI2, cI3, cI4, cI5) |
| 5 | CHECKSUM | 0x80 — 0x8F Least significant nibble is XOR of all previous bytes (LS Nibbles only) except STX |
| 6 | ETX | End of message (0x03) |

Table 38. Iris Status

Lens Iris Status Byte statusIR

- statusIR 0x30 - 0x35
  - 0x30 Manual Iris
  - 0x31 Auto Iris
  - 0x32 DC Iris Manual
  - 0x33 DC Iris Auto
  - 0x34 Motorize Iris manual
  - 0x35 Motorize Iris auto

| Byte | Data | Description |
|---|---|---|
| 0 | STX | Start of message |
| 1 | ID | Camera address |
| 2 | "c" | Message type (camera) |
| 3 | "Z" | (cZ) |
| 4 | 0x30 ⟶ 0x68 | Zoom magnification factor |
| 5 | CHECKSUM | 0x80 — 0x8F Least significant nibble is XOR of all previous bytes (LS Nibbles only) except STX |
| 6 | ETX | End of message (0x03) |

Table 39. Zoom Status

cZ<ASCII> <ASCII> is a single ASCII character which ranges from 0 through h to give a range of $57_{10}$ zoom values.

| Byte | Data | Description |
|---|---|---|
| 0 | STX | Start of message |
| 1 | ID | Camera address |
| 2 | "c" | Message type |
| 3 | "F" | (cF) |
| 4 | StatusF1 | Camera statusF1 |
| 5 | StatusF2 | Camera statusF2 |
| 6 | CHECKSUM | 0x80 — 0x8F Least significant nibble is XOR of all previous bytes (LS Nibbles only) except STX |
| 7 | ETX | End of message (0x03) |

Table 40. Status

Format Status Byte (statusF1)

- statusF1 0x30 - 0x34
  - 0x30 NTSC color
  - 0x31 PAL color
  - 0x32 NTSC Black and White
  - 0x33 PAL black and white
- Format Status Byte (statusF2)
  - statusF2 undefined (0x30)
- cF## Two 1 byte fields (cF#,# or cF00, cF10, cF20, cF30, the second field is undefined and is always 0.)

| Byte | Data | Description |
|---|---|---|
| 0 | STX | Start of message |
| 1 | ID | Camera address |
| 2 | "c" | Message type |
| 3 | "M" | (cM) |
| 4 | statusM1 | Misc status 1 |
| 5 | statusM2 | Misc status 2 Gamma "0" ⟶ "7" |
| 6-7 | statusM3 | Misc status 3 Aperture "00" ⟶ "31" |
| 8-9 | statusM4 | Misc status 4 Av. Picture Lv "00" ⟶ "99" |
| 10-11 | statusM5 | Misc status 5 Contrast "00" ⟶ "63" |
| 12-14 | statusM6 | Misc status 6 Black level "000" ⟶ "255" |
| 15 | CHECKSUM | 0x80 — 0x8F Least significant nibble is XOR of all previous bytes (LS Nibbles only) except STX |
| 16 | ETX | End of message (0x03) |

Table 41. Miscellaneous Status

- Misc Status Byte (statusM1)
  - statusM1 0x30 - 0x33
    - bit0 0 = backlight OFF, 1 = backlight compensation ON
    - bit1 0 = pedestal OFF, 1 = pedestal ON (7.5 IRE) only for NTSC
- Misc Status Byte (statusM2) Gamma "0" ⟶ "7"
- Misc Status Word (statusM3) Aperture correction value "00" ⟶ "31"
- Misc Status word (statusM4) Av. Picture Level "00" ⟶ "99"
- Misc status (statusM5) Contrast value "00" ⟶ "63" (YFADE)
- Misc status (statusM6) Black level value "000" ⟶ "255"
- cM########### Two 1 byte fields, three 2 byte fields and one 3 byte field (cM#,#,##,##,##,###)

| Byte | Data | Description |
|---|---|---|
| 0 | STX | Start of message |
| 1 | ID | Camera address |
| 2 | "c" | Message type (Camera) |
| 3 | "C" | (cC) |
| 4 | "G" (cCG) or "P" (cCP) | "G" = get configs (upload) or "P" = put config (download) |
| 5 | 0x30 ⟶ 0x37 | 0x30 = default 0x31 ⟶ 0x37 configuration group number |
| 6-80 | Parameters | configuration parameters of camera |
| 81 | CHECKSUM | 0x80 — 0x8F Least significant nibble is XOR of all previous bytes (LS Nibbles only) except STX |
| 82 | ETX | End of message (0x03) |

Table 42. Upload/DownLoad Configuration

Parameters:

| Item | Bytes | Description |
|---|---|---|
| statusWB | 1 | WB status byte |
| param1 | 15 | R,G,B gains, Hue and Saturation |
| param2 | 12 | Window weights (6 bytes) and sizes (6 bytes) |
| statusSI | 1 | Shutter/Integration status byte |
| param3 | 3 | Shutter Mode (1 byte) and Integration time (frcnt = 2 bytes) |
| statusGM | 1 | Gain status byte |
| param4 | 3 | Gain level |
| statusID | 1 | Operator"s ID status byte |
| OperID | 24 | 24 bytes Operator"s ID |
| param5 | 1 | Zoom magnification factor |
| param6 | 2 | Format statusF1 and statusF2 |
| statusM1 | 1 | Miscellaneous status 1 |
| param7 | 10 | Gamma, Aperture, APL, contrast and black level |
| Total | 75 bytes | |

Note: See status response message formats for White balance, Sync, Shutter/Integration, Gain, Operator"s ID, Iris, Format and Misc. for details of data.

- param2 = 6 bytes
- weights 6 bytes sizes (horizontal h2, h3, h4, h5 and vertical dimensions V3 & V4)

cDG<75 bytes of numeric and ASCII data> See above descriptions.

| Byte | Data | Description |
|---|---|---|
| 0 | STX | Start of message |
| 1 | ID | Camera address |
| 2 | "c" | Message type (Camera) |
| 3 | "w" | (cw) |
| 4 | "F" (cwF) or "D" (cwD) | "F" = frontlight data or "D" = default weights/sizes data |
| 5-9 | horizParam | horizontal count values h1 ⟶ h5 |
| 10-14 | vertiParam | vertical count values v1 ⟶ v5 |
| 15-20 | weights | weights count values w1 ⟶ w6 |
| 21 | CHECKSUM | 0x80 — 0x8F Least significant nibble is XOR of all previous bytes (LS Nibbles only) except STX |
| 22 | ETX | End of message (0x03) |

Table 43. Window Status

Parameters

- horizParam 5 bytes Horizontal count values
- vertParam 5 bytes Vertical count values
- weights 6 bytes weights for 6 areas
- cwD################ Two 5 byte fields and one 6 byte field (cwD#####,#####,######)
- cwF################ Two 5 byte fields and one 6 byte field (cwF#####,#####,######)

| Byte | Data | Description |
|---|---|---|
| 0 | STX | Start of message |
| 1 | ID | Camera address |
| 2 | "c" | Message type (Camera) |
| 3 | "R" | (cR) Date and Revision number status |
| 4-5 | month | month of the year |
| 6 | "/" | slash |
| 7-8 | day | day of the month |
| 9 | "/" | slash |
| 10-11 | year | year — 98 |
| 12-15 | revision | revision number e.g. 02.0 |
| 16 | CHECKSUM | 0x80 — 0x8F Least significant nibble is XOR of all previous bytes (LS Nibbles only) except STX |
| 17 | ETX | End of message (0x03) |

Table 44. Date and Revision of Camera Software Status

Parameters
cR##/##/####.# One 6 byte field, slant delimited, one four byte field with included period "(.)" (cR##/##/##,###)

This page intentionally left blank

# 8 MPC System ER-2221B Receiver Communications Protocol

This interface[24] also applies to ER-2221AK and other ER versions.

## 8.1 General Description

The[25] MPC Master Control Panel contains an 8031 micro computer with on-board UART for communications with MPC Control Receivers.

This section supplies the details necessary to understand communications between the MPC Master Control Panel (or Host Computer) and Control Receivers as used in the camera control system (Figure 8, page 108).

## 8.2 Message Format

Any time a command message is sent from the MPC Master Control Panel (or Host Computer) to a Control Receiver, the Receiver will respond with an ACK or an NAK (Figure 8, page 108). Data is transmitted using 1 start bit, 8 data bits, and 1 stop bit (no parity). The baud rate is set us ing the monitor 97 function as an entry code. It is typically set to 9600 baud. This process is described in the installation instructions of the Control Panel manual.

The commands are sent using the command message format show in Table 45, page 109. The various commands that may be transmitted are listed in Table 46, page 110 through Table 48, page 111.

If the command message contains a latch function command, then the latch status response (Table 49, page 112, Format of Latch Status Response) will be sent back to the Master Control Panel. If the command messsage contains a home function command, then the home status response (Table 50, page 113, Format of Home Status Response from MPC) will be sent back to the Master Control Panel.

If the command is a position message, the response will be as shown in Table 51, page 114 and Table 51, page 114.

If the command message contains only a momentary function, no further response beyond the ACK or NAK will occur. If the command message contains a communications error (e.g., an address to an non-existent control receiver or a failure in the communications system), then the command error message response (in Table 50, page 113, with $B_3$ of byte 3 equal to 1) will be sent back.

Table 54, page 116 through Table 59, page 118 gives the ID message formats for communications from the master control panel to a control receiver. An ACK or NAK is returned.

## 8.3 Maintenance

No maintenance adjustments are required for the circuits on the processor board.

## 8.4 Parts List

Refer to the processor board for parts identifications.

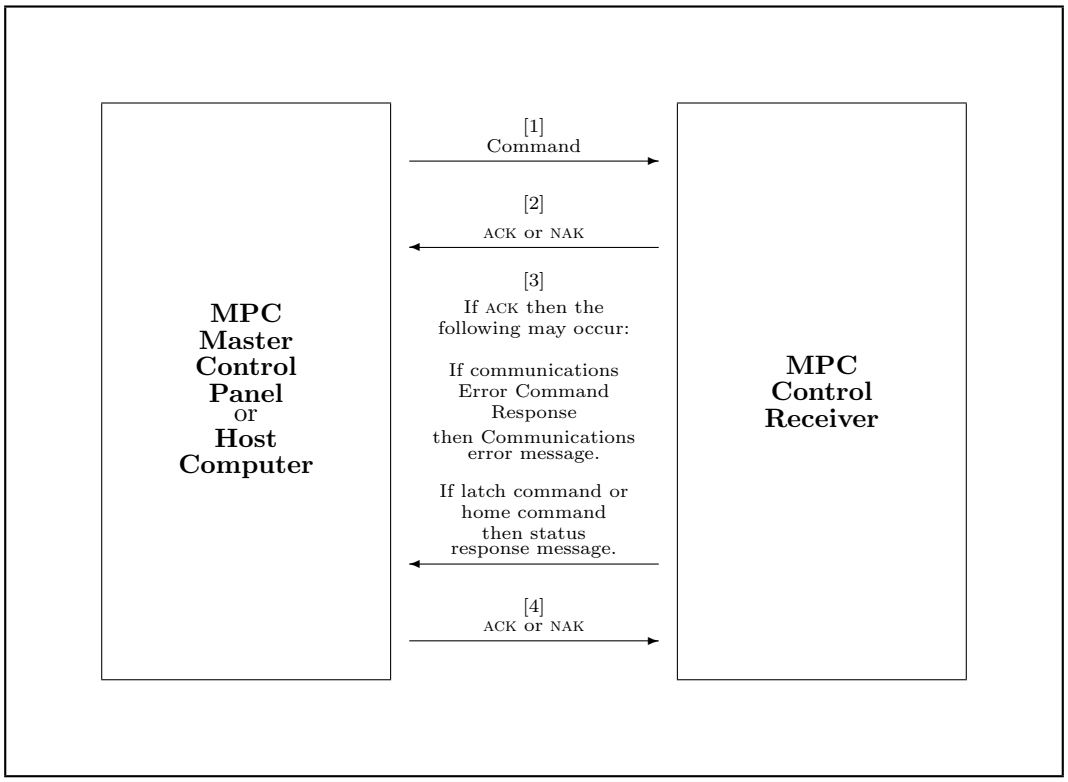
## 8.5 Schematic Diagrams

Refer to the processor board for the schematic diagram.

[23] `$Header: d:/TXB-Cohu/RCS/2221pro.inc,v 1.10 2006-08-29 10:15:17-07 Hamilton Exp Hamilton $`

[24] Cohu"s data: Phone (619) 277- 6700, FAX (619) 277-0221, December 7, 1998, E-mail `info@cohu.com`

[25] This is derived from the Cohu document: "mpc2221pro.pdf", Cohu document number 6X-5046(A).




$RCSfile: TCP.inc,v $

Figure 8. Typical Communications Protocol

## 8.6 Receiver Command Message Format

| Byte | Data | Description |
|---|---|---|
| 0 | 0xF8 | Auto baud character |
| 1 | Address | Camera address (0x01 0xDF) |
| 2 TO $2n+1$ | Command Data | See Table 46, page 110 through Table 48, page 111 |
| $2n+2$ | CHECKSUM | 0x80 0x8F Least significant nibble is XOR of all previous bytes (LS Nibble only) except 0xF8 |

Table 45. Receiver Command Message Format

NOTE: $n$ is the number of commands in the message

| Byte | Data | Description |
|---|---|---|
| Reset Control Receiver | "rs" | DATA consists of the ASCII codes for the two |
| Pan Left | "PL" | letters shown: rs = ASCII codes for r and s |
| Pan Right | "PR" | PL = ASCII codes for P and L, etc. |
| Pan Stop | "PS" | |
| Tilt Up | "TU" | |
| Tilt Down | "TD" | |
| Tilt Stop | "TS" | |
| Zoom In | "ZI" | |
| Zoom Out | "ZO" | |
| Zoom Stop | "ZS" | |
| Focus Near | "FN" | |
| Focus Far | "FF" | |
| Focus Stop | "FS" | |
| Iris Open | "IO" | |
| Iris Close | "IC" | |
| Iris Stop | "IS" | |
| P/T Position Request | "P?" | See Table 51, page 114 and Table 52, page 114 for position format |
| P/T GoTo Command | "p$A_2A_1A_0E_2E_1E_0$" | |
| Lens Position Request | "V?" | |
| Lens GoTo Command | "v$A_2A_1A_0E_2E_1E_0$" | |

Table 46. Momentary, Receiver Command Data

| Byte | Data | Description |
|---|---|---|
| Manual Iris Toggle | "LM" | See Table 49, page 112 and Table 50, page 113 for response format |
| Camera Power Toggle | "LP" | |
| Lens Speed Toggle | "LL" | |
| Latch Status Request | "L?" | |
| Aux Functions (Option 1) | "L1" $\longrightarrow$ "L3" | Toggles Aux 1, Aux 2, or Aux 3 (L1, L2, L3) |
| Color Balance (Option 2) | "L1" | Select auto/manual mode |
| | "B1" | Increase blue |
| | "B2" | Increase red |
| | "B0" | Balance stop |

Table 47. Latch, Receiver Command Data

| Byte | Data | Description |
|---|---|---|
| Home Position GoTo | "H0" ⟶ "H9" | Preset 0 ⟶ 9<br>A home active "HA" will be immediate response.<br>Later a home position "H0" to "H9" or home error "HE" will be generated.<br>(H0, H1, H2, H3, H4, H5, H6, H7, H8, H9) |
| Home Position Store | "P0" ⟶ "P9" | Program 0 ⟶ 9<br>(P0, P1, P2, P3, P4, P5, P6, P7, P8, P9) |
| Home Position | "H?" | Will generate a home status response |
| Status Request | | See Table 49, page 112 and Table 50, page 113 |

Table 48. Home, Receiver Command Data

## 8.7 Responses from MPC Control Receiver

| Byte | Data | Description |
|---|---|---|
| 0 | 0xF8 | Auto baud character |
| 1 | Address | Camera address 0x01 0xDF |
| 2 | "L" | Latch Status |
| 3 | 0x30 ⟶ 0x37<br><br>0x38 | LS Nibble is four bits of status<br>(L0, L1, L2, L3, L4, L5, L6, L7, L8)<br>Bit Value 0/1 (Comm error)<br>$B_0$ =Iris Auto Manual<br>$B_1$ =Camera power Off/On<br>$B_2$ =Lens Speed Slow/Fast<br>$B_3$ =Communications Error No/Yes |
| 4 | "A" | Aux Status |
| 5 | 0x30 — 0x37 | LS Nibble is four bits of status<br>(A0, A1, A2, A3, A4, A5, A6, A7)<br>Bit Value 0/1<br>$B_0$ =Aux 1<br>$B_1$ =Aux 2<br>$B_2$ =Aux 3 |
| 6<br>7 | "X"<br>0x30 — 0x37 | Extra Status<br>LS Nibble is four bits of status<br>(X0, X1, X2, X3, X4, X5, X6)<br>Bit Value 0/1<br>$B_0$ =Local Mode<br>$B_1$ =Pressure Loss<br>$B_2$ =Video Loss<br>$B_3$ =Not Used |
| 8 | CHECKSUM | 0x80 0x8F Least significant nibble is XOR of all previous bytes (LS Nibble only) except 0xF8 |

Table 49. Format of Latch Status Response

| Byte | Data | Description |
|---|---|---|
| 0 | 0xF8 | Auto baud character |
| 1 | Address | Camera address 0x01 ⟶ 0xDF |
| 2 | "H" | Home Status |
| 3 | "0" — "9" or<br><br>"A"<br>"I"<br>"E" | ie, 2 = Home position 2<br>(H0, H1, H2, H3, H4, H5, H6, H7, H8, H9)<br>(HA) Home active<br>(HI) Home Not — at — Home position, or active<br>(HE) Home error could not get to home position |
| 4 | CHECKSUM | 0x80 to 0x8F Least significant nibble is XOR of all previous bytes (LS Nibble only) except 0xF8 |

Table 50. Format of Home Status

## 8.8 Receiver Position Message

| Byte | Data | Description |
|---|---|---|
| 0 | 0xF8 | Auto baud character |
| 1 | Address | Camera address (0x01 to 0xDF) |
| 2 | "P" or "p" | Upper case "P" is response;<br>Lower case "p" is GOTO command |
| 3 | $A_2$ (MS nibble) | Azimuth position (See Notes) |
| 4 | $A_1$ | |
| 5 | $A_0$ (LS nibble) | |
| 6 | $E_2$ (MS nibble) | Elevation position (See Notes ) |
| 7 | $E_1$ | |
| 8 | $E_0$ (LS nibble) | |
| 9 | CHECKSUM | 0x80 0x8F Least significant nibble is XOR of all previous bytes (LS Nibble only) except 0xF8 |

Table 51. Format of Pan/Tilt Position Message

| Byte | Data | Description |
|---|---|---|
| 0 | 0xF8 | Auto baud character |
| 1 | Address | Camera address (0x01 to 0xDF) |
| 2 | "V" or "v" | Upper case "V" is response;<br>Lower case "v" is GOTO command |
| 3 | $A_2$ (MS nibble) | Zoom position (See Notes) |
| 4 | $A_1$ | |
| 5 | $A_0$ (LS nibble) | |
| 6 | $E_2$ (MS nibble) | Focus position (See Notes ) |
| 7 | $E_1$ | |
| 8 | $E_0$ (LS nibble) | |
| 9 | CHECKSUM | 0x80 0x8F Least significant nibble is XOR of all previous bytes (LS Nibble only) except 0xF8 |

Table 52. Format of Zoom/Focus Position Message

### 8.8.1 Position Encoding

The message data for the "Pan/Tilt Goto Command" above begins with an ASCII 'p' followed by the azimuth ($A_2A_1A_0$) and elevation ($E_2E_1E_0$) positions. The positions are 12-bit values encoded four-bits each in the least significant nibbles of the three bytes whose most significant nibbles are always 0x30. The subscript-2 indicates the byte containing the most significant bytes of the position and subscript-0 indicates the least significant.

The 12-bit position data is contained in nibbles $0 \longrightarrow 2$, with:

nibble $0 = b_0 \longrightarrow b_3$,
nibble $1 = b_4 \longrightarrow b_7$, and
nibble $2 = b_8 \longrightarrow b_{11}$.

The nibble is then added to `0x30` to produce a value from `0x30` to `0x3F`.

#### 8.8.1.1 Position Encoding Example:

Example: Encoding an azimuth position value of decimal $2748_{10}$ or `0xABC`, is broken up into three nibbles and added `0x30` to each so:

$A_2$ = `0x3A`,
$A_1$ = `0x3B`, and
$A_0$ = `0x3C`.

The same encoding scheme is used for the zoom and focus positions in the "Zoom/Focus Goto Command".

### 8.8.2 Pan/Tilt Positions

Pan readouts are based on the $CAL_0$ (North or $0^o$) value of the PTZ. Angular readouts then increment to $90^o$ (East), $180^o$ (South) and to $270^o$ (West) when looking down on the PTZ unit from above.

Tilt readouts are based on the horizon, with $0^o$ meaning that the PTZ is looking at the horizon. As the PTZ points down the protocol readout changes to $45^o$ and then $90^o$ when the PTZ is pointing straight down. The PTZ on-screen display will indicate - $45^o$ and then -$90^o$. The discrepancy in the two types of readouts is that the protocol value is unsigned and the on-screen is signed. As tilt passes through straight down (not physically possible, but conceptually convenient, the protocol values increment until a straight up readout is $270^o$ and the on-screen display shows $90^o$. At $45^o$ above the horizon the protocol readout will be $315^o$ and the on-screen display will be $45^o$. Here is a short table to attempt and explain these values.

| Position | D readouts | Displays as |
|---|---|---|
| $90^o$ up | 270 | 90 |
| $45^o$ up | 315 | 45 |
| Horizontal - $1^o$ | 359 | 1 |
| Horizontal | 0 | 0 |
| $45^o$ down | 45 | -45 |
| $90^o$ down | 90 | -90 |

Table 53. Screen *vs.* Protocol Tilt Readouts

**Note**

These are for Table 53, page 115:

1. **Position:** Pointing direction of the enclosure/camera.
2. **D readouts:** D protocol returned value for this angle.
3. **Displays as:** What is displayed on the Spectra screen.

## 8.9 Receiver ID Message Format

| Byte | Data | Description |
|---|---|---|
| 0 | 0xF8 | Auto baud character |
| 1 | Address | Camera address (0x01 to 0xDF) |
| 2 | "d" | Display message |
| 3 | "M" | Mode select |
| 4 | "M" | (MM) Menu mode |
| | "I" | (MI) ID mode |
| 5 | CHECKSUM | 0x80 0x8F Least significant nibble is XOR of all previous bytes (LS Nibble only) except 0xF8 |

Table 54. Format of Select Mode Message

| Byte | Data | Description |
|---|---|---|
| 0 | 0xF8 | Auto baud character |
| 1 | Address | Camera address (0x01 to 0xDF) |
| 2 | "d" | Display message |
| 3 | "I" | ID display |
| 4 | "D" | (ID) Disable ID display |
| | "E" | (IE) Enable ID display |
| 5 | CHECKSUM | 0x80 0x8F Least significant nibble is XOR of all previous bytes (LS Nibble only) except 0xF8 |

Table 55. Format of ID Enable/Disable Message

NOTE: For all messages from master control panel to control receiver, ACK or NAK is returned.

| Byte | Data | Description |
|---|---|---|
| 0 | 0xF8 | Auto baud character |
| 1 | Address | Camera address (0x01 to 0xDF) |
| 2 | "d" | Display message |
| 3 | "C" | (dC) Clear screen |
| 4 | CHECKSUM | 0x80 0x8F Least significant nibble is XOR of all previous bytes (LS Nibble only) except 0xF8 |

Table 56. Format of ID Clear Screen Message

| Byte | Data | Description |
|---|---|---|
| 0 | 0xF8 | Auto baud character |
| 1 | Address | Camera address (0x01 to 0xDF) |
| 2 | "d" | Display message |
| 3 | "I" | ID display |
| 4 | "T" | (dIT) ID at top |
| 5 | CHECKSUM | 0x80 0x8F Least significant nibble is XOR of all previous bytes (LS Nibble only) except 0xF8 |

Table 57. Format of Select ID Top Message

| Byte | Data | Description |
|---|---|---|
| 0 | 0xF8 | Auto baud character |
| 1 | Address | Camera address (0x01 to 0xDF) |
| 2 | "d" | Display message |
| 3 | "I" | ID display |
| 4 | "B" | (dIB) ID at bottom |
| 5 | CHECKSUM | 0x80 0x8F Least significant nibble is XOR of all previous bytes (LS Nibble only) except 0xF8 |

Table 58. Format of Select ID Bottom Message

| Byte | Data | Description |
|---|---|---|
| 0 | 0xF8 | Auto baud character |
| 1 | Address | Camera address (0x01 to 0xDF) |
| 2 | "d" | Display message |
| 3 | "L" | Line of text |
| 4 | "1" — "<" (0x31 – 0x3C) | Line number (1–12) + 0x30 (dL1, dL2, dL3, dL4, dL5, dL6, dL7, dL8, dL9, dL:, dL;, dL<) |
| 5 | (ID text) | ASCII characters (up to 24) for the line starting with the left most character |
| $n$ | CHECKSUM | 0x80 0x8F Least significant nibble is XOR of all previous bytes (LS Nibble only) except 0xF8 |

Table 59. Format of Update Line Message

| Byte | Data | Description |
|---|---|---|
| 0 | 0xF8 | Auto baud character |
| 1 | Address | Camera address (0x01 to 0xDF) |
| 2 | "d" | Display message |
| 3 | "A" | Alarm message |
| 4 | "1" — "3" (0x31 – 0x33) | Display line number (1–3) + 0x30 |
| 5 | "B" | Blink alarm message (dA1B, dA2B, dA3B) |
| | "E" | Enable alarm message (dA1E, dA2E, dA3E) |
| | "D" | Disable alarm message (dA1D, dA2D, dA3D) |
| 6 | CHECKSUM | 0x80 0x8F Least significant nibble is XOR of all previous bytes (LS Nibble only) except 0xF8 |

Table 60. Format of Alarm Display Message

# APPENDIX A

# A Cohu's Web Page

**Note**

This is an extract from Cohu's technical support web page. (`www.cohu-cameras.com`) It was obtained on 25AUG06. Cohu has an active policy of updating this page quite often.

## A.1 Operations Manuals

This is a list of current Manual Numbers *vs.* camera model numbers.

| Camera Model | Manual # | Camera Model | Manual # | Camera Model | Manual # |
|---|---|---|---|---|---|
| 1100 | 970b.pdf | 3950 (i-View) | 1027.pdf | 7500 | 1018.pdf |
| 1300 | 971a.pdf | 3960 (i-View II) | 1050.pdf | 7800 | 1030.pdf |
| 2100 | 979.pdf | 4710 | 862c.pdf | 7700 | 1029.pdf |
| 2200 | 972a.pdf | 4810 | 845e.pdf | 8210 | 917a.pdf |
| 2600 | 1009.pdf | 4910 | 924e.pdf | 8240 | 937a.pdf |
| 2700 | 1040.pdf | 4920 | 1001.pdf | 9200 i-commander | 1033.pdf |
| 2740 | 1048.pdf | 4990 | 947.pdf | 9300 i-control | 1034.pdf |
| 2750 (serial ctl) | 1064.pdf | 6310 | 931a.pdf | DSPX/DSPE | 957a.pdf |
| 2900 | 1016.pdf | 6400 | 882b.pdf | MPC-D | 836a.pdf |
| 3600 | 1042.pdf | 6600-1000 | 987.pdf | MPC-M & MPC-R | 829a.pdf |
| 3810 | 1003.pdf | 6600-3000 | 997.pdf | MPC RS422 I'face | 5035.pdf |
| 3820 | 1010.pdf | 6600 Timing | 1024.pdf | MPC RS232 I'face | 5030.pdf |
| 3830 | 1005.pdf | 6710 | 890a.pdf | Win MPC GUI | 1032.pdf |
| 3920 (i-Dome) | 1026.pdf | 7100 Operations | 1043.pdf | — | — |
| 3930 (housing only) | 1037.pdf | 7100 Reference | 1044.pdf | — | — |

## A.2 Software & Protocol Downloads

WinMPC v4703 download here (22 August 06) Note: this is an .exe file. Software and descriptive files for setup of Cohu 3800 Series and 3900 Series.

## A.3 Camera/System Protocol

1. Camera Protocol: Model 3500
2. System Protocol: MPC RS-232
3. System Protocol: MPC RS-422

[26] `$Header: d:/TXB-Cohu/RCS/CohuWeb.inc,v 1.2 2006-08-25 11:06:23-07 Hamilton Exp Hamilton $`

4. System Protocol: Positioners and domes
5. System Protocol: ER2221B CCR
6. Camera Protocol: idome/iview/LCU

## A.4 Camera GUIs

1. GUI for 3600 Series cameras (122 KB zip)
2. GUI for 3900 (i-Series) Use WinMPC from "Software System Control", above. Select manual 1032 from list.

## A.5 Cohu Data

Cohu, Inc., Electronics Division

Correspondence:
P.O. Box 85623
San Diego, CA 92186, USA

Factory:
3912 Calle Fortunada
San Diego, CA 92123-1827
Tel. +1 (858) 277-6700
Fax: +1 (858) 277-0221

## APPENDIX B

# Index