Cambridge Digital Direct Vario (302)

Data Port User’s Guide SV-517 Revision 2.2

March 25, 2003

I. Introduction

This document describes the Cambridge 302 Data Port functions.

The Data Port is the 302's data interface to the outside world.   It is used to test and configure the 302 as well as to transfer data between the 302 and a terminal device. Data transfer from a terminal device to the 302 is called downloading, and data transfer from the 302 to the terminal device is called uploading.

The 302 Data Port connects to an asynchronous serial communications port of a computer running specialized software, but several 302 functions can be performed by a user at the keyboard of a terminal or terminal emulator. In this document, the term "user" could mean either the computer program or the person at the terminal.

The information in this document is valid as of the date on the cover page.

II. Hardware Issues

The Data Port of the 302 is a DB-9F connector wired as DCE (it sends data on pin 2 and receives data on pin 3; pin 5 is ground). The other RS-232 pins are not currently supported.

It is useful to understand the three types of 302 memory:

1. Internal Flash ROM: This memory contains the BIOS and security functions. There is no user access to this memory.

2. External Flash ROM. This memory contains the application firmware, configuration data, waypoints, and flight logs. There is no direct access to this memory. The various commands described in the document read and write from this memory as appropriate

3. Static RAM. This memory contains the variables necessary to run the application

4. One-Wire Battery-Backed Static RAM. This memory contains variables that are frequently changed but need to survive power cycles.

III. General Operation

The power-up settings for the port are 4800,N,8,1. The baud rate can be changed.

Logically, the port can be in one of the following modes:

Command mode
Log or Spew mode
Download mode
Upload mode
Calibrate mode
Pocket-Nav mode

At power-up, the port is in Log Mode 0 (NMEA Output).

At any time, the port can be switched to Command mode by sending it a Ctrl-C (03H).

In Command mode, the port can issue two error messages. If it receives a command it doesn't understand it will respond with "What?". If the command is understood but cannot be performed for any reason, the response will be "Not allowed". Sometimes a numeric error code is issued.

Command mode does not support editing. You can’t even use the backspace key!

In Command mode, Download mode, and Upload mode, the port sends a prompt to indicate that it is ready to accept the next character string from the user. The prompt is different for each of the three modes.

Command mode -- 7 bytes -- CR LF LF cmd>

Download mode -- 9 bytes -- XX YY ZZ CR LF LF dn>

The three bytes XX, YY, and ZZ are error-checking. See Section IV.

Upload mode -- 6 bytes -- CR LF LF up>

IV. Error Checking

Data transmission error-checking is available to the user in Download Port and Upload Port modes. No error checking is available in the other modes. The use of error checking by the user is optional.

It is useful at this point to define two character strings. The "Request" is the character string from the user to the 302. The first byte of the Request is the first byte received by the 302, and the last byte is the byte before the CR which terminates the Request (the CR is not considered part of the Request). The "Reply" is the character string from the 302 to the user. The first byte of the Reply is the XX byte, and the last byte is usually the ">" character.

There are two types of replies: Small and Large. Small Replies are 255 bytes or less in length. Large replies can be any length. Only the Upload Port issues large replies. Whether the command is large or small depends on the command and is noted with each command description in section VIII.

The 302 transmits three bytes of error-checking data for small replies, and 5 bytes of error checking data for large replies. No error-checking data is expected from the user. For small replies the three bytes are called XX, YY, and ZZ. For large replies the 5 bytes are XX XX YY ZZ ZZ. XX XX and ZZ ZZ for large replies are the 16 bit counterparts to XX and ZZ for small replies.

This section explains the use of XX, YY and ZZ for small replies:
The first byte of the Reply, XX, is the binary representation of the number of bytes in the Reply. (This means that the maximum length of the Reply is 255 decimal). The count includes the XX byte itself. For example, if the Reply is the Download Port prompt, then XX will always be 09H.

The second error-checking byte, YY, is the checksum of the Request. The checksum is calculated by sequentially XORing every byte of the Request. For example, if the Request to the Upload Port is "C 32" followed by CR, the YY byte of the Reply will be 62H.

The third error-checking byte, ZZ, is the checksum of the Reply. The checksum is calculated by XORing from the fourth byte of the Reply to the end of the Reply (i.e. the checksum calculation does not include the XX, YY, ZZ bytes). The ZZ byte of the Download Port prompt will always be 39H.

This section explains the use of XX XX, YY and ZZ ZZ for large replies:
The first two bytes of the Reply, XX XX, are the binary representation of the number of bytes in the Reply. The count includes the XX XX word itself. For example, if the Reply is the Download Port prompt, then XX will always be 09H.

The third error-checking byte, YY, is the checksum of the Request. The checksum is calculated by sequentially XORing every byte of the Request. For example, if the Request to the Upload Port is "C 32" followed by CR.

The forth and fifth error-checking bytes, ZZ ZZ, ar the checksum of the Reply. The checksum is calculated by XORing from the sixth byte of the Reply to the end of the Reply (i.e. the checksum calculation does not include the XX XX, YY, ZZ ZZ bytes).

The user can provide optional error checking for download port commands. Download error checking is provide by adding an extra field to the download command. The field consists of a tilde character (~) followed by the 16 bit checksum (In Ascii Format) of the command. The checksum is calculated by adding all of the characters of the command up to but not including the comma delimeter that begins the error checking field. If the error checking field is included, and an error is detected, the reported check sum of the request is incremented by one. If the error checking field is not included, no error checking is performed on download commands.

V. COMMAND MODE

The 302 Data Port always powers up in Log 0 mode at 4800 baud. See section VI. Command Mode is entered any time the port receives a Ctrl-C (03H) character.

When this mode is entered, the port sends the characters "Command Port", followed by CRLF (0DH, 0AH). This is followed by the Command mode prompt.

In this mode, the port accepts commands. Some commands leave the port in Command mode. Other commands change the port mode. Commands are alpha strings that may be abbreviated to the first three letters of each word, and are case insensitive. Commands may take alpha or numeric arguments. Commands are terminated with CR (0DH).

Commands:

VERSION	Gives 302 version, serial number etc.
CLEAR POINTS	Erases all Navpoints.
CLEAR PILOT	Erases Pilot Name
CLEAR LOG	Erases all log memory.

BAUD N Select baud rate. The new baud rate depends on N:

N	Baud
4	1200
5	2400
6	4800
7	9600
8	19200
9	38400
10	57600
11	115200

START Starts logging unconditionally.

STOP Stops unconditional logging.

DIE Turns power off if possible

FAT N Gives information about the Nth (Starting from 0) flight log in the instrument.

SIF Software Interface command; See section XI.

SCR Sets Screen Control Variables; See section XI.

KEY Dumps a part of Battery-Backed RAM; See section XI.

VOL Set audio volume 0 is loudest, 170 is silent.

NMEA Passes the GPS datastream through to the port.

GAR Gives the settings for the Garmin GPS receiver.

The following commands change the Data Port operating mode.
The command Ctrl-C always switches the 302 back to Command mode.

LOG or SPEW Switches to Log mode. See section VI.

DOWNLOAD Switches to Download mode. See section VII.

UPLOAD Switches to Upload mode. See section VIII.

CALIBRATE Switches to Calibrate model. See section IX.

PNP Switches to Pocket-Nav Mode under certain conditions. See section X.

VI. LOG MODE

Important Note: The 302 powers up in Log 0 mode at 4800 baud.

In LOG mode the instrument transmits a set of data every second. Data content depends on LOG sub mode. LOG sub-mode is determined by LOG or SPEW argument.

Most LOG sub-modes are used for debugging purposes; transmitted data can change from version to version. The following sub-modes have wider use and will be preserved through Firmware Version changes.

LOG 0 NMEA Output. See section XII for the specification of the NMEA output.

LOG 8 Data Logging information. This sub-mode shows the following:
A hexadecimal representation of the latest record,
The record length,
The address where the next record will be stored,
And the date and time of the latest record.

LOG 12 Raw sensor outputs. This sub-mode shows the following:
24 bit absolute pressure (~13000000 at sea level)
24 bit Pitot/static pressure(~8400000 with no pressure)
G Meter X Axis (32768 +- 2000 at zero G and 8000 Cts./G)
G Meter Y Axis (32768 +- 2000 at zero G and 8000 Cts./G)
Barometer sensor channel; raw outputs vary.
Barometer sensor Temperature channel; raw outputs vary.
10 bit Backup Battery Voltage.
10 bit Engine Noise Level.
10 bit Main Power Voltage.
Hexidecimal representation of “Port 7” logic levels.
OAT probe (High negative value (< -400) indicates trouble)

VII. DOWNLOAD MODE

When this mode is entered, the words "Download Port" (followed by CRLF) are transmitted. This is followed by the Download Port prompt. To exit this mode, type Ctrl-C.

To enter Download Port mode from the Command Port, the user issues the DOWNLOAD command, which may take an argument:

DOW 0 (or DOW with no argument) enables echoing

DOW 1 disables echoing

Note: When using this mode at baud rates of 19200 and higher, use of the echoing feature is recommended.

In Download mode, the 302 accepts data from the terminal device. Information is downloaded line-by-line (i.e. data followed by a CR, followed by a new prompt). The user should wait for the new prompt before sending the next line. The type of information being downloaded is called the "message type" and is identified by the first character of the line (the "message identifier"). The message identifier is an alphabetic character and may be in upper or lower case. Messages consist of displayable ASCII characters, and fields within a message are delimited by commas.

Here is a summary of the messages:

Identifier	Type
C	Waypoint
D	Declaration
G	Polar
O	Pilot

A detailed description of each message follows.

DOWNLOAD MESSAGE C: NAVPOINT

Message format:

```
   C,reserved,latitude,longitude,elevation,NPID,ATTR,NAME,REMARK
```

Here are the rules:

1.1. All characters are displayable ASCII, except the CR which terminates the message.

2. There must be at least six delimiters (commas) in the message.

3. The reserved field is ignored.

4. Latitude is always ten characters: DDMM.MMMMH, where H is 'N' or 'S', and must be in upper case.

5. Longitude is always eleven characters: DDDMM.MMMMH, where H is 'E' or 'W', and must be in upper case.

6. Elevation is in meters, whole numbers.

7. Navpoint ID (NPID) range is 1-9999.

8. Attribute (ATTR) range is 1-65535.

9. The "text" field begins after the seventh delimiter of the message and contains a 12-character NAME, and a 12-character REMARK. Any displayable character is legal in the NAME or REMARK. In the special case of points whose "Airspace" attribute bit is set, the REMARK field must contain three four-byte quantities (leading zeroes required): radius (tenths of nautical miles, 1-65535), floor (meters, 0-65535), ceiling (meters, 0-65535). For example, a REMARK of 025000002134 on a navpoint with the Airspace attribute sets the radius to 25.0 nmi, the floor to zero, and the ceiling to 2134 meters.

10. The handling of missing data:

11. Points carrying the Airspace bit may not have "short" NAME or REMARK fields. Other points may have short fields, and the 302 will pad them with blanks to yield 12 characters for NAME and 12 characters for REMARK.

Attribute encoding

The 16-bit ATTR is encoded as follows:

Bit	Meaning
0	Turnpoint
1	Airfield
2	Markpoint
3	Landing Point
4	Start Point
5	Finish Point
6	Home Point
7	Thermal Point
8	Waypoint (none of the above)
9	Airspace
10-15	Reserved

Example C messages:

```
C,7,4404.8000N,07200.5000W,185,1,11,Dean Memorl
C,4,4411.7000N,07145.0000W,223,2,11,Franconia   Rwy 36 S end
C,3,4310.7000N,07303.3000W,306,3,1,Manchester  Town Center
C,12,4415.5000N,07122.5000W,451,4,1,Mt Wash HotlHotel Bldg
C,13,4353.0500N,07215.2500W,213,101,122,Post Mills  Intersection
C,10,4241.7000N,07310.7000W,129,5,1,N Adams     Lakes
C,11,4351.1000N,07248.4000W,274,6,9,Rochester   Island
C,1,4446.0000N,07231.3000W,715,7,1,Belvidere   Mine Bldgs
C,2,4438.7000N,07250.5000W,70,8,1,Cambridge   Road X
C,5,4456.4000N,07305.8000W,102,9,11,Franklin CtyRwy 1 S end
C,6,4313.2000N,07242.8000W,240,10,11,N Windham   Rwy 9 W end
C,8,4413.3000N,07248.0000W,442,11,11,Philips     Rwy 30 S end
C,9,4346.7000N,07145.2000W,130,12,512,Restricted  025000002134
```

DOWNLOAD MESSAGE D: DECLARATION

Message format:

```
A: D,ord+128,latitude,longitude,NAME,Elevation (Enters a turnpoint)
B: D,255                                       (Saves the declaration)
```

Here are the rules:

1. Declaring a task is a two step process. First, use format A above to enter the individual turnpoints. Second, use format B above to save the declaration (Declare the Task)

2. The Turnpoints must be entered in order starting with 128 (128 is ordinal 0).

3. The task is not declared until format B is sent.

4. Latitude is always ten characters: DDMM.MMMMH, where H is 'N' or 'S', and must be in upper case.

5. Longitude is always eleven characters: DDDMM.MMMMH, where H is 'E' or 'W', and must be in upper case.

6. Name is up to 12 characters and can consist of any displayable characters. If Name is less than 12 characters it is padded with spaces.

7. Elevation is optional. If entered the units are meters. Elevation is not part of the declaration, it used for 303 navigational purposes only.

Example Declaration using the D Message:

```
D,128,4404.8000N,07200.5000W,Dean Memorl
D,129,4411.7000N,07145.0000W,Franconia
D,130,4310.7000N,07303.3000W,Manchester
D,131,4415.5000N,07122.5000W,Mt Wash Hotl
D,132,4404.8000N,07200.5000W,Dean Memorl
D 255
```

DOWNLOAD MESSAGE G: GLIDER INFORMATION

Message format:

```
G,GliderType,GliderID,L/D,VM,V2,Dry Weight, Liters, Reserved, Config Word
```

Here are the rules:

1. GliderType can be up to 12 bytes of displayable ascii characters. If GliderType is less than 12 characters it will be padded with spaces.

2. GliderID can be up to 12 bytes of displayable ascii characters. If GliderID is less than 12 bytes it will be padded with spaces.

3. The rest of the fields are numeric. They are defined as follows:

L/D	Best L/D
VM	Best L/D speed
V2	2 M/S sink speed
Dry Weight	In KG.
Liters	Ballast capacity
Reserved	Ignored
ConfigWord	Bits 15-1 of the Config word are reserved. Bit 0 controls the configuration lock feature. A 0 in this bit indicates that the Configuration is locked.

Example G Message:

```
G,Ventus CM,RB,48,98,172,200,60,0,65535
```

DOWNLOAD MESSAGE O: PILOT INFORMATION

```
O,NAME,N|UNITS,TUNITS,SINK,TEFG,DIFF,DATUM,APP,ARR,SLOW,FAST,GAP,SPD,DB,R,UW,MH
```

Here are the rules:

1. NAME can be up to 24 bytes of displayable ascii characters. If NAME is less than 24 characters it will be padded with spaces. If the NAME field is the single character ‘~’, the second argument will determine the active pilot.

2. The second argument is divided into two bytes. The low byte is old style units. High byte is The ordinal of the record.

   Units are designated by a number:

   0: NM,FT,KTS
   1: KM,M,KPH
   2: KM,FT,KTS
   3: SM,FT,KTS
   This is ignored if the Unit Word field is sent

   Ordinal:

   0: Replace active pilot record
   64 + N: Replace Pilot Record N
   255: Just add to the list if possible.

   Note: This was not done to torture software developers. It was done to maintain backward compatibility with old Utility programs that only knew about one pilot record.

3. The rest of the fields are defined as follows:

   TUNITS:1 = F

```
               2 = C
        SINK:  1 = Sink Tone On
               0 = Sink Tone Off
        TEFG:  1 = Total Energy Final Glide
               0 = No Total Energy Final Glide
        DIFF:  1 = Show Final Glide Altitude Difference
               0 = Show Absulute Altiude required
        DATUM: Map Datum (Ignored in IGC Versions)
        APP:   Approach Radius (Meters)
        ARR:   Arrival Radius (Meters)
        Slow:  Enroute Logging Interval (Seconds)
        Fast:  Close Logging Interval (Seconds)
        Gap:   Time between Flight Logs (Minutes)
        SPD:   Minimum Speed to force flight logging (Knots)
        DB:    STF Dead Band (M/S)
        R:     Reserved
        UW:    Unit Word: Bits described below
               Bit    Description
               0:     Vario
                      0:     M/S
                      1:     Kts
               1:     Altimeter
                      0:     Meters
                      1:     Feet
               2:     Temperature
                      0:     Celcius
                      1:     Fahrenheight
               3:     Barometer
                      0:     Millibars
                      1:     InHG
               4-5:   Distance
                      00:    Km
                      01:    NMI
                      10:    SMI
                      11:    Reserved
               6-7:   Speed
                      00:    M/S
                      01:    Kts
                      10:    MPH
                      11:    Reserved
               8-15:  Reserved
        MH:    Margin Height (10ths of Meters)
```

4. Information for multiple pilots may be stored in the 302

DOWNLOAD MESSAGE Y: USER DATA

Y,TYPE,OFFSET,LENGTH,DATA…

There are 512 bytes of general storage available to users. The Y message is used to store data in this area. The data can be stored directly as text or as hexidecimal represented binary data depending on the TYPE field.

Here are the rules:

1. TYPE describes the nature of the data. 0: Text 1:Hex
2. OFFSET is the offset into the 512 byte storage area.
3. LENGTH describes the length of the data. When the type is 1 length describes the actual number of bytes to be stored not the length of the transmitted data. For TYPE = 0 maximum length is 64. For TYPE = 1 maximum length is 32.
4. If TYPE is 0 DATA is stored directly as transmitted. If TYPE is one the data is a hex string (EE44DDAA would store the numbers 0EEH, 044H, 0DDH, 0AAH)

5. If length causes an overflow of the bounds of the user data area the sentence will be ignored.

VII. UPLOAD MODE

Upload mode is used to transfer data, including flight logs, from the 302 to the user. When this mode is entered, the words "Upload Port" (followed by CRLF) are transmitted. This is followed by the Upload mode prompt. To exit this mode, type Ctrl-C.

To enter Upload mode from Command mode, the user issues the UPLOAD command, which may take an argument:

```
      UPL 0   (or UPL with no argument) enables echoing

      UPL 1   disables echoing
```

Note: When using this mode at baud rates of 19200 and higher, use of the echoing feature is recommended.

In Upload mode, there are sub-commands which are used to request specific data from the 302. The sub-commands are terminated with CR and are case insensitive:

Sub-Command	Requests:
B	Flight logs and related data
C	Navpoints
D	The declaration
G	Polar Information
O	Pilot Information
W	Miscellaneous information

Unlike download messages, uploaded data is not necessarily in ASCII format. Most of the uploaded messages are in binary format and are of fixed length. At the end of each message, there will be a CRLF followed by the upload prompt.

Replies to the sub-commands begin with three or five error-checking bytes (see Section IV) and end with the Upload Port prompt:

```
       XX YY ZZ ...data... CR LF LF up>
```

A detailed description of the data portion of the response to each sub-command follows.

DATA IN REPLY TO SUB-COMMAND B: FLIGHT LOGS, ETC.

Error Checking: LARGE

The B sub-command is used for the following functions:

- Getting information about log files that are present in the 302.
- Initiating log file transfer.
- Transferring log files.
- Transferring log file signature.

B 196 + N
This variant of the B command gets information about log files N*8 to N*8+7. For example B 196 will get information about log files 0 through 7; B 197 will get information about log files 8 through 15. N must be less than 8.

This variant returns the following:

Fixed Length: 289 bytes

Element	Length	Comments
Num Files	1	Number of log files stored in 302
Start UTC	6	YYMMDDHHMMSS Start time of file
End UTC	6	YYMMDDHHMMSS End time of file
NAME	24	Pilot’s Name
Etc…	7 more copies of Start,End, and Name	

B 64 + N
This variant of the B command initiates an ascii format (IGC format in most cases) log file transfer of the Nth file. Log file 0 is always the most recent. N must be less than 64.

This variant returns the following:

Fixed Length: 5 bytes

Element	Length	Comments
Result	1	Y: File available N:File not available.
BBP	2	Bytes that will be transferred per block.
Num Blocks	2	APPROXIMATE number of blocks to be transferred.

Note: Num Blocks is approximate. It should not be used for array boundaries. See “B X” below for how to determine when transfer is complete.

B X
This variant of the B command transfers blocks of log file data. If X = ‘N’ the next block of data is transferred, and checksum info is updated. If X = ‘R’ the last block is retransferred and checksum info is not updated.

This variant returns the following:

Length: BBP+2 (Obtained from “B 64 + N” See Above) bytes

Element	Length	Comments
Valid Bytes	2	Number of bytes in block that apply to the file.
BLOCK	BBP	File data.

Note: If Valid Bytes is not equal to BBP the file transfer is complete.

B S
This variant of the B command transfers the signature in ascii format. This document does not cover the contents of the signature.

Fixed Length: 203 bytes

Element	Length	Comments
Size	2	Size of the signature
Sig	201	Signature

B 256 + N
This variant of the B command initiates in binary format (Direct memory dump) log file transfer of the Nth file. Log file 0 is always the most recent. N must be less than 64.

This variant returns the following:

Fixed Length: 8 bytes

Element	Length	Comments
Result	1	Y: File available N:File not available.
BBP	2	Bytes that will be transferred per block.
Num Bytes	3	Number of bytes to be transferred.
Logging Code	2	Information about the format of logging.

Note: Num bytes is usually accurate. See “B X” above for how to determine when transfer is complete.

DATA IN REPLY TO SUB-COMMAND C: Navpoints

Error Checking: SMALL

C <No Arg>
This variant of the C command gets the number of Navpoints and the size of a waypoint record.

Fixed Length: 3 bytes

Element	Length	Comments
Num Points	2	Number of Navpoints stored in the box
WSize	1	Size of a Navpoint record

C N
This variant of the C command gets a particular Navpoint.

Length: WSize (obtained in the C <No Arg> Command above)

Element	Length	Comments
Latitude	4	In TTOM (See below for explanation)
Longitude	4	In TTOM (See below for explanation)
Elevation	2	In Feet
ID	2	
Attribute	2	See C command in section VII
Name	12	
Remark	12	

Notes:
The units TTOM are Ten Thousandths of Minutes.

For latitude, the equator is at 54000000 TTOM. Latitudes north of the equator add, and latitudes south of the equator subtract.

For longitude, Greenwich England is at 108000000 TTOM. Longitudes east of Greenwich add, and longitudes west of Greenwich subtract.

For example, the coordinates 4413.1094N 07247.4121W correspond to 80531094 TTOM and 64325879 TTOM respectively.

DATA IN REPLY TO SUB-COMMAND D: Declaration

Error Checking: LARGE

D <no Args>
This is the only form of the D command. It gets the declaration

Fixed Length: 310

Element	Length	Comments
Date/Time	6	YYMMDDHHMMSS
Num Points	1	Number of Navpoints in the declaration.
Reserved	1	
Latitude	4	In TTOM see C Command
Longitude	4	In TTOM see C Command
Name	12	
14 more repetitions of Lat,Lon, and Name		
Reserved	2	

DATA IN REPLY TO SUB-COMMAND G: Polar

Error Checking: SMALL

G <No Args>

This variant of the G Command gets the size of a polar record.

Fixed Length: 1

Element	Length	Comments
PSize	1	Size of a polar record in bytes

G 0
This variant of the G Command gets the polar record

Length: PSize

Element	Length	Comments
Type	12	Glider Type
ID	12	Glider ID
L/D	1	Best L/D
VM	1	Best Glide Speed
V2	1	2 M/S Sink Speed
Reserved	1	
Weight	2	In Liters
Liters	2	Ballast capacity
Reserved	2	
Config	2	Config word (See G Command Sect VII)

DATA IN REPLY TO SUB-COMMAND O: Pilot Info

Error Checking: SMALL

O <No Args>
This variant of the O Command gets the number of Pilot records stored in the 302 and the size of a pilot record

Fixed Length: 2

Element	Length	Comments
NPilots	1	Number of pilot records stored
PSize	1	Size of a pilot record (bytes)

O A
This variant of the O Command gets the number of Pilot records stored in the 302, the size of a pilot record, and the index of the pilot record that is active

A	Mandatory argument upper or lower class 'A'

Fixed Length: 3

Element	Length	Comments
NPilots	1	Number of pilot records stored
PSize	1	Size of a pilot record (bytes)
APIndex	1	Index of the active pilot record

O N
This variant of the command gets a pilot record from the 302 dependant on N as follows:

N < 128	Gets active pilot record
N >= 128	Gets Pilot Record N-128

Length: PSize (Obtained from the O <No Args> or O A Command (See Above)

Element	Length	Comments
NAME	24	
Units	1	
Temp Units	1	Temperature Units
Sink Tone	1	
TEFG	1	
Diff Alt	1	
Datum	1	
App	2	
Arr	2	
Slow	2	
Fast	2	
Gap	2	
Speed	2	
Deadband	1	
Reserved	1	
Units Word	2	
Reserved	2	
Margin Height	2	

Note: See O command in section VII for interpretation of the fields.

O B S N
This variant of the command gets number of pilot records as a block.

Error Checking: Large

B	Mandatory argument upper or lower case 'b'
S	Start active pilot ordinal
N	Number of records in the block

Length: PSize * N (Psize Obtained from the O <No Args> or O A Command (See Above)

Note: The maximum block size 1024 bytes. Do not pass an N that would make the Block size larger than 1024 bytes

DATA IN REPLY TO SUB-COMMAND W: General Info

Error Checking: SMALL

W <No Arg>

Length: 32 bytes

Element	Length	Comments
Reserved	15	
ID	3	Base 36 Ascii
TYPE	1	‘P’ = Prototype ‘F’ = Production Firmware
VERSION	5	Version Number
Reserved	5	
302 ID	1	Always 3
Reserved	2	

Notes: TYPE and VERSION fields read as all 0 in versions previous to F17000

DATA IN REPLY TO SUB-COMMAND Y: User Data

Error Checking: LARGE

Y OFFSET,LENGTH

Length: LENGTH

Element	Length	Comments
OFFSET	2	Offset into the 512 byte user data area
LENGTH	2	Number of bytes to transmit

Note: Retrieves a number bytes from the user area. If LENGTH overflows the bounds of the USER area the affected bytes of the reply will be undefined.

IX. CALIBRATION MODE

The Calibration Port is used to calibrate various sensors in the instruments. When Calibration Port is entered, the outputs of all the sensors are diplayed in user units. A menu of possible modes is also shown. The Calibration Mode is selected by pressing the key corresponding to that mode. The Calibration Modes are as follows.

A	Barometer and Variometer absolute pressure sensors
G	G-Meters
S	Dynamic pressure (Airspeed) differential pressure sensor
T	Temperature Compensation of Airspeed pressure sensor
V	Variometer absolute pressure sensor only.

Note: The Barometer sensor is used for Altimeter display on the Home screen. The Variometer sensor (Pr Al on the LCD) is normally used only by the Variometer.

Pressing D sets all calibration constants to their default values.

ALTIMETER and VARIO CALIBRATION.
This mode is used to calibrate the Barometer and Variometer absolute pressure sensors. A pressure chamber is required since the Barometer sensor has no external connection.

The calibration has two steps:

Set altitude to low set point (in user units) and press <ENTER>

Set altitude to high set point (in user units) and press <ENTER>

Note: Set points can be changed by typing in the desired value.

At this point the offset and gain factors for each sensor will be displayed. Pressing <ENTER> again will return the Calibration Port Menu.

GMETER CALIBRATION

This calibration mode is used to calibrate the 302 accelerometers. The earth's gravitational field is used as the standard.

The calibration has 4 steps, 2 for each axis:

Set Instrument upright on a level surface and press <ENTER>
Set Instrument upside down on a level surface and press <ENTER>
Set Instrument with LCD facing straight up and press <ENTER>
Set Instrument with LCD facing straight down and press <ENTER>

At this point the offset and gain factors for each sensor will be displayed. Pressing <ENTER> again will return the Calibration Port Menu.

AIRSPEED CALIBRATION

This calibration mode is used to calibrate the Dynamic pressure (airspeed) sensor. An airspeed indicator or pressure gage can be used.

The calibration has 2 steps:
Set airspeed pressure to low set point (in user units) and press <ENTER>
Set airspeed pressure to high set point (in user units) and press <ENTER>

Note: Set points can be changed by typing in the desired value.

At this point the offset and gain factors for the sensor will be displayed. Pressing <ENTER> again will return the Calibration Port Menu.

TEMPERATURE COMPENSATION OF AIRSPEED

This calibration sets the compensation for zero drift with temperature of the differential pressure sensor used to measure airspeed.

When this mode is entered an automated data collection process begins. Data is collected for up to 5 hours. The environment for this procedure should be one where there is little air movement. Any external perturbations of the differential pressure sensor will skew the results.

The recommended procedure is as follows.

- Turn unit on for 1 half hour.
- Start TEMPERATURE COMPENSATION OF AIRSPEED by pressing T at the calibration port.
- Let Instrument stand for 1 hour.
- Heat unit up to 70 degrees C
- Let Instrument stand for 1 hour.
- Complete TEMPERATURE COMPENSATION OF AIRSPEED by pressing <ENTER> and <Y>

At this point information relating to the calibration will be displayed. Pressing <ENTER> again will return the Calibration Port Menu.

VARIO CALIBRATION:

This calibration is the same as ALTIMETER and VARIO CALIBRATION except only the Variometer absolute pressure sensor is affected.

X. POCKET NAV MODE

Pocket Nav mode is entered by entering PNP<ENTER> at the command port. In some versions, the Pocket Nav port will only be entered if a certain protocol was completed beforehand. As of version F17000 the protocol is not necessary. The protocol is not documented here.

In Pocket Nav mode the Port acts as if it is in LOG 0 mode. The difference is that commands can be sent to the dataport. Pocket-NAV commands always start with the "!" character (021H). Pocket-NAV commands terminate with a Carriage Return (0DH).

The Following Commands are accepted

!a	Set active point
!b	Set active task
!c	Audio Beep
!g	McCready, Bugs, and Ballast

ACTIVE POINT COMMAND

Format:
```
!aN<CR>
```

N: Active Point Index. Must be less than the number of points stored in the instrument.

ACTIVE TASK COMMAND

Format:
```
!bT,L<CR>
```

T: Active task Index
L: Active task leg

Note: Some versions do not support this command.

AUDIO BEEP COMMAND

Format:
```
!c,N,V,R<CR>
```

N: Beep Type
0: Restricted
1: Approach
2: Arrival
V: Absolute Volume: 0-Loudest 170-Silent 255-Use Relative Volume
R: Relative Volume: Subtracted from current instrument volume setting.

MCCREADY BUGS BALLAST COMMAND

Format:
```
!g,mM,bB,uU<CR>
```

M: MacCready (Knots * 10)
B: Percent Ballast: 0-None 10-Full
U: Bug degradation

Note: Fields are optional

Example !g messages:

```
!g,m24       Sets MacCready to 2.4 M/S
!g,b2,u90    Sets Ballast to 20% and bugs to 90% clean
```

XI. SOFTWARE INTERFACE AND OTHER DEBUGGING

Software Interface:

The SIF command port command provides access to a small subset of the 302 Secure BIOS Software Interface. The 302 Secure BIOS is fully documented in the Secure BIOS User’s Guide. The SIF command provides access to the restart functions of the Software Interface. There are two arguments to the SIF command.

```
SIF 0 0      Restarts the 302.

SIF 0 1      Restarts the 302 in BIOS mode.
```

LCD Screen Testing:

The SCR Command port command provides access to a debugging mode for the LCD screen. Use the command as follows to turn on the LCD debugging output.

```
SCR 32       Turns on the serial debugging output of the LCD processor.
```

Notes:
The serial debugging output of the LCD processor is transmitted about once per second at 38400 bbs from Pin 7 of the Connector U5. The datastream consists of four numbers as follows:

```
      Detector reading with LED On      (0-255) 0 = Max Signal
      Detector reading with LED Off     (0-255) 0 = Max Signal
      Reserved
      Requested Needle Positon.
```

One Wire SRAM Testing:

The functionality of the One-Wire SRAM can be tested using the KEY Command Port Command:

```
Key          Dumps the contents of the “User Page” of One-Wire SRAM.
```

Notes:
32 bytes are displayed in 4 rows of 8 bytes in hexadecimal format.

The User Page stores some of the user settings in the 302. The contents of this page change often. The first two Bytes are always A5H. If these two bytes are different from A5H, there is a malfunction.

XII. NMEA Output

This section describes the NMEA output in Log 0 and Spew 0 modes

300 series Transmitted sentences.

GPRMC

The GPRMC sentence format is:

```
$GPRMC,<1>,<2>,<3>,<4>,<5>,<6>,<7>,<8>,<9>,<10>,<11>,*hh<CR><LF>
```

<1> UTC Time of position fix, hhmmss format

<2> Status, A = Valid position, V = NAV receiver warning

<3> Latitude,ddmm.mmm format (leading zeros will be transmitted)

<4> Latitude hemisphere, N or S

<5> Longitude,dddmm.mmm format (leading zeros will be transmitted)

<6> Longitude hemisphere, E or W

<7> Speed over ground, 0.0 to 999.9 knots

<8> Course over ground 000.0 to 359.9 degrees, true (leading zeros will be transmitted)

<9> UTC date of position fix, ddmmyy format

<10> Magnetic variation, 000.0 to 180.0 degrees (leading zeros will be transmitted)

<11> Magnetic variation direction, E or W (westerly variation adds to true course)

hh Checksum, XOR of all bytes of the sentence after the ‘$’ and before the ‘’

GPRMB

The GPRMB sentence format is:

```
$GPRMB,<1>,<2>,<3>,,<5>,<6>,<7>,<8>,<9>,<10>,<11>,,<13>*hh<CR><LF>
```

<1> Position Valid (A = valid, V = invalid)

<2> Cross Track Error, format XX.X, nautical miles (leading zeroes will be transmitted)

<3> Direction to Steer – L/R

<5> Destination Navpoint identifier, three digits (leading zeros will be transmitted)

<6> Destination Npt. lat. format DDMM.MM (leading zeros will be transmitted)

<7> Destination Npt. lat. hemisphere – N/S

<8> Destination Npt. lon. format DDDMM.MM (leading zeros will be transmitted)

<9> Destination Npt. lat. hemisphere – E/W

<10> Range from present position to distination Navpoint, format XXXX.X, nautical miles (leading zeros will be transmitted)

<11> Bearing from present position to destination Navpoint, format XXX.X, degrees true (leading zeros will be transmitted)

<13> Arrival flag <A = arrival, V = not arrival)

hh Checksum, XOR of all bytes of the sentence after the ‘$’ and before the ‘’

GPGGA

The GPGGA sentence format is:

```
$GPGGA,<1>,<2>,<3>,<4>,<5>,<6>,<7>,<8>,<9>,M,<10>,M,<11>,<12>*hh<CR><LF>
```

```
<1>    UTC time of position fix, hhmmss format
<2>    Latitude, ddmm.mmmm format (leading zeros will be transmitted)
<3>    Latitude hemisphere, N or S
<4>    Longitude, dddmm.mmmm format (leading zeros will be transmitted)
<5>    Longitude hemisphere, E or W
<6>    GPS quality indication, 0 = fix not available, 1 = Non-differential GPS
       fix
       available, 2 = Differential GPS (DGPS) fix available, 6 = Estimated
<7>    Number of satellites in use, 00 to 12 (leading zeros will be transmitted)
<8>    Horizontal dilution of precision, 0.5 to 99.9
<9>    Antenna height above/below mean sea level, -9999.9 to 99999.9 meters
<10>   Geoidal height, -999.9 to 9999.9 meters
<11>   Differential GPS (RTCM SC-104) data age, number of seconds since last
       valid
       RTCM transmission (null if non-DGPS)
<12>   Differential Reference Station ID, 0000 to 1023 (leading zeros will be
       transmitted, null if non-DGPS)PCAIB
*hh    Checksum, XOR of all bytes of the sentence after the ‘$’ and before the
       ‘*’
```

PCAIB

The PCAIB sentence format is:

```
$PCAIB,<1>,<2>,<CR><LF>
```

```
<1>    Destination Navpoint elevation in meters, format XXXXX (leading zeros will
       be transmitted)
<2>    Destination Navpoint attribute word, format XXXXX (leading zeros will be
       transmitted)
```

PCAID

The PCAID sentence format is:

```
$PCAID,<1>,<2>,<3>,<4>*hh<CR><LF>
```

```
<1>    Logged 'L' Last point Logged 'N' Last Point not logged
<2>    Barometer Altitude in meters (Leading zeros will be transmitted)
<3>    Engine Noise Level
<4>    Log Flags
*hh    Checksum, XOR of all bytes of the sentence after the ‘$’ and before the
       ‘!’
```

!w

The !w sentence is a proprietary format sentence that contains air data and instrument settings

The format is:

```
!W,<1>,<2>,<3>,<4>,<5>,<6>,<7>,<8>,<9>,<10>,<11>,<12>,<13>*hh<CR><LF>
<1>    Vector wind direction in degrees
<2>    Vector wind speed in 10ths of meters per second
<3>    Vector wind age in seconds
<4>    Component wind in 10ths of Meters per second + 500 (500 = 0, 495 = 0.5 m/s
       tailwind)
<5>    True altitude in Meters + 1000
<6>    Instrument QNH setting
<7>    True airspeed in 100ths of Meters per second
<8>    Variometer reading in 10ths of knots + 200
<9>    Averager reading in 10ths of knots + 200
<10>   Relative variometer reading in 10ths of knots + 200
<11>   Instrument MacCready setting in 10ths of knots
<12>   Instrument Ballast setting in percent of capacity
<13>   Instrument Bug setting

*hh    Checksum, XOR of all bytes of the sentence after the ‘!’ and before the
       ‘*’
```

Section XIII. Change History

July 20, 2001:
Initial Version

July 23, 2001:
Section VIII. W Message; Added TYPE and VERSION Fields. Corrected an error for the ID Field.
Section XI. Noted that the “Protocol” was no longer necessary as of version F17000

August 2, 2001:
Changed to Rev 1
Reformatting

November 26, 2001
Changed to Rev 2
Added commands for binary format download in B command
Added fields for individual units in O command
Added Y upload and download commands

February 28, 2002; May 2, 2002; July 15, 2002
Changed to Rev 2.1
Added Elevation field to D download message.
Changed o upload and download commands to specify specific pilot records if desired. (Using old utility programs only acts on the active pilot to make the presence of multiple pilots transparent to those programs)
Fixed some errors in the component wind field of the w message

February 4, 2003
Fixed a couple of minor errors in the declaration documentation.
Changed to Rev 2.2

March 25, 2003
Add documentation for active pilot and margin height. Changed !W to !w