Programmable Camera
Preset and Command Information

25 June 2007

Eric Hamilton

Note

This is a short description of various characteristics of Pelco's programmable camera units and the protocol commands that control them.

[1] `$Header: /EHamilton/General Information/Presets and Commands/PandC.tex 17 6/25/07 9:12a Ehamilton $`

Change Log

1. **06DEC05**
   1.1. Updated entries on Auto Focus and Auto Iris to reflect what the equipment actually does.
2. **18MAY06**
   2.1. Added in information about how to turn the IR Cut Filter on and off.
   2.2. Added in information about the timing of preset commands with the Spectra III.
   2.3. Added in information about labels in Spectra III and ExSite.
   2.4. Added in information about how to download a Spectra III and ExSite.
3. **06JUN06**
   3.1. Added in information about a query (`0x45`) responses.
4. **07JUN06**
   4.1. Added in information about how to interpert pan/tilt query responses. (Information was updated some more on 25SEP06.)
5. **20JUL06**
   5.1. Deleted all references to the ASIC Spectra and changed them to Spectra IV.
6. **16AUG06**
   6.1. Addeded in some information about replies.
7. **08SEP06**
   7.1. Addeded in information about the Spectra IV.
8. **26SEP06**
   8.1. Addeded in some information about getting pan angle readouts.
9. **02OCT06**
   9.1. Addeded in some data line captures from downloading a Spectra III. Included some comments about the listings.
10. **30NOV06**
    10.1. Addeded in information about the on-screen label logic for Spectra IV.
11. **25JUN07**
    11.1. Added in information about the ERD97P21-U.
    11.2. Added in information about setting time for the Spectra IV.

TABLE OF CONTENTS

Section **Page**

LIST OF TABLES

Table **Page**

1 Miscellaneous Camera Characteristics

Pelco programmable camera systems have evolved over several years to have some special characterizes. One of the more important characteristic is that new models are compatible with earlier models. The primary area of difference here is that some of the integrated cameras have new features or have discontinued older features. The "basic" set of programmable systems is supported as well as it can be within the limitations of the integrated camera systems. In early systems, Intercept types, each unit only supported one communications protocol at a time, newer systems support a minimum of three communications protocols. Additional information about commands is in Section 2, page 13.

This is a short listing of some of the more important items. This list is not complete, nor is it completely correct for products that are being currently shipped. Pelco manufactures, and has manufactured in the past, many different systems. This is a general listing of features that are on many systems.

Most of this information came from the documentation for the Intercept and Spectra I systems.

1.1 Focusing

The default focus mode is auto mode (the mode after a dome or camera reset). This means that the device (the dome) controls whether auto-focus is turned on or off. There is also an always off mode. The mode is set/changed by a command.

When in auto mode, the camera starts out with auto-focus on. If the receiver receives a focus near or far command, auto-focus is turned off. It stays off until a pan or tilt command is received or the dome or camera is reset.

1.2 Iris

The default iris mode is auto mode (the mode after a dome or camera reset). This means that the device (the dome) controls whether auto-focus is turned on or off. There is also an always off mode. The mode is set/changed by a command.

When in auto mode, the camera starts out with auto-iris on. If the receiver receives an iris open or close command, auto-iris is turned off. It stays off until the device pans or tilts more than 15^o from the position where auto iris was turned off or the dome or camera is reset.

If the Spectra is in auto mode for auto iris and also in auto mode for AGC (see below), the following happens. If the iris is all the way open and the Spectra receives an iris stop command and then an iris open command within one second of receiving the stop, then AGC is turned off and the camera gain is slowly increased until an iris stop is received or the upper gain limit is increased.

If the gain has been increased by an auto iris command and a close iris command is received, the gain is slowly decreased until it reaches the value it had when the gain started to increase. Then AGC is turned back on and the iris is closed.

[2] `$Header: /EHamilton/General Information/Presets and Commands/CamData.inc 5 6/25/07 9:12a Ehamilton $`

1.3 Limits

All of Pelco's programmable cameras feature software controlled pan angle limits. There are two types of these limits, one type is manual limits which limits the amount that the unit may be panned manually and the other is scan limits. Scan limits limit how wide an area the auto and random scan commands will automatically move the unit in.

1.3.1 Setting Manual Limits

The manual limits must be set **more** than 10^o apart. The Spectra III, or later software will not allow closer limits to be set.

1. To set up manual limits using presets, do the following:
   - 1.1. Move the Spectra III, or later to the left manual limit position.
   - 1.2. Send SET PRESET 90.
   - 1.3. Move the Spectra III, or later to the right manual limit position.
   - 1.4. Send SET PRESET 91. And the Spectra III, or later will not be able to move out of the left and right manual limits that were just set.
2. To remove both manual limits using presets, do the following:
   - 2.1. Move to the left manual limit and send SET PRESET 90. If the Spectra III, or later moves past the exact position of the limit, due to having too much momentum, this will still work. Some models of the Spectra III, or later will normally move several degrees past where the limit is set when they approach the limit at high speed.
3. To move the manual limits using presets, do the following:
   - 3.1. Move to the desired new limit and send a second set preset command. This will always work for the right manual limit using SET PRESET 91 and will only work for the left manual limit using SET PRESET 90 if the position is not **exactly** ($\pm$ "overshoot" caused by rapid motion) at the left limit. I.e. the left limit may only be moved to the right, not to the left, while the right manual limit may be moved small amounts to the right and any amount to the left. However the right limit may not be moved closer than 10^o to the left limit.

1.3.2 Setting Scan Limits

The scan limits must be set **more** than 10^o apart. The Spectra III, or later software will not allow closer limits to be set.

1. To set up scan limits using presets, do the following:
   - 1.1. Move the Spectra III, or later to the left scan limit position.

1.2. Send SET PRESET 92.

1.3. Move the Spectra III, or later to the right scan limit position.

1.4. Send SET PRESET 93.

2. To remove both scan limits using presets, do the following:

2.1. Move to the left scan limit and send SET PRESET 92.

1.4 Pattern

Recording of a pattern starts when a start pattern record command is received. It ends when either an end pattern record command is received, or 60 seconds has elapsed since the start pattern record command was received. The message "`PROGRAMMING PATTERN`" is put on the first video line. When recording stops (either by command or timeout), the message is cleared.

Playing of the recorded pattern starts when a start pattern play command is received. When the end of the recorded pattern is reached, playback starts over again at the beginning of the pattern. This continues until any other command is received. If zones have not been enabled, the message "`RUNNING PATTERN`" is put on the first video line and remains there until playback stops, at which point it is cleared. If zones have been enabled, the "`RUNNING PATTERN`" message will not be shown. Instead, the zone labels will be shown as the pattern moves through the zones. When playback is stopped, the first video line is cleared.

There are two types of pattern processing: record and playback. Pattern processing occurs once each timer tick (14 times a second[3]). If recording, the current command is stored in the EEPROM. If the current command is not one that can be played back, an illegal command is saved (it will be skipped during playback). If playing back, a command is read out of the EEPROM and decoded. If it is an illegal command, it is skipped.

Additional information about patterns is in Section 2.1.1, page 15. Presets

1.5 Presets

Presets can be moved to, set, or cleared.

When a move to preset command is received, the preset position stored for the preset number specified in the command is checked. If the position is not valid, the command is ignored. Otherwise the unit moves to the preset pan, tilt, zoom, and focus positions. Once the preset has been reached, the preset label is displayed on the second video line.

If any command which causes motion is received during a move to preset, the move will be aborted and the new command will start. These commands are: a motion command, or another move to preset command, Also if the move is not completed within a timeout period, the move is aborted and motion is stopped.

[3]Or depending on the model it may be each vertical interrupt (60 or 50 times a second depending on whether the video is NTSC or PAL).

When a set preset command is received, the current pan, tilt, focus, and zoom positions are saved for the preset number specified in the command and the label for that preset becomes whatever is currently on the second video line.

The clear preset command makes the stored preset for the preset number specified in the command invalid so that it can not be moved to.

Additional information about presets is in Section 7, page 70.

1.5.1 Note on preset processing in a Spectra III

```
From:     Hannen, Craig
Sent:     Wednesday, October 26, 2005 12:13 PM
To:       Hamilton, Eric
Subject:  \indexit{Spectra III} and Preset Processing
```

My tests seem to show a general trend in how Spectra III processes back-to-back presets (or back-to-back commands in general). Spectra III will queue all commands and handle them in the order of arrival. If the commands happen to be back-to-back preset commands, Spectra III requires about 1.5 seconds (or more) to go to each preset position. So it's easy for Spectra III to fall behind if commands are arriving every .5 seconds. Spectra III will eventually "catch up" with the commands in the queue or the queue will overflow (causing dropped commands). The Spectra III queue can hold up to 128 bytes (or 18 D Protocol commands).

For my testing I was sending D Protocol commands directly with variable timing. I was sending 3 preset commands in the same order over and over. When I filled up the queue and halted the controller, Spectra III would play "catch up" and execute the left over preset commands in the queue.

There is a known case where a preset will be dropped. If the custom camera settings on 2 back-to-back presets are different (specifically if the focus mode of the 2 presets is different), Spectra III will drop the second preset.

I don't know why 1.5 seconds are required to get to a preset. I think the pan/tilt motion is almost immediate. Getting the camera into the correct position might be the part of the preset that takes the longest. There's a set of commands that are sent to the camera on every preset.

1.6 Creating Labels

Many devices have an ability to display labels on the video. Labels that identify the preset or zone being scanned are common. There is a special technique to establish a label that is associated with either a preset or a zone. First, send the label to the receiver/driver using the "Write Character to Screen" (`0x15`) command. After the label is on the screen, then set the preset or zone. That will establish the label and associate it with that preset or zone.

1.6.1 Write Character To Screen (`0x15`)

The parameter in byte 5 of this command indicates the column to write to. This parameter is interpreted as follows:

- Columns 0-19 are used to receive zone labels.
- Columns 20-39 are used to receive preset labels.
- Zone Scan On (`0x1B`): Used to turn the displaying of zone labels on.
- Zone Scan Off (`0x1D`): Used to turn the displaying of zone labels off.

For the Spectra III, Spectra III SE and ExSite only, characters written to these positions are not written directly to the screen. Once the SET_ZONE_START (opcode `0x11`) command is received, the characters are displayed.

For the Spectra III, Spectra III SE and ExSite only, characters written to these positions are not written directly to the screen. However, if characters are written to these columns and no SET_PRESET (opcode `0x03`) command is received within 250 milliseconds of receipt of the last character. The characters will be displayed on the screen beginning at the first column of the second row of the display

Does this add anything to the description?

What has to happen is somewhat unexpected. I think that I have covered all points below:

1. *Send "Zone Scan Off"*
2. *Send characters to positions* $0 \longrightarrow 19$.
3. *Send a "Zone Start"*
4. *Send a "Zone End"*
5. *Send a "Zone Scan On"*

Now when the camera is in one of the eight zones, the zone message will appear. The internal logic of the software, copies what is in the zone label buffer into EEPROM when it gets a "Zone End" command. With the older units, all units prior to the Spectra III, the zone label was always on the first displayed line. With the Spectra III the zone label can be anywhere on the screen and the logic to move it is internal to the Spectra III software.

The most general purpose method of putting label up is to use positions $20 \longrightarrow 39$*, which may or may not be on the second line.*

1.7 Label Logic in Spectra IV

In response to questions on labels in Spectra IV [4], This informal spec on how labels currently work in Spectra IV:

1. Azimuth Elevation, Direction, and Zoom Magnification:

   1.1. PTZ

[4] This information about labels came from: Jeremy Watson in Novemember 2006.

1.1.1. Labels pop on-screen when there is movement and disappear after movement stops and the display time has expired

1.2. Patterns

1.2.1. While recording, the labels are shown while there is movement just like normal PTZ

1.2.2. While playing, the labels are hidden

1.3. Presets

1.3.1. On a preset GO command, the labels are hidden

1.4. Scans

1.4.1. While scanning, the labels are hidden

Of course if a label's display time is set to CONSTANT, it always shows no matter what (except in menus of course).

2. Alarm:

2.1. 1 Alarm Active

2.1.1. Alarm label is displayed for display time, then goes off until alarm clears and retriggers

2.2. >1 Alarm Active

2.2.1. Alarm labels will sequence on-screen (switching every X seconds) to the next active label in the sequence

2.2.2. If the sequence time $>$ display time for the alarm labels, the alarm will disappear until the next sequence

In the case of CONSTANT display time, alarm labels will remain on-screen until that alarm clears or the next alarm in sequence fires.

How are alarm priorities handled when two different levels are active? Do they alternate, or only the higher priority displayed?

Only alarms of the highest priority count... If there are 4 alarms present and only 1 is HIGH priority, it falls into the 1 alarm active catagory.

3. Preset, Zone:

3.1. On a preset GO command, the preset label pop on-screen when the preset is reached

3.1.1. In the case of CONSTANT display time, this label remains as long as the camera sits at the preset

3.2. When the camera passes through a zone or is sitting on a zone, the zone label is displayed

3.2.1. In the case of CONSTANT display time, this label remains as long as the camera remains in the zone

4. Time/Date:

4.1. Time/Date information is either on the screen at all times or not

1.8 Speed Ramping

To avoid abrupt speed changes (which could cause clunking noises or even motor stalling), the speeds (angular velocities in degrees/second) are ramped up or down. A command that causes an speed change (such as a motion command or a move to preset) does not set the speed directly. Instead, it sets a desired speed and direction. The ramping task compares the current speed (degrees/second) and direction to the desired speed and direction and calculates a new current speed and direction. This new speed is calculated to keep the angular acceleration (degrees/second/second) approximately constant.

1.9 Zones

When a zone start command is received, the current pan position is saved as the start position for the zone number specified in the command. Also whatever is displayed on the first video line is saved as the zone label. Note that zone scan must be turned off before this command is received or the zone programming will not work correctly.

When a zone end command is received, the monitor is unlocked, and the current pan position is saved as the end position for the zone number specified in the command.

Zones extend from the start point clockwise to the end point. This means that if a zone start point is set, the Intercept is panned slightly clockwise, and the zone end point is set, the zone will be small. But if the Intercept is panned slightly counterclockwise between the start and end points, the zone will be almost all the way around the pan circle.

There are commands to turn zone scan on and off. If zone scan has been turned on, the current pan position is continuously read. If the current position is within a zone, the label for that zone is displayed on the first video line. If the current position is not within any zone, the line is cleared. If the current position is within more than one zone, the label for the highest-numbered zone will be displayed.

1.10 Zooming

When zooming in (tele), the camera first does optical zoom, meaning that the elements of the lens move to do the zooming. After reaching maximum optical zoom (the lower zoom limit), the camera then does electronic zooming. Electronic zooming involves no lens movement, until it reaches a preset limit (the higher zoom limit). If the receiver receives a zoom stop command after reaching the lower zoom limit, and then another zoom in command within one second of receiving the stop, the receiver increases electronically to the higher zoom limit.

When zooming out (wide), the camera zooms out through the electronic zoom range and then the optical zoom range. It does not stop at the lower zoom limit.

2 Protocol commands

Various Pelco products use different sets of the three basic command and control protocols. The currently supported protocols are:

1. **Coaxitron** this "one-way" protocol, it is sent "up the wire" over the same conductor that the camera video is coming down on. Camera addresses are not used as cameras may not be daisy-chained. There are at least three different sub-types[6]:

   1.1. Standard 15-bit, with two sub-divisions:

      1.1.1. Ordinary 15-bit commands consist of a full command being sent in a single field on line 18. Due to reliability problems motion commands, which include pan, tilt and zoom. These are sent repeatedly. When the command stops being sent the indicated type of motion also stops without having a stop motion command being sent.

      1.1.2. Special standard 15-bit commands are regarded as any command that is not repeated. The command is sent only once on line 18.
      In 15-bit Coaxitron, when longer commands have to be sent, they are sent on twenty six consecutive fields on line 18. These are: set preset, goto preset and preset scan commands.

   1.2. Extended 32-bit Coaxitron special messages include set preset, call preset, start pattern programming, end pattern programming, execute pattern, set focus speed, set zoom speed, set auxiliary, clear auxiliary, and write character. All extended commands have "odd op-codes".
   These commands consist of two 16-bit commands sent on lines 18 and 19 of a field.

2. **D Protocol** This is a two way protocol that is normally sent as an RS-422 or RS-485 command. Most commands receive a four byte response. There are several commands ("query") that get longer replies. All non-query commands, **including non-defined commands**, get a three byte response when they are received. The full command must be received in less than 250 ms, or the unit is expected to delete what has been received so far and set up to receive the next command. D Protocol is the most capable command set that Pelco utilizes. Camera addresses start at 1 and go up to 256. All D Protocol commands have "odd op-codes".

3. **P Protocol** This is a two way protocol that is normally sent as an RS-422 command. All commands receive a single byte response of either `ACK` or `NAK` or nothing (if the command times out, 250 ms, before being completely received.) This command is usually used with the CM-9760 series of matrices. All P Protocol commands have "odd op-codes".

[5] `$Header: /EHamilton/General Information/Presets and Commands/Cmnds.inc 19 11/30/06 8:47a Ehamilton $`

[6] *From the Technical File (TF-0019) describing Coaxitron dated 4/12/2004 on the bottom of page 5.*

2.1 Command Op-Codes In Use

	x1	x3	x5	x7	x9	xB	xD	xF
0x	XR	SP	CP	GOTO	SA	CA	D	Rst
1x	ZS	ZE	WC	CS	AA	SOn	SOf	PSta
2x	PSto	RP	SZS	SFS	RCD	AF	AI	AGC
3x	BLC	AWB	PD	SS	LLP	WBRB	WBMG	AG
4x	AIL	AIP	Q	PS	SZeP	SPP	STP	SZP
5x	QPP	QTP	QZP	DL	QPR	QTR	QZR	SM
6x	QM	QMR	AEM	SRBR	SDL	QDT	QDTR	QDI
7x	QDIR	VI	E	—	—	—	—	—
8x	—	—	—	—	—	—	—	—
9x	—	—	—	—	—	—	—	—
Ax	—	—	—	—	—	—	—	—
Bx	—	—	—	—	—	—	—	—
Cx	—	—	—	—	—	—	—	—
Dx	—	—	—	—	—	—	—	—
Ex	—	—	—	—	—	—	—	—
Fx	—	—	—	—	—	—	RS	IOC

Table 1. Protocol commands in use

Most commands up through `0x69` (Query) are supported in newer versions of the various protocols. There are several tables of exactly which command is supported in each protocol. These tables are:

1	Esprit 3012 rev 2.06	Table 5, page 43
2	Esprit rev 3.23	Table 6, page 44
3	ExSite rev 1.13	Table 7, page 45, Table 8, page 46
4	Intercept rev 3.08	Table 9, page 47
5	LRD rev 2.00	Table 10, page 47, Table 11, page 47
6	MRD	Table 12, page 48
7	Spectra I rev 2.04	Table 13, page 48
8	Spectra II rev 3.44	Table 14, page 49
9	Spectra II Lite rev 4.40	Table 15, page 49
10	Spectra III rev 1.27	Table 16, page 50, Table 17, page 51

The Intercept series of equipment supported only one protocol at a time. With the Intercept series there were options for a character generator and for the ability to work with presets.

2.1.1 Pattern numbers

The method of saving a pattern has changed over time. The original method was to save what the unit was doing every vertical sync time. This resulted in getting 60 (50 with PAL based cameras) records a second. Older units had patterns defined in maximum minutes of run time. With these systems the choice is one one minute pattern (pattern 0) or two one half minute patterns (pattern 1 and 2).

Starting with the Spectra III/ExSite series of units. Saving what the unit was doing each vertical sync time is used, however if the unit is doing the same thing on several vertical sync times in a row, then an eight bit counter is incremented and saved when a change is detected. This Run Length Limited (RLL) technique makes it so that the saved pattern length may be of greater duration. The total duration is now unknown in advance as it depends on the number of **changes** in what the unit is told to do while recording the pattern. (A patent is pending on this method of saving a pattern.)

The numbering and quantity of patterns varies on different units:

1. On many Intercept type units there was one pattern. On other Intercept units, those with out presets, there were no patterns.
2. The Spectra I, Spectra II and Esprit have pattern numbers from 0 $\longrightarrow$ 2.
3. With the Spectra III SE and ExSite patterns are numbered in the range of 1 $\longrightarrow$ 4.
4. With low cost units utilizing the "434" (18X), and the "934" (16X) type of camera, there is only one preset.
5. On the Spectra II Lite the only pattern was numbered 0.
6. On the Spectra III the only pattern is numbered 1.

2.1.2 Description of Commands

From FUNC.DOC that was in the Spectra I, rev 1.06 release.

1. **Adjust auto-iris level**, 65_{10}/`0x41`: If byte 1 is 0, byte 5 and byte 6 are the high and low bytes respectively of an unsigned 16-bit number which the new level. If byte 1 is 1, byte 5 and byte 6 are the high and low bytes respectively of a signed 16-bit number which is the change to the current level. The level is limited internally to the range from 0 to 255 (`0xFF`). If an attempt is made to set or change the level to a value outside this range, the gain will be set to the appropriate end of the range.
2. **Adjust auto-iris peak value**, 67_{10}/`0x43`: If byte 1 is 0, byte 5 and byte 6 are the high and low bytes respectively of an unsigned 16-bit number which the new peak value. If byte 1 is 1, byte 5 and byte 6 are the high and low bytes respectively of a signed 16-bit number which is the change to the current peak. The peak is limited internally to the range from 0 to 255 (`0xFF`). If an attempt is made to set or change the peak to a value outside this range, the peak will be set to the appropriate end of the range.

3. **Adjust gain**, $3F_{10}$/`0x63`: If byte 1 is 0, byte 5 and byte 6 are the high and low bytes respectively of an unsigned 16-bit number which is the new gain. If byte 1 is 1, byte 5 and byte 6 are the high and low bytes respectively of a signed 16-bit number which is the change to the current gain. The gain is limited internally to the range from 0 to 448 (`0x1C0`). If an attempt is made to set or change the gain to a value outside this range, the gain will be set to the appropriate end of the range.

4. **Adjust line lock phase delay**, 57_{10}/`0x39`: If byte 1 is 0, byte 5 and byte 6 are the high and low bytes respectively of an unsigned 16-bit number which is the new phase delay. If byte 1 is 1, byte 5 and byte 6 are the high and low bytes respectively of a signed 16- bit number which is the change to the current phase delay. The phase delay is the delay between the zero crossing of the AC power waveform and the line lock signal sent to the camera. It is in units of 1.085 microseconds. The phase delay is limited internally to the range from 0 to 32767. If an attempt is made to set or change the delay to a value outside this range, the delay will be set to the appropriate end of the range. This command disables device phase delay mode.

5. **Adjust white balance (R-B)**, 59_{10}/`0x3B`: If byte 1 is 0, byte 5 and byte 6 are the high and low bytes respectively of an unsigned 16-bit number which is the new red-blue white balance value. If byte 1 is 1, byte 5 and byte 6 are the high and low bytes respectively of a signed 16-bit number which is the change to the current red-blue white balance value. The balance value is limited internally to the range from 192 (`0xC0`) to 768 (`0x300`). If an attempt is made to set or change the balance to a value outside this range, the balance will be set to the appropriate end of the range. This command turns off auto white balance.

6. **Adjust white balance (M-G)**, 61_{10}/`0x3D`: If byte 1 is 0, byte 5 and byte 6 are the high and low bytes respectively of an unsigned 16-bit number which is the new magenta-green white balance value. If byte 1 is 1, byte 5 and byte 6 are the high and low bytes respectively of a signed 16-bit number which is the change to the current magenta-green white balance value. The balance value is limited internally to the range from 192 (`0xC0`) to 768 (`0x300`). If an attempt is made to set or change the balance to a value outside this range, the balance will be set to the appropriate end of the range. This command turns off auto white balance.

7. **AGC**, auto/on/off, 47_{10}/`0x2F`: If byte 6 is 0 the device automatically controls whether AGC (automatic gain control) is on or off (default), If byte 6 is 1, AGC is turned off (manual gain). If byte 6 is 2, AGC is turned on. Other values are ignored. Sending an adjust gain command turns AGC off.

8. **Auto focus**, auto/off 43_{10}/`0x2B`: If byte 6 is 0 the device automatically controls whether auto focus is on (default) or off. If byte 6 is 1, auto focus is turned off. Other values are ignored.

9. **Auto iris**, auto/off, 45_{10}/`0x2D`: If byte 6 is 0 the device automatically controls whether auto iris is on (default) or off. If byte 6 is 1, auto iris is turned off. Other values are ignored.

10. **Auto white balance**, off/on 51_{10}/`0x33`: If byte 6 is 1, auto white balance is turned on

(default). If byte 6 is 2, auto white balance is turned off. Other values are ignored. Sending an adjust white balance command turns auto white balance off.

11. **Backlight compensation**, off/on 49_{10}/`0x31`: If byte 6 is 1, backlight compensation is turned off (default). If byte 6 is 2, backlight compensation is turned on. Other values are ignored.

12. **Enable device phase delay mode**, 53_{10}/`0x35`: When device phase delay is set, the phase delay is set by the device (there may be a manual adjustment). Sending an adjust line lock phase delay command will disable device phase delay mode.

13. **Reset camera to defaults**, 41_{10}/`0x29`: Resets the camera to its default condition, except that the current phase delay (see below) is not changed.

14. **Set shutter speed** 55_{10}/`0x37`:

    14.1. With units preceeding the Spectra III series: Byte 5 and byte 6 are the high and low bytes respectively of 1 divided by the shutter speed. The shutter speed is limited internally to the range from $\frac{1}{60}$ second (NTSC) or $\frac{1}{50}$ second (PAL) to $\frac{1}{30000}$ second, corresponding to a sent number range from 60 (or 50) to 30000. If the sent number is, 0 the shutter speed is reset to its default value ($\frac{1}{60}$ or $\frac{1}{50}$ second). If the sent number is 1, the shutter speed is moved to the next faster speed in the shutter speed table. If the sent number is 2, the shutter speed is set to the next slower speed in the table.

    14.2. With the Spectra III series of units this has been changed as follows: Byte 6 is the only byte processed. Byte 6 is an index into an array of usable shutter speeds. An index of `0x00` represents auto-shutter, and indexes from $1 \longrightarrow 16$ represent the supported shutter speeds. The actual shutter speed values vary depending on the exact make and rev of camera installed.

2.2 Spectra Download Process Used Via The 422/485 Port

10/31/2002
Eric Bopp[8]
Original name = "`sp3 download.doc''`

2.2.1 Background

At the start of the process the download application (Windows, Palm or iPaq) does not know whether the Spectra application is running or if the unit is already at the "Ready for download" prompt. So the first step is to make sure we are at the "Ready for download" prompt. Additionally, our specification says that the user should not be able to set the baud used by the download application. This means that the download applilcation will have to test communications at all possible baud rates. The possible initial baud rates are as follows: if the BIOS is running the baud will be 2400, if the application is running the baud could be 2400, 4800 or 9600 (as set by the DIP switches).

2.2.2 Getting the unit into "Ready for download" mode

Send a "D" protocol message containing the "download" opcode (`0x57`, `EC_DOWNLOAD`). Cycle through the possible bauds (2400, 4800, 9600) starting with 2400, until a valid response is received. Once we have acknowledgement that this message has been received by the unit, we can be sure that the unit is at the "Ready for download" prompt and that the baud is 2400. (Section 2.2.5.1, page 22, lines 14 ⟶ 26 and Section 2.2.5.2, page 24, lines 16 ⟶ 26.) (`EC_DOWNLOAD` *forces a baud rate of 2400 and expects a reply at the original baud rate. I.e. if the Spectra is running at 4800 baud, when it receives an* `EC_DOWNLOAD` *command it will send a response at 4800 baud and then reconfigure to operate at 2400 baud.*)

Baud Codes	
Value	Baud
0	2,400
1	4,800
2	9,600
3	19,200
4	38,400
5	115,200

2.2.3 New Opcodes

Three new "D" protocol opcodes have been added for the purpose of determining the optimum baud for the download and for starting the download process. The following is a brief description of each opcode. For full details see the D protocol specification document.

[7] `$Header: /EHamilton/General Information/Presets and Commands/DNLD.INC 1 11/06/06 7:48a Ehamilton $`

[8] Additions by Eric Hamilton in November 2006

1. **"Activate Echo Mode"** message (opcode `0x65`, `EC_ECHO_MODE`). Sending this command puts the 422/485 port into a state where any character that is received is immediately retransmitted. The unit automatically comes out of this state when one of the following happens: more than 100 milliseconds pass without a character being received or more than 180 characters having been received. (*This command is sent at the current* `EC_SET_BAUD` *rate.*)

2. **"Set Baud",** `EC_SET_BAUD` message (opcode `0x67`). Sending this command changes the baud at which the unit communicates at on the 422/485 port. The unit does not change its baud until after it has sent a response to this message. The unit automatically falls back to 2400 baud if no characters are received for 100 milliseconds. (Note that this fallback condition does not apply once the unit begins the download process).

3. **"Start Download",** `EC_START_DOWNLOAD` message (opcode `0x69`). This message starts the download process.

2.2.4 Determining the optimum baud for downloading

The next step is to determine the maximum baud that can be reliably used for the download. The Spectra supports 2400, 4800, 9600, 19200, 38400 and 115200 baud rates for downloads via the 422/485 port. The following is the step-by-step process for determining the optimum baud for downloading. The term "recovery state" is used in the description below. "Recovery state" is defined as waiting more than 100 milliseconds and resetting the baud rate to 2400.

1. Start testing the communications link at 2400 baud. (Section 2.2.5.1, page 22, lines 27 ⟶ 29, Section 2.2.5.2, page 24, lines 28 ⟶ 30.) (*This assumes that the unit is in the "Ready for download" state described in Section 2.2.2, page 18, above.*)

2. Send the "Set Baud" (`0x67`, `EC_SET_BAUD`) message to set the baud to rate to be tested. If a valid response is received, the application should change its baud to the rate being tested and go to step 3. If a valid response is not received go to the recovery state, then resend the "Set Baud" message. Note that retry counts for the purpose of dropping to a lower baud do not need to be implemented for this action because we are always transmitting 2400 baud (the lowest baud). (`EC_SET_BAUD` *is always sent at 2400 baud and its response is always sent at 2400 baud. This works because a delay of at least 100 milliseconds has occurred since the last data byte transferred. Thus the Spectra has returned to the "recovery state" of 2400 baud.*)

Section 2.2.5.1, page 22 Lines	Section 2.2.5.2, page 24 Lines	
50, 51	49, 50	68, 69
58, 59	88, 89	103, 104
65, 66	116, 117	127, 128

3. Send the "Activate Echo Mode" message (`0x65`, `EC_ECHO_MODE`). If a valid response is received go to step 4, otherwise increment the retry count, go to the recovery state and do one of the following: (Section 2.2.5.1, page 22, lines 31, 32)

   3.1. If the maximum number of retries have occurred then step back to next lower baud and go to step 6.

3.2. Retry by going back to step 2.

4. Send the test packet. Go to step 5. (Section 2.2.5.1, page 22, lines 34 ⟶ 48)

Section 2.2.5.1, page 22 Lines	Section 2.2.5.2, page 24 Lines	
53 ⟶ 57	32 ⟶ 48	52 ⟶ 66
61 ⟶ 64	71 ⟶ 86	91 ⟶ 101
68 ⟶ 72	106 ⟶ 114	119 ⟶ 125

A typical 142 byte test packet is (In ASCII):

```
S345000002C0006D20F000CF000000800010FFFFFF00C7FFFFFFFF80FFFFFFFF00FFFFF
FF8FF63F724F00E72007C2470123CE00264070F7401C2E0DB746F2C8687046003960732
```

5. Receive the echoed back message and confirm that there are no errors. Go to the recovery state (this needs to be done regardless of success or failure). If there are no errors in the echo back data: if you are at the maximum baud then go to step 6 otherwise increment to the next baud and go to step 2. If there are errors in the echo back data or a timeout occurs then increment the retry count and do one of the following:

Section 2.2.5.1, page 22 Lines	Section 2.2.5.2, page 24 Lines	
53 ⟶ 57	32 ⟶ 48	52 ⟶ 66
61 ⟶ 64	71 ⟶ 86	91 ⟶ 101
68 ⟶ 72	106 ⟶ 114	119 ⟶ 125

5.1. if the maximum number of retries have occurred then begin the download at the next lower baud.

5.2. Retry by going back to step 2.

6. Send the "Set Baud" (`0x67`, `EC_SET_BAUD`) message to set the units baud to the "current baud" (the highest baud the passed the tests above). Repeat until a valid response is received. Once a valid response is received then go to step 7. (Section 2.2.5.1, page 22, lines 73, 74)

7. Send the "Start Download" (`0x69`, `EC_START_DOWNLOAD`) message. Repeat until a valid response is received. Once a valid response is received then continue the download using the algorithm that is used on the RJ-45 port (Section 2.3, page 31). (Section 2.2.5.1, page 22, lines 76, 77, Section 2.2.5.2, page 24, lines 130 ⟶ 136. The RJ-45 algorithm's operation is shown in Section 2.2.5.1, page 22, lines 79 ⟶ 89.)

2.2.5 Typical Download Captures

The following download captures were all done:

1. On a Spectra III SE,
2. Model `DD53CBW`,
3. Running software Version 1.34.000,

4. BIOS Version 1.12.0000,
5. With font version 3.03 loaded.
6. Initial Spectra communications parameters were to operate at 9600 baud.
7. The head end, a PC, was communicating at 9600 baud.
8. The capture software was running at 2400 baud.
9. Both downloads would have been successful, but were intentionally stopped as the only reason for doing them was to show how the start of a download sequence operates.
10. Many items of importance do not appear in the captures. This is because the actual events occur at data rates other than 2400 baud and only unintellageble junk was captured.
11. Other parts of the communication were intentionally deleted as having no importance to gaining an understanding of the basics of the download sequence.

A typical pair of lines from the capture file are (these came from Section 2.2.5.1, page 22, extra blanks have been removed):

```
A   B    C    D    E      F        G    H                      I
65 145, 202: DCE  365 4.398797 0.512087 ff 01 00 67 00 01 69 EC_SET_BAUD 4800
66 145, 164:  DTE 372 4.437848 0.004167 ff 01 00 69
```

A This is the sequential line number of the listing.

B This is the message number. This number is incremented every other time the "direction" of data changes.

C Byte number within "this" direction of data.

D Direction of data flow.

- `DCE`: this is for all data coming from the PC.
- `DTE`: this is for all data coming from the TXB, or Spectra.

E Overall total byte number from both directions.

F Time, in seconds, from when the first byte was detected by the capture software.

G Time, in seconds, from the preceding byte.

H The actual message. Usually these are arranged in full messages.

Many times, the data is being collected at the "wrong" baud rate, it is not always practable to try and group the data properly.

On long messages, only the start and end have been displayed.

I When it seems appropriate, a comment of what is happening on this line.

2.2.5.1 Download with the baud rate limited to 2400 baud

```
 1  $ Id: makesecs.l,v 1.8 2006-05-01 08:31:47-07 Hamilton Exp Hamilton $
 2  $Header: /EHamilton/General Information/Presets and Commands/dnldx01.out 2     11/03/06 9:37a
Ehamilton $
 3  FTS capture buffer (11/2/2006 9:27:49 AM)
 4  Event 1 (11/2/2006 9:18:58.625999 AM) through
 5  Event 2,150 (11/2/2006 9:19:11.900819 AM)
 6
 7  Capture between the Headend and the TXB
 8  Baud rate = 2400
 9  Using Perrin's computer
10  DCE = PC
11  DTE = TXB
12  Download would have been a success if I hadn't stopped it.
13
14       1,       1: DCE       1     0.000000   0.000000 ff
15       1,       2: DCE       2     0.405675   0.405675 ff 01 00 57 00 00 58 EC_DOWNLOAD
16       1,       9: DCE       9     0.751281   0.320688 0f
17       1,      10: DCE      10     0.755436   0.004155 20
18       1,      11: DCE      11     0.759574   0.004138
19       1,      12: DCE      12     0.759585   0.000011 00
20       1,      13: DCE      13     0.763561   0.003976 f2
21       1,      14: DCE      14     1.178100   0.414539 38
22       1,      15: DCE      15     1.182207   0.004107 f6
23
24       1,       1:  DTE     16     1.196256   0.004107 e8
25       1,       2:  DTE     17     1.217455   0.021199 ff 01 00 58
26
27       2,      16: DCE      21     1.419371   0.189416 ff 01 00 67 00 00 68 EC_SET_BAUD 2400
28       2,      23: DCE      28     1.966221   0.521932 ff 01 00 67 00 00 68 EC_SET_BAUD 2400
29       2,       6:  DTE     35     2.005346   0.004140 ff 01 00 68
30
31       3,      30: DCE      39     2.082392   0.064417 ff 01 00 65 00 00 66 EC_ECHO_MODE
32       3,      10:  DTE     46     2.121102   0.004141 ff 01 00 66
33
34       4,      37: DCE      50     2.154551   0.020845 53
35       4,      38: DCE      51     2.158692   0.004141 33
36       4,      39: DCE      52     2.162913   0.004221 34
37       4,      40: DCE      53     2.166999   0.004086 35
38       4,      14:  DTE     54     2.168523   0.004086 53
39       5,      41: DCE      55     2.171165   0.004086 30
40       5,      15:  DTE     56     2.172742   0.001577 33
41  <Echo Data>
42     141,     177: DCE     327     2.736086   0.000766 33
43     141,     151:  DTE    328     2.739513   0.003427 36
44     142,     178: DCE     329     2.740252   0.000739 32
45     142,     152:  DTE    330     2.743679   0.000739 30
46     142,     153:  DTE    331     2.747845   0.004166 37
47     142,     154:  DTE    332     2.752013   0.004168 33
48     142,     155:  DTE    333     2.756178   0.004165 32
49
```

```
50     143,     179: DCE     334     3.013095   0.256917 ff 01 00 67 00 01 69 EC_SET_BAUD 4800
51     143,     156:  DTE    341     3.051882   0.004166 ff 01 00 69
52
53     144,     186: DCE     345     3.129271   0.064783 0f
54     144,     187: DCE     346     3.133430   0.004159 80
55     144,     188: DCE     347     3.137571   0.004141
56     144,     189: DCE     348     3.137582   0.000011 00
57     144,     190: DCE     349     3.141554   0.003972 f5
58     144,     191: DCE     350     3.758199   0.616645 ff 01 00 67 00 01 69 EC_SET_BAUD 4800
59     144,     160:  DTE    357     3.797116   0.004168 ff 01 00 69
60
61     145,     198: DCE     361     3.874267   0.064653 1f
62     145,     199: DCE     362     3.878412   0.004145 80
63     145,     200: DCE     363     3.882569   0.004157 08
64     145,     201: DCE     364     3.886710   0.004141 f5
65     145,     202: DCE     365     4.398797   0.512087 ff 01 00 67 00 01 69 EC_SET_BAUD 4800
66     145,     164:  DTE    372     4.437848   0.004167 ff 01 00 69
67
68     146,     209: DCE     376     4.514868   0.064524 0f
69     146,     210: DCE     377     4.519030   0.004162 80
70     146,     211: DCE     378     4.523167   0.004137
71     146,     212: DCE     379     4.523177   0.000010 00
72     146,     213: DCE     380     4.527153   0.003976 f5
73     146,     214: DCE     381     5.278667   0.751514 ff 01 00 67 00 00 68 EC_SET_BAUD 2400
74     146,     168:  DTE    388     5.317610   0.004164 ff 01 00 68
75
76     147,     221: DCE     392     5.394915   0.064728 ff 01 00 69 00 00 6a EC_START_DOWNLOAD
77     147,     172:  DTE    399     5.433703   0.004167 ff 01 00 6a
78
79     148,     228: DCE     403     5.466989   0.020732 03
80     148,     176:  DTE    404     5.480918   0.020732 01
81
82     149,     229: DCE     405     5.501813   0.020732 53 30 30 . . . 31 42 0d
83     149,     177:  DTE    422     5.766859   0.198586 01
84
85     150,     246: DCE     423     5.787506   0.020647 53 33 34 . . . 35 34 0d
86     150,     178:  DTE    566     6.393469   0.004141 01
87
88  <Much more boring data>
89
90  There were a total of     2149 bytes transferred
91
92  There were a total of     1961 DCE bytes transferred
93  The first DCE byte came in at 0.000000 seconds from the start of data collection
94  The last DCE byte was at 13.265195 seconds from the start of data collection
95
96  There were a total of      188 DTE bytes transferred
97  The first DTE byte came in at 1.196256 seconds from the start of data collection
98  The last DTE byte was at 12.654480 seconds from the start of data collection
```

2.2.5.2 Download with the baud rate not limited

```
 1  $ Id: makesecs.l,v 1.8 2006-05-01 08:31:47-07 Hamilton Exp Hamilton $
 2  $Header: /EHamilton/General Information/Presets and Commands/dnldx02.out 1     11/02/06 2:13p
Ehamilton $
 3  FTS capture buffer (11/2/2006 1:37:52 PM)
 4  Event 1 (11/2/2006 1:35:38.932693 PM) through
 5  Event 2,084 (11/2/2006 1:36:33.261134 PM)
 6
 7  Capture between the headend and the Spectra (No TXB Installed)
 8  Baud RaTE = 2400
 9  uSING pERRIN'S COMPUTER
10  DCE = PC
11  DTE = Spectra
12  Download would have been a success if I hadn't stopped it.
13
14           1     0.000000   0.000000
15           2     1.320357   0.000000
16       1,       1: DCE       3    43.262720   0.000000 ff
17       1,       2: DCE       4    43.567539   0.304819 ff 01 00 57 00 00 58 EC_DOWNLOAD
18       1,       9: DCE      11    43.812538   0.220249 0f
19       1,      10: DCE      12    43.816683   0.004145 20
20       1,      11: DCE      13    43.820842   0.004159 00
21       1,      12: DCE      14    43.824981   0.004139 f2
22       1,      13: DCE      15    44.038343   0.213362
23       1,      14: DCE      16    44.038355   0.000012 00
24       1,      15: DCE      17    44.043236   0.004881 ff
25
26       1,       1:  DTE     18    44.046312   0.004881 e8
27
28       2,      16: DCE      19    44.259022   0.004881 ff 01 00 67 00 00 68 EC_SET_BAUD 2,400
29       2,      23: DCE      26    44.696419   0.004881 ff 01 00 67 00 00 68 EC_SET_BAUD 2,400
30       2,       2:  DTE     33    44.726129   0.004794 ff 01 00 68
31
32       3,      30: DCE      37    44.802797   0.064169 ff 01 00 65 00 00 66 EC_ECHO_MODE
33       3,       6:  DTE     44    44.832328   0.004308 ff 01 00 66
34
35       4,      37: DCE      48    44.865685   0.020859 53
36       4,      38: DCE      49    44.869825   0.004140 33
37       4,      10:  DTE     50    44.870243   0.004140 53
38       5,      39: DCE      51    44.873992   0.004140 34
39       5,      11:  DTE     52    44.874409   0.000417 33
40  <Echo Data>
41     142,     176: DCE     325    45.443080   0.002238 37
42     142,     148:  DTE    326    45.445008   0.002238 30
43     143,     177: DCE     327    45.447220   0.002238 33
44     143,     149:  DTE    328    45.449174   0.001954 37
45     144,     178: DCE     329    45.451387   0.002213 32
46     144,     150:  DTE    330    45.453342   0.002213 33
47     144,     151:  DTE    331    45.457509   0.004167 32
48
49     145,     179: DCE     332    45.712016   0.254507 ff 01 00 67 00 01 69 EC_SET_BAUD 4,800
```

```
 50     145,     152:  DTE    339    45.741545   0.004141 ff 01 00 69
 51
 52     146,     186: DCE     343    45.818314   0.064269 1f
 53     146,     187: DCE     344    45.822463   0.004149 80
 54     146,     188: DCE     345    45.826622   0.004159 08
 55     146,     189: DCE     346    45.830761   0.004139 f5
 56
 57     146,     156:  DTE    347    45.833114   0.004139 0f
 58     146,     157:  DTE    348    45.837274   0.004160 a8
 59
 60     147,     190: DCE     349    45.849799   0.012525 21
 61     147,     158:  DTE    350    45.852040   0.012525 41
 62  <Echo Data at 4800 baud>
 63     216,     259: DCE     487    46.136253   0.001510 a4
 64     216,     227:  DTE    488    46.139078   0.002825 a4
 65     217,     260: DCE     489    46.140580   0.001502 45
 66     217,     228:  DTE    490    46.143243   0.001502 a5
 67
 68     218,     261: DCE     491    46.378995   0.001502 ff 01 00 67 00 02 6a EC_SET_BAUD 9,600
 69     218,     229:  DTE    498    46.408550   0.004624 ff 01 00 6a
 70
 71     219,     268: DCE     502    46.486509   0.065461 38
 72     219,     269: DCE     503    46.490474   0.003965 fe
 73
 74     219,     233:  DTE    504    46.493965   0.003965 f8
 75
 76     220,     270: DCE     505    46.501421   0.003965 86
 77     220,     234:  DTE    506    46.502619   0.001198 86
 78  <Echo Data at 9600 Baud>
 79     252,     302: DCE     569    46.634772   0.002784 39
 80     252,     266:  DTE    570    46.636003   0.001231 39
 81     253,     303: DCE     571    46.638913   0.002910 62
 82     253,     267:  DTE    572    46.640294   0.002910 62
 83     254,     304: DCE     573    46.643080   0.002910 b9
 84     254,     268:  DTE    574    46.644468   0.001388 63
 85     255,     305: DCE     575    46.647220   0.002752 ff
 86     255,     269:  DTE    576    46.648628   0.002752 ff
 87
 88     256,     306: DCE     577    46.868240   0.002752 ff 01 00 67 00 03 6b EC_SET_BAUD 19,200
 89     256,     270:  DTE    584    46.897795   0.004637 ff 01 00 6b
 90
 91     257,     313: DCE     588    46.975009   0.064713 c6
 92     257,     274:  DTE    589    46.978704   0.064713 fe
 93     258,     314: DCE     590    46.982403   0.064713 4b
 94     258,     275:  DTE    591    46.983003   0.000600 4b
 95     259,     315: DCE     592    46.986403   0.003400 42
 96     259,     276:  DTE    593    46.986857   0.003400 42
 97  <Echo Data at 19,200 Baud>
 98     274,     330: DCE     622    47.048190   0.003671 42
 99     274,     291:  DTE    623    47.048896   0.000706 42
100     275,     331: DCE     624    47.052349   0.003453 b6
101     275,     292:  DTE    625    47.053054   0.003453 42
```

```
102
103     276,     332: DCE     626    47.269593   0.003453 ff 01 00 67 00 04 6c EC_SET_BAUD 38,400
104     276,     293:  DTE    633    47.299071   0.004550 ff 01 00 6c
105
106     277,     339: DCE     637    47.376078   0.064507 fb
107     277,     297:  DTE    638    47.377958   0.064507 6f
108     278,     340: DCE     639    47.380266   0.064507 42
109     278,     298:  DTE    640    47.382123   0.001857 42
110  <Echo Data at 38,400 Baud>
111     285,     347: DCE     653    47.408503   0.002395 18
112     285,     305:  DTE    654    47.410066   0.002395 8c
113     286,     348: DCE     655    47.412720   0.002395 c6
114     286,     306:  DTE    656    47.414018   0.001298 dc
115
116     287,     349: DCE     657    47.626025   0.212007 ff 01 00 67 00 05 6d EC_SET_BAUD 115,200
117     287,     307:  DTE    664    47.655450   0.004141 ff 01 00 6d 95
118
119     288,     356: DCE     669    47.734128   0.001205 2a
120     288,     312:  DTE    670    47.736889   0.001205 b4
121  <Echo data at 115,200 Baud>
122     289,     357: DCE     671    47.738085   0.001205 3a
123     289,     358: DCE     672    47.741861   0.001205 e1
124     289,     313:  DTE    673    47.741870   0.000009 07
125     289,     314:  DTE    674    47.745815   0.003945 ff
126
127     290,     359: DCE     675    47.952224   0.206409 ff 01 00 67 00 05 6d EC_SET_BAUD 115,200
128     290,     315:  DTE    682    47.981571   0.004142 ff 01 00 6d
129
130     291,     366: DCE     686    48.058383   0.064313 ff
131     291,     367: DCE     687    48.071317   0.012934 ff
132     291,     368: DCE     688    48.358495   0.287178 95
133     291,     369: DCE     689    48.362479   0.003984 59
134  <Download Data at 115,200 Baud>
135     291,     373: DCE     693    48.379545   0.003959 94
136     291,     374: DCE     694    48.383494   0.003949 53
137
138  There were a total of     2082 bytes transferred
139
140  There were a total of     1764 DCE bytes transferred
141  The first DCE byte came in at 43.262720 seconds from the start of data collection
142  The last DCE byte was at 54.328441 seconds from the start of data collection
143
144  There were a total of      318 DTE bytes transferred
145  The first DTE byte came in at 44.046312 seconds from the start of data collection
146  The last DTE byte was at 47.994070 seconds from the start of data collection
147
```

2.2.5.3 Details of the data start of a download

This is the start of the data transfer of download data. Note that the data is in "Motorola S Record" format. Each record starts with a "S" and ends with a cr (0x0D) byte.

This is a type "S0" record with 06 pairs of data bytes following it.

```
 1     151,     172:  DTE    608   100.440350   0.004141 ff 01 00 6a
 2
 3     152,     437: DCE     612   100.473695   0.020844 03
 4
 5     152,     176:  DTE    613   100.487434   0.020844 01
 6
 7                                                S  0  0  6  0  0  0  0  4  8  4  4  5  2
1  B  cr
 8     153,     438: DCE     614   100.508216   0.020844 53 30 30 36 30 30 30 30 34 38 34 34 35 32
31 42 0d
 9
10     153,     177:  DTE    631   100.771970   0.197179 01
11
12                                                S  3  4  5  0  0  0  0  0  0  0  0  0  0
0  0  0  6  4  3  0  0  0  0  0  7
13     154,     455: DCE     632   100.792786   0.020816 53 33 34 35 30 30 30 30 30 30 30 30 30
30 30 30 36 46 43 30 30 30 30 30 37
14
```

2.2.5.4 Fatal timeout errors in the middle of a download

On this fatal timeout, caused by unplugging the download cable, note that a total of 9 retrys, lines 4 ⟶ 12, were performed before quitting.

```
 1     178,    3980: DCE    4181   116.208871   0.004165 41
 2     178,    3981: DCE    4182   116.213013   0.004142 b0
 3
 4     178,     202:  DTE   4183   116.733497   0.004142 02
 5     178,     203:  DTE   4184   117.243159   0.509662 02
 6     178,     204:  DTE   4185   117.753472   0.510313 02
 7     178,     205:  DTE   4186   118.263612   0.510140 02
 8     178,     206:  DTE   4187   118.773576   0.509964 02
 9     178,     207:  DTE   4188   119.283576   0.510000 02
10     178,     208:  DTE   4189   119.793392   0.509816 02
11     178,     209:  DTE   4190   120.303705   0.510313 02
12     178,     210:  DTE   4191   120.814018   0.510313 02
```

2.2.5.5 Recovery following a timeout in the middle of a download

This shows the recovery to a forced time out in the middle of a download.

On line 2 is the last fully "good" message. It received a normal `LACK`. Then we have the start of the message, line 6, that was truncated in the middle by disconnecting the download cable.

In the timeout area, lines 8 ⟶ 13, there are several `LNACK`s, which are sent about every .5 second. When the cable is reconnected, line 16, the system starts to resend the message that was "broken" (line 6). It will have an `LACK` sent and the system continues on with no further problems.

```
 1                                                    S  3  4  5  0  0  0  0  0  4  0  0  9
 2     348,    6552: DCE    6955   223.858009   0.020653 53 33 34 35 30 30 30 30 30 34 30 30 39
 3     348,     404:  DTE   7098   224.463734   0.004167 01
 4
 5                                                    S  3  4  5  0  0  0  0  0  4  4  0  1
 6     349,    6695: DCE    7099   224.484555   0.004167 53 33 34 35 30 30 30 30 30 34 34 30 31
 7
 8     349,     405:  DTE   7158   225.238842   0.513285 02
 9     349,     406:  DTE   7159   225.748973   0.510131 02
10     349,     407:  DTE   7160   226.259048   0.510075 02
11     349,     408:  DTE   7161   226.769102   0.510054 02
12     349,     409:  DTE   7162   227.279102   0.510000 02
13     349,     410:  DTE   7163   227.789076   0.509974 02
14
15                                                    S  3  4  5  0  0  0  0  0  4  4  0  1
16     350,    6754: DCE    7164   227.809892   0.020816 53 33 34 35 30 30 30 30 30 34 34 30 31
17     350,     411:  DTE   7307   228.415584   0.004139 01
```

2.2.5.6 Ending sequence of a download

This is the normal end of a download sequence. Note here the sending of a "S7" record, line 7, to indicate the end by the PC. Which gets an immediate LACK, line 9, before the full message has been received.

The full last message is finished on line 12. Then the PC sends a LTERM, line 14, command to finish up the whole process. Which is given an LACK before the Spectra updates its flash and reboots.

A full interpreted dump of a typical, here the last, download data message is in Table 2, page 30.

```
 1                                                    S 3  4        0  1  2  cr
 2    2149,  263519: DCE  265728  1352.954333   0.004115 53 33 34 . . . 30 31 32 0d Full msg in
FULL_MESSAGE
 3
 4    2149,    2210:  DTE 265871  1353.559452   0.015250 01
 5
 6                                                    S  7  0  5  0
 7    2150,  263662: DCE  265872  1353.580198   0.020746 53 37 30 35 30
 8
 9    2150,    2211:  DTE 265877  1353.598199   0.004138 01
10
11                                                   0  0  1  C  0  0  0  3  9  cr
12    2151,  263667: DCE  265878  1353.601076   0.004138 30 30 31 43 30 30 30 33 39 0d
13
14    2151,  263677: DCE  265888  1353.642613   0.004138 85
15
16    2151,    2212:  DTE 265889  1353.656534   0.013921 01
17
```

Bytes	**Data**															
$1 \rightarrow 16$	S	3	4	5	0	0	0	1	F	F	3	0	F	F	0	0
---	53	33	34	35	30	30	30	31	46	46	43	30	46	46	30	30
$17 \rightarrow 32$	0	0	F	F	F	F	0	0	0	0	F	F	C	3	3	C
---	30	30	46	46	46	46	30	30	30	30	46	46	43	33	33	43
$33 \rightarrow 48$	3	C	C	3	3	3	C	C	3	3	C	C	3	3	C	C
---	33	43	43	33	33	33	43	43	33	33	43	43	33	33	43	43
$49 \rightarrow 64$	3	3	C	C	0	F	F	0	0	F	F	0	F	0	0	F
---	33	33	43	43	30	46	46	30	30	46	46	30	46	30	30	46
$65 \rightarrow 80$	F	0	0	F	C	C	3	3	C	C	3	3	C	C	3	3
---	46	30	30	46	43	43	33	33	43	43	33	33	43	43	33	33
$81 \rightarrow 96$	C	C	3	3	3	C	C	3	C	3	3	C	0	0	F	F
---	43	43	33	33	33	43	43	33	43	33	33	43	30	30	46	46
$97 \rightarrow 112$	F	F	0	0	0	0	F	F	F	F	0	0	0	0	0	0
---	46	46	30	30	30	30	46	46	46	46	30	30	30	30	30	30
$113 \rightarrow 128$	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
---	30	30	30	30	30	30	30	30	30	30	30	30	30	30	30	30
$129 \rightarrow 143$	0	0	0	0	0	0	0	0	0	0	0	0	1	2	cr	
---	30	30	30	30	30	30	30	30	30	30	30	30	31	32	0d	

Table 2: An interpreted version of the last data record of a download sequence for Spectra III, version 1.34, "FULL_MESSAGE"

2.3 Firmware Downloader Implementation (RJ-45 Download Protocol)

The software discussed in this section[9] is implemented in the Spectra III. As other units are designed and built, and equivalent functionality will be included. Commands are shown in Table 3, page 31.

The firmware downloader is designed to receive formatted Motorola S-records via the given STREAM interface and write the data portion of said records to display RAM. The records are received via a proprietary protocol exchange defined as follows:

1. Wait for the `LCCHAN` (clear channel) signal.
2. Allocate a buffer to receive the S-record data. If successful, transmit `LACK`, otherwise transmit `LALLOC` and restart.
3. Receive a record from the STREAM. Retries are allowed by utilizing `LNACK`. If the receive fails transmit an `LCOMM` and restart. If `LTERM` is received, go to step 6.
4. Write the received record to display RAM.
5. Transmit an `LACK` and go to line 3.
6. Load the routine ROMEXEC into RAM and execute it. This routine retrieves the received records from display RAM and writes them to flash.

Note that errors encountered during a firmware update may result in a application reboot with invalid code space. This has been addressed via LOADER_MAIN, which acts as an application backup in the case of code space corruption.

`LACK`	`0x01`	Acknowledgement
`LNACK`	`0x02`	Negative acknowledgement
`LCCHAN`	`0x03`	Clear channel
`LERASE`	`0x81`	Erase failure
`LWRITE`	`0x82`	Write failure
`LALLOC`	`0x83`	Memory allocation failure
`LCOMM`	`0x84`	Communications failure
`LTERM`	`0x85`	Termination request
`LCHIP`	`0x86`	Invalid chip id

Table 3. RJ-45 Port Download Command Set

[9] This protocol description has been copied from the source code for the Spectra III. A slightly different process is used on the ExSite, or at least there are additional commands in the protocol (Table 4, page 32).

2.3.1 ExSite Changes

For the ExSite the responses have been expanded:

LACK	0x01	Acknowledgement
LNACK	0x02	Negative acknowledgement
LCCHAN	0x03	Clear channel
LCONT	0x04	Continue Download
LDONE	0x05	All Done
LERASE	0x81	Erase failure
LWRITE	0x82	Write failure
LALLOC	0x83	Memory allocation failure
LCOMM	0x84	Communications failure
LTERM	0x85	Termination request
LCHIP	0x86	Invalid chip ID

Table 4. RJ-45 Port Download Command Set, ExSite version

2.4 Typical commands and their responses

In the following data capture, several commands are sent:

1. A Zoom In is sent for camera 1.
2. A Motion Stop is sent to stop the zooming.
3. A Call Preset 1 is sent.

```
DTE = D Protocol GlassKeyboard
DCE = Spectra III

FTS capture file: E:\Capture\newtest2.cfa (8/16/2006 10:16:21 AM)
Event 1 (8/16/2006 10:04:33.474165 AM) through
Event 33 (8/16/2006 10:05:58.512988 AM)

     1,       1:  DTE     0.000000   0.000000 ff 01 00 20 00 00 21   Zoom In
     2,       1: DCE      0.011001   0.004762 ff 01 00 21            Response
     2,       8:  DTE     0.219946   0.001041 ff 01 00 00 00 00 01   Motion Stop
     3,       5: DCE      0.413586   0.187405 ff 01 00 01            Response
     3,      15:  DTE    85.024799   0.001041 ff 01 00 07 00 01 09   Call Preset 1
     4,       9: DCE     85.035699   0.004666 ff 01 00 09            Response

There were a total of       33 bytes transferred

There were a total of       12 DCE bytes transferred
The first DCE byte came in at 0.011001 seconds from the start of data collection
The last DCE byte was at 85.038823 seconds from the start of data collection

There were a total of       21 DTE bytes transferred
```

```
The first DTE byte came in at 0.000000 seconds from the start of data collection
The last DTE byte was at 85.031033 seconds from the start of data collection
```

2.5 Typical QUERY command and its response

This is a data capture from a QUERY command and its response. The system was running at 2400 baud and the Spectra III SE was running rev 3.31 software.

Table Notes

The table consists of two parts:

- In the first part the various command/response bytes have been assembled to make the command/response easier to identify.
- In the second part is the semi-raw data returned by the data capture software.

At Pelco, we "post-process" the raw data capture files to change the date time stamp so as to have all times be relative to the start of the data capture and to get Δ times between bytes and messages.

Column headings are:

Msg # This is the message number. Numbers are sequential within each data source. I.e. there is one set of message numbers for each of DCE and DTE sources.

Byte # this is the total byte number within each data source.

DTE/DCE Byte This is the source identifier for each data source type. The abbreviations are: **DCE** = Data Communications Equipment, in this case the Spectra III and **DTE** = Data Terminal Equipment, which is a TXB that is under development.

Total Byte # This is the total byte number. I.e. it is the total of both DCE and DTE bytes.

Total Time This time has been normalized to start at 0.00000. The data capture software provides a date and time tag for each byte. The post-processing software deletes the date, knows that there 60 seconds in a minute and 60 minutes in an hour and knows how to work with the transition between noon (12:00) and one o'clock (01:00). This has been done to eliminate unexpected time jumps in the record.

Δ Time This is the time between this byte and the previous byte.

Data This is the hexadecimal value of the byte recorded.

Msg #	Byte #	DTE/DCE Source	Total Byte #	Total Time	Δ Time	Data
1	1	DTE	1	0.000000	0.000000	ff 01 00 45 00 00 46
2	1	DCE	8	0.029773	0.004778	ff 01 44 44 35 33 43 42 57 00 00 00 00 00 00 00 00 13

Msg #	Byte #	DTE/DCE Source	Total Byte #	Total Time	Δ Time	Data
1	1	DTE	1	0.000000	0.000000	ff
1	2	DTE	2	0.004162	0.004162	01
1	3	DTE	3	0.008329	0.004167	00
1	4	DTE	4	0.012496	0.004167	45
1	5	DTE	5	0.016661	0.004165	00
1	6	DTE	6	0.020828	0.004167	00
1	7	DTE	7	0.024995	0.004167	46
2	1	DCE	8	0.029773	0.004778	ff
2	2	DCE	9	0.033939	0.004166	01
2	3	DCE	10	0.038107	0.004168	44
2	4	DCE	11	0.042274	0.004167	44
2	5	DCE	12	0.046442	0.004168	35
2	6	DCE	13	0.050608	0.004166	33
2	7	DCE	14	0.054747	0.004139	43
2	8	DCE	15	0.058934	0.004187	42
2	9	DCE	16	0.063081	0.004147	57
2	10	DCE	17	0.067248	0.004167	00
2	11	DCE	18	0.071414	0.004166	00
2	12	DCE	19	0.075613	0.004199	00
2	13	DCE	20	0.079747	0.004134	00
2	14	DCE	21	0.083914	0.004167	00
2	15	DCE	22	0.088086	0.004172	00
2	16	DCE	23	0.092294	0.004208	00
2	17	DCE	24	0.096414	0.004120	00
2	18	DCE	25	0.100581	0.004167	13

To get the checksum for the QUERY response goes like this:

1. Add up all bytes in the QUERY response, except for the SYNC byte.
2. Add in the checksum from the QUERY command.
3. Use the lower eight bits of the sum as the checksum.

```
0x01 + 0x44 + 0x44 + 0x35 + 0x33 + 0x43 +
0x42 + 0x57 + 0x00 + 0x00 + 0x00 + 0x00 +
0x00 + 0x00 + 0x00 + 0x00 = 0x1CD
0x1CD + 0x46 = 0x213
0x213 & 0xFF = 0x13
```

2.6 Interpreting Pan and Tilt D Readout Replies

Pan and tilt angle values comes in in two bytes as degrees times 100 “hungrees”.

Position	D reads out as	Spectra displays as
90^o up	27000	90^o
45^o up	31500	45^o
Horizontal - 1^o	35900	1^o
Horizontal	000	0^o
45^o down	4500	-45^o
90^o down	9000	-90^o

Position Pointing direction of the enclosure/camera

D reads out as D protocol returned value for this angle

Spectra displays as What is displayed on the Spectra screen

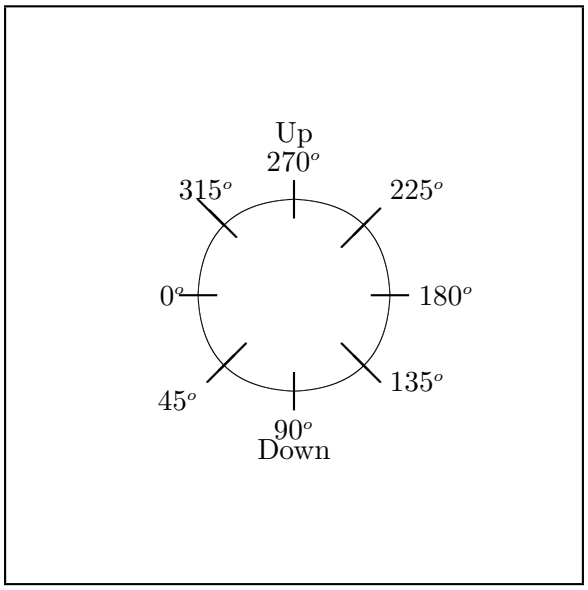

Figure 1. D Protocol Tilt Readout Positions

2.7 Sample Pan Angle Correction Software

In the below example several variables and functions are used:

1. Protocol Command Values:
   1.1. `SDcmnd1` and `SDcmnd2`, UNSIGNED CHARs are used to hold commands to the Spectra.
2. Calculated Intermediate Values:

2.1. `HR_offset` is a SIGNED INT which holds the results of asking the Spectra what the Azimuth offset is.

2.2. `HR_temp` is a SIGNED LONG which holds the result of modifying the reported value from the Spectra by the offset.

3. Macros/defines used:

3.1. `EC_EXTENDED_REPLY_LENGTH` is the length of a D Protocol reply that contains Azimuth or Elevation data. It is currently 7 (seven).

3.2. `DREPLY_DATA1` with an index value of 5 and

3.3. `DREPLY_DATA2` which has an index value of 6.

4. Arrays used:

4.1. `Dreply` is a 7 UNSIGNED CHAR to receive the Spectra reply into. The two positions used here are:

4.1.1. `DREPLY_DATA1` with an index value of 5 and

4.1.2. `DREPLY_DATA2` which has an index value of 6.

5. Functions called:

5.1. `SCheckSumD()`

5.2. `GetDReply()`

6. The results are in two UNSIGNED CHARS:

6.1. `HpanU` this is the upper half of the pan angle when modified by the Set Azimuth Zero value.

6.2. `HpanL` this is the lower half of the pan angle when modified by the Set Azimuth Zero value.

```
// Get pan angle offset from zero
SDcmnd1 = ECS_EVEREST_AZIMUTH_ZERO_OFFSET_QRY;
SDcmnd2 = EC_EVEREST;      // This is an Everest op-code
SCheckSumD(YES_REPLY);
GetDReply(EC_EXTENDED_REPLY_LENGTH);  // Put it in the reply buffer
HR_offset = ((DReply[DREPLY_DATA1]*256) + DReply[DREPLY_DATA2]);

// Get pan angle SDcmnd1, SDdata1 and SData2 don't change anymore
SDcmnd1 = 0x00;
SDcmnd2 = EC_QUERY_PAN;   // What is the current azimuth reading
SCheckSumD(YES_REPLY);
GetDReply(EC_EXTENDED_REPLY_LENGTH);  // Put it in the reply buffer

// Pan angle comes in in two bytes as degrees times 100 "hungrees"
// Value has to be rounded (i.e. that is why there is a "+ 50" here)
//
// The pan angle reported by an EC_QUERY_PAN command is not
// offset by the EC_SET_ZERO command. But the on screen display
// is. So here we have to modify the reported output by the
// changed pan offset value.
//
// If EC_SET_ZERO has been used to set pan zero to 25 degrees,
// and the on-screen display is now reading 50 degrees in pan,
// then the reply from a EC_QUERY_PAN command will be 75 degrees.
// In general we should have the angle reported to the outside
// world match what is seen on the screen. Thus there is logic to
// request the actual offset value and to use that in modifying
// the reported value so that it matches the on-screen value.
//
HR_temp  = ((DReply[DREPLY_DATA1]*256) + DReply[DREPLY_DATA2]);

HR_temp -= HR_offset;           // Get difference of real vs display
if (HR_temp < 0)                // Too small
{
    HR_temp += 36000;           // Yep, let it wrap up
}
HR_temp += 50;                  // Round
HR_temp /= 100;                 // Convert from hungrees to decimal
HpanU    = (unsigned char) (HR_temp/256);
HpanL    = HR_temp & 0xFF;
```

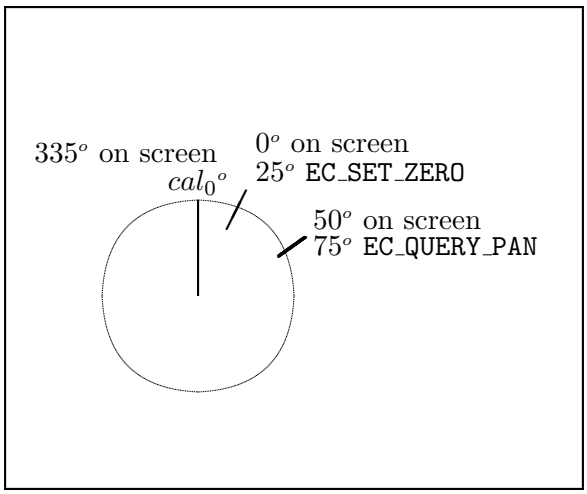

Figure 2. D Protocol Pan Readout Positions

2.8 Command Descriptions

This section is displayed twice. Once in alphabetical order and once in numerical order.

2.8.1 Alphabetically Ordered Descriptions

1. — —/—$_{10}$: This command op-code is not used.
2. **AA** 25_{10}/0x19: Alarm Acknowledge.
3. **AEM** 101_{10}/0x65: Activate Echo Mode.
4. **AF** 43_{10}/0x2B: Auto Focus.
5. **AGC** 47_{10}/0x2F: Automatic Gain Control.
6. **AG** 63_{10}/0x3F: Adjust Gain.
7. **AIL** 65_{10}/0x41: Adjust Auto Iris Level.
8. **AIP** 67_{10}/0x43: Adjust Auto Iris Peak.
9. **AI** 45_{10}/0x2D: Auto Iris, on/off.
10. **AWB** 51_{10}/0x33: Automatic White Balance.
11. **BLC** 49_{10}/0x31: Back Light Compensation.
12. **CA** 11_{10}/0x0B: Clear Auxiliary, range 1 $\longrightarrow$ 7.
13. **CP** 5_{10}/0x05: Clear Preset, see Section 7, page 70 for more information.
14. **CS** 23_{10}/0x17: Clear Screen of textual data.
15. **DL** 87_{10}/0x57: Down Load, used in Spectra III and newer units that support downloading of firmware. D Protocol only.
16. **D** 13_{10}/0x0D: Dummy, used for command debugging.
17. **E** 117_{10}/0x75: Everest Macro op-code, used in Spectra III, ExSite and Esprit units with software newer than 3.17. D Protocol only. Valid sub-op-codes are:
    17.1. ECS_EVEREST_AZIMUTH_ZERO_OFFSET_QRY 0x00
    17.2. ECS_EVEREST_AZIMUTH_ZERO_OFFSET_RSP 0x01
    17.3. ECS_EVEREST_SET_MAX_MAG 0x02
    17.4. ECS_EVEREST_MAX_MAG_QRY 0x03
    17.5. ECS_EVEREST_MAX_MAG_RSP 0x04
    17.6. ECS_EVEREST_ALARM_QRY 0x05
    17.7. ECS_EVEREST_ALARM_RSP 0x06
    17.8. ECS_EVEREST_DELETE_PATTERN 0x07
    17.9. ECS_EVEREST_SET_MAN_PAN_LEFT_LIMIT 0x08
    17.10. ECS_EVEREST_SET_MAN_PAN_RIGHT_LIMIT 0x09
    17.11. ECS_EVEREST_SET_SCAN_PAN_LEFT_LIMIT 0x0A
    17.12. ECS_EVEREST_SET_SCAN_PAN_RIGHT_LIMIT 0x0B
    17.13. ECS_EVEREST_LIMIT_QRY 0x0C
    17.14. ECS_EVEREST_LIMIT_RSP 0x0D
    17.15. ECS_EVEREST_ENABLE_LIMITS 0x0E
    17.16. ECS_EVEREST_DEFINED_PRESETS_QRY 0x0F
    17.17. ECS_EVEREST_DEFINED_PRESETS_RSP 0x10
    17.18. ECS_EVEREST_DEFINED_PATTERNS_QRY 0x11
    17.19. ECS_EVEREST_DEFINED_PATTERNS_RSP 0x12
18. **GOTO** 7_{10}/0x07: Goto indicated preset position. AKA "CALL". See Section 7, page 70 for more information.
19. **IOC** 255_{10}/0xFF: Invalid Op-Code, used in LRD units only.
20. **LLP** 57_{10}/0x39: Adjust Line Lock Phase Delay.
21. **PD** 53_{10}/0x35: Enable Phase Delay mode.
22. **PSta** 31_{10}/0x1F: Set Pattern Start, see Section 2.1.1, page 15.
23. **PSto** 33_{10}/0x21: Set Pattern Stop, see Section 2.1.1, page 15.
24. **PS** 71_{10}/0x47: Preset Scan.
25. **QDIR** 113_{10}/0x71: Query Diagnostic Information Response.
26. **QDI** 111_{10}/0x6F: Query Diagnostic Information.
27. **QDTR** 109_{10}/0x6D: Query Device Type Response.
28. **QDT** 107_{10}/0x6B: Query Device Type.
29. **QMR** 99_{10}/0x63: Query Magnification Response.
30. **QM** 97_{10}/0x61: Query Magnification.
31. **QPP** 81_{10}/0x51: Query Pan Position.
32. **QPR** 89_{10}/0x59: Query Pan Position Response.
33. **QTP** 83_{10}/0x53: Query Tilt Position.
34. **QTR** 91_{10}/0x5B: Query Tilt Position Response.
35. **QZP** 85_{10}/0x55: Query Zoom Position.
36. **QZR** 93_{10}/0x5D: Query Zoom Position Response.
37. **Q** 69_{10}/0x45: Query, does not examine the address field all units that decode this will respond immediately. Introduced in Spectra software starting in revision 2.06. Required for TXBs to operate. D Protocol only.
38. **RCD** 41_{10}/0x29: Reset Camera Defaults.

39. **RP** 35_{10}/`0x23`: Run Pattern, see Section 2.1.1, page 15.
40. **RS** 253_{10}/`0xFD`: Return Status, used in ERD97 and Intercept units with P Protocol. These `define`s are from the Intercept software, rev 3.07.
    40.1. `EC_STATUS 0xFD` Return status.
    40.2. `SSC_GET_FLA 0x01` Get flags.
    40.3. `SSC_GET_PRESET_DATA 0x02` Get preset data status.
    40.4. `SSC_GET_DESIRED_POSITION 0x03` Get desired position status.
    40.5. `SSC_GET_CURRENT_POSITION 0x04` Get current position status.
    40.6. `SSC_GET_VERSION 0x06` Get version/revision.
41. **Rst** 15_{10}/`0x0F`: Remote Reset.
42. **SA** 9_{10}/`0x09`: Set Auxiliary, range 1 $\longrightarrow$ 7.
43. **SDL** 109_{10}/`0x69`: Start Download.
44. **SFS** 39_{10}/`0x27`: Set Focus Speed.
45. **SM** 95_{10}/`0x5F`: Set Magnification.
46. **SOf** 29_{10}/`0x1D`: Zone Scan Off.
47. **SOn** 27_{10}/`0x1B`: Zone Scan On.
48. **SPP** 75_{10}/`0x4B`: Set Pan Position. Range is 0 $\longrightarrow$ 35999 in two bytes.
49. **SP** 3_{10}/`0x03`: Set Preset, see Section 7, page 70 for more information.
50. **SRBR** 103_{10}/`0x67`: Set Remote Baud Rate.
51. **SS** 55_{10}/`0x37`: Set Shutter Speed.
52. **STP** 77_{10}/`0x4D`: Set Tilt Position. Range is 0 $\longrightarrow$ 35999 in two bytes.
53. **SZP** 79_{10}/`0x4F`: Set Zoom Position.
54. **SZS** 37_{10}/`0x25`: Set Zoom Speed.
55. **SZeP** 73_{10}/`0x49`: Set Zero Position.
56. **VI** 115_{10}/`0x73`: Version Information. Valid responses are:
    56.1. `ECS_VERSION_INFO_MAIN_CPU_VERSION_QRY 0x00`
    56.2. `ECS_VERSION_INFO_MAIN_CPU_VERSION_RSP 0x01`
    56.3. `ECS_VERSION_INFO_MAIN_CPU_BUILD_QRY 0x02`
    56.4. `ECS_VERSION_INFO_MAIN_CPU_BUILD_RSP 0x03`
57. **WBMG** 61_{10}/`0x3D`: Adjust White Balance Magenta-Green.
58. **WBRB** 59_{10}/`0x3B`: Adjust White Balance Red-Blue.
59. **WC** 21_{10}/`0x15`: Write Character to Screen. Range 0 $\longrightarrow$ 39.
60. **XR** 1_{10}/`0x01`: Extended Response. Valid responses are:
    60.1. `ECS_STD_EXT_RESP_NACK 0x00`
    60.2. `ECS_STD_EXT_RESP_ACK 0x01`
61. **ZE** 19_{10}/`0x13`: Zone End. Range 1 $\longrightarrow$ 8.
62. **ZS** 17_{10}/`0x11`: Zone Start. Range 1 $\longrightarrow$ 8.

2.8.2 Numerically Ordered Descriptions

1. — —$_{10}$/—: This command op-code is not used.
2. **XR** 1_{10}/`0x01`: Extended Response. Valid responses are:
   2.1. `ECS_STD_EXT_RESP_NACK 0x00`
   2.2. `ECS_STD_EXT_RESP_ACK 0x01`
3. **SP** 3_{10}/`0x03`: Set Preset, see Section 7, page 70 for more information.
4. **CP** 5_{10}/`0x05`: Clear Preset, see Section 7, page 70 for more information.
5. **GOTO** 7_{10}/`0x07`: Goto indicated preset position. AKA "CALL". See Section 7, page 70 for more information.
6. **SA** 9_{10}/`0x09`: Set Auxiliary, range 1 ⟶ 7.
7. **CA** 11_{10}/`0x0B`: Clear Auxiliary, range 1 ⟶ 7.
8. **D** 13_{10}/`0x0D`: Dummy, used for command debugging.
9. **Rst** 15_{10}/`0x0F`: Remote Reset.
10. **ZS** 17_{10}/`0x11`: Zone Start. Range 1 ⟶ 8.
11. **ZE** 19_{10}/`0x13`: Zone End. Range 1 ⟶ 8.
12. **WC** 21_{10}/`0x15`: Write Character to Screen. Range 0 ⟶ 39.
13. **CS** 23_{10}/`0x17`: Clear Screen of textual data.
14. **AA** 25_{10}/`0x19`: Alarm Acknowledge.
15. **SOn** 27_{10}/`0x1B`: Zone Scan On.
16. **SOf** 29_{10}/`0x1D`: Zone Scan Off.
17. **PSta** 31_{10}/`0x1F`: Set Pattern Start, see Section 2.1.1, page 15.
18. **PSto** 33_{10}/`0x21`: Set Pattern Stop, see Section 2.1.1, page 15.
19. **RP** 35_{10}/`0x23`: Run Pattern, see Section 2.1.1, page 15.
20. **SZS** 37_{10}/`0x25`: Set Zoom Speed.
21. **SFS** 39_{10}/`0x27`: Set Focus Speed.
22. **RCD** 41_{10}/`0x29`: Reset Camera Defaults.
23. **AF** 43_{10}/`0x2B`: Auto Focus.
24. **AI** 45_{10}/`0x2D`: Auto Iris, on/off.
25. **AGC** 47_{10}/`0x2F`: Automatic Gain Control.
26. **BLC** 49_{10}/`0x31`: Back Light Compensation.
27. **AWB** 51_{10}/`0x33`: Automatic White Balance.
28. **PD** 53_{10}/`0x35`: Enable Phase Delay mode.
29. **SS** 55_{10}/`0x37`: Set Shutter Speed.
30. **LLP** 57_{10}/`0x39`: Adjust Line Lock Phase Delay.
31. **WBRB** 59_{10}/`0x3B`: Adjust White Balance Red-Blue.
32. **WBMG** 61_{10}/`0x3D`: Adjust White Balance Magenta-Green.
33. **AG** 63_{10}/`0x3F`: Adjust Gain.
34. **AIL** 65_{10}/`0x41`: Adjust Auto Iris Level.
35. **AIP** 67_{10}/`0x43`: Adjust Auto Iris Peak.
36. **Q** 69_{10}/`0x45`: Query, does not examine the address field all units that decode this will respond immediately. Introduced in Spectra software starting in revision 2.06. Required for TXBs to operate. D Protocol only.
37. **PS** 71_{10}/`0x47`: Preset Scan.
38. **SZeP** 73_{10}/`0x49`: Set Zero Position.
39. **SPP** 75_{10}/`0x4B`: Set Pan Position. Range is 0 ⟶ 35999 in two bytes.
40. **STP** 77_{10}/`0x4D`: Set Tilt Position. Range is 0 ⟶ 35999 in two bytes.
41. **SZP** 79_{10}/`0x4F`: Set Zoom Position.
42. **QPP** 81_{10}/`0x51`: Query Pan Position.
43. **QTP** 83_{10}/`0x53`: Query Tilt Position.
44. **QZP** 85_{10}/`0x55`: Query Zoom Position.
45. **DL** 87_{10}/`0x57`: Down Load, used in Spectra III and newer units that support downloading of firmware. D Protocol only.
46. **QPR** 89_{10}/`0x59`: Query Pan Position Response.
47. **QTR** 91_{10}/`0x5B`: Query Tilt Position Response.
48. **QZR** 93_{10}/`0x5D`: Query Zoom Position Response.
49. **SM** 95_{10}/`0x5F`: Set Magnification.
50. **QM** 97_{10}/`0x61`: Query Magnification.
51. **QMR** 99_{10}/`0x63`: Query Magnification Response.
52. **AEM** 101_{10}/`0x65`: Activate Echo Mode.
53. **SRBR** 103_{10}/`0x67`: Set Remote Baud Rate.
54. **QDT** 107_{10}/`0x6B`: Query Device Type.
55. **SDL** 109_{10}/`0x69`: Start Download.
56. **QDTR** 109_{10}/`0x6D`: Query Device Type Response.
57. **QDI** 111_{10}/`0x6F`: Query Diagnostic Information.
58. **QDIR** 113_{10}/`0x71`: Query Diagnostic Information Response.
59. **VI** 115_{10}/`0x73`: Version Information. Valid responses are:

59.1. `ECS_VERSION_INFO_MAIN_CPU_VERSION_QRY 0x00`

59.2. `ECS_VERSION_INFO_MAIN_CPU_VERSION_RSP 0x01`

59.3. `ECS_VERSION_INFO_MAIN_CPU_BUILD_QRY 0x02`

59.4. `ECS_VERSION_INFO_MAIN_CPU_BUILD_RSP 0x03`

60. **E** 117_{10}/`0x75`: Everest Macro op-code, used in Spectra III, ExSite and Esprit units with software newer than 3.17. D Protocol only. Valid sub-op-codes are:

60.1. `ECS_EVEREST_AZIMUTH_ZERO_OFFSET_QRY 0x00`

60.2. `ECS_EVEREST_AZIMUTH_ZERO_OFFSET_RSP 0x01`

60.3. `ECS_EVEREST_SET_MAX_MAG 0x02`

60.4. `ECS_EVEREST_MAX_MAG_QRY 0x03`

60.5. `ECS_EVEREST_MAX_MAG_RSP 0x04`

60.6. `ECS_EVEREST_ALARM_QRY 0x05`

60.7. `ECS_EVEREST_ALARM_RSP 0x06`

60.8. `ECS_EVEREST_DELETE_PATTERN 0x07`

60.9. `ECS_EVEREST_SET_MAN_PAN_LEFT_LIMIT 0x08`

60.10. `ECS_EVEREST_SET_MAN_PAN_RIGHT_LIMIT 0x09`

60.11. `ECS_EVEREST_SET_SCAN_PAN_LEFT_LIMIT 0x0A`

60.12. `ECS_EVEREST_SET_SCAN_PAN_RIGHT_LIMIT 0x0B`

60.13. `ECS_EVEREST_LIMIT_QRY 0x0C`

60.14. `ECS_EVEREST_LIMIT_RSP 0x0D`

60.15. `ECS_EVEREST_ENABLE_LIMITS 0x0E`

60.16. `ECS_EVEREST_DEFINED_PRESETS_QRY 0x0F`

60.17. `ECS_EVEREST_DEFINED_PRESETS_RSP 0x10`

60.18. `ECS_EVEREST_DEFINED_PATTERNS_QRY 0x11`

60.19. `ECS_EVEREST_DEFINED_PATTERNS_RSP 0x12`

61. **RS** 253_{10}/`0xFD`: Return Status, used in ERD97 and Intercept units with P Protocol. These `define`s are from the Intercept software, rev 3.07.

61.1. `EC_STATUS 0xFD` Return status.

61.2. `SSC_GET_FLA 0x01` Get flags.

61.3. `SSC_GET_PRESET_DATA 0x02` Get preset data status.

61.4. `SSC_GET_DESIRED_POSITION 0x03` Get desired position status.

61.5. `SSC_GET_CURRENT_POSITION 0x04` Get current position status.

61.6. `SSC_GET_VERSION 0x06` Get version/revision.

62. **IOC** 255_{10}/`0xFF`: Invalid Op-Code, used in LRD units only.

2.9 Op-code Names as Defined and Used on Various Systems

Note

In the following tables, the following abbreviations have the these meanings:

1. C: A column header for Coaxitron type commands.
2. D: A column header for D Protocol type commands.
3. `define`: This is the actual value that appears in either the source code "`.C`" or "`.H`" file.
4. **No**: This indicates that this command is not supported.
5. P: A column header for P Protocol type commands.
6. Value: This is the defined value.
7. Y: This indicates that this command in this mode is supported.
8. Y_{32}: This indicates that this command is supported in 32 bit Coaxitron mode only. It may be supported in other protocols.

define	Value	C	D	P	define	Value	C	D	P
EC_SET_PRESET	3_{10} 0x03	Y	Y	Y	EC_ALARM_ACK	25_{10} 0x19	**No**	**No**	**No**
EC_CLEAR_PRESET	5_{10} 0x05	Y	Y	Y	EC_ZONE_ON	27_{10} 0x1B	Y	Y	Y
EC_MOVE_PRESET	7_{10} 0x07	Y	Y	Y	EC_ZONE_OFF	29_{10} 0x1D	Y	Y	Y
EC_SET_AUX	9_{10} 0x09	Y	Y	Y	EC_START_RECORD	31_{10} 0x1F	Y	Y	Y
EC_CLEAR_AUX	11_{10} 0x0B	Y	Y	Y	EC_END_RECORD	33_{10} 0x21	Y	Y	Y
EC_DUMMY_1	13_{10} 0x0D	**No**	**No**	**No**	EC_START_PLAY	35_{10} 0x23	Y	Y	Y
EC_RESET	15_{10} 0x0F	Y	Y	Y	EC_ZOOM_SPEED	37_{10} 0x25	**No**	**No**	**No**
EC_ZONE_START	17_{10} 0x11	Y	Y	Y	EC_FOCUS_SPEED	39_{10} 0x27	**No**	**No**	**No**
EC_ZONE_END	19_{10} 0x13	Y	Y	Y	EC_QUERY	69_{10} 0x45	**No**	Y	**No**
EC_WRITE_CHAR	21_{10} 0x15	Y	Y	Y	EC_PRESET_SCAN	71_{10} 0x47	Y_{32}	Y	Y
EC_CLEAR_SCREEN	23_{10} 0x17	Y	Y	Y	—	—			

Table 5. Esprit 3012 protocol defines for rev 2.06

define	Value	C	D	P	define	Value	C	D	P
STANDARD_EXTENDED_-RESP	1_{10} 0x01	**No**	Y	**No**	EC_AWB	51_{10} 0x33	**No**	Y	Y
EV_NACK	0_{10} 0x00	**No**	Y	**No**	EC_DEVICE_PHASE	53_{10} 0x35	**No**	**No**	**No**
EV_ACK	1_{10} 0x01	**No**	Y	**No**	EC_SHUTTER_SPEED	55_{10} 0x37	**No**	Y	Y
EC_SET_PRESET	3_{10} 0x03	Y	Y	Y	EC_ADJUST_PHASE	57_{10} 0x39	**No**	Y	Y
EC_CLEAR_PRESET	5_{10} 0x05	Y	Y	Y	EC_ADJUST_RB_WB	59_{10} 0x3B	**No**	Y	Y
EC_MOVE_PRESET	7_{10} 0x07	Y	Y	Y	EC_ADJUST_MG_WB	61_{10} 0x3D	**No**	Y	Y
EC_SET_AUX	9_{10} 0x09	Y	Y	Y	EC_ADJUST_GAIN	63_{10} 0x3F	**No**	**No**	**No**
EC_CLEAR_AUX	11_{10} 0x0B	Y	Y	Y	EC_ADJUST_AI_LEVEL	65_{10} 0x41	**No**	Y	Y
EC_DUMMY_1	13_{10} 0x0D	**No**	**No**	**No**	EC_ADJUST_AI_PEAK	67_{10} 0x43	**No**	Y	Y
EC_RESET	15_{10} 0x0F	Y	Y	Y	EC_QUERY	69_{10} 0x45	**No**	Y	**No**
EC_ZONE_START	17_{10} 0x11	Y	Y	Y	EC_PRESET_SCAN	71_{10} 0x47	Y_{32}	Y	Y
EC_ZONE_END	19_{10} 0x13	Y	Y	Y	EC_SET_ZERO	73_{10} 0x49	**No**	Y	**No**
EC_WRITE_CHAR	21_{10} 0x15	Y	Y	Y	EC_SET_PAN	75_{10} 0x4B	**No**	Y	**No**
EC_CLEAR_SCREEN	23_{10} 0x17	Y	Y	Y	EC_SET_TILT	77_{10} 0x4D	**No**	Y	**No**
EC_ALARM_ACK	25_{10} 0x19	**No**	**No**	**No**	EC_SET_ZOOM	79_{10} 0x4F	**No**	**No**	**No**
EC_ZONE_ON	27_{10} 0x1B	Y	Y	Y	EC_QUERY_PAN	81_{10} 0x51	**No**	Y	**No**
EC_ZONE_OFF	29_{10} 0x1D	Y	Y	Y	EC_QUERY_TILT	83_{10} 0x53	**No**	Y	**No**
EC_START_RECORD	31_{10} 0x1F	Y	Y	Y	EC_PAN_RESP	89_{10} 0x59	**No**	Y	**No**
EC_END_RECORD	33_{10} 0x21	Y	Y	Y	EC_TILT_RESP	91_{10} 0x5B	**No**	Y	**No**
EC_START_PLAY	35_{10} 0x23	Y	Y	Y	EC_SET_MAG	95_{10} 0x5F	**No**	Y	**No**
EC_ZOOM_SPEED	37_{10} 0x25	Y	Y	Y	EC_QUERY_MAG	97_{10} 0x61	**No**	Y	**No**
EC_FOCUS_SPEED	39_{10} 0x27	**No**	**No**	**No**	EC_MAG_RESP	99_{10} 0x63	**No**	Y	**No**
EC_CAMERA_RESET	41_{10} 0x29	**No**	Y	Y	EC_VERSION_QUERY	115_{10} 0x73	**No**	Y	**No**
EC_AUTO_FOCUS	43_{10} 0x2B	**No**	Y	Y	EC_EVEREST_OPCODES	117_{10} 0x75	**No**	Y	**No**
EC_AUTO_IRIS	45_{10} 0x2D	**No**	Y	Y	EV_QUERY_AZ_OFFSET	0_{10} 0x00	**No**	Y	**No**
EC_AGC	47_{10} 0x2F	**No**	**No**	**No**	EV_AZ_OFFSET_RESP	1_{10} 0x01	**No**	Y	**No**
EC_BLC	49_{10} 0x31	**No**	Y	Y	—	—			

Table 6. Esprit protocol defines for rev 3.23

define	Value	C	D	P	define	Value	C	D	P
EC_STD_EXT_RESP	1_{10} 0x01	**No**	Y	**No**	EC_SHUTTER_SPEED	55_{10} 0x37	**No**	Y	Y
EC_STD_EXT_RESP sub opcodes					EC_ADJUST_PHASE	57_{10} 0x39	**No**	Y	Y
ECS_STD_EXT_RESP_NACK	0_{10} 0x00	**No**	Y	**No**	EC_ADJUST_RB_WB	59_{10} 0x3B	**No**	Y	Y
ECS_STD_EXT_RESP_ACK	1_{10} 0x01	**No**	Y	**No**	EC_ADJUST_MG_WB	61_{10} 0x3D	**No**	Y	Y
EC_SET_PRESET	3_{10} 0x03	Y	Y	Y	EC_ADJUST_GAIN	63_{10} 0x3F	**No**	**No**	**No**
EC_CLEAR_PRESET	5_{10} 0x05	Y	Y	Y	EC_ADJUST_AI_LEVEL	65_{10} 0x41	**No**	Y	Y
EC_MOVE_PRESET	7_{10} 0x07	Y	Y	Y	EC_ADJUST_AI_PEAK	67_{10} 0x43	**No**	Y	Y
EC_SET_AUX	9_{10} 0x09	Y	Y	Y	EC_QUERY	69_{10} 0x45	**No**	Y	**No**
EC_CLEAR_AUX	11_{10} 0x0B	Y	Y	Y	EC_PRESET_SCAN	71_{10} 0x47	Y_{32}	Y	Y
EC_DUMMY_1	13_{10} 0x0D	**No**	**No**	**No**	EC_SET_ZERO	73_{10} 0x49	**No**	Y	**No**
EC_RESET	15_{10} 0x0F	Y	Y	Y	EC_SET_PAN	75_{10} 0x4B	**No**	Y	**No**
EC_ZONE_START	17_{10} 0x11	Y	Y	Y	EC_SET_TILT	77_{10} 0x4D	**No**	Y	**No**
EC_ZONE_END	19_{10} 0x13	Y	Y	Y	EC_SET_ZOOM	79_{10} 0x4F	**No**	Y	**No**
EC_WRITE_CHAR	21_{10} 0x15	Y	Y	Y	EC_QUERY_PAN	81_{10} 0x51	**No**	Y	**No**
EC_CLEAR_SCREEN	23_{10} 0x17	Y	Y	Y	EC_QUERY_TILT	83_{10} 0x53	**No**	Y	**No**
EC_ALARM_ACK	25_{10} 0x19	**No**	**No**	**No**	EC_QUERY_ZOOM	85_{10} 0x55	**No**	Y	**No**
EC_ZONE_ON	27_{10} 0x1B	Y	Y	Y	EC_DOWNLOAD	87_{10} 0x57	**No**	Y	**No**
EC_ZONE_OFF	29_{10} 0x1D	Y	Y	Y	EC_PAN_RESP	89_{10} 0x59	**No**	Y	**No**
EC_START_RECORD	31_{10} 0x1F	Y	Y	Y	EC_TILT_RESP	91_{10} 0x5B	**No**	Y	**No**
EC_END_RECORD	33_{10} 0x21	Y	Y	Y	EC_ZOOM_RESP	93_{10} 0x5D	**No**	Y	**No**
EC_START_PLAY	35_{10} 0x23	Y	Y	Y	EC_SET_MAG	95_{10} 0x5F	**No**	Y	**No**
EC_ZOOM_SPEED	37_{10} 0x25	Y	Y	Y	EC_QUERY_MAG	97_{10} 0x61	**No**	Y	**No**
EC_FOCUS_SPEED	39_{10} 0x27	**No**	**No**	**No**	EC_MAG_RESP	99_{10} 0x63	**No**	Y	**No**
EC_CAMERA_RESET	41_{10} 0x29	**No**	Y	Y	EC_ECHO_MODE	101_{10} 0x65	**No**	Y	**No**
EC_AUTO_FOCUS	43_{10} 0x2B	**No**	Y	Y	EC_SET_BAUD	103_{10} 0x67	**No**	Y	**No**
EC_AUTO_IRIS	45_{10} 0x2D	**No**	Y	Y	EC_START_DOWNLOAD	105_{10} 0x69	**No**	Y	**No**
EC_AGC	47_{10} 0x2F	**No**	**No**	**No**	EC_QUERY_DEV_TYPE	107_{10} 0x6B	**No**	Y	**No**
EC_BLC	49_{10} 0x31	**No**	Y	Y	EC_QUERY_DEV_TYPE_RESP	109_{10} 0x6D	**No**	Y	**No**
EC_AWB	51_{10} 0x33	**No**	Y	Y	EC_QUERY_DIAG_INFO	111_{10} 0x6F	**No**	Y	**No**
EC_DEVICE_PHASE	53_{10} 0x35	**No**	**No**	**No**	EC_QUERY_DIAG_INFO_RESP	113_{10} 0x71	**No**	Y	**No**

Table 7. ExSite protocol defines for rev 1.13, part 1 of 2

define	Value	C	D	P
EC_VERSION_INFO	115_{10} 0x73	**No**	Y	**No**
EC_VERSION_INFO sub opcodes				
ECS_VERSION_INFO_MAIN_CPU_VERSION_QRY	0_{10} 0x00	**No**	Y	**No**
ECS_VERSION_INFO_MAIN_CPU_VERSION_RSP	1_{10} 0x01	**No**	Y	**No**
ECS_VERSION_INFO_MAIN_CPU_BUILD_QRY	2_{10} 0x02	**No**	Y	**No**
ECS_VERSION_INFO_MAIN_CPU_BUILD_RSP	3_{10} 0x03	**No**	Y	**No**
EC_EVEREST	117_{10} 0x75	**No**	Y	**No**
EC_EVEREST sub opcodes				
ECS_EVEREST_AZIMUTH_ZERO_OFFSET_QRY	0_{10} 0x00	**No**	Y	**No**
ECS_EVEREST_AZIMUTH_ZERO_OFFSET_RSP	1_{10} 0x01	**No**	Y	**No**
ECS_EVEREST_SET_MAX_MAG	2_{10} 0x02	**No**	Y	**No**
ECS_EVEREST_MAX_MAG_QRY	3_{10} 0x03	**No**	Y	**No**
ECS_EVEREST_MAX_MAG_RSP	4_{10} 0x04	**No**	Y	**No**
ECS_EVEREST_ALARM_QRY	5_{10} 0x05	**No**	Y	**No**
ECS_EVEREST_ALARM_RSP	6_{10} 0x06	**No**	Y	**No**
ECS_EVEREST_DELETE_PATTERN	7_{10} 0x07	**No**	Y	**No**
ECS_EVEREST_SET_MAN_PAN_LEFT_LIMIT	8_{10} 0x08	**No**	Y	**No**
ECS_EVEREST_SET_MAN_PAN_RIGHT_LIMIT	9_{10} 0x09	**No**	Y	**No**
ECS_EVEREST_SET_SCAN_PAN_LEFT_LIMIT	10_{10} 0x0A	**No**	Y	**No**
ECS_EVEREST_SET_SCAN_PAN_RIGHT_LIMIT	11_{10} 0x0B	**No**	Y	**No**
ECS_EVEREST_LIMIT_QRY	12_{10} 0x0C	**No**	Y	**No**
ECS_EVEREST_LIMIT_RSP	13_{10} 0x0D	**No**	Y	**No**
ECS_EVEREST_ENABLE_LIMITS	14_{10} 0x0E	**No**	Y	**No**
ECS_EVEREST_DEFINED_PRESETS_QRY	15_{10} 0x0F	**No**	Y	**No**
ECS_EVEREST_DEFINED_PRESETS_RSP	16_{10} 0x10	**No**	Y	**No**
ECS_EVEREST_DEFINED_PATTERNS_QRY	17_{10} 0x11	**No**	Y	**No**
ECS_EVEREST_DEFINED_PATTERNS_RSP	18_{10} 0x12	**No**	Y	**No**
Extended command data				
ECD_EVEREST_LIMIT_MAN_LEFT_PAN	0_{10} 0x00	**No**	Y	**No**
ECD_EVEREST_LIMIT_MAN_RIGHT_PAN	1_{10} 0x01	**No**	Y	**No**
ECD_EVEREST_LIMIT_SCAN_LEFT_PAN	2_{10} 0x02	**No**	Y	**No**
ECD_EVEREST_LIMIT_SCAN_RIGHT_PAN	3_{10} 0x03	**No**	Y	**No**
Limit Value Query Constants				
Enable limits constants				
ECD_EVEREST_DISABLE_LIMITS	0_{10} 0x00	**No**	Y	**No**
ECD_EVEREST_ENABLE_LIMITS	1_{10} 0x01	**No**	Y	**No**

Table 8. ExSite protocol defines for rev 1.13, part 2 of 2

define	Value	C	D	P	define	Value	C	D	P
EC_SET_PRESET	3_{10} 0x03	Y	Y	Y	EC_CLEAR_SCREEN	23_{10} 0x17	Y	Y	Y
EC_CLEAR_PRESET	5_{10} 0x05	Y	Y	Y	EC_ZONE_ON	27_{10} 0x1B	Y	Y	Y
EC_MOVE_PRESET	7_{10} 0x07	Y	Y	Y	EC_ZONE_OFF	29_{10} 0x1D	Y	Y	Y
EC_SET_AUX	9_{10} 0x09	Y	Y	Y	EC_START_RECORD	31_{10} 0x1F	Y	Y	Y
EC_CLEAR_AUX	11_{10} 0x0B	Y	Y	Y	EC_END_RECORD	33_{10} 0x21	Y	Y	Y
EC_RESET	15_{10} 0x0F	Y	Y	Y	EC_START_PLAY	35_{10} 0x23	Y	Y	Y
EC_ZONE_START	17_{10} 0x11	Y	Y	Y	EC_ZOOM_SPEED	37_{10} 0x25	Y	Y	Y
EC_ZONE_END	19_{10} 0x13	Y	Y	Y	EC_FOCUS_SPEED	39_{10} 0x27	Y	Y	Y
EC_WRITE_CHAR	21_{10} 0x15	Y	Y	Y	EC_STATUS	253_{10} 0xFD	**No**	**No**	Y
Aux 2 = Auto Scan		Y	Y	Y	Aux 4 = Frame Scan		Y	Y	Y
Aux 3 = Random Scan		Y	Y	Y	—				

Table 9. Intercept protocol defines for rev 3.08

define	Value	define	Value
CMD_PRS_SET	3_{10}/0x03	CMD_AUX_CLR	11_{10}/0x0B
CMD_PRS_CLR	5_{10}/0x05	CMD_RMT_RESET	15_{10}/0x0F
CMD_PRS_GOTO	7_{10}/0x07	CMD_INVALID	255_{10}/0xFF
CMD_AUX_SET	9_{10}/0x09	—	—

Table 10. LRD D Protocol defines for rev 2.00

Byte 1		Byte 2	
define	Value	define	Value
CMD_AUX_4	192_{10}/0xC0	CMD_FOCUS_FAR	128_{10}/0x80
CMD_AUX_3	64_{10}/0x40	CMD_ZOOM_WIDE	64_{10}/0x40
CMD_AUX_1	160_{10}/0xA0	CMD_ZOOM_TELE	32_{10}/0x20
CMD_AUX_2	32_{10}/0x20	CMD_TILT_DOWN	16_{10}/0x10
CMD_SCAN_AUTO	144_{10}/0x90	CMD_TILT_UP	8_{10}/0x08
CMD_SCAN_MAN	16_{10}/0x10	CMD_PAN_LEFT	4_{10}/0x04
CMD_CAM_ON	136_{10}/0x88	CMD_PAN_RIGHT	2_{10}/0x02
CMD_CAM_OFF	8_{10}/0x08	CMD_PRS_SCAN	1_{10}/0x01
CMD_IRIS_CLOSE	4_{10}/0x04	—	—
CMD_IRIS_OPEN	2_{10}/0x02	—	—
CMD_FOCUS_NEAR	1_{10}/0x01	—	—

Table 11. LRD Coaxitron 15-bit protocol defines for rev 2.00

define	Value	define	Value
SETPRESET	3_{10}/0x03	SETAUXILIARY	9_{10}/0x09
CLEARPERSET	5_{10}/0x05	CLEARAUXILIARY	11_{10}/0x0B
GOTOPRESET	7_{10}/0x07	QUERY	69_{10}/0x45

Table 12. MRD D Protocol and P Protocol defines

define	Value	C	D	P	define	Value	C	D	P
EC_SET_PRESET	3_{10} 0x03	Y	Y	Y	EC_ZOOM_SPEED	37_{10} 0x25	Y	Y	Y
EC_CLEAR_PRESET	5_{10} 0x05	Y	Y	Y	EC_FOCUS_SPEED	39_{10} 0x27	**No**	Y	Y
EC_MOVE_PRESET	7_{10} 0x07	Y	Y	Y	EC_CAMERA_RESET	41_{10} 0x29	**No**	Y	Y
EC_SET_AUX	9_{10} 0x09	Y	Y	Y	EC_AUTO_FOCUS	43_{10} 0x2B	**No**	Y	Y
EC_CLEAR_AUX	11_{10} 0x0B	Y	Y	Y	EC_AUTO_IRIS	45_{10} 0x2D	**No**	Y	Y
EC_DUMMY_1	13_{10} 0x0D	**No**	**No**	**No**	EC_AGC	47_{10} 0x2F	**No**	Y	Y
EC_RESET	15_{10} 0x0F	Y	Y	Y	EC_BLC	49_{10} 0x31	**No**	Y	Y
EC_ZONE_START	17_{10} 0x11	Y	Y	Y	EC_AWB	51_{10} 0x33	**No**	Y	Y
EC_ZONE_END	19_{10} 0x13	Y	Y	Y	EC_DEVICE_PHASE	53_{10} 0x35	**No**	**No**	**No**
EC_WRITE_CHAR	21_{10} 0x15	Y	Y	Y	EC_SHUTTER_SPEED	55_{10} 0x37	**No**	Y	Y
EC_CLEAR_SCREEN	23_{10} 0x17	Y	Y	Y	EC_ADJUST_PHASE	57_{10} 0x39	**No**	Y	Y
EC_ALARM_ACK	25_{10} 0x19	**No**	**No**	**No**	EC_ADJUST_RB_WB	59_{10} 0x3B	**No**	Y	Y
EC_ZONE_ON	27_{10} 0x1B	Y	Y	Y	EC_ADJUST_MG_WB	61_{10} 0x3D	**No**	Y	Y
EC_ZONE_OFF	29_{10} 0x1D	Y	Y	Y	EC_ADJUST_GAIN	63_{10} 0x3F	**No**	Y	Y
EC_START_RECORD	31_{10} 0x1E	Y	Y	Y	EC_ADJUST_AI_LEVEL	65_{10} 0x41	**No**	Y	Y
EC_END_RECORD	33_{10} 0x21	Y	Y	Y	EC_ADJUST_AI_PEAK	67_{10} 0x43	**No**	Y	Y
EC_START_PLAY	35_{10} 0x23	Y	Y	Y	—	—			

Table 13. Spectra I protocol defines for rev 2.04

define	Value	C	D	P	define	Value	C	D	P
EC_SET_PRESET	3_{10} 0x03	Y	Y	Y	EC_FOCUS_SPEED	39_{10} 0x27	**No**	**No**	**No**
EC_CLEAR_PRESET	5_{10} 0x05	Y	Y	Y	EC_CAMERA_RESET	41_{10} 0x29	**No**	Y	Y
EC_MOVE_PRESET	7_{10} 0x07	Y	Y	Y	EC_AUTO_FOCUS	43_{10} 0x2B	**No**	Y	Y
EC_SET_AUX	9_{10} 0x09	Y	Y	Y	EC_AUTO_IRIS	45_{10} 0x2D	**No**	Y	Y
EC_CLEAR_AUX	11_{10} 0x0B	Y	Y	Y	EC_AGC	47_{10} 0x2F	**No**	**No**	**No**
EC_DUMMY_1	13_{10} 0x0D	**No**	**No**	**No**	EC_BLC	49_{10} 0x31	**No**	Y	Y
EC_RESET	15_{10} 0x0F	Y	Y	Y	EC_AWB	51_{10} 0x33	**No**	Y	Y
EC_ZONE_START	17_{10} 0x11	Y	Y	Y	EC_DEVICE_PHASE	53_{10} 0x35	**No**	**No**	**No**
EC_ZONE_END	19_{10} 0x13	Y	Y	Y	EC_SHUTTER_SPEED	55_{10} 0x37	**No**	Y	Y
EC_WRITE_CHAR	21_{10} 0x15	Y	Y	Y	EC_ADJUST_PHASE	57_{10} 0x39	**No**	Y	Y
EC_CLEAR_SCREEN	23_{10} 0x17	Y	Y	Y	EC_ADJUST_RB_WB	59_{10} 0x3B	**No**	Y	Y
EC_ALARM_ACK	25_{10} 0x19	**No**	**No**	**No**	EC_ADJUST_MG_WB	61_{10} 0x3D	**No**	Y	Y
EC_ZONE_ON	27_{10} 0x1B	Y	Y	Y	EC_ADJUST_GAIN	63_{10} 0x3F	**No**	**No**	**No**
EC_ZONE_OFF	29_{10} 0x1D	Y	Y	Y	EC_ADJUST_AI_LEVEL	65_{10} 0x41	**No**	Y	Y
EC_START_RECORD	31_{10} 0x1F	Y	Y	Y	EC_ADJUST_AI_PEAK	67_{10} 0x43	**No**	Y	Y
EC_END_RECORD	33_{10} 0x21	Y	Y	Y	EC_QUERY	69_{10} 0x45	**No**	Y	**No**
EC_START_PLAY	35_{10} 0x23	Y	Y	Y	EC_PRESET_SCAN	71_{10} 0x47	Y_{32}	Y	Y
EC_ZOOM_SPEED	37_{10} 0x25	Y	Y	Y	—	—			

Table 14. Spectra II protocol defines for rev 3.44

define	Value	C	D	P	define	Value	C	D	P
EC_SET_PRESET	3_{10} 0x03	Y	Y	Y	EC_FOCUS_SPEED	39_{10} 0x27	**No**	**No**	**No**
EC_CLEAR_PRESET	5_{10} 0x05	Y	Y	Y	EC_CAMERA_RESET	41_{10} 0x29	**No**	Y	Y
EC_MOVE_PRESET	7_{10} 0x07	Y	Y	Y	EC_AUTO_FOCUS	43_{10} 0x2B	**No**	Y	Y
EC_SET_AUX	9_{10} 0x09	Y	Y	Y	EC_AUTO_IRIS	45_{10} 0x2D	**No**	Y	Y
EC_CLEAR_AUX	11_{10} 0x0B	Y	Y	Y	EC_AGC	47_{10} 0x2F	**No**	Y	Y
EC_DUMMY_1	13_{10} 0x0D	**No**	**No**	**No**	EC_BLC	49_{10} 0x31	**No**	Y	Y
EC_RESET	15_{10} 0x0F	Y	Y	Y	EC_AWB	51_{10} 0x33	**No**	Y	Y
EC_ZONE_START	17_{10} 0x11	Y	Y	Y	EC_DEVICE_PHASE	53_{10} 0x35	**No**	**No**	**No**
EC_ZONE_END	19_{10} 0x13	Y	Y	Y	EC_SHUTTER_SPEED	55_{10} 0x37	**No**	Y	Y
EC_WRITE_CHAR	21_{10} 0x15	Y	Y	Y	EC_ADJUST_PHASE	57_{10} 0x39	**No**	Y	Y
EC_CLEAR_SCREEN	23_{10} 0x17	Y	Y	Y	EC_ADJUST_RB_WB	59_{10} 0x3B	**No**	Y	Y
EC_ALARM_ACK	25_{10} 0x19	**No**	**No**	**No**	EC_ADJUST_MG_WB	61_{10} 0x3D	**No**	Y	Y
EC_ZONE_ON	27_{10} 0x1B	Y	Y	Y	EC_ADJUST_GAIN	63_{10} 0x3F	**No**	Y	Y
EC_ZONE_OFF	29_{10} 0x1D	Y	Y	Y	EC_ADJUST_AI_LEVEL	65_{10} 0x41	**No**	Y	Y
EC_START_RECORD	31_{10} 0x1F	Y	Y	Y	EC_ADJUST_AI_PEAK	67_{10} 0x43	**No**	Y	Y
EC_END_RECORD	33_{10} 0x21	Y	Y	Y	EC_QUERY	69_{10} 0x45	**No**	Y	**No**
EC_START_PLAY	35_{10} 0x23	Y	Y	Y	EC_PRESET_SCAN	71_{10} 0x47	Y_{32}	Y	Y
EC_ZOOM_SPEED	37_{10} 0x25	Y	Y	Y	—	—			

Table 15. Spectra II Lite protocol defines for rev 4.40

define	Value	C	D	P	define	Value	C	D	P
EC_STD_EXT_RESP	1_{10} 0x01	**No**	Y	**No**	EC_DEVICE_PHASE	53_{10} 0x35	**No**	**No**	**No**
EC_STD_EXT_RESP Sub Opcodes					EC_SHUTTER_SPEED	55_{10} 0x37	**No**	Y	Y
ECS_STD_EXT_RESP_NACK	0_{10} 0x00	**No**	Y	**No**	EC_ADJUST_PHASE	57_{10} 0x39	**No**	Y	Y
ECS_STD_EXT_RESP_ACK	1_{10} 0x01	**No**	Y	**No**	EC_ADJUST_RB_WB	59_{10} 0x3B	**No**	Y	Y
EC_SET_PRESET	3_{10} 0x03	Y	Y	Y	EC_ADJUST_MG_WB	61_{10} 0x3D	**No**	Y	Y
EC_CLEAR_PRESET	5_{10} 0x05	Y	Y	Y	EC_ADJUST_GAIN	63_{10} 0x3F	**No**	**No**	**No**
EC_MOVE_PRESET	7_{10} 0x07	Y	Y	Y	EC_ADJUST_AI_LEVEL	65_{10} 0x41	**No**	Y	Y
EC_SET_AUX	9_{10} 0x09	Y	Y	Y	EC_ADJUST_AI_PEAK	67_{10} 0x43	**No**	Y	Y
EC_CLEAR_AUX	11_{10} 0x0B	Y	Y	Y	EC_QUERY	69_{10} 0x45	**No**	Y	**No**
EC_DUMMY_1	13_{10} 0x0D	**No**	**No**	**No**	EC_PRESET_SCAN	71_{10} 0x47	Y_{32}	Y	Y
EC_RESET	15_{10} 0x0F	Y	Y	Y	EC_SET_ZERO	73_{10} 0x49	**No**	Y	**No**
EC_ZONE_START	17_{10} 0x11	Y	Y	Y	EC_SET_PAN	75_{10} 0x4B	**No**	Y	**No**
EC_ZONE_END	19_{10} 0x13	Y	Y	Y	EC_SET_TILT	77_{10} 0x4D	**No**	Y	**No**
EC_WRITE_CHAR	21_{10} 0x15	Y	Y	Y	EC_SET_ZOOM	79_{10} 0x4F	**No**	Y	**No**
EC_CLEAR_SCREEN	23_{10} 0x17	Y	Y	Y	EC_QUERY_PAN	81_{10} 0x51	**No**	Y	**No**
EC_ALARM_ACK	25_{10} 0x19	**No**	**No**	**No**	EC_QUERY_TILT	83_{10} 0x53	**No**	Y	**No**
EC_ZONE_ON	27_{10} 0x1B	Y	Y	Y	EC_QUERY_ZOOM	85_{10} 0x55	**No**	Y	**No**
EC_ZONE_OFF	29_{10} 0x1D	Y	Y	Y	EC_DOWNLOAD	87_{10} 0x57	**No**	Y	**No**
EC_START_RECORD	31_{10} 0x1F	Y	Y	Y	EC_PAN_RESP	89_{10} 0x59	**No**	Y	**No**
EC_END_RECORD	33_{10} 0x21	Y	Y	Y	EC_TILT_RESP	91_{10} 0x5B	**No**	Y	**No**
EC_START_PLAY	35_{10} 0x23	Y	Y	Y	EC_ZOOM_RESP	93_{10} 0x5D	**No**	Y	**No**
EC_ZOOM_SPEED	37_{10} 0x25	Y	Y	Y	EC_SET_MAG	95_{10} 0x5F	**No**	Y	**No**
EC_FOCUS_SPEED	39_{10} 0x27	**No**	**No**	**No**	EC_QUERY_MAG	97_{10} 0x61	**No**	Y	**No**
EC_CAMERA_RESET	41_{10} 0x29	**No**	Y	Y	EC_MAG_RESP	99_{10} 0x63	**No**	Y	**No**
EC_AUTO_FOCUS	43_{10} 0x2B	**No**	Y	Y	EC_ECHO_MODE	101_{10} 0x65	**No**	Y	**No**
EC_AUTO_IRIS	45_{10} 0x2D	**No**	Y	Y	EC_SET_BAUD	103_{10} 0x67	**No**	Y	**No**
EC_AGC	47_{10} 0x2F	**No**	**No**	**No**	EC_START_DOWNLOAD	105_{10} 0x69	**No**	Y	**No**
EC_BLC	49_{10} 0x31	**No**	Y	Y	EC_QUERY_DEV_TYPE	107_{10} 0x6B	**No**	Y	**No**
EC_AWB	51_{10} 0x33	**No**	Y	Y	EC_QUERY_DEV_TYPE_RESP	109_{10} 0x6D	**No**	Y	**No**

Table 16. Spectra III protocol defines for rev 1.27, part 1 of 2

define	Value	C	D	P
EC_QUERY_DIAG_INFO	111_{10} 0x6F	**No**	Y	**No**
EC_QUERY_DIAG_INFO_RESP	113_{10} 0x71	**No**	Y	**No**
EC_VERSION_INFO	115_{10} 0x73	**No**	Y	**No**
EC_VERSION_INFO Sub Opcodes				
ECS_VERSION_INFO_MAIN_CPU_VERSION_QRY	0_{10} 0x00	**No**	Y	**No**
ECS_VERSION_INFO_MAIN_CPU_VERSION_RSP	1_{10} 0x01	**No**	Y	**No**
ECS_VERSION_INFO_MAIN_CPU_BUILD_QRY	2_{10} 0x02	**No**	Y	**No**
ECS_VERSION_INFO_MAIN_CPU_BUILD_RSP	3_{10} 0x03	**No**	Y	**No**
EC_EVEREST	117_{10} 0x75	**No**	Y	**No**
EC_EVEREST Sub Opcodes				
ECS_EVEREST_AZIMUTH_ZERO_OFFSET_QRY	0_{10} 0x00	**No**	Y	**No**
ECS_EVEREST_AZIMUTH_ZERO_OFFSET_RSP	1_{10} 0x01	**No**	Y	**No**
ECS_EVEREST_SET_MAX_MAG	2_{10} 0x02	**No**	Y	**No**
ECS_EVEREST_MAX_MAG_QRY	3_{10} 0x03	**No**	Y	**No**
ECS_EVEREST_MAX_MAG_RSP	4_{10} 0x04	**No**	Y	**No**
ECS_EVEREST_ALARM_QRY	5_{10} 0x05	**No**	Y	**No**
ECS_EVEREST_ALARM_RSP	6_{10} 0x06	**No**	Y	**No**
ECS_EVEREST_DELETE_PATTERN	7_{10} 0x07	**No**	Y	**No**
ECS_EVEREST_SET_MAN_PAN_LEFT_LIMIT	8_{10} 0x08	**No**	Y	**No**
ECS_EVEREST_SET_MAN_PAN_RIGHT_LIMIT	9_{10} 0x09	**No**	Y	**No**
ECS_EVEREST_SET_SCAN_PAN_LEFT_LIMIT	10_{10} 0x0A	**No**	Y	**No**
ECS_EVEREST_SET_SCAN_PAN_RIGHT_LIMIT	11_{10} 0x0B	**No**	Y	**No**
ECS_EVEREST_LIMIT_QRY	12_{10} 0x0C	**No**	Y	**No**
ECS_EVEREST_LIMIT_RSP	13_{10} 0x0D	**No**	Y	**No**
ECS_EVEREST_ENABLE_LIMITS	14_{10} 0x0E	**No**	Y	**No**
ECS_EVEREST_DEFINED_PRESETS_QRY	15_{10} 0x0F	**No**	Y	**No**
ECS_EVEREST_DEFINED_PRESETS_RSP	16_{10} 0x10	**No**	Y	**No**
ECS_EVEREST_DEFINED_PATTERNS_QRY	17_{10} 0x11	**No**	Y	**No**
ECS_EVEREST_DEFINED_PATTERNS_RSP	18_{10} 0x12	**No**	Y	**No**
Limit Value Query Constants				
ECD = Extended Command Data				
ECD_EVEREST_LIMIT_MAN_LEFT_PAN	0_{10} 0x00	**No**	Y	**No**
ECD_EVEREST_LIMIT_MAN_RIGHT_PAN	1_{10} 0x01	**No**	Y	**No**
ECD_EVEREST_LIMIT_SCAN_LEFT_PAN	2_{10} 0x02	**No**	Y	**No**
ECD_EVEREST_LIMIT_SCAN_RIGHT_PAN	3_{10} 0x03	**No**	Y	**No**
Enable Limits Constants				
ECD_EVEREST_DISABLE_LIMITS	0_{10} 0x00	**No**	Y	**No**
ECD_EVEREST_ENABLE_LIMITS	1_{10} 0x01	**No**	Y	**No**

Table 17. Spectra III protocol defines for rev 1.27, part 2 of 2

3 Spectra IV Protocol Data

Mnemonic	Command		C	D	P	Response
DUMMY	0x0001	1_{10}				
SET_PRESET	0x0003	3_{10}		Y		SEND_GEN_RESPONSE
MOTION_PRESET_SET	—	—		Y		
CLEAR_PRESET	0x0005	5_{10}		Y		SEND_GEN_RESPONSE
MOTION_PRESET_CLEAR	—	—		Y		
GO_TO_PRESET	0x0007	7_{10}		Y		SEND_GEN_RESPONSE
FLIP	0x0007	7_{10}		Y		
GO_TO_ZERO_PAN	0x0007	7_{10}		Y		
MOTION_PRESET_GO	—	—		Y		
SET_AUXILIARY	0x0009	9_{10}		Y		SEND_GEN_RESPONSE
PERIPH_AUX_OUT_SET	—	—		Y		
CLEAR_AUXILIARY	0x000B	11_{10}		Y		SEND_GEN_RESPONSE
PERIPH_AUX_OUT_CLEAR	—	—		Y		
DUMMY	0x000D	13_{10}				
REMOTE_RESET	0x000F	15_{10}		Y		SEND_GEN_RESPONSE
SYSTEM_REMOTE_RESET	—	—		Y		
SET_ZONE_START	0x0011	17_{10}		Y		SEND_GEN_RESPONSE
MOTION_ZONE_START_SET	—	—		Y		
SET_ZONE_END	0x0013	19_{10}		Y		SEND_GEN_RESPONSE
MOTION_ZONE_END_SET	—	—		Y		
WRITE_CHARACTER_TO_SCREEN	0x0015	21_{10}		Y		SEND_GEN_RESPONSE
SCREEN_WRITE_CHARACTER	—	—		Y		
CLEAR_SCREEN	0x0017	23_{10}		Y		SEND_GEN_RESPONSE
SCREEN_CLEAR	—	—		Y		
ALARM_ACKNOWLEDGE	0x0019	25_{10}		Y		SEND_GEN_RESPONSE
PERIPH_ALARM_ACKNOWLEDGE	—	—		Y		

Table 18. Spectra IV Protocol Commands, Part 1 of 4

[10] $Header: /EHamilton/General Information/Presets and Commands/SPECTRA4.INC 2 9/08/06 1:45p Ehamilton $

[11] $Header: /EHamilton/General Information/Presets and Commands/Cmnds4.inc 3 11/30/06 8:47a Ehamilton $

Mnemonic	Command		C	D	P	Response
ZONE_SCAN_ON	0x001B	27_{10}		Y		SEND_GEN_RESPONSE
MOTION_ZONE_SCAN_ON	—	—		Y		
ZONE_SCAN_OFF	0x001D	29_{10}		Y		SEND_GEN_RESPONSE
MOTION_ZONE_SCAN_OFF	—	—		Y		
SET_PATTERN_START	0x001F	31_{10}		Y		SEND_GEN_RESPONSE
MOTION_PATTERN_START_SET	—	—		Y		
SET_PATTERN_STOP	0x0021	33_{10}		Y		SEND_GEN_RESPONSE
MOTION_PATTERN_STOP_SET	—	—		Y		
RUN_PATTERN	0x0023	35_{10}		Y		SEND_GEN_RESPONSE
MOTION_PATTERN_RUN	—	—		Y		
SET_ZOOM_SPEED	0x0025	37_{10}		Y		SEND_GEN_RESPONSE
CAMERA_ZOOM_SPEED_SET	—	—		Y		
SET_FOCUS_SPEED	0x0027	39_{10}		**N**		SEND_GEN_RESPONSE
CAMERA_FOCUS_SPEED_SET	—	—		**N**		
RESET_CAMERA_DEFAULTS	0x0029	41_{10}		Y		SEND_GEN_RESPONSE
CAMERA_DEFAULTS_RESET	—	—		Y		
AUTO_FOCUS_MODE	0x002B	43_{10}		Y		SEND_GEN_RESPONSE
CAMERA_FOCUS_MODE_AUTO	—	—		Y		
AUTO_IRIS_MODE	0x002D	45_{10}		Y		SEND_GEN_RESPONSE
CAMERA_IRIS_MODE_AUTO	—	—		Y		
AGC_MODE	0x002F	47_{10}		**N**		SEND_GEN_RESPONSE
CAMERA_AGC_MODE	—	—		**N**		
BACKLIGHT_COMPENSATION	0x0031	49_{10}		Y		SEND_GEN_RESPONSE
CAMERA_BACKLIGHT_COMPENSATION	—	—		Y		
AUTO_WHITE_BALANCE	0x0033	51_{10}		Y		SEND_GEN_RESPONSE
CAMERA_WHITE_BALANCE_AUTO	—	—				
ENABLE_DEVICE_PHASE_DELAY_MODE	0x0035	53_{10}		**N**		SEND_GEN_RESPONSE
PERIPH_DEVICE_PHASE_DELAY_MODE_ENABLE	—	—		**N**		
SET_SHUTTER_SPEED	0x0037	55_{10}		Y		SEND_GEN_RESPONSE
CAMERA_SHUTTER_SPEED_SET	—	—		Y		

Table 19. Spectra IV Protocol Commands, Part 2 of 4

Mnemonic	Command		C	D	P	Response
ADJUST_LINE_LOCK_PHASE_DELAY	0x0039	57_{10}		**N**		SEND_GEN_RESPONSE
PERIPH_LINE_LOCK_PHASE_DELAY_ADJUST	—	—		**N**		
ADJUST_WHITE_BALANCE_RB	0x003B	59_{10}		Y		SEND_GEN_RESPONSE
CAMERA_WHITE_BALANCE_RB_ADJUST	—	—		Y		
ADJUST_WHITE_BALANCE_MG	0x003D	61_{10}		Y		SEND_GEN_RESPONSE
CAMERA_WHITE_BALANCE_MG_ADJUST	—	—		Y		
ADJUST_GAIN	0x003F	63_{10}		**N**		SEND_GEN_RESPONSE
CAMERA_GAIN_ADJUST	—	—		**N**		
ADJUST_AUTO_IRIS_LEVEL	0x0041	65_{10}		Y		SEND_GEN_RESPONSE
CAMERA_IRIS_AUTO_LEVEL_ADJUST	—	—		Y		
ADJUST_AUTO_IRIS_PEAK_VALUE	0x0043	67_{10}		Y		SEND_GEN_RESPONSE
CAMERA_IRIS_AUTO_PEAK_VALUE_ADJUST	—	—		Y		
QUERY	0x0045	69_{10}		Y		18 byte part number
PRESET_SCAN	0x0047	71_{10}		**N**		SEND_GEN_RESPONSE
MOTION_PRESET_SCAN	—	—		**N**		
SET_ZERO_POSITION	0x0049	73_{10}		**N**		SEND_GEN_RESPONSE
MOTION_POSITION_ZERO_SET	—	—		**N**		
SET_PAN_POSITION	0x004B	75_{10}		Y		SEND_GEN_RESPONSE
MOTION_POSITION_PAN_SET	—	—		Y		
SET_TILT_POSITION	0x004D	77_{10}		Y		SEND_GEN_RESPONSE
MOTION_POSITION_TILT_SET	—	—		Y		
SET_ZOOM_POSITION	0x004F	79_{10}		Y		SEND_GEN_RESPONSE
CAMERA_ZOOM_POSITION_SET	—	—		Y		
QUERY_PAN_POSITION	0x0051	81_{10}		Y		SEND_GEN_RESPONSE
QUERY_TILT_POSITION	0x0053	83_{10}		Y		SEND_GEN_RESPONSE
QUERY_ZOOM_POSITION	0x0055	85_{10}		Y		SEND_GEN_RESPONSE
DOWNLOAD	0x0057	87_{10}		Y		SEND_GEN_RESPONSE
SYSTEM_DOWNLOAD	—	—		Y		
QUERY_PAN_RESPONSE	0x0059	89_{10}		Y		SEND_DATA_RESPONSE

Table 20. Spectra IV Protocol Commands, Part 3 of 4

Mnemonic	Command		C	D	P	Response
QUERY_TILT_RESPONSE	0x005B	91_{10}		Y		SEND_DATA_RESPONSE
QUERY_ZOOM_RESPONSE	0x005D	93_{10}		Y		SEND_DATA_RESPONSE
SET_MAGNIFICATION	0x005F	95_{10}		Y		SEND_GEN_RESPONSE
CAMERA_ZOOM_MAG_SET	—	—		Y		
QUERY_MAGNIFICATION	0x0061	97_{10}		Y		SEND_GEN_RESPONSE
QUERY_MAGNIFICATION_RESPONSE	0x0063	99_{10}				SEND_DATA_RESPONSE
ACTIVATE_ECHO_MODE	0x0065	101_{10}		Y		SEND_GEN_RESPONSE
SYSTEM_ECHO_MODE_ACTIVATE	—	—		Y		
SET_REMOTE_BAUD_RATE	0x0067	103_{10}		Y		SEND_GEN_RESPONSE
SYSTEM_BAUD_RATE_REMOTE_SET	—	—		Y		
START_DOWNLOAD	0x0069	105_{10}		Y		SEND_GEN_RESPONSE
SYSTEM_DOWNLOAD_START	—	—		Y		
QUERY_DEVICE_TYPE	0x006B	107_{10}		Y		SEND_GEN_RESPONSE
QUERY_DEVICE_TYPE_RESPONSE	0x006D	109_{10}				SEND_DATA_RESPONSE
QUERY_DIAGNOSTIC_INFO	0x006F	111_{10}		Y		SEND_GEN_RESPONSE
QUERY_DIAGNOSTIC_INFO_RESPONSE	0x0071	113_{10}				SEND_DATA_RESPONSE
VERSION_INFO_MACRO_OPCODE	0x0073	115_{10}		Y		SEND_DATA_RESPONSE
SYSTEM_MACRO_OPCODE_VERSION_INFO	—	—		Y		
See Table 27, page 59						
EVEREST_MACRO_OPCODE	0x0075	117_{10}		Y		SEND_DATA_RESPONSE
SYSTEM_MACRO_OPCODE_EVEREST	—	—		Y		
See Table 24, page 57						
TIMESET_MACRO_OPCODE	0x0077	119_{10}		Y		SEND_GEN_RESPONSE or
See Table 28, page 60						SEND_DATA_RESPONSE
and Table 29, page 60						
RETURN_STATUS	0x00FD	253_{10}		N		
INVALID_OP_CODE	0x00FF	255_{10}		N		
SET_LINE_LOCK_PHASE_DELAY	0x0139	313_{10}		T		
SET_WHITE_BALANCE_RB	0x013B	315_{10}		T		
SET_WHITE_BALANCE_MG	0x013D	317_{10}		T		
SET_GAIN	0x013F	319_{10}		T		
SET_AUTO_IRIS_LEVEL	0x0141	321_{10}		T		
SET_AUTO_IRIS_PEAK_VALUE	0x0143	323_{10}		T		

Table 21. Spectra IV Protocol Commands, Part 4 of 4

Response Types		
Mnemonic	Value	Action
SEND_NO_RESPONSE	0_{10}	Nothing further is done.
SEND_GEN_RESPONSE	1_{10}	A General response is sent.
SEND_ACK_RESPONSE	2_{10}	An ACK response is sent.
SEND_NAK_RESPONSE	3_{10}	A NAK response is sent.
SEND_ECHO_RESPONSE	4_{10}	A General response, followed by echoing data.
SEND_BAUD_CHANGE	5_{10}	A General response, at 2400 baud, then change baud.
SEND_ERROR_RESPONSE	6_{10}	An error message is sent.
SEND_DATA_RESPONSE	7_{10}	A data message is sent.
SEND_RESET_RESPONSE	8_{10}	A General response, then a full system reboot.
SEND_START_DOWNLOAD	9_{10}	A General response, then start downloading.
SEND_DOWNLOAD	10_{10}	A General response, then change baud.

Table 22. Response Op-codes

Response Formats							
Mnemonic	BYTE 1	BYTE 2	BYTE 3	BYTE 4	BYTE 5	BYTE 6	BYTE 7
SEND_NO_RESPONSE	—	—	—	—	—	—	—
SEND_GEN_RESPONSE	SYNC	ADDR	ALARM_DATA	CKSM	—	—	—
SEND_ACK_RESPONSE	SYNC	ADDR	STD_RESPONSE	D_ACK	0	0	CKSM
SEND_NAK_RESPONSE	SYNC	ADDR	STD_RESPONSE	D_NAK	0	0	CKSM
SEND_ECHO_RESPONSE	SYNC	ADDR	ALARM_DATA	CKSM	—	—	—
SEND_BAUD_CHANGE	SYNC	ADDR	ALARM_DATA	CKSM	—	—	—
SEND_ERROR_RESPONSE	SYNC	ADDR	STD_RESPONSE	D_NAK	ERROR CODE		CKSM
SEND_DATA_RESPONSE	SYNC	ADDR	STD_RESPONSE	D_ACK	DATA		CKSM
SEND_RESET_RESPONSE	SYNC	ADDR	ALARM_DATA	CKSM	—	—	—
SEND_START_DOWNLOAD	SYNC	ADDR	ALARM_DATA	CKSM	—	—	—
SEND_DOWNLOAD	SYNC	ADDR	ALARM_DATA	CKSM	—	—	—

Table 23. Response Formats

Note

The new response formats in Table 23, page 56 are only generted when the build time define ONLY_GEN__RESPONSE is not defined. When ONLY_GEN__RESPONSE is defined then only SEND_GEN_RESPONSE and SEND_DATA_RESPONSE formatted responses are generated.

[12] $Header: /EHamilton/General Information/Presets and Commands/Resp.inc 3 11/30/06 8:47a Ehamilton $

Everest_ops					
Mnemonic	Value		C	D	P
AZIMUTH_ZERO_QUERY	0x00	0_{10}		Y	
AZIMUTH_ZERO_RESPONSE	0x01	1_{10}		—	
ZOOM_LIMIT_SET	0x02	2_{10}		Y	
ZOOM_LIMIT_QUERY	0x03	3_{10}		Y	
ZOOM_LIMIT_RESPONSE	0x04	4_{10}		—	
ALARMS_QUERY	0x05	5_{10}		Y	
ALARMS_RESPONSE	0x06	6_{10}		—	
PATTERN_DELETE	0x07	7_{10}		Y	
MANUAL_PAN_LIMIT_LEFT_SET	0x08	8_{10}		Y	
MANUAL_PAN_LIMIT_RIGHT_SET	0x09	9_{10}		Y	
SCAN_PAN_LIMIT_LEFT_SET	0x0A	10_{10}		Y	
SCAN_PAN_LIMIT_RIGHT_SET	0x0B	11_{10}		Y	
LIMIT_QUERY See Table 25, page 58	0x0C	12_{10}		Y	
LIMIT_RESPONSE	0x0D	13_{10}		—	
LIMIT_CONTROL	0x0E	14_{10}		Y	
DEFINED_PRESETS_QUERY	0x0F	15_{10}		Y	
DEFINED_PRESETS_RESPONSE	0x10	16_{10}		—	
DEFINED_PATTERNS_QUERY	0x11	17_{10}		Y	
DEFINED_PATTERNS_RESPONSE	0x12	18_{10}		—	

Table 24. Everest Op-Codes

[13] $Header: /EHamilton/General Information/Presets and Commands/Everest.inc 3 11/30/06 8:47a Ehamilton $

Query_type					
Mnemonic	Value		C	D	P
MANUAL_PAN_LIMIT_LEFT_QRY	0x00	0_{10}		Y	
MANUAL_PAN_LIMIT_RIGHT_QRY	0x01	1_{10}		Y	
SCAN_PAN_LIMIT_LEFT_QRY	0x02	2_{10}		Y	
SCAN_PAN_LIMIT_RIGHT_QRY	0x03	3_{10}		Y	

Table 25. Query Types

Mnemonic	Value
D_ACK	0x01
D_NAK	0x00
STD_RESPONSE	0x0001

Table 26. Misc Protocol Values

[14] $Header: /EHamilton/General Information/Presets and Commands/Query.inc 3 11/30/06 8:47a Ehamilton $

[15] $Header: /EHamilton/General Information/Presets and Commands/Version.inc 3 11/30/06 8:47a Ehamilton $

Version_ops					
Mnemonic	Value		C	D	P
VERSION_REQUEST	0x00	0_{10}		Y	
VERSION_RESPONSE	0x01	1_{10}		Y	
BUILD_REQUEST	0x02	2_{10}		Y	
BUILD_RESPONSE	0x03	3_{10}		Y	
BL_VERSION_REQUEST	0x04	4_{10}		Y	
BL_VERSION_RESPONSE	0x05	5_{10}		Y	
FONT_VERSION_REQUEST	0x06	6_{10}		Y	
FONT_VERSION_RESPONSE	0x07	7_{10}		Y	
STRING_VERSION_REQUEST	0x08	8_{10}		Y	
STRING_VERSION_RESPONSE	0x09	9_{10}		Y	
LAT_TABLE_REQUEST	0x0A	10_{10}		Y	
LAT_TABLE_RESPONSE	0x0B	11_{10}		Y	
LONG_TABLE_REQUEST	0x0C	12_{10}		Y	
LONG_TABLE_RESPONSE	0x0D	13_{10}		Y	

Table 27. Version Op-Codes

For original design information see Section 4, page 62.

Time_Ops					
Mnemonic	Value		C	D	P
SET_SECONDS	0x00	0_{10}		Y	
GET_SECONDS	0x01	1_{10}		Y	
SET_HR_MIN	0x02	2_{10}		Y	
GET_HR_MIN	0x03	3_{10}		Y	
SET_MON_DATE	0x04	4_{10}		Y	
GET_MON_DATE	0x05	5_{10}		Y	
SET_YEAR	0x06	6_{10}		Y	
GET_YEAR	0x07	7_{10}		Y	

Table 28. Time Op-Codes

Time Command Format			
CMND1	CMND2	DATA1	DATA2
SET_SECONDS	TIMESET_MACRO_OPCODE	seconds	
GET_SECONDS	TIMESET_MACRO_OPCODE	seconds	
SET_HR_MIN	TIMESET_MACRO_OPCODE	hour	minute
GET_HR_MIN	TIMESET_MACRO_OPCODE	hour	minute
SET_MON_DATE	TIMESET_MACRO_OPCODE	month	day
GET_MON_DATE	TIMESET_MACRO_OPCODE	month	day
SET_YEAR	TIMESET_MACRO_OPCODE	year	
GET_YEAR	TIMESET_MACRO_OPCODE	year	

Table 29. Time Command Formats

Note

1. When SET_SECONDS is received, all updated times are copied over into the active time array.
2. If the upper byte of the data field of SET_YEAR is equal to 0 (00), then 2000 is added to the year value.

[16] $Header: /EHamilton/General Information/Presets and Commands/Time.inc 4 6/25/07 9:04a Ehamilton $

[17] $Header: /EHamilton/General Information/Presets and Commands/Ps.inc 3 11/30/06 8:47a Ehamilton $

Mnemonic	Value	32 preset mode	Description
PRESET_FLIP	33_{10}	—	Call Preset # for flip
PRESET_ZERO	34_{10}	—	Call Move to zero position
PRESET_83	83_{10}	—	no-op
PRESET_84	84_{10}	—	no-op
PRESET_85	85_{10}	—	no-op
PRESET_86	86_{10}	—	no-op
PRESET_87	87_{10}	—	no-op
PRESET_IR_FILTER_IN	88_{10}	21_{10}	Call
PRESET_IR_FILTER_OUT	89_{10}	22_{10}	Call
PRESET_MANUAL_LEFT_LIMIT	90_{10}	23_{10}	Set Preset # for manual left limit
PRESET_MANUAL_RIGHT_LIMIT	91_{10}	24_{10}	Set Preset # for manual right limit
PRESET_SCAN_LEFT_LIMIT	92_{10}	25_{10}	Set Preset # for scan left limit
PRESET_SCAN_RIGHT_LIMIT	93_{10}	26_{10}	Set Preset # for scan right limit
PRESET_RESET	94_{10}	27_{10}	no-op Do reset (not implemented)
PRESET_MENU_MODE	95_{10}	28_{10}	Set Go into menu mode
PRESET_STOP_SCAN	96_{10}	29_{10}	Call
PRESET_RANDOM_SCAN	97_{10}	30_{10}	Call
PRESET_FRAME_SCAN	98_{10}	31_{10}	Call
PRESET_AUTO_SCAN	99_{10}	32_{10}	Call

Table 30. Special Presets processed in CDPcontrol.cpp

4 Time Setting Commands in D Protocol

Robert D. Sexton
Sr. Software Engineer
Pelco
March 2, 2006

4.1 Overview

The Pelco D Protocol does not include any remote time setting commands, simply because a real-time clock has never before been implemented. For Spectra III+[19], such a feature is included and requires an extension in the protocol to accommodate this.

A time setting function must be handled as a set of commands. This is because the structure of D packets includes only four bytes of data, including the command identifier. Effectively, only two bytes (16 bits) of data can be passed to or from the receiver per transmission.

One solution to this limitation is effectively to extend the command value to 16 bits, rather than the 8 bits of the standard commands. This method has been employed in the Version Information Macro Opcode, and the Everest Macro Opcode commands. These commands employ a primary opcode (`0x73` and `0x75` respectively) and a set of sub-opcodes to expand the utility of the command. The opcode is transmitted as data byte 2, the sub-opcode as byte 1, and any data as bytes 3 and 4. Where data is returned, the same opcode is used with a different sub-opcode. Where the command sends data to the receiver, the responses are `ACK` or `NACK` in the extended response format. It must also be noted that the opcodes are limited to odd numbers, the least significant bit must be binary 1 to identify it as an opcode as opposed to a motion command.

4.2 Proposed Time Setting Commands

According to the D Protocol document, V4.1 dated April 11, 2005, the last assigned commands are the two described above. The opcode value next in line is `0x77`. It is proposed that this value be reserved for time setting and reading functions. The sub-opcodes will be assigned such that even values (`0x00`, `0x02`, etc.) will be time setting commands, and odd value (`0x01`, `0x03`, etc.) will be time reporting sub-opcodes.

[18] `$Header: /EHamilton/General Information/Presets and Commands/TIMECMND.INC 1 6/25/07 9:04a Ehamilton $`

[19] Now named the Spectra IV.

4.3 Sub-opcode usage

0x00	Set seconds and synchronize time
0x01	Report seconds
0x02	Set hour and minutes
0x03	Report hour and minutes
0x04	Set month and date
0x05	Report month and date
0x06	Set year
0x07	Report year

Set commands return an ACK if successful, or NACK if not. Report commands will return a NACK if time is not set. In order to assure a successful time setting operation, time and date should be sent top-down, that is, year, month-date, hour-minute, seconds. Similarly, a time request will only be guaranteed to be correct if no intervening transmissions occur.

0x00 Set seconds and synchronize time	
byte 3	0x00
byte 4	Seconds to set (0-59)

On receipt of this command, the receiver clock will be set to the time and date previously transmitted. If an unrelated transmission occurs between this command and other time setting commands, no action will be taken. The response to this command is ACK if the seconds value is in range and time is set. It is NACK if out of range or time is not set.

0x01 Report seconds	
byte 3	0x00
byte 4	Current value of seconds.

When reporting any segment of the clock data, the current time will be read and held in a buffer until an unrelated transmission occurs. Thus no ambiguities or anomalies will be reported if the time requests occur in succession.

0x02 Set hour and minutes	
byte 3	Hour to set (0-23)
byte 4	Minutes to set (0-59)

Time will always be transmitted in 24-hour format. That is, midnight is 00:00, etc.

0x03 Report hour and minutes	
byte 3	Current hour.
byte 4	Current minute.

0x04 Set month and date	
byte 3	Month (1-12)
byte 4	Date (1-31)

The date will be out of range if not valid for the month specified. If the month specified is February, the range is limited to 1 — 28 unless the year has been determined to be a leap year, in which case 29 is an acceptable value. If the year has not been set, it is assumed that it is not a leap year.

0x05 Report month and date	
byte 3	Current month.
byte 4	Current date.

Month and date are reported in the same format as the corresponding set command.

0x06 Set year	
byte 3	MSB of year.
byte 4	LSB of year.

The year may be sent as a complete value (i.e. 2006) or as the last two digits (i.e. 06.) If only the last two digits are sent, the century value is assumed to be 2000 and is added to value sent to determine the year.

0x07 Report year	
byte 3	MSB of year.
byte 4	LSB of year.

The year reported is always the absolute value, that is, 2006 is always sent as 2006, not 06.

5 IRD/ERD information

Note from Craig Hannen about the ways that the IRD2024/ERD2200 processes commands and presets. The name **IRD** stands for **I**nternal **R**eciever **D**river and the name **ERD** stands for **E**nvironmental **R**eciever **D**river, i.e. usable outside.

```
From:       Hannen, Craig
Sent:       Friday, August 26, 2005 5:05 PM
To:         Hamilton, Eric
Subject:    IRD2024/ERD2200 Preset and Command Information
```

..., there are a small set of commands that are supported in the IRD/ERD. All commands are received via Coaxitron (Standard or Extended). Pan/tilt/lens control is done at the bit level. The Coaxitron Bit for each pan/tilt/lens control is transferred directly to the control line that corresponds to the bit. Since the IRD/ERD controls fixed speed pan and tilts, all speed information is ignored.

The IRD/ERD has no support for preset positioning, but there is support for the preset commands that invoke scanning. The preset commands that are supported in Extended Coaxitron are:

1. **Preset 96** — Manual Scan (turns off the next three scan modes)
2. **Preset 97** — Random Scan (scans for 0-60 seconds; pauses for 0-60 seconds; restarts in a Random direction; repeat)
3. **Preset 98** — Continuous Scan (in other words; scan forever)
4. **Preset 99** — 30 minutes of Continuous Scanning followed by Random Scanning

Unlike many of our receivers, PTZ control will not take the The IRD/ERD out of scan mode. Only Preset 96 (or the Standard Coaxitron manual scan command) will do this.

To support presets 97 and 99 in Standard Coaxitron, the auto scan command is used. When the The IRD/ERD receives the first Standard Coaxitron auto scan command, it goes into Random Scan mode (Preset 97 above) . It's important that any auto scan command is followed by an all stop command (I have found that not all Standard Coaxitron transmitters work this way, even though they should). The all stop command is used as a command separator. After the all stop command, the next auto scan command enables the Preset 99 scan described above. A third auto scan command puts the The IRD/ERD back in Random Scan mode. The Standard Coaxitron manual scan command will turn off either scan mode.

The The IRD/ERD supports 2 auxes (aux 1 is a 5V, 20mA open collector output and aux 2 is a relay). In Standard Coaxitron the auxes are momentary (auxes are switched on with the corresponding aux command and turned off via an all stop command). In Extended Coaxitron auxes can be momentary or latching. The Extended aux 1 and aux 2 on commands will momentarily

[20] `$Header: /EHamilton/General Information/Presets and Commands/IrdErd.inc 2 6/25/07 8:22a Ehamilton $`

switch auxes on for 100 milliseconds. The Extended aux 3 and aux 4 on/off commands will latch auxes 1 and 2 respectively.

The list of predefined constants in the source code is small. Here's the list:

```
PresetGo    equ 07h   ;extended preset go command.
SetAux      equ 09h   ;extended set auxiliary command.
ClrAux      equ 0bh   ;extended clear auxiliary command.
SetManual   equ 96    ;extended command - manual mode.
SetRandom   equ 97    ;extended command - random mode.
SetAllAuto  equ 98    ;extended command - infinite auto.
SetAuto     equ 99    ;extended command - 30 minute auto then to random.
```

The The IRD/ERD checks the validity of a command by counting the number of bits in the command. If the number of bits is not 15, 16 or 32, the command is discarded.

I think this sums up just about everything anyone would ever want to know about the IRD/ERD. If I left out anything important, let me know.

Craig Hannen
channen@pelco.com
1-800-289-9100 (2395)

5.1 ERD97P21-U

The ERD97P21-U uses P Protocol and a reduced set of commands. It does support the normal P Protocol set of motion commands of pan, tilt, iris, telephoto and focus. In addition it supports only the following commands:

Name	Value
Commands	
EC_SET_PRESET	3
EC_CLEAR_PRESET	5
EC_MOVE_PRESET	7
EC_SET_AUX	9
EC_CLEAR_AUX	11
EC_RESET	15
EC_ACK_ALARM	25
EC_ZOOM_SPEED	37
EC_FOCUS_SPEED	39
Special Presets	
PRESET_STOP_SCAN	96
PRESET_RANDOM_SCAN	97
PRESET_FRAME_SCAN	98
PRESET_AUTO_SCAN	99
If enabled only	
EC_STATUS	0xFD

Table 31. ERD97P21-U commands and presets

6 LRD Information

The **LRD** is a receiver driver designed for use with legacy pan/tilt units. There are three different basic models. These are:

1. **LRDA41**, Coaxitron only control.
2. **LRD41C11**, Coaxitron with P and D Protocol control for fixed speed Pan/Tilt units without presets.
3. **LRD41C12**, Coaxitron with P and D Protocol control for variable speed Pan/Tilt units without presets.
4. **LRD41C21**, Coaxitron with P and D Protocol control for fixed speed Pan/Tilt units with presets.
5. **LRD41C22**, Coaxitron with P and D Protocol control for variable speed Pan/Tilt units with presets.

This section is directly from the manual for the LRD41C21/22 C557M (10/99) page 20.
Auto and random scan can be operated in either of two ways, depending on the type of control unit you have. One method involves using the AUTO/MAN switch (or keys) if your control unit has these functions. The other method involves setting presets.

- AUTO/MAN Switch: The Random Scan and Auto Scan functions are controlled by the same momentary switch on the control panel labeled AUTO and MAN. The first activation of the switch to the AUTO position will put the pan/tilt into Random Scan. In Random Scan operation, the pan/tilt will travel between the preset limits with a random scan period of about 2 to 30 seconds, and a random dwell period of between 2 to 30 seconds.

  At the completion of a dwell period, another random scan period is started. The direction of this scan period is also randomly determined. When a pan limit is reached, scan direction is reversed automatically.

  A second activation of the AUTO switch will put the pan/tilt into continuous duty Auto Scan (limit switch to limit switch). After approximately $\frac{1}{2}$ hour of auto scan, the circuit will reset to random scan. Commanding AUTO while in RANDOM mode causes a shift to AUTO mode and starts the $\frac{1}{2}$-hour timer. Similarly, commanding AUTO while in AUTO mode causes a shift to the RANDOM mode and zeros the $\frac{1}{2}$-hour timer.

- Presets: Auto and random scan also can be started by programming presets. The presets will work only when your system is configured for Extended (32-bit) Coaxitron mode.

[21] `$Header: /EHamilton/General Information/Presets and Commands/Lrd.inc 2 6/07/06 10:32a Ehamilton $`

6.1 Advantages of Random Scan

Because scan direction, scan period and dwell period are unpredictable, unauthorized activities or intrusions are discouraged.

Because of the reduced duty cycle, gear train wear, cable fatigue, drive motor wear and temperature rise are reduced. These factors all contribute to higher system reliability and increased equipment life.

7 Preset Data

Presets on Pelco's series of programmable domes/receiver-drivers have evolved over time.

In older systems there were a maximum of 32 fully programmable presets. At this point two special presets were added in, 33 for "flip" and 34 for "goto factory zero". (This was done on some models of the Intercept series of domes.)

Somewhat after adding in special presets 33 and 34 a decision was made to implement 64 fully programmable presets and so the maximum preset number became 66[23]. (This was done on the Spectra I, Spectra II and Esprit.)

On the LRD presets 33 and 34 are "normal" and the maximum preset number allowed is 64. The Spectra I, Spectra II and Esprit all have 64 fully programmable presets, however because of the special use of presets 33 and 34, the highest numbered preset is 66.

Additionally some presets were given special uses:

1. 1 was made into an optional "park" preset on the Spectra, Esprit and ExSite units. Not on LRD based systems.
2. Presets 1 through 7 were made into optional "alarm" presets. (Spectra and ExSite series only.)
3. On the LRD series that support presets there are eight alarm inputs which may call presets $1 \longrightarrow 8$ when activated.

As time went on it was decided to use the 9X series of presets ($88 \longrightarrow 99$) as additional specials with the following presets having defined uses:

1. 88/— = Infrared filter in. Not downconverted.
2. 89/— = Infrared filter out. Not downconverted.
3. 90/23 = Set left manual limit.
4. 91/24 = Set right manual limit.
5. 92/25 = Set left scan limit.
6. 93/26 = Set right scan limit.
7. 94/— = No defined use (was reset dome, but that use was never implemented).
8. 95/28 = Access the built in menu system.
9. 96/29 = Stop a scan
10. 97/30 = Start a Random scan.

[22] `$Header: /EHamilton/General Information/Presets and Commands/PresetL.inc 7 11/30/06 8:47a Ehamilton $`

[23] $66 = 64 + 2$ with the 2 being those with predefined uses.

11. 98/31 = Start a Frame scan.
12. 99/32 = Start an Automatic scan.

The next event was that Pelco discovered some matrices had an upper preset limit of 32 and thus it was difficult to access the menu system. To get around this problem an option was added into newer systems that allowed the special presets in the 9X series to be down converted to values less than 33. (Presets 33 and 34 were not downconverted and remained inaccessible.) This resulted in a reduction of the fully programmable presets that were available. The downconverted values are shown as the second number in the above listing.

Some of Pelco's equipment does not have an ability to execute a flip. These are the Esprit and ExSite.

There were some concerns that the total number of presets had been reduced to an unacceptable low level and so it was made possible to use some of the downconverted special presets as "normal" presets when they were not being used for their "special" uses. Thus downconverted presets in the range of 23 $\longrightarrow$ 26 may be used as programmable presets when "limit stops" are turned off.

When the Spectra III was being designed it was decided to increase the total number of presets to 80 fully programmable presets with the high preset number being 82[24]. Eventually in software revision 1.28, the maximum preset number was increased to 150, giving a total of 131 presets[25].

The Spectra III also specified an additional use for one of the lower numbered presets (8).

On those camera types that support it, an ability was added to enable/disable the use of the infrared filter by using presets 88 and 89.

7.1 IR Cut Filter

How I can turn on/off the IR-filter[26]?

With cameras that have an IR cut-filter and we can capture the same scene in Black and White with no "noise" artifacts we can limit the source to the color burst and its phase/saturation/etc properties.

With a camera capable of C/BW such as, the 23x or 18x, Spectra III, the IR cut filter can be put into manual mode Via the menus...

```
<MAIN>
  |
  |--<DOME SETTINGS>
       |
       |--<CAMERA>
            |
            |--IR CUT FILTER --- Set to "OFF" for manual control
```

[24] $80 = 82 - 2$ with the 2 being those with predefined uses of flip and pan zero.
[25] $131 = 80 + 51$ ($51 = 150 - 100 + 1$ with the +1 being needed as both presets 100 and 150 are fully programmable)
[26] From an e-mail by Derek Springer, 24APR06.

Once the camera is set to "OFF" for IR cut filter control, Preset 89 can be used to switch the cut filter out and put the camera into black and white mode. Preset 88 puts the IR cut filter back in for normal color operation. If the "noise" artifacts are no longer present in the same scene [i.e. we don't now have black and white image artifacts] we would then be able to deduct that the artifacts are coming from the color properties....

7.2 Maximum Programmable Presets

The maximum number of programmable presets varies by the unit type and software revision level:

1. **22**: Spectra, Esprit and ExSite all models with AD Mode turned on and manual/scan limits set.
2. **28**: Spectra, Esprit and ExSite all models with AD Mode turned on.
3. **32**: Intercept, Spectra II Lite and Spectra III.
4. **64**: LRD, Spectra I and Spectra II, and Esprit.
5. **80**: Spectra III SE before software revision 1.28 and ExSite.
6. **131**: Spectra III SE with software revision 1.28.
7. **128**: Spectra IV.
8. **256**: Spectra IV SE.

Preset table notes

Lower case letters apply when high numbered presets are down converted for use on 32 preset systems.

- —: = Not implemented
- A: = Alarm preset
- CAL_0: = Rotate dome to factory zero azimuth (34)
- Flip: = Flip, rotate dome 180^o(33). Intentionally suppressed in the Esprit and ExSite lines of units.
- IR_{In}: = Infrared Filter In (88)
- IR_{Out}: = Infrared Filter Out (89)
- M_L: = Set Left Manual Limit (90 or 23 m_l)
- M_R: = Set Right Manual Limit (91 or 24 m_r)
- Menu: = Menu Access (95 or 28 $menu$)
- P: = Fully programmable preset (most of $1 \longrightarrow 66$)
- P_3: = Fully programmable preset on the Spectra III ($67 \longrightarrow 82$ and $100 \longrightarrow 150$ for software revision 1.28 and above.)
- pl: = Protocol limitation, may not be defined, protocol only has a single 8-bit value for presets
- PR: = Park preset. (Used when enabled with the built in menu system.)
- S_{Auto}: = Start Automatic Scanning (99 or 32 s_{auto})
- S_{Ex}: = Special use presets for use with the ExSite only. (Do not get downconverted.)
  1. 84: = Auxiliary 1
  2. 85: = Auxiliary 2
  3. 86: = Wiper
  4. 87: = Washer
  5. 100: = Position Calibration
- S_{Frame}:= Start Frame Scanning (98 or 31 s_{frame})
- S_L: = Set Left Scan Limit (92 or 25 s_l)
- S_R: = Set Right Scan Limit (93 or 26 s_r)
- S_{Rand}: = Start Random Scanning (97 or 30 s_{rand})
- S_{Stop}: = Stop a Scan (96 or 29 s_{stop})

	x0	x1	x2	x3	x4	x5	x6	x7	x8	x9
0x	—	P PR A	P A	P A	P A	P A	P A	P A	P PR	P
1x	P	P	P	P	P	P	P	P	P	P
2x	P	P	P	P m_l	P m_r	P s_l	P s_r	P	P $menu$	P s_{stop}
3x	P s_{rand}	P s_{frame}	P s_{auto}	Flip	CAL_0	P	P	P	P	P
4x	P	P	P	P	P	P	P	P	P	P
5x	P	P	P	P	P	P	P	P	P	P
6x	P	P	P	P	P	P	P	P_3	P_3	P_3
7x	P_3	P_3	P_3	P_3	P_3	P_3	P_3	P_3	P_3	P_3
8x	P_3	P_3	P_3	—	S_{Ex}	S_{Ex}	S_{Ex}	S_{Ex}	IR_{In}	IR_{Out}
9x	M_L	M_R	S_L	S_R	—	Menu	S_{Stop}	S_{Rand}	S_{Frame}	S_{Auto}

Table 32. Preset types as seen by the user, 0 ⟶ 99

	x0	x1	x2	x3	x4	x5	x6	x7	x8	x9
10x	S_{Ex}, P_3	P_3	P_3	P_3	P_3	P_3	P_3	P_3	P_3	P_3
11x	P_3	P_3	P_3	P_3	P_3	P_3	P_3	P_3	P_3	P_3
12x	P_3	P_3	P_3	P_3	P_3	P_3	P_3	P_3	P_3	P_3
13x	P_3	P_3	P_3	P_3	P_3	P_3	P_3	P_3	P_3	P_3
14x	P_3	P_3	P_3	P_3	P_3	P_3	P_3	P_3	P_3	P_3
15x	P_3	—	—	—	—	—	—	—	—	—
16x	—	—	—	—	—	—	—	—	—	—
17x	—	—	—	—	—	—	—	—	—	—
18x	—	—	—	—	—	—	—	—	—	—
19x	—	—	—	—	—	—	—	—	—	—

Table 33. Preset types as seen by the user, 100 ⟶ 199

	x0	x1	x2	x3	x4	x5	x6	x7	x8	x9
20x	—	—	—	—	—	—	—	—	—	—
21x	—	—	—	—	—	—	—	—	—	—
22x	—	—	—	—	—	—	—	—	—	—
23x	—	—	—	—	—	—	—	—	—	—
24x	—	—	—	—	—	—	—	—	—	—
25x	—	—	—	—	—	—	pl	pl	pl	pl
26x	pl	pl	pl	pl	pl	pl	pl	pl	pl	pl
27x	pl	pl	pl	pl	pl	pl	pl	pl	pl	pl
28x	pl	pl	pl	pl	pl	pl	pl	pl	pl	pl
29x	pl	pl	pl	pl	pl	pl	pl	pl	pl	pl

Table 34. Preset types as seen by the user, 200 ⟶ 255

7.3 Listing of all presets and their uses

- **0:** not used
- **1:** Normally programmable all units, Alarm #1 preset, Park preset on Spectra domes
- **2:** Normally programmable all units, Alarm #2 preset
- **3:** Normally programmable all units, Alarm #3 preset
- **4:** Normally programmable all units, Alarm #4 preset
- **5:** Normally programmable all units, Alarm #5 preset
- **6:** Normally programmable all units, Alarm #6 preset
- **7:** Normally programmable all units, Alarm #7 preset
- **8:** Normally programmable, all units, Park preset on Spectra III domes. Alarm #8 preset on LRD units.
- **9:** Normally programmable, all units
- **10:** through **22** Normally programmable, all units
- **23:** Normally programmable, all units, in 32 preset mode this is Set Left Manual Limit. See notes on preset 90 for additional information.
- **24:** Normally programmable, all units, in 32 preset mode this is Set Right Manual Limit. See notes on preset 91 for additional information.
- **25:** Normally programmable, all units, in 32 preset mode this is Set Left Scan Limit. See notes on preset 92 for additional information.
- **26:** Normally programmable, all units, in 32 preset mode this is Set Right Scan Limit. See notes on preset 93 for additional information.
- **27:** Normally programmable, all units, in 32 preset mode this is not used and is not programmable
- **28:** Normally programmable, all units, in 32 preset mode this calls the internal menu system. See notes on preset 95 for additional information.
- **29:** Normally programmable, all units, in 32 preset mode this Stops a scan. See notes on preset 96 for additional information.
- **30:** Normally programmable, all units, in 32 preset mode this starts a Random scan. See notes on preset 97 for additional information.
- **31:** Normally programmable, all units, in 32 preset mode this starts a Frame scan. See notes on preset 98 for additional information.
- **32:** Normally programmable, all units, in 32 preset mode this starts an Automatic scan. See notes on preset 99 for additional information.
- **33:** Flip, all units except for the LRD where it is a normally programmable preset.
- **34:** Go to pan zero, all units except for the LRD where it is a normally programmable preset.
- **35:** through **63** Normally programmable, all units
- **64:** Normally programmable, all units last LRD preset
- **65:** Normally programmable, all units except for the LRD
- **66:** Normally programmable, all units except for the LRD last Esprit, Spectra I and Spectra II preset
- **67:** through **82** Normally programmable, Spectra III and ExSite
- **83:** not used
- **84:** Used with the ExSite to operate Auxiliary 1.
- **85:** Used with the ExSite to operate Auxiliary 2.
- **86:** Used with the ExSite to operate the Wiper.
- **87:** Used with the ExSite to operate the Washer.
- **88:** Infrared Filter In, Spectra III and Esprit. Call only, set is ignored. Must have the "`IR CUT FILTER`" in the menu system set to `OFF` for this operation to work. When sent it is possible to see the filter changing position. Sending this preset a second time does not cause any obvious action on the part of the camera.
- **89:** Infrared Filter Out, Spectra III and Esprit. Call only, set is ignored. Must have the "`IR CUT FILTER`" in the menu system set to `OFF` for this operation to work. When sent it is possible to see the filter changing position. Sending this preset a second time does not cause any obvious action on the part of the camera.
- **90:** Set Left Manual Limit, all units. Set only, call is ignored. Must be set before the right limit is set. Moving to this limit and "setting" it a second time will clear both left and right manual limits.
- **91:** Set Right Manual Limit, all units. Set only, call is ignored.

- **92:** Set Left Scan Limit, all units. Set only, call is ignored. Must be set before the right limit is set. Moving to this limit and "setting" it a second time will clear both left and right scan limits.
- **93:** Set Right Scan Limit, all units. Set only, call is ignored. Must be set before the right limit is set.
- **94:** not used. Set and call are ignored.
- **95:** Enter Menu Mode, all units except for the LRD. Set only, call is ignored.
- **96:** Stop a Scan, all units. Call only, set is ignored. Sending a call preset to any other preset number is ignored if that preset is not defined.
- **97:** Start Random Scan, all units. Call only, set is ignored.
- **98:** Start Frame Scan, all units except for the LRD. Call only, set is ignored.
- **99:** Start Auto Scan, all units. Call only, set is ignored.
- **100:** through **128** Normally programmable, Spectra III software rev 1.28 and newer also on all models of the Spectra IV. Used on the ExSite for position calibration
- **129:** through **150** Normally programmable, Spectra III software rev 1.28 and newer, also on the Spectra IV.
- **151:** thhough **255** Normally programmable on the Spectra IV SE.
- Higher numbered presets are **not addressable** in D Protocol, or in any other Pelco protocol. Pelco protocols use an 8 bit byte for protocol address.

8 Glossary

Automatic Scan This refers to an automatic scanning left and right for about two seconds when limits are set. Tilt does not move. The unit will move continuously. If limits are not set it will pan continuously. If limits are set it will turn around and continue scanning forever. May be stopped by any motion command or the stop scan preset. The scan speed is programmable and may be set up using the built in menu system.

CX9000 This refers to four different but similar Coaxitron Receiver/Drivers, that differ in the input voltages they use and the output voltages that they supply to the Camera/Pan/Tilt unit. These are:

1. **CX9024RX-PP**, In = 120 VAC, 60 Hz. Out = 24 VAC.
2. **CX9024RX-PP220**, In = 230 VAC, 60 Hz. Out = 24 VAC.
3. **CX9115RX-PP**, In = 120 VAC, 60 Hz. Out = 120 VAC.
4. **CX9220RX-PP**, In = 230 VAC, 60 Hz. Out = 120 VAC.

Downconverted Applies to high numbered presets, in the range of 88 $\longrightarrow$ 99, that have been moved to values less than 33, 21 $\longrightarrow$ 32, to accomidate older model matrices and controllers.

ERD is an environmental version of the IRD. **ERD** stands for **E**nvironmental **R**eciever **D**river, which is usable outside. Uses 15 and 32 bit Coaxitron.

Esprit A pan/tilt system that has three different basic models. Two models utilize Hitachi integrated camera packages, on one of these models the camera is in a pressurized pod. The other unit in the series does not have an integrated camera, but supports a "box camera" system system wherein the user may place any camera in the housing that will fit. This system utilizes A/D converters to read out the settings of the lens and then uses that data for presets.

ExSite This is an explosion proof system based on the Spectra III code set, but utilizing a newer compatible CPU. Comes in two basic configurations, one is fixed and the other has normal pan/tilt motion. Uses Hitachi integrated camera systems.

Factory Zero This is the pan angle where the calibration tab is on the unit (CAL_0).

Flip Indicates that the unit may be requested to move 180^o rapidly. This is so that when someone passes directly underneath the unit, it may be rapidly

[27] `$Header: /EHamilton/General Information/Presets and Commands/GLOSSARY.INC 4 9/07/05 10:56a Ehamilton $`

turned around to follow the person (or thing). Is suppressed on the Esprit and ExSite products. There is a special very high rate of motion used for flips.

Frame Scan This refers to an automatic scanning left and right for about two seconds within the scan limits. Tilt not moved. The unit then stops for about two seconds and repeats the scan. When the unit reaches a limit, it will turn around and continue scanning until the two second period is up. If no limits are set the unit scans to the left for two seconds, stops for two seconds and repeats forever. May be stopped by any motion command or the stop scan preset. The scan speed is programmable and may be set up using the built in menu system.

Fully Programmable Indicates that the preset may be set, cleared and called. I.e. P and D Protocol opcodes `0x03`, `0x05` and `0x07` all work correctly.

Infrared Filter Some camera models allow the internal infrared filter to be removed for improved low light level performance. When this happens the camera stops functioning as a color camera and changes into black/white (monochrome) mode.

Intercept These are two different dome types, one is an 8 inch dome and the other is a 14 inch dome. With this series of domes, each dome could only work with one communications protocol at at time. To change protocols required that a new set of software be loaded. All of these systems utilized "box cameras" from various manufactures. Some units had presets, some were variable speed drives, some no preset, some fixed speed drives, some had a character generator, some not, etc.

IRD The name **IRD** stands for **I**nternal **R**eciever **D**river. The IRD is identical to the ERD except for its type of housing which is intended for indoor use. Uses 15 and 32 bit Coaxitron.

LRD This is a stand alone unit that is used to drive various older pan/tilt units. Older pan/tilt units usually used different wires for each function. I.e. one wire provided power for the right motor, another for the left motor, etc. The older units did not have a built in receiver unit as the more modern units do.

LRD41A11 This is used with fixed speed Pan/Tilt units such as: PT780-P and PT780-SL. It operates with 15-bit Coaxitron only.

LRDC21 This is used with fixed speed Pan/Tilt units such as: PT780-P, PT780-P/PP, PT780-SL and PT780-SL/PP. It operates with Coaxitron and P/D Protocol over RS-422 circuits.

LRDC22 This is used with variable speed Pan/Tilt units such as: PT780-VS, PT780-VS/PP, PT780-VSSL and PT780-VSSL/PP. It operates with Coaxitron and P/D Protocol over RS-422 circuits.

Manual Limit Manual Limits are used to prevent the operator from moving the unit too far to the right and left. To set a manual limit, first set the left limit, then move to the right limit and set that. The left and right limits must be at least 10^o apart and less than 360^o. When each limit is set there is no feedback from the camera. The left limit has no meaning until a right limit is set. To clear both limits, run the unit into the left limit and send a second "set left limit" command and both limits will be cleared.

MRD This is a small stand alone translator that converts D Protocol and P Protocol to LG Protocol for use with fixed, i.e. non-moving, cameras.

PT780 This is a series of Legacy Pan/Tilt units and includes all of the following:

1. **PT780-P**, Fixed speed, non-slip ring Pan/Tilt unit.
2. **PT780-P/PP**, Fixed speed, non-slip ring Pan/Tilt unit with presets.
3. **PT780-SL**, Fixed speed, slip ring Pan/Tilt unit.
4. **PT780-SL/PP**, Fixed speed, slip ring Pan/Tilt unit with presets..
5. **PT780-VS**, Variable speed, non-slip ring Pan/Tilt unit.
6. **PT780-VS/PP**, Variable speed, non-slip ring Pan/Tilt unit with presets..
7. **PT780-VSSL**, Variable speed, slip ring Pan/Tilt unit.
8. **PT780-VSSL/PP**, Variable speed, slip ring Pan/Tilt unit with presets..

Random Scan This refers to using random numbers for the time to cause scanning left and right only. Tilt does not change. The unit stops for a random amount of time and repeats the scan. If limits are set when the unit reaches a limit, it will turn around and continue scanning until the random time period is up. If no limits are set the unit scans to the left for two seconds, stops, goes a random direction and repeats forever. May be stopped by any motion command or the stop scan preset. The scan speed is programmable and may be set up using the built in menu system.

Scan Limit Scan Limits are used to prevent the unit from moving past given left and right positions in any of the three scan modes. (Automatic, Random and Frame)

Spectra I A 7 inch diameter dome that incorporates many integrated features. Is the follow on to the Intercept series of domes. The original model used a Hitachi 12X integrated camera system.

Spectra II A 7 inch diameter dome that has many improvements over the older Spectra I. All models had fully integrated camera systems and there were several different models with differing capabilities. The various model utilized cameras from Hitachi, Sony and LG. This was the first of the Spectras that supported TXBs for translating competitor's control protocols into Pelco protocols. The first software release for this series was 2.06.

Spectra III A 7 inch diameter dome that was a complete redesign of the older members of the Spectra series. It utilizes a more powerful CPU chip (MCORE *vs.* 8051 series), is downloadable, has foreign language support and utilizes only Hitachi integrated cameras. There are several different models in this series with differing capabilities on each model. Lower models are a sub-set of the full system. The full system is called a Spectra III SE.

Index