FAQ TXB-S422

What should be

Frequently Asked Questions

Sensormatic RS422 Translator

22 March 2002

TABLE OF CONTENTS

Section **Page**

[1] `$Header: d:/sears/RCS/txbs4faq.tex,v 1.44 2002-02-08 16:06:06-08 Hamilton Exp Hamilton $`

LIST OF TABLES

Table **Page**

Note

A change bar will be used where any changes have been made since the last version of this FAQ was printed.

As is common, those pages with changed page numbers and references to those numbers, (this includes the Table of Contents and the Index) do not qualify for a change bar. Neither will there be a change bar on this page.

A summary of the changes will also be listed here.

Date	Notes
None	

This page intentionally left blank

1 What should be Frequently Asked Questions (FAQ)

General Note

Through out this document there will be references to a Spectra type of dome. This product is also designed to work with a Esprit type of pan and tilt unit. Many times I will forget to mention this fact, just pretend that I included Esprit each time I mention Spectra. Unless of course I only mean one or the other items (which I hope that I will clearly identify.)

1.1 General questions

1.1.1 Q: What product is this FAQ about?

A: The product that will be made on this project is a translator to allow a Spectra to connect into a Sensormatic RS422 system. The translator will be called a TXB-S422.

As a side note it should be obvious that if a Sensormatic installation is using a Manchester type code that is compatible with that generated by an American Dynamics AD2091 code translator. Then the existing TXB-AD may be used in a Spectra/Esprit.

I believe that this indicates that if a site is using Manchester to control its domes, then we can immediately install Spectras with TXB-ADs installed to replace/supplement them with.

1.1.2 Q: What is the TXB-S422's general function?

A: Its general function is to allow a Spectra to emulate a Sensormatic SpeedDome™ when controlled by a Sensormatic matrix.

1.1.3 Q: When powered up with a TXB-S422 installed in my Spectra, the second line of the initial screen changed to "`TXB-S422 Rev N.NN AA`". What does this mean?

A: This indicates that the TXB-S422 has been successfully installed in the Spectra. The various fields have the following meaning:

`TXB-S422`	This is the model of translator board installed.
`Rev`	Indicates that following it is the revision level of the software. (There wasn't enough room to indicate the address field. Sorry.)
`N.NN`	This is the revision level of the software on the TXB-S422. I.e. "`1.00`" is used for the first released version of the software, etc. There are three special values that may be used here instead of a revision level:

[2] `$Header: d:/sears/RCS/txbs4faq.inc,v 1.54 2002-02-28 12:57:42-08 Hamilton Exp Hamilton $`

`BETA`	Indicates Beta test software may be spelled any of several ways to indicate different versions of early "Beta" test software. Typical "alternate" ways to spell "beta" are "`BETA`", "`beta`", "`BETa`", "`BEta`", "`beTA`", "`Beta`", and so on. There is no specific order to the different versions.
`TEST`	Indicates "in house" test software
`DEBUG`	Indicates that the TXB-S422 is operating in special debug mode. In this mode the TXB-S422 sends out a debug stream of ASCII characters that indicate what the received data consisted of and what is output to the Spectra. This mode is enabled when bit 6 of the switch is turned on.

`AA` This is the value loaded into the address switch. It is a hexadecimal value.

Note: The power up display indicates the position of all switch sub-parts, **except** for SW1-7 and SW1-8, the communications line terminations control. This is because it is not possible to feed the status of the switch directly into the CPU without causing reliability problems.

All this information will be cleared when the TXB-S422 receives its first valid command from the controller.

1.2 Project Status and Project Problems

1.2.1 Q: What is the current project status?

A: Current design work involves understanding how an RC58 (also known as ViewManager™ of various types) type of keyboard/matrix interfaces to various SpeedDomes. Pelco has a document on the likely protocol used to control the domes which is titled "User's Guide, RS-422/RS-485, Communication Protocols, For SpeedDome™ Ultra IV and AD DeltaDome™ Camera Domes, and later versions". This has a Sensormatic part number of 8000-2694-01 on it and is rev O.

I have since discovered that the AD2083/02 uses a superset of the commands in this document and have asked Dennis Dodrill to attempt to get a newer version.

There is a additional document on using the Sensormatic set of communications techniques. The document has a Sensormatic part number of 8000-2573-19 and is tilted "Communication Protocols and Cable Networks". We have a copy of this four page document.

An additional Sensormatic manual that might be nice to have would be "Theory for SpeedDome LT Camera Domes", document number 8000-1692-01. This manual was referred to in the SpeedDome LT Installation and Service manual on page V.

1.2.2 Q: What are your current problems?

A: Current problems still revolve around not knowing exactly what types of domes and head ends are actually to be used with this translator. As far as can be determined, there are several different versions of the head ends and several different versions of domes that must be emulated by a Spectra equipped with a translator. The current design plans consist of one translator which will be able to process all types of S422 commands.

The translator will understand most of the S422 commands so that when a Spectra is installed there will be minimal operator retraining required for the Spectra's use. An effort will be made to allow some of the Spectra's improved features to be available to the system operator. It is currently unclear if the additional features will be "Pelcoish" additions to the basic capabilities of a SpeedDome™ or if the additions will be similar to those incorporated into a DeltaDome™.

I have had an opportunity to visit both the Fresno and Clovis Sears stores and have discovered that they have two different Sensormatic controllers. Fresno Sears has an RC58 type of controller and Clovis Sears has an RC216H type of controller. **What other stores have is unknown.**

1.2.3 Q: What types of matrix/controller have been used for testing?

A: There are at lease three types of matrix/controller used for testing. They are:

1. An RC58 at Sears Fresno.
2. An RC216H (VM96) at Sears Clovis. A second RC216H is available at Pelco which is incomplete but may be useful in the future.
3. An American Dynamics AD2083/02 Code translator, connected to an AD2050 which is controlled by an AD2078 keyboard.

4. Then some short command sequences were generated on a PC using a `GWBASIC` program.

See Table 1, page 38 for a listing of the various types of known Sensormatic controllers.

1.2.4 Q: Do you feel that this assortment of controllers is sufficient?

A: No. There is a significant amount of information that indicates the Sensormatic product is at least as broad and varied as Pelco's is. This means that what we have found is probably just the "tip of the ice berg".

1.2.5 Q: What will be the size and shape of the TXB-S422?

A: The finished device **must** fit inside a Spectra and/or an Esprit. The final physical size **must** be the same approximate size as current production models of the TXB series of translators. This includes all of the following units:

- TXB-AB
- TXB-AD
- TXB-B
- TXB-V

I have attempted to find a formal document that defines this size and was unable to do so.

1.2.6 Q: How much power will it consume?

A: The unit **must not** use more power that the above mentioned TXB type translators.
I have attempted to find a formal document that defines this power and was unable to do so.

1.2.7 Q: What power supply design do you plan to use?

A: The power supply should be copied from the proven design used in the TXB-AB, TXB-AD and TXB-B products.

1.2.8 Q: What type of crystal oscillator do you plan to use?

A: The oscillator circuitry should be copied from the proven design used in the TXB-AB, TXB-AD and TXB-B products.

1.2.9 Q: Resistor packs cost more than a similar number of surface mount resistors. Which type do you plan to use?

A: The "resistor pack" indicated for use with the switch should actually be several individual 100 K- surface mount resistors. I used the phrase "resistor pack", as that was easier to draw in than 6 little resistors would have been. As more is known about the real requirements of this project, it may be possible/necessary to reduce the number of switch positions.

1.2.10 Q: How much memory will be needed?

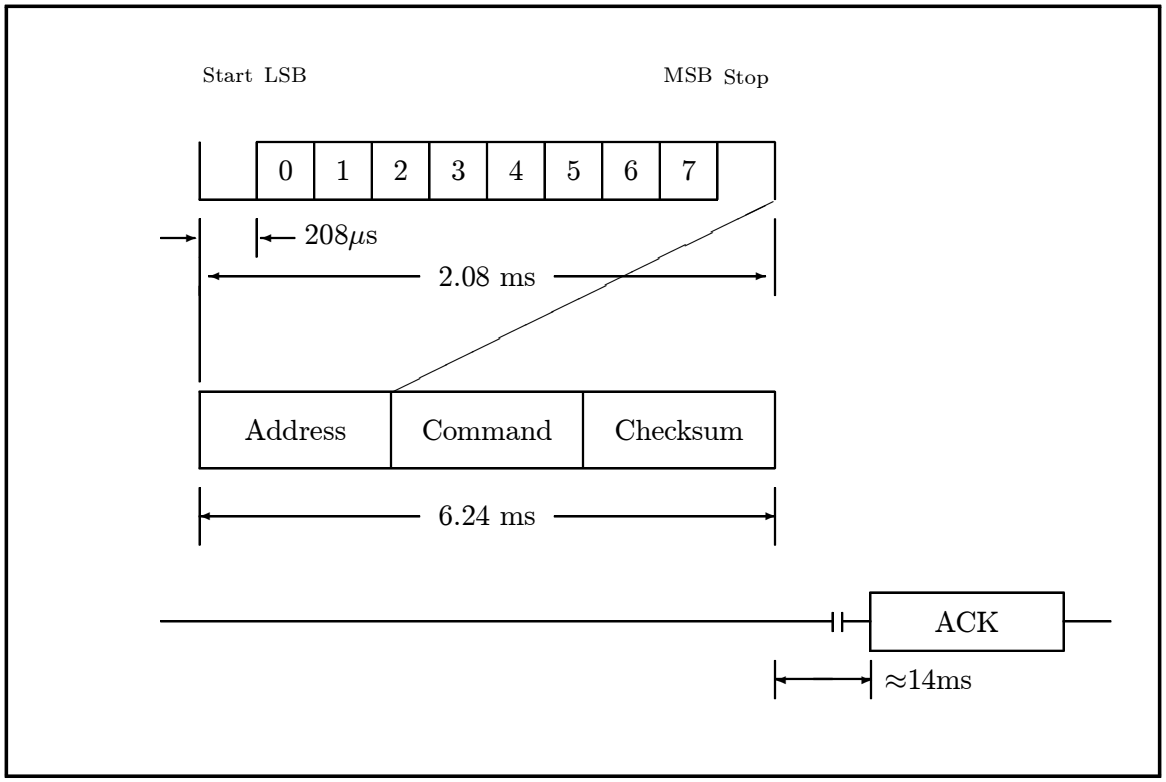
A: The PIC CPU chip contains all the required RAM/ROM/EEPROM memory needed for this project.

1.2.11 Q: What will the input logic be like?

A: Referring to the block diagram, the "KBD IO" block should be whatever is required to receive signals as shown in the attached oscilloscope pictures. Probably a MAX485/MAX491E type of interface chip will work OK. Although a more modern MAXIM 3157 or MAXIM 3089E would probably be more appropriate.

There is a possibility that terminations will be needed here. If so the type and size are the responsibility of hardware support to identify. As this translator will be installed inside a Spectra, protection from data line transients are probably unneeded. (Again this is an area for hardware support to verify.)

The command's format is:



$RCSfile: 3bytfmt.inc,v $

Figure 1. 3-byte format

A typical sample of dome poll messages are:

```
<20MAR01C.CAP>
00:06,  42: 0-60 0-94 0-0C          <96: Poll, Dome type query>
00:07,  43: 0-61 0-94 0-0B          <97: Poll, Dome type query>
00:08,  44: 0-62 0-94 0-0A          <98: Poll, Dome type query>
00:09,  45: 0-63 0-94 0-09          <99: Poll, Dome type query>

00:10,  47: 0-01 0-94 0-6B          < 1: Poll, Dome type query>
00:11,  48: 0-02 0-94 0-6A          < 2: Poll, Dome type query>
00:12,  49: 0-03 0-94 0-69          < 3: Poll, Dome type query>
00:13,  50: 0-04 0-94 0-68          < 4: Poll, Dome type query>
```

1.2.12 Q: Are you planning to be able to enable/disable the communications line terminations?

A: Yes. When SW1-7 is off, then the input line is unterminated. When SW1-8 is off, then the output line is unterminated. **Note:** The power up display indicates the position of all switch sub-parts, **except** for these two sub-switches. This is because it is not possible to feed the status of the switch directly into the CPU without causing reliability problems.

1.2.13 Q: What are you going to do about Brown outs and other power supply problems?

A: Brown out protection should be provided in hardware. It will be enabled in the PIC chip.

1.2.14 Q: Will you be using ICSP? (In Circuit Serial Programming)

A: I plan to, ICSP support should be provided. I have attempted to indicate which pins are associated with this feature.

1.2.15 Q: What type of PIC CPU chip will you be using?

A: 16F876 This chip was decided on because of its on board EEPROM memory and UART. I believe that the "flash" memory interface will be identical to the ICSP interface which I have specified to be provided.

1.2.16 Q: What types of power are available for the TXB-S422 to use?

A: The Spectra only supplies +12 VDC on its expansion socket, while the Esprit only provides +5 VDC on its expansion socket. (There is no indication of the maximum amount of power available from either unit.)

1.2.17 Q: What are you using to debug your design with?

A: Currently I am using a piece of American Dynamics equipment to simulate a controller for the Sensormatic domes. So far I have found out the following about it:

1. On power up, the AD2083/02 sends a REQUEST DOME TYPE (`SGETDOMETYPE`) query to all dome addresses, starting at 1 and going to 99 (0x63).

2. Following the power up query of the system, the AD2083/02 continues to send querys to all domes at a one dome address per second rate. I.e. it sends a query to dome 1 then a second later it sends a query to dome 2, etc. until it gets to dome 99 at which time it restarts with dome 1. (I.e. there is no dome address zero (0).)

   The RC58 at Fresno Sears recycles at dome 58. Probably with an RC20 or an RC32, recycling will occur at 20 and 32.

   The RC216 at Clovis Sears, recycles at dome 96 and polls every $\frac{1}{2}$ second.

1.2.18 Q: What types of domes are you using in checking the protocol?

A: I have at least two types of Sensormatic domes. I have two sub-types of SpeedDomes, two are single PC board units and two are dual PC board units. I can also get two similar UltraDomes loaned from Marketing. I used to have three MiniDomes on load from Sears Fresno, however I was never able to get them to work.

1.3 Features

1.3.1 Q: What types of Sensormatic dome communications will be supported?

A: As far as can be determined, Sensormatic domes support three different methods of signal modulation.

1. RS422/RS485 format two way communications. This consists of having the controller (head end) send a command to the dome and the dome interpreting the message and sending a response to the controller. For most commands the command is three bytes long and the response is one byte long.

2. Sensornet format communications. This is a SDLC encoded signal sent at 230.4 KHz using a proprietary logical link protocol.

3. Manchester format communications. Nothing known about this, other that the fact that it is mentioned several places in Sensormatic documentation.

Because of the presence of three very different types of communications, it is assumed that limiting this project to work with RS422/RS485 type of communications, will be acceptable. This type of communications will be referred to as S422 to differentiate it from other types of Sensormatic communications.

1.3.2 Q: What is this DIP switch I seen in the block diagram used for? And what is its size?

A: The DIP Switch consists of one eight position switch. It used to appear that some of the Sensormatic protocols (S422) appear to used a four bit group ID address with the rest of the address coming from the protocol. The extra portion of the address will be entered into the Spectra's address switch. The highest known Sensormatic dome address is 99.

Depending on the exact model of Sensormatic dome, some utilize address 64_{10} as a "special" address and others use 64_{10} as an "ordinary" address. To get around this ambiguity, a single switch position is used to "honor address 64", instead of the default of ignoring all messages sent to address 64_{10}.)

There is one switch position available that is used to enable debug outputs of the TXB-S422.

There are two switch positions available to enable terminating, or not, the input and output communications lines.

Two of the "address" switch bit positions, SW-1 and SW-2, are currently being used to control the translation of RC216 pan and tilt speeds into Spectra speeds.

Another of the "address" switch bits, SW-2, is used to select the amount of time that the TXB-S422 asserts its self on the RS422 return line. It has been found that SpeedDomes seize the line about 200 us before transmitting data and release it about 1 ms after the last bit is sent. UltraDomes/DeltaDomes have almost a 0 sec delay before and after seizing the line. When this bit is off, which is the default, then the TXB-S422 behaves like a SpeedDome and with it on, it acts like an UltraDome.

As of 25JAN02, SW-4 is the only unused switch position.

As more is known about the real requirements of this project, it may be necessary to change the number of required switch positions.

1.3.3 Q: What do you know about presets?

A: In the land of Sensormatic there are several ways that presets are handled. The exact method depends on the type of controller used.

1.3.3.1 AD2083/02 and RC216 type presets

The AD2083/02 and RC216 utilizes a rather strange method of doing presets. This consists of:

1. The controller (head end) asks the dome where it is currently pointing.
2. The dome responds with a message that gives details about where it is currently pointing. The message contains all of the following:
   A. Pan position.
   B. Tilt position.
   C. Zoom factor.
   D. And other information.
3. Then when the user wants to call a preset, such as 22, the controller then echoes back the saved dome position information to the dome. This data is in the form of a "GOTO" message.

Thus there is a vary basic difference between the way that Pelco and Sensormatic handle presets on a AD2083/02 type system. Sensormatic stores the pointing information in the controller (head end) and Pelco stores the pointing information in the dome.

Normally this difference of philosophy does not cause a problem, however in the two messages to the dome, the first one asking where the dome is pointing and the second one saying to point "that-a-way", there is no indication of the actual preset number associated with the preset. Thus the dome has no way of knowing which preset it is responding to, as would happen in a Pelco system.

What will happen with the TXB-S422 is that the TXB-S422 will save the most recently used preset ("SET") number. When the user sends a dome position request, the Spectra will respond, not with its "true position", but with fake message containing the saved preset number incremented by one. And the TXB-S422 will use the incremented preset number to generate a "SET PRESET" command to the Spectra. When the user sends a "GOTO PRESET" command to the Spectra it will now have a "fake" idea of the correct pointing information which will have the Spectra's preset number in it.

So far so good, eventually the Spectra will wrap around from preset 64 to preset 1 and will slowly replace older preset locations. **This means that after awhile, the Spectra will have a random assortment of old preset locations stored in it and will basicly have only the**

most recent 64 preset locations saved. Even if the most recent 64 presets consist of a lot of fine tuning of only two or three actual locations. An additional problem is that some of the special features of the Spectra may not be available, i.e. alarm presets for presets $1 \rightarrow 7$ on the Spectra, etc

1.3.3.2 RC58 type presets

The RC58 controller has a preset system that is similar to that used by Pelco. I.e. the dome remembers where each preset is and it is called by number not absolute position. However the RC58 is limited by having only seven presets available per dome. With the RC58 controller, presets are termed "targets".

1.3.4 Q: Can presets be "lost"?

A: Yes. In the above section (Section 1.3.3.1, page 15) it was pointed out that presets "wrap" when more than 64 presets have been loaded into the Spectra. There is another problem and that is if a GOTO TARGET # command is received by the TXBS422, then a GOTO PRESET command for one of the hard-coded presets between $1 \rightarrow 7$ will be sent to the Spectra, irreguardeless of whether on not a preset already is there. The **only** way to force a preset in the $1 \rightarrow 7$ range is to use one of the MARK TARGET # commands. However most of the newer types of equipment have no way of generating these older commands.

1.3.5 Q: Can the Spectra go the "wrong" preset?

A: Yes. See the above discussion. The key here is to remember that when using the newer method of specifying preset, i.e. sending a multibyte message back and forth. There is no way for the TXBS422 to know where the Spectra is pointing so it just "fakes" it. A GOTO TARGET # command may cause the Spectra to go to an unexpected place. And the value saved by a MARK TARGET # may be overwritten by the information for a new format QUICKVIEW command.

1.3.6 Q: I notice that the DIP switch and the crystal are identical to the ones on a Spectra. Is this accidental?

A: No.

1.4 Communications

1.4.1 Q: How will you send your data to the Spectra?

A: By utilizing the circuitry within the "Spectra IO" block. This circuitry should be similar to the simplified IO used with the TXB-AB, TXB-AD, TXB-B and TXB-V series of translators. This is possible because of the short protected distance between the TXB-S422 and the Spectra.

1.4.2 Q: What is the signal that the Sensormatic system sends that is translated by a TXB-S422?

A: The signal output by the Sensormatic equipment is called an RS422 signal. This signal is sent at baud rate of 4.8 KHz. For additional information about the Sensormatic protocol, see me for a copy of "Sensormatic Protocols". (Which is about 100 pages long!)

1.5 General Problems

1.5.1 Q: How do I turn on/off the windshield wiper on an Esprit?

A: On an Esprit the windshield wiper is controlled with Auxilary #4. It may be turned on by sending an AUX 1 ON and off by sending an AUX 1 OFF. If you are using a system that includes a TouchTracker (such as a VM96 type system), then using key "B" on the TouchTracker will alternately turn the windshield wiper on and off.

1.5.2 Q: How do I use the alarm inputs of a Spectra?

A: There is no good way to use the alarm inputs. The reason for this is that the protocol that Sensormatic uses does not have an alarm acknowledge command. Thus there is no way to tell the Spectra to stop reacting to an alarm. The Spectra will still go to one of presets 1 through 7 when an alarm is connected up to the seven alarm inputs, and it will still go back to active alarm inputs after being moved elsewhere. But without the alarm getting acknowledged this returning to a preset might get tiresome.

1.5.3 Q: When a "recalibrate" command is sent to the Spectra it does not display the TXB-S422's configuration information. What is going on here, you can display it on power up, why not now?

A: The TXB-S422 has no way of knowing that the Spectra is going through a recalibrate sequence, thus it can not know when to send its configuration information to the Spectra.

1.5.4 Q: Sometimes when using the joy stick to move around in a Spectra menu, it gets "stuck" and continues to move when I let go of the handle. What gives?

A: The TXB-S422 does extensive checks on every command that comes into it. This includes checking parity on each byte, checking the checksum on the whole message and verifying that the logical structure of the command is correct. Sometimes line errors cause commands to be "lost".

All motion commands, that result in movement of the Spectra, time out inside the actual Spectra, in about 15 seconds, to get around a possible "dropped command" problem. However there is no time out associated with cursor movement commands on a Spectra's menu. The easiest way to get around this is to use the joy stick again and this time the cursor usually behaves.

1.5.5 Q: Sometimes when accessing control menus the Iris control has to be hit once and sometimes twice, Why?

A: See the previous question and answer.

1.5.6 Q: I can't seem to get motion going when in Spectra's menu. Why?

A: When in menu mode, the Spectra will only respond to a very limited set of commands. All of the commands relate to selecting, etc., menu items. No other commands are accepted and this

includes recalibrate and scan commands.

1.5.7 Q: I used the Spectra's menu system and enabled some alarms. Now when an alarm goes off the Spectra points to the preset assigned to the alarm and keeps pointing at it following any inactivity. Why doesn't it respond to my joy stick commands as I expect it to?

A: The Spectra internally handles up to seven alarm inputs. When an alarm occurs the Spectra will automatically move to the preset for that alarm. (I.e. preset 4 for alarm 4 and so on.) If the alarm is not acknowledged **and** still active, then the Spectra will return to the alarm's preset location after about one minute of inactivity. The only way to stop this behavior is to "Acknowledge the alarm". To acknowledge the alarm it is necessary to send an appropriate command. *Commands for each alarm are* **currently unknown**.

1.5.8 Q: Does the TXB-S422 ever intentionally loose commands?

A: Yes. It takes about 29 ms to send a command to the Spectra. During this time no more commands are accepted. Thus if a second command comes in very quickly, it will be ignored. In early testing, the TXB-S422 does not loose commands that are 30.2 ms apart and does loose those that are 29 ms or less apart. The exact point as to where commands are lost is unknown.

1.5.9 Q: I have connected my Esprit (or Spectra) and nothing is working correctly. What is wrong?

A: Check that the DIP switch in the Spectra is not set to operate in American Dynamics mode. In American Dynamics mode, SHOTs are limited to 32, which makes it so that the full range of SHOTs required to setup and control an Esprit/Spectra are not being processed by the Esprit/-Spectra[3]. With the Esprit the switch is `SW1-5` and with the Spectra it is switch `SW1-7`. In future revisions of the equipment, the switch positions may change. Be sure to use the manual that came with your unit.

1.5.10 Q: I installed a TXB-S422 into my Spectra and everything worked perfectly. I then moved it to an Esprit and it didn't work at all. What is wrong here?

A: The most likely problem here is that the Spectra can only receive D protocol commands at 2400 baud and the TXB-S422 can only send them at 2400 baud. While the Esprit can receive D protocol commands at several different baud rates. Be sure that the Esprit is set to 2400 baud (`SW1-1`, `SW1-2` and `SW1-3` all `OFF`.) In future revisions of the equipment, the switch positions may change. Be sure to use the manual that came with your unit.

[3]There is no feed back from the Esprit/Spectra that would allow the TXB-S422 to know that the switch is incorrectly set. If it was possible to detect this problem, then it wouldn't be a problem.

1.5.11 Q: How is proportional pan enabled?

A: With the menu system of the Spectra on those units that support it. There are at least five different models of Spectra that the TXB-S422 (along with 12, or more, models of Esprit) may be used in. Some of them do not support proportional pan.

1.5.12 Q: Sensormatic has a several features that I can't get to work, Why?

A: These features are not implemented. Sensormatic and Pelco designs are fundamentally different is several places. This results in Sensormatic having some capabilities that Pelco can not emulate, or can not emulate well. The reverse of this is also true, i.e. some of Pelco's capabilities are "lost" when running on a Sensormatic controller.

1.5.13 Q: When programming a pattern, the Spectra never goes into turbo pan mode. Why?

A: There a is programming characteristic of the Spectra which causes the Spectra to always throw away any turbo command when in pattern programming mode. To get around this problem, the TXB-S422 translates all turbo commands, when in pattern programming mode, into the highest speed non-turbo speed available.

1.5.14 Q: On my Sensormatic domes there are XX zones. I can only find commands for 8 zones with the Spectra. What gives?

A: The Spectra has 8 variable width zones instead of XX zones. To enable a Spectra zone, its left and right limits must be set.

To clear a zone you must set the start and end the zone definition at the same location. The best way to do this is to set the start and then the end without moving the Spectra.

If two, or more, zones "overlap" then the highest numbered zone has "priority", is active and has its label displayed.

1.5.15 Q: Why can't I edit labels?

A: This is caused by a fundamental difference between the way that Pelco and Sensormatic handle labels (both zone and shot). With Pelco all label editing is done in the controller and then down loaded into the Spectra. With the Sensormatic system all editing is done inside the controller and not sent to the dome. The controller keeps track of the position of the dome and displays the correct label.

1.5.16 Q: I have heard that the new Spectra III has new features. Does the TXB-S422 support them?

A: Most new features of the Spectra III are not supported. Usually this is because there is no accurate information available as to what they are and how the D protocol has been augmented to permit their support.

1.6 C'est la vie

1.6.1 Q: How do I get an Apple Peal pattern?

A: No Pelco product supports this Sensormatic feature.

1.6.2 Q: How do can my computer tell where the Spectra is pointing?

A: No Pelco product currently, Feb 2002, supports this Sensormatic feature. It may be available when the Spectra III goes into full production. Currently the method of accesing this information is unknown/undefined.

1.6.3 Q: Are there any Sensormatic features that are not being supported?

A: Yes. This is because of design differences between the product lines of the two companies. The two companies do not share design information and both companies make changes without consulting the other. Many of the design philosophies are different and incompatible.

Another part of the problem is that it is difficult to get a full set of the very detailed capabilities of the Sensormatic design. What is known has been discovers by careful reading of public documents and extensive testing using a communications line analyzed (Breakout).

1.7 Addressing

1.7.1 Q: Sensormatic reserves address 64 as a broadcast address on some systems. The TXB-S422 does not appear to support it. Why?

A: This is one of the areas where Pelco and Sensormatic have fundamental differences in philosophy. Pelco does not have a broadcast address and there was no practical way to implement one. So it isn't supported.

1.7.2 Q: How are the bits assigned on the address switch?

A: Exactly the same as they are on the Spectra. I.e. to select address block 5, set the switches exactly as they would be for a Spectra camera #5 (`ON OFF ON OFF OFF`).

1.7.3 Q: How are Sensormatic domes addressed?

A: It appears that each type/series of Sensormatic domes/matrices/controllers have its own set of addressing peculiarities.

- With some types of dome/controller combinations, there are some unavailable addresses.
- With each type of dome there are maximum addresses that they can address to.
- For some, might be all, dome/controller combinations, address 64 is special. In some Sensormatic/American Dynamics publications, address 64 is called out as "NOT USED", in others it is not mentioned.
- Then each matrix can only address up to some convenient maximum number of cameras, this varies from a low of 8 to a maximum of 1,024.
- And for last the peculiarities of the Spectra shouldn't be forgotten.

1.7.4 Q: How can I determine what address to use with each dome type?

A: To attempt to get around these problems I have prepared several tables that will attempt to show how to work with all these conflicting capabilities.

An addressing table is available in a separate document entitled "Addressing various Sensormatic domes".

1.7.5 Q: What is the maximum address for a Spectra dome when connected to a TXB-S422?

A: 255.

1.7.6 Q: What is the minimum address for a Spectra dome when connected to a TXB-S422?

A: 1.

1.7.7 Q: What are the minimum and maximum addresses for an Esprit?

A: For most purposes the Esprit and Spectra behave identically. Thus I rarely mention the differences in this document. Since they are so similar they also have identical minimum and maximum addresses.

1.7.8 Q: What is the difference between a DeltaDome and a SpeedDome?

A: None, kinda. Just the names were changed to protect themselves, Sensormatic brands their domes as "SpeedDomes" while American Dynamics brands their exact equivalent as "DeltaDomes". (*Or is it the other way around?*)

1.7.9 Q: What is the highest direct address used by Sensormatic?

A: It might be 99. However on some dome types there is a third address switch. I.e. on the older two PC card type of SpeedDomes there are only two address switches, on the newer one PC card type of SpeedDome (and the DeltaDome/SpeedDome Ultra) there are three address switches. Each switch goes from 0 to 9, thus giving either a maximum upper dome address of either 99 or 999 and a lower value of 0. 0 does not appear to be used by any Sensormatic equipment.

1.7.10 Q: How many domes may be daisy-chained together?

A: It depends on the exact protocol in use.

Protocol	Maximum
SensorNet	32
RS422	10
Manchester	3

(This table was taken from: "DeltaDome II, Programmable, Surveillance Camera, AD716LS and AD715LS Series". Publication 8000-2708-01, Rev A, dated 1999.)

1.7.11 American Dynamics AD2083/02 addressing

1.7.12 Q: What is the maximum number of SpeedDomes on an American Dynamics system, per group, and driven by an AD2083/02A?

A: 99, kinda. The AD2083/02A outputs an eight bit address in the protocol. However it appears that only seven bits are actually used. In the American Dynamics manual[4] 8000-0938-01 Rev A dated June, 1997. There are tables in the appendix that show dome addressing up to camera 1,024. However every dome address of 64 is marked as "NOT USED". Thus the addressing is $1 \rightarrow 63$, $65 \rightarrow 99$ (note that address 0 and 64 are skipped). These camera numbers then repeat until the table gets to camera number 1,024 with a dome address of 34.

[4]This manual is titled "Model 2083/02A, Code Translator, SpeedDome LT, Outdoor SpeedDome, SpeedDome Ultra, and Mini Dome, Installation and Operation Instructions"

This addressing table is in a separate document entitled "Addressing various Sensormatic domes".

Camera address numbers that are not usable with this combination of domes/controllers are: 64, 163, 262, 361, 460, 559, 658, 757, 856 and 955 (note the steps of 99 between each unavailable camera number).

This addressing peculiarity is probably true with other dome types when connected to the AD2083/02A.

From examining the captured protocol data, address 64 appears to be a broadcast address, similar to the unimplemented (but reserved) address 0 on the Intercept line of domes.

For example, the Spectra should be set to 77 when the Sensormatic system is communicating with camera 77 in group 1-99, and 47 when Sensormatic is addressing 938 in group 892-990.

1.7.13 Q: How about using a dome address of 64?

A: As indicated above address 64_{10} (`0x40`) is special. The AD2083/02 will respond to a power up message from a dome at address 64_{10} and will send "poll" messages to address 64_{10}, but it will do nothing else. I.e. it can not control the dome with motion commands.

Thus an address of 64 is normally ignored by the TXBS422. To get the TXBS422 to honor address 64 set bit 5 on SW1.

1.7.14 Q: What is the maximum number of MiniDomes on an American Dynamics system, per group, when driven by an AD2083/02?

A: 79, kinda. The AD2083/02 outputs an eight bit address in the protocol. However it appears that only seven bits are actually used. In the American Dynamics manual[5] 8000-0872-01 dated February, 1997. In table 1, in the text, shows dome addressing for cameras from 1 to 1,024. The addressing is $1 \rightarrow 79$ (note that only address 0 is indicated as being skipped). These camera numbers then repeat.

This addressing table is in a separate document entitled "Addressing various Sensormatic domes".

1.7.15 Q: What is the maximum number of CobraDomes on an American Dynamics system when driven by an AD2083/02?

A: It is unlikely that a CobraDome will be replaced with this translator, however the answer is 64. In the American Dynamics manual[6] OP1273 B dated March 1994. In table 1, in the text, shows dome addressing for cameras from 1 to 1,024. The addressing is $1 \rightarrow 64$ (note that only address 0 is indicated as being skipped). These camera numbers then repeat.

This addressing table is in a separate document entitled "Addressing various Sensormatic domes".

[5]This manual is titled "Model 2083/02, Code Translator, Installation and Operation Instructions"

[6]This manual is titled "Model 1273 CobraDome, Indoor Surveillance Unit, Installation Instructions"

1.7.16 Q: How do you set the address switches on a Spectra?

A: For some unknown reason the address bit switches on a Spectra are "backwards", i.e. bit one, which is the least significant address bit, is on the left of the switch and bit eight, the most significant address bit, is on the right.

This addressing table is in a separate document entitled "Addressing various Sensormatic domes".

1.7.17 Q: How do you set the address switches on a TXB-S422?

A: As insufficant information is available to determine the actual addressing of a Sensormatic dome these switch positions $1 \rightarrow 6$ in SW1 are unused. If we find that the TXBS422 needs to be addressed, in addition to the address provided by the Spectra then the ability is here.

1.8 Dome Motion

1.8.1 Q: I saw a Spectra at a show and was impressed with its smooth motion. With a TXB-S422 installed in mine, the motion is not as smooth. Why?

A: Depending on the exact type of controller, different Sensormatic/American Dynamics controllers have different numbers of variable speeds available. Spectra domes support about 50 different speeds. Some of Sensormatic/American Dynamics controllers generate from eight to about 30 different speeds.

This results in Pelco systems having very smooth pan and tilt capabilities. The TXB-S422 only translates what it is given and there is no practical way to generate the "in between" speed values.

There are two situations here. One involves pan speeds and the other involves tilt speeds. Although the methods of determining the actual speeds are similar, the actual calculations are different.

For variable speed keyboards, the TXB-S422 uses the speeds that come in and attempts to identify the type of system it is connected to. If it is an American Dynamics system, which only supports eight different speeds, then it will generate the following Spectra speeds.

If it is determined that the TXB-S422 is under the control of a **Sensormatic RC216** then the following logic is used. The low order two bits on the switch are examined to determine how to modify the input speed. The following is then done to the input speed before it has either 8 or 6 added to them (8 is for pan, 6 is for tilt) and the result is range checked to determine if it is a turbo speed or past turbo speed. If the result is past turbo speed then it is either set to turbo speed (`0x40` for pan) or to one less than turbo speed (`0x3F` for tilt):

SW1-1	SW1-0	Value	Input speed modifications
0	0	0	The input speed is taken as is. (**Default value.**)
0	1	1	The input speed is multiplied by $\frac{3}{4}$.
1	0	2	The input speed is multiplied by $\frac{1}{2}$.
1	1	3	The input speed is multiplied by $\frac{1}{4}$.

Either 8 or 6 is added to the scaled input speed in order to get over the first few repeating speeds in the Spectra.

If the controlling source can not be identified then it is assumed that the controller has the same characteristics as an American Dynamics controller has and the American Dynamics table is used.

For fixed speed keyboards, the TXB-S422 uses a hard coded default value that gives 27/12 ^{o}/sec (pan/tilt). If the user wishes to, the speed may be changed.

1.8.2 Q: On my variable speed system, when I hold the joy stick all the way to the right or left, the camera goes real fast. Why?

A: The TXB-S422 automatically issues a "turbo speed" command when the joy stick is at maximum left or right. Turbo speed is greater than 100 degrees per second.

1.8.3 Q: On my fixed speed system, will proportional pan work?

A: On those Spectra types that support it yes. Proportional pan is done inside the Spectra.

1.8.4 At what exact points are speed changes made with an AD2083/02 translator?

In the two following tables the column marked "Decision point" indicates that at this decision point value, all lower speeds are translated into the values shown in the "Generated speeds" column. Remember that the lower values are checked first.

Tilt speeds and decision points		
Observed speeds	Decision point	Generated speeds
0x03	0x04	3
0x05	0x07	11
0x09	0x0B	19
0x0D	0x10	27
0x14	0x16	35
0x18	0x1C	43
0x21	0x27	51
0x2D and other	0x27+	63

Pan speeds and decision points		
Observed speeds	Decision point	Generated speeds
0x04	0x05	7
0x06	0x08	15
0x0A	0x0C	23
0x0F	0x13	31
0x18	0x1C	39
0x21	0x27	47
0x2D	0x43	55
0x5A and other	0x43+	64

1.8.5 Q: Can you compare the various pan speeds that are available using the TXB-S422 and various controllers?

A: Grudgingly yes. There are at least three different sets of values that have to be used with the TXBS422. This has been required by the known different characteristics of the known Sensormatic controllers.

This section has tables that compare the speed commands sent to a Spectra/Esprit from several sources. Note that fixed speed commands do not have a speed field and use a value that is stored inside the translator. With the TXB-S422 these values may be changed with the RAIL RIGHT, FAST and RAIL LEFT set of keyboard commands.

1.8.5.1 Fixed speed values

Fixed speed values are:

Key	Name	Increase	Pan Speed	Tilt Speed
"plain"	—	X1	27	12
RAIL RIGHT	FASTEST	X2	54	19
FAST	FASTER	X3	81	30
RAIL LEFT	FAST	X4	108	44

1.8.5.2 American Dynamics Speed Table

1. **Input** This is the American Dynamics protocol speed field value. This is an 8 bit wide field which has 8 different values in it.
2. **Output** This is the speed field sent in D protocol to the Spectra.
3. **Speed** This is the of rotational, in either pan or tilt, in degrees per second.

American Dynamics Spectra speed values					
Pan			**Tilt**		
Input	Output	Speed	Input	Output	Speed
0x04	0x07	0.5	0x03	0x03	0.5
0x06	0x0F	1.0	0x05	0x0B	2.3
0x0A	0x17	2.0	0x09	0x13	5.2
0x0F	0x1F	4.2	0x0D	0x1B	8.2
0x18	0x27	8.7	0x14	0x23	11.6
0x21	0x2F	18.3	0x18	0x2B	15.7
0x2D	0x37	38.2	0x21	0x33	21.7
0x5A	0x40	100+	0x2D	0x3F	44.0

1.8.5.3 Sensormatic Speed Table

The RC216 controller, and possiblely others, generates many different pan/tilt speed commands, at least 30 of each have been detected so far. The problem here is to translate the non-linear Sensormatic commands to non-linear Pelco commands.

In preliminary testing the following variable pan speeds were observed: $1 \rightarrow 20$, 22, 24, 32, 48, 64, 80 and 99. A similar set of speeds was observed for the tilt axis.

These speeds represent dome speeds in different degrees per second. Thus there is a potential for having 99 different speeds while Pelco only has about 53 different speeds. (It must be remembered that some slow speeds are repeated inside the Spectra and others are blocked to avoid mechanically resonance induced noise.)

Taking advantage of the fact that the Spectra repeats the first nine pan speed values we can simplify the conversion process quite a lot by always dividing the input value by 2 (the input range

is now $0 \rightarrow 48$) and then adding eight to it giving a low Spectra speed of 0.5^oper second and a high speed of 41.9^oper second. As this last speed is a little slow (it's NOT turbo speed) a special check is made to see if the input speed is 99_{10} (`0x63`) which we translate into a turbo pan speed of 64_{10} (`0x40`). For tilt speeds we ensure that the output speed is always less than the turbo value of 64_{10} (`0x40`).

1. When a true variable speed pan command is detected, the input speed is multiplied by a constant and 8 is added to it. If the result is greater than 64 then the speed is forced to 64 and used. The reasoning behind this is that many of the input speed are too big for the Spectra to use (several of them are "turbo" speed values) and that all Spectra pan speeds of seven and less are the same.

2. When a true variable speed tilt command is detected, the input speed is multiplied by a constant and 6 is added to it. If the result is greater than 63 then the speed is forced to 63 and used. The reasoning behind this is that many of the input speed are too big for the Spectra to use and that all Spectra tilt speeds of six and less are the same.

1.8.5.4 Pelco Speed Table

- The following table is ordered by the hexadecimal value sent to the Spectra/Esprit with values actually observed **in bold type**.

- Although the Spectra/Esprit listing indicates a given speed for a given input, there are some speeds that are intentionally blocked to reduce potential problems with resonance caused mechanical noise. These blocked speed values are dependent on the actual zoom factor in use and vary in different revisions of the Spectra/Esprit software. This results in a table that is should be considered a "rough guide" to the actual speeds that a Spectra/Esprit will rotate/tilt at.

- Spectra and Esprit pan and tilt speeds come from the source code for Spectra software revision 3.29 and Esprit software revision 1.01. These values are given in ^{o}/sec, with the lens at maximum wide angle and ignoring any "skipped" speeds. This table is specific to each revision/model of the Spectra/Esprit and may change at any time.

- In the past Pelco has done considerable research into the ergonomics of using a joy stick to control camera motion. The results of this research resulted in the following tables of camera motion *vs.* value reported out from Pelco's joy sticks. It should be noted that the table is not linear and has different values in pan and tilt (and NTSC and PAL tilt (and Esprit *vs.* Spectra)). Since pan has a total travel (360^o) which is four times that of tilt (90^o), which makes for some of the differences. Many values were empericaly derived from studies with various test subjects using Pelco keyboard/joy stick combinations. With the Sensormatic and American Dynamics variable speed keyboards the transfer characteristic of joy stick motion is different from that generated by a Pelco joy stick. Depending on the exact model Sensormatic/American Dynamics use as few as three bits (which gives 8 different speeds) in

their variable speed commands to designate speeds to as many as 7 bits (which gives a range of 1 to 99°sec), while Pelco utilizes 6 bits (giving 64 different potential speeds).

- In the following table, most entries are not used by the TXB-S422. I have marked those that are used in **bold** type. The last column (marked "Fixed Speed ON 1X Cmnd") indicates the command used with fixed speed keyboards to select a given set of pan/tilt speeds. The default speed is given by ON 15 which marked in **bold**.

Table Index		Pan		Tilt		
		Spectra	Esprit	Spectra		Esprit
				NTSC	PAL	
		°/sec	°/sec	°/sec	°/sec	°/sec
0x00	0_{10}	0.5	0.2	0.5	0.5	0.5
0x01	1_{10}	0.5	0.2	0.5	0.5	0.5
0x02	2_{10}	0.5	0.3	0.5	0.5	0.5
0x03	3_{10}	0.5	0.3	0.5	0.5	0.5
0x04	4_{10}	0.5	0.4	0.5	0.5	0.5
0x05	5_{10}	0.5	0.5	0.5	0.5	0.5
0x06	6_{10}	0.5	0.7	0.5	0.5	0.5
0x07	7_{10}	0.5	0.9	0.9	0.9	0.7
0x08	8_{10}	0.5	1.2	1.3	1.3	0.9
0x09	9_{10}	0.5	1.5	1.6	1.6	1.1
0x0A	10_{10}	0.6	1.8	2.0	2.0	1.3
0x0B	11_{10}	0.7	2.1	2.3	2.3	1.4
0x0C	12_{10}	0.7	2.5	2.7	2.7	1.6
0x0D	13_{10}	0.8	2.9	3.0	3.0	1.8
0x0E	14_{10}	0.9	3.3	3.4	3.4	2.0
0x0F	15_{10}	1.0	3.7	3.7	3.7	2.2
0x10	16_{10}	1.0	4.1	4.1	4.1	2.3
0x11	17_{10}	1.1	4.5	4.5	4.5	2.5
0x12	18_{10}	1.3	4.8	4.8	4.8	2.7
0x13	19_{10}	1.4	5.2	5.2	5.2	2.9
0x14	20_{10}	1.5	5.6	5.6	5.6	3.1
0x15	21_{10}	1.7	5.9	5.9	5.9	3.3
0x16	22_{10}	1.8	6.3	6.3	6.3	3.5
0x17	23_{10}	2.0	6.7	6.7	6.7	3.6
0x18	24_{10}	2.2	7.1	7.1	7.1	3.8
0x19	25_{10}	2.4	7.5	7.5	7.5	4.0

Continued on the next page.

Continued from the previous page.

Table Index		Pan		Tilt		
		Spectra	Esprit	Spectra		Esprit
				NTSC	PAL	
		°/sec	°/sec	°/sec	°/sec	°/sec
0x1A	26_{10}	2.6	7.8	7.8	7.8	4.2
0x1B	27_{10}	2.9	8.2	8.2	8.2	4.4
0x1C	28_{10}	3.2	8.6	8.6	8.6	4.6
0x1D	29_{10}	3.5	9.0	9.0	9.0	4.9
0x1E	30_{10}	3.8	9.4	9.4	9.4	5.1
0x1F	31_{10}	4.2	9.9	9.9	9.9	5.3
0x20	32_{10}	4.6	10.3	10.3	10.3	5.5
0x21	33_{10}	5.0	10.7	10.7	10.6	5.7
0x22	34_{10}	5.5	11.1	11.1	10.7	6.0
0x23	35_{10}	6.0	11.6	11.6	11.8	6.2
0x24	36_{10}	6.6	12.1	12.1	12.1	6.4
0x25	37_{10}	7.3	12.5	12.5	12.5	6.7
0x26	38_{10}	8.0	13.0	12.8	13.0	6.9
0x27	39_{10}	8.7	13.5	13.1	13.5	7.2
0x28	40_{10}	9.6	14.1	13.9	14.1	7.5
0x29	41_{10}	10.5	14.6	14.6	14.6	7.8
0x2A	42_{10}	11.5	15.2	15.2	15.2	8.1
0x2B	43_{10}	12.6	15.7	15.7	15.7	8.4
0x2C	44_{10}	13.9	16.4	16.4	16.4	8.7
0x2D	45_{10}	15.2	17.0	17.0	17.0	9.0
0x2E	46_{10}	16.7	17.7	17.7	17.7	9.4
0x2F	47_{10}	18.3	18.4	18.4	18.4	9.7
0x30	48_{10}	20.0	19.1	19.1	19.1	10.1
0x31	49_{10}	22.0	19.9	19.9	19.9	10.5
0x32	50_{10}	24.1	20.8	20.8	20.4	11.0
0x33	51_{10}	26.4	21.7	21.7	20.6	11.4
0x34	52_{10}	29.0	22.7	22.7	24.9	11.9
0x35	53_{10}	31.8	23.7	23.7	25.2	12.4
0x36	54_{10}	34.9	24.8	24.8	25.3	13.0
0x37	55_{10}	38.2	26.0	25.3	25.5	13.5
0x38	56_{10}	41.9	27.3	29.0	26.0	14.2
0x39	57_{10}	46.0	28.7	30.0	26.5	14.8
0x3A	58_{10}	50.4	30.2	31.0	27.0	15.5

Continued on the next page.

Continued from the previous page.

Table Index		Pan		Tilt		
		Spectra	Esprit	Spectra		Esprit
				NTSC	PAL	
		°/sec	°/sec	°/sec	°/sec	°/sec
0x3B	59_{10}	55.3	31.8	32.0	38.0	16.3
0x3C	60_{10}	60.7	33.6	33.6	29.0	17.1
0x3D	61_{10}	66.5	35.6	35.6	37.6	18.0
0x3E	62_{10}	72.9	37.7	37.0	40.0	19.0
0x3F	63_{10}	80.0	40.0	44.0	44.0	20.0

1.8.6 Q: How do I change the pan/tilt speeds that are available to me on a fixed speed system?

A: Use the RAIL RIGHT, FAST and RAIL LEFT keys to get the indicated speeds. (In an Esprit the speeds will differ and in future versions of the Spectra they may also differ. In future revisions of the equipment, the speed available with the TXB-S422 may change. Be sure to use the manual that came with your unit.)

1.8.7 Q: I notice that when I am using a fixed speed keyboard to pan a camera, and I zoom the lens at the same time. That the panning rate changes. I thought that panning speed stayed the same in a fixed speed system. What gives?

A: You probably have proportional pan enabled. As you pan and zoom simultaneously, you are able to watch the effects that proportional pan has on the pan rate of the Spectra.

1.9 Maintenance Information

1.9.1 Q: Gosh, this must have been difficult to debug. How was it done?

A: Very carefully with built in debug code. To enable debug mode:

- Set bit 6 on the switch.
- Connect an RS-232 ASCII terminal to pin 7 and ground of U1 in the TXB-S422 as follows:
  - For a 25 pin RS-232 connector:
    - U1-7 (J4-square) to pin 3 on a DB-25 type connector.
    - Ground (J4-circular) to pin 7 on a DB-25 type connector.
  - For a 9 pin RS-232 connector:
    - U1-7 (J4-square) to pin 2 on a DB-9 type connector.
    - Ground (J4-circular) to pin 5 on a DB-9 type connector.

  Then on the left side of the screen there will be a command count and the command data as it was received from the Sensormatic equipment. On the right side there is the command data that is being sent to the Spectra.

  Sometimes in debug mode Sensormatic commands will be lost, but so what? You are running in debug mode, not real mode.

  The RS-232 ASCII terminal must be set up for:

  - 115200 baud
  - One start bit
  - One stop bit
  - Seven or eight data bits
  - No parity

1.9.2 Q: Does the debug display show up on the video monitor?

A: No it does not, it only shows up on an external serial ASCII terminal. It is a debugging aid and showing this type of information on a video monitor would make debugging the interface to the Spectra very difficult.

1.9.3 Q: What does this display look like? I saw the writeup, but it still isn't clear to me what it contains.

A: It is unlikely that anyone other than a software maintainer of the TXB-S422 program will ever need to know the answer to that question. However here is a small sample of what it looks like, with some notes as the contents of each field for a some keyboard button pushes and some "pan" and "tilt" command sequences from an AD2083/02 code translator driven by an AD2078 variable speed keyboard connected to an AD2050 controller.

```
    |A|  |<------B----->|   C
 1 "5A   06 C0 85 21 94 --> 06"
 2 " 5B  FF 06 00 10 00 00 16"
 3 "5C   06 86 74 --> 06"
 4 " 5D  FF 06 00 00 00 00 06"
 5 "5E   06 C0 82 04 B4 --> 06"
 6 " 5F  FF 06 00 02 8A 00 92"

 7 "60   06 C0 85 03 B2 --> 06"
 8 " 61  FF 06 00 10 04 00 1A"
 9 "62   06 C0 82 5A 5E --> 06"
10 " 63  FF 06 00 12 B5 03 D0"
11 "64   06 C0 85 21 94 --> 06"
12 " 65  FF 06 00 10 5A 00 70"
13 "66   06 C0 85 2D 88 --> 06"
14 " 67  FF 06 00 10 5A 00 70"
15 "68   06 83 77 --> 06"
16 " 69  FF 06 00 10 00 00 16"
17 "6A   06 8B 6F --> 06"
18 " 6B  FF 06 00 40 00 00 46"
19 "6C   06 8C 6E --> 06"
20 " 6D  FF 06 00 00 00 00 06"
21 "6E   06 8B 6F --> 06"
22 " 6F  FF 06 00 40 00 00 46"
```

Note that the numbers on the left ($1 \rightarrow 22$), the letters on the top ($A \rightarrow C$) and the quote marks " are only used for documentation and do not appear on the screen.

Between data lines 6 and 7 there is a blank line. This is placed there to make counting the number of commands easier. A blank line is output every 16 commands.

Notes to the above sample listing.

A This is a hex field that increments by one for each command received from a Sensormatic controller or for commands sent to a Spectra dome. It is an 8 bit number that wraps at `0xFF` around to `0x00`.

For commands that come from a controller the field is left justified, for commands sent out to a Spectra it is a right justified value.

It is rare that the total number of commands sent from a controller is important, but it is quite often important to know if two or three identical commands were sent. So by carefully watching the Ⓐ column, it is easy to identify multiple command sendings. (As long as there are less than 256 of them!)

B For commands that originate from a controller, those whose sequential ID is left justified, this is the command as received by the TXB-S422. Most commands are three bytes long. For

longer commands the first two bytes and the last byte are the same as the bytes on a three byte command, only additional information is placed inside the message after the op-code byte and before the checksum byte.

An arrow follows the command and the acknowledgment from the dome is then printed out. (Here all acknowledgements are `0x06`, the dome's address.)

1. First comes one byte as an address for the command. This is used by the TXB-S422 and the Spectra to actually receive the message. If the first byte does not match the address switches on the Spectra the message is thrown away.
2. The second byte is the Sensormatic op-code.
3. *Optional* at this place there may be several extra bytes of command information. The most common information is that required for variable speed commands and preset commands.
4. Then comes a checksum for the command.

C When this is the output from the TXB-S422, indicated by having the sequential ID being right justified, this is the D protocol command generated by the TXB-S422. The various bytes are:

1. The first byte is the "sync" byte and always consists of all ones.
2. The next byte is the D protocol address byte which gets sent to the Spectra. In all of this example the commands were sent to unit #`6`, i.e. the Spectra and the Sensormatic/-American Dynamics keyboard were both set to `6`.
3. Then next two bytes are the "Command 1" and "Command 2" bytes of the D protocol.
4. The command bytes are followed by the "Data 1" and " Data 2" bytes of the D protocol.
5. Last there is the D protocol checksum. Note that it does not include the "sync", or first byte in it.

It should be noted that:

1. The pan bit is set on second command byte of D protocol.
2. The pan speed is indicated in data byte 1 and starts out at other than "zero speed" and goes to a high speed (not "turbo speed", I didn't push the joy stick enough) and then back down.
3. Most commands are three bytes long.

The Sensormatic commands used in this example are:

1. `0x83` Stop Pan, three bytes in length.
2. `0x86` Stop Tilt, three bytes in length.
3. `0x8B` Zoom out, three bytes in length.

4. `0x8C` Stop zoom, three bytes in length.
5. `0xC0 0x82` Variable speed pan right, five bytes in length.
6. `0xC0 0x85` Variable speed tilt down, five bytes in length.

1.9.4 Q: In addition to the serial debug data that you output, are there any other debug aids that will be left in the final product?

A: No.

1.9.5 Q: What are the new part numbers for the TXB-S422?

A: An effort was made to not introduce any new part numbers into the Pelco supply system. As a result there are very few additional part numbers required for the TXB-S422. The most significant of these are:

Unknown

Each time a software revision is made, the revision number on the `IC51`, `PG51`, `BH51` and `FW00` parts, must be updated.

1.10 Q: How closely does the Spectra match the capabilities of my Sensormatic domes?

A: The Spectra dome does not have some of the capabilities/concepts of the Sensormatic equipment, and it has some capabilities/concepts that the Sensormatic equipment does not have. (The most obvious of these it that Pelco does not implement "Tour"s.)

Still other capabilities/characteristics are implemented differently. (Here the most obvious difference is that Pelco makes extensive use of "menus" that are generated by the Spectra unit. While Sensormatic's menus are different in form and content. Both techniques have similar capabilities.)

The TXB-S422 attempts to "hide" as many of the differences as possible. However Sensormatic and Pelco developed, and are actively maintaining and enhancing, their domes independently. Some differences are inevitable. The following items attempt to list the known differences, non-compatibilites and work arounds.

There is an additional problem with defining differences between Sensormatic's domes and Pelco's units. These problems revolve around the fact that each model of Spectra tends to have some differences from others in the same product line. I.e. a Spectra (being a dome type unit) can "look straight down and do a flip". While an Esprit (being a pan/tilt type unit) can look straight down, but can not continue to rotate on through zero degrees and continue to observe most subjects. And just to complicate matters more, both Pelco and Sensormatic make continuous fixes, updates and changes to their products. So what is correct today may change by tomorrow.

The net result of the various problems mentioned here is that we believe the information to be correct as of "press time" however since then, there may be changes that are inevitable and unpredictable. (Then there may have been some changes that I haven't found out about yet.)

[7] `$Header: d:/sears/RCS/faq2.inc,v 1.18 2001-11-28 11:50:40-08 Hamilton Exp Hamilton $`

Pelco has an additional problem with our various models of the Spectra/Esprit some have some subtle and not so subtle differences from model to model. (The Spectra-Lite can not do proportional pan while the other members of the Spectra family can, etc.)

This list of capabilities comes from the manual for the DeltaDome II.

1. **Area Boundaries** Sensormatic allows up to 16 "Areas". Pelco has up to 8 "Zones". The two are similar but not identical. One of the biggest differences is that when a camera crosses a boundary, a message is sent to the controller. The Spectra does not do this.
2. **On-Screen Text Displays** Sensormatic has four different types of on screen text displays. A Spectra may only turn on/off the displaying of "Zone" and "Preset" labels. With the TXB-S422 installed all zone labels are of the form `ZONE x` where `x` is the zone number. Likewise all preset labels are of the form `PRESET xx` with `xx` being the preset number.
3. **Text Attribute Options** Sensormatic has two attributes each of which may be selected. The Spectra's only attribute is whether or not the display of characters is enabled or not.
4. **Language/Password** Sensormatic has six language options and the ability to use a password for access to camera menus. The Spectra only uses the english language and does not have password protection.
5. **Presets** Sensormatic domes support up to 96 presets. The Spectra supports 64 presets.
6. **Name Configuration** Sensormatic domes have internal names for:
   A. Camera Names
   B. Area Names
   C. Preset Names
   D. Pattern Names
   E. Alarm Names

   The Spectra only has internal names for Zones and Presets.
7. **Pan/Tile/Zoom/Shutter Settings** Sensormatic has many options for their cameras. The Spectra has a similar set of options which vary depending on the exact model, sub-model, of dome under discussion. All Spectra are accessed via the Spectra's menu system.
8. **Camera and Lens Functions** See the previous item. Sensormatic and the Spectra have similar capabilities.
9. **Alarm Actions** Sensormatic domes support four alarms. Each one may activate any of the presets available ($1 \rightarrow 96$), or any of three patterns or do nothing. The Spectra supports seven alarms which are tied to presets $1 \rightarrow 7$ or cause no action.

Sensormatic may have alarms initiate a message to the controller. The Spectra sends the status of its alarms out following the receipt of any valid message (alarm status is part of the command acknowledge message.) Thus the Spectra will give out the status of alarms on receiving every "poll" command and following every other valid command. The TXB-S422 will send an alarm message whenever the command acknowledgment message indicates that an alarm has occurred.

10. **Alarm Input Normal States** Sensormatic and the Spectra are roughly compatible here, with the different being the options available to the Sensormatic type units and the number if alarms possible with the Spectra. For more information see the answers to the above two points.

11. **Home Position** Both Sensormatic and the Spectra have similar capabilities here with the details setup via the dome's menu system.

12. **North Position** The Spectra does not support this concept.

13. **Privacy Zones** The Spectra does not support this concept.

14. **Presets**

Presets			
Equipment	Communications	Presets	Presets with TXB-S422 installed
TXB-S422	**RS422**	**64**	—
RC20, RC32, RC58	RS422	4	4
VM16/ADTT16	SensorNet 485	96	64?
VM32/AD32	SensorNet 485	96	64?
AD2150	Manchester	64	—
With AD2083/02	RS422	16	16
VM96	Virtual, RS422 or SensorNet 485	64	64
VM168/AD168	Manchester, RS422 or Sensornet	64	—
With AD2083/02	RS422	16	16
AD2050	Manchester	64	—
With AD2083/02	RS422	16	16

Table 1. Sensormatic controller types

Type font usage

In this table the following conventions have been followed for type font usage.

[8] `$Header: d:/sears/RCS/presets.inc,v 1.28 2001-11-30 15:12:03-08 Hamilton Exp Hamilton $`

1. *Every thing in italics is used to give amplifying information. In some places there are short statements about what is happening. These usually consist of statements such as "Not Implemented" and "Use the Spectra's Menu". Their meaning should be self explanatory.*

   A common example of this is:

   This means that the original item is not implemented and that the Spectra's Menu must be used. The small box on the right is used in debugging so that the tester may place a mark here when that item is tested.

2. The small caps font is used to identify actual markings on the equipment. The only equipment that I have consistent access to is an American Dynamics model AD2083/02 type data translator and AD2078 keyboard with an AD2050 controller. "Real" Sensormatic equipment may be marked differently, or even work differently!

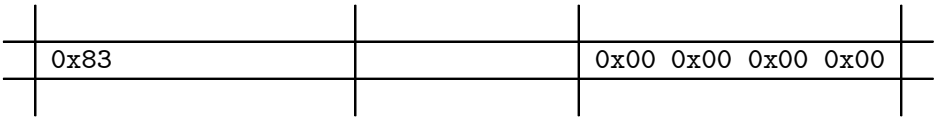
   For example:

   This means that the information is for an "CALL" keyboard sequence.

3. `The teletype font is used to identify command strings, in hex, that are received or are expected to be used when transmitted by the TXB-S422.`

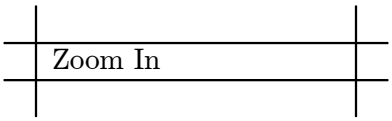
   For example:

   Here the Sensormatic equipment sends a command with `0x83` as bytes 2 (the other bytes are the unit ID and checksum). The TXB-S422 generates and sends a command containing `0x00 0x00 0x00 0x00` as bytes 3, 4, 5 and 6 (the other bytes are sync, address and checksum).

4. Items in "normal font" are used for various descriptions of what the commands are or do.

   For example:



5. Horizontal lines have different levels of significance.
   A. A "block of commands" refers to all commands that have the same numeric value. I.e. these consist of up to four commands (OFF, ON, CALL, and SET) for, say, 22.
   B. A single horizontal line is used to separate parts of block of commands.
   C. A double horizontal line is used to separate different blocks of commands.
   D. A triple horizontal line is used to indicate where several blocks of commands have been omitted.

Sensormatic Command		Type	Controller data	Comments	Output data	
TARGET	1	RC58	0xB4	Go To Preset #1	0x00 0x07 0x00 0x01	
"Use menu"		RC58	0xA9	Preset set #1	0x00 0x03 0x00 0x01	
1	AUXILIARY OFF	AD2083/02	0xE0	Auxiliary 1 Off	—	
1	AUXILIARY ON	AD2083/02	0xE1	Auxiliary 1 On	—	
1	PRESET CALL	AD2083/02	0xA6 0x33 0x01 0x06 0x01 0x26 0x01 0x28 0x01 0x00 0x00	Go To Preset #1	0x00 0x07 0x00 0x01	
1	PRESET SET	AD2083/02	0xA5	Preset set #1	0x00 0x03 0x00 0x01	
TARGET	2	RC58	0xB5	Go To Preset #2	0x00 0x07 0x00 0x02	
"Use menu"		RC58	0xAA	Preset set #2	0x00 0x03 0x00 0x02	
2	AUXILIARY OFF	AD2083/02	0xE0	Auxiliary 3 Off	—	
2	AUXILIARY ON	AD2083/02	0xE2	Auxiliary 3 On	—	
2	PRESET CALL	AD2083/02	0xA6 0x33 0x02 0x06 0x02 0x26 0x02 0x28 0x02 0x00 0x00	Go To Preset #2	0x00 0x07 0x00 0x02	
2	PRESET SET	AD2083/02	0xA5	Preset set #2	0x00 0x03 0x00 0x02	
TARGET	3	RC58	0xB6	Go To Preset #3	0x00 0x07 0x00 0x03	
"Use menu"		RC58	0xAB	Preset set #3	0x00 0x03 0x00 0x03	
3	AUXILIARY OFF	AD2083/02	0xE0	Auxiliary 3 Off	—	
3	AUXILIARY ON	AD2083/02	0xE4	Auxiliary 3 On	—	
3	PRESET CALL	AD2083/02	0xA6 0x33 0x03 0x06 0x03 0x26 0x03 0x28 0x03 0x00 0x00	Go To Preset #3	0x00 0x07 0x00 0x03	

Continued on the next page.

Continued from the previous page.

Sensormatic Command		Type	Controller data	Comments	Output data	
3	PRESET SET	AD2083/02	0xA5	Preset set #3	0x00 0x03 0x00 0x03	
TARGET	4	RC58	0xB7	Go To Preset #4	0x00 0x07 0x00 0x04	
"Use menu"		RC58	0xAC	Preset set #4	0x00 0x03 0x00 0x04	
4	PRESET CALL	AD2083/02	0xA6 0x33 0x04 0x06 0x04 0x26 0x04 0x28 0x04 0x00 0x00	Go To Preset #4	0x00 0x07 0x00 0x04	
4	PRESET SET	AD2083/02	0xA5	Preset set #4	0x00 0x03 0x00 0x04	
TARGET	5	RC58	0xBC	Go To Preset #5	0x00 0x07 0x00 0x05	
"Use menu"		RC58	0xB9	Preset set #5	0x00 0x03 0x00 0x05	
5	PRESET CALL	AD2083/02	0xA6 0x33 0x05 0x06 0x05 0x26 0x05 0x28 0x05 0x00 0x00	Go To Preset #5	0x00 0x07 0x00 0x05	
5	PRESET SET	AD2083/02	0xA5	Preset set #5	0x00 0x03 0x00 0x05	
TARGET	6	RC58	0xBD	Go To Preset #6	0x00 0x07 0x00 0x06	
"Use menu"		RC58	0xBA	Preset set #6	0x00 0x03 0x00 0x06	
6	PRESET CALL	AD2083/02	0xA6 0x33 0x06 0x06 0x06 0x26 0x06 0x28 0x06 0x00 0x00	Go To Preset #6	0x00 0x07 0x00 0x06	
6	PRESET SET	AD2083/02	0xA5	Preset set #6	0x00 0x03 0x00 0x06	
TARGET	7	RC58	0xBE	Go To Preset #7	0x00 0x07 0x00 0x07	
"Use menu"		RC58	0xBB	Preset set #7	0x00 0x03 0x00 0x07	
7	PRESET CALL	AD2083/02	0xA6 0x33 0x07 0x06 0x07 0x26 0x07 0x28 0x07 0x00 0x00	Go To Preset #7	0x00 0x07 0x00 0x07	

Continued on the next page.

Continued from the previous page.

Sensormatic Command	Type	Controller data	Comments	Output data
7 PRESET SET	AD2083/02	0xA5	Preset set #7	0x00 0x03 0x00 0x07
8 PRESET CALL	AD2083/02	0xA6 0x33 0x08 0x06 0x08 0x26 0x08 0x28 0x08 0x00 0x00	Go To Preset #8	0x00 0x07 0x00 0x08
8 PRESET SET	AD2083/02	0xA5	Preset set #8	0x00 0x03 0x00 0x08
9 PRESET CALL	AD2083/02	0xA6 0x33 0x09 0x06 0x09 0x26 0x09 0x28 0x09 0x00 0x00	Go To Preset #9	0x00 0x07 0x00 0x09
9 PRESET SET	AD2083/02	0xA5	Preset set #9	0x00 0x03 0x00 0x09
10 PRESET CALL	AD2083/02	0xA6 0x33 0x0A 0x06 0x0A 0x26 0x0A 0x28 0x0A 0x00 0x00	Go To Preset #10	0x00 0x07 0x00 0x0A
10 PRESET SET	AD2083/02	0xA5	Preset set #10	0x00 0x03 0x00 0x0A
11 PRESET CALL	AD2083/02	0xA6 0x33 0x0B 0x06 0x0B 0x26 0x0B 0x28 0x0B 0x00 0x00	Go To Preset #11	0x00 0x07 0x00 0x0B
11 PRESET SET	AD2083/02	0xA5	Preset set #11	0x00 0x03 0x00 0x0B
12 PRESET CALL	AD2083/02	0xA6 0x33 0x0C 0x06 0x0C 0x26 0x0C 0x28 0x0C 0x00 0x00	Go To Preset #12	0x00 0x07 0x00 0x0C

Continued on the next page.

Continued from the previous page.

Sensormatic Command	Type	Controller data	Comments	Output data	
12 PRESET SET	AD2083/02	0xA5	Preset set #12	0x00 0x03 0x00 0x0C	
13 PRESET CALL	AD2083/02	0xA6 0x33 0x0D 0x06 0x0D 0x26 0x0D 0x28 0x0D 0x00 0x00	Go To Preset #13	0x00 0x07 0x00 0x0D	
13 PRESET SET	AD2083/02	0xA5	Preset set #13	0x00 0x03 0x00 0x0D	
14 PRESET CALL	AD2083/02	0xA6 0x33 0x0E 0x06 0x0E 0x26 0x0E 0x28 0x0E 0x00 0x00	Go To Preset #14	0x00 0x07 0x00 0x0E	
14 PRESET SET	AD2083/02	0xA5	Preset set #14	0x00 0x03 0x00 0x0E	
15 PRESET CALL	AD2083/02	0xA6 0x33 0x0F 0x06 0x0F 0x26 0x0F 0x28 0x0F 0x00 0x00	Go To Preset #15	0x00 0x07 0x00 0x0F	
15 PRESET SET	AD2083/02	0xA5	Preset set #15	0x00 0x03 0x00 0x0F	
16 PRESET CALL	AD2083/02	0xA6 0x33 0x10 0x06 0x10 0x26 0x10 0x28 0x10 0x00 0x00	Go To Preset #16	0x00 0x07 0x00 0x10	
16 PRESET SET	AD2083/02	0xA5	Preset set #16	0x00 0x03 0x00 0x10	

Sensormatic Command	Type	Controller data	Comments	Output data	
17 PRESET CALL	AD2083/02	0xAC	End Pattern #1	0x00 0x1F 0x00 0x00	
17 PRESET SET	AD2083/02	0xA0	Define Pattern #1	0x00 0x21 0x00 0x00	
18 PRESET CALL	AD2083/02	0xAD	End Pattern #2	0x00 0x1F 0x00 0x01	
18 PRESET SET	AD2083/02	0xA1	Define Pattern #2	0x00 0x21 0x00 0x01	
19 PRESET CALL	AD2083/02	0xAE	End Pattern #3	0x00 0x1F 0x00 0x02	
19 PRESET SET	AD2083/02	0xA2	Define Pattern #3	0x00 0x21 0x00 0x02	
20 PRESET CALL	AD2083/02	0xAF	GoTo Pattern #4	—	
21 PRESET CALL	AD2083/02	0xB0	Run Pattern #1	0x00 0x23 0x00 0x00	
22 PRESET CALL	AD2083/02	0xB1	Run Pattern #2	0x00 0x23 0x00 0x01	
23 PRESET CALL	AD2083/02	0xB2	Run Pattern #3	0x00 0x23 0x00 0x02	
24 PRESET CALL	AD2083/02	0xB3	Run Pattern #4	—	
25 PRESET CALL	AD2083/02	0x9A 0xAC 0xB0 0x9B	Auto Repeat Pattern #1	—	
26 PRESET CALL	AD2083/02	0x9A 0xAD 0xB1 0x9B	Auto Repeat Pattern #2	—	
27 PRESET CALL	AD2083/02	0x9A 0xAE 0xB2 0x9B	Auto Repeat Pattern #3	—	
28 PRESET CALL	AD2083/02	0x9A 0xAF 0xB3 0x9B	Auto Repeat Pattern #4	—	
33 PRESET SET	AD2083/02	0xB8	End Pattern Definition	—	
34 PRESET SET	AD2083/02	0xA3	Accept (save) New Pattern	—	
40 PRESET CALL	AD2083/02	0x8D 0x8E	Flip	0x00 0x07 0x00 0x21	
41 PRESET SET	AD2083/02	0xA0 0x8E 0xB8 0xA3 0x8F	Clear Pattern #1	—	

Continued on the next page.

Continued from the previous page.

Sensormatic Command	Type	Controller data	Comments	Output data	
42 PRESET SET	AD2083/02	0xA1 0x8E 0xB8 0xA3 0x8F	Clear Pattern #2	—	
43 PRESET SET	AD2083/02	0xA2 0x8E 0xB8 0xA3 0x8F	Clear Pattern #3	—	
51 PRESET SET	AD2083/02	0x95	Set Switch 1 Closed	0x00 0x09 0x00 0x01	
52 PRESET SET	AD2083/02	0x95	Set Switch 1 Open	0x00 0x0B 0x00 0x01	
53 PRESET SET	AD2083/02	0x95	Set Switch 2 Closed	0x00 0x09 0x00 0x02	
54 PRESET SET	AD2083/02	0x95	Set Switch 2 Open	0x00 0x0B 0x00 0x02	
61 PRESET SET	AD2083/02	0x95	Set Switch 3 Closed	0x00 0x09 0x00 0x03	
62 PRESET SET	AD2083/02	0x95	Set Switch 3 Open	0x00 0x0B 0x00 0x03	
63 PRESET SET	AD2083/02	0x95	Set Switch 4 Closed	0x00 0x09 0x00 0x04	
64 PRESET SET	AD2083/02	0x95	Set Switch 4 Open	0x00 0x0B 0x00 0x04	
66 PRESET CALL	AD2083/02	0x9A 0x8B	Reset Dome, step 1	—	
66 PRESET SET	AD2083/02	0x88 0x90	Reset Dome, step 2	0x00 0x0F 0x00 0x00	
67 PRESET SET	AD2083/02	0x9A	Enter V-Phase Adjust Mode	—	
68 PRESET SET	AD2083/02	0x9B	Exit V-Phase Adjust Mode	—	
69 PRESET CALL	AD2083/02	0x90 0x91	Return to Auto-Iris/Auto Focus	—	
69 PRESET SET	AD2083/02	0x90 0x91	Reset Iris	—	
70 PRESET CALL	AD2083/02	0xE0	Set Auxiliary 4 Off	—	
71 PRESET CALL	AD2083/02	0xE8	Set Auxiliary 4 On	—	

Sensormatic Command	Type	Controller data	Comments	Output data
88 PRESET CALL	AD2083/02	—	Insert IR filter	0x00 0x07 0x00 0x58
89 PRESET CALL	AD2083/02	—	Remove IR filter	0x00 0x07 0x00 0x59
90 PRESET SET	AD2083/02	—	Manual left limit stop	0x00 0x03 0x00 0x5A
91 PRESET SET	AD2083/02	—	Manual right limit stop	0x00 0x03 0x00 0x5B
92 PRESET SET	AD2083/02	—	Scan left limit stop	0x00 0x03 0x00 0x5C
93 PRESET SET	AD2083/02	—	Scan right limit stop	0x00 0x03 0x00 0x5D
95 PRESET SET	AD2083/02	—	Enable Menu Mode	0x00 0x03 0x00 0x5F
96 PRESET CALL	AD2083/02	—	Stop a Scan	0x00 0x07 0x00 0x60
97 PRESET CALL	AD2083/02	—	Activate Random Scanning	0x00 0x07 0x00 0x61
98 PRESET CALL	AD2083/02	—	Activate Frame Scanning	0x00 0x07 0x00 0x62
99 PRESET CALL	AD2083/02	—	Activate Continous Scanning	0x00 0x07 0x00 0x63

APPENDIX A

A Command decoding

The following Sensormatic RS422/RS485 commands are decoded. There are additional notes included with many of them to indicate what actually gets done when they are received.

This is a typical entry:

`0x80`	This is the second byte of a command from the controller to the TXBS422. The first byte is always the address and the last byte is always a checksum.
Unknown three byte command.	This is a short description of the command.
ACK. Send `0xC0` to the controller, nothing to the Spectra.	This is what the TXBS422 does with the command. In this case it Acknowledges the command by sending its address to the controller. The TXBS422 then builds up a three byte reply of <address>, `0xC0` and <checksum> and sends it to the controller. In this case the TXBS422 sends nothing to the Spectra. On many commands it will send a D Protocol command to the Spectra.

0x00 → 0x7F Undefined messages which get no response. Nothing is sent to the Spectra.

0x80 Unknown command. ACK. Send `0xC0` to the controller, nothing to the Spectra.

0x81 PAN LEFT. ACK. Send a default Pan Left speed of 27^o/sec to the Spectra. (See comments at the end for a discussion on how the actual speed values are determined for all three detected types of speeds.)

0x82 PAN RIGHT. ACK. Send a default Pan Right speed of 27^o/sec to the Spectra. (See comments at the end for a discussion on how the actual speed values are determined for all three detected types of speeds.)

0x83 STOP PANNING. ACK. Restore default pan speed of 27^o/sec. Send a command with both pan motion bits cleared.

0x84 TILT UP. ACK. Send a default Tilt Up speed of 12^o/sec to the Spectra. (See comments at the end for a discussion on how the actual speed values are determined for all three detected types of speeds.)

0x85 TILT DOWN. ACK. Send a default Tilt Down speed of 12^o/sec to the Spectra. (See comments at the end for a discussion on how the actual speed values are determined for all three detected types of speeds.)

[9] `$Header: d:/sears/RCS/decoded.inc,v 1.8 2002-02-28 12:57:20-08 Hamilton Exp Hamilton $`

0x86 STOP TILTING. ACK. Restore default tilt speed of 12^o/sec. Send a command with both tilt motion bits cleared.

0x87 FOCUS NEAR. ACK. Command sent to the Spectra with Focus Far bit set and Focus Near cleared. The pan/tilt data is not changed if movement is currently taking place.

0x88 FOCUS FAR. ACK. Command sent to the Spectra with Focus Near bit set and Focus Far cleared. The pan/tilt data is not changed if movement is currently taking place.

0x89 STOP FOCUS. ACK. Command sent to Spectra with both Focus bits cleared. However pan/tilt data is still valid if motion is still occurring.

0x8A ZOOM IN. ACK. Command sent to the Spectra with Zoom In bit set and Zoom Out cleared. The pan/tilt data is not changed if movement is currently taking place.

0x8B ZOOM OUT. ACK. Command sent to the Spectra with Zoom Out bit set and Zoom In cleared. The pan/tilt data is not changed if movement is currently taking place.

0x8C STOP ZOOM. ACK. Command sent to Spectra with both Zoom bits cleared. However pan/tilt data is still valid if motion is still occurring.

0x8D Increase pan and tilt speeds to $108/44^o$/sec ("FAST"). ACK. Send new pan/tilt command to Spectra with new values if moving. Might be step one of a Flip command.

0x8E Increase pan and tilt speeds to $54/19^o$/sec ("FASTEST"). ACK. Send new pan/tilt command to Spectra with new values if moving. Might be step one of a Flip command. If this is part two of a Flip command then send a Preset move command to Preset 33.

0x8F STOP FAST/FASTEST speeds (back to normal $27/12^o$/sec). ACK. Send new pan/tilt command to Spectra with new values if moving.

0x90 OPEN IRIS. ACK. Command sent to the Spectra with Open Iris bit set and Close Iris cleared. The pan/tilt data is not changed if movement is currently taking place.

0x91 CLOSE IRIS. ACK. Command sent to the Spectra with Close Iris bit set and Open Iris cleared. The pan/tilt data is not changed if movement is currently taking place.

0x92 STOP IRIS adjustment. ACK. Command sent to Spectra with both Iris bits cleared. However pan/tilt data is still valid if motion is still occurring.

0x93 STOP ALL movement. ACK. Send a no motion command to the Spectra. Reset all motion speeds to default values.

0x94 REQUEST DOME TYPE (poll). No ACK. Send a dome type response of `0xE8` to the controller. In a real Sensormatic dome this value should change, but with the TXBS422 it doesn't. The change appears to occur after a preset is sent to the dome.

0x95 REQUEST ALARM STATUS. No ACK. Send `0x00` to the controller, nothing is sent to the Spectra.

0x96 Unknown command. ACK. Send `0xC0` to the controller, nothing is sent to the Spectra.

0x97 ACKNOWLEDGE TO DOME. No ACK. Nothing to the Spectra.

0x98 START TEMP NO TRANSMIT. No ACK. Nothing to the Spectra.

0x99 END TEMP NO TRANSMIT. No ACK. Nothing to the Spectra.

0x9A Increase pan and tilt speeds to $81/30^o$/sec ("FASTER"). ACK. Send new pan/tilt command to Spectra with new values if moving.

0x9B STOP FASTER speeds (back to normal $27/12^o$/sec). ACK. Send new pan/tilt command to Spectra with new values if moving.

0x9C DEFINE BOUNDARY. ACK. Send `0xC4` to the controller, nothing to Spectra.

0x9D MARK BOUNDARY. ACK. Send `0xB0` to the controller, nothing to Spectra.

0x9E SET ON AIR status to tell the dome to send the ID of the current boundary. ACK. Send `0xB0` to the controller, nothing to the Spectra.

0x9F RESET ON AIR status. ACK. Nothing to the Spectra.

0xA0 DEFINE PATTERN 1. ACK. Send Start Defining Pattern 1 to the Spectra.

0xA1 DEFINE PATTERN 2. ACK. Send Start Defining Pattern 3 to the Spectra.

0xA2 DEFINE PATTERN 3. ACK. Send Start Defining Pattern 3 to the Spectra.

0xA3 ACCEPT NEW PATTERN. ACK. Run the most recent pattern defined to the Spectra.

0xA4 MEMORY DUMP. No ACK. Send a huge 104 byte message to the controller. This response was copied from one captured from a freshly powered up single board SpeedDome 2000 (dome #5).

0xA5 REQUEST DOME POSITION coordinates. ACK. Send a 12 byte response to the controller. <ID>, `0x00`, `0x00`, `0x12`, `0xXX`, `0x26`, `0xXX`, `0x28`, `0xXX`, `0x31`, `0xXX`, <checksum>. (`0xXX` indicates that this is the current preset ID.)

This request is normally sent three times by the controller. The TXBS422 sets a timer following the first one and ignores any following ones in the next one or two seconds.

0xA6 GOTO ABSOLUTE POSITION (Multiple-byte format). No ACK. Send the value in byte 4 of the 13 byte long message to the Spectra as a preset value.

0xA7 Undetected unknown command, place holder. No ACK.

0xA8 MARK TARGET 1. ACK. Sends a Set Preset #1 to the Spectra.

0xA9 MARK TARGET 2. ACK. Sends a Set Preset #2 to the Spectra.

0xAA MARK TARGET 3. ACK. Sends a Set Preset #3 to the Spectra.

0xAB MARK TARGET 4. ACK. Sends a Set Preset #4 to the Spectra.

0xAC GOTO START OF PATTERN 1. ACK. Send a Run Pattern #1 to the Spectra.

0xAD GOTO START OF PATTERN 2. ACK. Send a Run Pattern #2 to the Spectra.

0xAE GOTO START OF PATTERN 3. ACK. Send a Run Pattern #2 to the Spectra.

0xAF GOTO START OF PATTERN 4. ACK. Send a Run Pattern #3 to the Spectra. (Note that usage of pattern #3 is not a "typo" as the Spectra only has three patterns.)

0xB0 RUN PATTERN 1. ACK. Send a Run Pattern #1 to the Spectra.

0xB1 RUN PATTERN 2. ACK. Send a Run Pattern #2 to the Spectra.

0xB2 RUN PATTERN 3. ACK. Send a Run Pattern #3 to the Spectra.

0xB3 RUN NEW PATTERN to review it . ACK. Nothing to the Spectra.

0xB4 GO TO TARGET 1. ACK. Sends a Goto Preset #1 to the Spectra.

0xB5 GO TO TARGET 2. ACK. Sends a Goto Preset #2 to the Spectra.

0xB6 GO TO TARGET 3. ACK. Sends a Goto Preset #3 to the Spectra.

0xB7 GO TO TARGET 4. ACK. Sends a Goto Preset #4 to the Spectra.

0xB8 Tells the dome to STOP RECORDING (defining). ACK. Send a command to the Spectra to stop defining a pattern.

0xB9 MARK TARGET 5. ACK. Sends a Set Preset #5 to the Spectra.

0xBA MARK TARGET 6. ACK. Sends a Set Preset #6 to the Spectra.

0xBB MARK TARGET 7. ACK. Sends a Set Preset #7 to the Spectra.

0xBC GO TO TARGET 5. ACK. Sends a Goto Preset #5 to the Spectra.

0xBD GO TO TARGET 6. ACK. Sends a Goto Preset #6 to the Spectra.

0xBE GO TO TARGET 7. ACK. Sends a Goto Preset #7 to the Spectra.

0xBF Unknown command, place holder. No ACK.

0xC0 VARIABLE SPEED CONTROL (Multiple-byte format). ACK. Interprets byte 3 for motion type: `0x81` = Pan Left, `0x82` = Pan Right, `0x84` = Tilt Up and `0x85` = Tilt Down. Byte 4 is the speed in ^{o}/sec. (See comments at the end for a discussion on how the actual speed values are determined for all three detected types of speeds.)

0xC1 Unknown command. ACK. Nothing else.

0xC2 Unknown command. No ACK. Nothing is sent to the Spectra.

0xC3 PROPORTIONAL SPEED pan or tilt movement. ACK. Nothing else is done as I couldn't determine which axis is used and I never found it in an `.CAP` file.

0xC4 Unknown command. No ACK. Nothing is sent to the Spectra.

0xC5 Unknown command. No ACK. Nothing is sent to the Spectra.

0xC6 RUN DEFAULT "Apple Peel" pattern for a spiral. ACK. A Reset command (`0x0F`) is sent to the dome. Then a full reinitialization is done to the TXBS422.

0xC7 Unknown command. No ACK. Nothing is sent to the Spectra.

0xC8 Unknown command. No ACK. Nothing is sent to the Spectra.

0xC9 Get SOFTWARE VERSION from dome. ACK. A message containing: <ID>, `0x09`, `0x06`, `0x07`, `0x01`, `0x00`, `0x01`, `0x03`, `0x16`, <checksum>. Is sent to the controller.

0xCA → 0xDF Unknown commands. No ACK. Nothing is sent to the Spectra.

0xE0 SET OUTPUT DRIVERS range = 0xE0 → 0xEF. ACK. This set of eight commands are bit encoded to turn on/off all four auxiliaries in the Sensormatic type of domes. Receiving this command always results in sending four Auxiliary commands to the Spectra to individually turn on/off each of the first four Auxiliary outputs.

0xF0 TERMINATE CURRENT PATTERN. ACK. Send an 0xB5 to the controller. Send nothing to the Spectra as D Protocol doesn't have a "Stop Pattern" command.

0xF1 → 0xFF Unknown commands. No ACK. Nothing is sent to the Spectra.

Sensormatic Dome speeds

In working with Sensormatic domes I have detected at least three different systems of specifying speeds for pan and tilt operations to proceed at.

1. For the "fixed speed" (`0x81`, `0x82`, `0x84` and `0x85`) commands there are four different speeds for each of pan and tilt. These are controlled by using the "Finger Tracker" to generate the base line speeds of 27^o/sec for pan and 12^o/sec for tilt. These speeds may be increased by factors of 2X, 3X or 4X by simultaneously using the Finger Tracker **and** depressing one of the FASTEST (RAIL RIGHT gives 2X), FASTER (FAST gives 3X), or FAST (RAIL LEFT gives 4X) keys on the VM-1 (RC20, RC32 or RC58) control keyboard. (The names of these speeds come from a protocol document that we have. The speeds come from using a stop watch and an attempt to get as close as we can to the actual speed of a SpeedDome 2000. The keyboard key names come from what is written on them.)

Key	Name	Increase	Pan Speed	Tilt Speed
"plain"	—	X1	27	12
RAIL RIGHT	FASTEST	X2	54	19
FAST	FASTER	X3	81	30
RAIL LEFT	FAST	X4	108	44

2. For the semi-variable speed units, such as the AD2083/02, there are a total of eight speeds that are sent out (using `0xC0`) depending on how far the joy stick is moved. Translating these speeds is done using a modified table lookup. I had to translate the eight input speeds into Spectra speeds by using "ranges". This means that for any input speed that falls within a given results in a given output speed. The ranges are:

Speed	Input Value	Decision Low	Decision High	Output Value	Input Value	Decision Low	Decision High	Output Value
1	`0x04`	`0x00`	`0x05`	7	`0x03`	`0x00`	`0x04`	3
2	`0x06`	`0x06`	`0x08`	15	`0x05`	`0x05`	`0x07`	11
3	`0x0A`	`0x09`	`0x0C`	23	`0x09`	`0x08`	`0x0B`	19
4	`0x0F`	`0x0D`	`0x13`	31	`0x0D`	`0x0C`	`0x10`	27
5	`0x18`	`0x14`	`0x1C`	39	`0x14`	`0x11`	`0x16`	35
6	`0x21`	`0x1D`	`0x27`	47	`0x18`	`0x17`	`0x1C`	43
7	`0x2D`	`0x28`	`0x43`	55	`0x21`	`0x1D`	`0x27`	51
8	`0x5A`	`0x44`	—	64	`0x2D`	`0x28`	—	63

In the above table it should be noted that the symbol "—" in the last of the high speed ranges indicates that all speeds higher than the indicated low value generates the given output.

3. True variable speed commands. These commands are generated using the same `0xC0` commands that are used with the AD2083/02. The values used are in °/sec this gives more "granularity" to them. Whereas with the AD2083/02 there are a total of eight different values per axis, with the "TouchTracker" there may be up to 100 different speeds generated. (So far only around 30 have been detected.)

   These speeds cause some consternation until examination reviewed that the presence of a RC216/Touch Tracker could be identified by examining the pan speed requests. That pan speeds of 1, 2, 80_{10} and 99_{10} are unique to the RC216/Touch Tracker pair. Thus a check is made on all pan speeds related to `0xC0` commands and if they are found to be from a true variable speed device, a flag is set.

   When a true variable speed pan command is detected, the input speed is multiplied by 75% and 8 is added to it. If the result is greater than 64 then the speed is forced to 64 and used. The reasoning behind this is that many of the input speed are too big for the Spectra to use (several of them are "turbo" speed values) and that all Spectra pan speeds of seven and less are the same.

   When a true variable speed tilt command is detected, the input speed is multiplied by 50% and 6 is added to it. If the result is greater than 63 then the speed is forced to 63 and used. The reasoning behind this is that many of the input speed are too big for the Spectra to use and that all Spectra tilt speeds of six and less are the same.

   The speeds detected so far are:

APPENDIX B

B Speeds of an RC216 controller

Here I have a listing of the detected variable speed values from an RC216 controller. In this listing different speeds were detected for pan and tilt. In some cases there are some missing speed values, these were probably caused when the operator "skiped over" them.

Note

- In the higher pan speeds that these speeds appear to increase in steps of $8_{10}{}^{o}$/sec (0x08) with a limit of $99_{10}{}^{o}$/sec (0x63).
- In the higher tilt speeds that these speeds appear to increase in steps of $2_{10}{}^{o}$/sec or $4_{10}{}^{o}$/sec with a limit of 56^{o}/sec 0x38).
- At lower speeds the increments are quite small, i.e. steps of 1^{o}/sec and 2^{o}/sec are common.
- Data is based on speed values found in the 14JAN02A.CAP file, which was collected at Sears Clovis on their RC216 (VM96) system.
- PM is used to indicate that this speed value is Probably Missing, as it has never been detected as comming from an RC216.
- PT is used to indicate that this speed value is Probably There, even though it was not detected in this sample data capture.

Left 0x81	Right 0x82	Up 0x84	Down 0x85	°/sec ---
0x01	0x01	0x01	0x01	1
0x02	0x02	0x02	0x02	2
0x03	0x03	0x03	0x03	3
0x04	0x04	0x04	0x04	4
0x05	0x05	PT	PT	5
0x06	0x06	0x06	PT	6
0x07	0x07	0x07	PT	7
0x08	0x08	0x08	0x08	8
0x09	0x09	0x09	0x09	9
0x0A	0x0A	0x0A	0x0A	10
0x0B	0x0B	0x0B	0x0B	11
0x0C	0x0C	0x0C	0x0C	12
0x0D	0x0D	PT	PT	13

Continued on the next page.

[10] $Header: d:/sears/RCS/speeds.inc,v 1.4 2002-02-08 16:05:59-08 Hamilton Exp Hamilton $

Continued from the previous page.				
Left	Right	Up	Down	°/sec
0x81	0x82	0x84	0x85	---
0x0E	0x0E	0x0E	PT	14
0x0F	0x0F	0x0F	0x0F	15
0x10	0x10	0x10	0x10	16
0x11	0x11	0x11	0x11	17
0x12	0x12	0x12	PT	18
0x13	PT	PT	PT	19
0x14	0x14	0x14	0x14	20
PM	PM	PT	0x15	21
0x16	0x16	0x16	0x16	22
PM	PM	0x17	PT	23
0x18	0x18	0x18	0x18	24
PM	PM	PT	PT	25
PM	PM	0x1A	PT	26
PM	PM	0x1B	PT	27
0x1C	0x1C	0x1C	0x1C	28
PM	PM	0x1E	PT	30
0x20	0x20	0x20	0x20	32
PM	PM	0x22	PT	34
0x24	0x24	0x24	0x24	36
0x28	0x28	0x28	0x28	40
PM	PM	0x2C	0x2C	44
0x30	0x30	0x30	0x30	48
PM	PM	0x32	0x32	50
PM	PM	0x34	0x34	52
0x38	0x38	0x38	0x38	56
0x40	0x40	PM	PM	64
0x48	0x48	PM	PM	72
0x50	0x50	PM	PM	80
0x58	0x58	PM	PM	88
0x63	0x63	PM	PM	99

APPENDIX C

C Recommended testing instructions

```
$Header: d:/sears/RCS/testing.txt,v 1.3 2002-02-08 07:41:09-08 Hamilton Exp
Hamilton $

Special instructions to be used when manufacturing a TXB-S422.

1. Test equipment required in addition to the American Dynamics equipment
   listed in item 4 below:

    A. Items required if the TXB-S422 is to be programmed at Pelco:

        a. A programming/test jig. This will provide 12V DC to the board
           for programming the PIC CPU and allow access to pins 6, 8, 10
           and 12 for resistance checks.

        b. An ohm meter.

        c. Programming cable and a PC to drive it. This will allow
           connections from a PC to the PIC. (And anything else needed
           for programming the PIC.)

    B. Items required if the TXB-S422 is to use preprogrammed parts:

        a. A test jig. This will allow access to pins 6, 8, 10 and 12 for
           resistance checks.

        b. An ohm meter.

2. When programming the PIC, U1 which is a PIC 16F876, do not attempt to
   verify that the chip has been correctly programmed. The chip has its
   "code protect" logic enabled and you will ALWAYS get an error. (In
   fact if it does compare then an error has occurred!)

3. While the TXB-S422 is in the programming jig, check that the two
   termination resistors may be connected and removed from the host line
   in the following manner.

        A. Remove all power and the programmer from any contact with the
           board.
```

B. Connect an ohm-meter between pins 6 and 8. With switch 7 open it should be a high value. With switch 7 closed it should be about 120 ohms.

C. Connect an ohm-meter between pins 10 and 12. With switch 8 open it should be a high value. With switch 8 closed it should be about 120 ohms.

When finished this testing step, LEAVE switches 7 and 8 ON!

4. For testing a fully built TXB-S422 the following items are required:

A. An American Dynamics Keyboard, type AD2087.

B. An American Dynamics CPU, type AD1996.

C. An American Dynamics Code Translator, type AD2083/02A.

D. An American Dynamics Power Supply, type TP8129.

E. An American Dynamics Terminal Block, type

F. A Spectra and its power supply.

G. A monitor directly connected to the Spectra.

H. Cableing on an as needed basis.

If it is necessary to use a substitute, verify with me (Eric Hamilton x3375) that it has a chance of working and being useful.

5. Connect everything up. For most of the equipment, each item has a different connector. So this should be a simple step.

6. Set the Spectra to some address from 1 to 99. (DO NOT use address 64!) Later on this address will have to be entered into the American Dynamics equipment, but for now just remember it. (It also shows up on the monitor screen during the Spectra's power up sequence.)

7. Turn on the Monitor so that it will "warm up" and be able to display the initial Spectra configuration screens. Then the Spectra. When this

is done all the switch positions on the TXB-S422 should be turned ON.

8. The monitor will now show the normal Spectra screen with the second line replaced with the TXB-S422's configuration line.

    "TXB-S422 Rev Debug3F"

   This indicates that the TXB-S422 is correctly installed and that most portions of the switch will turn on correctly. It is not possible for software to detect if which switch positions 7 and 8 are in.

9. Turn off the Spectra and change the switch so that switch positions 1-6 are OFF, and power up the Spectra again. (Switch positions 7 and 8 are to be shipped in the ON position, so they should be left that way.)

10. The monitor will now show the normal Spectra screen with the second line replaced with the TXB-S422's configuration line.

    "TXB-S422 Rev x.xx 00"

    This indicates that the TXB-S422 is correctly installed and that most portions of the switch will turn OFF correctly. It should be possible to control the camera at this time. If the American Dynamics equipment and the Spectra have the same address.

    x.xx will usually be the software revision level, however there are two exceptions to this and they are:

    A. The word "BETA" will be used for all beta test software.
    B. The word "TEST" is used for internal test software that MUST NEVER be sent to a customer.
    C. The third option is that the word "DEBUG" will appear whenever bit 6 of the switch is turned on. This indicates that the unit is in debug mode. This is completely normal and debug mode is always available.

    Note: Following a power up, the first command to the Spectra will erase the configuration message.

11. On the AD2078 keyboard, enter in a monitor number ("1" then "monitor" key). If this isn't done then the keyboard will not act on any other

keystrokes, and it doesn't give any indication that nothing is happening! (Note that actually having a monitor attached somewhere to the American Dynamics equipment isn't needed. It only has to be specified.)

12. On the AD2078 keyboard, enter in a camera number (the number that you set the Spectra to in step 6, above).

    Note: A monitor and camera number only have to be entered once, as long as the Spectra address doesn't change, per powering up of the American Dynamics system.

13. Try to cause the Spectra to move. If it moves then testing for this TXB-S422 is over. There is a special problem with the American Dynamics system in that it will not recognize the Spectra if it doesn't "see" it go through its power up sequence. So the Spectra/TXB-S422 MUST be connected to the American Dynamics equipment before the Spectra is powered up for this step in testing.

14. And you are done! Be sure to ship the TXB-S422 with switch positions 1-6 OFF and positions 7 and 8 ON.

APPENDIX D

Index