for each satellite (up to 32). The health information is provided in three categories. The first one is that obtained from the broadcast Almanac. The second is from the ephemeris that the unit collected for that satellite (note that most satellite's ephemeris are not collected since only those being used for navigation are needed, the corresponding health would be "unavailable"). The last set of health information corresponds to that specified by the user through "Include/Exclude Sats" messages (FMGLU,04).

The "Sats Data" message (FMGLU,03) provides detailed information about a given satellite.

Include/exclude satellites

The "Include/Exclude Sats" message (FMGLU,04) can be used to control satellite selection by the OEM software. With this message the user can control the use of satellites for position determination. The user can specify that certain satellites are NEVER to be user for position computation. Also the user can remove previous restrictions on the use of a satellite by returning it to its natural health. The format of the message would indicate that you can force include satellites. Through v1.3 software this function does not work. It will be functional in v1.4.

Note that the user cannot affect satellites which are not included in the units current almanac. Thus a satellite for which no almanac data is being broadcast can not be unconditionally included.

Almanac/ephemeris data

The "Almanac Data" and "Ephemeris Data" messages (T1 and T2, available in binary only) can be used to either retrieve almanac/ephemeris data from the unit or to load such information into the unit. The primary use of these messages would be to provide the unit with information so that it would not need to collect almanac or ephemeris data from the satellites. This would markedly improve time to first fix, especially in the case of a cold start. Because the ephemeris must be renewed every two hours it's not realistic to load the ephemeris over but it can be done.

Differential corrections

The "Differential Corrections" message (J0, binary only) enables the user to specify differential pseudorange corrections that will be used by the OEM software for position computation. The format and content of this message are based on the proposed RTCM standard. Note that a different message must be sent for each satellite whose ranges are to be corrected. Also the OEM software currently "times out" a correction after more than 200 seconds have expired past its time of validity.

Pseudo-continuous messages

Output of Raw Data, Doppler, C/N0, Almanac Data, and Ephemeris Data has a few special considerations. If a single output of these data messages is requested they will only be output if they are available. In general this is not a problem, but at unit start up, especially from cold-start, certain data may not be available for a time. If continuous output of the data is requested it will only be sent when it is updated. Thus almanac data may only be sent once, ephemeris every so often, etc.

If a single output of a data message is required at start up, when it may not be available, it is recommended that continuous output be initially requested. Then when the message is finally received it can be turned off.

SECTION II - OEM MESSAGES DEFINITIONS

PORT CONFIGURATION

Baud Rate options: 1200, 2400, 4800, 9600 baud.
Data Bits: 8
Parity: None
Stop bits: 1

There are four Eprom sets available with the OEM module. Each set has one of the default baud rates listed above. All of the sets will allow you to change the baud rate of operation to any of the listed above but on cold start it will come up in its default.

FORMATS

Both binary and ASCII formats are provided for most of the sentences. However, some sentences are binary format only, some sentences are ASCII format only.

a. BINARY

A sentence starts with "$$" (two HEX 24) followed by one byte of sentence identifier( from A to Z in ASCII) and one optional byte of sub-index (from 1 to 255 in binary) and followed by binary data field. All the binary data are in integers with defined precision. The sentence is terminated with line feed (HEX 0A). One byte before the line feed is the checksum of the sentence. The checksum is calculated by XOR'ing the 8 binary
data bits of each valid byte in the sentence, between the "$\$" and the checksum. For example:

```
$\$R100000000CL
xxxxxxxxxxxx
('x' denotes data field to be xor'ed, 'C' denotes checksum, 'L' denotes line feed)
```

b. ASCII

The ASCII sentences conform to the NMEA 0183 software protocol (National Marine Electronics Association). Two types of sentences are used. Where the 0183 protocol has an already defined sentence for a certain data this sentence is used. These include GPSSA, GPZDA, GPVTG, GPBOD, GPWKC, GPWDO, GPWDT, GPWPA etc. For information that does not have a predefined sentence we used the method defined in the protocol for designing our own proprietary sentences so they would be compatible with the standard sentences. The proprietary sentences use 'P' for proprietary, 'MGL' for Magellan as manufacturer's identification and a primary index of 'A' through 'Z'. The first data field is a sub-index which has a range of from 00 to 99. The checksum is calculated by XOR'ing the 8 binary data bits of each byte in the sentence, between the '$' and the '"'. If you don't want to calculate the checksum leave out the '"' and two following bytes.

Example with check sum:

```
$SPMGLJ,00,T01,1,B, *CKRL
xxxxxxxxxxxxxxxx
```

('x' denotes data field to be XOR'ed, 'CK' denotes 2 bytes checksum, 'R' denotes carriage return - hex OD, 'L' denotes line feed - hex OA)

Example without check sum:

```
$SPMGLJ,00,T01,1,B, RL
xxxxxxxxxxxxxxxx
```

Example with check sum:

```
$SPMGLJ,00,T01,1,B, *CKRL
xxxxxxxxxxxxxxxx
```

('x' denotes data field to be XOR'ed, 'CK' denotes 2 bytes checksum, 'R' denotes carriage return - hex OD, 'L' denotes line feed - hex OA)

Example without check sum:

```
$SPMGLJ,00,T01,1,B, RL
xxxxxxxxxxxxxxxx
```

---

**OUTPUT MESSAGES**

(Output from the board set)

<table>
<thead>
<tr>
<th>TIME AND DATE</th>
</tr>
</thead>
</table>

**BINARY**

```
$SA0xxxxxxxxCL
123456
```

1: 1 byte, UTC hour
2: 1 byte, UTC minute
3: 1 byte, UTC second
4: 1 byte, day
5: 1 byte, month
6: 2 bytes, year

**ASCII**

```
$GPZDA,xxxxxx,xx,xx,xxxx,*CKRL
```

1: UTC, hhmmms
2: day
3: month
4: year

**POSITION AND ALTITUDE**

**BINARY**

```
$BB0xxxxxxxxxxxxxxxxCL
```

1: 4 bytes, timetag in 1 second
2: 4 bytes, lat in 10^-7 degree
3: 4 bytes, lon in 10^-7 degree
4: 4 bytes, altitude in 0.01 meter/feet

**ASCII**

```
$GPSSA,xxxxxx,xxx.xxx,N,N,xxxxx.xx,W,x,x,xx,xxx,uxx,M,uxxx,M*CKRL
```

1: UTC of position (hhmmss) & time of signal used in calculation of position
2: GPS Latitude
3: Lat. N or S
4: GPS longitude
5: Lon. E or W
6: GPS Quality Indicator (0=GPS not available, 1=GPS available)
7: Number of satellites being used.
8: HDOP - recalc every 3 min last fix
9-10: Antenna Height, Meters/feet
11-12: Geocidal Height, Meters/feet

---

**CONTROL M**: <CR>
**CONTROL J**: LF

DON'T NEED THE SPACES FOR BLANK ARN #. CAN OMIT & JUST GO

Crtl M: <CR>
Crtl J: LF

ECHO ON
NO BUFFER
**POSITION ONLY**

**BINARY**

```
S$B1xxxxxxxC
```

1
2

3: 4 bytes, LAT IN 10^-7, Same as BDP

**ASCII**

```
$GPGLL,xxx.xx,N,xxxx.xx,W*CRKL
```

1 2 3 4

1: last fix latitude
2: lat N or S
3: last fix longitude
4: lon E or W

**POSITION**

**BINARY**

```
S$C0xxxxxxxC
```

1 2 3

1: 4 bytes, ECEF X - coordinate, 0.01 meter
2: 4 bytes, ECEF Y - coordinate, 0.01 meter
3: 4 bytes, ECEF Z - coordinate, 0.01 meter

**ASCII**

```
$PMGLC,00,xxxxxx.xx,xxxxxx.xx,xxxxxx.xx*CRKL
```

1 2 3

1: ECEF X - coordinate, meter
2: ECEF Y - coordinate, meter
3: ECEF Z - coordinate, meter

**MODE**

**BINARY**

```
S$DOxCL
```

1

1: 2=2D mode
2: 3=Automatic/3D mode

**ASCII**

```
$PMGLD,00,x*CRKL
```

1

1: 2=2D mode
2: 3=Automatic/3D mode

---

**GROUND COURSE AND VELOCITY**

**BINARY**

```
S$E0xxxxxCL
```

1

1: heading(true), 0.01 Deg
2: velocity, 0.01 KNOTS, KMPH/MPH

**ASCII**

```
$GPVTG,xxx.xx,T,xxx.xx,M,xxx.x,N,xxx.x,K*CRKL
```

1 2 3 4 5 6 7 8

1-2: Heading, degrees, True
3-4: Heading, degrees, Mag.
5-6: Speed, Knots
7-8: Speed, KMPH/MPH

---

**SATS ID PRN USED**

**BINARY**

```
S$F0xxxxxCL
```

1

1: 1 byte each for 4 Sats PRN number

**ASCII**

```
$PMGLF,00,xx,xx,xx,xx*CRKL
```

1 2 3 4

1-4: Sats PRN number

---

**SATS ID PRN USED DURING ACQUISITION OR EPHEMERIS COLLECT**

**BINARY**

```
S$F1xxxxxxxC
```

121212123

4 sets of field 1 and 2 for 4 sat:

1: 1 byte each for sat PRN number.
2: 1 byte each for the sat status for the sat in field 1.
   0 = yet to be looked for or collected
   1 = now being looked for or collected
   2 = has already been found or collected
3: OBRM receiver status (same as described in H00 message)

**ASCII**

```
$PMGLF,01,xx,xx,xx,xx,xx,xx,xx,xx,xx,xx,xx,xx,xx,xx*CRKL
```

1 2 3
4 sets of field 1 and 2 for 4 sats:

1: 1 byte each for sats PRN number.
2: 1 byte each for the sat status for the sat in field 1.
   0 = yet to be looked for or collected
   1 = now being looked for or collected
   2 = has already been found or collected
3: OEM receiver status (same as described in H00 message)

PDOP, GDOP, ERROR ESTIMATE

BINARY

```
$SOGxxxxxxCL
1 2 3
```

1: 2 bytes, PDOP, 0.01
2: 2 bytes, GDOP, 0.01
3: 2 bytes, Error Estimate, meter

ASCII

```
$PMGLG,00,xxx.xxx,xxx.xxx,xxx*CKRL
1 2 3
```

1: PDOP
2: GDOP
3: Error Estimate, meter

RECEIVER STATUS

BINARY

```
$$H0xxxxxxxxCL
1234567890
```

1: software version number, 0.1
2: customer number
3: battery power for memory back-up (0=ok, 1=low power)
4: Oscillator (0=ok, 1=out of tune)
5: SQ (Signal quality number)
6: GQ (Geometric quality number)
7: nav solution (0=continuous,1=interrupted)
8: Almanac data (0=ok, 1=no almanac data, 2=almanac is old)
9: Memory (0=memory ok to use, 1=lost memory data, need re-init)
10: OEM unit status: 0=INI, 1=IDL, 2=STS, 3=ALM, 4=EPH, 5=ACQ, 6=POS, 7=NAV

ASCII

```
$PMGLH,00,xxx.xxx,xxx.xxx,xxx.xxx,xxx,xxx*CKRL
1 2 3 4 5 6 7 8 9 10
```

Data fields definitions are the same as in binary format

AUTOPILOT

BINARY

```
$$R1xxxxxxxxxxxDL
12345678
```

1: 1 byte, OR'ed Blink and SNR; 1=Valid, 0=Invalid
2: 1 byte, Cycle Lock; 1=Valid, 0=Invalid
3: 2 bytes, Cross track error; 0.01 N. Miles/KM/Stat. Miles
4: 1 byte, 0 = steer right, 1 = steer left
5: 1 byte, Arrival circle entered; 1 = yes, 0 = no
6: 1 byte, Perp. Crossed; 1 = yes, 0 = no
7: 2 bytes, bearing Dest. Wpt. from Origin Wpt. 0.1 degree,
magnetic
8: 4 bytes, Waypoint identifier

ASCII

```
$GPAPA,A,A,A.x.xx,L,N,A,A,Axxx,*,H,cccc*CKRL
1 2 3 4 5 6 7 8 9 10
```

1: OR'ed Blink and SNR; A= Valid, V = Invalid
2: Cycle Lock; A = Valid, V = Invalid
3: -4.5. Cross Track Error, Steer Left (L) or Right(R),
   N.Miles(N)/KM(K)/Stat. Miles(S) units
4: Arrival circle, Arrival Perpendicular (crossing of
   the Dest. Wpt.)
5: A = yes, V = No
8: Departure Wpt. from Origin Wpt., Magnetic

BEARING

BINARY

```
$$R2xxxxxxxxxxxxxxxCL
12345
```

1: 2 bytes, Bearing, true, 0.1 degree
2: 2 bytes, Bearing, Mag., 0.1 degree
3: 2 bytes, distance, 0.1 N. Miles/KM/Stat. Miles

ASCII

```
$GPBOD,xxx..T,xxx..H,cccc,cccc,*CKRL
1 2 3 4 5 6
```

1-2: Bearing., true
3-4: Bearing., Mag.
5: 1 Dest. Wpt. Identifier
6: Origin Wpt. Identifier
BEARING AND DISTANCE

BINARY
$$SR3xxxxxxxxxxxxxxxxxCL
1234 5 6 7 8 9
1: 1 byte, UTC hour
2: 1 byte, UTC minute
3: 1 byte, UTC second
4: 4 bytes, Lat. of Wpt. 10^-7 degree
5: 4 bytes, Lon. of Wpt. 10^-7 degree
6: 2 bytes, Bearing, true, 0.1 degree
7: 2 bytes, Bearing, Mag., 0.1 degree
8: 2 bytes, distance, 0.1 N. Miles/KM/Stat. Miles

ASCII
$GPWPL,xxxx.xx,N,xxxx.xx,W,cccc*CRNL
1 2 3 4 5
1: Lat. degree
2: N or S
3: Lon. degree
4: E or W
5: Wpt. identifier

TIME TO GO

BINARY
$$SR6xxxxxxxxxxxxCL
1234567
1: 1 byte, UTC hour
2: 1 byte, UTC minute
3: 1 byte, UTC second
4: 1 byte, time to go hours
5: 1 byte, time to go minutes
6: 1 byte, time to go seconds
7: 4 bytes, destination wpt. identifier

ASCII
$GPZTG,xxxxxx,xxxxxxx,CCCC*CRNL
1 2 3
1: UTC in Hr. Min and Sec.
2: Time to go to waypoint (hh:mm:ss)
3: Waypoint identifier

ESTIMATED TIME OF ARRIVAL

BINARY
$$SR7xxxxxxxxxxxxCL
1234567
1: 1 byte, UTC hour
2: 1 byte, UTC minute
3: 1 byte, UTC second
4: 1 byte, estimate time of arrivial, hour
5: 1 byte, estimate time of arrivial, minute
6: 1 byte, estimate time of arrival, second
7: 4 bytes, destination wpt. identifier

MAGNETIC VARIATION

BINARY
$$SR4xxCL
1
1: 2 bytes, variation, 0.1 degree

ASCII
$GPBWC,xxxxx,xxxx.xx,N,xxxx.xx,W,
1 2 3 4 5
xxx.x,T,xxx.x,N,xxx.x,N,cccc*CRNL
6 7 8 9 10 11 12
1: UTC of bearing
2-3: Lat., N or S of Wpt.
4-5: Lon. E or W of Wpt.
6-7: Bearing., true
8-9: Bearing., Mag.

WAYPOINTS

BINARY
$$SR5xxxxxxxxxxxxCL
12

ASCII
$GPWHD,xx.x,E*CRNL
$GPHVM,xx.x,E*CRNL
(Derived)
(Manually Set)
1
1: Variation, degrees
2: Variation sense, E or W
**ALMANAC DATA**

**BINARY**

$$S$lxxxx.........xxxxx$C$L

1:  (924 bytes in data field)

1:  924 bytes almanac message. Formatted as follows:

- bytes 1-24: almanac for SV 1
- bytes 25-26: week number for SV 1's almanac
- bytes 27-28: almanac for SV 2
- bytes 29-30: week number for SV 2's almanac
- ... (similar for other SVs)

Each page of data consists of 24 bytes representing words 3 through 10 of the transmitted data with parity removed. The two exceptions are page 25 of subframe 5 which contains only words 3 through 9 of the data and page 17 of subframe 4 which does not include the least significant byte of word 10. Bytes are transmitted in the same order as from the satellite (most significant byte of word 3 is transmitted first for each page).

**EPHEMERIS DATA**

**BINARY**

$$S$t2xxxxx.........xxxxx$C$L

12:  (61 bytes in data field 2)

1:  PRN number
2:  61 bytes ephemeris message. Formatted as follows:

- bytes 1-3: word 3 of subframe 1
- byte 4: 1lsb of word 7 of subframe 1
- bytes 5-7: word 8 of subframe 1
- bytes 8-10: word 9 of subframe 1

Each word contains the 24 data bits of the transmitted data (parity removed). The most significant byte of each word is transmitted first and the least significant byte is transmitted last.

**SATS SCHEDULE**

**BINARY**

$$S$u1xxxxxxxxxxx.........xx$C$L

1:  (396 bytes in data field)

1:  4 bytes, lat. in 10^-7 deg.
2:  4 bytes, lon. in 10^-7 deg.
3:  1 byte, day
4:  1 byte, month
5:  2 bytes, year
6:  96 bytes, 24 hours sat schedule with 15 minutes window

**ASCII**

$$F$mlu,01,A,xxxx.xx,N,xxxx.xx,W,xx,xx,xxxx*CR$L

1:  4 bytes, sub-index of sat data module
2:  4 bytes, sequence number of this message
3:  1 byte, Lat.
4:  1 byte, Lat. N or S
5:  1 byte, Lon.
6:  1 byte, Lon. E or W
7:  1 byte, day
8:  1 byte, month
9:  1 byte, year
10:  First 12 hours of sat's schedule with 15 minutes window showing the number of available sats.
11:  Second 12 hours of sat's schedule

**SATS HEALTH**

**BINARY**

$$S$u2xxxxxxxxxxx$C$L

1:  (6 bytes)

1:  8 bytes, Almanac health status
2:  2 bits for each sat, 01=healthy, 00=unhealthy
2: 8 bytes, Ephemeris health status,
01=healthy, 00=not available, 11=unhealthy, 00=default

3: 8 bytes, User's health status, 01=healthy, 11=unhealthy,
00=default

ASCII
$PMGLU,02,A,xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx*CKRL
$PMGLU,02,B,xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx*CKRL
$PMGLU,02,C,xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx*CKRL

1 2 3

1: sub-index of sats data module
2: sequence number of this message, A=almanac, B=ephemeris,
C=user
3: 1 character each for 32 sats health status, character - =
(hex 2D)
represents '-1' as defined in binary format

SATS DATA

BINARY
$SUxxxxxxxxxxCL
1234

1: 1 byte, PRN number of sats
2: 2 bytes, azimuth, 1 degree
3: 2 bytes, C/N0, in db-Hz,
   if the specified satellite is not tracked, the value will
   be 0.
4: 2 bytes, elevation angle, 1 degree

ASCII
$PMGLU,03,xx,xxx,xxx,xxx*CKRL
1 2 3 4 5

1: sub-index of sats data module
2: PRN number of sats
3: azimuth, degree
4: C/N0, in db-Hz
5: elevation angle, degree

RAW DATA, DOPPLER, C/N0

BINARY
$eVxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxCL
1 234 1 234 1 234 5

4 sets of the data fields 1-4:
1: 4 bytes, doppler, in 0.01 cycles/sec
2: 1 byte, C/N0, in db-Hz

3: 1 byte, PRN
4: 4 bytes, time of transmission, in nsec.
5: 4 bytes, seconds into week, in 0.001 second

DATUM AND UNITS SETUP

BINARY
$SIxxxxxCL
123456

data fields are the same as described in ASCII format.

ASCII
$PMGLS,01,x,xx,x,x,x,x*CKRL
1 2 3 4 5 6

1: sub-index of setup module
2: Terrain setting
   0 = clear
   1 = interrupted
   2 = obstructed
3: Map datum 1-47 choices of map datum
4: Distance/Speed Units
   0 = N. Miles, KNOTS
   1 = KM, KM/HR
   2 = Stat. Miles, MPH
5: Altitude Units
   0 = Feet
   1 = Meter
6: Magnetic Variation
   0 = User entered
   1 = Auto derived

INPUT MESSAGES
(Into The Module)

TIME AND DATE

Same as output

POSITION AND ALTITUDE

Same as output except the only valid fields are L/L and
altitude.

MODE

Same as output
DIFFERENTIAL CORRECTIONS

BINARY contents

This message format will conform to the RTCM message type 1 format.

```
$J0bbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbCL
1 2 3 4 5 6
```

("b" denotes 1 bit)

1: 3 bits, sets health summary and UDRE
2: 5 bits, sets ID
3: 16 bits, pseudorange correction, 0.02 - 0.64 meters
4: 8 bits, range rate correction, 0.004 - 0.128 m/s
5: 8 bits, issue of data
6: 16 bits, time of validity, 0-3599.4 sec, LSB = 0.075 sec, nominally bottom 3 bits are zeros as per RTCM.

MESSAGE ON/OFF, BINARY/ASCII SELECTION

ASCII

```
$PMGL,00,xxx,x,x,xx*CKRL
12 3 4 5
```

1-2:

1: 'A-Z' primary index for messages
2: '0-99' sub-index for messages

1&2: '100' indicates all messages.

3:

0 = turns off output. If field 1 = 'A-Z' and field 2 = '00-99' then the individual sentence is turned off. If field 1&2 = '100' then all output is turned off and the previous configuration is retained, but a field 3 command of 1 with a field 1&2 command of '100' must be issued before any output can be obtained or the configuration can be altered again.

1 = Provided a field 3 command of 'O' with field 1&2 = '100' was not the last command issued, if field 1 = 'A-Z' and field 2 = '00-99' then this command will produce a one-time output for the selected sentence. If this particular sentence was turned on as continuous, it will be subsequently shut off. If the last command issued was field 3 command 'O' with a field 1&2 = '100' then this command will turn on the configuration retained else it will be ignored.

2 = turns on data for the specified sentence in field 1&2 with continuous output and a default interval setting (the current default interval is 1 second).

If field 1&2 = '100' then all sentences currently selected for output will be changed to the binary/ASCII mode indicated in field 4. This is also true for a field 3 value of 3 through 6 (described below).

3 and greater = on with continuous output with interval choices:
3 = 5 seconds
4 = 10 seconds
5 = 20 seconds
6 = 60 seconds

4: A = ASCII format
   B = Binary format

5: PRN number if sat data or ephemeris data message is specified.

Note: There are no ASCII format messages for Almanac, Ephemeris and raw data messages. If continuous output selection is made for Almanac, Ephemeris or raw data messages, they will be output only when a new data set is collected instead of outputting at a fixed interval rate.

COLLECT ALMANAC

BINARY

```
$SK0xxCL
12
```

1: 1 byte, 1 = collect almanac, 0 = don't collect almanac
2: 1 byte, Sat PRN number to collect almanac from

ASCII

```
$PMGLR,00,xxx*CKRL
1 2
```

1: 1 = collect almanac, 0 = don't collect almanac
2: Sat PRN number to collect almanac from

FORCE SATELLITE SELECTION

BINARY

```
$SM0xxCL
1
```

1: 1 byte, 1 = force satellite selection
   0 = don't force satellite selection
TOTAL NUMBER OF WAYPOINTS

BINARY
$$R9\times CL$
1

1: 1 byte, total number of waypoints defined.

ASCII
$PMGR+09,xxx*CKRL$
1 2

1: Sub-index of waypoint module
2: Total number of waypoints defined.

CLEAR/RENAME THE WAYPOINT

BINARY
$$RAccccxxxxCL$
12 3

1: 1 byte, Sub-index of waypoint module (hex 0a = dec 10)
2: 4 bytes, waypoint to be cleared or renamed.
3: 4 bytes, clear waypoint if "0000", rename if ascii.

ASCII
$PMGR+10,cccc,xxxx*CKRL$
1 2 3

1: 1 byte, Sub-index of waypoint module.
2: 4 bytes, waypoint to be cleared or renamed.
3: 4 bytes, clear waypoint if "0000", rename if ascii.

ROUTES

BINARY
$$RBxxxxxxxxCL$
12 3

1: 1 byte, route number
2: 4 bytes, from wpt. identifier
3: 4 bytes, to wpt. identifier

ASCII
$PMGR+08,xx,cccc,cccc*CKRL$
1 2 3 4

1: sub-index of waypoint module
2: route number
3: from wpt. identifier
4: to wpt. identifier
MAGNETIC VARIATION

same as output (HVM only)

DATUM AND UNITS SETUP

BINARY

$S$lxxxxxCL

123456

data fields are the same as described in ASCII format.

ASCII

$PMGLS,01,xx,xx,xx,xx,xx,xx,xx,xx,xx,xx,xx*CR

1 2 3 4 5 6

1: sub-index of setup module
2: Terrain setting
   0= clear
   1= interrupted
   2= obstructed
3: Map datum 1-47 choices of map datum
4: Distance/Speed Units
   0= N. Miles, KNOTS
   1= KM, KM/HR
   2= Stat. Miles, MPH
5: Altitude Units
   0= Feet
   1= Meter
6: Magnetic Variation
   0= User entered
   1= Auto derived

ALMANAC DATA

BINARY

same as output

EPHEMERIS DATA

BINARY

same as output

INCLUDE/EXCLUDE SATS

BINARY

$SU4xxxxCL

123

data fields are the same as described in ASCII format.

ASCII

$PMGLU,04,xx,xx,xx,xx,xx,xx,xx,xx,xx,xx,xx,xx*CR

1 2 3

1: sub-index of Sats Data Module
2: Sats PRN number
3: include/exclude sats
   0= exclude
   1= include (not functional through v1.3)
   2= return sats to its natural health
OEM SOFTWARE MESSAGES

This is a quick reference to locate the different data sentence by ASCII, by Binary, or by data content. The Binary reference used here is the same as you would use in the PMGLI sentence reference to the ASCII or Binary data sentence in the protocol. (B00 as opposed to B0.) The I/O indicates whether the sentence is both Input/Output, just Input, or just Output. Page is the location in this document.

SORTED BY ASCII CODE

<table>
<thead>
<tr>
<th>ASCII</th>
<th>Binary</th>
<th>Function</th>
<th>I/O</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>GPDPA</td>
<td>R01</td>
<td>Differential Corrections</td>
<td>I</td>
<td>18</td>
</tr>
<tr>
<td>GPBDA</td>
<td>A00</td>
<td>Time and Date</td>
<td>I/O</td>
<td>7</td>
</tr>
<tr>
<td>GPAGA</td>
<td>B00</td>
<td>Position, Altitude etc.</td>
<td>I/O</td>
<td>7</td>
</tr>
<tr>
<td>GPGLL</td>
<td>B01</td>
<td>Position (Last Fix)</td>
<td>0</td>
<td>8</td>
</tr>
<tr>
<td>PMGLC00</td>
<td>C00</td>
<td>ECEF Position</td>
<td>0</td>
<td>8</td>
</tr>
<tr>
<td>PMGLD00</td>
<td>D00</td>
<td>Mode</td>
<td>I/O</td>
<td>8</td>
</tr>
<tr>
<td>GPVTG</td>
<td>E00</td>
<td>Ground Course and Velocity</td>
<td>0</td>
<td>9</td>
</tr>
<tr>
<td>PMGLF00</td>
<td>F00</td>
<td>SatS Used</td>
<td>0</td>
<td>9</td>
</tr>
<tr>
<td>PMGLF01</td>
<td>F01</td>
<td>SatS Used Including Status</td>
<td>0</td>
<td>9</td>
</tr>
<tr>
<td>PMGLG00</td>
<td>G00</td>
<td>PDOP, GDOP, Error Estimate</td>
<td>0</td>
<td>10</td>
</tr>
<tr>
<td>PMGLH00</td>
<td>H00</td>
<td>Receiver Status</td>
<td>0</td>
<td>10</td>
</tr>
<tr>
<td>PMGLI00</td>
<td>I00</td>
<td>Message On/Off, Binary/ASCII</td>
<td>I</td>
<td>18</td>
</tr>
<tr>
<td>PMGLK00</td>
<td>K00</td>
<td>Collect Almanac</td>
<td>I</td>
<td>18</td>
</tr>
<tr>
<td>PMGLM00</td>
<td>M00</td>
<td>Force Satellite Reselection</td>
<td>I</td>
<td>20</td>
</tr>
<tr>
<td>PMGLN00</td>
<td>N00</td>
<td>Master Reset</td>
<td>I</td>
<td>20</td>
</tr>
<tr>
<td>PMGLQ00</td>
<td>O00</td>
<td>Change Baud Rate</td>
<td>I</td>
<td>20</td>
</tr>
<tr>
<td>GPAP0</td>
<td>R01</td>
<td>Autopilot</td>
<td>0</td>
<td>11</td>
</tr>
<tr>
<td>GPBD0</td>
<td>R02</td>
<td>Bearing (Dest and Orig Wypts)</td>
<td>0</td>
<td>11</td>
</tr>
<tr>
<td>GPBN0</td>
<td>R03</td>
<td>Bearing and Distance to Waypt</td>
<td>0</td>
<td>12</td>
</tr>
<tr>
<td>GPVTD</td>
<td>R04</td>
<td>Magnetic Variation (derived)</td>
<td>0</td>
<td>12</td>
</tr>
<tr>
<td>GPVH0</td>
<td>R04</td>
<td>Magnetic Variation (derived)</td>
<td>I/O</td>
<td>12</td>
</tr>
<tr>
<td>GPWPL</td>
<td>R05</td>
<td>Waypoints</td>
<td>I/O</td>
<td>12</td>
</tr>
<tr>
<td>GPZT0</td>
<td>R06</td>
<td>Time To Go to Waypt</td>
<td>0</td>
<td>13</td>
</tr>
<tr>
<td>GPZTA</td>
<td>R07</td>
<td>ETA to Waypt</td>
<td>0</td>
<td>13</td>
</tr>
<tr>
<td>PMGLR00</td>
<td>R08</td>
<td>Routes</td>
<td>I</td>
<td>21</td>
</tr>
<tr>
<td>PMGLR09</td>
<td>R09</td>
<td>Number of Waypoints</td>
<td>I</td>
<td>21</td>
</tr>
<tr>
<td>PMGLR10</td>
<td>R10</td>
<td>Clear/Remove Waypt</td>
<td>I</td>
<td>21</td>
</tr>
<tr>
<td>PMGLS01</td>
<td>S01</td>
<td>Datum, Terrain, and Units Setup</td>
<td>I/O</td>
<td>17</td>
</tr>
<tr>
<td>PMGLU01</td>
<td>U01</td>
<td>Sat Schedule</td>
<td>0</td>
<td>15</td>
</tr>
<tr>
<td>PMGLU02</td>
<td>U02</td>
<td>Sat Health (Alm, Eph, User)</td>
<td>0</td>
<td>15</td>
</tr>
<tr>
<td>PMGLU03</td>
<td>U03</td>
<td>Sat Status</td>
<td>0</td>
<td>15</td>
</tr>
<tr>
<td>PMGLU04</td>
<td>U04</td>
<td>Include/Exclude Sats</td>
<td>I</td>
<td>22</td>
</tr>
<tr>
<td>V01</td>
<td></td>
<td>Doppler, C/No</td>
<td>0</td>
<td>16</td>
</tr>
<tr>
<td>ASCII</td>
<td>BINARY</td>
<td>FUNCTION</td>
<td>I/O</td>
<td>PAGE</td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
<td>-----------------------------------------------</td>
<td>-----</td>
<td>------</td>
</tr>
<tr>
<td>PMGHLH00</td>
<td>H00</td>
<td>Almanac Data OK?</td>
<td>O</td>
<td>10</td>
</tr>
<tr>
<td>PMGLH00</td>
<td>T01</td>
<td>Almanac Data</td>
<td>O/I</td>
<td>14</td>
</tr>
<tr>
<td>PMGLS01</td>
<td>S01</td>
<td>Almanac Data</td>
<td>I/O</td>
<td>17</td>
</tr>
<tr>
<td>GPGGA</td>
<td>B00</td>
<td>Antenna Height</td>
<td>I/O</td>
<td>7</td>
</tr>
<tr>
<td>GPADA</td>
<td>R01</td>
<td>Autopilot</td>
<td>O</td>
<td>11</td>
</tr>
<tr>
<td>PMGLU03</td>
<td>U03</td>
<td>Azimuth of Sat</td>
<td>O</td>
<td>15</td>
</tr>
<tr>
<td>PMGLU00</td>
<td>H00</td>
<td>Continuous Fix</td>
<td>I</td>
<td>10</td>
</tr>
<tr>
<td>PMGLU00</td>
<td>G00</td>
<td>Baud Rate, Change</td>
<td>I</td>
<td>20</td>
</tr>
<tr>
<td>GPBOD</td>
<td>R02</td>
<td>Bearing (Wpt to Wpt)</td>
<td>O</td>
<td>11</td>
</tr>
<tr>
<td>GPBWC</td>
<td>R03</td>
<td>Bearing (to Wpt Dest)</td>
<td>I</td>
<td>12</td>
</tr>
<tr>
<td>PMGL100</td>
<td>I00</td>
<td>Binary/ASCII</td>
<td>I</td>
<td>18</td>
</tr>
<tr>
<td>PMGL000</td>
<td>O00</td>
<td>Change Baud Rate</td>
<td>I</td>
<td>20</td>
</tr>
<tr>
<td>PMGLR10</td>
<td>R10</td>
<td>Clear Waypoint</td>
<td>I</td>
<td>21</td>
</tr>
<tr>
<td>PMGLN00</td>
<td>N00</td>
<td>Clear Memory, Reset</td>
<td>I</td>
<td>20</td>
</tr>
<tr>
<td>PMGLX00</td>
<td>K00</td>
<td>Collect Almanac</td>
<td>I</td>
<td>19</td>
</tr>
<tr>
<td>PMGLU03</td>
<td>U03</td>
<td>C/No</td>
<td>O</td>
<td>15</td>
</tr>
<tr>
<td>PMGLU03</td>
<td>V01</td>
<td>Date</td>
<td>I/O</td>
<td>16</td>
</tr>
<tr>
<td>GPEMA</td>
<td>A00</td>
<td>Date</td>
<td>I/O</td>
<td>7</td>
</tr>
<tr>
<td>PMGLS01</td>
<td>S01</td>
<td>Date Select</td>
<td>I/O</td>
<td>17</td>
</tr>
<tr>
<td>GPBOD</td>
<td>R02</td>
<td>Destination Waypoint</td>
<td>O</td>
<td>11</td>
</tr>
<tr>
<td>GPBOD</td>
<td>J00</td>
<td>Differential Corrections</td>
<td>I</td>
<td>18</td>
</tr>
<tr>
<td>GPBWC</td>
<td>R03</td>
<td>Distance to Waypoint</td>
<td>I/O</td>
<td>12</td>
</tr>
<tr>
<td>PMGLS01</td>
<td>S01</td>
<td>Distance Units</td>
<td>I/O</td>
<td>17</td>
</tr>
<tr>
<td>PMGLC00</td>
<td>C00</td>
<td>Doppler</td>
<td>O</td>
<td>16</td>
</tr>
<tr>
<td>PMGLU03</td>
<td>U03</td>
<td>ECEF Position</td>
<td>O</td>
<td>8</td>
</tr>
<tr>
<td>PMGLU00</td>
<td>T02</td>
<td>Elevation of Sat</td>
<td>O</td>
<td>15</td>
</tr>
<tr>
<td>PMGLG00</td>
<td>G00</td>
<td>Ephemeris Data</td>
<td>I/O</td>
<td>14</td>
</tr>
<tr>
<td>GPETA</td>
<td>R07</td>
<td>Error Estimate</td>
<td>O</td>
<td>10</td>
</tr>
<tr>
<td>PMGLU04</td>
<td>U04</td>
<td>ETA to Waypoint</td>
<td>O</td>
<td>13</td>
</tr>
<tr>
<td>PMGLM00</td>
<td>M00</td>
<td>Exclude Sats</td>
<td>I</td>
<td>22</td>
</tr>
<tr>
<td>PMGLS01</td>
<td>I00</td>
<td>Force Satellite Reselection</td>
<td>I</td>
<td>20</td>
</tr>
<tr>
<td>PMGLH00</td>
<td>G00</td>
<td>Frequency of Message</td>
<td>I</td>
<td>18</td>
</tr>
<tr>
<td>PMGLG00</td>
<td>G00</td>
<td>GDOP</td>
<td>O</td>
<td>10</td>
</tr>
<tr>
<td>GPGGA</td>
<td>B00</td>
<td>Geodical Height</td>
<td>I/O</td>
<td>7</td>
</tr>
<tr>
<td>GPGGA</td>
<td>B00</td>
<td>GPS Quality</td>
<td>I/O</td>
<td>7</td>
</tr>
<tr>
<td>PMGLH00</td>
<td>H00</td>
<td>GQ</td>
<td>O</td>
<td>10</td>
</tr>
<tr>
<td>GPVSG</td>
<td>E00</td>
<td>Ground Course</td>
<td>O</td>
<td>9</td>
</tr>
<tr>
<td>GPGGA</td>
<td>B00</td>
<td>HDOP</td>
<td>I/O</td>
<td>7</td>
</tr>
<tr>
<td>GPVSG</td>
<td>E00</td>
<td>Heading</td>
<td>O</td>
<td>9</td>
</tr>
<tr>
<td>PMGLU02</td>
<td>U02</td>
<td>Health (Alm, Eph, User)</td>
<td>O</td>
<td>15</td>
</tr>
<tr>
<td>PMGLU04</td>
<td>U04</td>
<td>Include Sats</td>
<td>I</td>
<td>22</td>
</tr>
<tr>
<td>GPGLL</td>
<td>B01</td>
<td>Last Fix</td>
<td>O</td>
<td>8</td>
</tr>
<tr>
<td>GPBWC</td>
<td>R03</td>
<td>Lat of Waypoint</td>
<td>O</td>
<td>12</td>
</tr>
<tr>
<td>GPBWC</td>
<td>R03</td>
<td>Lon of Waypoint</td>
<td>O</td>
<td>12</td>
</tr>
<tr>
<td>PMGLS01</td>
<td>S01</td>
<td>Magnetic Variation (User vs Auto)</td>
<td>I/O</td>
<td>17</td>
</tr>
<tr>
<td>.GPHVD</td>
<td>R04</td>
<td>Magnetic Variation (Derived)</td>
<td>O</td>
<td>12</td>
</tr>
<tr>
<td>.GPHVM</td>
<td>R04</td>
<td>Magnetic Variation (Manual)</td>
<td>I/O</td>
<td>12</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ASCII</th>
<th>BINARY</th>
<th>FUNCTION</th>
<th>I/O</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>PMGLS01</td>
<td>S01</td>
<td>Map Datum</td>
<td>I/O</td>
<td>17</td>
</tr>
<tr>
<td>PMGLH00</td>
<td>H00</td>
<td>Memory</td>
<td>O</td>
<td>10</td>
</tr>
<tr>
<td>PMGLI00</td>
<td>I00</td>
<td>Message Frequency</td>
<td>I</td>
<td>18</td>
</tr>
<tr>
<td>PMGLL00</td>
<td>I00</td>
<td>Message On/Off</td>
<td>I</td>
<td>18</td>
</tr>
<tr>
<td>PMGLD00</td>
<td>D00</td>
<td>Mode</td>
<td>I/O</td>
<td>8</td>
</tr>
<tr>
<td>PMGLH00</td>
<td>H00</td>
<td>Nav Solution Quality</td>
<td>O</td>
<td>10</td>
</tr>
<tr>
<td>GPGGA</td>
<td>B00</td>
<td>Number of Sats Used</td>
<td>I/O</td>
<td>7</td>
</tr>
<tr>
<td>PMGLR09</td>
<td>R09</td>
<td>Number of Waypoints</td>
<td>I</td>
<td>21</td>
</tr>
<tr>
<td>GPBOD</td>
<td>R02</td>
<td>Origin Waypoint</td>
<td>O</td>
<td>11</td>
</tr>
<tr>
<td>PMGLG00</td>
<td>G00</td>
<td>PDOP</td>
<td>O</td>
<td>10</td>
</tr>
<tr>
<td>GPGGA</td>
<td>B00</td>
<td>Position</td>
<td>I/O</td>
<td>7</td>
</tr>
<tr>
<td>GPGLL</td>
<td>B01</td>
<td>Position (Last Fix)</td>
<td>O</td>
<td>8</td>
</tr>
<tr>
<td>PMGLR10</td>
<td>R10</td>
<td>Rename Waypoint</td>
<td>I</td>
<td>21</td>
</tr>
<tr>
<td>PMGLN00</td>
<td>N00</td>
<td>Reset, Clear Memory</td>
<td>I</td>
<td>20</td>
</tr>
<tr>
<td>PMGLR08</td>
<td>R08</td>
<td>Routes</td>
<td>I</td>
<td>21</td>
</tr>
<tr>
<td>PMGLU02</td>
<td>U02</td>
<td>Sat Health (Alm, Eph, User)</td>
<td>O</td>
<td>15</td>
</tr>
<tr>
<td>PMGLU03</td>
<td>U03</td>
<td>Sat Location (Elev, Azimuth, C/No)</td>
<td>O</td>
<td>15</td>
</tr>
<tr>
<td>PMGLU01</td>
<td>U01</td>
<td>Sat Schedule</td>
<td>O</td>
<td>15</td>
</tr>
<tr>
<td>PMGLF01</td>
<td>F01</td>
<td>Sat Status</td>
<td>O</td>
<td>9</td>
</tr>
<tr>
<td>PMGLP01</td>
<td>F00</td>
<td>Sats Used</td>
<td>O</td>
<td>9</td>
</tr>
<tr>
<td>PMGLP01</td>
<td>F01</td>
<td>Sats Used (With Sat Status)</td>
<td>O</td>
<td>9</td>
</tr>
<tr>
<td>PMGLS01</td>
<td>S01</td>
<td>Satellite Reselection, Forced</td>
<td>I</td>
<td>20</td>
</tr>
<tr>
<td>PMGLS01</td>
<td>S01</td>
<td>Speed Units</td>
<td>I/O</td>
<td>17</td>
</tr>
<tr>
<td>PMGLH00</td>
<td>H00</td>
<td>SQ</td>
<td>O</td>
<td>10</td>
</tr>
<tr>
<td>PMGLS01</td>
<td>S01</td>
<td>Terrain Setting</td>
<td>I/O</td>
<td>17</td>
</tr>
<tr>
<td>GPEDA</td>
<td>A00</td>
<td>Time</td>
<td>I/O</td>
<td>7</td>
</tr>
<tr>
<td>GP2TG</td>
<td>R06</td>
<td>Time To Go to Waypoint</td>
<td>O</td>
<td>13</td>
</tr>
<tr>
<td>PMGLH00</td>
<td>H00</td>
<td>Unit Status</td>
<td>O</td>
<td>10</td>
</tr>
<tr>
<td>PMGLS01</td>
<td>S01</td>
<td>Units</td>
<td>I/O</td>
<td>17</td>
</tr>
<tr>
<td>GPVSG</td>
<td>B00</td>
<td>Velocity</td>
<td>O</td>
<td>9</td>
</tr>
<tr>
<td>GPVSG</td>
<td>R05</td>
<td>Waypoints</td>
<td>I/O</td>
<td>12</td>
</tr>
</tbody>
</table>
Appendix E

Specifications for Data Logger and Related Software
15 CONTROL PORT SERIAL

1. SPECIFICATIONS

FUNCTION
Send/receive half duplex serial data through the CR10 control ports.

INPUT
Data sent by the sensor can be ASCII data values, hexadecimal pairs in ASCII representation, or decimal encoded binary bytes (Section 3, Parameter 2).

OUTPUT
Output can be a preamble set of commands to the sensor or data from the CR10's input locations (Section 3, Parameter 6).

BAUD RATES
1200 baud
300 baud

HANDSHAKE CONTROL LINES
REQUEST TO SEND or DATA TERMINAL READY (RTS/DTR)
The CR10 signals the sensor that it is ready to send or receive data. This line is always asserted at the start of the instruction. If no output is specified then the line acts like DTR and remains asserted until all input is received. If output is specified, the line acts like RTS, asserted during output, not asserted while receiving (Section 3, Parameter 4).

CLEAR-TO-SEND (CTS)
The sensor signals the CR10 that it is ready to receive the preamble or data (Section 3, Parameters 3 and 9).

LOGIC LEVELS
RS232 - Logical 1 is low voltage.
RTS/DTR asserted is high voltage.
TTL - Logical 1 is high voltage.
RTS/DTR asserted is low voltage.

PARITY AND STOP BITS
RECEIVE
- ASCII and Hexadecimal Pairs in ASCII Representation:
  1 start bit, 8 data bits, no parity,
  1 stop bit 8th bit ignored
- Binary to decimal equivalent:
  1 start bit, 8 data bits, no parity,
  1 stop bit

TRANSMIT
1 start bit, 8 data bits, no parity, 1 stop bit

2. SELECTED OPERATING DETAILS
Standard RS232 logic levels range from 3 to 25 volts and -3 to -25 volts. The maximum and minimum input to CR10 control ports is 5 and 0 volts, respectively. Inputs exceeding these limits may damage the CR10. Figure 1 gives a circuit which may be used to limit voltage levels to 0 to 5 volts. Each input (data and handshake) that is out of the 0 to 5 volt range requires one of these circuits, or equivalent.

![Circuit Diagram]

FIGURE 1. Circuit To Limit Input to 0 to 5 Volts

March 19, 1991
COPYRIGHT (c) 1988, 1991 CAMPBELL SCIENTIFIC, INC.
The CR10 consists of a Measurement and Control Module and a detachable Wiring Panel. The Keyboard Display is recommended for on-site communication, station setup, and troubleshooting, but may be replaced by a computer where environmental conditions allow.

**MEASUREMENT AND CONTROL MODULE**

Protected in a sealed, rugged, stainless steel canister, the programmable module provides sensor measurement, timekeeping, communication, data reduction, data/program storage and control functions. A multi-tasking operating system allows simultaneous communication and measurement functions. Operating temperature range is -25° to +50°C, standard; -55° to +85°C, on request.

The standard instruction set includes 30 measurement, 43 processing/math, and 15 program control instructions. Optional instructions are available for specialized measurement or processing capabilities.

The standard memory stores 29,900 data points in two Final Storage areas. Solid-state or SRAM card storage modules provide additional on-site data storage.

The Measurement and Control Module interfaces with the Wiring Panel via two D-style connectors. The CR10's electronics are RF shielded and glitch protected by the sealed, stainless steel packaging. A "watch-dog" hardware reset function restores normal microprocessor function if lost due to an input transient or intermittent component failure.

**WIRING PANEL**

The Wiring Panel consists of a top panel, end bracket, and baseplate. The top panel includes screw terminals for sensor connections and a 9-pin serial I/O port; the end bracket attaches the Wiring Panel to the Control Module and to an enclosure-mounted or free-standing baseplate. The Control Module easily disconnects from the Wiring Panel allowing field replacement without rewiring the sensors. All wiring panel connections are protected with spark gaps or transzors.

**CR10KD KEYBOARD/DISPLAY**

The portable CR10KD programs the CR10, manually initiates data transfer, and displays sensor readings, stored values, or flag/port status. One CR10KD may be carried from station to station in a CR10 network. The CR10KD features an 8-character LCD and a 16-character keyboard. Operating temperature range is -25° to +50°C. The CR10KD is powered by the CR10's power supply.

**SC12 AND SC12R CABLES**

The SC12 ribbon cable (included) or the SC12R, a rugged temperature-resistant cable that is purchased separately, connect peripherals or interfaces to the CR10's serial port.

**CR10TCR THERMCOUPLE REFERENCE**

The CR10TCR thermistor provides a temperature reference for thermocouple measurements. It requires one single-ended analog input. An aluminum cover to reduce temperature gradients along the input terminals is included.

**PERIPHERALS**

The CR10 is powered by a 9.6 to 16 VDC supply and housed in a weather-resistant enclosure (page 8). Measurement, control, and data storage/transfer peripherals are optional depending on the application (pages 4 and 5).

* Differential measurements measure the voltage difference between two inputs. Single-ended measurements measure the inputs with respect to ground. All inputs must be within the ±2.5 V common mode range.
The following electrical specifications are valid for an ambient temperature range of -25°C to +50°C unless otherwise specified.

**PROGRAM EXECUTION RATE**
System tasks initiated in sync with real-time up to 64 Hz. One measurement with data transfer is possible at this rate without interruption. A single input may be measured over short intervals at rates up to 750 Hz using Burst Measurement.

**ANALOG INPUTS**
NUMBER OF CHANNELS: 6 differential or up to 12 single-ended. Each differential channel can be configured as two single-ended channels.
CHANNEL EXPANSION: The AM416 Relay Multiplier allows an additional 64 single-ended channels to multiplex into four CR10 single-ended channels. The AM257 allows an additional 25 differential channels to multiplex into a single CR10 differential channel. Up to three multiplexers can be connected to one CR10.
ACCURACY OF VOLTAGE MEASUREMENTS AND ANALOG OUTPUT VOLTAGES: ±0.1% of FSR, ±0.05% of FSR (0 to 40°C) (e.g., ±0.1% FSR ±5.0 mV for ±2500 mV range)
RANGE AND RESOLUTION: Ranges are software selectable for any channel. Resolution for a single-ended measurement is twice the value shown.

Full Scale Range (mV) Resolution (μV)
+ 2500 333
+ 250 33.3
+ 25 3.33
± 7.5 1.00
± 25 0.33

INPUT SAMPLE RATES: The fast A/D conversion uses a 0.25 ms signal integration time and the slow conversion uses a 2.72 ms signal integration. Two integrations, separated in time by 1/2 of an AC line cycle, are used with the 60 Hz or 50 Hz noise rejection option. Differential measurements include a second sampling with reversed input polarity to reduce thermal offset and common mode errors. Input sample rates are the time required to measure and convert the result to engineering units.
Fast single-ended voltage: 2.6 ms
Fast differential voltage: 4.2 ms
Slow single-ended voltage: 5.1 ms
Slow differential voltage: 9.2 ms
Differential with 60 Hz rejection: 25.9 ms
Fast differential thermocouple: 8.6 ms

INPUT NOISE VOLTAGE:
Fast differential --- 0.82 microvolts RMS
Slow differential --- 0.25 microvolts RMS
Differential with 60 Hz rejection --- 0.18 microvolts RMS

COMMON MODE RANGE: ± 2.5 volts.
DC COMMON MODE REJECTION: > 140 dB.
NORMAL MODE REJECTION: 70 dB (60 Hz with slow differential measurement).
INPUT CURRENT: 3 naneamps maximum.
INPUT RESISTANCE: 200 gigohms.

**EXCITATION OUTPUTS**
DESCRIPTION: The CR10 has 3 switched excitals, active only during measurement, with only one output active at a time. The off state is high impedance.
RANGE: ±2.5 volts.
RESOLUTION: 0.67 millivolts.
ACCURACY: Same as voltage input.
OUTPUT CURRENT: 20 mA @ ±2.5 V; 35 mA @ ± 2.0 V; 50 mA @ ± 1.5 V.

**FREQUENCY SWEEP FUNCTION:** A swept frequency, square wave output between 0 and 25 volts is provided for vibrating wire transducers. Timing and frequency range are specified by the instruction.

**RESISTANCE AND CONDUCTIVITY MEASUREMENTS**
ACCURACY: 0.015% of full scale bridge output, limited by the matching bridge resistors. The excitation voltage should be programmed so the bridge output matches the full scale input voltage range.
MEASUREMENT TYPES: 6-wire and 4-wire full bridge, 4-wire, 3-wire, and 2-wire half bridge. Bridge measurements are ratiometric and dual polarity to eliminate thermal enfl. AC resistance measurements use a dual polarity 0.75 mS excitation pulse for ionic depolarization, with the signal integration occurring over the last 0.25 ms.

**PERIOD AVERAGING MEASUREMENTS**
DEFINITION: The time period for a specified number of cycles of an input frequency is measured, then divided by the number of cycles to obtain the average period of a single cycle.

INPUTS: Any single-ended analog channel; signal dividing or AC coupling is normally required.

**INPUT FREQUENCY RANGE:**

<table>
<thead>
<tr>
<th>Range</th>
<th>Peak to Peak Volts</th>
<th>Maximum Code</th>
<th>Required @ Max. Freq.</th>
<th>Frequency</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2 mV</td>
<td>8 kHz</td>
<td>100 ppms</td>
<td>80 Hz</td>
</tr>
<tr>
<td>2</td>
<td>3 mV</td>
<td>20 kHz</td>
<td>700 ppms</td>
<td>12 kHz</td>
</tr>
<tr>
<td>3</td>
<td>12 mV</td>
<td>50 kHz</td>
<td>3000 ppms</td>
<td>25 kHz</td>
</tr>
<tr>
<td>4</td>
<td>2000 mV</td>
<td>200 kHz</td>
<td>30000 ppms</td>
<td>50 kHz</td>
</tr>
</tbody>
</table>

*AC voltage must be centered around CR10 ground.

REFERENCE ACCURACY: (-25°C to 0°C) ± 80 ppm
(0°C to +50°C) ± 30 ppm

**RESOLUTION:** ± 100 nanoseconds divided by the number of cycles measured. Resolution is reduced by signal noise and for signals with a slow transition through the zero voltage threshold.

**TIME REQUIRED FOR MEASUREMENT:** Signal period divided by the number of cycles measured plus 1.5 cycles.

**PULSE COUNTERS**
NUMBER OF PULSE COUNTER CHANNELS: 2...eight bit or 16 sixteen bit; software selectable.
MAXIMUM COUNT RATE: 2000 Hz, eight bit counter; 250 kHz, sixteen bit counter. Pulse counter channels are scanned at 8 Hz.

MODES: Switch closure, high frequency pulse, and low level AC.

**SWITCH CLOSURE MODE**

**HIGH FREQUENCY PULSE MODE**
Minimum Pulse Width: 0.002 milliseconds. Maximum Input Frequency: 250 kHz. Voltage Thresholds: Count upon transition from below 1.5 V to above 3.5 V. Maximum Input Voltage: ± 20 V.

**LOW LEVEL AC MODE**
(Typical of magnetic pulse flow transducers or other low voltage, sine wave outputs)
Minimum AC Input Voltage: 6 mV RMS. Input Hysteresis: 11 mV. Maximum AC Input Voltage: 20 V RMS.

**DIGITAL I/O PORTS**
8 ports, software selectable as binary inputs or output.

**OUTPUT VOLTAGES (no load):** high 5.0 V ± 0.1V; low < 0.1 V.

**OUTPUT RESISTANCE:** 500 Ω.

**INPUT STATE:** high 3.0 V to 5.5 V; low -0.5 V to 0.8 V.

**INPUT RESISTANCE:** 100 kΩ.

**SDI-12 INTERFACE STANDARD**
This communication protocol, developed for microprocessor-based hydraulic and environmental sensors, is available as a software option in the CR10.

**SENSOR CONNECTIONS:** Digital IO Port #8 (for asynchronous communication), 12V power, and ground. Up to ten SDI-12 sensors can be connected to a CR10.

**CR10TC THERMOCOUPLE REFERENCE**
POLYNOMIAL LINEARIZATION ERROR: Typically <0.05°C over -35 to +50°C range and <0.1°C over -24 to +45°C range.
INTERCHANGEABILITY ERROR: Typically <0.2°C over 0 to +50°C range increasing to ±0.5°C at -40°C.

**TRANSIENT PROTECTION**
All input and output connections to the CR10 module are protected using RC filters or transzors connected to a heavy copper bar between the circuit card and the case. The Wiring Panel includes additional spark gap and transzorb protection.

**CPU AND INTERFACE**
PROCESSOR: Hitachi 6303.
MEMORY: 32K ROM, 64K RAM.
DISPLAY: 8-digit LCD (0.5" digits)

PERIPHERAL INTERFACE: 9 pin D-type connector for keyboard display, storage module, modem, printer, card storage module, and RS-232 adapter. SMD rates selectable at 300, 1200, 9600, and 76800. ASCII communication protocol is one start bit, one stop bit, eight data bits (no parity).

**CLOCK ACCURACY:** ± 1 minute per month.

**SYSTEM POWER REQUIREMENTS**
VOLTAGE: 9.6 to 16 volts.

**TYPICAL CURRENT DRAIN:** 0.7 mA quiescent, 13 mA during processing, and 46 mA during analog measurement.

**BATTERIES:** Any 12 volt battery can be connected as a primary power source. Several power supply options are available from Campbell Scientific.

**PHYSICAL SPECIFICATIONS**
SIZE: 7.8" x 3.5" x 1.5" - Measurement & Control Module; 9" x 3.5" x 2.9" - CR10WP Wiring Panel. Additional room required for connectors.

**WEIGHT:** 2 lbs.

**WARRANTY**
Three years against defects in materials and workmanship.
Program: CR10 W/P15 Interface Code for Magellan GPS
Revision: 06 APR 1994
Filename: c:\magellan\gps.doc
*L Pgm : +21504.
IHO# : 46659

Input Locations Used

Location 1: GMT Time (hh:mm)
  2: GMT Time (sec)
  3: Latitude (ddmm)
  4: Latitude (.mm)
  5: Longitude (ddmm)
  6: Longitude (.mm)
  7: GPS Available (0-no, 1=yes)
  8: # of Satellites used
     (3=2 dimensions (lat & long)
     (4=3 dimensions (lat, long, altitude)
  9: Altitude (feet)
 10: Heading True (0 to 360 degrees)
 11: Heading Magnetic (0 to 360 degrees)
 12: Speed (mph)
 13: CR10 battery voltage (vdc)
 14: CR10 internal temperature [deg F]

50 to 92: GPS Polling and Conversion

Output Data:

All readings are sampled and output to final storage every 15 minutes (user can adjust). At this rate the CR10 can store data continuously for 17.3 days.

* 1 Table 1 Programs
  01: 10 Sec. Execution Interval

-> Load polling sentence $PMGLI,00,B00,1,A, CRLF
   into sequential input locations 50 to 71.

-> Data string: time, lat, long, GPS avail, # sats

  01: P30 Z=F
  01: 36 F
  02: 0 Exponent of 10
  03: 50 Z Loc [: $ ]
02: P30 Z=F
01: 80 F
02: 0 Exponent of 10
03: 51 Z Loc [: P ]

03: P30 Z=F
01: 77 F
02: 0 Exponent of 10
03: 52 Z Loc [: M ]

04: P30 Z=F
01: 71 F
02: 0 Exponent of 10
03: 53 Z Loc [: G ]

05: P30 Z=F
01: 76 F
02: 0 Exponent of 10
03: 54 Z Loc [: L ]

06: P30 Z=F
01: 73 F
02: 0 Exponent of 10
03: 55 Z Loc [: I ]

07: P30 Z=F
01: 44 F
02: 0 Exponent of 10
03: 56 Z Loc [: ]

08: P30 Z=F
01: 48 F
02: 0 Exponent of 10
03: 57 Z Loc [: 0 ]

09: P30 Z=F
01: 48 F
02: 0 Exponent of 10
03: 58 Z Loc [: 0 ]

10: P30 Z=F
01: 44 F
02: 0 Exponent of 10
03: 59 Z Loc [: ]

11: P30 Z=F
01: 66 F
02: 0 Exponent of 10
03: 60 Z Loc [: B -> B ]

12: P30 Z=F
01: 48 F
02: 0 Exponent of 10
03: 61 Z Loc [: 0 ]
13: P30  \text{Z=F}  01: 48 \text{ F}  02: 0 \text{ Exponent of 10}  03: 62 \text{ Z Loc [:0->2->0]}  

14: P30  \text{Z=F}  01: 44 \text{ F}  02: 0 \text{ Exponent of 10}  03: 63 \text{ Z Loc [: 1]}  

15: P30  \text{Z=F}  01: 49 \text{ F}  02: 0 \text{ Exponent of 10}  03: 64 \text{ Z Loc [: 1]}  

16: P30  \text{Z=F}  01: 44 \text{ F}  02: 0 \text{ Exponent of 10}  03: 65 \text{ Z Loc [: 1]}  

17: P30  \text{Z=F}  01: 65 \text{ F}  02: 0 \text{ Exponent of 10}  03: 66 \text{ Z Loc [: A]}  

18: P30  \text{Z=F}  01: 44 \text{ F}  02: 0 \text{ Exponent of 10}  03: 67 \text{ Z Loc [: 1]}  

19: P30  \text{Z=F}  01: 32 \text{ F}  02: 0 \text{ Exponent of 10}  03: 68 \text{ Z Loc [: SPACE]}  

20: P30  \text{Z=F}  01: 32 \text{ F}  02: 0 \text{ Exponent of 10}  03: 69 \text{ Z Loc [: SPACE]}  

21: P30  \text{Z=F}  01: 13 \text{ F}  02: 0 \text{ Exponent of 10}  03: 70 \text{ Z Loc [: CR]}  

22: P30  \text{Z=F}  01: 10 \text{ F}  02: 0 \text{ Exponent of 10}  03: 71 \text{ Z Loc [: LF]}  

- Send polling sentence $PMCLI,00,000,1,A$, CRLF
- Read position fix data string from GPS receiver
23: P15  Port Serial I/O (Special)
   01:  1  Rep
   02:  01 Configuration code
   03:  10 CTS/Delay
   04:  1 First control port
   05:  50 Output Loc $ 
   06:  22 No. of locs to send
   07:  42 Termination character
   08:  56 Maximum characters
   09:  200 CTS/Input wait
   10:  72 Loc [:Raw Time ]
   11:  1 Mult
   12:  0 Offset

-> New polling sentence $PMGLI,00,B02,1,A,  CRLF
-> Changed B00 to B02    -->2<-  
-> Data string returned: Altitude in meters

24: P30  Z=F
   01:  50 F
   02:  0 Exponent of 10
   03:  62 Z Loc [:0-> 2-> 0]

-> Send polling sentence $PMGLI,00,B02,1,A,  CRLF
-> Read altitude data string from GPS receiver

25: P15  Port Serial I/O (Special)
   01:  1  Rep
   02:  01 Configuration code
   03:  10 CTS/Delay
   04:  1 First control port
   05:  50 Output Loc $ 
   06:  22 No. of locs to send
   07:  42 Termination character
   08:  19 Maximum characters
   09:  200 CTS/Input wait
   10:  80 Loc [: 2 ]
   11:  1 Mult
   12:  0 Offset

-> New polling sentence $PMGLI,00,E00,1,A,  CRLF
-> Change E00 to E00     -->E00<-  
-> Data string returned: speed and direction heading

26: P30  Z=F
   01:  69 F
   02:  0 Exponent of 10
   03:  60 Z Loc [: B -> E ]

27: P30  Z=F
   01:  48 F
   02:  0 Exponent of 10
   03:  62 Z Loc [:0-> 2-> 0]

-> Send polling sentence $PMGLI,00,E00,1,A,  CRLF
Read speed and direction data string

20: P15  Port Serial I/O (Special)
   01:  1  Rep
   02:  01  Configuration code
   03: 10  CTS/Delay
   04:  1  First control port
   05: 50  Output Loc $
   06: 22  No. of locs to send
   07: 42  Termination character
   08: 41  Maximum characters
   09: 200  CTS/Input wait
   10:  82  Loc [:Head True]
   11:  1  Mult
   12:  0  Offset

Convert raw GPS readings <-
Edit time data string <-

29: P37  Z*X*F
   01:  72  X Loc Raw Time
   02:  .01  F
   03:  90  Z Loc [:Time Raw ]

30: P45  Z=INT(X)
   01:  90  X Loc Time Raw
   02:  1  Z Loc [:Time hhmm]

31: P44  Z=FRAC(X)
   01:  90  X Loc Time Raw
   02:  90  Z Loc [:Time Raw ]

32: P37  Z*X*F
   01:  90  X Loc Time Raw
   02:  100  F
   03:  90  Z Loc [:Time Raw ]

33: P45  Z=INT(X)
   01:  90  X Loc Time Raw
   02:  2  Z Loc [:Time sec ]

Edit latitude data string <-

34: P45  Z=INT(X)
   01:  73  X Loc Raw Lat
   02:  3  Z Loc [:Lat DDMM ]

35: P44  Z=FRAC(X)
   01:  73  X Loc Raw Lat
   02:  91  Z Loc [:Lat Frac ]

36: P37  Z*X*F
   01:  91  X Loc Lat Frac
   02:  100  F
   03:  91  Z Loc [:Lat Frac ]
37: P45  Z=INT(X)
01: 91  X Loc Lat Frac
02: 91  Z Loc [:Lat Frac ]

38: P37  Z=X*F
01: 91  X Loc Lat Frac
02: .01  F
03: 4  Z Loc [:Lat .MM ]

-> Edit longitude data string <-

39: P45  Z=INT(X)
01: 74  X Loc Raw Long
02: 5  Z Loc [:Lon DDDMM]

40: P44  Z=FRAC(X)
01: 74  X Loc Raw Long
02: 92  Z Loc [:Long Frac]

41: P37  Z=X*F
01: 92  X Loc Long Frac
02: 100  F
03: 92  Z Loc [:Long Frac]

42: P45  Z=INT(X)
01: 92  X Loc Long Frac
02: 92  Z Loc [:Long Frac]

43: P37  Z=X*F
01: 92  X Loc Long Frac
02: .01  F
03: 6  Z Loc [:Long .MM ]

-> Move GPS available to location 7 <-

44: P31  Z=X
01: 75  X Loc GPS Avail
02: 7  Z Loc [:GPS avail]

-> Move # of satellites used to location 8 <-

45: P31  Z=X
01: 76  X Loc # of Sats
02: 8  Z Loc [:# Sats ]

-> Convert altitude to units of feet <-

46: P37  Z=X*F
01: .1  X Loc Alt meter
02: 3.281  F
03: 9  Z Loc [:Alt feet ]

-> Move heading true to location 10 <-
47: P31  Z=X
01: 82  X Loc Head True
02: 10  Z Loc [:Head True]

-> Move heading magnetic to location 11 <-

48: P31  Z=X
01: 83  X Loc Head Mag
02: 11  Z Loc [:Head Mag]

-> Convert speed from km/hr to mph <-

49: P17  Z=X*F
01: 85  X Loc Spd km/hr
02: .62137  F
03: 12  Z Loc [:Speed mph]

-> Measure the CR10's Battery Voltage <-

50: P10  Battery Voltage
01: 13  Loc [:Battery ]

-> Measure the CR10's internal temperature <-

51: P17  Module Temperature
01: 14  Loc [:CR10 Temp]

-> Convert CR10 Temperature to degrees F <-

52: P37  Z=X*F
01: 14  X Loc CR10 Temp
02: 1.8  F
03: 14  Z Loc [:CR10 Temp]

53: P34  Z=X*F
01: 14  X Loc CR10 Temp
02: 32  F
03: 14  Z Loc [:CR10 Temp]

-> Set the Output Flag HIGH (10) once every 15 minutes

54: P92  If time is
01: 0  minutes into a
02: 56  minute interval
03: 10  Set high Flag 0 (output)

-> Time Stamp the data <-

55: P77  Real Time
01: 220  Day,Hour-Minute

-> When the output Flag is set HIGH the output processing statements do their final calculations and the data is sent to the CR10's Final Storage
The Longitude output must be in high resolution. High Resolution is 5 characters xxxxx that will properly represent dddmm for longitude. High resolution requires 4 bytes per data point.

The rest of the output table can be reset to low resolution. Low Resolution is 4 characters xxxx. Low resolution requires 2 bytes per data point and maximizes the data storage.

* 2 Table 2 Programs
  01: 0.0000 Sec. Execution Interval

* 3 Table 3 Subroutines

* A Mode 10 Memory Allocation
  01: 100 Input Locations
  02: 64 Intermediate Locations
  03: 0.0000 Final Storage Area 2

* C Mode 12 Security
  01: 0000 LOCK 1
  02: 0000 LOCK 2
  03: 0000 LOCK 3
<table>
<thead>
<tr>
<th>T</th>
<th>E</th>
<th>L</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>30</td>
<td>Z Loc [:Time hhmm]</td>
</tr>
<tr>
<td>1</td>
<td>33</td>
<td>Z Loc [:Time sec ]</td>
</tr>
<tr>
<td>1</td>
<td>34</td>
<td>Z Loc [:Lat DDMM ]</td>
</tr>
<tr>
<td>1</td>
<td>38</td>
<td>Z Loc [:Lat .MM ]</td>
</tr>
<tr>
<td>1</td>
<td>39</td>
<td>Z Loc [:Lon DDMM]</td>
</tr>
<tr>
<td>1</td>
<td>43</td>
<td>Z Loc [:Long .MM ]</td>
</tr>
<tr>
<td>1</td>
<td>44</td>
<td>Z Loc [:GPS avail]</td>
</tr>
<tr>
<td>1</td>
<td>45</td>
<td>Z Loc [:# Sats ]</td>
</tr>
<tr>
<td>1</td>
<td>46</td>
<td>Z Loc [:Alt feet ]</td>
</tr>
<tr>
<td>1</td>
<td>47</td>
<td>Z Loc [:Head True]</td>
</tr>
<tr>
<td>1</td>
<td>48</td>
<td>Z Loc [:Head Mag ]</td>
</tr>
<tr>
<td>1</td>
<td>49</td>
<td>Z Loc [:Speed mph]</td>
</tr>
<tr>
<td>1</td>
<td>50</td>
<td>Loc [:Battery ]</td>
</tr>
<tr>
<td>1</td>
<td>51</td>
<td>Loc [:CR10 Temp]</td>
</tr>
<tr>
<td>1</td>
<td>52</td>
<td>Z Loc [:CR10 Temp]</td>
</tr>
<tr>
<td>1</td>
<td>53</td>
<td>Z Loc [:CR10 Temp]</td>
</tr>
<tr>
<td>1</td>
<td>54</td>
<td>Z Loc [: $ ]</td>
</tr>
<tr>
<td>1</td>
<td>55</td>
<td>Z Loc [: P ]</td>
</tr>
<tr>
<td>1</td>
<td>56</td>
<td>Z Loc [: M ]</td>
</tr>
<tr>
<td>1</td>
<td>57</td>
<td>Z Loc [: G ]</td>
</tr>
<tr>
<td>1</td>
<td>58</td>
<td>Z Loc [: L ]</td>
</tr>
<tr>
<td>1</td>
<td>59</td>
<td>Z Loc [: I ]</td>
</tr>
<tr>
<td>1</td>
<td>60</td>
<td>Z Loc [: O ]</td>
</tr>
<tr>
<td>1</td>
<td>61</td>
<td>Z Loc [: B -&gt; E ]</td>
</tr>
<tr>
<td>1</td>
<td>62</td>
<td>Z Loc [: B -&gt; E ]</td>
</tr>
<tr>
<td>1</td>
<td>63</td>
<td>Z Loc [: 0 ]</td>
</tr>
<tr>
<td>1</td>
<td>64</td>
<td>Z Loc [: 0-&gt; 2-&gt; 0]</td>
</tr>
<tr>
<td>1</td>
<td>65</td>
<td>Z Loc [: 0-&gt; 2-&gt; 0]</td>
</tr>
<tr>
<td>1</td>
<td>66</td>
<td>Z Loc [: 0-&gt; 2-&gt; 0]</td>
</tr>
<tr>
<td>1</td>
<td>67</td>
<td>Z Loc [: SPACE ]</td>
</tr>
<tr>
<td>1</td>
<td>68</td>
<td>Z Loc [: SPACE ]</td>
</tr>
<tr>
<td>1</td>
<td>69</td>
<td>Z Loc [: CR ]</td>
</tr>
<tr>
<td>1</td>
<td>70</td>
<td>Z Loc [: LF ]</td>
</tr>
<tr>
<td>1</td>
<td>71</td>
<td>Z Loc [: 2 ]</td>
</tr>
<tr>
<td>1</td>
<td>72</td>
<td>Z Loc [: :Raw Time ]</td>
</tr>
<tr>
<td>1</td>
<td>73</td>
<td>Z Loc [: :Raw Time ]</td>
</tr>
<tr>
<td>1</td>
<td>74</td>
<td>Z Loc [: :Raw Time ]</td>
</tr>
<tr>
<td>1</td>
<td>75</td>
<td>Z Loc [: :Raw Time ]</td>
</tr>
<tr>
<td>1</td>
<td>76</td>
<td>Z Loc [: :Lat Frac ]</td>
</tr>
<tr>
<td>1</td>
<td>77</td>
<td>Z Loc [: :Lat Frac ]</td>
</tr>
</tbody>
</table>
Page 10  Input Location Assignments (cont.):

1: 37: 91:  Z Loc [:Lat Frac ]
1: 40: 92:  Z Loc [:Long Frac]
1: 41: 92:  Z Loc [:Long Frac]
1: 42: 92:  Z Loc [:Long Frac]
Appendix F

Radio Direction Finding System Specifications
Beacon Parts List
302 MHz transmitter
AAA alkaline battery
¼ wavelength coaxial dipole

Receiving System Parts List
AOR model AR-8000 hand held receiver
Three element yagi-uda antenna

Beacon Diagram

Operating Characteristics
The transmitter is a 75 MHz crystal controlled, pulse-mode circuit with an output antenna resonant at the third harmonic (302 MHz). The antenna used is called a coaxial dipole and is made out of RG/58 coaxial cable. The radiating end of the antenna has the outer conducting shield removed and is 22 cm in length (a quarter wavelength). The AAA battery powers the circuit for up to two months at maximum power output which is about 0.1 mW. The beacon is housed in a 7 cm long acrylic tube and sealed with rubber stops for environmental protection. The hand held radio and yagi-uda antenna are used to determine the signal bearing.
Appendix G

Electronic System Schematic
Basic Schematic
for Electronics
in Pressure
Vessel

Battery bank using lithium batteries

CR10 Data Logger

Magellan GPS receiver

GPS Antenna outside of the Pressure Vessel

Data Cable between GPS and CR10

Placement of Limiter Circuit

Self-contained RDF Transmitter Circuit, with AA Battery

Clear Acrylic Housing containing circuit and Alkaline Battery

RDF Antenna outside of the Pressure Vessel
Appendix H

Specifications for Syntactic Foam
Theoretically, the concept of introducing reinforced air to a polymer matrix could include any hollow particle. But from the beginning, the material universally used in deepsea applications, for example, has been glass.

Glass is much stronger than any other medium available, and strength is the key requirement for withstanding hydrostatic pressures in subsea buoyancy. Strength remains a key property in non-ocean uses as well, essential to microsphere survivability in aggressive high energy industrial processes.

Although glass spheres and resin are the fundamental ingredients, two other materials are incorporated in 90% of subsea buoyancy – macrospheres and external skins. These are the four ingredients which form the traditional image of syntactic materials.

**Microspheres**
Since 1960 Emerson & Cuming has manufactured microspheres by an exclusive patented process. They are produced for our own internal use in syntactic end-products and are marketed for use in syntactics developed and manufactured by our customers.

Microballoon microspheres range from 6 to 120 microns in median diameter. Most Emerson & Cuming microspheres are made of high strength water glass feedstock. Internal atmosphere is inert.

**Resins**
Glass microspheres can be incorporated successfully into a wide variety of polymers. Traditionally, most subsea buoyancy utilizes an epoxy matrix. Viscosity ranges from 10 to 20,000 centipoises, and color from clear to deep amber. Other polymer types include urethane, polypropylene, thermoplastics and silicone rubber.

**Macrospheres**
Macrospheres are added to the resin-microsphere mix to provide additional density reduction at low cost, and to provide voids for absorbing the heat of the exothermic reaction, allowing large sections to be cured in one piece. Thin-walled, high strength, fiber-reinforced epoxy Microballoon™ macrospheres are manufactured by Emerson & Cuming by a proprietary process. Like microspheres, Emerson & Cuming macrospheres are produced for both internal and external use.

**External skin**
Most subsea buoyancy incorporates a single surface layer of epoxy-saturated fiberglass to aid in resisting hydrostatic pressure and to provide a smooth, impact-resistant surface for finishing, handling, and maintenance.

---

An exponential strength increase
A sphere is the strongest of any possible shapes in the presence of isostatic pressure. All the strength which might otherwise be considered flexural is translated into compressive strength in a sphere. In a theoretical example, a single glass microsphere, although it will fail under a uniaxial force of only 8 grams, will withstand 2,400 psi (16,560 kPa) when subjected to isostatic pressure. Moreover, when the same microsphere is encapsulated in epoxy, the strength of the resulting syntactic structure increases by an estimated six times again, to a total of 14,400 psi (99,360 kPa).
The making of a syntactic

To make a cube of typical subsea buoyancy 2" on the side, the ingredients above are needed, shown here in the exact volumes required. From left, microspheres, macrospheres and epoxy resin.

Microspheres: three perspectives

Loose microspheres through an optical microscope.

Loose microspheres through a scanning electron microscope.

Scanning electron microscope micrograph of an intentionally fractured surface of a syntactic, showing unbroken microspheres still in their cavities.

Syntactic sizes

A sample of subsea buoyancy (upper left) is composed of trillions of microscopic microspheres and a few macrospheres, all immobilized in epoxy resin.
A broad array of syntactics, syntactic ingredients and related structures.

At Canton, Massachusetts, USA, Emerson & Cuming maintains the world’s leading facility for developing and producing three product lines: syntactic materials and high performance microspheres and macrospheres, as well as specialty fiber-resin composites. All products are available to the world market, and all can be produced to standard or custom specifications.

**Syntactics**
Eccofloat® syntactic products for the ocean buoyancy market can be supplied in block form or as custom-molded structures. They can be equipped with attachment points or other hardware, or can be built as complete functional assemblies ready for service. They can be combined with other composites. Pour-in-place, pack-in-place or castable kits can be supplied for customer use on-site, or Emerson & Cuming can inject on site. Syntac® syntactics are supplied in block and castable form for non-ocean applications.

**Microspheres**
Eccosphere® Microballoon® hollow, thin-walled glass microspheres are supplied in several grades. Emerson & Cuming specializes in microspheres with unusually high performance properties to meet critical specifications. They have high strength-to-weight ratios, are chemically pure inside and out, and have high heat resistance. One group, the SDT series, is made in particle sizes up to ten times smaller than ordinary microspheres. A combination of small size and narrow particle size distribution makes this series unique in the industrial world. Emerson & Cuming can apply coupling agents and can custom-design microspheres to meet a wide range of specifications.

**Macrospheres**
Emerson & Cuming’s unique, high strength Eccosphere® Macrobloon™ macrospheres of the type used in ocean syntactics are manufactured in three standard median diameters, 1/4” (6.4 mm), 3/8” (9.5 mm) and 2” (50.8 mm). Macrospheres can be custom-designed to any diameter specified. Compositions include fiberglass/epoxy and carbon/carbon.

**Fiber-resin composites**
Emerson & Cuming is equipped to develop and manufacture large-scale, thick-section fiber-resin composites of high quality and tight dimensions. Cores of syntactics or other materials can be incorporated. Services include design, lay-up, molding by any of several processes, machining and hardware attachment to produce complete functional assemblies.
Syntactic composites

Epoxy/microspheres.
This is the basic syntactic for ocean buoyancy and other high performance uses. Binary system microspheres can be utilized for lower density at similar strengths.

Epoxy/microspheres/macrospheres.
Added to the basic syntactic for lower density and larger segment size. More than one size can be included in a single system.

Epoxy/microspheres/macrospheres/skin.
A protective Ecoloid® skin is a single layer of fiberglass laid in the mold before the pour. It becomes an integral part of the structure.

Epoxy/microspheres/macrospheres/fiber-glass reinforced epoxy.
Two composites joined to form strong structures with the benefits of each.

Hardware attachment
Inserts can be included for attachment of hardware directly to the syntactic material.

Additional syntactic forms

FRP/epoxy/microspheres/FRP.
A stressed-skin, prefabricated sandwich board, a structural concept adaptable to other shapes as well.

Epoxy/microspheres/macrospheres/urethane.
One of several external protective laminates available.

Polypropylene/microspheres.
A flexible syntactic.

Urethane/microspheres.
A flexible syntactic.

Microspheres

Hollow, thin-walled glass Microballoon® microspheres. Available in standard grades from 6 to 70 microns median diameter, and in custom-designed versions. Microspheres appear to the eye as a white, free-flowing powder.

Macrospheres

Hollow, thin-walled fiber-reinforced polymer Macrobloon™ macrospheres. Can be custom-designed to any feasible diameter and composition.

Fiber-resin composites

Complete structures of fiber-reinforced polymer can be fabricated by any of several processes. Samples shown are a submarine buoyancy mounting bracket and a partial cross-section of a submarine mast fairing.

<table>
<thead>
<tr>
<th>Microballoon microsphere grades</th>
<th>Grade</th>
<th>Particle density (g/cc)</th>
<th>Median particle size (µm)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Treated – small, high strength, low density</td>
<td>SDT-28</td>
<td>0.28</td>
<td>16</td>
</tr>
<tr>
<td></td>
<td>SDT-40</td>
<td>0.40</td>
<td>12</td>
</tr>
<tr>
<td></td>
<td>SDT-60</td>
<td>0.60</td>
<td>6</td>
</tr>
<tr>
<td>Treated – high strength, low density</td>
<td>FTD-200</td>
<td>0.20</td>
<td>70</td>
</tr>
<tr>
<td></td>
<td>FTD-202</td>
<td>0.25</td>
<td>55</td>
</tr>
<tr>
<td></td>
<td>FTD-235</td>
<td>0.35</td>
<td>40</td>
</tr>
<tr>
<td>High silica</td>
<td>SI</td>
<td>0.25</td>
<td>55</td>
</tr>
<tr>
<td>Industrial grade – sodium borosilicate glass</td>
<td>IG-101</td>
<td>0.35</td>
<td>55</td>
</tr>
<tr>
<td></td>
<td>IGD-101</td>
<td>0.30</td>
<td>55</td>
</tr>
<tr>
<td></td>
<td>IG-25</td>
<td>0.27</td>
<td>60</td>
</tr>
</tbody>
</table>
As a company, Emerson & Cuming focuses its energies and skills on anticipating and meeting our customers’ needs. This philosophy shapes our dual mission: high quality syntactic structures and high performance microspheres and macrospheres.

As we continue to pioneer in both technologies, experience gained in one reinforces the other. Our work in formulating syntactics every day adds to our experience in microsphere development. Our work with functional microsphere applications in varied systems contributes to our syntactics expertise.

Emerson & Cuming maintains a professional staff of chemists and engineers who work closely with customers at every stage of product design and use. This experienced team draws on 35 years of data to arrive quickly at solutions they can recommend with confidence.

In the development of subsea buoyancy, for example, the customer will specify the amount of lift required and the maximum service depth, and state parameters of dimension, longevity and acceptable buoyancy loss over time. Within that envelope, we customize the syntactic and the assembly.

**Syntactic strength**

For most ocean applications, strength is the first consideration, then density. In general, small microspheres produce greater strength than large ones. The development of maximum strength depends partly on the interplay of sphere and epoxy. A bond forms at the interface, enhanced by the excellent adhesive properties of epoxy.

Possibly an equally important factor, not generally recognized, is the ability of a glass sphere to deform. This property brings the effective modulus of the glass much closer to that of the matrix. As the matrix experiences pressure, the microsphere will deform in response rather than break. Glass surface and epoxy cell wall share the load in the synergy that is characteristic of a true composite.

To gain every last bit of strength, Emerson & Cuming has reduced almost to zero the remaining small percentage of unreinforced air in the resin – voids without glass reinforcement.

**Syntactic density**

Strength and density both relate to packing – the proportion of spheres to resin. In buoyancy, the end-result is the lowest density achievable, given a certain strength needed for the depth required, plus a wide margin of safety.

Emerson & Cuming’s data bank contains optimum proprietary formulations for specific service depths. Macrospheres are usually part of the mix. For extreme depths, however, or for manned submersibles, only microspheres are used. Lower density can be achieved at about the same strengths with the use of a binary system – two sizes of microspheres to improve the packing. Conversely, extremely low density materials for use in shallow water have been developed using a tertiary system –
microspheres with two sizes of macrospheres.

Over the past few years, Emerson & Cuming has pioneered in dramatically reducing densities to record levels. A syntactic for service to 10,000 ft. (3,048 m) now has a density of 28 pcf (0.45 S.G.), reduced from 34.5 pcf (0.55 S.G.) a short time ago. For shallow water use, a density of 16.5 pcf (0.26 S.G.) has been achieved in a true syntactic.

Product design
While syntactic composition and volume are being specified, we also CAD-design the optimum shape of the module and its segments, and the configuration of attachment points and accompanying hardware. Increasingly we are designing and fabricating complete assemblies.

Manufacture
Like the technical staff, the manu-

facturing team at Canton has benefitted from a long learning curve in processing syntactics and microspheres. The result is a finely-tuned process, utilizing unique equipment developed or modified for high quality production.

We closely control all major product and process factors. We developed and make our own microspheres and macrospheres and we write the formulas for our resins. The entire company operates on TQM philosophy, and the company is certified to ISO 9001. A portion of the production meets military specifications.

Managing the future
Along with our day-to-day projects for customers, Emerson & Cuming pursues long range studies in syntactic structure, density, integrity and processing.

With a modern plant, an experienced and motivated staff, and a record of innovative, successful structures and products, Emerson & Cuming is committed to maintaining its leadership position in syntactics worldwide into the next century.
Performance

The ocean challenge:
To survive one mile (1,610 m) deep for 30 years with negligible buoyancy loss.

A mile (1.6 km) down, pressure is more than one ton per square inch (16,200 kPa). When it comes to selecting deepwater ocean buoyancy, one material excels: syntactic. And in syntactics, one company excels. In 25 years, no Emerson & Cumings Eccofoat® buoyancy structure in service has ever suffered a catastrophic buoyancy loss.

Hollow cans or spheres, no matter how strong, risk immediate total failure if breached. Plastic foams, containing unreinforced air voids, are crushed usually within a few hundred feet of the surface.

A well-prepared syntactic, however, if it loses any significant buoyancy, does so gradually. Buoyancy does not depend on an external shell. An Eccofoat buoyant unit, or any section taken from it, demonstrates uniform, inherent strength throughout. Superficial damage has little effect on performance.

Every syntactic formulation, however, has its own failure threshold – its crush depth. To determine it, samples are tested to destruction. By general agreement, the industry rates syntactics for service depths far short of their crush depths. Both we and our customers continually ensure the manufacture and use of the buoyant unit only for the designated service depth.

When syntactics are tested to destruction, failure of a sample of microsphere/macrosphere/epoxy occurs in three stages. First, water pressure breaks the exposed single walls of the few macrospheres tangent to the surface. This minor damage stops here, because pressure then encounters only double walls consisting of the unbroken remainder of the broken macrosphere plus the walls of adjoining spheres. With greater pressure, however, water breaks through these double walls and floods the macrospheres. Next, the remaining epoxy structure collapses. An Ecohide skin is routinely specified, partly to provide an extra "wall" for the vulnerable macrospheres at the surface.

When syntactics are submerged, therefore subject to pressure, some small degree of buoyancy loss occurs. During the years when deepwater oil activity consisted mainly of exploration, the industry tested full scale riser modules by the weight gain method, which provided sufficient data to predict buoyancy for the relatively short and cyclical immersion periods required. In recent years, however, with emphasis shifting to production, the industry needs assurance of 10-, 20- and even 30-year survivability during continuous deployment at depth, with buoyancy losses typically less than 3-5% over the entire time span.

Only one method is sufficiently precise to allow projections out to 30 years with assurance. This is fully instrumented hydrostatic testing. To the best of our knowledge, Emerson & Cumings is the only syntactic facility in the world to have installed the necessary large, fully instrumented hydrostatic test vessels, in which we now routinely test full size deepwater modules or segments as they are manufactured.

Factors Affecting Hydrostatic Strength

- Formulation of resin for viscosity, bonding, wetting, chemical resistance and strength.
- Strengths of microspheres and macrospheres.
- Sizes and packing of spheres in matrix.
- Elimination of unreinforced air voids.
- Use of sodium-free microspheres to prevent water attraction and sphere-matrix disbonding.
- Similar sphere-matrix moduli to permit synergistic response to compression and minimize disbonding.
1. Segment is hoisted and fitted with weights.

2. Segment is lowered into test vessel filled with sea water.

3. Cap is lowered and bolted.

4. As pressure is applied, instrumentation records buoyancy loss at regular intervals.

5. Graph shows typical test results.

**Fully-instrumented Hydrostatic Testing**

A large deepsea buoyancy segment off the production line is tested at Emerson & Cuming's Canton, Massachusetts, facility by the extremely precise fully instrumented hydrostatic test method. This vessel, the largest of six at Emerson & Cuming, was specially designed and installed in 1992. It is a massive steel cylinder weighing 42.5 tons installed vertically in an underground concrete silo. The vessel can accept modules just under 66" (168 cm) wide and 18' (5.5 m) long, and can develop pressure equivalent to a service depth of 8,300 ft (2,540 m).
Appendix I

Specifications for Zinc Releases
GALVANIC RELEASE APPLICATIONS

There are many uses for the inexpensive galvanic releases we manufacture. You may have a specific application in mind at this time; however, for the future, here are some of the many uses for which we have designed and manufactured releases:

1) Release transmitters attached to tagged animals.
2) Trigger the uncovering and recovering of sediment traps.
3) Accurately assess the daily catch and yield of experimental fish traps by triggering their closure in exactly 24 hours.
4) Trigger underwater deployment of drogue-chutes after 20 minutes.
5) Submerge instrumentation marker buoys away from ships and ice for up to 35 days.
6) Serve as back-ups on expensive acoustic release packages.
7) Release emergency marker floats for location of lost equipment.

Don't be misled by the very simple design and low price of our devices. They have saved millions of dollars worth of equipment for customers ranging in size from Exxon Oil Company to poorly-funded researchers.

We sincerely hope our timed releases will help you save time, money, data and equipment.

Sincerely,

Andre’ LaBonte’, President

AL/rrl
Appendix K

Acknowledgments
Acknowledgments

The members of the design group for the oceanographic dropsonde would like to thank the following individuals for their assistance in the successful completion of the project:

Brian K. Dallas at Flotation Technologies, Inc., for sharing his knowledge of syntactic foam

Peter Williams, and Craig Quentin at Magellan Systems, for helping us with the GPS card

L. Petterson at Emerson and Cuming in Canton, Massachusetts, for his gracious donation of the required syntactic foam that was required for buoyancy of the dropsonde

Wayne Orcutt at International Fishing Devices, for helping us obtain the required zinc releases

Raymond B. Esteban at Impulse Enterprise, for his assistance in the selection of the proper through-hull connector for the pressure vessel

Art Knox at SeaCon, for his assistance in through-hull connection design

Carmen Serrano in DMES for her patience while we figured out the payment procedures

Dr. Graeme Rae in DMES for his assistance, and many suggestions leading to the completion of the project, and much more, but too much to list here... Thanks a bunch, Graeme!

Peter Lazarevich for his donation, and assistance in construction, and implementation of the Radio Direction Finding transmitter, and receiver for the dropsonde

Linda S. at McMaster-Carr Supply Co. for assisting us with our material order, and resulting return of extra material.

Dan Anderson at Campbell Scientific for his enormous assistance in the operation of the CR10 Data Logger unit

Dan Simpson, Lab Coordinator at Florida Tech, for his great deal of assistance in allowing us to use the machine shop, as well as assisting us, and providing us with ideas and improvements for the construction of the project
Fred Brisol in the Welding Shop at Florida Tech, for his patience, and superior job in satisfying our, off-the-beaten-path, requests

Chuck Inman in the Machine Shop at Florida Tech, for his expert craftsmanship in the construction of the pressure vessel

Andy Sherrell for his interest in, and rapport to, our inquires concerning the overall construction completion, as well as the pressure vessel design, and completion

Bill Battin, OE Instrumentation Lab Coordinator, for the great deal of assistance he offered us which is too much to list here... Thanks a bunch, Bill!!

Bill at HDI Aircraft, West Melbourne, for a quick weld while Fred Bristol was on vacation

Ryan Smith at the National Oceanic and Atmospheric Administration, who provided much information concerning the original dropsonde design developed by Mark Bushnell (also at NOAA-Miami), and who allowed us to assist him on a cruise out of West Palm Beach on the Baby Max where he was making dropsonde deployments... We owe you one, Ryan!

We would like to thank the following corporations who responded to our inquires concerning the original development of the unit, as well as provided raw materials leading to the construction of the dropsonde.

Builder’s Square, Palm Bay, Florida
E&B Marine, Melbourne Florida
Don Bell Steel, Melbourne, Florida
Florida Seal, and Rubber Co., Tampa, Florida
Emerson, and Cumming Composite Materials, Canton, Massachusetts
Radio Shack, Melbourne, Florida
Tedco Electronics, Melbourne, Florida
Furuno Marine Electronics, San Francisco, California
Balmoral Group Houston, Houston, Texas
Benthos, Falmouth, Massachusetts
Raytheon Marine Electronics, Manchester, New Hampshire
Syntech Materials, Alexandria, Virginia
West Marine, Eau Gallie, Florida
Appendix J

Lithium Battery Specifications
Lithium Battery Specifications

Manufacturer: Radio Shack
Division of the Tandy Corporation
Fort Worth, TX 76102

Voltage Output: 6 volts

Capacity: 1300 mA/h

Catalog Number: 23-178
Appendix L

References
References

for each satellite (up to 32). The health information is provided in three categories. The first one is that obtained from the broadcast almanac. The second is from the ephemeris that the unit collected for that satellite (note that most satellite’s ephemeris are not collected since only those being used for navigation are needed, the corresponding health would be "unavailable"). The last set of health information corresponds to that specified by the user through "Include/Exclude Sats" messages (PMGLU,04). The "Sats Data" message (PMGLU,03) provides detailed information about a given satellite.

Include/exclude satellites

The "Include/Exclude Sats" message (PMGLU,04) can be used to control satellite selection by the OEM software. With this message the user can control the use of satellites for position determination. The user can specify that certain satellites are NEVER to be used for position computation. Also the user can remove previous restrictions on the use of a satellite by returning it to its natural health. The format of the sentence would indicate that you can force include satellites. Through v1.3 software this function does not work. It will be functional in v1.4.

Note that the user cannot affect satellites which are not included in the units current almanac. Thus a satellite for which no almanac data is being broadcast can not be unconditionally included.

Almanac/ephemeris data

The "Almanac Data" and "Ephemeris Data" messages (T1 and T2, available in binary only) can be used to either retrieve almanac/ephemeris data from the unit or to load such information into the unit. The primary use of these messages would be to provide the unit with information so that it would not need to collect almanac or ephemeris data from the satellites. This would markedly improve time to first fix, especially in the case of a cold start. Because the ephemeris must be renewed every two hours it's not realistic to load the ephemeris over but it can be done.

Differential corrections

The "Differential Corrections" message (J0, binary only) enables the user to specify differential pseudorange corrections that will be used by the OEM software for position computation. The format and content of this message are based on the proposed RTCM standard. Note that a different message must be sent for each satellite whose ranges are to be corrected. Also the OEM software currently "times out" a correction after more than 200 seconds have expired past its time of validity.

Pseudo-continuous messages

Output of Raw Data, Doppler, C/N0, Almanac Data, and Ephemeris Data has a few special considerations. If a single output of these data messages is requested they will only be output if they are available. In general this is not a problem, but at unit start up, especially from cold-start, certain data may not be available for a time. If continuous output of the data is requested it will only be sent when it is updated. Thus almanac data may only be sent once, ephemeris every so often, etc.

If a single output of a data message is required at start up, when it may not be available, it is recommended that continuous output be initially requested. Then when the message is finally received it can be turned off.

SECTION II - OEM MESSAGES DEFINITIONS

PORT CONFIGURATION

Baud Rate options: 1200, 2400, 4800, 9600 baud.
Data Bits: 8
Parity: None
Stop bits: 1

There are four Eprom sets available with the OEM module. Each set has one of the default baud rates listed above. All of the sets will allow you to change the baud rate of operation to any of the listed above but on cold start it will come up in its default.

FORMATS

Both binary and ASCII formats are provided for most of the sentences. However, some sentences are binary format only, some sentences are ASCII format only.

a. BINARY

A sentence starts with "$$" (two HEX 24) followed by one byte of sentence identifier (from A to Z in ASCII) and one optional byte of sub-index (from 1 to 255 in binary) and followed by binary data field. All the binary data are integers with defined precision. The sentence is terminated with line feed (HEX 0A). One byte before the line feed is the checksum of the sentence. The checksum is calculated by XOR'ing the 8 binary
data bits of each valid byte in the sentence, between the "$\$" and the checksum. For example:

$$SR1000000000CL
xxxxxxxxxxx

("x" denotes data field to be xor'ed, "C" denotes checksum, "L" denotes line feed)

b. ASCII

The ASCII sentences conform to the NMEA 0183 software protocol (National Marine Electronics Association). Two types of sentences are used. Where the 0183 protocol has an already defined sentence for a certain data this sentence is used. These include GPXGA, GPGDA, GPVGTG, GPBOD, GPW6C, GPHDAM, GPHDT, GPAPR etc. For information that does not have a predefined sentence we used the method defined in the protocol for designing our own proprietary sentences so they would be compatible with the standard sentences. The proprietary sentences use 'P' for proprietary, 'MGL' for Magellan as manufacturer's identification and a primary index of 'A' through 'Z'. The first data field is a sub-index which has a range of from 00 to 99. The checksum is calculated by XOR'ing the 8 binary data bits of each byte in the sentence, between the "$\$" and the "". If you don't want to calculate the check sum leave out the "" and two following bytes.

Example with check sum:

$SPMLI,00,T01,1,B, *CKRL
xxxxxxxxxxxxxxxx

("x" denotes data field to be XOR'ed, 'CK' denotes 2 bytes checksum, 'R' denotes carriage return - hex 0D, 'L' denotes line feed - hex 0A)

Example without check sum:

$SPMLI,00,T01,1,B, RL
xxxxxxxxxxxxxxxxxx

TIME AND DATE

(Output from the board set)

<table>
<thead>
<tr>
<th>BINARY</th>
<th>ASCII</th>
</tr>
</thead>
<tbody>
<tr>
<td>$$SA000000XXCL</td>
<td>$GPXGA,xxxxx,xx,xx,xxxx,CKRL</td>
</tr>
<tr>
<td>123456</td>
<td>1 2 3 4</td>
</tr>
<tr>
<td>1: 1 byte, UTC hour</td>
<td>1: UTC, hhmmss</td>
</tr>
<tr>
<td>2: 1 byte, UTC minute</td>
<td>2: day</td>
</tr>
<tr>
<td>3: 1 byte, UTC second</td>
<td>3: month</td>
</tr>
<tr>
<td>4: 1 byte, day</td>
<td>4: year</td>
</tr>
<tr>
<td>5: 1 byte, month</td>
<td></td>
</tr>
<tr>
<td>6: 2 bytes, year</td>
<td></td>
</tr>
</tbody>
</table>

POSITION AND ALTITUDE

<table>
<thead>
<tr>
<th>BINARY</th>
<th>ASCII</th>
</tr>
</thead>
<tbody>
<tr>
<td>$$SB0000000000000CL</td>
<td>$GPXGA,xxxxx,xx,xx,xxxx,CKRL</td>
</tr>
<tr>
<td>1 2 3 4</td>
<td></td>
</tr>
<tr>
<td>1: 4 bytes, timetag in 1 second</td>
<td>1: UTC of position (hhmmss) &amp; time of signal used in calculation of position</td>
</tr>
<tr>
<td>2: 4 bytes, lat in 10^-7 degree</td>
<td>2: GPS Latitude</td>
</tr>
<tr>
<td>3: 4 bytes, lon in 10^-7 degree</td>
<td>3: Lat. N or S</td>
</tr>
<tr>
<td>4: 4 bytes, altitude in 0.01 meter/feet</td>
<td>4: GPS longitude</td>
</tr>
<tr>
<td>5: Lon. E or W</td>
<td>5: Lon. E or W</td>
</tr>
<tr>
<td>6: GPS Quality Indicator (0=GPS not available, 1=GPS available)</td>
<td>6: GPS Quality Indicator (0=GPS not available, 1=GPS available)</td>
</tr>
<tr>
<td>7: Number of sat being used,</td>
<td>7: Number of sat being used,</td>
</tr>
<tr>
<td>8: HDOP - recalce every 3 min</td>
<td>8: HDOP - recalce every 3 min</td>
</tr>
<tr>
<td>9-10: Antenna Height, Meters/feet</td>
<td>9-10: Antenna Height, Meters/feet</td>
</tr>
<tr>
<td>11-12: Geodidal Height, Meters/feet</td>
<td>11-12: Geodidal Height, Meters/feet</td>
</tr>
</tbody>
</table>
**POSITION ONLY**

**BINARY**

$$81xxxxxxxCL$$

1 2

1: 4 bytes, LAT in 10^-7
2: 4 bytes, lon in 10^-7

**ASCII**

$GPGLL, xxx.xN,x,xxx.xW*CKRL$

1 2 3 4

1: last fix latitude
2: lat N or S
3: last fix longitude
4: lon E or W

**GROUND COURSE AND VELOCITY**

**BINARY**

$$E0xxxxCL$$

1 2

1: heading(true), 0.01 Deg
2: velocity, 0.01 KNOTS, KM/HR, MPH

**ASCII**

$GPVTG, xxx.xT, xxx.xM, xxx.xN, xxx.xK*CKRL$

1 2 3 4 5 6 7 8

1-2: Heading, degrees, True
3-4: Heading, degrees, Mag.
5-6: Speed, Knots
7-8: Speed, Km./Hr.

**SATS ID PRN USED**

**BINARY**

$$F0xxxxCL$$

1

1: 1 byte each for 4 Sats PRN number

**ASCII**

$PMGLF, 00, xx, xx, xx*CKRL$

1 2 3

1-4: Sats PRN number

**SATS ID PRN USED DURING ACQUISITION OR EPHEMERIS COLLECT**

**BINARY**

$$F1xxxxxxxCL$$

12121212

4 sets of field 1 and 2 for 4 sats;

1: 1 byte each for sats PRN number.
2: 1 byte each for the sat status for the sat in field 1.
   0 = yet to be looked for or collected
   1 = now being looked for or collected
   2 = has already been found or collected
3: OEM receiver status (same as described in H00 message)

**ASCII**

$PMGLF, 01, xx, xx, xx, xx, xx, xx, xx*CKRL$

1 2 3
4 sets of field 1 and 2 for 4 sat:

1: 1 byte each for sat PRN number.
2: 1 byte each for the sat status for the sat in field 1.
   0 = yet to be looked for or collected
   1 = now being looked for or collected
   2 = has already been found or collected
3: OEM receiver status (same as described in H00 message)

**PDOP, GDOP, ERROR ESTIMATE**

**BINARY**

```
SSG0xxxxxxCL
1 2 3
```

1: 2 bytes, PDOP, 0.01
2: 2 bytes, GDOP, 0.01
3: 2 bytes, Error Estimate, meter

**ASCII**

```
$PMGLG,00,xxx.xx,xxx.xx,xxx*CRNL
1 2 3
```

1: PDOP
2: GDOP
3: Error Estimate, meter

**RECEIVER STATUS**

**BINARY**

```
$$H0xxxxxxxxxxxxCL
1234567890
```

1: software version number, 0.1
2: customer number
3: battery power for memory back-up (0=OK, 1=low power)
4: Oscillator (0=OK, 1=out of tune)
5: SQ (Signal quality number)
6: GQ (Geometric quality number) 0=OK 1=BAD
7: nav solution (0=Continuous, 1=interrupted)
8: Almanac data (0=OK, 1=no almanac data, 2=almanac is old)
9: Memory (0=memory ok to use, 1=lost memory data, need re-init)
10: OEM unit status: 0=INIT, 1=IDL, 2=STS, 3=ALM, 4=EPH, 5=ACQ, 6=POS, 7=NAV

**ASCII**

```
$PMGLH,00,xxx.xxx,xxx.xxx,xxx.xxx,*CRNL
1 2 3 4 5 6
```

Data fields definitions are the same as in binary format

**AUTOPILOT**

**BINARY**

```
$$RLxxxxxxxxxDL
12345678
```

1: 1 byte, OR'ed Blink and SNR; 1=Valid, 0=Invalid
2: 1 byte, Cycle Lock; 1=Valid, 0=Invalid
3: 2 bytes, Cross track error ; 0.01 N. Miles/KM/Stat. Miles
4: 1 byte, 0=steer right, 1=steer left
5: 1 byte, Arrival circle entered; 1=yes, 0=no
6: 1 byte, Perp. Crossed; 1=yes, 0=no
7: 2 bytes, bearing Dest. Wpt. from Origin Wpt. 0.1 degree, magnetic
8: 4 bytes, Waypoint identifier

**ASCII**

```
$GPAPA,A,A,x.xx,L,N,A,A,xxx,,M,cccc*CRNL
1 2 3 4 5 6 7 8
```

1: OR'ed Blink and SNR; A= Valid, V= Invalid
2: Cycle Lock; A= Valid, V= Invalid
3: -. Cross Track Error, Steer Left (L) or Right(R), N/Miles(N)/KM(K)/Stat. Miles(S) units
6: -. Arrival circle, Arrival Perpendicular (crossing of the Dest. Wpt.);
A= yes, V= No
8: -. Bearing Dest. Wpt. from Origin Wpt., Magnetic

**BEARING**

**BINARY**

```
$$R2xxxxxxxxxxxxxCL
```

1: 2 bytes, Bearing, true, 0.1 degree
2: 2 bytes, Bearing, Mag., 0.1 degree
3: 2 bytes, distance, 0.1 N. Miles/KM/Stat. Miles

**ASCII**

```
$GPBOD,xxx,,T,xxx,,M,cccc,cccc*CRNL
1 2 3 4 5 6
```

1-2: Bearing, true
3-4: Bearing, Mag.
5: Dest. Wpt. Identifier
6: Origin Wpt. Identifier
BEARING AND DISTANCE

**BINARY**

```plaintext
$SPRD, xxxxxxxx, xxxxx, xX, xX, xX, xX, xX
1234 5 6 7 8 9

1: 1 byte, UTC hour
2: 1 byte, UTC minute
3: 1 byte, UTC second
4: 4 bytes, Lat. of Wpt. 10^-7 degree
5: 4 bytes, Lon. of Wpt. 10^-7 degree
6: 2 bytes, Bearing, true, 0.1 degree
7: 2 bytes, Bearing, Mag., 0.1 degree
8: 2 bytes, distance, 0.1 N. Miles/KM/Stat. Miles
```

**ASCII**

```plaintext
$SPBD, xxxxxxxx, xxxxx, xX, xX, xX, xX, xX, xX

1 2 3 4 5

xxx.xT, xxx.xH, xxx.xN, cc+c
6 7 8 9 10 11 12

1: UTC of bearing
2-3: Lat., N or S of Wpt.
4-5: Lon. E or W of Wpt.
6-7: Bearing, true
8-9: Bearing, Mag.
```

MAGNETIC VARIATION

**BINARY**

```plaintext
$SPM, xX, xX
1

1: 2 bytes, variation, 0.1 degree
```

**ASCII**

```plaintext
$SPV, xX, xX

1: Variation, degrees
2: Variation sense, E or W
```

WAYPOINTS

**BINARY**

```plaintext
$SPW, xX, xX, xX
12
```

**ASCII**

```plaintext
$SPW, xX, xX, xX

1 2 3

1: Lat. degree
2: N or S
3: Lon. degree
4: E or W
5: Wpt. identifier
```

TIME TO GO

**BINARY**

```plaintext
$SPG, xX, xX, xX, xX, xX
1234567

1: 1 byte, UTC hour
2: 1 byte, UTC minute
3: 1 byte, UTC second
4: 1 byte, time to go hours
5: 1 byte, time to go minutes
6: 1 byte, time to go seconds
7: 4 bytes, destination wpt. identifier
```

**ASCII**

```plaintext
$SPG, xX, xX, xX, xX, xX

1 2 3

1: UTC in Hr. Min and Sec.
2: Time to go to waypoint (hh:mm:ss)
3: Waypoint identifier
```

ESTIMATED TIME OF ARRIVAL

**BINARY**

```plaintext
$SPR, xX, xX, xX, xX, xX
1234567

1: 1 byte, UTC hour
2: 1 byte, UTC minute
3: 1 byte, UTC second
4: 1 byte, estimate time of arrival, hour
5: 1 byte, estimate time of arrival, minute
6: 1 byte, estimate time of arrival, second
7: 4 bytes, destination wpt. identifier
```
ASCII
$GPZTA,xxxxx,xxxxx,CCCC*CRKL
1 2 3

1: UTC in Hr. Min and Sec.
2: Estimate Time of Arrival (hhmmss)
3: Waypoint identifier

ALMANAC DATA

BINARY
$$Tixxxxxx........xxxxxCL
1 (924 bytes in data field)

1: 924 bytes almanac message. Formatted as follows:
   bytes 1-24  almanac for SV 1
   bytes 25-26  week number for SV 1's almanac
   bytes 27-50  almanac for SV 2
   bytes 51-52  week number for SV 2's almanac
   bytes 781-804  almanac for SV 31
   bytes 805-806  week number for SV 31's almanac
   bytes 807-830  almanac for SV 32
   bytes 831-832  week number for SV 31's almanac
   bytes 833-853  healths from page 25 subframe 5
   bytes 854-877  healths from page 25 subframe 4
   bytes 878-900  special message from page 17 subframe 4
   bytes 901-924  iono and UTC param (page 18 subframe 4)

Each page of data consists of 24 bytes representing words 3 through 10 of the transmitted data with parity removed. The two exceptions are page 25 of subframe 5 which contains only words 3 through 9 of the data and page 17 of subframe 4 which does not include the least significant byte of word 10. Bytes are transmitted in the same order as from the satellite (most significant byte of word 3 is transmitted first for each page).

EPHEMERIS DATA

BINARY
$$T2xxxxx........xxxxxCL
12 (61 bytes in data field 2)

1: PRN number
2: 61 bytes ephemeris message. Formatted as follows:
   bytes 1-3  word 3 of subframe 1
   byte 4  lsb of word 7 of subframe 1
   bytes 5-7  word 8 of subframe 1
   bytes 8-10  word 9 of subframe 1
   bytes 11-13  word 10 of subframe 1
   bytes 14-37  words 3 through 10 of subframe 2
   bytes 38-61  words 3 through 10 of subframe 3

Each word contains the 24 data bits of the transmitted data (parity removed). The most significant byte of each word is transmitted first and the least significant byte is transmitted last.

SATS SCHEDULE

BINARY
$$Uixxxxxxxxxxxxxxxx...xCL
1 2 3 4 5 6

1: 4 bytes, lat. in 10^-7 deg.
2: 4 bytes, lon. in 10^-7 deg.
3: 1 byte, day
4: 1 byte, month
5: 2 bytes, year
6: 96 bytes, 24 hours sat schedule with 15 minutes window

ASCII
$PMGLU,01,A,xxxx.xx,N,xxxx.xx,W,xx,xx,xxxx*CRKL
1 2 3 4 5 6 7 8 9

$PMGLU,01,B,0011233332100000000000000000001112233456666777777*CRKL
10

$PMGLU,01,C,7888776655555544443333211111000000000000000000000000*CRKL
11

1: sub-index of sats data module
2: sequence number of this message
3: Lat.
4: Lat. N or S
5: Lon.
6: Lon. E or W
7: day
8: month
9: year
10: First 12 hours of sats schedule with 15 minutes window
    showing the number of available sats.
11: Second 12 hours of sats schedule

SATS HEALTH

BINARY
$$U2xxxxxxxxxxCL
1 2 3

1: 8 bytes, Almanac health status
2 bits for each sat, 01=healthy, 00=unhealthy

14
2: 8 bytes, Ephemera health status, 01=healthy, 00=unhealthy, not available
3: 8 bytes, User's health status, 01=healthy, 11=unhealthy, 00=default

ASCII
$PMGLU,02,A,xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx*CKRL
$PMGLU,02,B,xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx*CKRL
$PMGLU,02,C,xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx*CKRL

1: sub-index of sats data module
2: sequence number of this message, A=almanac, B=ephemeras, C=user
3: 1 character each for 32 sats health status, character"=" (hex 2D) represents '"1' as defined in binary format

SATS DATA

BINARY
$$U3xxxxxxxxxxCL
1234

1: 1 byte, PRN number of sats
2: 2 bytes, azimuth, 1 degree
3: 2 bytes, C/N0, in db-Hz, if the specified satellite is not tracked, the value will be 0.
4: 2 bytes, elevation angle, 1 degree

ASCII
$PMGLU,03,xx,xxx,xxx,xxx*CKRL
12345

1: sub-index of sats data module
2: PRN number of sats
3: azimuth, degree
4: C/N0, in db-Hz
5: elevation angle, degree

RAW DATA, DOPPLER, C/N0

BINARY
$$V1xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxCL
12341234123412345

4 sets of the data fields 1-4:
1: 4 bytes, doppler, in 0.01 cycles/sec
2: 1 byte, C/N0, in db-Hz

3: 1 byte, PRN
4: 4 bytes, time of transmission, in nsec.
5: 4 bytes, seconds into week, in 0.001 second

DATUM AND UNITS SETUP

BINARY
$$S1xxxxxxxxCL
123456

data fields are the same as described in ASCII format.

INPUT MESSAGES

(Into The Module)

TIME AND DATE

Same as output

POSITION AND ALTITUDE

Same as output except the only valid fields are L/L and altitude.

MODE

Same as output
DIFFERENTIAL CORRECTIONS

This message format will conform to the RTCM message type 1 format.

```
$S0bbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbCL
1 2 3 4 5 6
(*b* denotes 1 bit)
```

1: 3 bits, sets health summary and UDRE
2: 5 bits, sets ID
3: 16 bits, pseudorange correction, 0.02 - 0.64 meters
4: 8 bits, range rate correction, 0.004 - 0.128 m/s
5: 8 bits, issue of data
6: 16 bits, time of validity, 0-3599.4 sec, LSB = 0.075 sec, nominally bottom 3 bits are zeros as per RTCM.

MESSAGE ON/OFF, BINARY/ASCII SELECTION

ASCII

```
$PMGLI,00,xxx,x,x,xx*CRDL
12 3 4 5
```

1-2: 'A-Z' primary index for messages
2: '0-99' sub-index for messages
1&2: '100' indicates all messages.

3: 0= turns off output. If field 1 = 'A-Z' and field 2 = '00-99' then the individual sentence is turned off.
   If field 1&2 = '100' then all output is turned off and the previous configuration is retained, BUT a field 3 command of 1 with a field 1&2 command of '100' must be issued before any output can be obtained or the configuration can be altered again.
   1= Provided a field 3 command of '0' with field 1&2 = '100' was not the last command issued, if field 1 = 'A-Z' and field 2 = '00-99' then this command will produce a one time output for the selected sentence. If this particular sentence was turned on as continuous, it will be subsequently shut off. If the last command issued was field 3 command '0' with a field 1&2 = '100' then this command will turn on the configuration retained else it will be ignored.

4: A= ASCII format
   B= Binary format

5: PRN number if sat data or ephemeris data message is specified.

Note: There are no ASCII format messages for Almanac, Ephemeris and raw data messages. If continuous output selection is made for Almanac, Ephemeris or raw data messages, they will be output only when a new data set is collected instead of outputting at a fixed interval rate.

COLLECT ALMANAC

```
BINARY
$S0K0xxCL
12
```

1: 1 byte, 1 = collect almanac, 0 = don't collect almanac
2: 1 byte, Sat PRN number to collect almanac from

```
ASCII
$PMGLK,00,xx,xx*CRDL
1 2
```

1: 1 = collect almanac, 0 = don't collect almanac
2: Sat PRN number to collect almanac from

FORCE SATELLITE SELECTION

```
BINARY
$S0M0xCL
1
```

1: 1 byte, 1 = force satellite selection
   0 = don't force satellite selection
TOTAL NUMBER OF WAYPOINTS

BINARY

\$S89xCL

1

1: 1 byte, total number of waypoints defined.

ASCII

\$PMGLR,09,xxx*CKRL

1 2

1: Sub-index of waypoint module
2: Total number of waypoints defined.

CLEAR/RENAME THE WAYPOINT

BINARY

\$RAccccxxxxCL

12 3

1: 1 byte, Sub-index of waypoint module (hex 0a = dec 10)
2: 4 bytes, waypoint to be cleared or renamed.
3: 4 bytes, clear waypoint if "0000", rename if ascii.

ASCII

\$PMGLR,10,cccc,xxxx*CKRL

LFix = 1a+xFix

1: 1 byte, Sub-index of waypoint module.
2: 4 bytes, waypoint to be cleared or renamed.
3: 4 bytes, clear waypoint if "0000", rename if ascii.

ROUTES

BINARY

\$S8xxxxxxxxxxCL

12 3

1: 1 byte, route number
2: 4 bytes, from wpt. identifier
3: 4 bytes, to wpt. identifier

ASCII

\$PMGLR,08,xx,cccc,cccc*CKRL

1 2 3 4

1: sub-index of waypoint module
2: route number
3: from wpt. identifier
4: to wpt. identifier
MAGNETIC VARIATION

same as output (HVM only)

DATUM AND UNITS SETUP

BINARY

```plaintext
$SISPCC
123456
```

data fields are the same as described in ASCII format.

ASCII

```plaintext
$PMGLS,01,xx,xx,xx,xx,xx*CRCL
  1 2 3 4 5 6
```

1: sub-index of setup module
2: Terrain setting
   0= clear
   1= interrupted
   2= obstructed
3: Map datum 1-47 choices of map datum
4: Distance/Speed Units
   0= N. Miles, KNOTS
   1= KM, KM/HR
   2= Stat. Miles, MPH
5: Altitude Units
   0= Feet
   1= Meter
6: Magnetic Variation
   0= User entered
   1= Auto derived

ALMANAC DATA

BINARY

same as output

EPHEMERIS DATA

BINARY

same as output

INCLUDE/EXCLUDE SATS

BINARY

```plaintext
$SU4xxCC
123
```

Data fields are the same as described in ASCII format.
## OEM SOFTWARE MESSAGES

This is a quick reference to locate the different data sentence by ASCII, by BINARY, or by data content. The BINARY reference used here is the same as you would use in the PMGLI sentence to reference the ASCII or BINARY data sentence in the protocol. (B00 as opposed to B0.) The I/O indicates whether the sentence is both Input/Output, just Input, or just Output. Page is the location in this document.

### SORTED BY ASCII CODE

<table>
<thead>
<tr>
<th>ASCII</th>
<th>BINARY</th>
<th>FUNCTION</th>
<th>I/O</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>GFDX</td>
<td>A00</td>
<td>Time and Date</td>
<td>I/O</td>
<td>7</td>
</tr>
<tr>
<td>GFGA</td>
<td>B00</td>
<td>Position, Altitude etc.</td>
<td>I/O</td>
<td>7</td>
</tr>
<tr>
<td>GFGLL</td>
<td>B01</td>
<td>Position (Last Fix)</td>
<td>0</td>
<td>8</td>
</tr>
<tr>
<td>PMGLC0</td>
<td>C00</td>
<td>ECEF Position</td>
<td>0</td>
<td>8</td>
</tr>
<tr>
<td>PMGLD0</td>
<td>D00</td>
<td>Mode</td>
<td>I/O</td>
<td>8</td>
</tr>
<tr>
<td>GPVTG</td>
<td>E00</td>
<td>Ground Course and Velocity</td>
<td>0</td>
<td>9</td>
</tr>
<tr>
<td>PMGLP0</td>
<td>F00</td>
<td>Sats Used</td>
<td>0</td>
<td>9</td>
</tr>
<tr>
<td>PMGLF01</td>
<td>F01</td>
<td>Sats Used Including Status</td>
<td>0</td>
<td>9</td>
</tr>
<tr>
<td>PMGLG0</td>
<td>G00</td>
<td>PDOP, GDOP, Error Estimate</td>
<td>0</td>
<td>10</td>
</tr>
<tr>
<td>PMGLH0</td>
<td>H00</td>
<td>Receiver Status</td>
<td>0</td>
<td>10</td>
</tr>
<tr>
<td>PMGLI0</td>
<td>I00</td>
<td>Message On/Off, Binary/ASCII</td>
<td>I</td>
<td>18</td>
</tr>
<tr>
<td>PMGLK0</td>
<td>K00</td>
<td>Collect Almanac</td>
<td>I</td>
<td>18</td>
</tr>
<tr>
<td>PMGLM0</td>
<td>M00</td>
<td>Force Satellite Reselection</td>
<td>I</td>
<td>20</td>
</tr>
<tr>
<td>PMGLN0</td>
<td>N00</td>
<td>Master Reset</td>
<td>I</td>
<td>20</td>
</tr>
<tr>
<td>PMGLO0</td>
<td>O00</td>
<td>Change Baud Rate</td>
<td>I</td>
<td>20</td>
</tr>
<tr>
<td>GPAPA</td>
<td>R01</td>
<td>Autopilot</td>
<td>0</td>
<td>11</td>
</tr>
<tr>
<td>GPBOD</td>
<td>R02</td>
<td>Bearing (Dest and Orig Wypts)</td>
<td>0</td>
<td>11</td>
</tr>
<tr>
<td>GPBNW</td>
<td>R03</td>
<td>Bearing and Distance to Waypoint</td>
<td>0</td>
<td>12</td>
</tr>
<tr>
<td>GPGVD</td>
<td>R04</td>
<td>Magnetic Variation (derived)</td>
<td>0</td>
<td>12</td>
</tr>
<tr>
<td>GPVTD</td>
<td>R04</td>
<td>Magnetic Variation (manual)</td>
<td>I/O</td>
<td>12</td>
</tr>
<tr>
<td>GPWPL</td>
<td>R05</td>
<td>Waypoints</td>
<td>I</td>
<td>12</td>
</tr>
<tr>
<td>GPZTG</td>
<td>R06</td>
<td>Time To Go to Waypoint</td>
<td>O</td>
<td>13</td>
</tr>
<tr>
<td>GPZTA</td>
<td>R07</td>
<td>ETA to Waypoint</td>
<td>O</td>
<td>13</td>
</tr>
<tr>
<td>PMGLR08</td>
<td>R08</td>
<td>Routes</td>
<td>I</td>
<td>21</td>
</tr>
<tr>
<td>PMGLR09</td>
<td>R09</td>
<td>Number of Waypoints</td>
<td>I</td>
<td>21</td>
</tr>
<tr>
<td>PMGLR10</td>
<td>R10</td>
<td>Clear/Rename Waypoint</td>
<td>I</td>
<td>21</td>
</tr>
<tr>
<td>PMGLS01</td>
<td>S01</td>
<td>Datum, Terrain, and Units Setup</td>
<td>I/O</td>
<td>17</td>
</tr>
<tr>
<td>T01</td>
<td>I0</td>
<td>Almanac Data</td>
<td>I/O</td>
<td>14</td>
</tr>
<tr>
<td>T02</td>
<td>I1</td>
<td>Ephemeris Data</td>
<td>I/O</td>
<td>14</td>
</tr>
<tr>
<td>U01</td>
<td>C/No</td>
<td>Sat Schedule</td>
<td>0</td>
<td>15</td>
</tr>
<tr>
<td>PMGLU02</td>
<td>U02</td>
<td>Sat Health (Alm, Eph, User)</td>
<td>0</td>
<td>15</td>
</tr>
<tr>
<td>PMGLU03</td>
<td>U03</td>
<td>Sat Status</td>
<td>0</td>
<td>15</td>
</tr>
<tr>
<td>PMGLU04</td>
<td>U04</td>
<td>Include/Exclude Sats</td>
<td>I</td>
<td>22</td>
</tr>
<tr>
<td>V01</td>
<td>C/No</td>
<td>Doppler, C/No</td>
<td>0</td>
<td>16</td>
</tr>
</tbody>
</table>

### SORTED BY ASCII CODE

<table>
<thead>
<tr>
<th>ASCII</th>
<th>BINARY</th>
<th>FUNCTION</th>
<th>I/O</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>J00</td>
<td>T01</td>
<td>Differential Corrections</td>
<td>I</td>
<td>18</td>
</tr>
<tr>
<td>T01</td>
<td>T02</td>
<td>Almanac Data</td>
<td>I/O</td>
<td>14</td>
</tr>
<tr>
<td>T02</td>
<td>V01</td>
<td>Ephemeris Data</td>
<td>I/O</td>
<td>14</td>
</tr>
<tr>
<td>U01</td>
<td>C/No</td>
<td>Doppler, C/No</td>
<td>0</td>
<td>16</td>
</tr>
<tr>
<td>GPAPA</td>
<td>R01</td>
<td>Autopilot</td>
<td>0</td>
<td>11</td>
</tr>
<tr>
<td>GPBOD</td>
<td>R02</td>
<td>Bearing (Dest and Orig Wypts)</td>
<td>0</td>
<td>11</td>
</tr>
<tr>
<td>GPBNW</td>
<td>R03</td>
<td>Bearing and Distance to Waypoint</td>
<td>0</td>
<td>12</td>
</tr>
<tr>
<td>PMGLI0</td>
<td>I00</td>
<td>Position, Altitude etc.</td>
<td>I/O</td>
<td>7</td>
</tr>
<tr>
<td>PMGLD0</td>
<td>B01</td>
<td>Position (Last Fix)</td>
<td>0</td>
<td>8</td>
</tr>
<tr>
<td>GPGGA</td>
<td>B00</td>
<td>Magnetic Variation (derived)</td>
<td>0</td>
<td>12</td>
</tr>
<tr>
<td>GPVTD</td>
<td>R05</td>
<td>Magnetic Variation (manual)</td>
<td>1/O</td>
<td>12</td>
</tr>
<tr>
<td>GPWPL</td>
<td>R05</td>
<td>Ground Course, Velocity</td>
<td>0</td>
<td>9</td>
</tr>
<tr>
<td>PMGLD0</td>
<td>D00</td>
<td>Waypoints</td>
<td>I/O</td>
<td>12</td>
</tr>
<tr>
<td>PMGLF00</td>
<td>F00</td>
<td>Time and Date</td>
<td>I/O</td>
<td>7</td>
</tr>
<tr>
<td>PMGLF01</td>
<td>F01</td>
<td>ETA to Waypoint</td>
<td>0</td>
<td>13</td>
</tr>
<tr>
<td>PMGLG0</td>
<td>G00</td>
<td>Time To Go to Waypoint</td>
<td>0</td>
<td>13</td>
</tr>
<tr>
<td>PMGLH0</td>
<td>H00</td>
<td>ECEF Position</td>
<td>0</td>
<td>8</td>
</tr>
<tr>
<td>PMGLI0</td>
<td>I00</td>
<td>Mode</td>
<td>I/O</td>
<td>8</td>
</tr>
<tr>
<td>PMGLK0</td>
<td>K00</td>
<td>Sats Used</td>
<td>0</td>
<td>9</td>
</tr>
<tr>
<td>PMGLM0</td>
<td>M00</td>
<td>Sats Used Including Status</td>
<td>0</td>
<td>9</td>
</tr>
<tr>
<td>PMGLN0</td>
<td>N00</td>
<td>PDOP, GDOP, Error Estimate</td>
<td>0</td>
<td>10</td>
</tr>
<tr>
<td>PMGLO0</td>
<td>O00</td>
<td>Receiver Status</td>
<td>0</td>
<td>10</td>
</tr>
<tr>
<td>GPAPA</td>
<td>R01</td>
<td>Message On/Off, Binary/ASCII</td>
<td>I</td>
<td>18</td>
</tr>
<tr>
<td>GPBNW</td>
<td>R03</td>
<td>Collect Almanac</td>
<td>I</td>
<td>18</td>
</tr>
<tr>
<td>GPBNW</td>
<td>R03</td>
<td>Bearing and Distance to Waypoint</td>
<td>I</td>
<td>20</td>
</tr>
<tr>
<td>GPGVD</td>
<td>R04</td>
<td>Master Reset</td>
<td>I</td>
<td>20</td>
</tr>
<tr>
<td>GPAPA</td>
<td>R01</td>
<td>Force Satellite Reselection</td>
<td>I</td>
<td>20</td>
</tr>
<tr>
<td>GPAPA</td>
<td>R01</td>
<td>Change Baud Rate</td>
<td>I</td>
<td>20</td>
</tr>
<tr>
<td>GPAPA</td>
<td>R01</td>
<td>Autopilot</td>
<td>0</td>
<td>11</td>
</tr>
<tr>
<td>GPAPA</td>
<td>R01</td>
<td>Bearing (Dest and Orig Wypts)</td>
<td>0</td>
<td>11</td>
</tr>
<tr>
<td>GPAPA</td>
<td>R01</td>
<td>Bearing and Distance to Waypoint</td>
<td>0</td>
<td>12</td>
</tr>
<tr>
<td>GPGVD</td>
<td>R04</td>
<td>Magnetic Variation (derived)</td>
<td>0</td>
<td>12</td>
</tr>
<tr>
<td>GPAPA</td>
<td>R01</td>
<td>Magnetic Variation (manual)</td>
<td>I/O</td>
<td>12</td>
</tr>
<tr>
<td>GPAPA</td>
<td>R01</td>
<td>Waypoints</td>
<td>I</td>
<td>12</td>
</tr>
<tr>
<td>GPAPA</td>
<td>R01</td>
<td>Time To Go to Waypoint</td>
<td>O</td>
<td>13</td>
</tr>
<tr>
<td>GPAPA</td>
<td>R01</td>
<td>ETA to Waypoint</td>
<td>O</td>
<td>13</td>
</tr>
<tr>
<td>PMGLR08</td>
<td>R08</td>
<td>Routes</td>
<td>I</td>
<td>21</td>
</tr>
<tr>
<td>PMGLR09</td>
<td>R09</td>
<td>Number of Waypoints</td>
<td>I</td>
<td>21</td>
</tr>
<tr>
<td>PMGLR10</td>
<td>R10</td>
<td>Clear/Rename Waypoint</td>
<td>I</td>
<td>21</td>
</tr>
<tr>
<td>PMGLS01</td>
<td>S01</td>
<td>Datum, Terrain, and Units Setup</td>
<td>I/O</td>
<td>17</td>
</tr>
<tr>
<td>PMGLU01</td>
<td>U01</td>
<td>Almanac Data</td>
<td>I/O</td>
<td>14</td>
</tr>
<tr>
<td>PMGLU02</td>
<td>U02</td>
<td>Ephemeris Data</td>
<td>I/O</td>
<td>14</td>
</tr>
<tr>
<td>PMGLU03</td>
<td>U03</td>
<td>Sat Schedule</td>
<td>0</td>
<td>15</td>
</tr>
<tr>
<td>PMGLU04</td>
<td>U04</td>
<td>Sat Health (Alm, Eph, User)</td>
<td>0</td>
<td>15</td>
</tr>
<tr>
<td>PMGLU05</td>
<td>U05</td>
<td>Sat Status</td>
<td>0</td>
<td>15</td>
</tr>
<tr>
<td>PMGLU04</td>
<td>U04</td>
<td>Include/Exclude Sats</td>
<td>I</td>
<td>22</td>
</tr>
<tr>
<td>PMGLU04</td>
<td>U04</td>
<td>Doppler, C/No</td>
<td>0</td>
<td>16</td>
</tr>
<tr>
<td>ASCII</td>
<td>BINARY</td>
<td>FUNCTION</td>
<td>I/O</td>
<td>PAGE</td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
<td>---------------------------------</td>
<td>-----</td>
<td>------</td>
</tr>
<tr>
<td>PMLHL00</td>
<td>H00</td>
<td>Almanac Data OK?</td>
<td>0</td>
<td>10</td>
</tr>
<tr>
<td>T01</td>
<td></td>
<td>Almanac Data</td>
<td>I/O</td>
<td>14</td>
</tr>
<tr>
<td>PMLHL01</td>
<td>S01</td>
<td>Altitude Units</td>
<td>I/O</td>
<td>17</td>
</tr>
<tr>
<td>GPVGA</td>
<td>B00</td>
<td>Antenna Height</td>
<td>I/O</td>
<td>7</td>
</tr>
<tr>
<td>GPVGA</td>
<td>B01</td>
<td>Autopilot</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PMGLH03</td>
<td>U03</td>
<td>Azimuth of Sat</td>
<td>0</td>
<td>11</td>
</tr>
<tr>
<td>PMGLH00</td>
<td>H00</td>
<td>Continuous Fix</td>
<td>0</td>
<td>10</td>
</tr>
<tr>
<td>GPVOD</td>
<td>R02</td>
<td>Baud Rate, Change</td>
<td>I</td>
<td>20</td>
</tr>
<tr>
<td>GPVDC</td>
<td>R03</td>
<td>Bearing (Wpt to Wpt)</td>
<td>I</td>
<td>11</td>
</tr>
<tr>
<td>PMGL100</td>
<td>I00</td>
<td>Bearing (to Wpt Dest)</td>
<td>0</td>
<td>12</td>
</tr>
<tr>
<td>PMGL000</td>
<td>O00</td>
<td>Binary/ASCII</td>
<td>I</td>
<td>18</td>
</tr>
<tr>
<td>PMGLR00</td>
<td>U00</td>
<td>Change Baud Rate</td>
<td>0</td>
<td>15</td>
</tr>
<tr>
<td>PMGLR10</td>
<td>R10</td>
<td>Clear Waypoint</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PMGLN00</td>
<td>N00</td>
<td>Clear Memory, Reset</td>
<td>I</td>
<td>21</td>
</tr>
<tr>
<td>PMGLK00</td>
<td>K00</td>
<td>Collect Almanac</td>
<td>0</td>
<td>19</td>
</tr>
<tr>
<td>PMGLU03</td>
<td>U03</td>
<td>C/No</td>
<td>0</td>
<td>16</td>
</tr>
<tr>
<td>GPZDA</td>
<td>A00</td>
<td>Date</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PMLLS01</td>
<td>S01</td>
<td>Datum Select</td>
<td>I/O</td>
<td>7</td>
</tr>
<tr>
<td>GPBOD</td>
<td>R02</td>
<td>Destination Waypoint</td>
<td></td>
<td></td>
</tr>
<tr>
<td>J00</td>
<td></td>
<td>Differential Corrections</td>
<td>0</td>
<td>11</td>
</tr>
<tr>
<td>GPBNC</td>
<td>R03</td>
<td>Distance to Waypoint</td>
<td>I</td>
<td>18</td>
</tr>
<tr>
<td>PMGLS01</td>
<td>S01</td>
<td>Distance Units</td>
<td>I</td>
<td>12</td>
</tr>
<tr>
<td>V01</td>
<td></td>
<td>Doppler</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PMGLC00</td>
<td>C00</td>
<td>ECEF Position</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PMGLU03</td>
<td>U03</td>
<td>Elevation of Sat</td>
<td>0</td>
<td>15</td>
</tr>
<tr>
<td>PMGLG00</td>
<td>G00</td>
<td>Ephemeris Data</td>
<td>I/O</td>
<td>14</td>
</tr>
<tr>
<td>GPZTA</td>
<td>R07</td>
<td>Error Estimate</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PMGLU04</td>
<td>U04</td>
<td>ETA to Waypoint</td>
<td>I</td>
<td>13</td>
</tr>
<tr>
<td>PMLMG00</td>
<td>M00</td>
<td>Exclude Sats</td>
<td>I</td>
<td>22</td>
</tr>
<tr>
<td>PMGLG10</td>
<td>I00</td>
<td>Force Satellite Reselection</td>
<td></td>
<td></td>
</tr>
<tr>
<td>GPVGA</td>
<td>B00</td>
<td>Frequency of Message</td>
<td>I</td>
<td>18</td>
</tr>
<tr>
<td>PMGLG00</td>
<td>G00</td>
<td>GDOP</td>
<td>0</td>
<td>10</td>
</tr>
<tr>
<td>GPVGA</td>
<td>B00</td>
<td>Geocidal Height</td>
<td>I/O</td>
<td>7</td>
</tr>
<tr>
<td>GPVGA</td>
<td>B00</td>
<td>GPS Quality</td>
<td>I/O</td>
<td>7</td>
</tr>
<tr>
<td>PMGLH00</td>
<td>H00</td>
<td>GQ</td>
<td>0</td>
<td>10</td>
</tr>
<tr>
<td>GPVRC</td>
<td>E00</td>
<td>Ground Course</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>GPVGA</td>
<td>B00</td>
<td>HDOP</td>
<td>I/O</td>
<td></td>
</tr>
<tr>
<td>GPVGA</td>
<td>B00</td>
<td>Heading</td>
<td>0</td>
<td>9</td>
</tr>
<tr>
<td>PMGLU02</td>
<td>U02</td>
<td>Health (Alm, Eph, User)</td>
<td>0</td>
<td>15</td>
</tr>
<tr>
<td>PMGLU04</td>
<td>U04</td>
<td>Include Sats</td>
<td>I</td>
<td>22</td>
</tr>
<tr>
<td>GPVDC</td>
<td>R03</td>
<td>Last Fix</td>
<td>I</td>
<td>8</td>
</tr>
<tr>
<td>GPVDC</td>
<td>R03</td>
<td>Lat of Waypoint</td>
<td>0</td>
<td>12</td>
</tr>
<tr>
<td>PMGLS01</td>
<td>S01</td>
<td>Lon of Waypoint</td>
<td>0</td>
<td>12</td>
</tr>
<tr>
<td>GPHVD</td>
<td>R04</td>
<td>Magnetic Variation (User vs Auto)</td>
<td>I/O</td>
<td>17</td>
</tr>
<tr>
<td>GPHVH</td>
<td>R04</td>
<td>Magnetic Variation (Derived)</td>
<td>0</td>
<td>12</td>
</tr>
<tr>
<td>GPHVH</td>
<td>R04</td>
<td>Magnetic Variation (Manual)</td>
<td>I/O</td>
<td>12</td>
</tr>
</tbody>
</table>
Appendix E

Specifications for Data Logger and Related Software
15 CONTROL PORT SERIAL

1. SPECIFICATIONS

FUNCTION
Send/receive half duplex serial data through the CR10 control ports.

INPUT
Data sent by the sensor can be ASCII data values, hexadecimal pairs in ASCII representation, or decimal encoded binary bytes (Section 3, Parameter 2).

OUTPUT
Output can be a preamble set of commands to the sensor or data from the CR10's input locations (Section 3, Parameter 6).

BAUD RATES
1200 baud
300 baud

HANDSHAKE CONTROL LINES
REQUEST TO SEND or DATA TERMINAL READY (RTS/DTR)
The CR10 signals the sensor that it is ready to send or receive data. This line is always asserted at the start of the instruction. If no output is specified then the line acts like DTR and remains asserted until all input is received. If output is specified, the line acts like RTS, asserted during output, not asserted while receiving (Section 3, Parameter 4).

CLEAR-TO-SEND (CTS)
The sensor signals the CR10 that it is ready to receive the preamble or data (Section 3, Parameters 3 and 9).

LOGIC LEVELS
- RS232 - Logical 1 is low voltage.
  RTS/DTR asserted is high voltage.
- TTL - Logical 1 is high voltage.
  RTS/DTR asserted is low voltage.

PARITY AND STOP BITS
RECEIVE
- ASCII and Hexadecimal Pairs in ASCII Representation:
  1 start bit, 8 data bits, no parity,
  1 stop bit 8th bit ignored
- Binary to decimal equivalent:
  1 start bit, 8 data bits, no parity,
  1 stop bit

TRANSMIT
1 start bit, 8 data bits, no parity, 1 stop bit

2. SELECTED OPERATING DETAILS
Standard RS232 logic levels range from 3 to 25 volts and -3 to -25 volts. The maximum and minimum input to CR10 control ports is 5 and 0 volts, respectively. Inputs exceeding these limits may damage the CR10. Figure 1 gives a circuit which may be used to limit voltage levels to 0 to 5 volts. Each input (data and handshake) that is out of the 0 to 5 volt range requires one of these circuits, or equivalent.

![Circuit Diagram]

FIGURE 1. Circuit To Limit Input to 0 to 5 Volts

March 19, 1991
COPYRIGHT (c) 1988, 1991 CAMPBELL SCIENTIFIC, INC.
The CR10 consists of a Measurement and Control Module and a detachable Wiring Panel. The Keyboard Display is recommended for on-site communication, station setup, and troubleshooting, but may be replaced by a computer where environmental conditions allow.

12 Analog Inputs - (single-ended channels, each pair differential*) Five software selectable input voltage ranges. Resolution is 0.25 mV on the 2.5 V range (-0.006°C on type E thermocouple). Multiplexers provide additional inputs. AG terminals are for analog ground connections.

3 Switched Excitation Channels for precision excitation of sensors or short-term actuation of external devices. Excitation is programmable over a ±250 V mV range.

5 V terminals provide power to some peripherals.

Power and Ground Connections for 12 V external batteries or peripherals. Switched 12 V terminal is controlled by any digital output.

9-Pin Serial I/O Port for connection of data storage, retrieval, and telecommunications peripherals.

8 Digital Inputs/Outputs for output control, sensing status, and reading SDM peripherals or SDH-12 sensors.

2 Pulse Counting Channels are software selectable for switch closures, high frequency pulses, or low level AC measurement.

MEASUREMENT AND CONTROL MODULE

Protected in a sealed, rugged, stainless steel cannister, the programmable module provides sensor measurement, timekeeping, communication, data reduction, data/program storage and control functions. A multi-tasking operating system allows simultaneous communication and measurement functions. Operating temperature range is -25°C to +50°C, standard; -55°C to +85°C, on request.

The standard instruction set includes 30 measurement, 43 processing/math, and 15 program control instructions. Optional instructions are available for specialized measurement or processing capabilities.

The standard memory stores 29,900 data points in two Final Storage areas. Solid-state or SRAM card storage modules provide additional on-site data storage.

The Measurement and Control Module interfaces with the Wiring Panel via two D-style connectors. The CR10's electronics are RF shielded and glitch protected by the sealed, stainless steel packaging. A "watch-dog" hardware reset function restores normal microprocessor function if lost due to an input transient or intermittent component failure.

WIRING PANEL

The Wiring Panel consists of a top panel, end bracket, and baseplate. The top panel includes screw terminals for sensor connections and a 9-pin serial I/O port; the end bracket attaches the Wiring Panel to the Control Module and to an enclosure-mounted or free-standing baseplate. The Control Module easily disconnects from the Wiring Panel allowing field replacement without rewiring the sensors. All wiring panel connections are protected with spark gaps or transzors.

CR10KD KEYBOARD/DISPLAY

The portable CR10KD programs the CR10, manually initiates data transfer, and displays sensor readings, stored values, or flag/port status. One CR10KD may be carried from station to station in a CR10 network. The CR10KD features an 8-character LCD and a 16-character keyboard. Operating temperature range is -25°C to +50°C. The CR10KD is powered by the CR10's power supply.

SC12 AND SC12R CABLES

The SC12 ribbon cable (included) or the SC12R, a rugged temperature-resistant cable that is purchased separately, connect peripherals or interfaces to the CR10's serial port.

CR10TCR THERMOCOUPLE REFERENCE

The CR10TCR thermistor provides a temperature reference for thermocouple measurements. It requires one single-ended analog input. An aluminum cover to reduce temperature gradients along the input terminals is included.

PERIPHERALS

The CR10 is powered by a 9.6 to 16 VDC supply and housed in a weather-resistant enclosure (page 8). Measurement, control, and data storage/transfer peripherals are optional depending on the application (pages 4 and 5).

* Differential measurements measure the voltage difference between two inputs. Single-ended measurements measure the inputs with respect to ground. All inputs must be within the ±2.5 V common mode range.